

Wavelets en Elektromagnetische Modelling van Vertical Cavity Surface Emitting Lasers

Michaël Van Damme

Samenvatting

Dit afstudeerwerk bestudeert de mogelijkheid om de elektromagnetische veldverdeling in de caviteit van vertical cavity surface emitting lasers te berekenen met behulp van wavelets. We bestuderen in detail hoe operatoren kunnen geconstrueerd worden die werken op de waveletexpansie van functies. We bespreken zowel de standaard als de non-standaard vorm van deze operatoren in één en twee dimensies. We demonstreren in de praktijk dat ze het mogelijk maken om simpele elektromagnetische problemen numeriek te benaderen. Complexere problemen stellen echter specifieke problemen. De grootste moeilijkheid bij praktische implementaties is de complexiteit van het uitvoeren van punts-gewijze vermenigvuldigingen in waveletbasis. Zolang deze vermenigvuldigingen niet efficiënt kunnen berekend worden blijft de nodige computerinfrastructuur voor het vinden van de elektromagnetische veldverdeling in een vertical cavity surface emitting laser met de in dit werk bestudeerde technieken erg groot.

Abstract

This master's thesis studies the possibility to calculate the electromagnetic field profiles inside the cavity of vertical cavity surface emitting lasers using wavelets. We study in detail how operators can be constructed that act on wavelet expansions of functions. We describe both the standard and non-standard forms of these operators in one and two dimensions. We demonstrate that they make it possible to approximate numerically the solution of simple electromagnetic problems. More complex problems, however, cause specific difficulties. The biggest problem for practical implementations is the complexity of performing pointwise multiplications in wavelet basis. As long as we cannot efficiently calculate these pointwise multiplications, the necessary computing power needed for determining the electromagnetic field distribution within a vertical cavity surface emitting laser using the methods studied in this master's thesis remains large.

Résumé

Dans ce mémoire on étudie la possibilité d'appliquer l'analyse à ondelettes pour calculer les champs électromagnétiques dans la cavité d'un microlaser à émission verticale. Nous étudions en détail la construction des opérateurs qui agissent sur des fonctions écrites en base d'ondelettes. Nous décrivons les formes standard et non standard de ces opérateurs dans une et deux dimensions. Nous démontrons qu'elles permettent de calculer approximativement la solution des problèmes électromagnétiques simples. Des problèmes plus complexes, cependant, ont des difficultés spécifiques. Le plus grand problème pour des réalisations pratiques est la complexité de multiplier point par point dans la base d'ondelettes. Aussi longtemps que nous ne pouvons pas efficacement calculer ces multiplications, les ressources de calcul nécessaires pour déterminer la distribution des champs électromagnétiques restent grands.

Woord vooraf

In dit woord vooraf wil ik een aantal mensen bedanken die hebben bijgedragen tot dit afstudeerwerk. Eerst en vooral heel veel dank aan Ir. Michael Peeters, zonder wiens aanstekelijke enthousiasme, volledige inzet en uniek talent om te zorgen dat de dingen werken dit project er nooit was geweest. Dank aan Prof. Irina Veretennicoff die me de kans heeft gegeven dit afstudeerwerk te maken. Veel dank ook aan Prof. Christine Demol (ULB) die de weg wees in het doolhof van de wavelets, altijd bereid was vragen te beantwoorden en veel nuttige opmerkingen heeft gegeven over de tekst. Tenslotte dank aan Prof. L. Van Hamme voor het vertalen van een aantal Engelstalige begrippen die in de literatuur over wavelets voorkomen.

Inhoudsopgave

Woord vooraf	ii
1 Inleiding	1
2 Vertical cavity surface emitting lasers	3
2.1 VCSEL structuren	3
2.1.1 De proton geïmplanteerde VCSEL	4
2.1.2 De air-post VCSEL	5
2.1.3 De oxide-confined VCSEL	5
2.2 VCSEL vs Edge Emitting Laser	7
2.3 Numerieke simulatie van VCSEL's	8
2.3.1 Specifieke problemen	8
2.3.2 Overzicht van mogelijke oplossingsmethodes	9
2.4 Besluit	11
3 Wavelets en multiresolutie analyse	12
3.1 Decorrelatie van functiesamples	12
3.2 Wavelets	15
3.2.1 Tijd-frequentie analyse	15
3.2.2 Tijd-frequentie en tijd-schaal voorstellingen	18
3.2.3 De continue wavelettransformatie	24
3.3 Multiresolutie analyse	28
3.3.1 Multiresolutie benaderingen	29
3.3.2 De snelle wavelet transformatie	40

3.3.3	Eindig aantal coëfficiënten	44
3.3.4	Keuze van de wavelet basis	44
3.4	Besluit	47
4	Wavelets en operatoren	48
4.1	Tweedimensionale wavelets	48
4.1.1	Standaard tensor-product basis	49
4.1.2	Separabele multiresoluties	49
4.2	Lineaire operatoren in wavelet basis	50
4.2.1	Standaard vorm operatoren	52
4.2.2	Non-Standaard vorm operatoren	60
4.2.3	Enkele eigenschappen van de projectieoperatoren P_j en Q_j	66
4.2.4	Herprojectie na non-standaard vorm matrix-vector vermenigvuldiging	68
4.2.5	De operator $\frac{d}{dx}$	71
4.2.6	Combinatie van twee non-standaard vorm operatoren	74
4.3	Meerdere dimensies	82
4.3.1	Non-standaard vorm operator voor tweedimensionale functies	82
4.3.2	Standaard vorm operator voor tweedimensionale functies	88
4.3.3	Separabele operatoren	93
4.3.4	Tensor-matrix en tensor-tensor produkten	96
4.4	Toepassing: vermenigvuldigen in waveletbasis	97
4.5	Besluit	99
5	Numerieke simulaties	102
5.1	Wavelet-gebaseerde technieken voor numerieke elektromagnetische bereke- ningen	102
5.1.1	Methodes gebaseerd op de Green's functie	102
5.1.2	De multiresolutie tijdsdomein methode	103
5.2	De methode van Werthen & Wolff	104
5.2.1	Eén dimensie	106
5.2.2	Twee dimensies	109
5.2.3	Drie dimensies	119
5.3	Besluit	124

INHOUDSOPGAVE

v

6 Besluit	125
A Broncode	127
B Afkortingen	148
C Engelstalige termen	149
D Notaties	150
Bibliografie	151

Hoofdstuk 1

Inleiding

In dit afstudeerwerk bestuderen we het gebruik van wavelets bij numerieke elektromagnetische simulaties. Het is de bedoeling te onderzoeken of er op wavelets gebaseerde technieken bestaan die het mogelijk maken om de elektromagnetische veldverdeling in de caviteit van een vertical cavity surface emitting laser (VCSEL) efficiënt te berekenen.

Dit type laser stelt de meer traditionele numerieke methoden zwaar op de proef. De kleine gelokaliseerde structuren die voorkomen in sommige VCSEL-caviteiten en het grote aantal materiaallaagjes waaruit zulke lasers bestaan vereisen namelijk het gebruik van een bijzonder fijn discretisatierooster. Het berekenen van de veldverdeling in een ééndimensionale benadering van de VCSEL is in de praktijk niet zo interessant. Het staande golfprofiel dat men dan vindt kan immers nauwkeurig berekend worden met analytische technieken (matrixformalisme). Een tweedimensionale benadering kan nuttig zijn voor VCSEL's waarin transversaal geen grote brekingsindexverschillen voorkomen. Dit is zo bij bv. de proton geïmplanteerde VCSEL. Voor vele interessante en nieuwe VCSEL's zoals VCSEL's met één of meerdere oxide-aperturen voldoet de tweedimensionale benadering niet meer en moet de laser beschouwd worden als een driedimensionaal systeem. Dit geeft aanleiding tot een enorm groot aantal discretisatiepunten. De meeste modellen voor het simuleren van VCSEL's zijn daarom of benaderingen die slechts in een beperkt aantal gevallen bruikbaar zijn of leggen heel zware eisen op aan de gebruikte computerinfrastructuur (zowel qua rekenkracht als geheugen).

In dit afstudeerwerk bekijken we of het mogelijk is VCSEL's efficiënter te simuleren met wavelet-gebaseerde technieken dan met de traditionele methodes. Er zijn twee belangrijke en verwante factoren die dit mogelijk zouden kunnen maken:

1. De compressie eigenschappen van wavelets kunnen misschien gebruikt worden om het aantal discretisatiepunten waar we de waarden van de elektromagnetische velden moeten bijhouden te reduceren. Misschien is het mogelijk om met wavelets een meer 'economische' voorstelling van de velden te bekomen.

2. Een meer economische voorstelling van de elektromagnetische velden zou ook het aantal bewerkingen die nodig zijn om het dynamisch gedrag van de velden te berekenen kunnen verminderen.

De voornaamste moeilijkheid bij het gebruik van wavelets voor elektromagnetische simulaties is het rekenen met de waveletvoorstelling van de velden. Simpele bewerkingen als de puntsgewijze vermenigvuldiging van twee functies worden heel wat ingewikkelder als men ze wil uitvoeren met de waveletvoorstellingen van deze functies.

We zullen beginnen met in het volgende hoofdstuk een korte inleiding te geven over vertical cavity surface emitting lasers en de typische moeilijkheden zijn die men moet overwinnen om de velden in een VCSEL te berekenen. In hoofdstuk 3 zullen we uitleggen wat wavelets juist zijn en hoe we snel wavelettransformaties kunnen berekenen. Hoofdstuk 4, Wavelets en operatoren, is het belangrijkste hoofdstuk en bespreekt hoe waveletvoorstellingen van één- en meerdimensionale functies kunnen bewerkt worden door middel van operatoren in waveletbasis. Tenslotte zullen we in hoofdstuk 5 een prototype-implementatie bespreken van een wavelet gebaseerde methode en uitleggen welke problemen daarbij opduiken.

Hoofdstuk 2

Vertical cavity surface emitting lasers

In dit hoofdstuk zullen we kort de vertical cavity surface emitting laser (VCSEL) bespreken. We zullen de meest courante types bekijken en kort ingaan op hun bijzonderheden. Daarna zullen we de VCSEL vergelijken met de meer courante edge emitting laser (EEL).

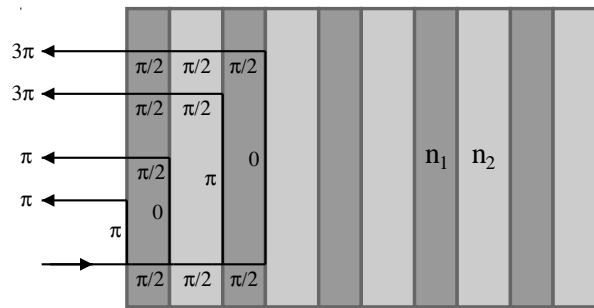
Tenslotte zullen we uitleggen wat de voornaamste moeilijkheden zijn bij het numeriek berekenen van de elektromagnetische veldverdeling binnen VCSEL caviteiten.

2.1 VCSEL structuren

De VCSEL is een type halfgeleiderlaser waarbij de lichtbundel loodrecht op het substraat uitgezonden wordt (zie figuren 2.2, 2.3 en 2.4). De lasercaviteit wordt gevormd door twee gedistribueerde Bragg reflectoren (distributed Bragg reflector of DBR) met daartussen een actieve laag (met kwantumputten of 'quantum wells'), waarin het licht door gestimuleerde emissie versterkt wordt.

Een DBR bestaat uit een aantal laagjes met alternerende brekingsindices n_1 en n_2 . De laagjes met brekingsindex n_1 hebben dikte $\lambda_{vac}/4n_1$, degenen met brekingsindex n_2 zijn $\lambda_{vac}/4n_2$ dik. Hierbij is λ_{vac} de golflengte (in vacuüm) waarvoor de DBR (en de VCSEL) ontworpen is. In figuur 2.1 is een DBR schematisch afgebeeld. Bij stationaire laserwerking moeten de verliezen die optreden aan de spiegels en in de caviteit zelf exact gecompenseerd worden door de versterking in de actieve laag. Omdat bij rondgang in een VCSEL het licht maximaal over enkele tientallen nanometer versterkt wordt moet de reflectiviteit van de DBR's zeer hoog zijn ($> 99\%$). Dit kan enkel bereikt worden door veel periodes van laagjes met alternerende brekingsindices te gebruiken en door het brekingsindexverschil $|n_1 - n_2|$ zo groot mogelijk te maken. Typisch gebruikt men in VCSEL's gebaseerd op het GaAs/AlAs materiaalsysteem (voor golflengtes tussen 780 en 980 nm) 20 à 30 lagenparen.

Er zijn zeer veel verschillende VCSEL structuren mogelijk. De meeste zijn cilindrisch-symmetrisch. We zullen er enkele kort bespreken.



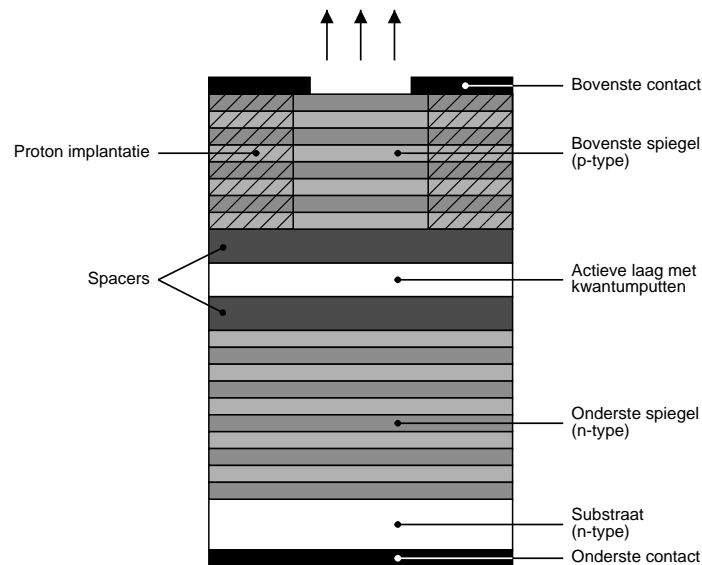
Figuur 2.1: Gedistribueerde Bragg reflector (DBR) bestaande uit laagjes met brekingsindices n_1 en n_2 met $n_1 > n_2$. Bij doorgang door een laagje draait de fase van licht met vacuüm golflengte λ_{vac} over $\pi/2$. Een bundel met dezelfde golflengte die zich in materiaal met brekingsindex n_1 bevindt ondergaat bij reflectie op een $n_1|n_2$ oppervlak een faseslag van π . Deze twee effecten zorgen ervoor dat de verschillende reflecties constructief interfereren zodat een grote reflectiecoëfficiënt bekomen wordt.

2.1.1 De proton geïmplanteerde VCSEL

Een doorsnede van een mogelijke VCSEL structuur is afgebeeld in figuur 2.2. De bovenste DBR (of bovenste spiegel) is bijna altijd in p-type materiaal uitgevoerd, de onderste in n-type materiaal, zodat de VCSEL structuur een diode vormt. Men maakt het bovenste contact meestal ringvormig zodat de laserbundel langs daar de caviteit kan verlaten. Het nadeel hiervan is dat het gebied van de actieve laag dat zich onder de opening in het contact bevindt maar weinig gepompt wordt. Daarom wordt in de VCSEL structuur een current confinement mechanisme ingebouwd. Dat moet ervoor zorgen dat vooral het nuttige gebied van de actieve laag gepompt wordt. Bij de VCSEL afgebeeld in figuur 2.2 werd dit gedaan door het deel van de bovenste spiegel waarboven het ringvormig contact zich bevindt met protonen te bombarderen. De proton-geïmplanteerde gebieden zijn elektrisch niet geleidend, zodat de stroom door het nuttige deel van de actieve laag gedwongen wordt. Het implanteren van protonen verandert de brekingsindex van het materiaal nauwelijks.

Het voornaamste nadeel van de proton geïmplanteerde VCSEL is het ontbreken van een goed golfgeleidingsmechanisme, zodat diffractie in de caviteit bijna niet tegengegaan wordt. Dit resulteert in een hoge drempelstroom. Licht wordt enkel versterkt in delen van de actieve laag die gepompt worden, dus ruwweg onder de opening in het ringvormig contact. Het materiaal zal op die plaatsen dus opwarmen (Joule effect), wat een kleine brekingsindexverandering veroorzaakt en waardoor er thermal lensing ontstaat.

De meeste VCSEL's die vandaag commercieel beschikbaar zijn zijn van het proton geïmplanteerde type.



Figuur 2.2: Schematische weergave van een proton geïmplanteerde VCSEL (doorsnede).

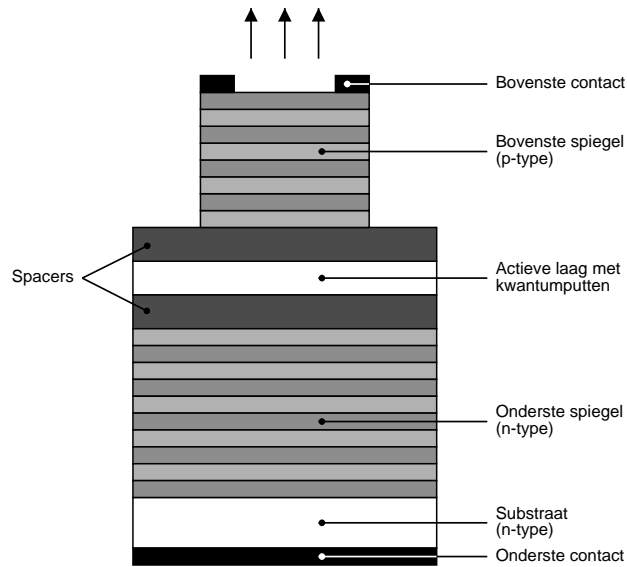
2.1.2 De air-post VCSEL

Een doorsnede van een air-post VCSEL is afgebeeld in figuur 2.3. De current confinement van de structuur is niet erg goed. Er is immers geen enkel mechanisme dat ervoor zorgt dat de stroom door het midden van de caviteit gaat lopen.

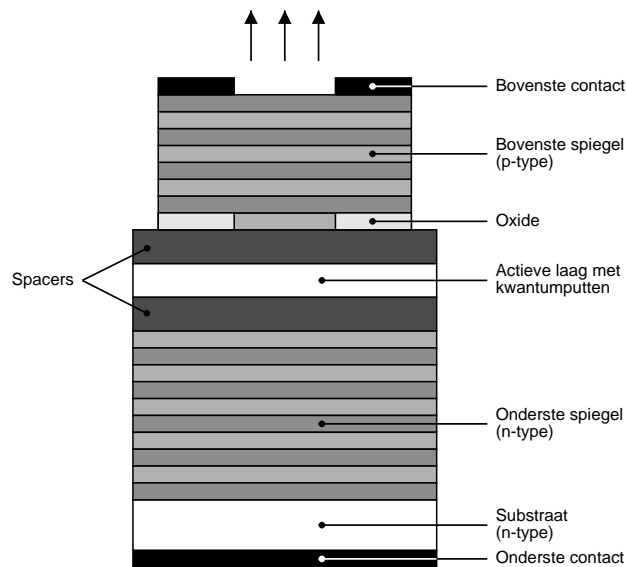
In de air-post VCSEL is er optical confinement veroorzaakt door het grote brekingsindexverschil tussen het halfgeleidermateriaal en de omringende lucht in de bovenste spiegel. Nadeel van de structuur zijn de grote verstrooiingsverliezen veroorzaakt door de oppervlakteruwheid van het manteloppervlak van de bovenste spiegel. Ook de hoeken tussen de losstaande bovenste spiegel en het onderste deel van de structuur verstrooien het laserlicht. Verder vermindert niet-stralende hercombinatie aan de buitenwand van de bovenste spiegel de efficiëntie [1].

2.1.3 De oxide-confined VCSEL

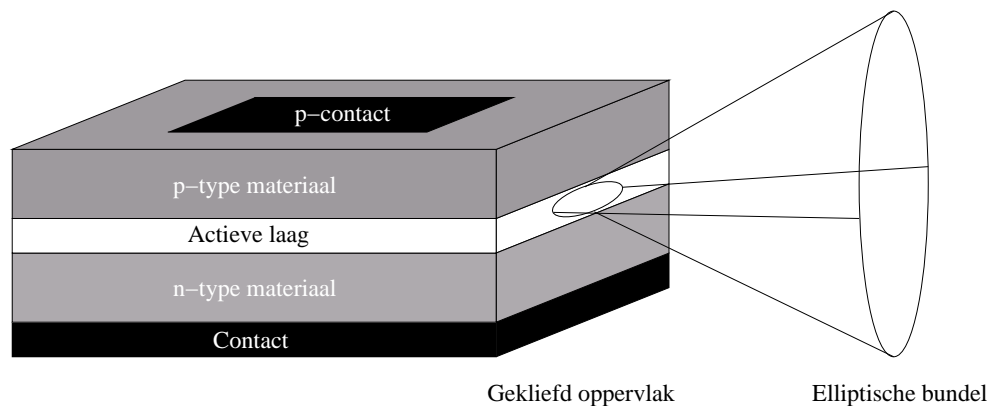
Een andere mogelijkheid die zowel 'current confinement' als 'optical confinement' (golfgeleiding) combineert is de oxide-confined VCSEL (zie figuur 2.4). Om dit type te maken in het GaAs/AlAs materiaalsysteem wordt een laagje AlGaAs met een Al inhoud van meer dan 80% in de bovenste spiegel ingebouwd. Door het hoge Al gehalte van deze laag kan men ze selectief laten oxideren en zo een oxide apertuur bekomen (zie figuur 2.4). Het resulterende oxide is elektrisch niet geleidend en heeft een veel lagere brekingsindex (1.6) dan de rest



Figuur 2.3: Schematische weergave van een air-post VCSEL (doorsnede).



Figuur 2.4: Schematische weergave van een oxide-confined VCSEL (doorsnede).



Figuur 2.5: Schematische weergave van een edge emitting laser (EEL).

Parameter	EEL	VCSEL
Dikte actieve laag	10 nm - 100 nm	8 nm - 100 nm
Oppervlakte actieve laag	$3 \times 300 \mu\text{m}^2$	$5 \times 5 \mu\text{m}^2$
Actief volume	$60 \mu\text{m}^3$	$0.07 \mu\text{m}^3$
Lengte v.d. caviteit	$300 \mu\text{m}$	$\approx 1 \mu\text{m}$
Reflectiviteit	0.3	0.99 - 0.999

Tabel 2.1: Typische waarden van verschillende parameters van EEL's en VCSEL's. Waarden gebaseerd op [1].

van de structuur. De oxide apertuur zal dus voor current confinement en optical confinement (door index guiding) zorgen. Een nadeel van de apertuur is dat ze de lasermode ook verstrooit. De invloed van de oxide apertuur op de golfgeleiding in de VCSEL kan veranderd worden door de positie ervan aan te passen.

Oxide-confined VCSEL's hebben een hoog rendement dankzij de combinatie van current confinement en verminderde optische verliezen door de index guiding.

2.2 VCSEL vs Edge Emitting Laser

De edge emitting laser (EEL, zie figuur 2.5) is de 'standaard' halfgeleider laser. Bij een EEL loopt de as van de caviteit parallel met het substraat zodat het licht zijdelings uitgezonden wordt. Om de juiste caviteitslengte te bekomen worden de EEL's gekleefd. De oppervlakken die zo ontstaan doen dienst als de spiegels van de laser. Dit kan omdat het actieve gebied in een EEL veel groter is dan bij de VCSEL zodat spiegels met een lagere reflectiviteit volstaan. Tabel 2.1 geeft een overzicht van enkele parameters van VCSEL's en EEL's.

Hoewel EEL's in de praktijk vaker gebruikt worden hebben ze een aantal nadelen ten opzichte van VCSEL's. EEL's moeten bv. gekleefd en verpakt worden voor men ze kan testen. VCSEL's kunnen op de wafer zelf getest worden. Een tweedimensionale matrix van EEL's kan niet monolithisch gemaakt worden, voor VCSEL's is dat geen probleem. Een ander nadeel van de EEL is dat de uitgekoppelde laserbundel niet circulair is. Dit maakt het inkoppelen van de bundel in optische vezels moeilijk. VCSEL's daarentegen hebben wel een cirkelvormige bundel. Ook op gebied van drempelstroom presteert de VCSEL in sommige gevallen beter dan de EEL.

Door de voordelen die VCSEL's in sommige gevallen hebben ten opzichte van EEL's worden ze steeds meer in de praktijk toegepast, bv. in optische netwerken (gigabit ethernet LAN's), laserprinters, parallelle optische interconnecties enz. [1]. Hoewel een aantal jaren geleden de VCSEL nog maar een vrij beperkt golflengtegebied bestreek, bestaan er vandaag VCSEL's voor allerlei golflengten: VCSEL's voor de twee communicatiegolflengten 1300 en 1550 nm, de infrarood GaAs/AlAs-gebaseerde VCSEL's (780 - 980 nm), een VCSEL in het zichtbare gebied (rood licht, 630 - 670 nm), ... Ook golflengte-instelbare (tunable) VCSEL's bestaan [2], en onlangs werd de eerste elektrisch gepompte 1.55 μm VCSEL matrix met meerdere golflengten gedemonstreerd [3].

2.3 Numerieke simulatie van VCSEL's

Bij het numeriek simuleren van de elektromagnetische veldverdeling binnen een VCSEL probeert men de waarde van de elektromagnetische velden te berekenen op een eindig aantal punten/gebieden binnen en/of buiten de caviteit. Afhankelijk van hoe men deze punten/gebieden kiest spreekt men van eindige verschillen, eindige elementen of eindige volumemethodes. Eenmaal het discretisatieschema gekozen is kan men dit gebruiken om de vergelijkingen van Maxwell numeriek op te lossen.

Omdat de simulaties meestal erg zware eisen stellen aan de gebruikte computerinfrastructuur maakt men vaak één- of tweedimensionale benaderingen. Een volledig één-dimensionale benadering heeft echter weinig zin daar de veldprofielen in dat geval analytisch kunnen berekend worden met een matrixformalisme. Een tweedimensionale benadering is in orde voor bv. proton geïmplanteerde VCSEL's omdat daarin geen grote transversale brekingsindexsprongen voorkomen. Voor oxide-confined VCSEL's zijn er lokaal wel grote verschillen in brekingsindex zodat een driedimensionale simulatie aangewezen is [4].

2.3.1 Specifieke problemen

Voor VCSEL's zijn er een aantal problemen bij numerieke simulaties:

- De twee Braggreflectoren samen bestaan als snel uit 80 laagjes materiaal, meestal zelfs meer. Om goede simulatieresultaten te bekomen moeten er volgens de richting van de symmetrieas van de VCSEL voor elk laagje minstens enkele discretisatiepunten beschouwd worden (voor eindige verschillen methodes wordt vaak de vuistregel van 10 punten per golflengte gehanteerd). Het totaal aantal discretisatiepunten zal dus erg groot zijn. Daarbij komt nog dat sterk resonante structuren als VCSEL's zeer gevoelig zijn voor fasefouten.
- VCSEL's bevatten lokaal gemiddeld tot grote brekingsindexverschillen. Een goed voorbeeld is de oxide apertuur bij oxide-confined VCSEL's. In moderne VCSEL's heeft men soms zelfs meerdere aperturen en aperturen die dunner worden naarmate ze dichter bij het centrum van de laser komen. Het juist simuleren van zulke lokale structuren vereist een heel fijne discretisatiegrid. Sommige numerieke methoden werken enkel goed als er geen plotse en grote brekingsindexverschillen voorkomen. Deze methoden kunnen dus niet gebruikt worden bij oxide-confined VCSEL's.
- Om de juiste veldverdeling in de lasercaviteit te bekomen is het nodig dat uitgezonden straling uit het simulatiedomein kan wegstralen: straling in de richting van de rand van het simulatiegebied moet het gebied kunnen verlaten zonder reflecties te genereren. Bij veel simulatiemethoden moeten hiervoor speciale voorzorgen genomen worden en meestal is succes niet gegarandeerd [5, 6].
- De elektromagnetische veldverdeling wordt in de eerste plaats bepaald door het brekingsindexprofiel van de VCSEL. Bij een werkende VCSEL is er echter een continue wisselwerking tussen temperatuur, ruimtelijke ladingsdragersverdeling en brekingsindex.

Een VCSEL is dus een *uitdagende* component om te simuleren.

2.3.2 Overzicht van mogelijke oplossingsmethodes

Een eerste criterium bij de indeling van numerieke methodes geschikt voor VCSEL's is het al dan niet in rekening brengen van het vectoriële karakter van het elektrisch veld. De hier besproken methodes bestaan vaak in zowel scalaire als vectoriële vorm. Algemeen wordt gesteld dat de scalaire vorm voldoende nauwkeurig is tenzij men specifiek wil kijken naar effecten waar de polarisatie van belang is, zoals bij zeer kleine oxideaperturen. Ook voor het bestuderen van polarisatieschakelen in VCSEL's [7] moet het vectoriële aspect in rekening gebracht worden. Los hiervan heeft men drie mogelijkheden om de velden te berekenen: eigenmodes zoeken, veldpropagatie tussen de twee spiegels en volledige simulaties in het tijdsdomein.

Eigenmodes

Een voor de hand liggende vereenvoudiging is het vinden van die oplossingen die niet veranderen in de tijd en die men dus de eigenmodes van de caviteit kan noemen. In het driedimensionale geval kost het vinden van dit soort oplossingen veel rekentijd. Meestal worden er dus benaderingen ingevoerd. De meest courante benadering is het veronderstellen van de scheidbaarheid van de oplossing: men lost het probleem apart op voor het longitudinale en transversale deel van de velden. Vaak stelt men nog dat het transversale deel cilindrisch symmetrisch is zodat er weer een dimensie gewonnen wordt. In deze klasse van methoden vindt men de effectieve brekingsindexmethodes [8, 4] en aanpassingen hiervan die de fout die gemaakt wordt bij het veronderstellen van scheidbare oplossingen in rekening brengen [9].

Indien men niet veronderstelt dat de oplossing separabel is gebruikt men meestal een ontwikkeling in eigenmodes in elk laagje. Deze worden dan aan elkaar 'genaaid' om een oplossing voor de volledige caviteit te vinden. De eigenmodes van elke laag worden volledig analytisch [10], analytisch maar met de niet-uniformiteit als perturbatie beschouwd [11] of numeriek bepaald [12]. Hiernaast is er ook nog de aanpak die gebruik maakt van de Green's functie van de totale structuur [13]. Deze kan semi-analytisch bepaald worden voor uniforme lagen. De niet-uniformiteit kan dan als extra bronterm in rekening gebracht worden.

Veldpropagatie

Omwille van de conceptuele en algoritmische eenvoud was er vroeger veel interesse voor veldpropagatiemethodes. Kort gezegd neemt men in deze methodes het veld in een zeker vlak als beginvoorwaarde en *propageert* het dan stapsgewijs verder in de ruimte. Als men een spiegel raakt wordt de voortplantingsrichting omgekeerd en bij doorgang door het actieve gebied wordt het veld versterkt. Men blijft propageren tot een stabiele oplossing bereikt is. Een overzicht van deze en andere oudere methodes is te vinden in [14].

Tijdsdomein

Methodes die stapje per stapje de tijdsevolutie van het veld bepalen zijn pas recent terug actueel geworden. Vroeger werden ze (voor VCSEL's) vaak als te rekenintensief beschouwd. De steeds verdergaande drang naar integratie van diverse componenten en de steeds hogere transmissiesnelheden bij dewelke deze componenten moeten werken gecombineerd met de snel stijgende rekenkracht van de moderne computers hebben deze methodes nieuw leven ingeblazen [15]. Niettemin zal het aantal discretisatiepunten snel heel groot worden wanneer men de VCSEL beschouwt als een driedimensionale structuur: als men 100 discretisatiepunten volgens de x-, y- en z-richting neemt zijn er in totaal al 10^6 punten waar de

velden tijdens elke tijdstap moeten berekend worden. Daarom zal men vaak proberen om het simulatieprobleem terug te brengen tot een één- of tweedimensionaal probleem.

2.4 Besluit

In dit hoofdstuk hebben we enkele belangrijke VCSEL types besproken:

- De proton-implanted VCSEL, die een vrij goede current confinement biedt maar bijna geen optical confinement. Momenteel zijn de meeste commercieel verkrijgbare VCSEL's van dit type omdat het relatief gemakkelijk te produceren is.
- De air-post VCSEL, die een redelijk zwakke current confinement heeft maar goede optical confinement.
- De oxide-confined VCSEL, die zowel current confinement als optical confinement biedt en daardoor een hoog rendement heeft.

We hebben ook een korte vergelijking gemaakt tussen de vertical cavity surface emitting laser (VCSEL) en de edge emitting laser (EEL). Verder werden de moeilijkheden bekeken die moeten overwonnen worden om VCSEL's numeriek te simuleren. De voornaamste problemen zijn het grote aantal discretisatiepunten die nodig zijn om een nuttige simulatie te bekomen, de randvoorwaarden (die het uitstralen van licht uit het simulatiedomein moeten toelaten) en de wisselwerking tussen temperatuur, ruimtelijke ladingsdragerdichtheid en brekingsindex.

In de rest van dit afstudeerwerk zullen we manieren zoeken om de elektromagnetische velden binnen VCSEL's te berekenen met wavelet-gebaseerde technieken. De bedoeling is om het aantal nodige discretisatiepunten te beperken door de compressie-eigenschappen van wavelets te gebruiken en te werken met een discretisatierooster dat zich kan aanpassen aan de velden die in de simulatie voorkomen.

Wat wavelets precies zijn bespreken we in het volgende hoofdstuk.

Hoofdstuk 3

Wavelets en multiresolutie analyse

Dit hoofdstuk geeft een inleiding tot de wavelettheorie vanuit het oogpunt van de multiresolutie analyse. Na een inleidend voorbeeld zullen we de wavelettransformatie uitleggen in de context waarin ze waarschijnlijk het bekendst is: de tijd-frequentie analyse. Daarna zullen we de multiresolutie analyse bespreken en het verband met de wavelettheorie duidelijk maken. We zullen zien dat het multiresolutieconcept aanleiding geeft tot orthonormale basissen en in sommige gevallen tot economische (ijle) beschrijvingen van functies. Tenslotte zullen we het over de snelle wavelettransformatie hebben en kort uitleggen volgens welke criteria men gewoonlijk bepaalt welke wavelet voor welke toepassingen geschikt is.

De nadruk zal gelegd worden op de concepten die we in de andere hoofdstukken gaan gebruiken, met name de multiresolutie analyse.

Wavelets hebben een uitgebreide wiskundige achtergrond. Veel wiskundige uitwerkingen en bewijzen zullen hier niet herhaald worden. Voor een meer volledige aanpak verwijzen we naar o.a. [16, 17, 18].

3.1 Decorrelatie van functiesamples

Om het belangrijkste idee achter de wavelettransformatie duidelijk te maken zullen we beginnen met een voorbeeld. Stel dat we over acht samples beschikken van een functie f , bemonsterd met genormaliseerde sampleperiode $T = 1$. De acht samples zijn

$$s_0 = (s_{0,1}, s_{0,2}, s_{0,3}, s_{0,4}, s_{0,5}, s_{0,6}, s_{0,7}, s_{0,8})$$

Een voor de hand liggende manier om uit deze samples een benadering f_0 voor de originele functie f te construeren is de benadering door een stuksgewijs constante functie:

$$f_0(x) = \sum_{i=1}^8 s_{0,i} \tilde{\varphi}_{0,i}(x)$$

met $\tilde{\varphi}_{0,i}(x) = \varphi(x - i)$ en

$$\varphi(x) = \begin{cases} 1, & -1 \leq x < 0 \\ 0, & \text{elders} \end{cases}$$

f_0 is linksboven in figuur 3.1 (a) afgebeeld voor het geval

$$s_0 = (1, 2, 3, 4.5, 5, -1, -0.5, 1.5).$$

De functies $\tilde{\varphi}_{0,i}$ vormen een orthogonale basis.

Stel dat we nu telkens 2 samples vervangen door hun gemiddelde. Om de originele samples terug te kunnen vinden moeten we ook de fout bijhouden, nl. het verschil tussen de originele samples en de gemiddelden. De stuksgewijs constante functie gevormd door de gemiddelden noemen we f_1 , de foutfunctie noemen we e_1 (zie figuur 3.1). We kunnen f_0 terugvinden door f_1 en e_1 op te tellen:

$$f_0 = f_1 + e_1$$

Zoals te zien op de figuur kan ook f_1 beschouwd worden als een lineaire combinatie van pulsvormige basisfuncties. De basisfuncties van e_1 zijn daarentegen blokgolven beperkt tot één periode. f_1 en e_1 kunnen dus als volgt geschreven worden:

$$f_1 = \sum_{i=1}^4 s_{1,i} \tilde{\varphi}_{1,i}(x)$$

$$e_1 = \sum_{i=1}^4 d_{1,i} \tilde{\psi}_{1,i}(x)$$

met

$$s_{1,i} = \frac{s_{0,2i} + s_{0,2i+1}}{2}$$

$$d_{1,i} = \frac{s_{0,2i} - s_{0,2i+1}}{2},$$

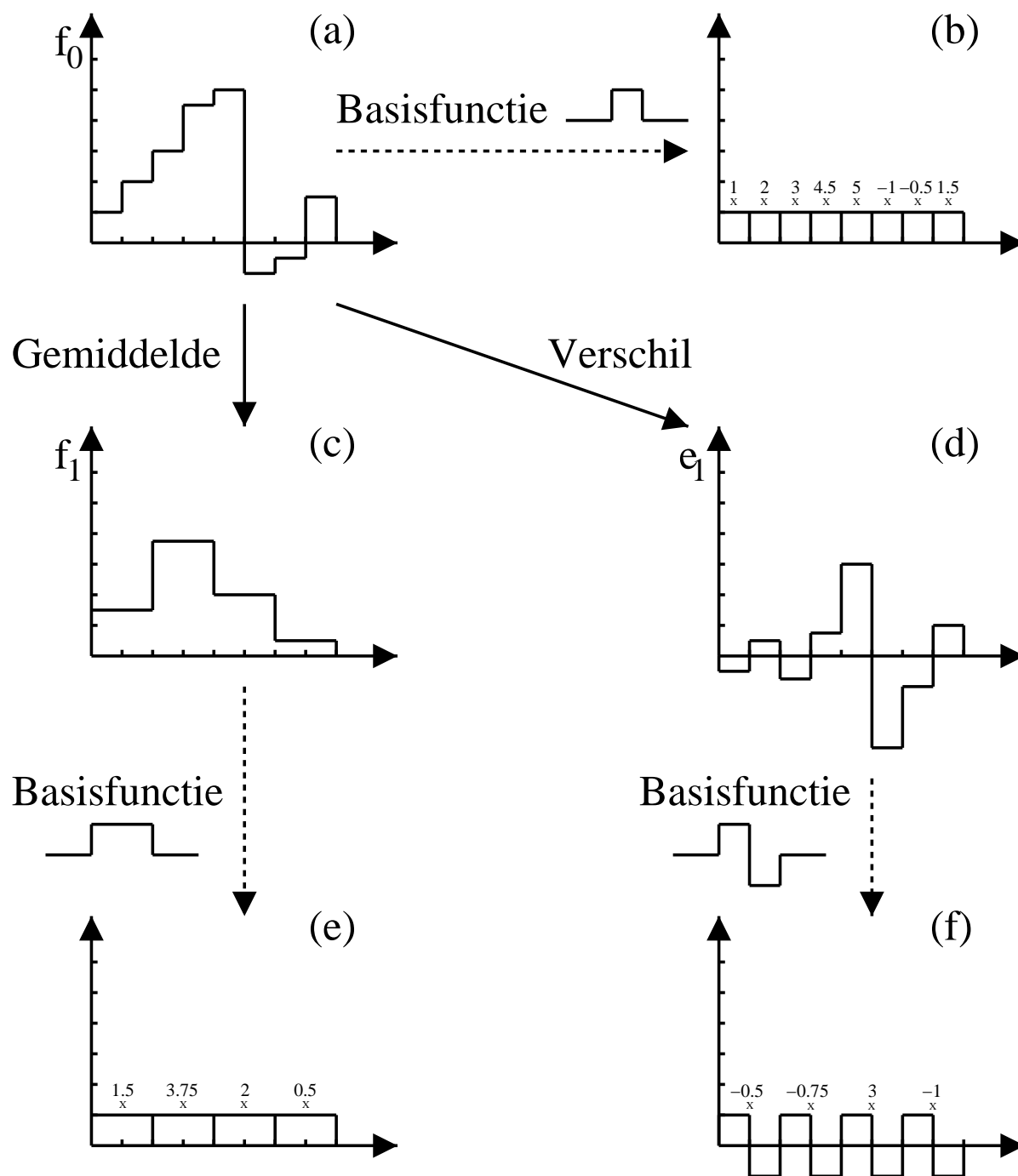
$\tilde{\varphi}_{1,i}(x) = \varphi\left(\frac{x-2i}{2}\right)$, $\tilde{\psi}_{1,i}(x) = \psi\left(\frac{x-2i}{2}\right)$ en

$$\psi(x) = \begin{cases} 1, & -1 \leq x < -1/2 \\ -1, & -1/2 \leq x < 0 \\ 0 & \text{elders} \end{cases}$$

De $s_{1,i}$ noemen we schalingcoëfficiënten, de $d_{1,i}$ detailcoëfficiënten.

Op dezelfde manier als we f_0 ontbonden hebben, kunnen we f_1 ontbinden in f_2 en e_2 , en f_2 weer in f_3 en e_3 :

$$f_0 = e_1 + f_1 = e_1 + e_2 + f_2 = e_1 + e_2 + e_3 + f_3$$



Figuur 3.1: Eén stap van de Haar-transformatie: f_0 (figuur (a)) wordt ontbonden in $f_1 + e_1$ (respectievelijk figuren (c) en (d)). f_0 is een lineaire combinatie van puls-vormige basisfuncties (figuur (b)). De basisfuncties van f_1 zijn 'uitgerekte' versies van die van f_0 (figuur (c)) en die van e_1 zijn blokgolven beperkt tot één periode (figuur (f)).

De basisfuncties voor f_j en e_j zijn respectievelijk

$$\begin{aligned}\tilde{\varphi}_{j,i}(x) &= \varphi\left(\frac{x-2^j i}{2^j}\right) \\ \tilde{\psi}_{j,i}(x) &= \psi\left(\frac{x-2^j i}{2^j}\right)\end{aligned}$$

Dit zijn uitgerekte en verschoven versies van de functies φ en ψ . Het aantal coëfficiënten $s_{j,i}$ en $d_{j,i}$ halveert bij elke ontbindingsstap j . We kunnen blijven ontbinden tot we nog maar één schalingcoëfficiënt overhebben. Als we oorspronkelijk 2^J samples hebben, kunnen we dus J keer ontbinden. f_0 schrijven als $\sum_{j=1}^J (e_j) + f_J$ noemt men de Haartransformatie. φ is de Haar schalingfunctie en ψ is de Haar waveletfunctie.

Voor de wiskundige theorie achter wavelets is het handig als de $L^2(\mathbb{R})$ norm van de basisfuncties op de verschillende ontbindingsniveaus of schalen dezelfde is. De $L^2(\mathbb{R})$ norm $\|f\|$ van een (complexe) functie $f(x)$ is

$$\|f\|^2 = \int_{-\infty}^{+\infty} |f(x)|^2 dx \quad (3.1)$$

Voor de functies $\tilde{\varphi}_{j,i}$ blijft deze norm niet gelijk op de verschillende schalen (voor de $\tilde{\psi}_{j,i}$ is de $L^2(\mathbb{R})$ norm steeds nul). Daarom definieert men de basisfuncties meestal als volgt:

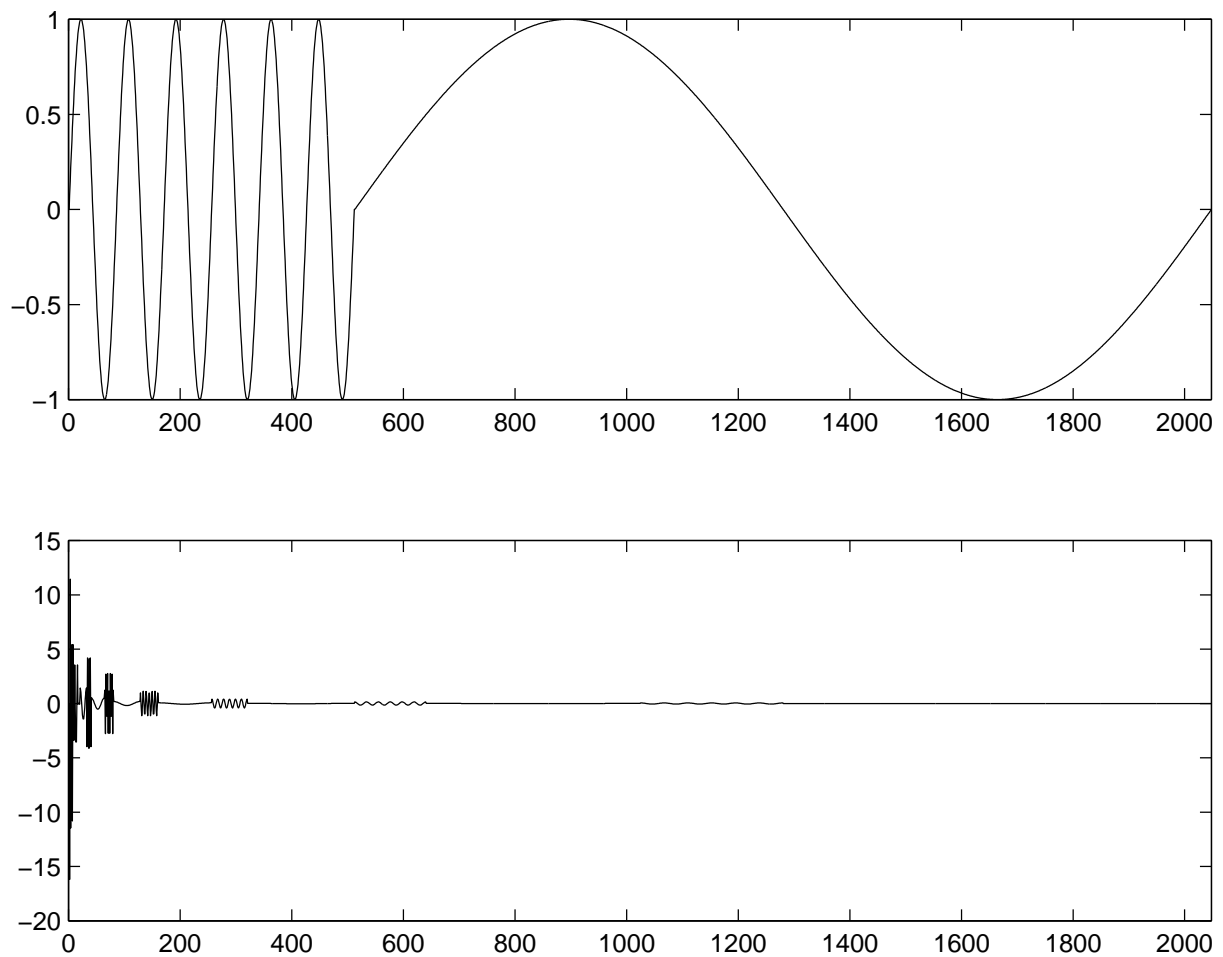
$$\begin{aligned}\varphi_{j,i}(x) &= \frac{1}{\sqrt{2^j}} \varphi\left(\frac{x-2^j i}{2^j}\right) \\ \psi_{j,i}(x) &= \frac{1}{\sqrt{2^j}} \psi\left(\frac{x-2^j i}{2^j}\right)\end{aligned}$$

De functie f_1 , de ‘gemiddelde’ functie van f_0 , is een minder gedetailleerde versie van f_0 . f_1 kunnen we beschouwen als f_0 bekeken op een groffere schaal of een lagere resolutie. We zien op figuur 3.1 dat op plaatsen waar f_0 sterk varieert de functie e_1 en dus de coëfficiënten $d_{1,i}$ grote waarden aannemen. Op plaatsen waar de functie f_0 niet sterk varieert (‘smooth’ is), zullen e_1 en de detailcoëfficiënten $d_{1,i}$ klein zijn. Dit is veel duidelijker te zien bij een voorbeeld met meer samples zoals in figuur 3.2. Door heel kleine coëfficiënten gelijk aan nul te stellen (dit noemt men ‘thresholding’), bekomen we een economische, ijle benaderingen van functies (en maken we dus een kleine fout!). Dit noemt men het *decorrelerend* vermogen van wavelets. Het vormt de basis van bijna alle wavelet-gebaseerde compressiemethodes.

3.2 Wavelets

3.2.1 Tijd-frequentie analyse

We bekijken een functie $f(t)$, waar t zowel ruimte als tijd kan voorstellen. Voorlopig zullen we t benoemen als tijd. We veronderstellen dat f behoort tot een Hilbert ruimte met als



Figuur 3.2: Signaal (boven) en de Haartransformatie (onder). De Haarcoëfficiënten zijn als volgt georganiseerd: $(s_{11,1}; d_{11,1}; d_{10,1}, d_{10,2}; d_{9,1}, d_{9,2}, d_{9,3}, d_{9,4}; \dots; d_{1,1}, \dots, d_{1,1024})$. We zien dat veel coëfficiënten zeer klein of nul zijn.

inwendig produkt

$$\langle f, g \rangle = \int_{-\infty}^{+\infty} f(t) \overline{g(t)} dt \quad (3.2)$$

en dat f kwadratisch integreerbaar is, m.a.w. de $L^2(\mathbb{R})$ norm van f is eindig:

$$\|f\|^2 = \int_{-\infty}^{+\infty} |f(t)|^2 dt < +\infty,$$

$\|f\|^2$ noemt men ook de *energie* van f . In (3.2) gebruikten we \bar{g} als notatie voor het complex toegevoegde van g . Voor het gemak zullen we $f(t)$ reëel veronderstellen.

Er zijn twee voorstellingen van f die veel gebruikt worden: de tijdsvoorstelling (Diracbasis)

$$f(t) = \int_{-\infty}^{+\infty} f(u) \delta(t-u) du \quad (3.3)$$

en de frequentievoorstelling (Fourierbasis)

$$f(t) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} \hat{f}(\omega) e^{i\omega t} d\omega \quad (3.4)$$

met

$$\hat{f}(\omega) = \int_{-\infty}^{+\infty} f(t) e^{-i\omega t} dt \quad (3.5)$$

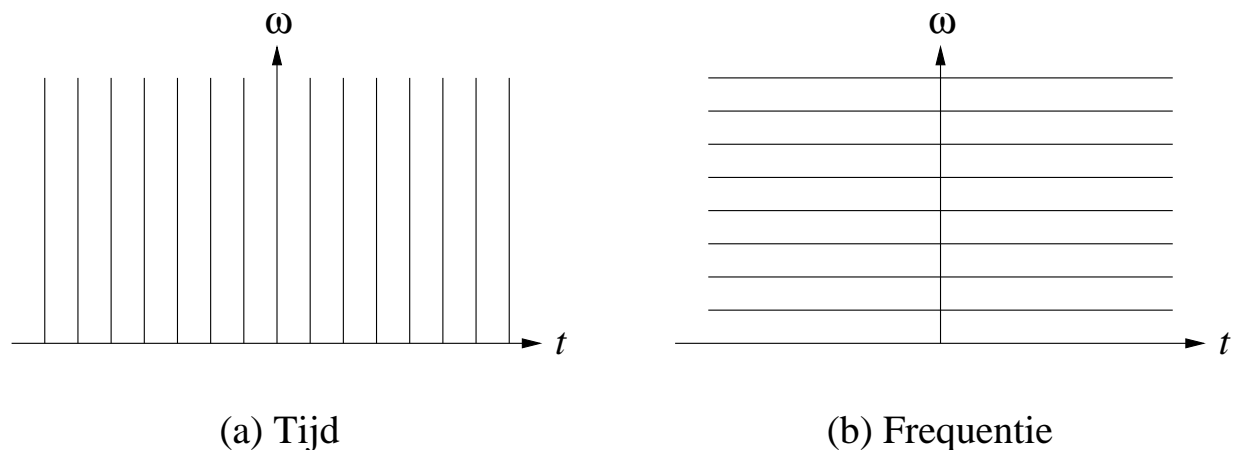
(we zullen steeds de notatie \hat{f} gebruiken voor de Fouriergetransformeerde van een functie f). Zowel (3.3) als (3.4) bevatten alle informatie over f , maar er is telkens maar één type informatie beschikbaar: tijdsinformatie in (3.3) en frequentieinformatie in (3.4). Dit komt omdat de Dirac distributies $\delta(t-u)$ perfect gelokaliseerd zijn in de tijd maar een constant spectrum hebben en omdat de functies $e^{i\omega t}$ oneindig uitgestrekt zijn in de tijd maar perfect gelokaliseerd in frequentie.

Wavelets zijn ontstaan tijdens de zoektocht naar een voorstelling voor f die het midden houdt tussen deze extremen: men wilde basisfuncties gebruiken die zowel in tijd als frequentie gelokaliseerd zijn. De welbekende theoretische limiet hierop is het *onzekerheidsprincipe van Heisenberg* (een algemene eigenschap van functies, niet beperkt tot de quantumfysica). Om een functie f bv. beter te lokaliseren in de tijd, kunnen we f schalen met een factor $0 < s < 1$, terwijl we de energie $\|f\|^2$ (zie (3.1)) constant houden:

$$f_s(t) = \frac{1}{\sqrt{s}} f\left(\frac{t}{s}\right)$$

Het gevolg is dat de Fouriergetransformeerde $\hat{f}_s(\omega) = \sqrt{s} \hat{f}(s\omega)$ geschaald wordt met $1/s$. We verliezen dus in frequentielokalisatie wat we gewonnen hebben in de tijd. Als we het centrum $c(f)$ en de straal $\Delta(f)$ van een functie f definiëren als

$$\begin{aligned} c(f(t)) &= \frac{1}{\|f\|^2} \int_{-\infty}^{+\infty} t |f(t)|^2 dt \\ \Delta(f(t))^2 &= \frac{1}{\|f\|^2} \int_{-\infty}^{+\infty} (t - c(f))^2 |f(t)|^2 dt \end{aligned}$$



Figuur 3.3: Indeling van het tijd-frequentie vlak geassocieerd met de Dirac en Fourier basis.

dan kan het onzekerheidsprincipe van Heisenberg geschreven worden als [16, 19, 17]

$$\Delta(f) \Delta(\hat{f}) \geq \frac{1}{2} \quad (3.6)$$

3.2.2 Tijd-frequentie en tijd-schaal voorstellingen

Met elke functie f kunnen we een tijd-frequentie venster associëren in het (t, ω) vlak, gecentreerd rond $(c(f), c(\hat{f}))$ en met afmetingen $2\Delta(f) \times 2\Delta(\hat{f})$. Deze tijd-frequentie vensters noemt men *Heisenberg rechthoeken*. De oppervlakte $4\Delta(f)\Delta(\hat{f})$ van een Heisenberg rechthoek moet groter zijn dan 2 wegens het Heisenberg onzekerheidsprincipe (3.6).

Als we een functie schrijven als een superpositie van basisfuncties, kunnen we aan deze basis een indeling van het tijd-frequentievlak associëren, gebaseerd op de Heisenberg rechthoeken van de basisfuncties. Voor de Dirac en Fourier basisfuncties zijn de Heisenberg rechthoeken oneindig lang en plat (lijnen). De tijd-frequentievlak indelingen voor deze basissen zijn te zien op figuur 3.3. Op deze figuur is slechts een eindige subset van het oneindige aantal lijnen (= platgedrukte Heisenberg rechthoeken) weergegeven. Bij discrete functies zijn de Heisenberg rechthoeken niet volledig platgedrukt: hun kortste zijde is dan gelijk aan het sample interval.

Tijd-frequentie aanpak

Een manier om zowel in tijd als frequentie gelokaliseerde basisfuncties te bekomen is de *gevensterde Fouriertransformatie* (windowed Fourier transform) of *korte tijd Fouriertransformatie* (short time Fourier transform): in plaats van de basisfuncties $e^{i\omega t}$ te gebruiken

(zoals bij de gewone Fouriertransformatie, zie (3.5)), gebruiken we ontwikkelingen in functie van zogenaamde tijd-frequentie atomen (time frequency atoms)

$$g_{u,\omega}(t) = e^{i\omega t} g(t - u)$$

$g(t)$ is een vensterfunctie, die nul of heel klein wordt als $|t|$ groot genoeg is, symmetrisch is ($g(t) = g(-t)$), reëel en genormaliseerd zodat $\|g\| = 1$. De vensterfunctie beperkt $e^{i\omega t}$ in de tijd. De $g_{u,\omega}$ zijn dus in de tijd verschoven versies van de vensterfunctie, gemoduleerd met frequentie ω . Ze zijn zowel gelokaliseerd in tijd (rond u) als frequentie (rond ω). Het eenvoudigste voorbeeld van een vensterfunctie is

$$g(t) = \begin{cases} 1 & -1/2 \leq t \leq 1/2 \\ 0 & \text{elders} \end{cases}$$

Andere voorbeelden zijn afgebeeld in figuur 3.4.

De prijs die we voor de tijdslokalisatie moeten betalen is een verminderde frequentielokalisatie (ten opzichte van de perfecte frequentielokalisatie van de functies $e^{i\omega t}$). Hoe geleidelijker $g(t)$ naar nul gaat, hoe minder frequentielokalisatie we verliezen (en hoe minder tijdslokalisatie we winnen!).

Daar de tijd-frequentie atomen gelokaliseerd zijn in de tijd, volstaat het niet meer een functie te schrijven als superpositie van deze atomen voor alle mogelijke frequenties (zoals in (3.4)). We moeten ook de tijdstranslaties beschouwen.

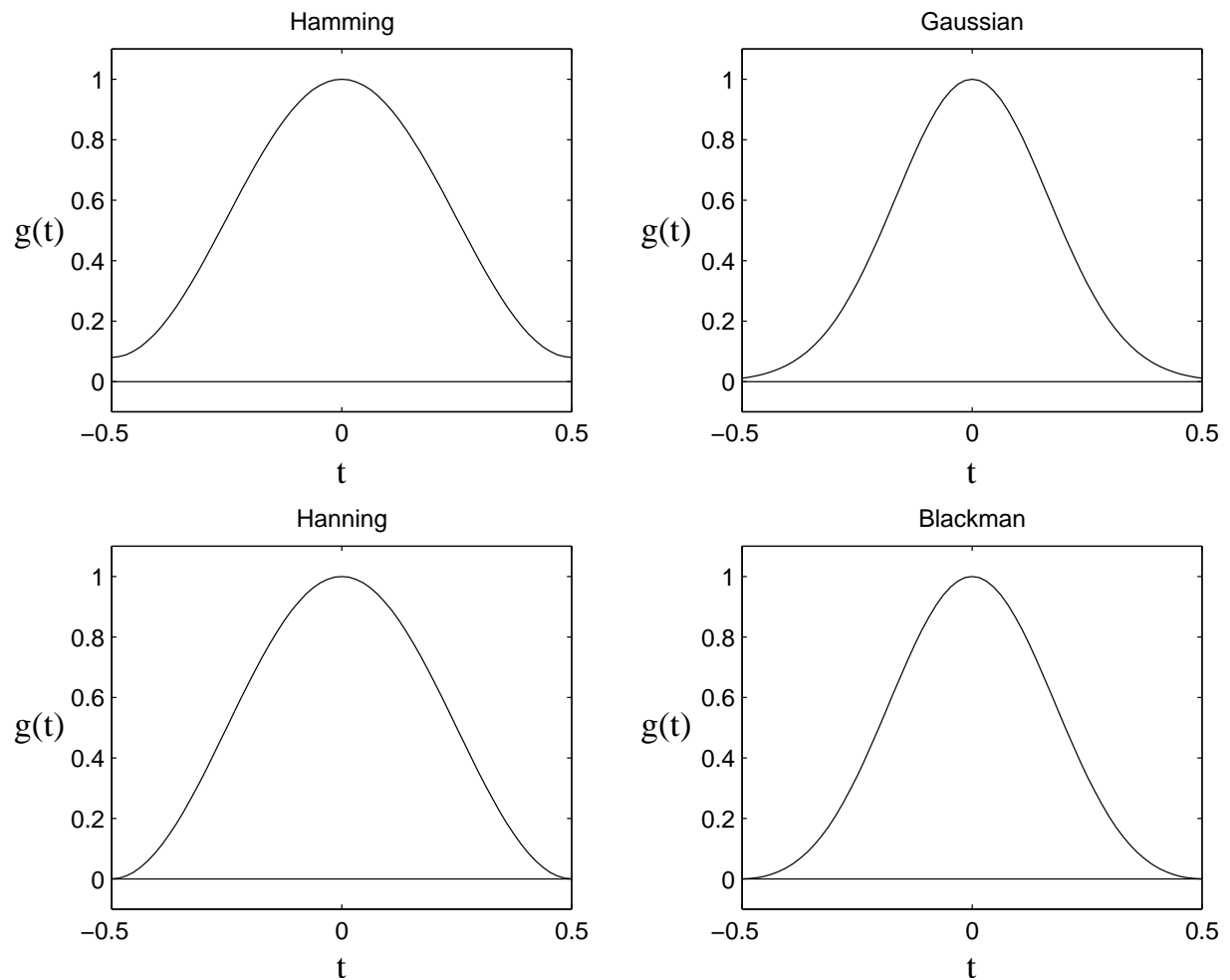
De definitie van de gevensterde Fouriertransformatie is analoog aan die van de gewone Fouriertransformatie (3.5) [16, 17]:

$$\begin{aligned} \widetilde{\mathcal{W}}f(u, \omega) &= \int_{-\infty}^{+\infty} f(t) e^{-i\omega t} g(t - u) dt \\ &= \int_{-\infty}^{+\infty} f(t) \overline{g_{u,\omega}(t)} dt \\ &= \langle f, g_{u,\omega} \rangle \end{aligned} \tag{3.7}$$

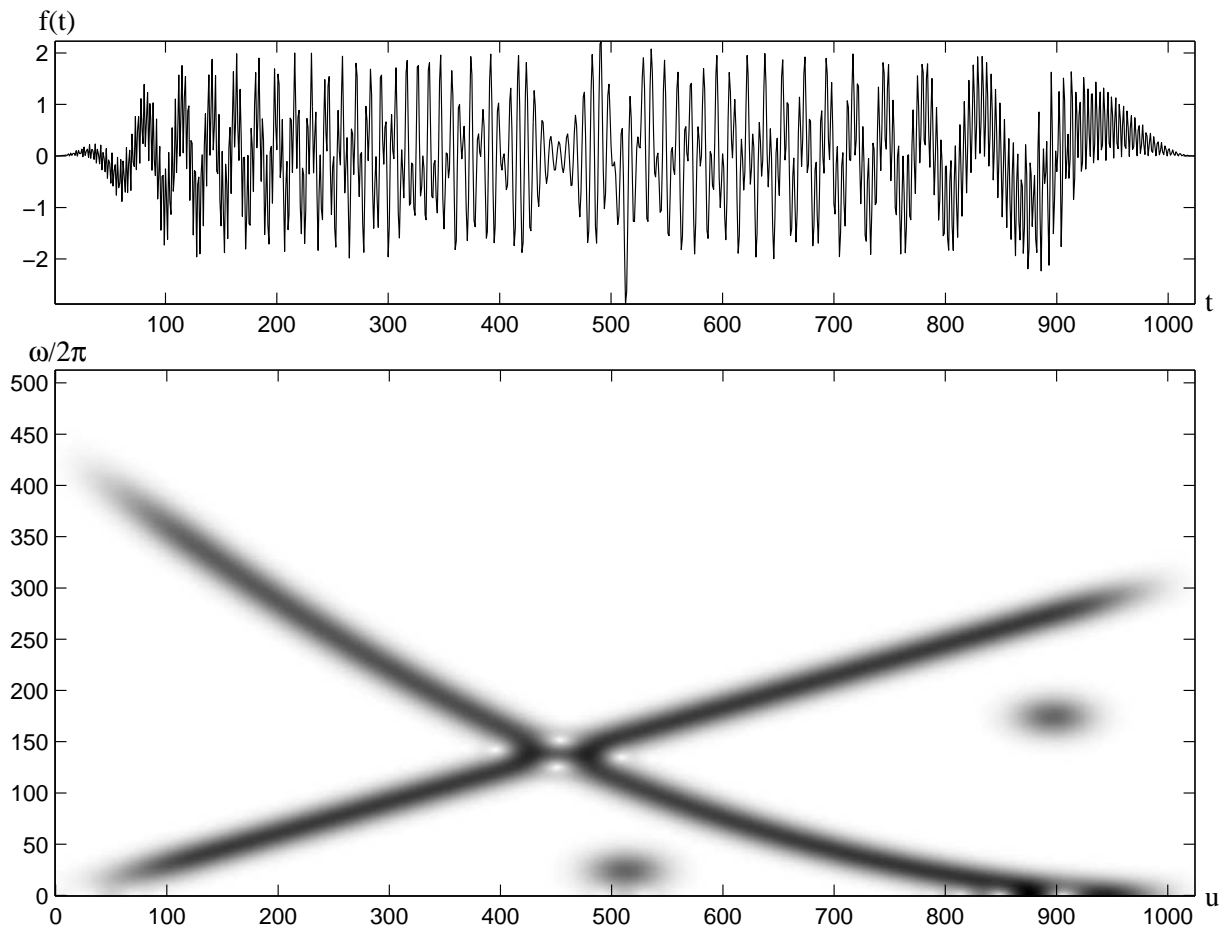
Een voorbeeld van een gevensterde Fouriertransformatie is te zien in figuur 3.5. De inverse transformatie heeft als extra t.o.v. de gewone inverse Fouriertransformatie dat we ook alle tijdstranslaties van de gemoduleerde vensterfunctie in rekening moeten brengen:

$$\begin{aligned} f(t) &= \frac{1}{2\pi} \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} \widetilde{\mathcal{W}}f(u, \omega) e^{i\omega t} g(t - u) d\omega du \\ &= \frac{1}{2\pi} \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} \widetilde{\mathcal{W}}f(u, \omega) g_{u,\omega}(t) d\omega du \end{aligned}$$

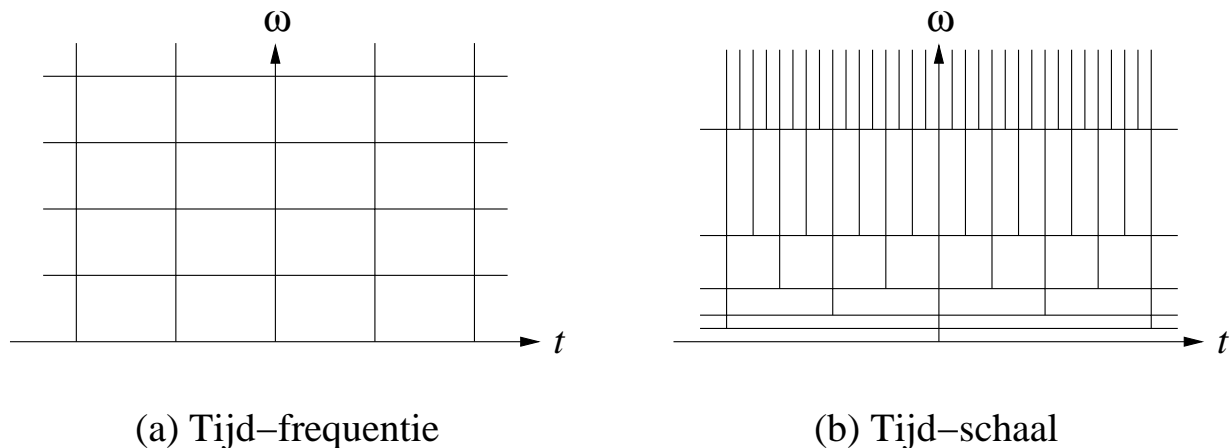
Daar de gevensterde Fouriertransformatie $\widetilde{\mathcal{W}}f(u, \omega)$ van $f(t)$ een functie is van twee onafhankelijke variabelen, is het duidelijk dat deze transformatie redundant is. Deze redundantie



Figuur 3.4: Voorbeelden van vensterfuncties g . Ze zijn beperkt tot het interval $[-1/2, 1/2]$ (daarbuiten stellen we ze nul). Voor praktisch gebruik moeten we wel nog normaliseren, zodat $\|g\| = 1$. De gevensterde Fouriertransformatie die gebruik maakt van een gaussische vensterfunctie noemt men ook *Gabor transformatie*. Figuur overgenomen uit [16].



Figuur 3.5: Een functie $f(t)$ (boven) en gekwadrateerde norm van haar gevensterde Fourier transformatie $|\widetilde{W}f(u, \omega)|^2$ (onder). De onderste figuur noemt men het spectrogram van f . Het meet de energie van f in de tijd-frequentie buurt van (u, ω) , bepaalt door de Heisenberg rechthoek van $g_{u, \omega}$. f is de som van een lineaire chirp waarvan de frequentie toeneemt, een kwadratische chirp waarvan de frequentie afneemt en twee gemoduleerde gaussische functies. Deze vier componenten zijn duidelijk te onderscheiden. Figuur overgenomen uit [16].



Figuur 3.6: Indeling van het tijd-frequentie vlak bij de tijd-frequentie en tijd-schaal aanpak.

maakt dat niet elke (kwadratisch integreerbare) functie van twee variabelen u en ω de gevensterde Fouriertransformatie van een functie f is: de gevensterde Fouriertransformaties moeten voldoen aan [16, 17]

$$\widetilde{\mathcal{W}}f(u_0, \omega_0) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} \widetilde{\mathcal{W}}f(u, \omega) K(u_0, u, \omega_0, \omega) dud\omega$$

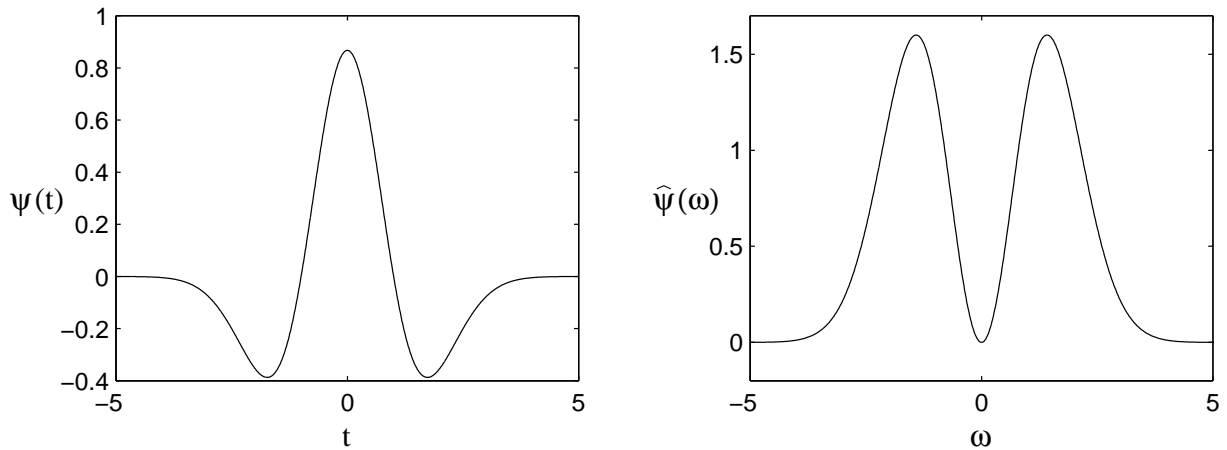
met

$$K(u_0, u, \omega_0, \omega) = \langle g_{u, \omega}, g_{u_0, \omega_0} \rangle$$

K noemt men de herproducerende kern ('reproducing kernel'). In veel gevallen kan men de redundantie verminderen door discrete waarden van u en ω te beschouwen. Zie [17] voor meer informatie.

De vorm van de Heisenberg rechthoeken van de functies $g_{u, \omega}(t)$ is onafhankelijk van u en ω [16]. De tijd-frequentievlak indeling van de tijd-frequentie atomen $g_{u, \omega}(t)$ voor discrete u en ω ziet er dus uit zoals te zien op figuur 3.6 (a).

Stel dat het venster $g(t)$ dat we gebruiken heel klein of nul wordt vanaf $|t| > T/2$. We kunnen T dus beschouwen als de 'breedte' van het venster. Deze breedte vormt een soort scheidingslijn tussen tijd en frequentie: variaties van $f(t)$ over tijdsintervallen veel groter dan T duiken op in het tijdsgedrag van $\widetilde{\mathcal{W}}f(u, \omega)$, terwijl variaties over intervallen veel kleiner dan T Fouriergetransformeerd worden en in het frequentiegedrag van $\widetilde{\mathcal{W}}f(u, \omega)$ belanden [17]. De vensterbreedte T legt een 'tijdschaal' op aan de analyse. Dit heeft als gevolg dat de ontwikkeling van een functie f als superpositie van tijd-frequentie atomen $g_{u, \omega}$ (voor discrete u en ω) geen economische beschrijving zal opleveren als het tijdsgedrag van f heel snel of heel traag is ten opzichte van de vensterbreedte T . Snelle variaties bekomen door superposities van $g_{u, \omega}(t)$ atomen zijn nl. enkel mogelijk door constructieve en destructieve interferentie tussen de atomen. Daarvoor zullen veel frequenties nodig zijn, zodat $\widetilde{\mathcal{W}}f(u, \omega)$



Figuur 3.7: Mexican hat wavelet $\psi(t) = \frac{-2}{\pi^{1/4}\sqrt{3}\sigma} \left(\frac{t^2}{\sigma^2} - 1 \right) \exp\left(\frac{-t^2}{2\sigma^2}\right)$ (links) en de Fouriertransformatie ervan (rechts). Figuur overgenomen uit [16].

erg uitgespreid zal zijn in frequentie. Trage variaties kunnen enkel bekomen worden door vele in de tijd verschoven atomen, waardoor $\widetilde{\mathcal{W}}f(u, \omega)$ uitgespreid zal zijn in de tijd. Dit is de motivatie om de tijd-schaal aanpak van wavelets te gebruiken.

Tijd-schaal aanpak

We hebben gezien dat bij de tijd-frequentie aanpak de 'breedte' T van de vensterfunctie een belangrijke grootte is. Een bepaalde waarde van T kan voor sommige functies geschikt zijn en voor andere ongeschikt. Het invoeren van vensterfuncties met vaste T houdt in dat men de te analyseren functie f steeds op dezelfde 'schaal' bekijkt. Met eigenschappen van f die men op die schaal moeilijk kan waarnemen (kleine details en grote lijnen (variaties veel trager dan T)) heeft de tijd-frequentie aanpak moeite.

De tijd-schaal aanpak is een oplossing voor dit probleem. We beginnen met een functie $\psi(t)$, de *moeder wavelet* (mother wavelet). We zullen zien dat ψ een heel ander soort functie is dan de vensterfuncties die bij de gevensterde Fouriertransformatie gebruikt worden. We veronderstellen dat ψ behoort tot $L^2(\mathbb{R})$, genormaliseerd is $\|\psi\| = 1$, gecentreerd is in de buurt van $u = 0$ en dat $\psi(t) \rightarrow 0$ voor grote $|t|$ (m.a.w., ψ heeft een voldoende snel verval (decay), is voldoende gelokaliseerd). Later zullen nog meer eisen op ψ volgen. Zie figuur 3.7 voor een voorbeeld van een mogelijke moeder wavelet .

Net zoals g introduceert ψ een schaal (zijn breedte). Omdat we dit willen vermijden, zullen we f niet enkel schrijven als een superpositie van getransleerde ψ 's, maar ook van alle mogelijke schalingen van ψ met een factor $s \neq 0$:

$$\psi_s(t) = \frac{1}{\sqrt{|s|}} \psi\left(\frac{t}{s}\right)$$

s noemen we de schaalfactor. De factor $\frac{1}{\sqrt{|s|}}$ zorgt ervoor dat de $L^2(\mathbb{R})$ norm van ψ_s steeds constant blijft en gelijk aan 1. Voor bv. $s > 1$ is ψ_s een versie van ψ die met een factor s uitgerokken is in de tijd en kleiner geworden is in amplitude (door de $\frac{1}{\sqrt{|s|}}$). Als we ψ uitrekken en dus de tijdslokalisatie verslechteren, verbeteren we de frequentielokalisatie. Functies ψ_s met grote $|s|$ zullen laagfrequente variaties van een functie f kunnen weergeven. Omgekeerd zullen hoogfrequente variaties weergegeven worden door functies met kleine $|s|$. Tijdslokalisatie bekomen we door functies te ‘bekijken’ door in de tijd verschoven versies van de ψ_s :

$$\psi_{s,u}(t) = \frac{1}{\sqrt{|s|}} \psi\left(\frac{t-u}{s}\right)$$

Als de moeder wavelet $\psi(t)$ ‘breedte’ T heeft en gelokaliseerd is rond $t = 0$ dan heeft $\psi_s(t)$ breedte $|s|T$ rond $t = 0$ en $\psi_{s,u}(t)$ heeft dezelfde breedte rond $t = u$. De functies $\psi_{s,u}$ zijn *wavelets*, geassocieerd aan de moeder wavelet ψ .

De tijd-frequentievlak indeling geassocieerd aan de wavelets $\psi_{s,u}$ voor discrete waarden van s en u is te zien op figuur 3.6 (b) (pagina 22).

3.2.3 De continue wavelettransformatie

Een wavelet expansie van een functie f is een expansie in functie van geschaalde en verschoven versies van één vaste functie, de moeder wavelet $\psi \in L^2(\mathbb{R})$. Bij de continue wavelettransformatie (ingevoerd in 1984 door Morlet en Grossmann) variëren de schaal- en translatieparameters continu.

De continue wavelet transformatie is gedefinieerd als [19, 17, 16]

$$\mathcal{W}f(s, u) = \langle f, \psi_{s,u} \rangle = \int_{-\infty}^{+\infty} f(t) \overline{\psi_{s,u}(t)} dt \quad (3.8)$$

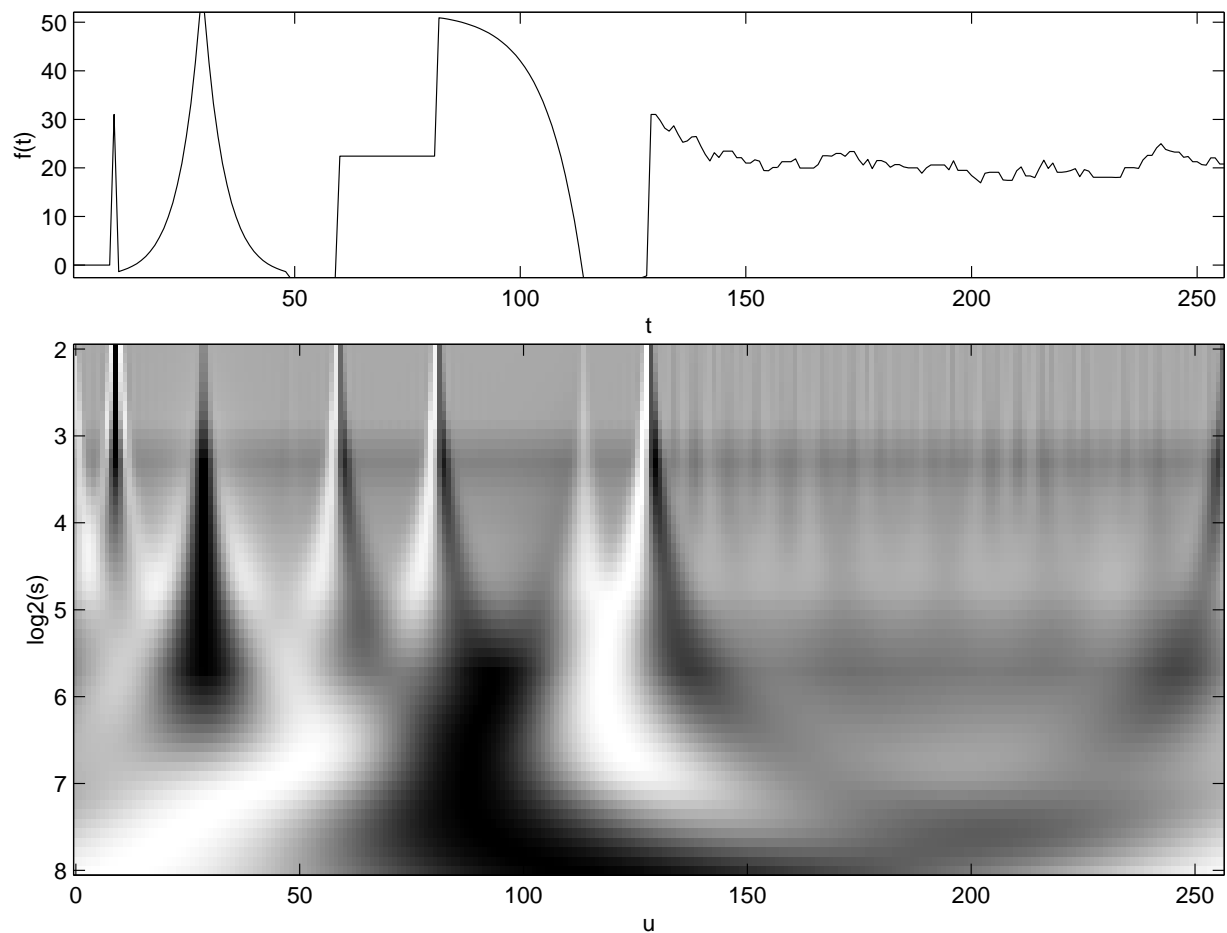
Een voorbeeld van een continue wavelet transformatie berekend met een reële moeder wavelet is te zien in figuur 3.8. Als ψ voldoet aan

$$C_\psi = \int_{-\infty}^{+\infty} \frac{|\hat{\psi}(\omega)|}{\omega} d\omega < +\infty, \quad (3.9)$$

dan is de continue wavelettransformatie $\mathcal{W}f(s, u)$ inverteerbaar:

$$f(t) = \frac{1}{C_\psi} \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} \mathcal{W}f(s, u) \psi_{s,u}(t) du \frac{ds}{s^2} \quad (3.10)$$

f wordt dus geschreven als superpositie van geschaalde en getransleerd versies van ψ . De inversieformule (3.10) werd al in 1964 bewezen door Calderón in een andere context dan de wavelet analyse. Morlet en Grossmann hebben ze in 1984 herontdekt.



Figuur 3.8: Continue wavelet transform $\mathcal{W}f(s, u)$ (onder) van de functie $f(t)$ (boven). De gebruikte wavelet ψ is reëel, met $\psi = -\psi_{\text{Mexican hat}} \cdot \psi_{\text{Mexican hat}}$ is te zien in figuur 3.7. Zwarte, grijze en witte punten komen overeen met respectievelijk positieve, nul en negatieve coëfficiënten. Figuur naar [16].

Opdat voorwaarde (3.9) voldaan kan zijn moet gelden $\widehat{\psi}(0) = 0$, of

$$\begin{aligned}\widehat{\psi}(0) &= \int_{-\infty}^{+\infty} \psi(t) e^{-i \cdot 0 \cdot t} dt \\ &= \int_{-\infty}^{+\infty} \psi(t) dt \\ &= 0\end{aligned}$$

We zien dus dat $\int_{-\infty}^{+\infty} \psi(t) dt$, de gemiddelde waarde van ψ , nul moet zijn. $\psi(t)$ kan dus niet positief of negatief zijn $\forall t$ en moet dus oscilleren. Deze eigenschap heeft aanleiding gegeven tot de naam wavelet ('kleine golf').

Net zoals de gevensterde Fouriertransformatie is de continue wavelettransformatie redundant: $\mathcal{W}f(s, u)$ voldoet aan

$$\mathcal{W}f(s_0, u_0) = \frac{1}{C_\psi} \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} \mathcal{W}f(s_0, u_0) K(u_0, u, s_0, s) du \frac{ds}{s^2} \quad (3.11)$$

met als herproducerende kern

$$K(u_0, u, s_0, s) = \langle \psi_{s,u}, \psi_{s_0,u_0} \rangle$$

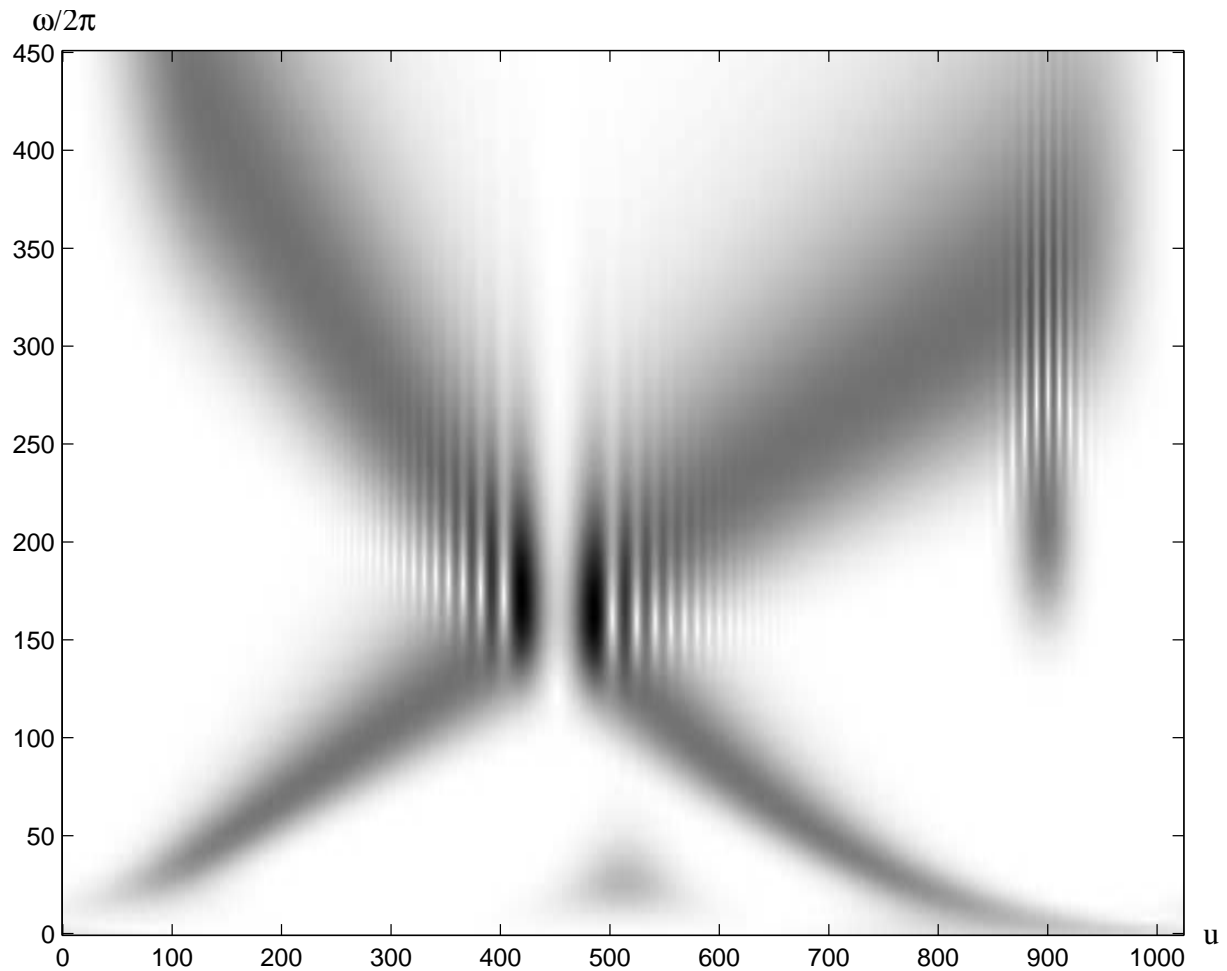
Dit is te zien door de waveletexpansie (3.10) in de continue wavelettransformatie (3.8) $\mathcal{W}f(s_0, u_0)$ te substitueren en de volgorde van integratie om te wisselen.

Een continue wavelet transformatie kan de tijdsevolutie van frequentie-transiënts meten. Hiervoor moet de wavelet ψ complex analytisch zijn. We noemen een functie f analytisch als de Fouriertransformatie \widehat{f} van f nul is voor negatieve ω :

$$\widehat{f} = 0 \text{ als } \omega < 0$$

Een analytische functie is noodzakelijk complex (de Fouriergetransformeerde van reële functies is steeds symmetrisch rond $\omega = 0$), maar wordt volledig bepaald door haar reëel deel [16]. Het genormaliseerde scalogram (scalogram) $\eta^{-1}\omega |\mathcal{W}f(s, u)|^2 = \eta^{-1}\omega |\mathcal{W}f(\frac{\eta}{\omega}, u)|^2$, afgeleid van de continue wavelet transformatie (met ψ analytisch) van de functie $f(t)$ uit figuur 3.5 is te zien in figuur 3.9. Hierin is η de centerfrequentie van $\widehat{\psi}$

$$\begin{aligned}\eta &= c(\widehat{\psi}) \\ &= \frac{1}{\|\widehat{\psi}\|^2} \int_{-\infty}^{+\infty} \omega |\widehat{\psi}|^2 d\omega \\ &= \frac{1}{\|\widehat{\psi}\|^2} \int_0^{+\infty} \omega |\widehat{\psi}|^2 d\omega\end{aligned}$$



Figuur 3.9: Genormaliseerd schalogram $\eta^{-1}\omega|\mathcal{W}f(s,u)|^2$ van de functie $f(t)$ uit figuur 3.5. Voor lage frequenties is de resolutie van het schalogram beter dan van het spectrogram uit figuur 3.5, voor hoge frequenties is ze duidelijk slechter.

De Heisenberg rechthoeken van erg tijdsgelocaliseerde waveletfuncties (nodig om de hoogfrequente inhoud van f voor te stellen) zijn noodzakelijk breed in frequentie. Dit uit zich in het verbreden van de chirp-curves in de hogere frequenties. Figuur overgenomen uit [16].

De Heisenberg rechthoek van $\psi_{s,u}$ is gecentreerd rond $(u, \omega = \frac{u}{s})$ in het tijd-frequentievlak. $|\mathcal{W}f(\frac{u}{s}, u)|^2$ meet de energie van f in het tijd-frequentievenster geassocieerd met $\psi_{s,u}$. Omdat de frequentie-bandbreedte van $\psi_{s,u}$ evenredig is met $1/s = \omega/\eta$ is de frequentieresolutie van het schalogram (figuur 3.9) beter dan die van het spectrogram (figuur 3.5) bij lage frequenties en slechter bij hogere frequenties.

Een andere toepassing van de continue wavelettransformatie is het meten van scherpe signaaltransities. Hiervoor gebruikt men reële wavelets. Zie bv. figuur 3.8.

Door de redundantie (zie (3.11)) volstaat het meestal om discrete waarden van de parameters u en s te gebruiken en bv. enkel positieve waarden voor s [17, 19]. Een veelgebruikte discretisatie is het dyadisch rooster ('dyadic grid'):

$$\begin{aligned} s &= 2^j \\ u &= ks \end{aligned}$$

met $j, k \in \mathbb{Z}$. Dit rooster werd gebruikt voor de tijd-frequentievlak indeling te zien op figuur 3.6 (b). Een wavelettransformatie die gebruik maakt van een dyadisch rooster is een voorbeeld van een *discrete wavelet transformatie*. Ook de gevensterde Fouriertransformatie kan gediscetiseerd worden.

Zowel bij de discrete gevensterde Fouriertransformatie als de discrete wavelettransformatie worden functies voorgesteld als superpositie van andere functies, respectievelijk de tijd-frequentie atomen $g_{u,\omega}$ en de wavelets $\psi_{s,u}$ voor discrete waarden van s , u en ω . De verzamelingen functies $\{g_{u,\omega}\}$ en $\{\psi_{s,u}\}$ vormen echter geen orthogonale basissen van $L^2(\mathbb{R})$ (ze zijn immers redundant!). Het zijn zogenaamde *veralgemeende tralies* (generalized frames) [17].

3.3 Multiresolutie analyse

In 1910 bewees Haar dat schalingen en verschuivingen van de stuksgewijs constante functie ψ

$$\psi(t) = \begin{cases} 1, & 0 \leq t < 1/2 \\ -1, & 1/2 \leq t < 1 \\ 0 & \text{elders} \end{cases} \quad (3.12)$$

een orthonormale basis van $L^2(\mathbb{R})$ opleveren:

$$\left\{ \psi_{j,n}(t) = \frac{1}{\sqrt{2^j}} \psi\left(\frac{t - 2^j n}{2^j}\right) \right\}_{(j,n) \in \mathbb{Z}^2}$$

Elke kwadratisch integreerbare functie kan ontbonden worden over deze basis:

$$f = \sum_{j=-\infty}^{+\infty} \sum_{n=-\infty}^{+\infty} \langle f, \psi_{j,n} \rangle \psi_{j,n}$$

(3.12) noemt men de Haar wavelet.

Y. Meyer, geïnspireerd door het werk van Morlet en Grossmann over de continue wavelettransformatie, probeerde te bewijzen dat er geen basis van regelmatige (regular) wavelets ψ bestaat die een orthonormale basis oplevert. Dit mislukte, want in 1986 construeerde hij een hele familie orthonormale waveletbasisen met functies ψ die oneindig continu afleidbaar zijn. Dit was de impuls die in 1988 leidde tot de beroemde Daubechies wavelets.

S. Mallat introduceerde het concept van de multiresolutie analyse, gebaseerd op ideeën uit de beeldbewerking, in de wavelettheorie. Dit leidde in 1989 tot een systematische theorie, ontwikkeld door Mallat en Meyer, voor het construeren van orthonormale waveletbasisen. Deze theorie had verrassende links met de filterbanktheorie en leidde tot een snel algoritme voor het berekenen van wavelettransformaties.

Waar de continue wavelettransformatie vooral nuttig is voor de karakterisatie (analyse) van functies, wordt de discrete multiresolutiebenadering vooral gebruikt voor het bewerken van signalen.

3.3.1 Multiresolutie benaderingen

De schaling functie en de ruimten V_j

Het idee achter een multiresolutie analyse is het benaderen van functies op verschillende resoluties 2^{-j} of schalen 2^j [16]. De benadering van een functie f op schaal 2^j wordt bepaald door een discrete set 'samples' van de functie, lokale gemiddelden over tijden (afstanden) evenredig met 2^j . Formeel: de benadering van f op schaal 2^j wordt gedefinieerd als de orthogonale projectie van f op een ruimte $V_j \subset L^2(\mathbb{R})$. V_j is de ruimte van alle mogelijke benaderingen op schaal 2^j . Zie figuur 3.10 voor een voorbeeld van een functie benaderd op verschillende schalen.

De definitie van een multiresolutie analyse is [16, 19]:

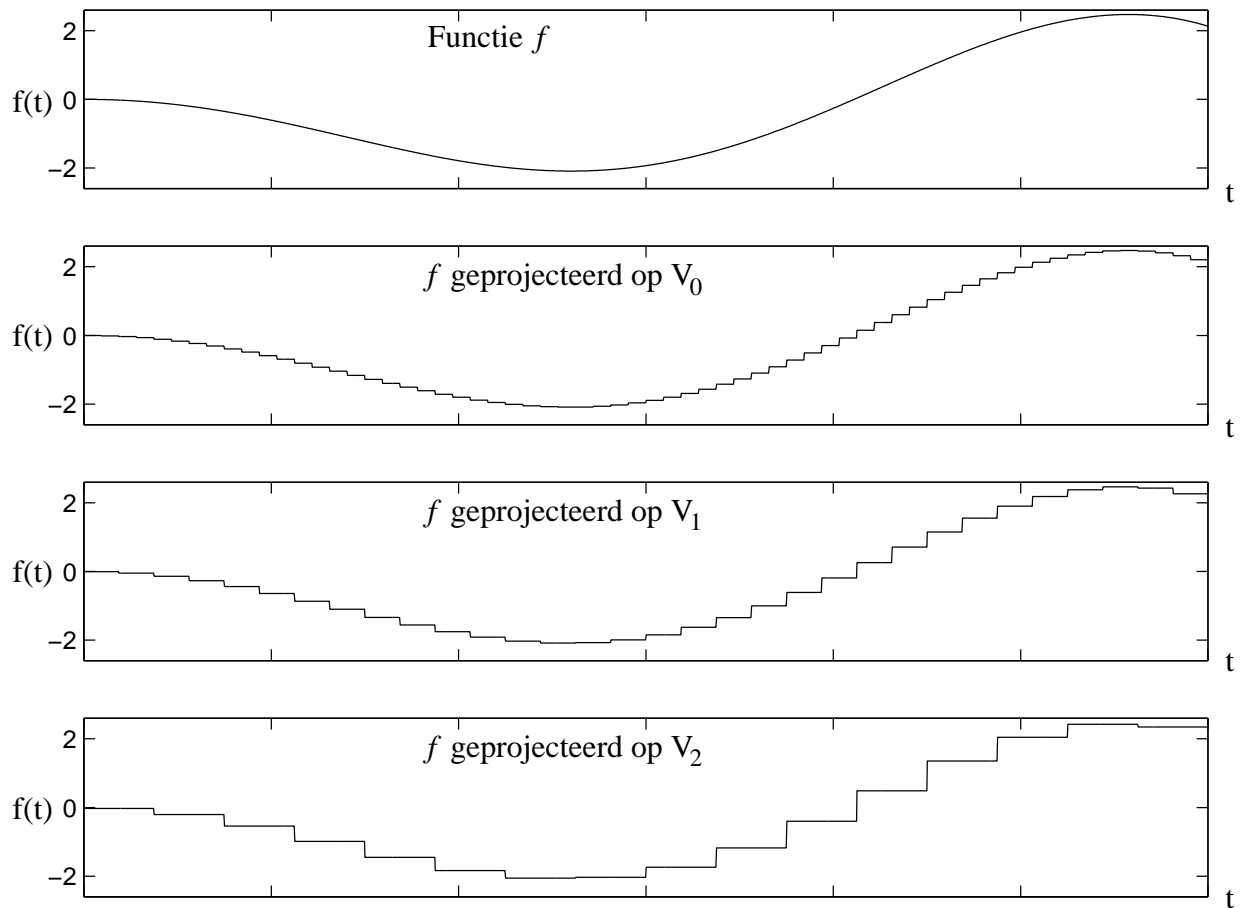
Definitie [Mallat] Een reeks $\{V_j\}_{j \in \mathbb{Z}}$ gesloten deelruimtes van $L^2(\mathbb{R})$ vormt een multiresolutie analyse als voldaan is aan volgende eigenschappen:

$$\forall j \in \mathbb{Z}, V_{j+1} \subset V_j \quad (3.13)$$

$$\forall j, k \in \mathbb{Z}, f(t) \in V_j \Leftrightarrow f(t - 2^j k) \in V_j \quad (3.14)$$

$$\forall j \in \mathbb{Z}, f(t) \in V_j \Leftrightarrow f\left(\frac{t}{2}\right) \in V_{j+1} \quad (3.15)$$

$$\lim_{j \rightarrow +\infty} V_j = \bigcap_{j=-\infty}^{+\infty} V_j = \{0\} \quad (3.16)$$



Figuur 3.10: Benadering van een functie f op verschillende schalen. De benaderingen zijn berekend met de Haar multiresolutie analyse.

$$\lim_{j \rightarrow -\infty} V_j = \text{Sluiting} \left(\bigcup_{j=-\infty}^{+\infty} V_j \right) = L^2(\mathbb{R}) \quad (3.17)$$

$$\text{Er bestaat een functie } \varphi \in V_0 \text{ zodat } \{\varphi(t-k)\}_{k \in \mathbb{Z}} \text{ een orthonormale basis is van } V_0. \quad (3.18)$$

We zullen de eigenschappen (3.13)-(3.18) kort bespreken:

- Eigenschap (3.13),

$$\forall j \in \mathbb{Z}, V_{j+1} \subset V_j,$$

toont aan dat een benadering op schaal 2^j alle informatie bevat die nodig is om een benadering op de grotere (groffere) schaal 2^{j+1} te berekenen¹.

- De tweede eigenschap (3.14),

$$\forall j, k \in \mathbb{Z}, f(t) \in V_j \Leftrightarrow f(t - 2^j k) \in V_j,$$

betekent dat een benadering $f_j \in V_j$ van f op schaal 2^j nog steeds een benadering op deze schaal is als we f_j verschuiven over een geheel aantal keer het 'samplinginterval' 2^j .

Een functie $f_j \in V_j$ met een factor 2 uitrekken verdeelt de informatie uit één 'samplinginterval' met breedte 2^j over twee zulke intervallen, met gezamenlijke breedte $2 \cdot 2^j = 2^{j+1}$.

- De derde eigenschap (3.15),

$$\forall j \in \mathbb{Z}, f(t) \in V_j \Leftrightarrow f\left(\frac{t}{2}\right) \in V_{j+1},$$

zorgt ervoor dat de uitgerokken versie van f_j kan geïnterpreteerd worden als een benadering op de groffere schaal 2^{j+1} .

- Eigenschap (3.16),

$$\lim_{j \rightarrow +\infty} V_j = \bigcap_{j=-\infty}^{+\infty} V_j = \{0\},$$

zegt dat als we de schaal waarop we een functie benaderen oneindig grof maken (of de resolutie oneindig laag), er geen informatie meer overblijft. (3.17),

$$\lim_{j \rightarrow -\infty} V_j = \text{Sluiting} \left(\bigcup_{j=-\infty}^{+\infty} V_j \right) = L^2(\mathbb{R}),$$

¹In de literatuur bestaat ook nog een andere conventie (met bijhorende definitie van de multiresolutie analyse), waarbij de benaderingen groffer worden voor afnemende index j (m.a.w. $V_j \subset V_{j+1}$) in plaats van voor toenemende j . Beide conventies worden ongeveer evenveel gebruikt.

daarentegen stelt dat als de benaderingsschaal van f oneindig fijn wordt, de benaderingsruimte V_j samenvalt met de ruimte $L^2(\mathbb{R})$. Daar $f \in L^2(\mathbb{R})$ is deze oneindig fijne benadering geen benadering meer, maar valt ze samen met de originele functie.

- In de laatste eigenschap (3.18),

Er bestaat een functie $\varphi \in V_0$ zodat $\{\varphi(t-k)\}_{k \in \mathbb{Z}}$,
een orthonormale basis is van V_0 .

noemt men φ de *scalingfunctie* (scaling function) of *vader wavelet* (father wavelet). Ondanks de benaming 'vader wavelet' is φ geen wavelet: $\int_{-\infty}^{+\infty} \varphi(t) dt \neq 0$. De restrictie dat $\{\varphi(t-k)\}_{k \in \mathbb{Z}}$ een orthonormale basis voor V_0 moet zijn is vrij streng. Ze kan verslapt worden tot de eis dat $\{\varphi(t-k)\}_{k \in \mathbb{Z}}$ een Riesz basis² vormt voor V_0 (zie [16, 19]). Omdat we enkel orthonormale basissen beschouwen zullen we echter (3.18) gebruiken. φ kan niet zomaar eender welke functie zijn. Uit de orthonormaliteitseigenschap (3.18) zien we dat

$$\langle \varphi, \varphi \rangle = \|\varphi\|^2 = 1 \quad (3.19)$$

De eigenschappen (3.18), (3.14) en (3.15) samen zorgen ervoor dat de familie functies

$$\left\{ \frac{1}{\sqrt{2^j}} \varphi \left(\frac{t-2^j k}{2^j} \right) \right\}_{k \in \mathbb{Z}} = \left\{ \frac{1}{\sqrt{2^j}} \varphi (2^{-j} t - k) \right\}_{k \in \mathbb{Z}} = \{\varphi_{j,k}(t)\}_{k \in \mathbb{Z}} \quad (3.20)$$

een orthonormale basis vormen voor V_j . Door de factor $\frac{1}{\sqrt{2^j}}$ is de $L^2(\mathbb{R})$ norm van de basisfuncties gelijk is aan 1 voor eender welke $j \in \mathbb{Z}$. Dit is nodig om de orthonormaliteit van de basis te behouden. We zullen verder nog enkele eigenschappen van φ bespreken.

De wavelet functie en de ruimten W_j

We weten dat de projectie f_j van f op V_j meer informatie bevat dan de projectie f_{j+1} op V_{j+1} . Uit (3.13) weten we ook dat $V_{j+1} \subset V_j$. Definieer nu W_{j+1} als het orthogonaal complement van V_{j+1} in V_j , zodat

$$V_j = V_{j+1} \oplus W_{j+1} \quad (3.21)$$

²Uit [16]: een familie functies $\{e_n\}_{n \in \mathbb{N}}$ vormt een Riesz basis voor een Hilbertruimte \mathcal{H} als ze lineair onafhankelijk is en er constanten $A > 0$ en $B > 0$ bestaan zodanig dat men voor eender welke $f \in \mathcal{H}$ getallen λ_n kan vinden zodanig dat

$$f = \sum_{n=0}^{+\infty} \lambda_n e_n$$

en

$$\frac{1}{B} \|f\|^2 \leq \sum_n |\lambda_n|^2 \leq \frac{1}{A} \|f\|^2$$

Het symbool \oplus betekent directe som: elk element van V_j kan op unieke wijze geschreven worden als de som van een element van V_{j+1} en één van W_{j+1} . De informatie die f_j wel bevat en f_{j+1} niet, moet dus in de projectie op W_{j+1} 'opgeslagen liggen'. W_{j+1} is het verschil tussen V_j en V_{j+1} , de projectie op deze ruimte bevat de details die we verliezen als we van een benadering op schaal 2^j overgaan naar één op schaal 2^{j+1} . Daarom noemt men de W_j soms de detailruimten.

Mallat en Meyer bewezen in 1989 dat we aan de schalingfunctie φ van de multiresolutieanalyse een waveletfunctie ψ kunnen associëren (zie verder) zodanig dat de familie wavelets

$$\left\{ \frac{1}{\sqrt{2^j}} \psi \left(\frac{t - 2^j k}{2^j} \right) \right\}_{k \in \mathbb{Z}} = \left\{ \frac{1}{\sqrt{2^j}} \psi \left(2^{-j} t - k \right) \right\}_{k \in \mathbb{Z}} = \{ \psi_{j,k}(t) \}_{k \in \mathbb{Z}} \quad (3.22)$$

een orthonormale basis vormen voor de ruimten W_j ($j \in \mathbb{Z}$).

(3.21) kunnen we iteratief toepassen:

$$\begin{aligned} V_j &= W_{j+1} \oplus V_{j+1} \\ &= W_{j+1} \oplus W_{j+2} \oplus V_{j+2} \\ &= W_{j+1} \oplus W_{j+2} \oplus \dots \oplus W_{j+n} \oplus V_{j+n} \end{aligned} \quad (3.23)$$

Voor $j \rightarrow -\infty$ en $n \rightarrow +\infty$ hebben we wegens (3.17) dus

$$L^2(\mathbb{R}) = \bigoplus_{l=-\infty}^{+\infty} W_l \quad (3.24)$$

$\{ \psi_{j,k}(t) \}_{j,k \in \mathbb{Z}}$ vormt dus een basis voor $L^2(\mathbb{R})$. Als we een grofste schaal N invoeren (m.a.w. we beperken n tot een vaste eindige waarde) wordt dit

$$L^2(\mathbb{R}) = V_N \oplus \bigoplus_{l=-\infty}^N W_l \quad (3.25)$$

De projectieoperatoren P_j en Q_j

Beschouw een functie $f_j \in V_j$. f_j is dus een superpositie van basisfuncties $\varphi_{j,k}$ van V_j :

$$f_j(t) = \sum_{k=-\infty}^{+\infty} s_{jk} \varphi_{j,k}(t)$$

Omdat de $\{ \varphi_{j,k}(t) \}_{k \in \mathbb{Z}}$ een orthonormale basis vormen kunnen we de s_{jk} berekenen door in beide leden van deze vergelijking het inwendig product te nemen met $\varphi_{j,k'}$:

$$\begin{aligned} \langle f_j, \varphi_{j,k'} \rangle &= \sum_{k=-\infty}^{+\infty} s_{jk} \langle \varphi_{j,k}, \varphi_{j,k'} \rangle \\ &= \sum_{k=-\infty}^{+\infty} s_{jk} \delta_{kk'} \\ &= s_{jk'} \end{aligned}$$

De ontbinding van f_j wordt dus

$$f_j(t) = \sum_{k=-\infty}^{+\infty} \langle f_j, \varphi_{j,k} \rangle \varphi_{j,k}(t)$$

Naar analogie hiermee wordt de orthogonale projectie $P_j f$ van een functie f (waarvoor in het algemeen geldt $f \notin V_j$) op V_j gegeven door:

$$P_j f(t) = \sum_{k=-\infty}^{+\infty} \langle f, \varphi_{j,k} \rangle \varphi_{j,k}(t) \quad (3.26)$$

De projectieoperator P_j toegepast op f levert dus een benadering van f op schaal 2^j .

Op dezelfde manier kunnen we Q_j invoeren, de orthogonale projectieoperator die op W_j projecteert:

$$Q_j f(t) = \sum_{k=-\infty}^{+\infty} \langle f, \psi_{j,k} \rangle \psi_{j,k}(t) \quad (3.27)$$

Uit (3.21) zien we dat er steeds moet gelden

$$P_j = P_{j+1} + Q_{j+1} \quad (3.28)$$

Omwillen van (3.24) kunnen we elke $f \in L^2(\mathbb{R})$ schrijven als

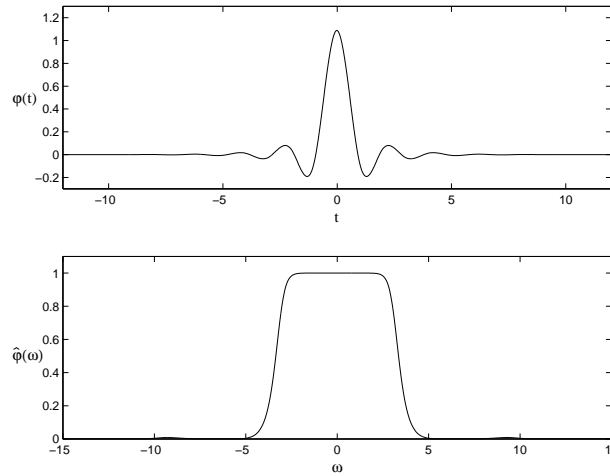
$$\begin{aligned} f &= \sum_{j=-\infty}^{+\infty} Q_j f \\ &= \sum_{j=-\infty}^{+\infty} \sum_{k=-\infty}^{+\infty} \langle f, \psi_{j,k} \rangle \psi_{j,k} \end{aligned} \quad (3.29)$$

Als we een grofste schaal N invoeren hebben we omwillen van (3.25)

$$\begin{aligned} f &= \sum_{j=-\infty}^N Q_j f + P_N f \\ &= \sum_{j=-\infty}^N \sum_{k=-\infty}^{+\infty} \langle f, \psi_{j,k} \rangle \psi_{j,k} + \sum_{k=-\infty}^{+\infty} \langle f, \varphi_{N,k} \rangle \varphi_{N,k} \end{aligned} \quad (3.30)$$

(3.29) en (3.30) zijn waveletontwikkelingen van f . We zullen naar de coëfficiënten $d_{jk} = \langle f, \psi_{j,k} \rangle$ refereren als wavelet- of detailcoëfficiënten. De $s_{Nk} = \langle f, \varphi_{N,k} \rangle$ zullen we schaal- of schalingcoëfficiënten noemen.

Voor praktische berekeningen zijn (3.29) en (3.30) niet geschikt. Daarvoor moeten we ook een fijnste schaal invoeren. Bij discrete computerberekeningen is f namelijk nooit gekend met een oneindig grote resolutie, de resolutie wordt bepaald door het samplinginterval. Dit



Figuur 3.11: Kubische spline schaling functie φ (boven) en de Fouriertransformatie $\hat{\varphi}$ (onder). Figuur overgenomen uit [16].

interval zal de fijnste schaal bepalen. Als we het samplinginterval normaliseren tot 1 is V_0 de fijnste benaderingsruimte die we beschouwen (zie hiervoor sectie 4.2). Als we K samples hebben van f wordt (3.30)

$$\begin{aligned}
 P_0 f &= \sum_{j=1}^N Q_j f + P_N f & (3.31) \\
 &= \sum_{j=1}^N \sum_{k=1}^K \langle f, \psi_{j,k} \rangle \psi_{j,k} + \sum_{k=1}^K \langle f, \varphi_{N,k} \rangle \varphi_{N,k}
 \end{aligned}$$

We schrijven hier $P_0 f$ omdat we beginnen met een benadering van f op schaal 2^0 . Deze formule is praktisch bruikbaar. Als K een macht van 2 is kunnen de coëfficiënten $\langle f, \psi_{j,k} \rangle$ en $\langle f, \varphi_{N,k} \rangle$ berekend worden met een snel filterbank algoritme.

Omdat benaderingen op schaal 2^j overeenkomen met projecties op de ruimte V_j , zullen we in het vervolg de 2 niet meer schrijven. We spreken vanaf nu dus over schaal j in plaats van 2^j .

De functie φ

We bekijken de schalingfunctie φ nu wat meer in detail. Zie figuur 3.11 voor een voorbeeld.

Uit (3.20) weten we dat de functies

$$\left\{ \frac{1}{\sqrt{2^j}} \varphi(2^{-j}t - k) \right\}_{k \in \mathbb{Z}} \Big|_{j=-1} = \left\{ \sqrt{2} \varphi(2t - k) \right\}_{k \in \mathbb{Z}} \quad (3.32)$$

een orthonormale basis vormen voor V_{-1} . Wegens (3.13) en (3.18) geldt er ook $\varphi \in V_0 \subset V_{-1}$. Dit betekent dat we φ kunnen ontbinden in basisfuncties van V_{-1} :

$$\varphi(t) = \sqrt{2} \sum_{k=-\infty}^{+\infty} h_k \varphi(2t - k) \quad (3.33)$$

met

$$h_k = \langle \varphi(t), \varphi(2t - k) \rangle \quad (3.34)$$

Betrekking (3.33) staat bekend als de *verfijningsvergelijking* (refinement equation, ook dilation equation, two-scale difference equation of scaling equation). In sommige publicaties ([19] bijvoorbeeld) gebruikt men de factor 2 in plaats van $\sqrt{2}$. Dit zal andere waarden voor de h_k opleveren.

Als de functie φ absoluut integreerbaar is ($\varphi \in L^1(\mathbb{R})$) dan wordt ze onder zeer algemene voorwaarden volledig bepaald door de verfijningsvergelijking (3.33), samen met de normalisatie [19]:

$$\int_{-\infty}^{+\infty} \varphi(t) dt = 1 \quad (3.35)$$

Beide leden van (3.33) integreren over alle t levert wegens (3.35)

$$\sum_{k=-\infty}^{+\infty} h_k = \sqrt{2} \quad (3.36)$$

Stel dat we een constante functie $f \equiv c$ projecteren op V_0 :

$$\begin{aligned} P_0 f &= \sum_{k=-\infty}^{+\infty} \langle f, \varphi_{0,k} \rangle \varphi_{0,k}(t) \\ &= \sum_{k=-\infty}^{+\infty} \langle f(t), \varphi(t-k) \rangle \varphi(t-k) \end{aligned}$$

Daar $\langle f(t), \varphi(t-k) \rangle = c$ (wegens (3.35)) hebben we dus

$$P_0 f = \sum_{k=-\infty}^{+\infty} c \varphi(t-k)$$

Het is geen onredelijke eis dat zeer simpele functies als constanten exact kunnen benaderd worden ($P_0 f = c$). Daarom is het logisch te veronderstellen dat φ en haar translaties over gehele tijden een zogenaamde partitie van de eenheid (partition of unity) vormen [19]:

$$\forall t \in \mathbb{R} : \sum_{k=-\infty}^{+\infty} \varphi(t-k) = 1$$

Berekenen we van beide leden van (3.33) de Fouriertransformatie dan hebben we

$$\begin{aligned}
\hat{\varphi}(\omega) &= \sqrt{2} \sum_{k=-\infty}^{+\infty} h_k \int_{-\infty}^{+\infty} \varphi(2t-k) e^{-i\omega t} dt \\
&\Downarrow u = 2t - k \\
&= \sqrt{2} \sum_{k=-\infty}^{+\infty} h_k \frac{1}{2} \int_{-\infty}^{+\infty} \varphi(u) e^{-i\omega(u+k)/2} du \\
&= \frac{1}{\sqrt{2}} \sum_{k=-\infty}^{+\infty} h_k e^{-i\frac{\omega}{2}k} \int_{-\infty}^{+\infty} \varphi(u) e^{-i\frac{\omega}{2}u} du \\
&= \frac{1}{\sqrt{2}} \hat{h}\left(\frac{\omega}{2}\right) \hat{\varphi}\left(\frac{\omega}{2}\right)
\end{aligned} \tag{3.37}$$

met de 2π -periodieke functie

$$\hat{h}(\omega) = \sum_{k=-\infty}^{+\infty} h_k e^{-i\omega k} \tag{3.38}$$

$\hat{\varphi}(\omega) / \hat{\varphi}(\frac{\omega}{2})$ moet dus periodiek zijn met periode 2π .

Uit (3.37) blijkt dat we de coëfficiënten h_k kunnen interpreteren als een discreet filter $h = \{h_k\}$ met als transferfunctie $\hat{h}(\omega)$. Merk op dat we (3.36) met (3.38) kunnen herschrijven als

$$\hat{h}(0) = \sqrt{2}$$

(3.37) kunnen we iteratief toepassen. Omdat $\hat{h}(0) = \sqrt{2} \neq 0$ hebben we

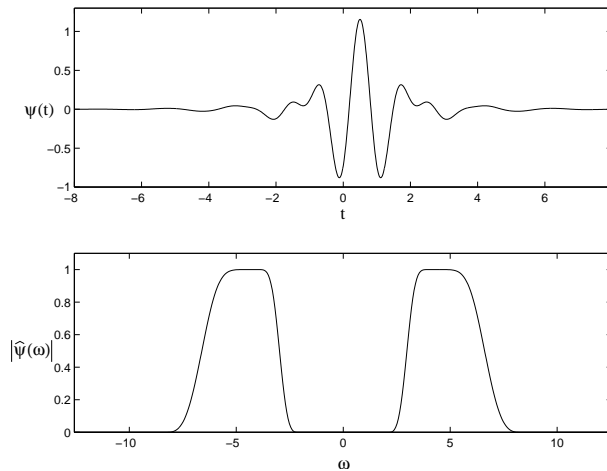
$$\begin{aligned}
\hat{\varphi}(\omega) &= \prod_{p=1}^{+\infty} \frac{\hat{h}(\frac{\omega}{2^p})}{\sqrt{2}} \cdot \hat{\varphi}(0) \\
&= \prod_{p=1}^{+\infty} \frac{\hat{h}(\frac{\omega}{2^p})}{\sqrt{2}}
\end{aligned} \tag{3.39}$$

want $\hat{\varphi}(0) = 1$ wegens (3.35). Dit kan enkel geldig zijn als $\hat{\varphi}(\omega)$ continu is in de buurt van $\omega = 0$. Zie [16] voor de exacte voorwaarden voor de geldigheid en convergentie van (3.39). Daar wordt ook bewezen dat steeds moet gelden

$$\forall \omega \in \mathbb{R} : \left| \hat{h}(\omega) \right|^2 + \left| \hat{h}(\omega + \pi) \right|^2 = 2 \tag{3.40}$$

opdat we een geldige schalingfunctie φ zouden hebben. Discrete filters waarvan de transferfuncties voldoen aan (3.40) noemt men in het Engels *conjugate mirror filters* [16].

In veel gevallen is er geen expliciete uitdrukking voor φ beschikbaar. Er bestaan wel snelle algoritmen om de waarde van $\varphi(t)$ te berekenen op de roosterpunten van een dyadisch rooster ($t = 2^j k$, met $j, k \in \mathbb{Z}$). Meestal is een exacte uitdrukking voor φ echter niet nodig en kunnen we rechtstreeks met de coëfficiënten h_k werken.



Figuur 3.12: Meyer wavelet ψ (boven) en $|\hat{\psi}|$ (onder). Figuur overgenomen uit [16].

De functie ψ

In figuur 3.12 staat een waveletfunctie ψ afgebeeld. We zullen de eigenschappen van ψ nader bekijken.

Uit (3.21) weten we dat

$$V_{-1} = V_0 + W_0,$$

en dus $W_0 \subset V_{-1}$. De familie functies (3.32) vormt een orthonormale basis voor V_{-1} . $\psi \in V_{-1}$ kan dus ontbonden worden in deze basis:

$$\psi(t) = \sqrt{2} \sum_{k=-\infty}^{+\infty} g_k \varphi(2t - k) \quad (3.41)$$

met $g_k = \langle \psi(t), \varphi(2t - k) \rangle$. Net zoals bij de schalingfunctie hebben we

$$\hat{\psi}(\omega) = \frac{1}{\sqrt{2}} \hat{g}\left(\frac{\omega}{2}\right) \hat{\varphi}\left(\frac{\omega}{2}\right) \quad (3.42)$$

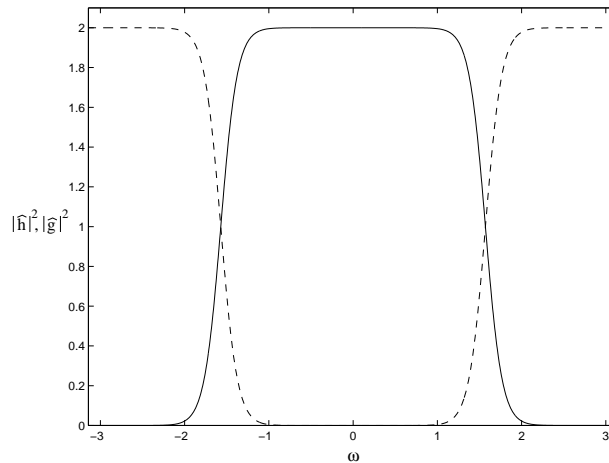
met

$$\hat{g}(\omega) = \sum_{k=-\infty}^{+\infty} g_k e^{-i\omega k}$$

Mallat en Meyer bewezen (zie o.a. [16]) dat \hat{g} ook gegeven wordt door

$$\hat{g}(\omega) = e^{-i\omega} \overline{\hat{h}(\omega + \pi)} \quad (3.43)$$

Dit toont aan dat de coëfficiënten g_k (en dus de waveletfunctie ψ) volledig bepaald worden door de schalingfunctie φ , of equivalent, door de coëfficiënten h_k .



Figuur 3.13: Transferfuncties van de conjugate mirror filters \hat{h} en \hat{g} geassocieerd aan de kubische spline schalingfunctie uit figuur 3.11 . Volle lijn: $|\hat{h}|^2$, stippellijn: $|\hat{g}|^2$. Figuur overgenomen uit [16].

Net als voor \hat{h} geldt voor \hat{g} [16]

$$\forall \omega \in \mathbb{R} : |\hat{g}(\omega)|^2 + |\hat{g}(\omega + \pi)|^2 = 2$$

De coëfficiënten g_k vormen dus ook een conjugate mirror filter $g = \{g_k\}$, de spiegelfilter van h . Zie figuur (3.13) voor een voorbeeld van de transferfuncties \hat{h} en \hat{g} . We zien dat h een laagdoorlaatfilter is en g een hoogdoorlaat.

(3.43) laat toe de coëfficiënten g_k te bepalen. Dat gaat als volgt: uit (3.42) en (3.43) weten we dat

$$\hat{\psi}(\omega) = \frac{1}{\sqrt{2}} e^{-i\frac{\omega}{2}} \overline{\hat{h}\left(\frac{\omega}{2} + \pi\right)} \hat{\varphi}\left(\frac{\omega}{2}\right)$$

(3.38) invullen geeft

$$\begin{aligned} \hat{\psi}(\omega) &= \frac{1}{\sqrt{2}} e^{-i\frac{\omega}{2}} \sum_{k=-\infty}^{+\infty} \overline{h_k} e^{i(\frac{\omega}{2} + \pi)k} \hat{\varphi}\left(\frac{\omega}{2}\right) \\ &= \frac{1}{\sqrt{2}} \sum_{k=-\infty}^{+\infty} (-1)^k \overline{h_k} e^{i\frac{\omega}{2}(k-1)} \hat{\varphi}\left(\frac{\omega}{2}\right) \end{aligned}$$

Beide leden invers Fouriertransformeren levert

$$\begin{aligned} \psi(t) &= \frac{1}{\sqrt{2}} \sum_{k=-\infty}^{+\infty} (-1)^k \overline{h_k} \frac{1}{2\pi} \int_{-\infty}^{+\infty} \hat{\varphi}\left(\frac{\omega}{2}\right) e^{i\frac{\omega}{2}(2t+k-1)} d\omega \\ &= \frac{2}{\sqrt{2}} \sum_{k=-\infty}^{+\infty} (-1)^k \overline{h_k} \varphi(2t+k-1) \end{aligned}$$

Als we nu $n = 1 - k$ stellen krijgen we

$$\psi(t) = \sqrt{2} \sum_{n=-\infty}^{+\infty} (-1)^{1-n} \overline{h_{1-n}} \varphi(2t - n)$$

Vergelijken met (3.41) leert dat de coëfficiënten g_k gegeven worden door

$$g_k = (-1)^{1-k} \overline{h_{1-k}}$$

3.3.2 De snelle wavelet transformatie

Principe

Veronderstel dat we de benadering $P_j f = f_j$ van f op schaal j kennen. We kennen dus de coëfficiënten $s_{jk} = \langle f, \varphi_{j,k} \rangle$ zodat

$$f_j = \sum_{k=-\infty}^{+\infty} \langle f, \varphi_{j,k} \rangle \varphi_{j,k}$$

Om de waveletontbinding van f_j tot op een grofste schaal N te vinden moeten we f schrijven in de basis behorende bij de directe som (zie (3.23))

$$W_j \oplus W_{j+1} \oplus W_{j+2} \oplus \dots \oplus W_N \oplus V_N$$

De snelle wavelet transformatie (fast wavelet transform) is een algoritme, recursief toepasbaar, om de basisverandering

$$V_j \rightarrow V_{j+1} \oplus W_{j+1}$$

snel te berekenen. We willen f_j schrijven als een lineaire combinatie van de basisfuncties

$$\{\varphi_{j+1,k}\}_{k \in \mathbb{Z}} \cup \{\psi_{j+1,k}\}_{k \in \mathbb{Z}},$$

dus (zie (3.28), (3.26) en (3.27))

$$\begin{aligned} f_j(t) = P_j f_j(t) &= P_{j+1} f_j(t) + Q_{j+1} f_j(t) \\ &= \sum_{k=-\infty}^{+\infty} \langle f_j, \varphi_{j+1,k} \rangle \varphi_{j+1,k}(t) + \sum_{k=-\infty}^{+\infty} \langle f_j, \psi_{j+1,k} \rangle \psi_{j+1,k}(t) \end{aligned}$$

We moeten de schaal- en detailcoëfficiënten $s_{j+1,k} = \langle f_j, \varphi_{j+1,k} \rangle$ en $d_{j+1,k} = \langle f_j, \psi_{j+1,k} \rangle$ op schaal $j+1$ berekenen, uitgaande van de schaalcoëfficiënten $s_{j,k} = \langle f, \varphi_{j,k} \rangle = \langle f_j, \varphi_{j,k} \rangle$ op schaal j .

We beginnen met de $\langle f_j, \varphi_{j+1,k} \rangle$:

$$\begin{aligned} \langle f_j, \varphi_{j+1,k} \rangle &= \left\langle \sum_{l=-\infty}^{+\infty} \langle f, \varphi_{j,l} \rangle \varphi_{j,l}, \varphi_{j+1,k} \right\rangle \\ &= \sum_{l=-\infty}^{+\infty} s_{j,l} \langle \varphi_{j,l}, \varphi_{j+1,k} \rangle \\ &= \sum_{l=-\infty}^{+\infty} s_{j,l} \int_{-\infty}^{+\infty} \frac{1}{\sqrt{2^j}} \varphi(2^{-j}t-l) \frac{1}{\sqrt{2^{j+1}}} \overline{\varphi(2^{-(j+1)}t-k)} dt \end{aligned}$$

Stellen we nu $t' = 2^{-(j+1)}t - k$ dan wordt dit

$$\begin{aligned} \langle f_j, \varphi_{j+1,k} \rangle &= \sum_{l=-\infty}^{+\infty} s_{j,l} \int_{-\infty}^{+\infty} \varphi(2t' - (l-2k)) \sqrt{2} \overline{\varphi(t')} dt' \\ &= \sum_{l=-\infty}^{+\infty} s_{j,l} \langle \varphi(2t' - (l-2k)), \varphi(t') \rangle \end{aligned}$$

Gebruik makend van (3.34) hebben we

$$\begin{aligned} \langle \varphi(2t' - (l-2k)), \varphi(t') \rangle &= \overline{\langle \varphi(t'), \varphi(2t' - (l-2k)) \rangle} \\ &= \overline{h_{l-2k}} \end{aligned}$$

zodat

$$\langle f_j, \varphi_{j+1,k} \rangle = s_{j+1,k} = \sum_{l=-\infty}^{+\infty} \overline{h_{l-2k}} s_{j,l} \quad (3.44)$$

Volledig analoog geldt er

$$\langle f_j, \psi_{j+1,k} \rangle = d_{j+1,k} = \sum_{l=-\infty}^{+\infty} \overline{g_{l-2k}} s_{j,l} \quad (3.45)$$

Formules (3.44) en (3.45) vormen samen één stap uit de voorwaartse wavelettransformatie. De $s_{j+1,k}$ vormen de input voor de volgende stap, enz.

Bij de inverse wavelettransformatie staan we voor het omgekeerde probleem. We kennen de projectie van f_j op de ruimten V_{j+1} en W_{j+1} , m.a.w. we kennen de coëfficiënten $s_{j+1,k} = \langle f_j, \varphi_{j+1,k} \rangle$ en $d_{j+1,k} = \langle f_j, \psi_{j+1,k} \rangle$, en we willen de projectie op $V_j = V_{j+1} \oplus W_{j+1}$ bepalen:

$$f_j = \sum_{k=-\infty}^{+\infty} \langle f_j, \varphi_{j,k} \rangle \varphi_{j,k},$$

We moeten de coëfficiënten $\langle f_j, \varphi_{j,k} \rangle$ berekenen:

$$\begin{aligned} \langle f_j, \varphi_{j,k} \rangle &= \left\langle \sum_{l=-\infty}^{+\infty} s_{j+1,l} \varphi_{j+1,l} + \sum_{l=-\infty}^{+\infty} d_{j+1,l} \psi_{j+1,l}, \varphi_{j,k} \right\rangle \\ &= \sum_{l=-\infty}^{+\infty} s_{j+1,l} \langle \varphi_{j+1,l}, \varphi_{j,k} \rangle + \sum_{l=-\infty}^{+\infty} d_{j+1,l} \langle \psi_{j+1,l}, \varphi_{j,k} \rangle \end{aligned}$$

Nu geldt (dit hebben we bij de voorwaartse wavelettransformatie aangetoond)

$$\begin{aligned} \langle \varphi_{j+1,l}, \varphi_{j,k} \rangle &= \overline{\langle \varphi_{j,k}, \varphi_{j+1,l} \rangle} \\ &= \overline{h_{k-2l}} = h_{k-2l} \\ \langle \psi_{j+1,l}, \varphi_{j,k} \rangle &= \overline{\langle \varphi_{j,k}, \psi_{j+1,l} \rangle} \\ &= \overline{g_{k-2l}} = g_{k-2l} \end{aligned}$$

zodat we voor de inverse wavelettransformatie krijgen

$$\langle f_j, \varphi_{j,k} \rangle = s_{j,k} = \sum_{l=-\infty}^{+\infty} h_{k-2l} s_{j+1,l} + \sum_{l=-\infty}^{+\infty} g_{k-2l} d_{j+1,l} \quad (3.46)$$

De complexiteit van de voorwaartse ((3.44) en (3.45)) en inverse wavelettransformatie (3.46) is $O(N)$ (met N het aantal schalingcoëfficiënten waarmee we beginnen) voor eindige impulsrespons filters (FIR filters) h en g [16]. Daarom noemt men het de snelle wavelet transformatie.

Filterbank interpretatie

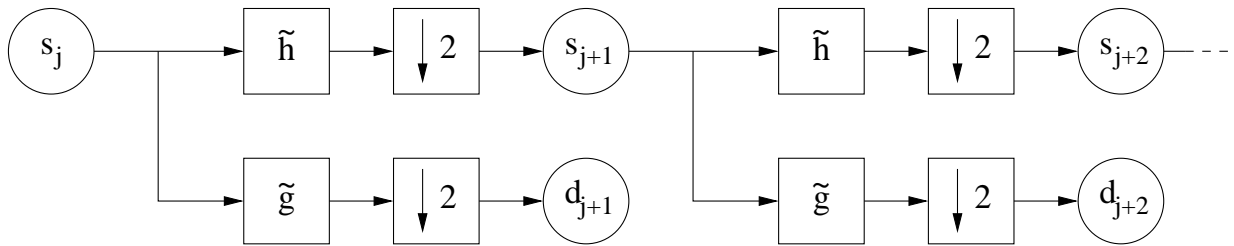
Als we twee reeksen $a = \{a_k\}$ en $b = \{b_k\}$ beschouwen ($k \in \mathbb{Z}$) dan noemt men de reeks $a \star b = \{(a \star b)_k\}$ gedefinieerd door

$$(a \star b)_k = \sum_{l=-\infty}^{+\infty} a_l b_{k-l}$$

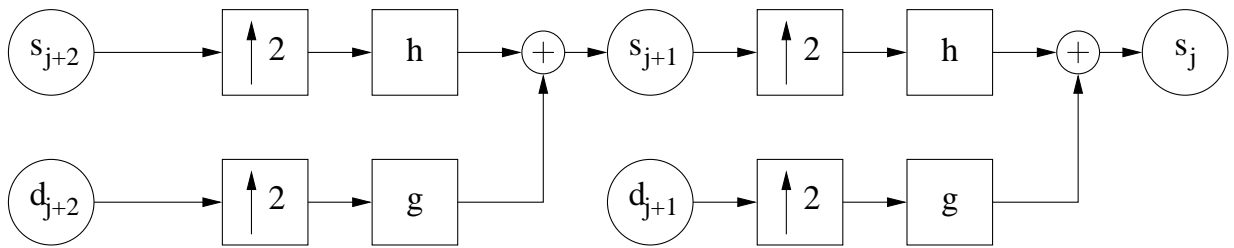
de convolutie van a en b . Als we (3.44) bekijken in het licht van deze definitie dan zien we dat

$$s_{j+1,k} = (s_j \star \tilde{h})_{2k}$$

met $s_{j+1} = \{s_{j+1,k}\}$, $s_j = \{s_{j,k}\}$ en $\tilde{h} = \{\tilde{h}_k = \overline{h_{-k}}\}$. s_{j+1} is dus de reeks die men bekomt door s_j te convolueren met \tilde{h} en van dat resultaat enkel elementen met even index bij te houden. Men zegt dat s_{j+1} een gedownsamplde versie is van $s_j \star \tilde{h}$.



Figuur 3.14: Twee stappen uit de voorwaartse wavelet transformatie. $\downarrow 2$ betekent downsamenen.



Figuur 3.15: Twee stappen uit de inverse wavelet transformatie. $\uparrow 2$ betekent upsamenen.

Volledig analoog hebben we voor (3.45)

$$d_{j+1,k} = (s_j \star \tilde{g})_{2k}$$

Deze filterinterpretatie van de voorwaartste wavelet transformatie is afgebeeld in figuur 3.14.

Ook de inverse wavelettransformatie (3.46) kunnen we als een convolutie schrijven. Noemen we $\hat{s}_{j+1} = \{\hat{s}_{j+1,k}\}$ en $\hat{d}_{j+1} = \{\hat{d}_{j+1,k}\}$ de geupsamplende reeksen s_{j+1} en d_{j+1} met als elementen

$$\hat{s}_{j+1,k} = \begin{cases} s_{j+1,p} & k = 2p \\ 0 & k = 2p + 1 \end{cases}$$

$$\hat{d}_{j+1,k} = \begin{cases} d_{j+1,p} & k = 2p \\ 0 & k = 2p + 1 \end{cases}$$

Dan wordt (3.46)

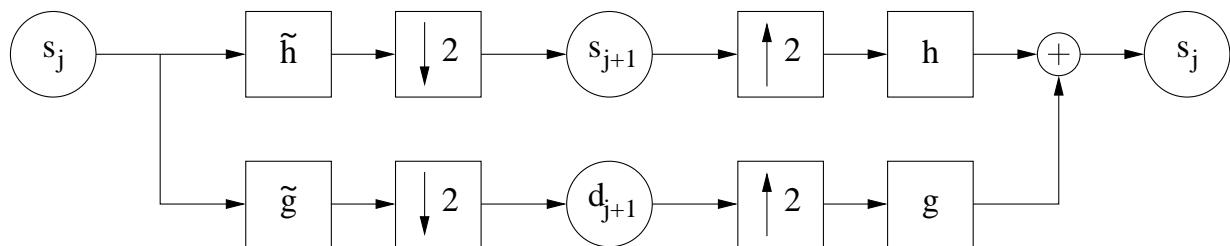
$$s_{j,k} = \sum_{l=-\infty}^{+\infty} h_{k-l} \hat{s}_{j+1,l} + \sum_{l=-\infty}^{+\infty} g_{k-l} \hat{d}_{j+1,l}$$

of

$$s_j = \hat{s}_{j+1} \star h + \hat{d}_{j+1} \star h$$

Dit is geïllustreerd in figuur 3.15.

De voorwaartse en inverse wavelettransformatie vormen samen een filterbank met perfecte reconstructie (zie figuur 3.16).



Figuur 3.16: Filterbank gevormd door de voorwaartse en inverse wavelettransformatie.

3.3.3 Eindig aantal coëfficiënten

Wegens de up- en downsamplings met factor 2 in de filterbank interpretatie zien we dat bij praktische berekeningen (met een eindig aantal begincoëfficiënten) het aantal schalingcoëfficiënten waarmee we beginnen steeds een macht van 2 moet zijn.

Bij een eindig aantal coëfficiënten duiken er ook aan de randen problemen op. Stel dat we N schalingcoëfficiënten s_k hebben van een functie f benaderd op schaal 0 en dat het samplinginterval T gelijk is aan 1. Als we de convoluties met de filters \tilde{h} en \tilde{g} willen berekenen in de buurt van de eerste of de laatste coëfficiënt hebben we coëfficiënten nodig die 'over de rand' liggen, coëfficiënten met indices kleiner dan 1 of groter dan N .

Er bestaan verschillende mogelijkheden om dit probleem te verhelpen, zoals folding of het gebruik van randwavelets [16]. De simpelste oplossing is echter het periodiek maken van de s_k . Deze methode zullen wij gebruiken. Het grootste nadeel ervan is dat als de waarden van f in de buurt de linkerrand van het beschouwde interval sterk verschillen van die aan de rechterrind er dicht bij de randen grote waveletcoëfficiënten gegenereerd worden.

3.3.4 Keuze van de wavelet basis

De meeste toepassingen van wavelets zijn gebaseerd op het feit dat ze in staat zijn bepaalde klassen functies efficiënt te benaderen met een klein aantal van nul verschillende coëfficiënten. Dit is niet enkel zo in datacompressie, maar ook bij bv. snelle numerieke algoritmen.

Een functie f zal weinig grote wavelet coëfficiënten hebben als de meeste van die coëfficiënten op de fijnste schaal klein zijn. De voornaamste eigenschappen van de wavelet ψ die hierin een rol spelen zijn het aantal momenten die nul zijn (vanishing moments) en de grootte van het gebied waarin $\psi(t) \neq 0$ (de draagwijdte of 'support').

Nulmomenten

ψ heeft p nulmomenten als

$$\int_{-\infty}^{+\infty} t^k \psi(t) dt = 0 \text{ voor } 0 \leq k \leq p$$

Het belang van de nulmomenten kunnen we als volgt inzien: beschouw de Taylorreeks van een functie $f(t)$ rond 0:

$$\begin{aligned} f(t) &= f(0) + \left. \frac{df}{dt} \right|_{t=0} t + \frac{1}{2} \left. \frac{d^2f}{dt^2} \right|_{t=0} t^2 + \dots \\ &= c_0 + c_1 t + c_2 t^2 + \dots \end{aligned}$$

met $c_n = \frac{1}{n!} \left. \frac{d^n f}{dt^n} \right|_{t=0}$. De waveletcoëfficiënten op schaal 0 worden gegeven door

$$\langle f(t), \psi(t-k) \rangle = \langle c_0, \psi(t-k) \rangle + \langle c_1 t, \psi(t-k) \rangle + \langle c_2 t^2, \psi(t-k) \rangle + \dots$$

Als ψ p nulmomenten heeft zullen de eerste p termen van deze som nul zijn. Voor veel functies worden de Taylor coëfficiënten c_k klein als k groter wordt. Omdat de waveletcoëfficiënten enkel afhankelijk van de Taylor coëfficiënten c_k met $k \geq p$ zullen ze voor zulke functies erg klein zijn als p groot is.

Draagwijdte

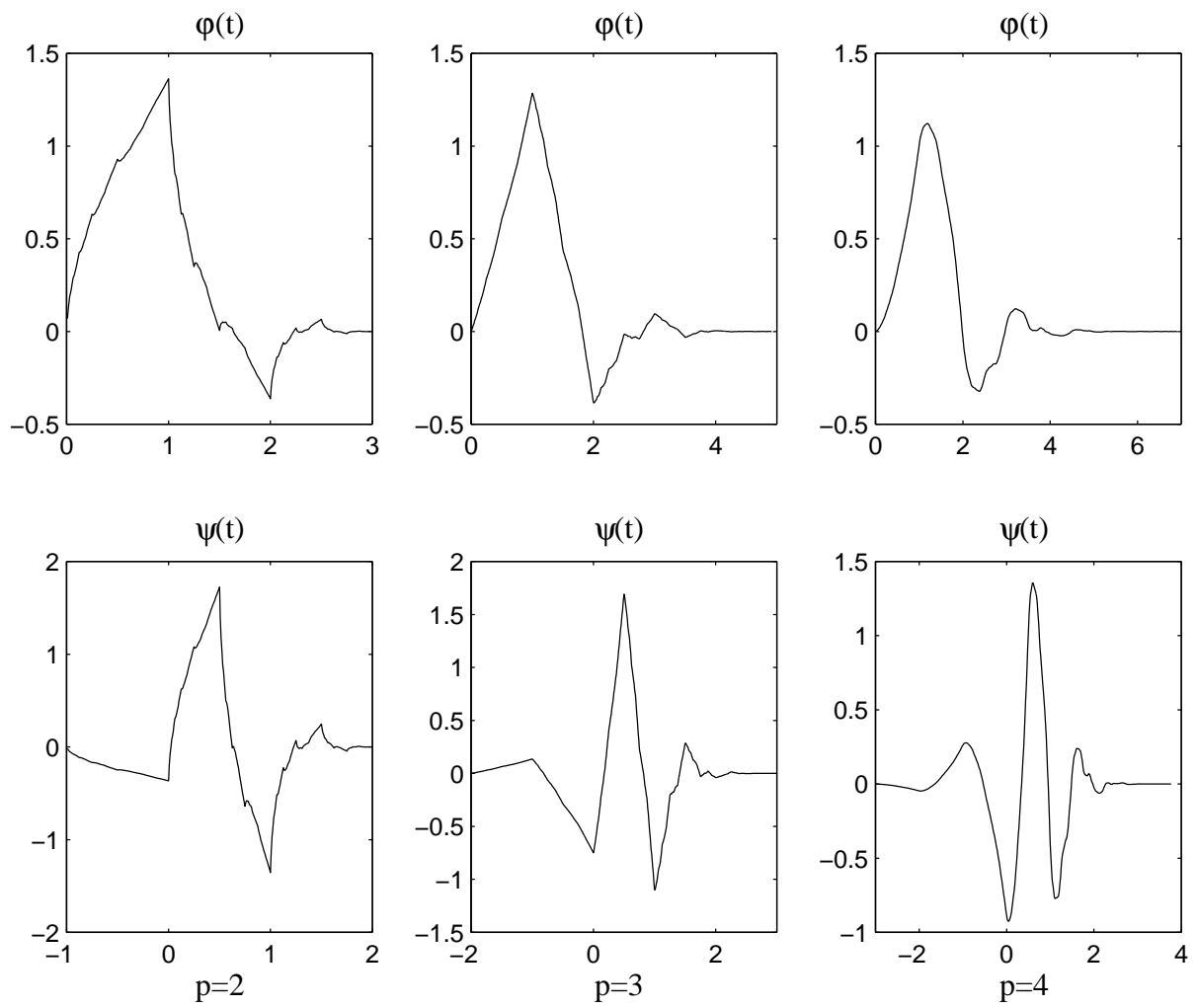
Wavelets $\psi(t)$ die enkel verschillend van nul zijn binnen een eindig aantal eindige intervallen noemt men wavelets met compacte drager (compact support). Meestal is er maar één zo'n interval en de draagwijdte (support) is dan de grootte van dit interval.

Stel dat een functie $f(t)$ een geïsoleerde singulariteit heeft op $t = t_0$. Als t_0 binnen de draagwijdte ligt van de wavelet $\psi_{j,k}(t)$ dan kan de coëfficiënt $\langle f, \psi_{j,k} \rangle$ groot zijn. Als de draagwijdte van $\psi_{j,k}$ T is, zijn er op elke schaal j T wavelets die t_0 binnen hun drager hebben [16]. We kunnen het aantal grote coëfficiënten dus laag houden door de draagwijdte van ψ klein te kiezen.

Voor orthogonale wavelets met p nulmomenten is de minimale draagwijdte $2p - 1$ [16]. Er is dus een compromis nodig tussen het aantal nulmomenten en de draagwijdte. Als een functie f weinig geïsoleerde singulariteiten heeft en glad (smooth) is tussen de singulariteiten kunnen we best een wavelet met veel nulmomenten kiezen. Als het aantal singulariteiten groter wordt kan het voordeliger zijn om de draagwijdte te verkleinen (minder overlapping met de singulariteiten) en dus voor minder nulmomenten te kiezen.

De wavelets met minimale draagwijdte voor een gegeven aantal nulmomenten zijn de Daubechies wavelets (zie figuur 3.17).

Buiten de Haar wavelet zijn er geen wavelets met compacte drager waarvoor een expliciete uitdrukking bestaat. Tekeningen van deze functies zijn bekomen aan de hand van hun filtercoëfficiënten.



Figuur 3.17: Daubechies schaling functies φ en wavelet functies ψ met p nulmomenten. Figuur overgenomen uit [16].

3.4 Besluit

In dit hoofdstuk werd een inleiding gegeven tot de wavelettheorie aan de hand van de geraadpleegde literatuur. We hebben gezien dat wavelets aanleiding kunnen geven tot ijle voorstellingen van functies (weinig grote coëfficiënten). Een waveletexpansie van een functie is een voorstelling van de functie als lineaire combinatie van verschoven en geschaalde versies van een waveletfunctie ψ . ψ is lokaal in zowel tijd als frequentie. Het manipuleren van de waveletcoëfficiënten heeft dus een lokaal effect.

Het idee van multiresolutie, het bekijken van een functie op verschillende schalen, blijkt aanleiding te geven tot orthonormale basissen van waveletfuncties. De op multiresolutie analyse gestoelde wavelettransformatie kan geïmplementeerd worden door een cascade van filters en heeft lineaire complexiteit. Het is dus een snel algoritme.

Er bestaat een grote variëteit aan mogelijke waveletfuncties die aangepast zijn aan verschillende toepassingen.

Men kan ook waveletontwikkelingen maken van functies van meerdere veranderlijken. Hoe dit aanleiding kan geven tot snelle numerieke algoritmen zullen we zien in hoofdstuk 4.

Hoofdstuk 4

Wavelets en operatoren

Het numeriek oplossen van (partiële) differentiaal- en integraalvergelijkingen vereist dat men het probleem discretiseert. Functies van één variabele worden dus vectoren met als componenten samples van de functie, gediscretiseerde operatoren worden matrices.

In het algemeen vereist het toepassen van een gediscretiseerde operator op een vector met N samples van een functie $O(N^2)$ vermenigvuldigingen. Voor meerdimensionale functies en operatoren (in tensorvorm geschreven) stijgt het aantal benodigde bewerkingen snel ($O(N^4)$ voor functies van twee veranderlijken, $O(N^6)$ voor functies van drie veranderlijken, ...). Dit is zeker zo voor integraaloperatoren, die dense matrices opleveren.

Dit hoofdstuk behandelt methoden [20] om dit hoge aantal bewerkingen terug te brengen (voor het ééndimensionale geval tot $O(N \log N)$ (standaard vorm) en zelfs $O(N)$ (non-standaard vorm)) door de operatoren en functies voor te stellen in een wavelet basis.

4.1 Tweedimensionale wavelets

Een gediscretiseerde operator die werkt op een vector met samples van een functie van één veranderlijke is een matrix. Deze matrix kunnen we beschouwen als de discretisatie van een functie van twee veranderlijken. Om hem in waveletbasis te schrijven hebben we dus een tweedimensionale waveletontbinding nodig.

Zoals de theorie van ééndimensionale Fouriertransformaties, filters, e.d. leidt tot de ééndimensionale wavelet basis, leiden tweedimensionale Fouriertransformaties, tweedimensionale filters e.d. tot tweedimensionale wavelet basissen. Deze basissen zijn in het algemeen niet separabel ('separable'), $\psi(x, y)$ kan in het algemeen niet geschreven worden als $\psi_1(x)\psi_2(y)$. Het is echter veel gemakkelijker om de tweedimensionale basis te construeren als tensorproduct van ééndimensionale basissen. Zulke tensor-product basissen zijn natuurlijk wel separabel [18, 16]. We zullen enkel separabele basissen gebruiken en beginnen met het continue, niet gediscretiseerde geval.

4.1.1 Standaard tensor-produkt basis

In het geval van de standaard tensor-produkt basis kiezen we als basis van $L^2(\mathbb{R}^2)$ de functies

$$\psi_{j_1 k_1}(x) \psi_{j_2 k_2}(y) \text{ met } j_1, j_2, k_1, k_2 \in \mathbb{Z}$$

Een functie $f(x, y)$ heeft als waveletexpansie dus

$$\begin{aligned} f(x, y) &= \sum_{j_1, k_1} \sum_{j_2, k_2} \langle f, \psi_{j_1 k_1}(x) \psi_{j_2 k_2}(y) \rangle \psi_{j_1 k_1}(x) \psi_{j_2 k_2}(y) \\ &= \sum_{j_1, k_1} \sum_{j_2, k_2} d_{j_1 j_2 k_1 k_2} \psi_{j_1 k_1}(x) \psi_{j_2 k_2}(y) \end{aligned}$$

met

$$d_{j_1 j_2 k_1 k_2} = \langle f, \psi_{j_1 k_1}(x) \psi_{j_2 k_2}(y) \rangle$$

De basisfuncties mengen informatie van verschillende schalen (want algemeen is $j_1 \neq j_2$). Schaling en translatie in beide richtingen zijn onafhankelijk. De wavelets zijn dus verschillend van nul (supported) in rechthoeken van allerlei groottes.

Toegepast op een lineaire operator leidt deze basis tot de zogenaamde *standaard vorm* (standard form) voorstelling van de operator.

4.1.2 Separabele multiresoluties

We kunnen de tweedimensionale basis ook opbouwen aan de hand van tensorprodukten van de ééndimensionale multiresolutie analyses (MRA's) [16, 18]. We definiëren

$$V_j^2 = V_j^x \otimes V_j^y$$

als MRA voor $L^2(\mathbb{R}^2)$ met $\{V_j^x\}_{j \in \mathbb{Z}}$ en $\{V_j^y\}_{j \in \mathbb{Z}}$ de ééndimensionale MRA's van $L^2(\mathbb{R})$. Alle eigenschappen van de ééndimensionale MRA blijven behouden [16]. We hebben dus

$$V_{j-1}^2 = V_j^2 \oplus W_j^2$$

Uit

$$\begin{aligned} V_{j-1}^2 &= V_{j-1}^x \otimes V_{j-1}^y \\ &= (V_j^x \oplus W_j^x) \otimes (V_j^y \oplus W_j^y) \\ &= (V_j^x \otimes V_j^y) \oplus (V_j^x \otimes W_j^y) \oplus (W_j^x \otimes V_j^y) \oplus (W_j^x \otimes W_j^y) \\ &= V_j^2 \oplus (V_j^x \otimes W_j^y) \oplus (W_j^x \otimes V_j^y) \oplus (W_j^x \otimes W_j^y) \end{aligned}$$

zien we dat het orthogonale complement W_j^2 van V_j^2 in V_{j-1}^2 gegeven wordt door

$$W_j^2 = (V_j^x \otimes W_j^y) \oplus (W_j^x \otimes V_j^y) \oplus (W_j^x \otimes W_j^y)$$

Daar $\{\varphi_{j,k}(x)\}_{k \in \mathbb{Z}}$, $\{\psi_{j,k}(x)\}_{k \in \mathbb{Z}}$, $\{\varphi_{j,k}(y)\}_{k \in \mathbb{Z}}$ en $\{\psi_{j,k}(y)\}_{k \in \mathbb{Z}}$ orthonormale basissen vormen van respectievelijk V_j^x , W_j^x , V_j^y , W_j^y hebben we

$$\{\varphi_{j,k_1}(x) \varphi_{j,k_2}(y)\}_{k_1, k_2 \in \mathbb{Z}}$$

als basis voor V_j^2 en

$$\{\varphi_{j,k_1}(x) \psi_{j,k_2}(y), \psi_{j,k_1}(x) \varphi_{j,k_2}(y), \psi_{j,k_1}(x) \psi_{j,k_2}(y)\}_{k_1, k_2 \in \mathbb{Z}}$$

als basis voor W_j^2 .

De schaling volgens de x - en de y -richting is steeds dezelfde, zodat de gebieden waar de wavelets niet nul zijn (waar ze 'supported' zijn) steeds vierkanten zijn.

Toegepast op een lineaire operator leidt deze basis tot de *non-standaard* of BCR (naar Beylkin, Coifman, Rohklin [20]) vorm van de operator.

4.2 Lineaire operatoren in wavelet basis

Beschouw een lineaire operator T die werkt op een functie $f(x)$. Om Tf zo goed mogelijk te berekenen beginnen we met de discretisatie van f . We kiezen een sampleperiode T_s en schrijven f_d als de gediscrètiseerde versie van f :

$$f_d = \sum_{k=-\infty}^{+\infty} f(kT) \delta(x - kT_s)$$

Om bij het samplen aliasing te vermijden moeten we vanzelfsprekend voldoen aan het samplingtheorem, de samplefrequentie $1/T_s$ moet dus dubbel zo groot zijn als de hoogste frequentiecomponent die voorkomt in f . Na sampling bekomen we een reeks samples $\{f_k\}$, $k = 1 \dots 2^J$, met $f_k = f(kT_s)$.

In de praktijk kunnen we maar een eindig aantal samplepunten beschouwen. We moeten de waarden die de index k kan aannemen dus beperken. Laten we k bv. variëren van 1 tot 2^J . We kiezen een macht van 2 als hoogste waarde omdat dit toelaat snelle algoritmen te gebruiken (cf. 3.3.2).

Vanaf nu werken we enkel nog met de reeks samples $\{f_k\}$, waarbij we de afstand tussen de samples 1 stellen. Dit betekent dat we de sampleperiode normaliseren: $T_s = 1$. We verlaten

ook de notatie f als aanduiding van de originele, continue functie. Met f verwijzen we in het vervolg naar volgende discrete functie:

$$f(n) = \sum_{k=1}^{2^J} f_k \delta_{nk} = f_n$$

met

$$\delta_{nk} = \begin{cases} 1 & n = k \\ 0 & n \neq k \end{cases}$$

Om de wavelet-expansie van f praktisch te berekenen (met het filter-bank algoritme), moeten we aan de samples f_k eerst een functie f' associëren met $f' \in V_0$ [16]. We moeten f dus projecteren op V_0 : $f' = P_0 f$. 0 is hierbij de fijnste schaal, een keuze die zoals we zullen zien niet arbitrair is. Om de projectie f' te kennen moeten we de inproducten $\langle f, \varphi_{0k} \rangle$ berekenen voor alle $k = 1 \dots 2^J$, zodat we kunnen schrijven (zie (3.26))

$$\begin{aligned} f' = P_0 f &= \sum_{k=1}^{2^J} \langle f, \varphi_{0k} \rangle \varphi_{0k} \\ &= \sum_{k=1}^{2^J} \tilde{s}_{0k} \varphi_{0k} \end{aligned}$$

Hierin zijn φ_{0k} de schaling functies op schaal 0 ,

$$\varphi_{0k}(x) = \varphi(x - k),$$

met φ de schaling-functie van de beschouwde waveletbasis en $\tilde{s}_{0k} = \langle f, \varphi_{0k} \rangle$.

Dit is een ingewikkeld proces. Het zou veel eenvoudiger zijn om gewoonweg te stellen dat $f \in V_0$ en de functiesamples f_k te beschouwen als schaling-coëfficiënten op schaal 0 (wat niet noodzakelijk correct is!):

$$P_0 f = f = \sum_{k=1}^{2^J} f_k \varphi_{0k}$$

Het voordeel hiervan zou zijn dat we de functiesamples zelf door het filterbank algoritme kunnen laten bewerken, zonder voorafgaande berekeningen. De samples f_k zonder meer als schaling coëfficiënten beschouwen noemt men de *wavelet crime*. Mallat [16] toont echter aan dat dit in de praktijk meestal geen probleem vormt, tenminste als de samplefrequentie hoog genoeg gekozen wordt. Immers:

$$\tilde{s}_{0k} = \langle f, \varphi_{0k} \rangle = \int_{-\infty}^{+\infty} f(x) \varphi(x - k) dx$$

Hierbij werd verondersteld dat φ reëel is (we zullen in het vervolg steeds met reële wavelet- en schaling functies werken). Daar $\int_{-\infty}^{+\infty} \varphi(x) dx = 1$ (zie (3.35)) en de drager (support)

van $\varphi(x)$ zich voornamelijk rond $x = 0$ bevindt, zijn de getallen \tilde{s}_{0k} gewogen gemiddelden van f in de buurt van $x = k$, voor $k = 1 \dots 2^J$. Als f binnen één sample-interval niet te veel varieert, zal dus gelden dat

$$f_k \approx \tilde{s}_{0k} \quad (4.1)$$

We zien nu ook waarom schaal 0 werd gekozen als fijnste schaal. Op deze schaal worden de schalingfuncties getransleerd in stapjes van $2^0 = 1$, gelijk aan het samplinginterval. Zo zullen de functiesamples samenvallen met de positie van de grootste ‘massa’ van de schaalfuncties, zodat de gewogen gemiddeldes \tilde{s}_{0k} en de functiesamples f_k niet sterk zullen verschillen. Het is ook mogelijk om de sampleperiode te normaliseren op een macht van 2, bv. 2^j , en dan schaal j als fijnste schaal te kiezen.

(4.1) sust ons geweten genoeg om zonder bezwaren in de georganiseerde misdaad te stappen. We zullen dan ook steeds werken met 0 als fijnste schaal en de functiesamples beschouwen als schaling coëfficiënten op diezelfde schaal.

4.2.1 Standaard vorm operatoren

Standaard vorm operator toegepast op een functie

We beginnen met een discrete wavelet transformatie van de gediscretiseerde functie tot op schaal N (de grofste schaal of ‘coarsest scale’). De waarde van $N \geq 1$ is vrij te kiezen. De enige beperking is het aantal samples: als er oorspronkelijk 2^J samples zijn moet $N \leq J$, want op schaal J is er nog maar één schaal-coëfficiënt en kunnen we dus niet verder ontbinden. Voor 2^J samples krijgen we (zie (3.31)):

$$P_0 f = \sum_{j=1}^N Q_j f + P_N f \quad (4.2)$$

$$= \sum_{j=1}^N \sum_{k=1}^{2^{J-j}} \langle f, \psi_{j,k} \rangle \psi_{j,k} + \sum_{k=1}^{2^{J-N}} \langle f, \varphi_{N,k} \rangle \varphi_{N,k}$$

$$= \sum_{j=1}^N \sum_{k=1}^{2^{J-j}} d_{jk} \psi_{j,k} + \sum_{k=1}^{2^{J-N}} s_{Nk} \varphi_{N,k} \quad (4.3)$$

met $\langle f, \psi_{j,k} \rangle = d_{jk}$ en $\langle f, \varphi_{N,k} \rangle = s_{Nk}$. Merk op dat we in (4.2) en (4.3) verschillende vormen gebruiken. De vorm gebruikt in (4.3) is vooral handig bij de praktische implementatie, want ze bevat rechtstreeks de wavelet- en schalingcoëfficiënten d_{jk} en s_{Nk} . De vorm uit (4.2) is meestal nuttiger in theoretische berekeningen omdat hij minder ingewikkeld is. Beide vormen zijn natuurlijk equivalent.

Toepassen van de operator T op $P_0 f$ levert

$$\begin{aligned} TP_0 f &= \sum_{j=1}^N TQ_j f + TP_N f \\ &= \sum_{j=1}^N \sum_{k=1}^{2^{J-j}} d_{jk} T\psi_{j,k} + \sum_{k=1}^{2^{J-N}} s_{Nk} T\varphi_{N,k} \end{aligned} \quad (4.4)$$

De functies $\{T\psi_{j,k}\}_{j=1\dots N, k=1\dots 2^{J-j}}$ en $\{T\varphi_{N,k}\}_{k=1\dots 2^{J-N}}$ vormen in het algemeen geen orthonormale basissen meer voor respectievelijk W_j , $j = 1 \dots N$ en V_N . $TP_0 f$ bevindt zich niet noodzakelijk meer in V_0 . Als we de sommen van (4.4) uitwerken, bekommen we een resultaat in de natuurlijke basis. Om in de wavelet ruimte te blijven moeten we terug projecteren op de waveletbasis:

$$\begin{aligned} P_0 TP_0 f &= \sum_{j'=1}^N Q_{j'}(TP_0 f) + P_N(TP_0 f) \\ &= \sum_{j'=1}^N \sum_{j=1}^N Q_{j'} TQ_j f + \sum_{j'=1}^N Q_{j'} TP_N f \\ &\quad + \sum_{j=1}^N P_N TQ_j f + P_N TP_N f \\ &= \sum_{j=1}^N \sum_{j'=1}^N \sum_{k=1}^{2^{J-j}} \sum_{k'=1}^{2^{J-j'}} d_{jk} \langle T\psi_{j,k}, \psi_{j',k'} \rangle \psi_{j',k'} \\ &\quad + \sum_{j'=1}^N \sum_{k'=1}^{2^{J-j'}} \sum_{k=1}^{2^{J-N}} s_{Nk} \langle T\varphi_{N,k}, \psi_{j',k'} \rangle \psi_{j',k'} \\ &\quad + \sum_{k'=1}^{2^{J-N}} \sum_{j=1}^N \sum_{k=1}^{2^{J-j}} d_{jk} \langle T\psi_{j,k}, \varphi_{N,k'} \rangle \varphi_{N,k'} \\ &\quad + \sum_{k=1, k'=1}^{2^{J-N}} s_{Nk} \langle T\varphi_{N,k}, \varphi_{N,k'} \rangle \varphi_{N,k'} \end{aligned} \quad (4.5)$$

Als we stellen dat

$$\hat{d}_{j',k'} = \sum_{j=1}^N \sum_{k=1}^{2^{J-j}} \langle T\psi_{j,k}, \psi_{j',k'} \rangle d_{jk} + \sum_{k=1}^{2^{J-N}} \langle T\varphi_{N,k}, \psi_{j',k'} \rangle s_{Nk} \quad (4.6)$$

$$\hat{s}_{Nk'} = \sum_{j=1}^N \sum_{k=1}^{2^{J-j}} \langle T\psi_{j,k}, \varphi_{N,k'} \rangle d_{jk} + \sum_{k=1}^{2^{J-N}} s_{Nk} \langle T\varphi_{N,k}, \varphi_{N,k'} \rangle \quad (4.7)$$

zien we dat (4.5) inderdaad als gewone expansie in waveletbasis kan geschreven worden:

$$P_0 T P_0 f = \sum_{j'=1}^N \sum_{k'=1}^{2^{J-j'}} \widehat{d}_{j',k'} \psi_{j',k'} + \sum_{k'=1}^{2^{J-N}} \widehat{s}_{N,k'} \varphi_{N,k'}$$

(vgl. met (4.3)).

We kunnen de coëfficiënten uit (4.5) nu zo organiseren dat de berekening van de getallen $\widehat{d}_{j',k'}$ en $\widehat{s}_{N,k'}$ in één enkele matrix-vector vermenigvuldiging kan gebeuren. Om dat te zien zullen we de verschillende termen van (4.5) apart bekijken:

- We beginnen met de eerste term van (4.5) en splitsen hem in drie bijdragen waarbij respectievelijk $j = j'$, $j > j'$, $j < j'$:

$$\begin{aligned} & \sum_{j=1, j'=1}^N \sum_{k=1, k'=1}^{2^{J-j}} d_{jk} \langle T \psi_{j,k}, \psi_{j',k'} \rangle \psi_{j',k'} = \\ & \sum_{j=1}^N \sum_{k=1, k'=1}^{2^{J-j}} d_{jk} \langle T \psi_{j,k}, \psi_{j,k'} \rangle \psi_{j,k'} \\ & + \sum_{j=2}^N \sum_{j'=1}^{j-1} \sum_{k=1, k'=1}^{2^{J-j}} d_{jk} \langle T \psi_{j,k}, \psi_{j',k'} \rangle \psi_{j',k'} \\ & + \sum_{j'=2}^N \sum_{j=1}^{j'-1} \sum_{k=1, k'=1}^{2^{J-j}} d_{jk} \langle T \psi_{j,k}, \psi_{j',k'} \rangle \psi_{j',k'} \end{aligned}$$

Voor elke combinatie van j en j' vormen de getallen $\langle T \psi_{j,k}, \psi_{j',k'} \rangle$ een matrix. We gebruiken volgende naamconventie:

$$\begin{aligned} A_j &= \{ \alpha_{k'k}^j \} = \{ \langle T \psi_{j,k}, \psi_{j,k'} \rangle \} \quad 2^{J-j} \times 2^{J-j} \text{ matrix} \\ B_{j'}^j &= \{ \beta_{k'k}^{j,j'} \} = \{ \langle T \psi_{j,k}, \psi_{j',k'} \rangle \} \quad 2^{J-j'} \times 2^{J-j} \text{ matrix} \quad j > j' \\ \Gamma_j^{j'} &= \{ \gamma_{k'k}^{j,j'} \} = \{ \langle T \psi_{j,k}, \psi_{j',k'} \rangle \} \quad 2^{J-j'} \times 2^{J-j} \text{ matrix} \quad j < j' \end{aligned} \quad (4.8)$$

De grootste schaalindex staat bij de B - en Γ -matrices dus steeds bovenaan.

De sommen over k en k' in (4.5) zijn matrixprodukten van de verschillende matrices met kolomvectoren gevormd door de detailcoëfficiënten (op de juiste schaal) van de waveletontbinding van de functie.

De $B_{j'}^j$ en $\Gamma_j^{j'}$ -matrices geven aan in welke mate de operator de schalen j en j' 'mengt'. $B_{j'}^j$ bepaalt hoe sterk de detailinformatie op de groffere schalen (j) van de originele functie invloed heeft op de fijnere schalen ($j' < j$) van het resultaat (dus na toepassen van de operator). $\Gamma_j^{j'}$ bepaalt het omgekeerde effect, nl. hoe sterk de fijnere schalen (j') invloed hebben op de groffere schalen ($j < j'$).

- Ook de tweede term van (4.5)

$$\sum_{j'=1}^N \sum_{k'=1}^{2^{J-j}} \sum_{k=1}^{2^{J-N}} s_{Nk} \langle T\varphi_{N,k}, \psi_{j',k'} \rangle \psi_{j',k'}$$

bevat matrix-vectorprodukten. Geef deze matrices volgende namen:

$$B_{j'}^{N+1} = \left\{ \beta_{k'k}^{j',N+1} \right\} = \left\{ \langle T\varphi_{N,k}, \psi_{j',k'} \rangle \right\} \quad 2^{J-j'} \times 2^{J-N} \text{ matrix}$$

De $B_{j'}^{N+1}$ bepalen hoe sterk de minst gedetailleerde versie van de functie die we beschouwen (nl. het 'gemiddelde' van de functie, de functie bekeken op de grofste schaal, $P_N f$) de fijnere schalen (j') beïnvloedt bij toepassen van de operator.

- Derde term:

$$\sum_{k'=1}^{2^{J-N}} \sum_{j=1}^N \sum_{k=1}^{2^{J-j}} d_{jk} \langle T\psi_{j,k}, \varphi_{N,k'} \rangle \varphi_{N,k'}$$

Hier gebruiken we volgende conventie:

$$\Gamma_j^{N+1} = \left\{ \gamma_{k'k}^{j,N+1} \right\} = \left\{ \langle T\psi_{j,k}, \varphi_{N,k'} \rangle \right\} \quad 2^{J-N} \times 2^{J-j} \text{ matrix}$$

De Γ_j^{N+1} bepalen hoeveel de detailinformatie op de verschillende schalen (j) het 'gemiddelde' van het resultaat (het resultaat bekeken op de grofste schaal, $P_N T P_0 f$) beïnvloedt bij toepassen van de operator.

- De vierde term

$$\sum_{k=1, k'=1}^{2^{J-N}} s_{Nk} \langle T\varphi_{N,k}, \varphi_{N,k'} \rangle \varphi_{N,k'}$$

bevat de benadering van de operator T op de grofste schaal:

$$P_N T P_N = T_N = \left\{ t_{k'k}^N \right\} = \left\{ \langle T\varphi_{N,k}, \varphi_{N,k'} \rangle \right\} \quad 2^{J-N} \times 2^{J-N} \text{ matrix}$$

Als bv. $N = 3$ kunnen al deze matrices als volgt gegroepeerd worden:

$$\begin{bmatrix} A_1 & B_1^2 & B_1^3 & B_1^4 \\ \Gamma_1^2 & A_2 & B_2^3 & B_2^4 \\ \Gamma_1^3 & \Gamma_2^3 & A_3 & B_3^4 \\ \Gamma_1^4 & \Gamma_2^4 & \Gamma_3^4 & T_3 \end{bmatrix} \begin{bmatrix} d_1 \\ d_2 \\ d_3 \\ s_3 \end{bmatrix} = \begin{bmatrix} \hat{d}_1 \\ \hat{d}_2 \\ \hat{d}_3 \\ \hat{s}_3 \end{bmatrix} \quad (4.9)$$

Dit noemt men de *standaard vorm* van een operator [21, 20, 18]. We zien dat alle schalen hierin gemengd worden (alle schalen van de input hebben invloed op alle schalen van de output).

Dat het mengen van schalen in de matrix nadelig is [22, 16] (de matrix minder ijl (sparse) maakt) kunnen we als volgt begrijpen:

Beschouw als voorbeeld de elementen van de A -, B - en Γ -matrices:

$$\langle T\psi_{j,k}, \psi_{j',k'} \rangle = \int_{-\infty}^{+\infty} T\psi_{j,k}(x) \overline{\psi_{j',k'}(x)} dx \quad (4.10)$$

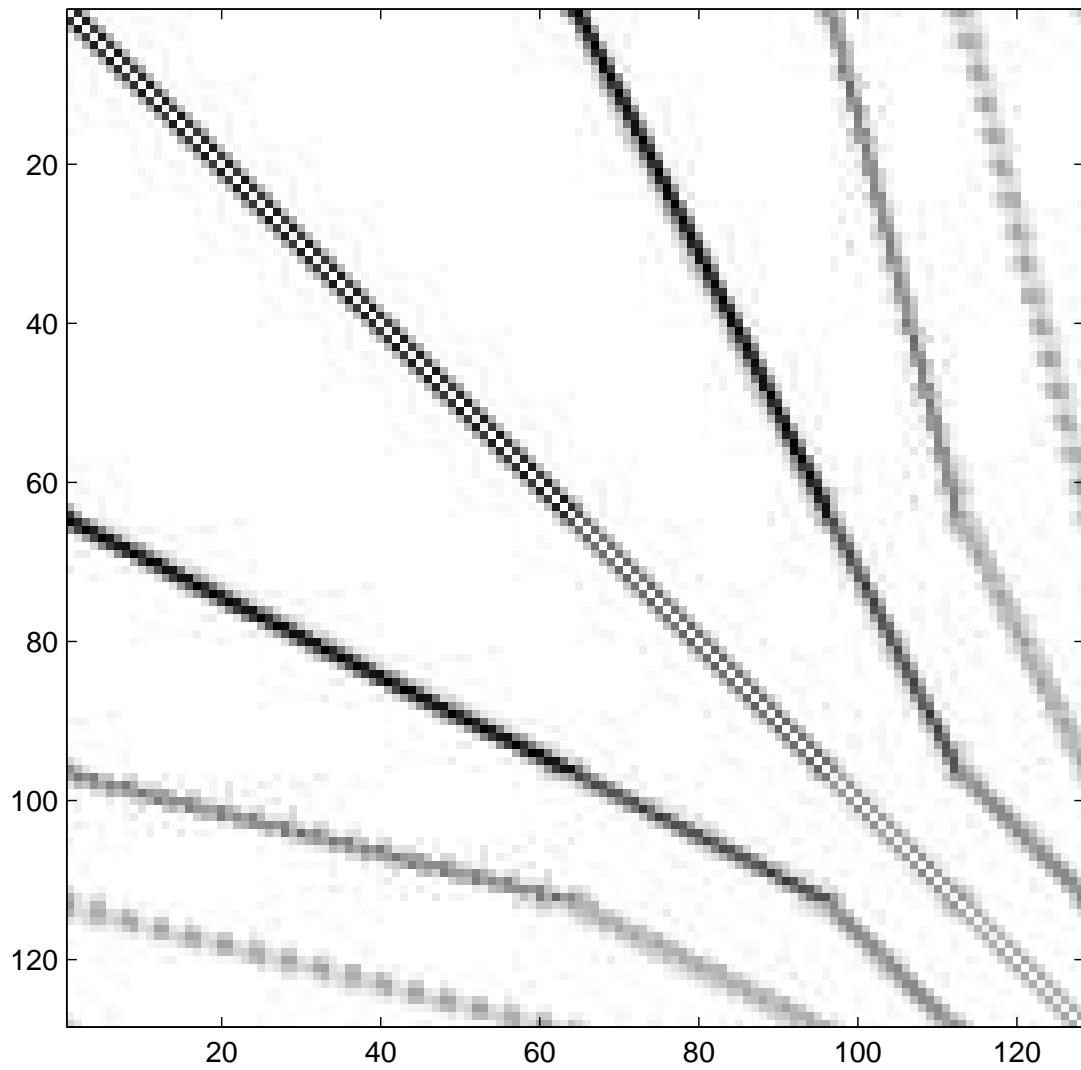
Als $j = j'$ zullen de matrixelementen klein zijn als het verschil tussen k en k' groot genoeg is: voor wavelets met compacte drager (compact support) zijn veel elementen nul en bij wavelets zonder compacte drager zullen veel elementen heel klein zijn, tenminste als de wavelet voldoende snel verval (voldoende snelle 'decay' heeft). Als we echter schalen mengen ($j \neq j'$) krijgen we grote matrixelementen voor $2^j k \approx 2^{j'} k'$. Dit is gemakkelijk te zien aan de definitie van $\psi_{j,k}$:

$$\psi_{j,k}(x) = \frac{1}{\sqrt{2^j}} \psi\left(\frac{x - 2^j k}{2^j}\right)$$

Als $2^j k \approx 2^{j'} k'$ zullen $\psi_{j,k}$ en $\psi_{j',k'}$ ongeveer op dezelfde plaats in de ruimte gelokaliseerd zijn. Als toepassen van T op $\psi_{j,k}$ de ruimtelokalisatie ervan niet ingrijpend verandert (zoals vaak het geval is) zullen de inproducten (4.10) meestal relatief groot zijn. We zien dus dat zelfs als het verval (decay) van de matrixcoëfficiënten binnen dezelfde schaal snel is, het verval tussen verschillende schalen toch traag kan zijn. Dit beperkt de mate van ijl-zijn die met de standaard vorm voorstelling kan bereikt worden.

Een voorbeeld van een operator in standaard vorm (operator $\frac{d}{dx}$) is te zien in figuur 4.1. De interactie van de verschillende schalen is heel duidelijk zichtbaar aan de niet-diagonaalbanden, de zogenaamde vingers. We zien ook dat schalen j en j' minder sterk interageren naarmate $|j - j'|$ groter wordt.

De standaard vorm wavelet representatie van de $\frac{d}{dx}$ -operator is minder ijl dan zijn voorstelling in natuurlijke basis. De winst in rekentijd moet hier komen van de voorstelling van de functie in wavelet basis (meestal na het gelijk aan nul stellen van waveletcoëfficiënten kleiner dan een bepaalde drempelwaarde). Bij sterk oscillerende functies (zoals bv. de velden in een VCSEL) is de wavelet basis echter meestal niet optimaal. De gewone Fourier-basis kan in dit geval beter zijn (= meer nul-coëfficiënten opleveren) [18]. Bij operatoren die een dense discretisatiematrix hebben (integraaloperatoren bijvoorbeeld) zal de standaard vorm voorstelling, eventueel na thresholding, ijl zijn. Bij deze operatoren is de winst door het gebruik van wavelets veel groter.



Figuur 4.1: Standaard vorm van de operator $\frac{d}{dx}$ berekend met de Daubechies wavelet met 2 nulmomenten. Donkere kleuren betekenen grote matricelementen, lichte kleuren kleine. De vinger-structuur, typisch voor de standaard vorm, wordt veroorzaakt door de interactie tussen de verschillende schalen.

Verband met tweedimensionale standaard tensor-produkt basis

We zullen nu het verband tussen de standaard vorm matrix (4.9) en de tweedimensionale standaard tensor-produkt basis, geïntroduceerd in sectie 4.1.1, verduidelijken. De matrix van de operator in natuurlijke basis beschouwen als een tweedimensionale discrete functie en deze in de standaard tensor-produkt wavelet basis schrijven levert nl. ook de standaard vorm operator-matrix op.

Om dit te zien bekijken deze basis wat meer in detail. In sectie 4.1.1 werd ze beschouwd voor continue functies van twee veranderlijken. Daar we nu een gediscretiseerde operator matrix willen behandelen, moeten we de basis eerst bekijken voor het discrete geval. Bij discrete waveletontbindingen is er altijd een grofste schaal N , met $1 \leq N \leq J$ als we een $2^J \times 2^J$ matrix beschouwen. Strikt genomen mag N ook nul zijn, maar dan hebben we geen wavelet ontbinding en blijft de matrix in natuurlijke basis staan. Het bestaan van een grofste schaal betekent dat enkel wavelet functies geen orthonormale basis kunnen vormen voor de ruimte V_0^2 (we begaan de wavelet crime en veronderstellen dat de matrixelementen schalingcoëfficiënten zijn, zodat de matrix, beschouwd als discrete functie, behoort tot V_0^2). De schaalfuncties gaan meespelen op de grofste schaal N . Daar we nu in twee dimensies werken, hebben we een schaalfunctie afhankelijk van x ($\varphi(x)$) en van y ($\varphi(y)$). Een orthonormale basis voor V_0 is dus

$$\begin{aligned} & \{\psi_{j_1 k_1}(x) \psi_{j_2 k_2}(y)\} \cup \{\psi_{j_1 k_1}(x) \varphi_{N l_2}(y)\} \\ & \cup \{\varphi_{N l_1}(x) \psi_{j_2 k_2}(y)\} \cup \{\varphi_{N l_1}(x) \varphi_{N l_2}(y)\} \end{aligned}$$

met

$$\begin{aligned} j_1 &= 1 \dots N \\ j_2 &= 1 \dots N \\ k_1 &= 1 \dots 2^{J-j_1} \\ k_2 &= 1 \dots 2^{J-j_2} \\ l_1 &= 1 \dots 2^{J-N} \\ l_2 &= 1 \dots 2^{J-N} \end{aligned}$$

De discretisatiematrix van T beschouwd als tweedimensionale functie $t(x, y)$, met $x, y = 1 \dots 2^J$, kunnen we dus op volgende manier ontbinden in standaard tensor-produkt wavelet

basis:

$$\begin{aligned}
t(x, y) &= \sum_{j_1=1}^N \sum_{j_2=1}^N \sum_{k_1=1}^{2^{J-j_1}} \sum_{k_2=1}^{2^{J-j_2}} \langle t(x, y), \psi_{j_1 k_1}(x) \psi_{j_2 k_2}(y) \rangle \psi_{j_1 k_1}(x) \psi_{j_2 k_2}(y) \\
&+ \sum_{j_1=1}^N \sum_{k_1=1}^{2^{J-j_1}} \sum_{b_2=1}^{2^{J-N}} \langle t(x, y), \psi_{j_1 k_1}(x) \varphi_{N b_2}(y) \rangle \psi_{j_1 k_1}(x) \varphi_{N b_2}(y) \\
&+ \sum_{j_2=1}^N \sum_{k_2=1}^{2^{J-j_2}} \sum_{h_1=1}^{2^{J-N}} \langle t(x, y), \varphi_{N h_1}(x) \psi_{j_2 k_2}(y) \rangle \varphi_{N h_1}(x) \psi_{j_2 k_2}(y) \\
&+ \sum_{l_1=1}^{2^{J-N}} \sum_{b_2=1}^{2^{J-N}} \langle t(x, y), \varphi_{N l_1}(x) \varphi_{N b_2}(y) \rangle \varphi_{N l_1}(x) \varphi_{N b_2}(y)
\end{aligned}$$

Voor de verschillende matrices die hierin voorkomen geldt nu:

$$\begin{aligned}
\{ \langle t(x, y), \psi_{j_1 k_1}(x) \psi_{j_2 k_2}(y) \rangle \} &= A_{j_1} \quad j_1 = j_2 \\
\{ \langle t(x, y), \psi_{j_1 k_1}(x) \psi_{j_2 k_2}(y) \rangle \} &= B_{j_2}^{j_1} \quad j_1 \geq j_2 \\
\{ \langle t(x, y), \psi_{j_1 k_1}(x) \psi_{j_2 k_2}(y) \rangle \} &= \Gamma_{j_1}^{j_2} \quad j_1 \leq j_2 \\
\{ \langle t(x, y), \psi_{j_1 k_1}(x) \varphi_{N l_2}(y) \rangle \} &= \Gamma_{j_1}^{N+1} \\
\{ \langle t(x, y), \varphi_{N l_1}(x) \psi_{j_2 k_2}(y) \rangle \} &= B_{j_2}^{N+1} \\
\{ \langle t(x, y), \varphi_{N l_1}(x) \varphi_{N l_2}(y) \rangle \} &= T_N
\end{aligned}$$

We zullen dit enkel uitwerken voor het eerste geval. De andere zijn volledig analoog.

We willen aantonen dat $\{ \langle t(x, y), \psi_{j k_1}(x) \psi_{j k_2}(y) \rangle \} = A_j = \{ \langle T \psi_{j, k_1}, \psi_{j, k_2} \rangle \}$ (zie (4.8)), of dat

$$\langle t(x, y), \psi_{j k_1}(x) \psi_{j k_2}(y) \rangle = \langle T \psi_{j, k_1}, \psi_{j, k_2} \rangle$$

We beginnen met het rechterlid, meer bepaald met $T \psi_{j, k_1}$. Daar we een verband willen aantonen met de gediscretiseerde operatormatrix, moeten we T en ψ_{j, k_1} hier ook als discreet beschouwen. Voor een bepaalde y , $y = 1 \dots 2^J$ wordt $T \psi_{j, k_1}(y)$ dus

$$T \psi_{j, k_1}(y) = \sum_{x=1}^{2^J} t(x, y) \psi_{j, k_1}(x)$$

Het inproduct $\langle T \psi_{j, k_1}, \psi_{j, k_2} \rangle$ is dus gelijk aan (we werken met reële wavelet- en schaling functies)

$$\langle T \psi_{j, k_1}, \psi_{j, k_2} \rangle = \sum_{y=-\infty}^{+\infty} \sum_{x=1}^{2^J} t(x, y) \psi_{j, k_1}(x) \psi_{j, k_2}(y)$$

Daar $t(x, y) = 0$ voor $x, y < 1$ en $x, y > 2^J$ kunnen we de som over x uitbreiden zodat

$$\begin{aligned}\langle T\psi_{j,k_1}, \psi_{j,k_2} \rangle &= \sum_{y=-\infty}^{+\infty} \sum_{x=-\infty}^{+\infty} t(x, y) \psi_{j,k_1}(x) \psi_{j,k_2}(y) \\ &= \langle t(x, y), \psi_{j,k_1}(x) \psi_{j,k_2}(y) \rangle\end{aligned}$$

Dit is wat we wilden aantonen. De andere gevallen zijn volledig analoog.

Daar de expansie van een matrix in standaard tensor-produkt basis berekend wordt door eerst alle rijen te transformeren, waarna we de kolommen transformeren (of omgekeerd), levert dit een gemakkelijke methode op voor het berekenen van de standaard vorm.

4.2.2 Non-Standaard vorm operatoren

De non-standaard of BCR vorm [20] is een wavelet-voorstelling van operatoren die de verschillende schalen zo maximaal mogelijk ontkoppelt. Bij deze voorstelling beschouwt men de gediscretiseerde operator als een discrete functie van twee variabelen. Die functie wordt geschreven in een wavelet basis geconstrueerd aan de hand van een tweedimensionale MRA, waarbij rekening gehouden wordt met wat er gebeurt met de vector waarop de matrix toegepast wordt [22].

Ook bij de non-standaard vorm beginnen we met een discretisatie. Dit betekent dat we de operator T beperken tot een bepaalde resolutie (= fijnste schaal, schaal 0) V_0 . We werken dus met $T_0 = P_0 T P_0$ als benadering van T (zodanig dat $T_0 f = P_0 T P_0 f \in V_0$). Beschouw V_N als de ruimte die de grofste schaal voorstelt en stel dat we de functie discretiseren in 2^J samples.

T_0 wordt geschreven in een zogenaamde *telescopische reeks* [20]:

$$\begin{aligned}T_0 &= P_0 T P_0 \\ &= P_0 T P_0 - P_1 T P_1 + P_1 T P_1 - P_2 T P_2 + P_2 T P_2 - \dots - P_N T P_N + P_N T P_N \\ &= \sum_{j=1}^N (P_{j-1} T P_{j-1} - P_j T P_j) + P_N T P_N \\ &= \sum_{j=1}^N ((P_{j-1} - P_j) T (P_{j-1} - P_j) + (P_{j-1} - P_j) T P_j + P_j T (P_{j-1} - P_j)) \\ &\quad + P_N T P_N \\ &\Downarrow P_{j-1} - P_j = Q_j \\ T_0 &= \sum_{j=1}^N (Q_j T Q_j + Q_j T P_j + P_j T Q_j) + P_N T P_N \\ &= \sum_{j=1}^N (A_j + B_j + \Gamma_j) + T_N\end{aligned}\tag{4.11}$$

In de laatste stap werden volgende definities gebruikt:

$$A_j = Q_j T Q_j \quad (W_j \rightarrow W_j) \quad (4.12)$$

$$B_j = Q_j T P_j \quad (V_j \rightarrow W_j) \quad (4.13)$$

$$\Gamma_j = P_j T Q_j \quad (W_j \rightarrow V_j) \quad (4.14)$$

$$T_j = P_j T P_j \quad (V_j \rightarrow V_j) \quad (4.15)$$

De expansie voor T_0 kan ook eenvoudiger bekomen worden door $P_j = P_{j+1} + Q_{j+1}$ toe te passen op de definitie van T_0 :

$$\begin{aligned} T_0 &= P_0 T P_0 \\ &= (P_1 + Q_1) T (P_1 + Q_1) \\ &= P_1 T P_1 + P_1 T Q_1 + Q_1 T P_1 + Q_1 T Q_1 \\ &= T_1 + \Gamma_1 + B_1 + A_1 \end{aligned}$$

Toepassen van dezelfde formule op T_2 , dan op T_3 , T_4 enz. levert ook (4.11) op. (4.11) wordt de *non-standaard* vorm genoemd.

Om te zien hoe we de non-standaard vorm in de praktijk moeten gebruiken, laten we hem werken op een functie:

$$T_0 f = \sum_{j=1}^N (A_j f + B_j f + \Gamma_j f) + T_N f \quad (4.16)$$

Alle termen uitwerken levert het resultaat in de natuurlijke basis. In tegenstelling tot de aanpak bij de standaard vorm zullen we dit keer niet beginnen met dit opnieuw in wavelet basis te schrijven. We zullen eerst kijken hoe we (4.16) kunnen uitwerken d.m.v. een matrix-vector vermenigvuldiging. Hiervoor beschouwen we elke tem van (4.16) apart.

- Bij term $A_j f = Q_j T Q_j f$ kijken we eerst naar $T Q_j f$, het inwerken van operator T op de projectie van f op W_j :

$$T Q_j f = \sum_{k=1}^{2^{J-j}} \langle f, \psi_{j,k} \rangle T \psi_{j,k} = \sum_{k=1}^{2^{J-j}} d_{jk} T \psi_{j,k}$$

Q_j opnieuw gebruiken projecteert $T Q_j f$ (dat zich door inwerking van T misschien niet meer in W_j bevindt) terug op W_j :

$$Q_j T Q_j f = \sum_{k'=1}^{2^{J-j}} \sum_{k=1}^{2^{J-j}} d_{jk} \langle T \psi_{j,k}, \psi_{j,k'} \rangle \psi_{j,k'}$$

Noemen we nu

$$A'_j = \{a'_{k'k}\} = \{\langle T \psi_{j,k}, \psi_{j,k'} \rangle\} \quad 2^{J-j} \times 2^{J-j} \text{ matrix} \quad (4.17)$$

en

$$d_{jk'}^a = \sum_{k=1}^{2^{J-j}} d_{jk} \langle T\psi_{j,k}, \psi_{j,k'} \rangle = \sum_{k=1}^{2^{J-j}} a_{k'k}^j d_{jk}$$

dan hebben we

$$Q_j T Q_j f = \sum_{k'=1}^{2^{J-j}} d_{jk'}^a \psi_{j,k'}$$

waarbij de getallen $d_{jk'}^a$ bekomen worden door A_j te vermenigvuldigen met de vector van coëfficiënten d_{jk} . Dit is zo voor elke schaal, er worden geen schalen gemengd.

Matrix A_j geeft aan hoe sterk de detailinformatie op een bepaalde schaal (j) de informatie op diezelfde schaal beïnvloedt bij toepassen van de operator.

- Bij de tweede term $B_j f = Q_j T P_j f$ gaan we volledig analoog te werk:

$$Q_j T P_j f = \sum_{k'=1}^{2^{J-j}} \sum_{k=1}^{2^{J-j}} s_{jk} \langle T\varphi_{j,k}, \psi_{j,k'} \rangle \psi_{j,k'}$$

Hier noemen we

$$B_j' = \{b_{k'k}^j\} = \{\langle T\varphi_{j,k}, \psi_{j,k'} \rangle\} \quad 2^{J-j} \times 2^{J-j} \text{ matrix} \quad (4.18)$$

en

$$d_{jk'}^b = \sum_{k=1}^{2^{J-j}} s_{jk} \langle T\varphi_{j,k}, \psi_{j,k'} \rangle = \sum_{k=1}^{2^{J-j}} b_{k'k}^j s_{jk}$$

zodat

$$Q_j T P_j f = \sum_{k'=1}^{2^{J-j}} d_{jk'}^b \psi_{j,k'}$$

We zien dat de $d_{jk'}^b$ afhangen van de schaling-coëfficiënten op schaal j s_{jk} . Daar bij de normale wavelet ontbinding enkel de schaling coëfficiënten op de grofste schaal (N) bijgehouden worden, zullen we een aangepaste ontbinding van f moeten gebruiken voor non-standaard vorm operatoren. Bij de ontbinding moeten op elke schaal zowel wavelet- als schaling-coëfficiënten bijgehouden worden. Deze ontbinding zal dus redundant zijn.

Matrix B_j' geeft aan hoe sterk de schaling-informatie op schaal j en de informatie op alle groffere schalen ($j' > j$) de detailinformatie op schaal j van het resultaat beïnvloedt [21]. Dit komt omdat de s_{jk} alle informatie van de groffere schalen ($j' > j$) bevatten.

Als we de invloed van $A_j f$ en $B_j f$ samen bekijken

$$\begin{aligned}
 A_j f + B_j f &= \sum_{k'=1}^{2^{J-j}} d_{jk'}^a \psi_{j,k'} + \sum_{k'=1}^{2^{J-j}} d_{jk'}^b \psi_{j,k'} \\
 &= \sum_{k'=1}^{2^{J-j}} (d_{jk'}^a + d_{jk'}^b) \psi_{j,k'} \\
 &= \sum_{k'=1}^{2^{J-j}} \hat{d}_{jk'} \psi_{j,k'} \tag{4.19}
 \end{aligned}$$

zien we dat de detailcoëfficiënten op schaal j van het resultaat gegeven worden door

$$\hat{d}_{jk'} = d_{jk'}^a + d_{jk'}^b$$

- Derde term van (4.16) $\Gamma_j f = P_j T Q_j f$:

$$P_j T Q_j f = \sum_{k'=1}^{2^{J-j}} \sum_{k=1}^{2^{J-j}} d_{jk} \langle T \psi_{j,k}, \varphi_{j,k'} \rangle \varphi_{j,k'}$$

Noem

$$\Gamma_j' = \{ \gamma_{k'k}^j \} = \{ \langle T \psi_{j,k}, \varphi_{j,k'} \rangle \} \quad 2^{J-j} \times 2^{J-j} \text{ matrix} \tag{4.20}$$

en

$$\hat{s}_{jk'} = \sum_{k=1}^{2^{J-j}} d_{jk} \langle T \psi_{j,k}, \varphi_{j,k'} \rangle = \sum_{k=1}^{2^{J-j}} \gamma_{k'k}^j d_{jk}$$

dan hebben we

$$P_j T Q_j f = \sum_{k'=1}^{2^{J-j}} \hat{s}_{jk'} \varphi_{j,k'} \tag{4.21}$$

We zien dat de term $\Gamma_j f$ dus resulteert in schaling-coëfficiënten op schaal j . Als we de non-standaard vorm van een operator louter als lineaire algebra tool gebruiken (in de hoop matrix-vectorprodukten te versnellen) stelt dit in principe geen probleem: we tellen dan gewoon alle bijdragen uit (4.16) op en verkrijgen het resultaat in natuurlijke basis. Praktisch is er toch een probleem, want van de functies $\psi_{j,k}$, en $\varphi_{j,k'}$ is meestal geen analytische uitdrukking beschikbaar. De waarden van deze functies in de samplepunten kunnen wel berekend worden uit de filters h en g , maar voor N discretisatiepunten zou dit $2N^2$ geheugenplaatsen vereisen. Dat is erg onpraktisch.

Als we het gebruik van de non-standaard vorm niet beperken tot een lineaire algebra tool, maar volledig in de wavelet ruimte werken hebben we een ander probleem. We zijn namelijk vertrokken van een redundante wavelet voorstelling van f . Al wat er moet gebeuren om deze niet-redundant te maken is het weggooien van alle schaling-coëfficiënten, behalve die op de grofste schaal (N). Na toepassing van de operator

Deze matrix is geen voorstelling van de operator in één of andere tweedimensionale waveletbasis. De matrix is louter een hulpmiddel om eenvoudig alle nodige coëfficiënten te berekenen. Daar de matrices met index $j+1$ steeds half zo groot zijn als de matrices met index j , kunnen al de matrices in elkaar geschoven worden tot één $2^J \times 2^J$ matrix. Deze matrix is wel een voorstelling in waveletbasis, nl. de voorstelling van de operator, beschouwd als discrete functie van twee veranderlijken, in de wavelet basis geconstrueerd aan de hand van een tweedimensionale MRA. De projecties op de ruimten W_j vinden we in (4.11) terug als de termen $A_j + B_j + \Gamma_j$, de projectie op V_N als de term T_N .

Deze basis wordt o.a. ook gebruikt voor beeldbewerking. De non-standaard vorm kan men dus vinden door de operator-matrix als 'beeld' te beschouwen en het beeld in waveletbasis te schrijven. Als het 'beeld' goed comprimeerbaar is, zal deze matrix ijl zijn, eventueel na verwijdering van coëfficiënten die kleiner zijn dan een bepaalde drempelwaarde. Deze matrix wordt dan 'uitgeschoven' om een matrix zoals in (4.24) te bekomen.

Het verband tussen de non-standaard vorm en de tweedimensionale MRA tensor-product wavelet basis kunnen we als volgt zien: zij $t(x,y)$ de discretisatiematrix van operator T beschouwd als discrete functie van twee veranderlijken, met $x, y = 1 \dots 2^J$. We kunnen $t(x,y)$ ontbinden in de tweedimensionale MRA basis (cf. 4.1.2):

$$\begin{aligned}
 t(x,y) = & \sum_{j=1}^N \left[\sum_{k=1}^{2^{J-j}} \sum_{l=1}^{2^{J-j}} \langle t(x,y), \psi_{jk}(x) \psi_{jl}(y) \rangle \psi_{jk}(x) \psi_{jl}(y) \right. \\
 & + \sum_{k=1}^{2^{J-j}} \sum_{l=1}^{2^{J-j}} \langle t(x,y), \varphi_{jk}(x) \psi_{jl}(y) \rangle \varphi_{jk}(x) \psi_{jl}(y) \\
 & + \left. \sum_{k=1}^{2^{J-j}} \sum_{l=1}^{2^{J-j}} \langle t(x,y), \psi_{jk}(x) \varphi_{jl}(y) \rangle \psi_{jk}(x) \varphi_{jl}(y) \right] \\
 & + \sum_{k=1}^{2^{J-N}} \sum_{l=1}^{2^{J-N}} \langle t(x,y), \varphi_{Nk}(x) \varphi_{Nl}(y) \rangle \varphi_{Nk}(x) \varphi_{Nl}(y)
 \end{aligned}$$

Voor de verschillende matrices die hierin voorkomen geldt:

$$\begin{aligned}
 \{ \langle t(x,y), \psi_{jk}(x) \psi_{jl}(y) \rangle \} &= A'_j \\
 \{ \langle t(x,y), \varphi_{jk}(x) \psi_{jl}(y) \rangle \} &= B'_j \\
 \{ \langle t(x,y), \psi_{jk}(x) \varphi_{jl}(y) \rangle \} &= \Gamma'_j \\
 \{ \langle t(x,y), \varphi_{Nk}(x) \varphi_{Nl}(y) \rangle \} &= T_N
 \end{aligned}$$

Dit kan op juist dezelfde manier aangetoond worden als we gedaan hebben bij de standaard vorm.

Daar de elementen van de matrices A'_j , B'_j en Γ'_j geen schalen mengen (zie (4.17), (4.18) en (4.20)) zal het verval van de matrixelementen snel zijn: $x_{k'l}^j$ met $x \in \{a, b, \gamma\}$ zal klein

zijn als k' en k ver genoeg uit elkaar liggen. Daar er geen grote coëfficiënten ontstaan door menging van schalen is de non-standaard vorm van een operator in het algemeen een stuk ijler dan de standaard vorm. Zeker voor grote matrices (dus veel discretisatiepunten) zal (4.24) dus een stuk sneller kunnen berekend worden dan (4.9). Daar staat tegenover dat, zoals reeds aangehaald, het werk niet af is na de matrix-vector vermenigvuldiging (4.24). De coëfficiënten $\widehat{d}_{jk'}$, $\widehat{s}_{jk'}$ en $\widehat{s}_{k'}$ vormen immers op niet-triviale manier een redundante voorstelling van de gewilde resultaat $T_0 f$. Hoe we de waveletprojectie van het gewilde resultaat eruit berekenen wordt behandeld in sectie 4.2.4. Deze berekeningen kosten natuurlijk ook tijd. Dat is de prijs die we moeten betalen voor de ont koppeling van de verschillende schalen. Bij standaard vorm operatoren zit de koppeling tussen de schalen in de matrix zelf. Bij de non-standaard vorm moeten we de koppeling achteraf nog in rekening brengen.

De non-standaard vorm van de $\frac{d}{dx}$ operator is te zien in figuur 4.2. De typische vingers van de standaard vorm hebben we hier niet, want er is geen interactie tussen de verschillende schalen. Hoewel de matrix een stuk groter is dan de standaard vorm (om precies te zijn is het een $R \times R$ matrix, met $R = \sum_{i=0}^{N-1} 2^{J-i}$, N de grofste schaal en $1 \leq N \leq J$), is hij beduidend ijler.

Voor we verdergaan bekijken we eerst enkele eigenschappen van de operatoren P_j en Q_j die we gaan nodig hebben.

4.2.3 Enkele eigenschappen van de projectieoperatoren P_j en Q_j

De operator P_j projecteert een functie f op de ruimte V_j , Q_j projecteert op W_j . Daar $V_j = V_{j+1} \oplus W_{j+1}$ hebben we

$$P_j = P_{j+1} + Q_{j+1}$$

Volgende regels gelden bij samenstellingen van de operatoren P_j en Q_j :

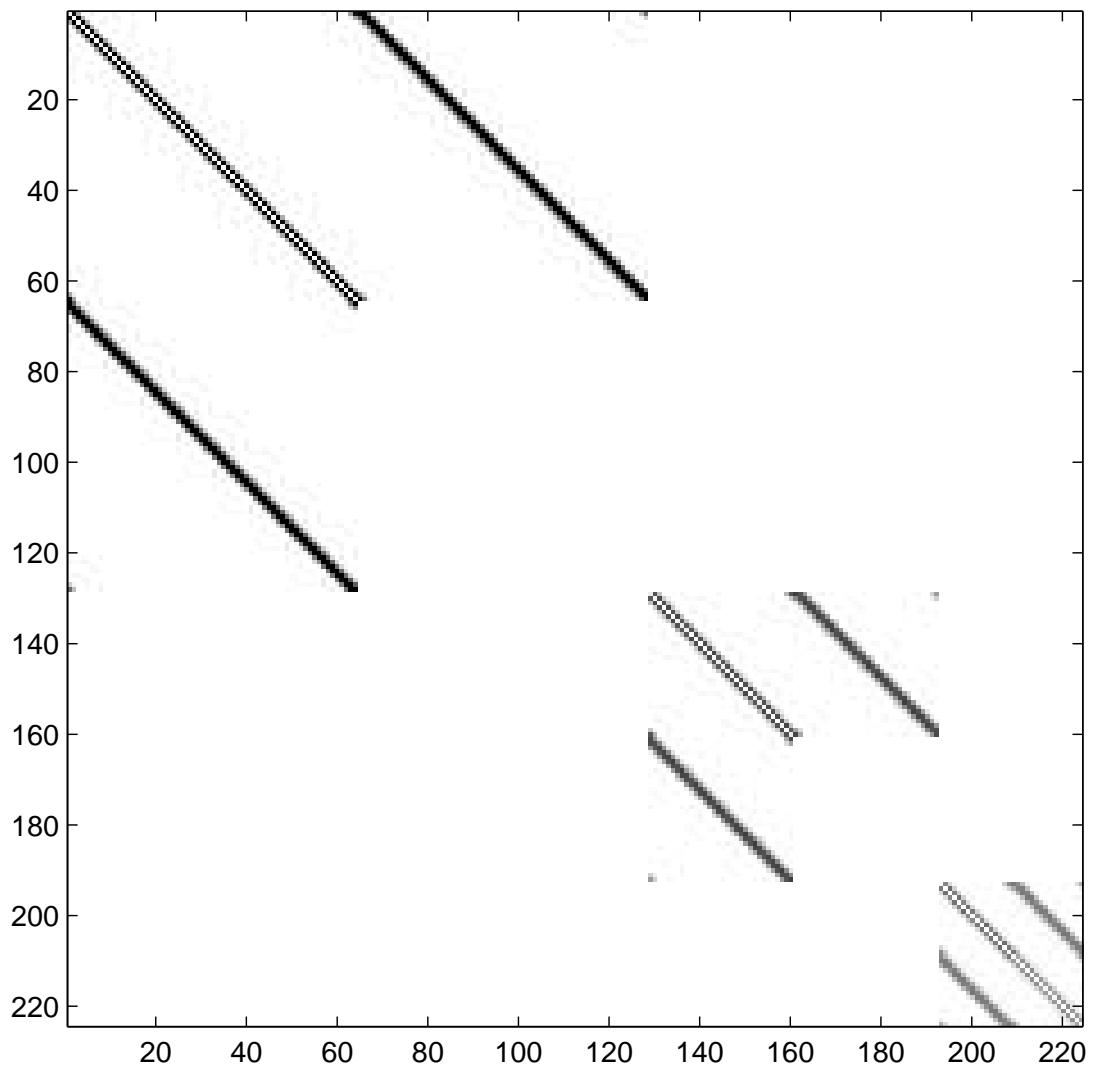
- Daar $V_k \subset V_j$ als $k \geq j$ geldt

$$P_k P_j = P_j P_k = P_k \quad k \geq j \quad (4.25)$$

-

$$\begin{aligned} Q_k Q_j &= 0 & k \neq j \\ Q_k Q_j &= Q_k = Q_j & k = j \end{aligned} \quad (4.26)$$

Dit kunnen we als volgt inzien: W_j is het orthogonale complement is van V_j in V_{j-1} ($V_{j-1} = V_j \oplus W_j$). Stel $k > j$. Dan geldt dus $W_k \subset V_j \perp W_j \Rightarrow W_k \perp W_j$, dus: $Q_k Q_j = 0$ voor $k > j$. Als we $k < j$ stellen hebben we: $W_j \subset V_k \perp W_k$ dus $Q_k Q_j = 0$ voor $k < j$. Voor $k = j$ geldt natuurlijk $Q_k Q_k = Q_k$.



Figuur 4.2: Non-standaard vorm van de operator $\frac{d}{dx}$ berekend met de Daubechies wavelet met 2 nulmomenten. We zien dat de matrix groter is dan de standaard vorm (zie figuur 4.1). Het is dus geen voorstelling in een bepaalde basis.

•

$$\begin{aligned} P_k Q_j &= 0 & k \geq j \\ P_k Q_j &= Q_j & k < j \end{aligned} \quad (4.27)$$

want: Als $k \geq j$, geldt $V_k \subset V_j \perp W_j$, dus $P_k Q_j = 0$. Voor $k < j$ hebben we $W_j \subset V_k$, zodat $P_k Q_j = Q_j$.

•

$$\begin{aligned} Q_k P_j &= 0 & k \leq j \\ Q_k P_j &= Q_k & k > j \end{aligned} \quad (4.28)$$

want: Voor $k \leq j$ geldt $V_j \subset V_k \perp W_k$, zodat $Q_k P_j = 0$. Als $k > j$ geldt $W_k \subset V_j$ zodat $Q_k P_j = Q_k$.

4.2.4 Herprojectie na non-standaard vorm matrix-vector vermenigvuldiging

Om na te gaan hoe we na een non-standaard vorm matrix-vectorvermenigvuldiging (4.24) een waveletrepresentatie van het resultaat kunnen bekijken beschouwen we weer een operator T met als benadering op schaal 0 (door discretisatie, 0 = fijnste schaal) $T_0 = P_0 T P_0$. T_0 heeft volgende non-standaard vorm ontbinding [20] (zie (4.11)):

$$T_0 = \sum_{j=1}^N (A_j + B_j + \Gamma_j) + T_N \quad (4.29)$$

met

$$\begin{aligned} A_j &= Q_j T Q_j \\ B_j &= Q_j T P_j \\ \Gamma_j &= P_j T Q_j \\ T_j &= P_j T P_j \end{aligned}$$

Na toepassen van de operator op een functie, staat het resultaat niet meer in de wavelet-basis. $T_0 f$ (met f de functie) bestaat uit een som van contributies op verschillende schalen,

die we als volgt terug op de wavelet-basis kunnen projecteren:

$$\begin{aligned}
T_0 f &= \sum_{k=1}^N Q_k T_0 f + P_N T_0 f \\
&= \sum_{k=1}^N \left(\sum_{j=1}^N (Q_k A_j f + Q_k B_j f + Q_k \Gamma_j f) + Q_k T_N f \right) \\
&\quad + \sum_{j=1}^N (P_N A_j f + P_N B_j f + P_N \Gamma_j f) + P_N T_N f \\
&= \sum_{k=1}^N \sum_{j=1}^N (Q_k A_j f + Q_k B_j f + Q_k \Gamma_j f) + \sum_{k=1}^N Q_k T_N f \\
&\quad + \sum_{j=1}^N (P_N A_j f + P_N B_j f + P_N \Gamma_j f) + T_N f
\end{aligned} \tag{4.30}$$

We zullen nu de termen van (4.30) nader bekijken:

- De som $\sum_{k=1}^N \sum_{j=1}^N (Q_k A_j f + Q_k B_j f + Q_k \Gamma_j f)$ bestaat uit drie types termen:
 - De termen $Q_k A_j f = Q_k Q_j T Q_j f$ $k, j = 1 \dots N$ zijn 0 voor $k \neq j$ wegens (4.26). Voor $k = j$ hebben we $Q_j A_j f = Q_j Q_j T Q_j f = Q_j T Q_j f = A_j f$.
 - Volledig analoog zien we dat de termen $Q_k B_j f = Q_k Q_j T P_j f$ $k, j = 1 \dots N$ nul zijn voor $k \neq j$ en gelijk aan $B_j f$ voor $k = j$.
 - De termen $Q_k \Gamma_j f = Q_k P_j T Q_j f$ $k, j = 1 \dots N$ zijn nul als $k \leq j$ (zie (4.28)).

We hebben dus:

$$\begin{aligned}
\sum_{k=1}^N \sum_{j=1}^N (Q_k A_j f + Q_k B_j f + Q_k \Gamma_j f) &= \sum_{j=1}^N (Q_j A_j f + Q_j B_j f) \\
&\quad + \sum_{j=1}^{N-1} \sum_{k=j+1}^N Q_k \Gamma_j f \\
&= \sum_{j=1}^N (A_j f + B_j f) \\
&\quad + \sum_{j=1}^{N-1} \sum_{k=j+1}^N Q_k \Gamma_j f
\end{aligned}$$

- De tweede som

$$\sum_{k=1}^N Q_k T_N f = \sum_{k=1}^N Q_k P_N T P_N f = 0$$

is nul want $Q_k P_j = 0$ voor $k \leq j$ (zie (4.28)).

- In $\sum_{j=1}^N (P_N A_j f + P_N B_j f + P_N \Gamma_j f)$ zijn de termen in A_j en B_j nul:

$$P_N A_j f = P_N Q_j T Q_j f = 0$$

$$P_N B_j f = P_N Q_j T P_j f = 0$$

want $P_j Q_k = 0$ voor $j \geq k$ (zie (4.27)).

De wavelet-expansie van T_0 (4.30) wordt dus:

$$T_0 f = \sum_{j=1}^N (A_j f + B_j f) + \underbrace{\sum_{j=1}^{N-1} \sum_{k=j+1}^N Q_k \Gamma_j f}_{\text{onderlijnde}} + \underbrace{\sum_{j=1}^N P_N \Gamma_j f}_{\text{onderlijnde}} + T_N f \quad (4.31)$$

Enkel de onderlijnde sommen bevatten termen die we nog niet kennen na het berekenen van een non-standaard vorm matrix-vector produkt (vgl. (4.31) met (4.29)).

De praktische methode om de onbekende termen te berekenen zullen we illustreren voor het geval $N = 3$. (4.31) wordt dan:

$$T_0 f = A_1 f + B_1 f + A_2 f + B_2 f + A_3 f + B_3 f + \underline{Q_2 \Gamma_1 f} \\ + \underline{Q_3 \Gamma_1 f} + \underline{Q_3 \Gamma_2 f} + \underline{P_3 \Gamma_1 f} + \underline{P_3 \Gamma_2 f} + \Gamma_3 f + T_N f$$

waarbij we de onderlijnde termen willen berekenen. Beylkin [23] doet dat zo:

$$\Gamma_1 f \rightarrow P_2 \Gamma_1 f \rightarrow P_2 (\Gamma_1 f + \Gamma_2 f) \rightarrow \underline{P_3 (\Gamma_1 f + \Gamma_2 f)} \\ \searrow \underline{Q_2 \Gamma_1 f} \qquad \qquad \qquad \searrow \underline{Q_3 (\Gamma_1 f + \Gamma_2 f)}$$

We beginnen met $\Gamma_1 f$ (de schaling coëfficiënten op schaal 1 na vermenigvuldiging met de operator-matrix). $\Gamma_1 f$ wordt geprojecteerd op de ruimten V_2 en W_2 (één stap uit de standaard wavelet transform). De projectie op W_2 levert al één onbekende term op. Om de anderen te vinden tellen we $P_2 \Gamma_2 f$ (gekend!) op bij $P_2 \Gamma_1 f$ (de projectie op V_2) en we projecteren opnieuw op twee orthogonale ruimten, dit keer V_3 en W_3 . Dit levert de overige termen. Optellen van de termen in de verschillende ruimten geeft nu $T_0 f$ in waveletbasis:

$$T_0 f = Q_1 (A_1 + B_1) f + Q_2 (A_2 + B_2 + \Gamma_1) f + Q_3 (A_3 + B_3 + \Gamma_1 + \Gamma_2) f \\ + P_3 (\Gamma_1 + \Gamma_2 + \Gamma_3) f + T_N f$$

In het algemene geval projecteert men op subruimten V_j en W_j , telt de resultaten op bij respectievelijk $P_j \Gamma_j f$ en $Q_j \Gamma_j f$, projecteert het resultaat van de eerste optelling dan verder op V_{j+1} en W_{j+1} , enz. tot de grofste schaal (N) bereikt is.

Het herprojecteren kost dus ongeveer *evenveel tijd* als een *gewone wavelet-transform* en heeft rekenkundige complexiteit $O(N)$.

4.2.5 De operator $\frac{d}{dx}$

Als belangrijk voorbeeld bespreken we het berekenen van de operator $\frac{d}{dx}$ in non-standaard vorm [24, 25].

Uit (4.17) weten we dat de matrixelement (i, l) van A_j^l gegeven worden door

$$a_{il}^j = \langle T\psi_{j,l}, \psi_{j,i} \rangle = \int_{-\infty}^{+\infty} T\psi_{j,l}(x) \overline{\psi_{j,i}(x)} dx \quad (4.32)$$

We zullen veronderstellen dat we met reële wavelet- en schalingfuncties werken. Daar

$$\psi_{j,k}(x) = \frac{1}{\sqrt{2^j}} \psi\left(\frac{x-2^j k}{2^j}\right) = \frac{1}{\sqrt{2^j}} \psi(2^{-j}x - k)$$

wordt (4.32)

$$\begin{aligned} a_{il}^j &= \int_{-\infty}^{+\infty} T\psi_{j,l}(x) \psi_{j,i}(x) dx \\ &= 2^{-j} \int_{-\infty}^{+\infty} \frac{d}{dx} \psi(2^{-j}x - l) \cdot \psi(2^{-j}x - i) dx \end{aligned}$$

Als we stellen $u = 2^{-j}x - l$ wordt dit

$$\begin{aligned} a_{il}^j &= 2^{-j} \int_{-\infty}^{+\infty} \frac{d}{du} \psi(u) \frac{du}{dx} \cdot \psi(u + l - i) dx \\ &= 2^{-j} \int_{-\infty}^{+\infty} \frac{d}{du} \psi(u) 2^{-j} \cdot \psi(u - (i - l)) 2^j du \\ &= 2^{-j} \int_{-\infty}^{+\infty} \frac{d}{du} \psi(u) \cdot \psi(u - (i - l)) du \\ &= 2^{-j} a_{i-l} \end{aligned}$$

met

$$a_l = \int_{-\infty}^{+\infty} \psi(x - l) \frac{d}{dx} \psi(x) dx \quad (4.33)$$

Volledig analoog vinden we uit (4.18), (4.20) en (4.22)

$$\begin{aligned} b_{il}^j &= 2^{-j} \int_{-\infty}^{+\infty} \frac{d}{du} \varphi(u) \cdot \psi(u - (i - l)) du = 2^{-j} b_{i-l} \\ \gamma_{il}^j &= 2^{-j} \int_{-\infty}^{+\infty} \frac{d}{du} \psi(u) \cdot \varphi(u - (i - l)) du = 2^{-j} \gamma_{i-l} \\ r_{il}^j &= 2^{-j} \int_{-\infty}^{+\infty} \frac{d}{du} \varphi(u) \cdot \varphi(u - (i - l)) du = 2^{-j} r_{i-l} \end{aligned}$$

met

$$b_l = \int_{-\infty}^{+\infty} \psi(x-l) \frac{d}{dx} \varphi(x) dx \quad (4.34)$$

$$\gamma_l = \int_{-\infty}^{+\infty} \varphi(x-l) \frac{d}{dx} \psi(x) dx \quad (4.35)$$

$$r_l = \int_{-\infty}^{+\infty} \varphi(x-l) \frac{d}{dx} \varphi(x) dx \quad (4.36)$$

De matrices $\{a_{il}^j\}$, $\{b_{il}^j\}$, $\{\gamma_{il}^j\}$ en $\{r_{il}^j\}$ zijn zogenaamde Toeplitz-matrices. Een Toeplitz matrix M is een matrix wiens elementen m_{ij} enkel afhangen van het verschil $i-j$. De elementen van een Toeplitz matrix veranderen niet langs de hoofddiagonaal en zijn parallellen.

Als de wavelet- en schaling-functies functies met compacte drager zijn zullen veel matrixelementen nul zijn (als $|i-l|$ groot genoeg is). Daar de Daubechies wavelets de wavelets met compacte drager met de kleinst mogelijke draagwijdte (support) zijn voor een gegeven aantal nulmomenten (vanishing moments) M , zal de operator $\frac{d}{dx}$ gebaseerd op de Daubechies wavelets het meest ijl zijn.

Deze wavelets hebben nog een aantrekkelijke eigenschap bij de $\frac{d}{dx}$ operator, de zogenaamde *superconvergentie* (superconvergence) [26]. Een eindige verschillen $\frac{d}{dx}$ operator matrix wordt 'orde N accuraat' genoemd als de afgeleide toegepast op veeltermen tot en met graad N een exact resultaat oplevert (op afrondingsfouten na) en vanaf graad $N+1$ niet meer (het resultaat is dan nog slechts een benadering van de afgeleide). Jameson [26] bewijst dat $\frac{d}{dx}$ operatoren gebaseerd op de Daubechies wavelets met M nulmomenten orde $2M$ accuraat zijn. Dat is opmerkelijk, want wavelets met M nulmomenten kunnen slechts veeltermen tot graad $M-1$ exact voorstellen [26]. Daarom noemt men dit fenomeen *superconvergentie*.

Om (4.33)-(4.35) concreet uit te rekenen beginnen we met

$$\varphi(x) = \sqrt{2} \sum_{k=0}^{L-1} h_k \varphi(2x-k) \quad (4.37)$$

$$\psi(x) = \sqrt{2} \sum_{k=0}^{L-1} g_k \varphi(2x-k) \quad (4.38)$$

waarin $g_k = (-1)^k h_{L-k-1}$, $k = 0 \dots L-1$. L is de lengte van de QMF filters. (4.38) invullen in (4.33) levert

$$a_l = 2 \sum_{k=0}^{L-1} \sum_{k'=0}^{L-1} g_k g_{k'} \int_{-\infty}^{+\infty} \varphi(2(x-l)-k) \frac{d}{dx} \varphi(2x-k') dx$$

Stellen we $v = 2x - k'$ en gebruiken we (4.36) dan wordt dit

$$\begin{aligned} a_l &= 2 \sum_{k=0}^{L-1} \sum_{k'=0}^{L-1} g_k g_{k'} \int_{-\infty}^{+\infty} \varphi(v - (2l + k - k')) \frac{d}{dv} \varphi(v) \frac{dv}{dx} \frac{dv}{2} \\ &= 2 \sum_{k=0}^{L-1} \sum_{k'=0}^{L-1} g_k g_{k'} \int_{-\infty}^{+\infty} \varphi(v - (2l + k - k')) \frac{d}{dv} \varphi(v) dv \\ &= 2 \sum_{k=0}^{L-1} \sum_{k'=0}^{L-1} g_k g_{k'} r_{2l+k-k'} \end{aligned}$$

Volledig analoog vinden we voor (4.34) en (4.35)

$$\begin{aligned} b_l &= 2 \sum_{k=0}^{L-1} \sum_{k'=0}^{L-1} g_k h_{k'} r_{2l+k-k'} \\ \gamma_l &= 2 \sum_{k=0}^{L-1} \sum_{k'=0}^{L-1} h_k g_{k'} r_{2l+k-k'} \end{aligned}$$

Alle matrixelementen van de non-standaard vorm van de afgeleide hangen dus af van de getallen r_l , gegeven door (4.36). De r_l vormen de 'stencil' waarmee alle matrices kunnen opgesteld worden.

Beylkin [24] bewijst dat de r_l kunnen berekend worden door oplossing van een lineair stelsel (voor wavelets met compacte drager!). Hiervoor schrijven we eerst (4.36) in functie van de Fouriergetransformeerde $\hat{\varphi}(\xi)$ van $\varphi(x)$,

$$\hat{\varphi}(\xi) = \frac{1}{\sqrt{2}} \int_{-\infty}^{+\infty} \varphi(x) e^{ix\xi} dx,$$

zodat we hebben

$$r_l = \int_{-\infty}^{+\infty} (-i\xi) |\hat{\varphi}(\xi)|^2 e^{-il\xi} d\xi \quad (4.39)$$

Stelling [Beylkin] Als de integralen (4.36) of (4.39) bestaan, dan voldoen de coëfficiënten r_l , $l \in \mathbb{Z}$ gegeven in (4.36) aan het volgende stelsel lineaire algebraïsche vergelijkingen:

$$r_l = 2 \left[r_{2l} + \frac{1}{2} \sum_{k=1}^{L/2} a_{2k-1} (r_{2l-2k+1} + r_{2l+2k-1}) \right] \quad (4.40)$$

en

$$\sum_l r_l = -1 \quad (4.41)$$

De coëfficiënten a_{2k-1} zijn gegeven door

$$a_n = 2 \sum_{i=0}^{L-1-n} h_i h_{i+n}, \quad n = 1 \dots L-1$$

(dit zijn de autocorrelatiecoëfficiënten van het filter $H = \{h_k\}_{k=0}^{L-1}$).

Als $M = \text{aantal vanishing moments} \geq 2$ dan hebben vgl. (4.40) en (4.41) een unieke oplossing met een eindig aantal r_l verschillend van nul, nl. $r_l \neq 0$ voor $-L+2 \leq l \leq L-2$ en

$$r_l = -r_{-l}$$

Het bewijs is te vinden in [24].

De non-standaard vorm van de $\frac{d}{dx}$ operator voor 128 functiesamples is te zien in figuur (4.2) op pagina 67.

De standaard vorm van $\frac{d}{dx}$ kan berekend worden door de matrix $\{a_{ij}^0\}$ op te stellen en deze in standaard tensor-product waveletbasis voor te stellen. Deze is afgebeeld in figuur (4.1) op pagina 57.

4.2.6 Combinatie van twee non-standaard vorm operatoren

Beschouw twee operatoren T en U met als benadering op schaal 0 respectievelijk T_0 en U_0 . Voorgesteld in non-standaard vorm hebben we:

$$T_0 = \sum_{j=1}^N (A_j + B_j + \Gamma_j) + T_N$$

$$U_0 = \sum_{j=1}^N (A'_j + B'_j + \Gamma'_j) + U_N$$

Opmerking: de accenten in A'_j , B'_j en Γ'_j staan er enkel om deze operatoren (die bij U_0 horen) te onderscheiden van A_j , B_j en Γ_j (die bij T_0 horen). Ze hebben niets te maken met de matrices die in (4.17), (4.18) en (4.20) gedefinieerd werden.

Stel dat we soms U_0 willen toepassen na inwerking van T_0 en soms T_0 alleen. We kunnen U_0 (in non-standaard vorm) niet gewoon toepassen op het resultaat van de bewerking $T_0 f$. $T_0 f$ moet eerst weer in wavelet-basis geschreven worden (cf. 4.2.4). Dan kan U_0 toegepast worden, waarna we meestal weer op de wavelet-basis moeten projecteren (tenminste als alle bewerkingen in de wavelet-ruimte gebeuren). Dat voortdurend projecteren vergt telkens rekentijd. We zullen nu bestuderen of het mogelijk is U_0 zodanig aan te passen (zonder T_0 te veranderen, zodat we T_0 ook nog apart kunnen gebruiken!) dat de tussentijdse projectie niet nodig is. Dit zou handig zijn, want dan volstaat het om 2 operatoren te berekenen (T_0 en de gewijzigde U_0). Voor discretisaties met veel punten vereist het opstellen van non-standaard vorm operatoren nl. veel rekentijd. Het zal blijken dat deze methode praktisch niet haalbaar is.

Daarna zullen we kort een andere methode [25] bespreken om non-standaard vorm operatoren samen te stellen. Deze is praktisch gemakkelijker te realiseren maar is rekenintensief

en vereist veel computergeheugen: we moeten 3 operatoren opstellen, T_0 , U_0 en $U_0 \circ T_0$ (U_0 na T_0).

Combinatie zonder tussentijdse projectie

Om te onderzoeken of het niet mogelijk is U_0 aan te passen zodanig dat we tussen de toepassing van T_0 en U_0 niet moeten projecteren bekijken we de samenstelling van U_0 en T_0 (zonder tussentijdse projectie):

$$\begin{aligned}
 U_0 T_0 f &= \sum_{k=1}^N (A'_k T_0 f + B'_k T_0 f + \Gamma'_k T_0 f) + U_N T_0 f \\
 &\Downarrow \\
 U_0 T_0 f &= \sum_{k=1}^N \sum_{j=1}^N (A'_k A_j f + A'_k B_j f + A'_k \Gamma_j f) \\
 &\quad + \sum_{k=1}^N \sum_{j=1}^N (B'_k A_j f + B'_k B_j f + B'_k \Gamma_j f) \\
 &\quad + \sum_{k=1}^N \sum_{j=1}^N (\Gamma'_k A_j f + \Gamma'_k B_j f + \Gamma'_k \Gamma_j f) \\
 &\quad + \sum_{k=1}^N (A'_k T_N f + B'_k T_N f + \Gamma'_k T_N f) \\
 &\quad + \sum_{j=1}^N (U_N A_j f + U_N B_j f + U_N \Gamma_j f) + U_N T_N f
 \end{aligned} \tag{4.42}$$

Met behulp van (4.25)-(4.28) kunnen we (4.42) vereenvoudigen:

- $\sum_{k=1}^N \sum_{j=1}^N (A'_k A_j f + A'_k B_j f + A'_k \Gamma_j f)$:
 - $A'_k A_j f = Q_k U Q_k Q_j T Q_j f \neq 0$ voor $k = j$ wegens (4.26).
 - $A'_k B_j f = Q_k U Q_k Q_j T P_j f \neq 0$ voor $k = j$ (idem).
 - $A'_k \Gamma_j f = Q_k U Q_k P_j T Q_j f \neq 0$ voor $k \geq j$ wegens (4.28).

Dus:

$$\begin{aligned}
 \sum_{k=1}^N \sum_{j=1}^N (A'_k A_j f + A'_k B_j f + A'_k \Gamma_j f) &= \sum_{j=1}^N (A'_j A_j f + A'_j B_j f) \\
 &\quad + \sum_{j=1}^{N-1} \sum_{k=j+1}^N A'_k \Gamma_j f
 \end{aligned}$$

- $\sum_{k=1}^N \sum_{j=1}^N (B'_k A_j f + B'_k B_j f + B'_k \Gamma_j f)$:
 - $B'_k A_j f = Q_k U P_k Q_j T Q_j f \neq 0$ voor $k < j$ wegens (4.27).
 - $B'_k B_j f = Q_k U P_k Q_j T P_j f \neq 0$ voor $k < j$ (idem).

Dus:

$$\begin{aligned} \sum_{k=1}^N \sum_{j=1}^N (B'_k A_j f + B'_k B_j f + B'_k \Gamma_j f) &= \sum_{k=1}^{N-1} \sum_{j=k+1}^N (B'_k A_j f + B'_k B_j f) \\ &+ \sum_{k=1}^N \sum_{j=1}^N B'_k \Gamma_j f \end{aligned}$$

- $\sum_{k=1}^N \sum_{j=1}^N (\Gamma'_k A_j f + \Gamma'_k B_j f + \Gamma'_k \Gamma_j f)$:
 - $\Gamma'_k A_j f = P_k U Q_k Q_j T Q_j f \neq 0$ voor $k = j$ wegens (4.26).
 - $\Gamma'_k B_j f = P_k U Q_k Q_j T P_j f \neq 0$ voor $k = j$ (idem).
 - $\Gamma'_k \Gamma_j f = P_k U Q_k P_j T Q_j f \neq 0$ voor $k \geq j$ wegens (4.28).

Dus:

$$\begin{aligned} \sum_{k=1}^N \sum_{j=1}^N (\Gamma'_k A_j f + \Gamma'_k B_j f + \Gamma'_k \Gamma_j f) &= \sum_{j=1}^N (\Gamma'_j A_j f + \Gamma'_j B_j f) \\ &+ \sum_{j=1}^{N-1} \sum_{k=j+1}^N \Gamma'_k \Gamma_j f \end{aligned}$$

- $\sum_{k=1}^N (A'_k T_N f + B'_k T_N f + \Gamma'_k T_N f)$:
 - $A'_k T_N f = Q_k U Q_k P_N T P_N f = 0$ (wegens (4.28)).
 - $\Gamma'_k T_N f = P_k U Q_k P_N T P_N f = 0$ (idem).

Dus:

$$\sum_{k=1}^N (A'_k T_N f + B'_k T_N f + \Gamma'_k T_N f) = \sum_{k=1}^N B'_k T_N f$$

- $\sum_{j=1}^N (U_N A_j f + U_N B_j f + U_N \Gamma_j f)$:
 - $U_N A_j f = P_N U P_N Q_j T Q_j f = 0$ (wegens (4.27)).
 - $U_N B_j f = P_N U P_N Q_j T P_j f = 0$ (idem).

Dus:

$$\sum_{j=1}^N (U_N A_j f + U_N B_j f + U_N \Gamma_j f) = \sum_{j=1}^N U_N \Gamma_j f$$

(4.42) wordt dus:

$$\begin{aligned} U_0 T_0 f &= \sum_{j=1}^N (A'_j A_j f + A'_j B_j f) + \sum_{j=1}^{N-1} \sum_{k=j+1}^N A'_k \Gamma_j f \\ &+ \sum_{k=1}^{N-1} \sum_{j=k+1}^N (B'_k A_j f + B'_k B_j f) + \sum_{k=1}^N \sum_{j=1}^N B'_k \Gamma_j f \\ &+ \sum_{j=1}^N (\Gamma'_j A_j f + \Gamma'_j B_j f) + \sum_{j=1}^{N-1} \sum_{k=j+1}^N \Gamma'_k \Gamma_j f \\ &+ \sum_{k=1}^N B'_k T_N f + \sum_{j=1}^N U_N \Gamma_j f + U_N T_N f \end{aligned}$$

Het berekenen van de onderliggende termen kost veel tijd. Als we $T_0 f$ terug op de waveletbasis hadden geprojecteerd, zouden we dan reeds het rekenwerk uitgevoerd hebben dat nu in het bepalen van de onderliggende termen kruipt. $U_0 T_0 f$ staat ook nog niet in waveletbasis. Daarvoor moeten we herprojecteren:

$$\begin{aligned} U_0 T_0 f &= \sum_{l=1}^N Q_l U_0 T_0 f + P_N U_0 T_0 f \tag{4.43} \\ &= \sum_{j=1}^N (A'_j A_j f + A'_j B_j f) + \sum_{j=1}^{N-1} \sum_{k=j+1}^N A'_k \Gamma_j f \\ &+ \sum_{k=1}^{N-1} \sum_{j=k+1}^N (B'_k A_j f + B'_k B_j f) + \sum_{k=1}^N \sum_{j=1}^N B'_k \Gamma_j f \\ &+ \sum_{j=1}^{N-1} \sum_{l=j+1}^N (Q_l \Gamma'_j A_j f + Q_l \Gamma'_j B_j f) + \sum_{j=1}^{N-2} \sum_{k=j+1}^{N-1} \sum_{l=k+1}^N Q_l \Gamma'_k \Gamma_j f \\ &+ \sum_{k=1}^N B'_k T_N f + \sum_{j=1}^N (P_N \Gamma'_j A_j f + P_N \Gamma'_j B_j f) + \sum_{j=1}^{N-1} \sum_{k=j+1}^N P_N \Gamma'_k \Gamma_j f \\ &+ \sum_{j=1}^N U_N \Gamma_j f + U_N T_N f \end{aligned}$$

Veel termen hierin vergen veel rekentijd. De voornaamste oorzaak hiervan is de interactie tussen de verschillende schalen (die normaal tijdens de tussentijdse herprojectie al gedeeltelijk in rekening gebracht wordt). (4.43) is ook heel ingewikkeld om praktisch te

realiseren. Een simpele organisatie van deze bewerkingen in een matrix plus een aantal wavelet-transform stappen lijkt zeer moeilijk of onmogelijk te realiseren. We zien dus dat de tussentijdse projectie op deze manier vermijden een enorme stijging van de complexiteit geeft, heel waarschijnlijk zonder winst in rekestijd (zelfs verlies!).

Expansie van de samengestelde operator

Een andere mogelijkheid [25] is de rechtstreekse expansie van de samengestelde operator in een telescopische reeks. Dit zorgt voor een ontkoppeling van de verschillende schalen (zoals ook al gebleken is bij het opstellen van de non-standaard vorm zelf). We gaan als

volgt te werk:

$$\begin{aligned}
T_0 U_0 &= T_0 U_0 - T_1 U_1 + T_1 U_1 - T_2 U_2 + T_2 U_2 - \dots - T_N U_N + T_N U_N \\
&\Downarrow \\
T_0 U_0 - T_N U_N &= T_0 U_0 - T_1 U_1 + T_1 U_1 - T_2 U_2 + T_2 U_2 - \dots - T_N U_N \\
&= \sum_{j=1}^N [T_{j-1} U_{j-1} - T_j U_j] \\
&= \sum_{j=1}^N [(P_{j-1} T P_{j-1}) (P_{j-1} U P_{j-1}) - (P_j T P_j) (P_j U P_j)] \\
&\Downarrow P_{j-1} = P_j + Q_j \\
T_0 U_0 - T_N U_N &= \sum_{j=1}^N [((P_j + Q_j) T (P_j + Q_j)) ((P_j + Q_j) U (P_j + Q_j)) \\
&\quad - (P_j T P_j) (P_j U P_j)] \\
&= \sum_{j=1}^N [(P_j T P_j + P_j T Q_j + Q_j T P_j + Q_j T Q_j) \cdot \\
&\quad (P_j U P_j + P_j U Q_j + Q_j U P_j + Q_j U Q_j) - (P_j T P_j) (P_j U P_j)] \\
&\Downarrow P_j Q_j = Q_j P_j = 0 \\
T_0 U_0 - T_N U_N &= \sum_{j=1}^N [(P_j T P_j) (P_j U Q_j) + (P_j T Q_j) (Q_j U P_j) \\
&\quad + (P_j T Q_j) (Q_j U Q_j) + (Q_j T P_j) (P_j U P_j) + (Q_j T P_j) (P_j U Q_j) \\
&\quad + (Q_j T Q_j) (Q_j U P_j) + (Q_j T Q_j) (Q_j U Q_j)] \\
&= \sum_{j=1}^N [T_j \Gamma'_j + \Gamma_j B'_j + \Gamma_j A'_j + B_j U_j + B_j \Gamma'_j + A_j B'_j + A_j A'_j] \\
&= \sum_{j=1}^N [(A_j A'_j + B_j \Gamma'_j) + (B_j T'_j + A_j B'_j) + (T_j \Gamma'_j + \Gamma_j A'_j) \\
&\quad + \Gamma_j B'_j] \tag{4.44}
\end{aligned}$$

We zien dat in (4.44) geen verschillende schalen gemengd worden. De operatoren uit (4.44) zijn (gebruik (4.26) en (4.25))

$$\begin{aligned}
A_j A'_j + B_j \Gamma'_j &= Q_j T Q_j U Q_j + Q_j T P_j U Q_j &: W_j \rightarrow W_j \\
B_j U_j + A_j B'_j &= Q_j T P_j U P_j + Q_j T Q_j U P_j &: V_j \rightarrow W_j \\
T_j \Gamma'_j + \Gamma_j A'_j &= P_j T P_j U Q_j + P_j T Q_j U Q_j &: W_j \rightarrow V_j \\
\Gamma_j B'_j &= P_j T Q_j U P_j &: V_j \rightarrow V_j \\
T_N U_N &= P_N T P_N U P_N &: V_N \rightarrow V_N
\end{aligned}$$

met $j = 1 \dots N$. Als we stellen

$$\begin{aligned}\bar{A}_j &= A_j A'_j + B_j \Gamma'_j \\ \bar{B}_j &= B_j U_j + A_j B'_j \\ \bar{T}_j &= T_j \Gamma'_j + \Gamma_j A'_j \\ \bar{T}_N &= T_N U_N\end{aligned}$$

wordt (4.44)

$$T_0 U_0 = \sum_{j=1}^N (\bar{A}_j + \bar{B}_j + \bar{T}_j) + \bar{T}_N + \sum_{j=1}^N \Gamma_j B'_j \quad (4.45)$$

Als we dit vergelijken met (4.11) en (4.12)-(4.15) zien we dat (4.45) bijna in de non-standaard vorm staat, op de term $\sum_{j=1}^N \Gamma_j B'_j$ na. De matrices van de operatoren \bar{A}_j , \bar{B}_j , \bar{T}_j ($j = 1 \dots N$) en \bar{T}_N kunnen eenvoudig berekend worden als produkten en sommen van de matrices van de samenstellende operatoren.

We bekijken nu de probleemterm $\sum_{j=1}^N \Gamma_j B'_j$. Voor $j = 1 \dots N-1$ kunnen we van $\Gamma_j B'_j$ de non-standaard vorm berekenen, want $\Gamma_j B'_j = P_j T Q_j U P_j = P_j P_j T Q_j U P_j P_j = P_j \Gamma_j B'_j P_j$ (zie (4.25)) zodat

$$\Gamma_j B'_j = P_j \Gamma_j B'_j P_j = \sum_{k=j+1}^N (\tilde{A}_k^j + \tilde{B}_k^j + \tilde{T}_k^j) + \tilde{T}_N^j \quad (4.46)$$

(4.45) kan nu geschreven worden als

$$\begin{aligned}T_0 U_0 &= \sum_{j=1}^N (\bar{A}_j + \bar{B}_j + \bar{T}_j) + \bar{T}_N \\ &+ \sum_{j=1}^N \left(\sum_{k=j+1}^N (\tilde{A}_k^j + \tilde{B}_k^j + \tilde{T}_k^j) + \tilde{T}_N^j \right) \\ &= \sum_{j=1}^N \left[\left(\bar{A}_j + \sum_{k=j+1}^N \tilde{A}_k^j \right) + \left(\bar{B}_j + \sum_{k=j+1}^N \tilde{B}_k^j \right) \right. \\ &\quad \left. + \left(\bar{T}_j + \sum_{k=j+1}^N \tilde{T}_k^j \right) \right] + \left(\bar{T}_N + \sum_{j=1}^N \tilde{T}_N^j \right)\end{aligned}$$

Als we

$$\begin{aligned}\hat{A}_j &= \bar{A}_j + \sum_{k=j+1}^N \tilde{A}_k^j \quad (W_j \rightarrow W_j) \\ \hat{B}_j &= \bar{B}_j + \sum_{k=j+1}^N \tilde{B}_k^j \quad (V_j \rightarrow W_j) \\ \hat{\Gamma}_j &= \bar{\Gamma}_j + \sum_{k=j+1}^N \tilde{\Gamma}_k^j \quad (W_j \rightarrow V_j) \\ \hat{T}_N &= \bar{T}_N + \sum_{j=1}^N \tilde{T}_N^j \quad (V_j \rightarrow V_j)\end{aligned}$$

noemen krijgen we

$$T_0 U_0 = \sum_{j=1}^N (\hat{A}_j + \hat{B}_j + \hat{\Gamma}_j) + \hat{T}_N$$

Dit is de gewenste non-standaard vorm van de samengestelde operator $T_0 U_0$. De matrices van \hat{A}_j , \hat{B}_j , $\hat{\Gamma}_j$ en \hat{T}_N kunnen we weer berekenen als sommen en produkten van de matrices van \bar{A}_j en de \tilde{A}_k^j . De meest rekenintensieve stap is (4.46), het berekenen van de non-standaard vorm van $\Gamma_j B_j'$ voor $j = 1 \dots N$ (dus voor alle beschouwde j). Dit kan echter aanzienlijk versneld worden [25]:

We beginnen met $\Gamma_1 B_1'$ één niveau te ontbinden:

$$\Gamma_1 B_1' = \tilde{A}_2^1 + \tilde{B}_2^1 + \tilde{\Gamma}_2^1 + \tilde{T}_2^1$$

Merk op dat we operatoren als \tilde{T}_2^1 , of algemeen \tilde{T}_k^j , $j = 1 \dots N$, $k = j+1 \dots N-1$ nog niet eerder zijn tegengekomen. In (4.46) hebben we alleen \tilde{T}_N^j gebruikt. Voor we de volgende stap in de ontbinding uitvoeren tellen we $\Gamma_2 B_2'$ op bij $\Gamma_1 B_1'$:

$$\begin{aligned}\Gamma_1 B_1' + \Gamma_2 B_2' &= \tilde{A}_2^1 + \tilde{B}_2^1 + \tilde{\Gamma}_2^1 + \tilde{T}_2^1 + \Gamma_2 B_2' \\ &= \tilde{A}_2^1 + \tilde{B}_2^1 + \tilde{\Gamma}_2^1 + \tilde{T}\end{aligned}$$

Hierin is $\tilde{T} = \tilde{T}_2^1 + \Gamma_2 B_2'$. Door nu \tilde{T} verder te ontbinden, vinden we de ontbinding van de som $\tilde{T}_2^1 + \Gamma_2 B_2'$. Omdat we de aparte matrices toch niet nodig hebben, is dit voldoende. Door de som te maken hebben we hier twee ontbindingsstappen in één gecombineerd. Nu hebben we

$$\tilde{T} = \tilde{A} + \tilde{B} + \tilde{C} + \tilde{T}$$

waarna we weer doorgaan met de ontbinding van $\tilde{T} + \Gamma_3 B_3'$, enz. tot de grofste schaal bereikt is. Dat levert N matrices van het A -type (die we bij de overeenkomstige matrices van \bar{A}_j optellen), N van het B -type (bij de matrices van \bar{B}_j optellen), N van het Γ -type (bij de matrices van $\bar{\Gamma}_j$ optellen) en één matrix van het T_N -type die we bij \bar{T}_N optellen. We beschikken nu over alle matrices die in de non-standaard vorm van $T_0 U_0$ voorkomen.

4.3 Meerdere dimensies

Het construeren van operatoren in waveletbasis voor functies van meerdere veranderlijken is niet triviaal. We kunnen die functies nl. al op verschillende manieren in wavelet-basis schrijven (zie bv. sectie 4.1). De operatoren zullen er anders uitzien voor de verschillende gevallen. We hebben ook geen voorbeelden of publicaties gevonden waarop we ons kunnen baseren.

We zullen enkel het tweedimensionale geval behandelen. Meerdere dimensies kunnen volledig analoog ingevoerd worden. De formules worden dan wel vrij ingewikkeld.

4.3.1 Non-standaard vorm operator voor tweedimensionale functies

We zullen de non-standaard vorm construeren van een operator bedoeld voor tweedimensionale functies. Deze functies worden ontbonden in een tweedimensionale wavelet-basis geconstrueerd aan de hand van een MRA.

Principe

Beschouw een tweedimensionale MRA met orthogonale deelruimten V_j^2 en W_j^2 , met (cf. 4.1.2)

$$\begin{aligned} V_j^2 &= V_j^x \otimes V_j^y \\ W_j^2 &= (V_j^x \otimes W_j^y) \oplus (W_j^x \otimes V_j^y) \oplus (W_j^x \otimes W_j^y) \end{aligned}$$

De basisfuncties van V_j^2 zijn

$$\{\varphi_{j,k}(x) \varphi_{j,l}(y)\}_{k,l \in \mathbb{Z}}$$

en die van W_j^2

$$\{\psi_{j,k}(x) \varphi_{j,l}(y), \varphi_{j,k}(x) \psi_{j,l}(y), \psi_{j,k}(x) \psi_{j,l}(y)\}_{k,l \in \mathbb{Z}}$$

De projectieoperatoren $P_j^{(2)}$ en $Q_j^{(2)}$, die een tweedimensionale functie $f(x, y)$ projecteren op respectievelijk V_j^2 en W_j^2 , worden dus gegeven door

$$\begin{aligned} P_j^{(2)} f &= \sum_{k=1}^{2^{j-1}} \sum_{l=1}^{2^{j-1}} \langle f(x, y), \varphi_{j,k}(x) \varphi_{j,l}(y) \rangle \varphi_{j,k}(x) \varphi_{j,l}(y) \\ &= \sum_{k=1}^{2^{j-1}} \sum_{l=1}^{2^{j-1}} s_{jkl} \varphi_{j,k}(x) \varphi_{j,l}(y) \end{aligned} \quad (4.47)$$

$$\begin{aligned} Q_j^{(2)} f &= \sum_{k=1}^{2^{j-1}} \sum_{l=1}^{2^{j-1}} \langle f(x, y), \psi_{j,k}(x) \varphi_{j,l}(y) \rangle \psi_{j,k}(x) \varphi_{j,l}(y) \\ &\quad + \sum_{k=1}^{2^{j-1}} \sum_{l=1}^{2^{j-1}} \langle f(x, y), \varphi_{j,k}(x) \psi_{j,l}(y) \rangle \varphi_{j,k}(x) \psi_{j,l}(y) \\ &\quad + \sum_{k=1}^{2^{j-1}} \sum_{l=1}^{2^{j-1}} \langle f(x, y), \psi_{j,k}(x) \psi_{j,l}(y) \rangle \psi_{j,k}(x) \psi_{j,l}(y) \\ &= \sum_{k=1}^{2^{j-1}} \sum_{l=1}^{2^{j-1}} d_{jkl}^x \psi_{j,k}(x) \varphi_{j,l}(y) + \sum_{k=1}^{2^{j-1}} \sum_{l=1}^{2^{j-1}} d_{jkl}^y \varphi_{j,k}(x) \psi_{j,l}(y) \\ &\quad + \sum_{k=1}^{2^{j-1}} \sum_{l=1}^{2^{j-1}} d_{jkl}^{xy} \psi_{j,k}(x) \psi_{j,l}(y) \end{aligned} \quad (4.48)$$

met

$$\begin{aligned} s_{jkl} &= \langle f(x, y), \varphi_{j,k}(x) \varphi_{j,l}(y) \rangle \\ d_{jkl}^x &= \langle f(x, y), \psi_{j,k}(x) \varphi_{j,l}(y) \rangle \\ d_{jkl}^y &= \langle f(x, y), \varphi_{j,k}(x) \psi_{j,l}(y) \rangle \\ d_{jkl}^{xy} &= \langle f(x, y), \psi_{j,k}(x) \psi_{j,l}(y) \rangle \end{aligned}$$

Beschouw nu een functie $f(x, y)$. Operator T toepassen op de benadering (door discretisatie) van f , $P_0^{(2)} f$ en terug projecteren op V_0 geeft $P_0^{(2)} T P_0^{(2)} f = T_0 f$. Net zoals in het ééndimensionale geval (cf. 4.2.2) kunnen we dit ontbinden in een telescopische reeks (met N de grofste schaal):

$$\begin{aligned} T_0 f &= \sum_{j=1}^N \left(Q_j^{(2)} T Q_j^{(2)} f + Q_j^{(2)} T P_j^{(2)} f + P_j^{(2)} T Q_j^{(2)} f \right) + P_N^{(2)} T P_N^{(2)} f \\ &= \sum_{j=1}^N \left(A_j^{(2)} f + B_j^{(2)} f + \Gamma_j^{(2)} f \right) + T_N f \end{aligned} \quad (4.49)$$

met

$$\begin{aligned} A_j^{(2)} &= Q_j^{(2)} T Q_j^{(2)} \quad (W_j^2 \rightarrow W_j^2) \\ B_j^{(2)} &= Q_j^{(2)} T P_j^{(2)} \quad (V_j^2 \rightarrow W_j^2) \\ \Gamma_j^{(2)} &= P_j^{(2)} T Q_j^{(2)} \quad (W_j^2 \rightarrow V_j^2) \\ T_j &= P_j^{(2)} T P_j^{(2)} \quad (V_j^2 \rightarrow V_j^2) \end{aligned}$$

We kunnen (4.49) beschouwen als T_0 geschreven in de non-standaard vorm.

Uitwerking

Om de notatie wat te verlichten zullen we in het vervolg de x - en y -afhankelijkheid van de basisfuncties niet meer noteren. De eerst geschreven functie zal steeds afhankelijk zijn van x , de tweede van y . $\varphi_{j,k}(x)\psi_{j,l}(y)$ bv. zullen we dus noteren als $\varphi_{j,k}\psi_{j,l}$.

We zullen alle termen van (4.49) apart behandelen:

- De eerste term $A_j^{(2)} f = Q_j^{(2)} T Q_j^{(2)} f$, is de ingewikkeldste. We beginnen met $T Q_j^{(2)} f$ (zie (4.48)):

$$\begin{aligned} T Q_j^{(2)} f &= \sum_{k=1}^{2^{J-1}} \sum_{l=1}^{2^{J-1}} d_{jkl}^x T \psi_{j,k} \varphi_{j,l} + \sum_{k=1}^{2^{J-1}} \sum_{l=1}^{2^{J-1}} d_{jkl}^y T \varphi_{j,k} \psi_{j,l} \\ &\quad + \sum_{k=1}^{2^{J-1}} \sum_{l=1}^{2^{J-1}} d_{jkl}^{xy} T \psi_{j,k} \psi_{j,l} \end{aligned}$$

Terug projecteren op W_j^2 levert $A_j^{(2)} f = Q_j^{(2)} T Q_j^{(2)} f$:

$$\begin{aligned} Q_j^{(2)} T Q_j^{(2)} f &= \sum_{k'=1}^{2^{J-1}} \sum_{l'=1}^{2^{J-1}} \sum_{k=1}^{2^{J-1}} \sum_{l=1}^{2^{J-1}} [d_{jkl}^x (\langle T \psi_{j,k} \varphi_{j,l}, \psi_{j,k'} \varphi_{j,l'} \rangle \psi_{j,k'} \varphi_{j,l'}) \\ &\quad + \langle T \psi_{j,k} \varphi_{j,l}, \varphi_{j,k'} \psi_{j,l'} \rangle \varphi_{j,k'} \psi_{j,l'} \\ &\quad + \langle T \psi_{j,k} \varphi_{j,l}, \psi_{j,k'} \psi_{j,l'} \rangle \psi_{j,k'} \psi_{j,l'}) \\ &\quad + d_{jkl}^y (\langle T \varphi_{j,k} \psi_{j,l}, \psi_{j,k'} \varphi_{j,l'} \rangle \psi_{j,k'} \varphi_{j,l'}) \\ &\quad + \langle T \varphi_{j,k} \psi_{j,l}, \varphi_{j,k'} \psi_{j,l'} \rangle \varphi_{j,k'} \psi_{j,l'} \\ &\quad + \langle T \varphi_{j,k} \psi_{j,l}, \psi_{j,k'} \psi_{j,l'} \rangle \psi_{j,k'} \psi_{j,l'}) \\ &\quad + d_{jkl}^{xy} (\langle T \psi_{j,k} \psi_{j,l}, \psi_{j,k'} \varphi_{j,l'} \rangle \psi_{j,k'} \varphi_{j,l'}) \\ &\quad + \langle T \psi_{j,k} \psi_{j,l}, \varphi_{j,k'} \psi_{j,l'} \rangle \varphi_{j,k'} \psi_{j,l'} \\ &\quad + \langle T \psi_{j,k} \psi_{j,l}, \psi_{j,k'} \psi_{j,l'} \rangle \psi_{j,k'} \psi_{j,l'})] \end{aligned}$$

Hierin komen negen vierdimensionale 'matrices' (tensoren) voor:

$$\begin{aligned}
 \{a_{k'l'l'}^{j,x,x}\} &= \{\langle T\psi_{j,k}\varphi_{j,l}, \psi_{j,k'}\varphi_{j,l'} \rangle\} \\
 \{a_{k'l'l'}^{j,x,y}\} &= \{\langle T\psi_{j,k}\varphi_{j,l}, \varphi_{j,k'}\psi_{j,l'} \rangle\} \\
 \{a_{k'l'l'}^{j,x,xy}\} &= \{\langle T\psi_{j,k}\varphi_{j,l}, \psi_{j,k'}\psi_{j,l'} \rangle\} \\
 \{a_{k'l'l'}^{j,y,x}\} &= \{\langle T\varphi_{j,k}\psi_{j,l}, \psi_{j,k'}\varphi_{j,l'} \rangle\} \\
 \{a_{k'l'l'}^{j,y,y}\} &= \{\langle T\varphi_{j,k}\psi_{j,l}, \varphi_{j,k'}\psi_{j,l'} \rangle\} \\
 \{a_{k'l'l'}^{j,y,xy}\} &= \{\langle T\varphi_{j,k}\psi_{j,l}, \psi_{j,k'}\psi_{j,l'} \rangle\} \\
 \{a_{k'l'l'}^{j,xy,x}\} &= \{\langle T\psi_{j,k}\psi_{j,l}, \psi_{j,k'}\varphi_{j,l'} \rangle\} \\
 \{a_{k'l'l'}^{j,xy,y}\} &= \{\langle T\psi_{j,k}\psi_{j,l}, \varphi_{j,k'}\psi_{j,l'} \rangle\} \\
 \{a_{k'l'l'}^{j,xy,xy}\} &= \{\langle T\psi_{j,k}\psi_{j,l}, \psi_{j,k'}\psi_{j,l'} \rangle\}
 \end{aligned} \tag{4.50}$$

Dit zijn allemaal $2^{J-j} \times 2^{J-j} \times 2^{J-j} \times 2^{J-j}$ tensoren. Noemen we

$$\begin{aligned}
 d_{jk'l'}^{a,x} &= \sum_{k=1}^{2^{J-1}} \sum_{l=1}^{2^{J-1}} (\langle T\psi_{j,k}\varphi_{j,l}, \psi_{j,k'}\varphi_{j,l'} \rangle d_{jkl}^x \\
 &\quad + \langle T\varphi_{j,k}\psi_{j,l}, \psi_{j,k'}\varphi_{j,l'} \rangle d_{jkl}^y \\
 &\quad + \langle T\psi_{j,k}\psi_{j,l}, \psi_{j,k'}\varphi_{j,l'} \rangle d_{jkl}^{xy}) \\
 d_{jk'l'}^{a,y} &= \sum_{k=1}^{2^{J-1}} \sum_{l=1}^{2^{J-1}} (\langle T\psi_{j,k}\varphi_{j,l}, \varphi_{j,k'}\psi_{j,l'} \rangle d_{jkl}^x \\
 &\quad + \langle T\varphi_{j,k}\psi_{j,l}, \varphi_{j,k'}\psi_{j,l'} \rangle d_{jkl}^y \\
 &\quad + \langle T\psi_{j,k}\psi_{j,l}, \varphi_{j,k'}\psi_{j,l'} \rangle d_{jkl}^{xy}) \\
 d_{jk'l'}^{a,xy} &= \sum_{k=1}^{2^{J-1}} \sum_{l=1}^{2^{J-1}} (\langle T\psi_{j,k}\varphi_{j,l}, \psi_{j,k'}\psi_{j,l'} \rangle d_{jkl}^x \\
 &\quad + \langle T\varphi_{j,k}\psi_{j,l}, \psi_{j,k'}\psi_{j,l'} \rangle d_{jkl}^y \\
 &\quad + \langle T\psi_{j,k}\psi_{j,l}, \psi_{j,k'}\psi_{j,l'} \rangle d_{jkl}^{xy})
 \end{aligned}$$

(dit zijn tensor-matrix vermenigvuldigingen) dan krijgen we

$$Q_j^{(2)} T Q_j^{(2)} f = \sum_{k'=1}^{2^{J-1}} \sum_{l'=1}^{2^{J-1}} \left(d_{jk'l'}^{a,x} \psi_{j,k'}\varphi_{j,l'} + d_{jk'l'}^{a,y} \varphi_{j,k'}\psi_{j,l'} + d_{jk'l'}^{a,xy} \psi_{j,k'}\psi_{j,l'} \right)$$

- De tweede term van (4.49), $B_j^{(2)} f = Q_j^{(2)} T P_j^{(2)} f$ is gelukkig eenvoudiger. We beginnen met $T P_j^{(2)} f$ (zie (4.47)):

$$T P_j^{(2)} f = \sum_{k=1}^{2^{J-1}} \sum_{l=1}^{2^{J-1}} s_{jkl} T \varphi_{j,k} \varphi_{j,l}$$

Projecteren op W_j^2 geeft

$$\begin{aligned} Q_j^{(2)} T P_j^{(2)} f &= \sum_{k'=1}^{2^{J-1}} \sum_{l'=1}^{2^{J-1}} \sum_{k=1}^{2^{J-1}} \sum_{l=1}^{2^{J-1}} s_{jkl} \left(\langle T \varphi_{j,k} \varphi_{j,l}, \psi_{j,k'} \varphi_{j,l'} \rangle \psi_{j,k'} \varphi_{j,l'} \right. \\ &\quad + \langle T \varphi_{j,k} \varphi_{j,l}, \varphi_{j,k'} \psi_{j,l'} \rangle \varphi_{j,k'} \psi_{j,l'} \\ &\quad \left. + \langle T \varphi_{j,k} \varphi_{j,l}, \psi_{j,k'} \psi_{j,l'} \rangle \psi_{j,k'} \psi_{j,l'} \right) \end{aligned}$$

Noemen we

$$\begin{aligned} \{b_{k'k'l'}^{j,x}\} &= \{\langle T \varphi_{j,k} \varphi_{j,l}, \psi_{j,k'} \varphi_{j,l'} \rangle\} \\ \{b_{k'k'l'}^{j,y}\} &= \{\langle T \varphi_{j,k} \varphi_{j,l}, \varphi_{j,k'} \psi_{j,l'} \rangle\} \\ \{b_{k'k'l'}^{j,xy}\} &= \{\langle T \varphi_{j,k} \varphi_{j,l}, \psi_{j,k'} \psi_{j,l'} \rangle\} \end{aligned}$$

(alle drie $2^{J-j} \times 2^{J-j} \times 2^{J-j} \times 2^{J-j}$ tensoren) en

$$\begin{aligned} d_{jk'l'}^{b,x} &= \sum_{k=1}^{2^{J-1}} \sum_{l=1}^{2^{J-1}} \langle T \varphi_{j,k} \varphi_{j,l}, \psi_{j,k'} \varphi_{j,l'} \rangle s_{jkl} \\ d_{jk'l'}^{b,y} &= \sum_{k=1}^{2^{J-1}} \sum_{l=1}^{2^{J-1}} \langle T \varphi_{j,k} \varphi_{j,l}, \varphi_{j,k'} \psi_{j,l'} \rangle s_{jkl} \\ d_{jk'l'}^{b,xy} &= \sum_{k=1}^{2^{J-1}} \sum_{l=1}^{2^{J-1}} \langle T \varphi_{j,k} \varphi_{j,l}, \psi_{j,k'} \psi_{j,l'} \rangle s_{jkl} \end{aligned}$$

dan krijgen we

$$Q_j^{(2)} T P_j^{(2)} f = \sum_{k'=1}^{2^{J-1}} \sum_{l'=1}^{2^{J-1}} \left(d_{jk'l'}^{b,x} \psi_{j,k'} \varphi_{j,l'} + d_{jk'l'}^{b,y} \varphi_{j,k'} \psi_{j,l'} + d_{jk'l'}^{b,xy} \psi_{j,k'} \psi_{j,l'} \right)$$

We zien dat de $d_{jk'l'}^{b,x}$, $d_{jk'l'}^{b,y}$ en $d_{jk'l'}^{b,xy}$ afhangen van de schalingcoëfficiënten s_{jkl} . Om de non-standaard vorm operator te kunnen toepassen moeten we van f dus de schalingcoëfficiënten op elke schaal bijhouden en niet enkel die op de grofste schaal N . Dit is typisch voor de non-standaard vorm.

Door te stellen

$$\begin{aligned}\widehat{d}_{jk'l'}^x &= d_{jk'l'}^{a,x} + d_{jk'l'}^{b,x} \\ \widehat{d}_{jk'l'}^y &= d_{jk'l'}^{a,y} + d_{jk'l'}^{b,y} \\ \widehat{d}_{jk'l'}^{xy} &= d_{jk'l'}^{a,xy} + d_{jk'l'}^{b,xy}\end{aligned}$$

vinden we $A_j^{(2)}f + B_j^{(2)}f$:

$$\begin{aligned}A_j^{(2)}f + B_j^{(2)}f &= \sum_{k'=1}^{2^{J-1}} \sum_{l'=1}^{2^{J-1}} \left(\widehat{d}_{jk'l'}^x \psi_{j,k'} \varphi_{j,l'} + \widehat{d}_{jk'l'}^y \varphi_{j,k'} \psi_{j,l'} \right. \\ &\quad \left. + \widehat{d}_{jk'l'}^{xy} \psi_{j,k'} \psi_{j,l'} \right)\end{aligned}\quad (4.51)$$

- Analoge aanpak bij de derde term van (4.49), $\Gamma_j^{(2)}f = P_j^{(2)}TQ_j^{(2)}f$. We beginnen weer met $TQ_j^{(2)}f$:

$$TQ_j^{(2)}f = \sum_{k=1}^{2^{J-1}} \sum_{l=1}^{2^{J-1}} \left(d_{jkl}^x T\psi_{j,k} \varphi_{j,l} + d_{jkl}^y T\varphi_{j,k} \psi_{j,l} + d_{jkl}^{xy} T\psi_{j,k} \psi_{j,l} \right)$$

Projectie op V_j^2 geeft

$$\begin{aligned}P_j^{(2)}TQ_j^{(2)}f &= \sum_{k'=1}^{2^{J-1}} \sum_{l'=1}^{2^{J-1}} \sum_{k=1}^{2^{J-1}} \sum_{l=1}^{2^{J-1}} \left[d_{jkl}^x \langle T\psi_{j,k} \varphi_{j,l}, \varphi_{j,k'} \varphi_{j,l'} \rangle \varphi_{j,k'} \varphi_{j,l'} \right. \\ &\quad \left. + d_{jkl}^y \langle T\varphi_{j,k} \psi_{j,l}, \varphi_{j,k'} \varphi_{j,l'} \rangle \varphi_{j,k'} \varphi_{j,l'} \right. \\ &\quad \left. + d_{jkl}^{xy} \langle T\psi_{j,k} \psi_{j,l}, \varphi_{j,k'} \varphi_{j,l'} \rangle \varphi_{j,k'} \varphi_{j,l'} \right]\end{aligned}$$

Met

$$\begin{aligned}\{\gamma_{k'l'l}^{j,x}\} &= \{\langle T\psi_{j,k} \varphi_{j,l}, \varphi_{j,k'} \varphi_{j,l'} \rangle\} \\ \{\gamma_{k'l'l}^{j,y}\} &= \{\langle T\varphi_{j,k} \psi_{j,l}, \varphi_{j,k'} \varphi_{j,l'} \rangle\} \\ \{\gamma_{k'l'l}^{j,xy}\} &= \{\langle T\psi_{j,k} \psi_{j,l}, \varphi_{j,k'} \varphi_{j,l'} \rangle\}\end{aligned}$$

($2^{J-j} \times 2^{J-j} \times 2^{J-j} \times 2^{J-j}$ tensoren) en

$$\begin{aligned}\widehat{s}_{jk'l'} &= \sum_{k'=1}^{2^{J-1}} \sum_{l'=1}^{2^{J-1}} \left(\langle T\psi_{j,k} \varphi_{j,l}, \varphi_{j,k'} \varphi_{j,l'} \rangle d_{jkl}^x \right. \\ &\quad \left. + \langle T\varphi_{j,k} \psi_{j,l}, \varphi_{j,k'} \varphi_{j,l'} \rangle d_{jkl}^y \right. \\ &\quad \left. + \langle T\psi_{j,k} \psi_{j,l}, \varphi_{j,k'} \varphi_{j,l'} \rangle d_{jkl}^{xy} \right)\end{aligned}$$

krijgen we

$$P_j^{(2)} T Q_j^{(2)} f = \sum_{k'=1}^{2^{J-1}} \sum_{l'=1}^{2^{J-1}} \hat{s}_{jk'l'} \varphi_{j,k'} \varphi_{j,l'} \quad (4.52)$$

- De laatste term van (4.49), $T_N f = P_N^{(2)} T P_N^{(2)} f$, is de eenvoudigste:

$$P_N^{(2)} T P_N^{(2)} f = \sum_{k'=1}^{2^{J-1}} \sum_{l'=1}^{2^{J-1}} \sum_{k=1}^{2^{J-1}} \sum_{l=1}^{2^{J-1}} s_{Nkl} \langle T \varphi_{N,k} \varphi_{N,l}, \varphi_{N,k'} \varphi_{N,l'} \rangle \varphi_{N,k'} \varphi_{N,l'}$$

Met

$$\{t_{k'l'kl}\} = \{ \langle T \varphi_{N,k} \varphi_{N,l}, \varphi_{N,k'} \varphi_{N,l'} \rangle \}$$

($2^{J-j} \times 2^{J-j} \times 2^{J-j} \times 2^{J-j}$ tensor) en

$$\hat{s}_{k'l'} = \sum_{k=1}^{2^{J-1}} \sum_{l=1}^{2^{J-1}} \langle T \varphi_{N,k} \varphi_{N,l}, \varphi_{N,k'} \varphi_{N,l'} \rangle s_{Nkl}$$

wordt dit

$$P_N^{(2)} T P_N^{(2)} f = \sum_{k'=1}^{2^{J-1}} \sum_{l'=1}^{2^{J-1}} \hat{s}_{k'l'} \varphi_{N,k'} \varphi_{N,l'} \quad (4.53)$$

(4.51), (4.52) en (4.53) in (4.49) substitueren geeft de gewilde non-standaard vorm:

$$\begin{aligned} T_0 f &= \sum_{j=1}^N \sum_{k'=1}^{2^{J-1}} \sum_{l'=1}^{2^{J-1}} \left(\hat{d}_{jk'l'}^x \psi_{j,k'} \varphi_{j,l'} + \hat{d}_{jk'l'}^y \varphi_{j,k'} \psi_{j,l'} + \hat{d}_{jk'l'}^{xy} \psi_{j,k'} \psi_{j,l'} \right. \\ &\quad \left. + \hat{s}_{jk'l'} \varphi_{j,k'} \varphi_{j,l'} \right) + \sum_{k'=1}^{2^{J-1}} \sum_{l'=1}^{2^{J-1}} \hat{s}_{k'l'} \varphi_{N,k'} \varphi_{N,l'} \end{aligned}$$

Om in wavelet basis te blijven werken moeten we nu een herprojectie uitvoeren. Dat is volledig analoog aan het ééndimensionale geval (cf. 4.2.4). Het enige waar men moet op letten is dat de projectie op de ruimten W_j^2 nu telkens drie sets coëfficiënten oplevert in plaats van één, want de basis van W_j^2 wordt gevormd door drie verschillende sets basisfuncties (cf. 4.1.2).

4.3.2 Standaard vorm operator voor tweedimensionale functies

Bij het opstellen van standaard vorm operatoren voor tweedimensionale functies kunnen we de handige notatievorm met de projectieoperatoren P_j en Q_j niet langer gebruiken. Deze veronderstellen immers dat we werken met een multiresolutie analyse bestaande uit

de ruimtes V_j en W_j ($j = 1 \dots N$). Daar we nu in de tweedimensionale standaard tensor-product basis werken, hebben we geen MRA en dus geen ruimtes V_j en W_j . We zullen enkel nog de ruimte V_0 met zijn projectieoperator P_0 beschouwen. De benadering van een continue tweedimensionale functie $f(x, y)$ die we verkrijgen door discretisatie noemen we $P_0 f$.

We zullen beginnen met het ontbinden van $P_0 f$ in de tweedimensionale standaard tensor-product waveletbasis:

$$\begin{aligned}
P_0 f(x, y) &= \sum_{j_1=1}^N \sum_{j_2=1}^N \sum_{k=1}^{2^{J-j_1}} \sum_{l=1}^{2^{J-j_2}} \langle f(x, y), \psi_{j_1 k}(x) \psi_{j_2 l}(y) \rangle \psi_{j_1 k}(x) \psi_{j_2 l}(y) \\
&+ \sum_{j=1}^N \sum_{k=1}^{2^{J-j}} \sum_{l=1}^{2^{J-N}} \langle f(x, y), \psi_{jk}(x) \varphi_{Nl}(y) \rangle \psi_{jk}(x) \varphi_{Nl}(y) \\
&+ \sum_{j=1}^N \sum_{k=1}^{2^{J-N}} \sum_{l=1}^{2^{J-j}} \langle f(x, y), \varphi_{Nk}(x) \psi_{jl}(y) \rangle \varphi_{Nk}(x) \psi_{jl}(y) \\
&+ \sum_{k=1}^{2^{J-N}} \sum_{l=1}^{2^{J-N}} \langle f(x, y), \varphi_{Nk}(x) \varphi_{Nl}(y) \rangle \varphi_{Nk}(x) \varphi_{Nl}(y)
\end{aligned}$$

Stel nu:

$$\langle f(x, y), \psi_{j_1 k}(x) \psi_{j_2 l}(y) \rangle = d_{j_1 j_2 k l}^{\psi \psi} \quad (4.54)$$

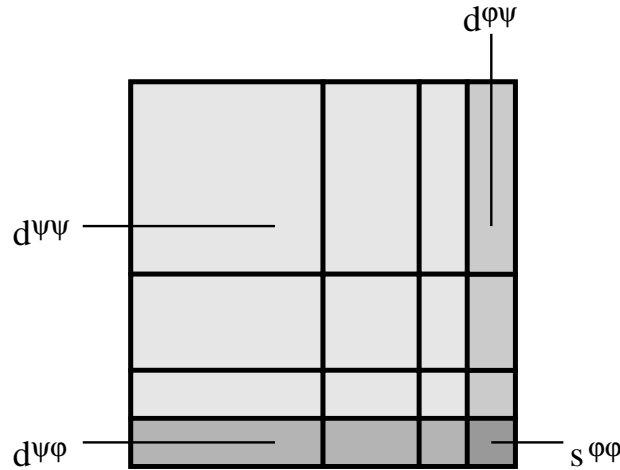
$$\langle f(x, y), \psi_{jk}(x) \varphi_{Nl}(y) \rangle = d_{j k l}^{\psi \varphi} \quad (4.55)$$

$$\langle f(x, y), \varphi_{Nk}(x) \psi_{jl}(y) \rangle = d_{j k l}^{\varphi \psi} \quad (4.56)$$

$$\langle f(x, y), \varphi_{Nk}(x) \varphi_{Nl}(y) \rangle = s_{k l}^{\varphi \varphi} \quad (4.57)$$

In figuur 4.3 is de organisatie van deze coëfficiënten in de standaard vorm matrix te zien. (4.54)-(4.57) substitueren in de uitdrukking voor $P_0 f(x, y)$ en operator T toepassen geeft

$$\begin{aligned}
T P_0 f(x, y) &= \sum_{j_1=1}^N \sum_{j_2=1}^N \sum_{k=1}^{2^{J-j_1}} \sum_{l=1}^{2^{J-j_2}} d_{j_1 j_2 k l}^{\psi \psi} T \psi_{j_1 k}(x) \psi_{j_2 l}(y) \\
&+ \sum_{j=1}^N \sum_{k=1}^{2^{J-j}} \sum_{l=1}^{2^{J-N}} d_{j k l}^{\psi \varphi} T \psi_{jk}(x) \varphi_{Nl}(y) \\
&+ \sum_{j=1}^N \sum_{k=1}^{2^{J-N}} \sum_{l=1}^{2^{J-j}} d_{j k l}^{\varphi \psi} T \varphi_{Nk}(x) \psi_{jl}(y) \\
&+ \sum_{k=1}^{2^{J-N}} \sum_{l=1}^{2^{J-N}} s_{k l}^{\varphi \varphi} T \varphi_{Nk}(x) \varphi_{Nl}(y)
\end{aligned} \quad (4.58)$$



Figuur 4.3: Organisatie van de coëfficiënten $d_{j_1 j_2 k l}^{\psi \psi}$, $d_{j k l}^{\psi \phi}$, $d_{j k l}^{\phi \psi}$ en $s_{k l}^{\phi \phi}$ in de standaard vorm matrix.

In het algemeen zal de verzameling functies

$$\begin{aligned} & \left\{ T \psi_{j_1 k}(x) \psi_{j_2 l}(y) \right\}_{\substack{j_1, j_2 = 1 \dots N \\ k = 1 \dots 2^{J-j_1} \\ l = 1 \dots 2^{J-j_2}}} \cup \left\{ T \psi_{j k}(x) \varphi_{N l}(y) \right\}_{\substack{j = 1 \dots N \\ k = 1 \dots 2^{J-j} \\ l = 1 \dots 2^{J-N}}} \\ & \cup \left\{ T \varphi_{N k}(x) \psi_{j l}(y) \right\}_{\substack{j = 1 \dots N \\ k = 1 \dots 2^{J-N} \\ l = 1 \dots 2^{J-j}}} \cup \left\{ T \varphi_{N k}(x) \varphi_{N l}(y) \right\}_{\substack{k = 1 \dots 2^{J-N} \\ l = 1 \dots 2^{J-N}}} \end{aligned}$$

geen orthonormale basis meer vormen voor V_0 . Om in waveletbasis te blijven werken moeten we terug projecteren:

$$\begin{aligned} P_0 T P_0 f &= \sum_{j'_1=1}^N \sum_{j'_2=1}^N \sum_{k'=1}^{2^{J-j'_1}} \sum_{l'=1}^{2^{J-j'_2}} \left\langle T P_0 f, \psi_{j'_1 k'}(x) \psi_{j'_2 l'}(y) \right\rangle \psi_{j'_1 k'}(x) \psi_{j'_2 l'}(y) \\ &+ \sum_{j'=1}^N \sum_{k'=1}^{2^{J-j'}} \sum_{l'=1}^{2^{J-N}} \left\langle T P_0 f, \psi_{j' k'}(x) \varphi_{N l'}(y) \right\rangle \psi_{j' k'}(x) \varphi_{N l'}(y) \\ &+ \sum_{j'=1}^N \sum_{k'=1}^{2^{J-N}} \sum_{l'=1}^{2^{J-j'}} \left\langle T P_0 f, \varphi_{N k'}(x) \psi_{j' l'}(y) \right\rangle \varphi_{N k'}(x) \psi_{j' l'}(y) \\ &+ \sum_{k'=1}^{2^{J-N}} \sum_{l'=1}^{2^{J-N}} \left\langle T P_0 f, \varphi_{N k'}(x) \varphi_{N l'}(y) \right\rangle \varphi_{N k'}(x) \varphi_{N l'}(y) \end{aligned}$$

Om de notatie te verlichten zullen we de x - en y -afhankelijkheid van de basisfuncties niet meer noteren. We gebruiken dezelfde conventie als bij de non-standaard vorm. $\varphi_{j,k}(x) \psi_{j,l}(y)$

bv. zullen we dus in het vervolg noteren als $\varphi_{j,k}\psi_{j,l}$. (4.58) substitueren in de uitdrukking voor P_0TP_0f geeft:

$$\begin{aligned}
P_0 T P_0 f &= \sum_{j'_1=1}^N \sum_{j'_2=1}^N \sum_{k'=1}^{2^{J-j'_1} 2^{J-j'_2}} \sum_{l'=1}^{2^{J-j'_1} 2^{J-j'_2}} \left[\sum_{j_1=1}^N \sum_{j_2=1}^N \sum_{k=1}^{2^{J-j_1} 2^{J-j_2}} \sum_{l=1}^{2^{J-j_1} 2^{J-j_2}} d_{j_1 j_2 k l}^{\psi \psi} \langle T \psi_{j_1 k} \psi_{j_2 l}, \psi_{j'_1 k'} \psi_{j'_2 l'} \rangle \right. \\
&+ \sum_{j=1}^N \sum_{k=1}^{2^{J-j}} \sum_{l=1}^{2^{J-N}} d_{j k l}^{\psi \varphi} \langle T \psi_{j k} \varphi_{N l}, \psi_{j'_1 k'} \psi_{j'_2 l'} \rangle \\
&+ \sum_{j=1}^N \sum_{k=1}^{2^{J-N}} \sum_{l=1}^{2^{J-j}} d_{j k l}^{\varphi \psi} \langle T \varphi_{N k} \psi_{j l}, \psi_{j'_1 k'} \psi_{j'_2 l'} \rangle \\
&+ \left. \sum_{k=1}^{2^{J-N}} \sum_{l=1}^{2^{J-N}} s_{k l}^{\varphi \varphi} \langle T \varphi_{N k} \varphi_{N l}, \psi_{j'_1 k'} \psi_{j'_2 l'} \rangle \right] \psi_{j'_1 k'} \psi_{j'_2 l'} \\
&+ \sum_{j'=1}^N \sum_{k'=1}^{2^{J-j'}} \sum_{l'=1}^{2^{J-N}} \left[\sum_{j_1=1}^N \sum_{j_2=1}^N \sum_{k=1}^{2^{J-j_1} 2^{J-j_2}} \sum_{l=1}^{2^{J-j_1} 2^{J-j_2}} d_{j_1 j_2 k l}^{\psi \psi} \langle T \psi_{j_1 k} \psi_{j_2 l}, \psi_{j' k'} \varphi_{N l'} \rangle \right. \\
&+ \sum_{j=1}^N \sum_{k=1}^{2^{J-j}} \sum_{l=1}^{2^{J-N}} d_{j k l}^{\psi \varphi} \langle T \psi_{j k} \varphi_{N l}, \psi_{j' k'} \varphi_{N l'} \rangle \\
&+ \sum_{j=1}^N \sum_{k=1}^{2^{J-N}} \sum_{l=1}^{2^{J-j}} d_{j k l}^{\varphi \psi} \langle T \varphi_{N k} \psi_{j l}, \psi_{j' k'} \varphi_{N l'} \rangle \\
&+ \left. \sum_{k=1}^{2^{J-N}} \sum_{l=1}^{2^{J-N}} s_{k l}^{\varphi \varphi} \langle T \varphi_{N k} \varphi_{N l}, \psi_{j' k'} \varphi_{N l'} \rangle \right] \psi_{j' k'} \varphi_{N l'} \\
&+ \sum_{j'=1}^N \sum_{k'=1}^{2^{J-N}} \sum_{l'=1}^{2^{J-j'}} \left[\sum_{j_1=1}^N \sum_{j_2=1}^N \sum_{k=1}^{2^{J-j_1} 2^{J-j_2}} \sum_{l=1}^{2^{J-j_1} 2^{J-j_2}} d_{j_1 j_2 k l}^{\psi \psi} \langle T \psi_{j_1 k} \psi_{j_2 l}, \varphi_{N k'} \psi_{j' l'} \rangle \right. \\
&+ \sum_{j=1}^N \sum_{k=1}^{2^{J-j}} \sum_{l=1}^{2^{J-N}} d_{j k l}^{\psi \varphi} \langle T \psi_{j k} \varphi_{N l}, \varphi_{N k'} \psi_{j' l'} \rangle \\
&+ \sum_{j=1}^N \sum_{k=1}^{2^{J-N}} \sum_{l=1}^{2^{J-j}} d_{j k l}^{\varphi \psi} \langle T \varphi_{N k} \psi_{j l}, \varphi_{N k'} \psi_{j' l'} \rangle \\
&+ \left. \sum_{k=1}^{2^{J-N}} \sum_{l=1}^{2^{J-N}} s_{k l}^{\varphi \varphi} \langle T \varphi_{N k} \varphi_{N l}, \varphi_{N k'} \psi_{j' l'} \rangle \right] \varphi_{N k'} \psi_{j' l'} \\
&+ \sum_{k'=1}^{2^{J-N}} \sum_{l'=1}^{2^{J-N}} \left[\sum_{j_1=1}^N \sum_{j_2=1}^N \sum_{k=1}^{2^{J-j_1} 2^{J-j_2}} \sum_{l=1}^{2^{J-j_1} 2^{J-j_2}} d_{j_1 j_2 k l}^{\psi \psi} \langle T \psi_{j_1 k} \psi_{j_2 l}, \varphi_{N k'} \varphi_{N l'} \rangle \right. \\
&+ \sum_{j=1}^N \sum_{k=1}^{2^{J-j}} \sum_{l=1}^{2^{J-N}} d_{j k l}^{\psi \varphi} \langle T \psi_{j k} \varphi_{N l}, \varphi_{N k'} \varphi_{N l'} \rangle \\
&+ \sum_{j=1}^N \sum_{k=1}^{2^{J-N}} \sum_{l=1}^{2^{J-j}} d_{j k l}^{\varphi \psi} \langle T \varphi_{N k} \psi_{j l}, \varphi_{N k'} \varphi_{N l'} \rangle \\
&+ \left. \sum_{k=1}^{2^{J-N}} \sum_{l=1}^{2^{J-N}} s_{k l}^{\varphi \varphi} \langle T \varphi_{N k} \varphi_{N l}, \varphi_{N k'} \varphi_{N l'} \rangle \right] \varphi_{N k'} \varphi_{N l'}
\end{aligned} \tag{4.59}$$

Zoals we zien worden alle schalen hierin gemengd. Dit is typisch voor de standaard vorm.

De inprodukten die voorkomen vormen vierdimensionale tensoren: de getallen

$$\left\langle T\psi_{j_1 k}\psi_{j_2 l}, \psi_{j_1' k'}\psi_{j_2' l'} \right\rangle$$

bv., vormen voor vaste j_1, j_2, j_1' en j_2' een vierdimensionale tensor geïndexeerd door (k', k, l', l) . Bij het ééndimensionale geval konden alle matrices gecombineerd worden tot één $2^J \times 2^J$ matrix. Hier kunnen alle tensoren gecombineerd worden tot één $2^J \times 2^J \times 2^J \times 2^J$ tensor. Volledig analoog aan het ééndimensionale geval geldt ook nu dat de originele vierdimensionale operator-tensor in de vierdimensionale standaard tensor-product wavelet basis schrijven dezelfde standaard vorm tensor oplevert.

4.3.3 Separabele operatoren

Standaard vorm

Als de lineaire operator T separabel is, m.a.w. $T = T_x T_y = T_y T_x$, of equivalent, als de discretisatietensor van T kan geschreven worden als het uitwendig produkt van twee matrices, dan geldt:

$$\begin{aligned} & \left\langle T\psi_{j_1 k}(x)\psi_{j_2 l}(y), \psi_{j_1' k'}(x)\psi_{j_2' l'}(y) \right\rangle \\ &= \left\langle T_x\psi_{j_1 k}(x)T_y\psi_{j_2 l}(y), \psi_{j_1' k'}(x)\psi_{j_2' l'}(y) \right\rangle \\ &= \left\langle T_x\psi_{j_1 k}(x), \psi_{j_1' k'}(x) \right\rangle \left\langle T_y\psi_{j_2 l}(y), \psi_{j_2' l'}(y) \right\rangle \end{aligned}$$

Hetzelfde geldt voor alle andere tensorelementen die voorkomen in (4.59). Dit betekent dat we, om T toe te passen, niet de volledige tensor moeten opstellen en gebruiken, maar dat het volstaat om T_x toe te passen volgens de x -richting (op alle rijen) en vervolgens T_y toe te passen volgens de y -richting (op alle kolommen), of omgekeerd. Om dit in te zien bekijken we (4.59). We beginnen met de eerste 'term', waarbij we enkele indices van naam

veranderd hebben:

$$\begin{aligned}
\text{Term 1} &= \sum_{j'_1=1}^N \sum_{j'_2=1}^N \sum_{k'=1}^{2^{J-j'_1}} \sum_{l'=1}^{2^{J-j'_2}} \left[\sum_{j_1=1}^N \sum_{j_2=1}^N \sum_{k=1}^{2^{J-j_1}} \sum_{l=1}^{2^{J-j_2}} d_{j_1 j_2 k l}^{\psi\psi} \langle T_x \psi_{j_1 k}, \psi_{j'_1 k'} \rangle \langle T_y \psi_{j_2 l}, \psi_{j'_2 l'} \rangle \right. \\
&\quad + \sum_{j=1}^N \sum_{k=1}^{2^{J-j}} \sum_{l=1}^{2^{J-N}} d_{j k l}^{\psi\varphi} \langle T_x \psi_{j k}, \psi_{j'_1 k'} \rangle \langle T_y \varphi_{N l}, \psi_{j'_2 l'} \rangle \\
&\quad + \sum_{j_2=1}^N \sum_{k=1}^{2^{J-N}} \sum_{l=1}^{2^{J-j}} d_{j k l}^{\varphi\psi} \langle T_x \varphi_{N k}, \psi_{j'_1 k'} \rangle \langle T_y \psi_{j_2 l}, \psi_{j'_2 l'} \rangle \\
&\quad \left. + \sum_{k=1}^{2^{J-N}} \sum_{l=1}^{2^{J-N}} s_{k l}^{\varphi\varphi} \langle T_x \varphi_{N k}, \psi_{j'_1 k'} \rangle \langle T_y \varphi_{N l}, \psi_{j'_2 l'} \rangle \right] \psi_{j'_1 k'} \psi_{j'_2 l'} \\
&= \sum_{j'_1=1}^N \sum_{j'_2=1}^N \sum_{k'=1}^{2^{J-j'_1}} \sum_{l'=1}^{2^{J-j'_2}} \left[\sum_{j_2=1}^N \sum_{l=1}^{2^{J-j_2}} \langle T_y \psi_{j_2 l}, \psi_{j'_2 l'} \rangle \cdot \right. \\
&\quad \left(\sum_{j_1=1}^N \sum_{k=1}^{2^{J-j_1}} d_{j_1 j_2 k l}^{\psi\psi} \langle T_x \psi_{j_1 k}, \psi_{j'_1 k'} \rangle + \sum_{k=1}^{2^{J-N}} d_{j k l}^{\varphi\psi} \langle T_x \varphi_{N k}, \psi_{j'_1 k'} \rangle \right) \\
&\quad + \sum_{l=1}^{2^{J-N}} \langle T_y \varphi_{N l}, \psi_{j'_2 l'} \rangle \cdot \left(\sum_{j=1}^N \sum_{k=1}^{2^{J-j}} d_{j k l}^{\psi\varphi} \langle T_x \psi_{j k}, \psi_{j'_1 k'} \rangle \right. \\
&\quad \left. + \sum_{k=1}^{2^{J-N}} s_{k l}^{\varphi\varphi} \langle T_x \varphi_{N k}, \psi_{j'_1 k'} \rangle \right) \left. \right] \psi_{j'_1 k'} \psi_{j'_2 l'}
\end{aligned}$$

Noemen we

$$\begin{aligned}
\widehat{d}_{j'_1 j'_2 k' l'}^{x, \psi\psi, \psi\varphi} &= \sum_{j_1=1}^N \sum_{k=1}^{2^{J-j_1}} d_{j_1 j_2 k l}^{\psi\psi} \langle T_x \psi_{j_1 k}, \psi_{j'_1 k'} \rangle + \sum_{k=1}^{2^{J-N}} d_{j k l}^{\varphi\psi} \langle T_x \varphi_{N k}, \psi_{j'_1 k'} \rangle \\
\widehat{d}_{j'_1 k' l'}^{x, \psi\varphi, \varphi\varphi} &= \sum_{j=1}^N \sum_{k=1}^{2^{J-j}} d_{j k l}^{\psi\varphi} \langle T_x \psi_{j k}, \psi_{j'_1 k'} \rangle + \sum_{k=1}^{2^{J-N}} s_{k l}^{\varphi\varphi} \langle T_x \varphi_{N k}, \psi_{j'_1 k'} \rangle
\end{aligned}$$

dan wordt dit

$$\begin{aligned}
\text{Term 1} &= \sum_{j'_1=1}^N \sum_{j'_2=1}^N \sum_{k'=1}^{2^{J-j'_1}} \sum_{l'=1}^{2^{J-j'_2}} \left[\sum_{j_2=1}^N \sum_{l=1}^{2^{J-j_2}} \widehat{d}_{j'_1 j'_2 k' l'}^{x, \psi\psi, \psi\varphi} \langle T_y \psi_{j_2 l}, \psi_{j'_2 l'} \rangle \right. \\
&\quad \left. + \sum_{l=1}^{2^{J-N}} \widehat{d}_{j'_1 k' l'}^{x, \psi\varphi, \varphi\varphi} \langle T_y \varphi_{N l}, \psi_{j'_2 l'} \rangle \right] \psi_{j'_1 k'} \psi_{j'_2 l'}
\end{aligned}$$

en met

$$\widetilde{d}_{j'_1 j'_2 k' l'}^{\psi\psi} = \sum_{j_2=1}^N \sum_{l=1}^{2^{J-j_2}} \widehat{d}_{j'_1 j'_2 k' l'}^{x, \psi\psi, \psi\varphi} \langle T_y \psi_{j_2 l}, \psi_{j'_2 l'} \rangle + \sum_{l=1}^{2^{J-N}} \widehat{d}_{j'_1 k' l'}^{x, \psi\varphi, \varphi\varphi} \langle T_y \varphi_{N l}, \psi_{j'_2 l'} \rangle$$

krijgen we

$$\text{Term 1} = \sum_{j'_1=1}^N \sum_{j'_2=1}^N \sum_{k'=1}^{2^{J-j'_1}} \sum_{l'=1}^{2^{J-j'_2}} \tilde{d}_{j'_1 j'_2 k' l'}^{\psi\psi} \psi_{j'_1 k'} \psi_{j'_2 l'}$$

Deze uitdrukking is degene die we zouden verwachten voor een tweedimensionale standaard tensor-produkt wavelet ontbinding. Door vergelijking met (4.6) zien we dat de getallen $\hat{d}_{j'_1 j'_2 k' l'}^{x, \psi\psi, \psi\varphi}$, met $j'_1 = 1 \dots N$, $k' = 1 \dots 2^{J-j'_1}$ en j_2 en l vast, de waveletcoëfficiënten zijn die we krijgen door de standaard vorm van operator T_x toe te passen op rij $\sum_{i=1}^{j_2-1} 2^{J-i} + l$ van de discretisatiematrix van de functie f (zie figuur 4.3). De getallen $\hat{d}_{j'_1 k' l'}^{x, \psi\varphi, \varphi\varphi}$, met l vast en de waarden voor de andere indices hetzelfde als voor de $\hat{d}_{j'_1 j'_2 k' l'}^{x, \psi\psi, \psi\varphi}$, verkrijgen we door T_x toe te passen op rij $2^J - 2^{J-N} + l$ van de discretisatiematrix van f . $\tilde{d}_{j'_1 j'_2 k' l'}^{\psi\psi}$, met $j'_2 = 1 \dots N$, $l' = 1 \dots 2^{J-j'_1}$ en j'_1 en l' vast, zijn de waveletcoëfficiënten die we bekommen door, na toepassing van T_x op alle rijen, T_y toe te passen op kolom $\sum_{i=1}^{j'_1-1} 2^{J-i} + l'$.

De andere termen van (4.59) zullen, volledig analoog, de coëfficiënten $d_{j'_1 k' l'}^{\psi\varphi}$, $d_{j'_1 k' l'}^{\varphi\psi}$ en $s_{k' l'}^{\varphi\varphi}$ opleveren. We zien dus dat, voor separabele operatoren $T = T_x T_y = T_y T_x$, toepassen van de standaard vorm operator T_x op alle rijen en vervolgens T_y (ook standaard vorm) op alle kolommen van het resultaat equivalent is aan het toepassen van T (standaard vorm, bedoeld voor tweedimensionale functies). Het omgekeerde (eerst T_y toepassen, dan T_x) kan volledig analoog aangetoond worden.

Non-standaard vorm

Voor de non-standaard vorm schijnt er geen eenvoudige manier te zijn om het toepassen van een separabele operator op te splitsen in twee stappen, zoals we dat bij de standaard vorm hebben gedaan. De voornaamste moeilijkheden zijn de benodigde herprojectie en het feit dat bij de waveletontbinding van de tweedimensionale functie de schaalcoëfficiënten op elke schaal moeten bijgehouden worden en niet enkel die op de grofste schaal zoals gebruikelijk is. De waveletontbinding past dus niet in een matrix van dezelfde grootte als de originele discretisatiematrix.

Het enige aspect van de non-standaard vorm dat baat heeft bij het separabel zijn van een operator is het berekenen van de verschillende tensoren. Als de operator T separabel is ($T = T_x T_y = T_y T_x$) kunnen de tensoren die voorkomen in de non-standaard vorm van T nl. berekend worden als uitwendige produkten van de matrices die voorkomen in de non-standaard vorm van T_x en T_y . Het enige waar we op moeten letten is dat we de matrices T_j^i (zie (4.22)) bijhouden voor elke schaal j bij het opstellen van de non-standaard vormen van T_x en T_y . Normaal houden we enkel de matrix T_N^i bij, met N de grofste schaal. Bekijken

we bv. de tensor $\{a_{k'l'l}^{j,x,x}\}$ (zie (4.50)):

$$\begin{aligned}
\{a_{k'l'l}^{j,x,x}\} &= \{\langle T\psi_{j,k}(x)\varphi_{j,l}(y), \psi_{j,k'}(x)\varphi_{j,l'}(y) \rangle\} \\
&= \{\langle T_x\psi_{j,k}(x)T_y\varphi_{j,l}(y), \psi_{j,k'}(x)\varphi_{j,l'}(y) \rangle\} \\
&= \{\langle T_x\psi_{j,k}(x), \psi_{j,k'}(x) \rangle \langle T_y\varphi_{j,l}(y), \varphi_{j,l'}(y) \rangle\} \\
&= \{\langle T_x\psi_{j,k}(x), \psi_{j,k'}(x) \rangle\} \cdot \{\langle T_y\varphi_{j,l}(y), \varphi_{j,l'}(y) \rangle\} \\
&= \{a_{k'l}^{j,T_x}\} \cdot \{t_{l'l}^{j,T_y}\} \\
&= A'_{j,T_x} \cdot T'_{j,T_y}
\end{aligned}$$

Met \cdot bedoelen we hier het uitwendig produkt. In de laatste twee stappen werd gebruik gemaakt van de definities (4.17) en (4.22). De redenering voor de andere tensoren die voorkomen in de non-standaard vorm van T is volledig analoog.

4.3.4 Tensor-matrix en tensor-tensor produkten

We hebben gezien dat voor het toepassen van operatoren op tweedimensionale functies tensor-matrix produkten moeten berekend worden. Analoog moeten bij hogerdimensionale functies tensor-tensor produkten uitgerekend worden, waarbij het aantal dimensies van de operatortensor steeds dubbel zo groot is als het aantal dimensies (variabelen) van de functie.

Het rechtstreeks bereken van deze produkten vergt enorm veel bewerkingen: complexiteit $O(N^{2n})$ met n het aantal variabelen waarvan de functie afhangt. Vaak zal de operatortensor echter ijl zijn, zodat het uitrekenen van de produkten enorm versneld kan worden. Als de operatortensor niet ijl is, is hij dat in waveletbasis misschien wel (bv. bij integraaloperatoren). Een probleem bij de implementatie is echter dat functiebibliotheken voor bewerkingen met ijle tensoren meestal niet voorhanden zijn, in tegenstelling tot die voor ijle matrixbewerkingen. We kunnen de ijle matrixfuncties echter ook gebruiken om de bewerkingen met tensoren te versnellen. Dat gaat als volgt:

Beschouw bv. het produkt $\overline{\overline{C}} = \overline{\overline{A}}\overline{\overline{B}}$, met $\overline{\overline{B}}$ en $\overline{\overline{C}}$ $N \times N$ matrices en $\overline{\overline{A}}$ een $N \times N \times N \times N$ tensor, gegeven door

$$c_{i,j} = \sum_{k=1}^N \sum_{l=1}^N a_{i,k,j,l} b_{kl}$$

Voor vaste i en j kunnen we dit schrijven als

$$c_{i,j} = \sum_{k=1}^N \sum_{l=1}^N \tilde{a}_{k,l}^{i,j} b_{kl} \quad (4.60)$$

met $\tilde{a}_{k,l}^{i,j} = a_{i,k,j,l}$ voor de gekozen vaste waarden van i en j . De $\tilde{a}_{k,l}^{i,j}$ vormen een $N \times N$ matrix, $\overline{\overline{A}}^{i,j} = \{\tilde{a}_{k,l}^{i,j}\}$.

Beschouw nu twee matrices $\overline{\overline{D}}$ en $\overline{\overline{E}}$, waarvan het produkt $\overline{\overline{F}} = \overline{\overline{D}}\overline{\overline{E}}$ gegeven wordt door

$$f_{i,j} = \sum_l d_{il}e_{lj}$$

Als we deze formule vergelijken met (4.60) zien we dat (4.60) gelijk is aan het spoor (= som van de diagonaalelementen) van de matrix $\overline{\overline{G}} = \overline{\overline{A}}\overline{\overline{B}}^T$, waarbij $\overline{\overline{B}}^T$ de getransponeerde van $\overline{\overline{B}}$ is. Immers, als we de elementen van $\overline{\overline{B}}^T$ definiëren als $b_{k,l}^T = b_{l,k}$ dan geldt voor (4.60):

$$\begin{aligned} c_{i,j} &= \sum_{k=1}^N \sum_{l=1}^N \tilde{a}_{k,l}^{i,j} b_{kl} \\ &= \sum_{k=1}^N \sum_{l=1}^N \tilde{a}_{k,l}^{i,j} b_{lk}^T \\ &= \text{Spoor} \left(\overline{\overline{A}}^{i,j} \overline{\overline{B}}^T \right) \end{aligned} \quad (4.61)$$

Het produkt $\overline{\overline{A}}^{i,j} \overline{\overline{B}}^T$ kan met de snelle routines voor het vermenigvuldigen van ijle matrices berekend worden. Men zou evengoed het spoor van $\overline{\overline{A}}^{i,j} \overline{\overline{B}}$ kunnen nemen, maar dan zouden we voor alle i,j de matrix $\overline{\overline{A}}^{i,j}$ moeten transponeren. Dat is minder efficiënt dan het eenmalig transponeren van $\overline{\overline{B}}$.

4.4 Toepassing: vermenigvuldigen in waveletbasis

Het probleem

Stel dat we een numeriek algoritme hebben waarin we een een functie f benaderen door iteratief de waveletexpansie ervan te berekenen. We beginnen met startwaarde f_0 , berekenen de waveletontwikkeling van f , starten daarmee het algoritme en itereren tot een bepaalde voorwaarde voldaan is. Stel dat we tijdens elke iteratiestap het produkt $f \cdot g$ moeten berekenen, waarbij we g kennen in natuurlijke basis. Op het eerste zicht zijn er twee mogelijkheden om dit produkt te berekenen:

1. We bereken f in natuurlijke basis (door inverse wavelettransformatie), nemen het produkt met g en transformeren het resultaat terug naar waveletbasis. Dit is niet ideaal, omdat we in elke iteratiestap een voorwaartse en inverse wavelettransformatie moeten berekenen. Zeker als f en g functies van meerdere variabelen zijn kan dit op gebied van rekentijd erg duur zijn.

2. We berekenen één keer de waveletvoorstelling van g en berekenen het produkt $f \cdot g$ rechtstreeks in waveletbasis.

Deze tweede mogelijkheid zullen we nader bekijken voor functies in één veranderlijke.

Stel dat we twee functies $f(x)$ en $g(x)$ in waveletbasis willen vermenigvuldigen. We beschikken over de waveletontwikkelingen

$$\begin{aligned} f &= \sum_{j=1}^N \sum_{k=1}^{2^{J-j}} d_{jk}^f \psi_{j,k} + \sum_{k=1}^{2^{J-N}} s_{Nk}^f \varphi_{N,k} \\ g &= \sum_{j=1}^N \sum_{k=1}^{2^{J-j}} d_{jk}^g \psi_{j,k} + \sum_{k=1}^{2^{J-N}} s_{Nk}^g \varphi_{N,k} \end{aligned}$$

Het produkt is $f \cdot g$ is

$$\begin{aligned} f \cdot g &= \left(\sum_{j=1}^N \sum_{k=1}^{2^{J-j}} d_{jk}^f \psi_{j,k} + \sum_{k=1}^{2^{J-N}} s_{Nk}^f \varphi_{N,k} \right) \left(\sum_{j'=1}^N \sum_{k'=1}^{2^{J-j'}} d_{j'k'}^g \psi_{j',k'} + \sum_{k'=1}^{2^{J-N}} s_{N'k'}^g \varphi_{N',k'} \right) \\ &= \sum_{j=1}^N \sum_{k=1}^{2^{J-j}} \sum_{j'=1}^N \sum_{k'=1}^{2^{J-j'}} d_{jk}^f d_{j'k'}^g \psi_{j,k} \psi_{j',k'} + \sum_{j=1}^N \sum_{k=1}^{2^{J-j}} \sum_{k'=1}^{2^{J-N}} d_{jk}^f s_{N'k'}^g \psi_{j,k} \varphi_{N',k'} \\ &\quad + \sum_{k=1}^{2^{J-N}} \sum_{j'=1}^N \sum_{k'=1}^{2^{J-j'}} s_{Nk}^f d_{j'k'}^g \varphi_{N,k} \psi_{j',k'} + \sum_{k=1}^{2^{J-N}} \sum_{k'=1}^{2^{J-N}} s_{Nk}^f s_{N'k'}^g \varphi_{N,k} \varphi_{N',k'} \end{aligned}$$

Om in waveletbasis te blijven moeten we herprojecteren. Hiervoor moeten de getallen

$$\begin{aligned} &\langle \psi_{j,k} \psi_{j',k'}, \psi_{j'',k''} \rangle && \langle \psi_{j,k} \psi_{j',k'}, \varphi_{N'',k''} \rangle \\ &\langle \psi_{j,k} \varphi_{N',k'}, \psi_{j'',k''} \rangle && \langle \psi_{j,k} \varphi_{N',k'}, \varphi_{N'',k''} \rangle \\ &\langle \varphi_{N,k} \psi_{j',k'}, \psi_{j'',k''} \rangle && \langle \varphi_{N,k} \psi_{j',k'}, \varphi_{N'',k''} \rangle \\ &\langle \varphi_{N,k} \varphi_{N',k'}, \psi_{j'',k''} \rangle && \langle \varphi_{N,k} \varphi_{N',k'}, \varphi_{N'',k''} \rangle \end{aligned}$$

gekend zijn, met $j, j', j'' = 1 \dots N$, $k = 2^{J-j}$, $k' = 2^{J-j'}$ en $k'' = 2^{J-j''}$. Meestal is dit een gigantisch aantal getallen. Als we bv. slechts één maal ontbinden in wavelet- en detailcoëfficiënten, m.a.w. $N = 1$, en als $2^J = 64$ zijn er al meer dan 16 miljoen getallen $\langle \psi_{j,k} \psi_{j',k'}, \psi_{j'',k''} \rangle$. In meerdere dimensies is het aantal nodige getallen nog veel groter. Zelfs als we alle getallen zouden kennen, is de rekentijd nodig om het produkt te berekenen enorm.

Het is duidelijk dat rechtstreeks vermenigvuldigen in waveletbasis geen goed idee is. Een snel algoritme wordt beschreven in [27], maar deze methode maakt gebruik van de multiresolutie-eigenschappen van de wavelettransformatie. In meerdere dimensies is ze dus enkel toepasbaar op wavelet-ontbindingen gebaseerd op multiresolutie analyses. Dit houdt in dat we de toepassing van separabele operatoren niet kunnen splitsen in verschillende stappen, zoals beschreven op pagina 95.

Oplossing door middel van operatoren

Stel dat we twee functies f en g , gekend onder de vorm van hun samples f_i en g_i ($i = 1 \dots 2^J$), willen vermenigvuldigen. De samples van het produkt $f \cdot g$ kunnen vanzelfsprekend bekomen worden door rechtstreekse vermenigvuldiging van de overeenkomstige samples van f en g , nl. $(f \cdot g)_i = f_i g_i$. We kunnen de $(f \cdot g)_i$ echter ook berekenen met een matrix-vector produkt:

$$\begin{bmatrix} g_1 & & & \\ & g_2 & & \\ & & \ddots & \\ & & & g_{2^J} \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_{2^J} \end{bmatrix} = \begin{bmatrix} (f \cdot g)_1 \\ (f \cdot g)_2 \\ \vdots \\ (f \cdot g)_{2^J} \end{bmatrix}$$

De matrix kan beschouwd worden als een (ijle) operator G die we toepassen op f . Als we nu zowel G als f uitdrukken in waveletbasis, zal het produkt ook in waveletbasis staan. Dit werkt ook in meerdere dimensies, zowel bij gebruik van de standaard- als de non-standaard vorm.

Als f en g functies zijn van n variabelen, zal G een $2n$ -dimensionale tensor zijn. Met $G = \{\tilde{g}_{i_1, j_1, i_2, j_2, \dots, i_n, j_n}\}$ wordt het tensor-tensorprodukt Gf gegeven door

$$(f \cdot g)_{i_1, i_2, \dots, i_n} = \sum_{j_1} \sum_{j_2} \cdots \sum_{j_n} \tilde{g}_{i_1, j_1, i_2, j_2, \dots, i_n, j_n} f_{j_1, j_2, \dots, j_n}$$

Als we eisen dat $(f \cdot g)_{i_1, i_2, \dots, i_n} = f_{i_1, i_2, \dots, i_n} g_{i_1, i_2, \dots, i_n}$ hebben we dus

$$\tilde{g}_{i_1, j_1, i_2, j_2, \dots, i_n, j_n} = \begin{cases} g_{i_1, i_2, \dots, i_n} & \text{als } (i_1, i_2, \dots, i_n) = (j_1, j_2, \dots, j_n) \\ 0 & \text{in alle andere gevallen} \end{cases}$$

Deze betrekking kunnen we gebruiken om de tensor G op te stellen (in natuurlijke basis).

4.5 Besluit

In dit hoofdstuk werd de voorstelling van lineaire operatoren in wavelet basis besproken. Er zijn twee hoofdredenen waarom dit voordelig kan zijn:

1. Als de discretisatiematrix van de operator dens is zal hij in waveletbasis meestal een stuk ijler zijn. Vermenigvuldigingen van deze matrix met vectoren kunnen dus snel berekend worden.
2. Als de operator in waveletbasis staat moet de functie waarop hij werkt natuurlijk ook in waveletbasis staan. In veel gevallen is de voorstelling van een functie in deze basis veel economischer (veel coëfficiënten die heel klein of nul zijn) dan in de natuurlijke of Fourierbasis. Dit bespaart geheugen en zorgt ervoor dat toepassingen van de operator op de functie snel kunnen berekend worden.

Integraaloperatoren profiteren in het algemeen van beide voordelen, want hun discretisatiematrix is dichter. Differentiaaloperatoren hebben van nature een heel ijle discretisatiematrix, die in waveletbasis zelfs minder ijel wordt. Zij hebben dus enkel baat bij de meer economische voorstelling van functies in waveletbasis. Het tweede voordeel is afhankelijk van het type functie. Voor sterk oscillerende functies bv. is de Fourierbasis soms beter (minder significante coëfficiënten) [18].

Er zijn twee belangrijke voorstellingen van operatoren die werken op functies in waveletbasis: de standaard vorm en de non-standaard vorm. Ze vinden hun oorsprong in de twee verschillende manieren waarop een separabele tweedimensionale waveletbasis kan geconstrueerd worden, nl. aan de hand van een multiresolutie analyse (non-standaard vorm) of niet (standaard vorm). Beide vormen hebben hun voor- en nadelen:

Standaard vorm Pluspunt is dat de standaard vorm matrix echt een voorstelling in waveletbasis is van de operatormatrix. Ook is de samenstelling van twee standaard vorm operatoren gewoon het produkt van de matrices is. Een nadeel is dat de standaard vorm niet zo ijel is, door interactie van de verschillende schalen.

Beylkin [20] bewijst dat toepassing van de standaard vorm van een brede klasse operatoren (Calderón-Zygmund en pseudodifferentiale operatoren) een complexiteit $O(N \log N)$ heeft.

Non-standaard vorm Positief is dat de non-standaard vorm matrix bijzonder ijel is, omdat er geen interacties tussen schalen zijn. Nadelig is dat de matrix niet werkt op een standaard wavelet expansie van functies, maar op een redundante expansie. Na toepassing van de matrix moet er nog een herprojectie plaatsvinden die de koppeling tussen schalen in rekening brengt. Deze herprojectie vereist ongeveer evenveel rekentijd als een fast wavelet transform ($O(N)$ vermenigvuldigingen). Samenstellingen van non-standaard vorm operatoren zijn ook een stuk moeilijker te berekenen dan van de standaard vorm operatoren. De non-standaard vorm matrix is geen voorstelling in waveletbasis, maar een matrixorganisatie die handig is voor de berekeningen.

Beylkin [20] bewijst dat toepassing van de non-standaard vorm van een brede klasse operatoren (Calderón-Zygmund en pseudodifferentiale operatoren) een complexiteit $O(N)$ heeft.

In de literatuur is weinig concrete, toepassingsgerichte informatie over wavelet-operatoren beschikbaar. De beschikbare publicaties zijn bijna allemaal van Beylkin [20, 24, 25, 21, 23]. [24, 25, 21, 23] bouwen allemaal verder op [20], waarin veel details die nodig zijn om tot een praktische implementatie te komen niet vermeld zijn. In dit hoofdstuk hebben we geprobeerd een degelijk overzicht te geven van operatoren in waveletbasis met de bedoeling ze in de praktijk te kunnen toepassen. Zo hebben we bv. het herprojectiealgoritme dat moet toegepast worden na een non-standaard vorm matrix-vectorvermenigvuldiging uitgelegd. Het algoritme zelf is te vinden in [23], doch zonder enige verklaring. Ook hebben we

geprobeerd (zonder succes) om een snellere methode te vinden voor de combinatie van twee non-standaard vorm operatoren.

Over wavelet-gebaseerde operatoren in meerdere dimensies heb ik geen publicaties kunnen vinden. In referentie [28] komen ze voor maar deze bevat geen verdere informatie over hoe deze operatoren opgesteld of gebruikt worden. Daar meerdimensionale operatoren absoluut nodig zijn voor praktische toepassingen hebben we in dit hoofdstuk de meeste details uitgewerkt en zijn we tot een bruikbaar geheel gekomen. Vooral de standaard vorm is interessant gebleken omdat die toelaat scheidbare operatoren in verschillende stappen toe te passen. Dit vereenvoudigt de praktische implementatie aanzienlijk.

In het volgende hoofdstuk zullen we operatoren in waveletbasis gebruiken voor het numeriek oplossen van de vergelijkingen van Maxwell.

Hoofdstuk 5

Numerieke simulaties

In dit hoofdstuk zullen we wavelet-gebaseerde technieken voor elektromagnetische simulaties bespreken. We zullen kort enkele mogelijke methoden bekijken en dan dieper ingaan op één bepaalde methode, geïntroduceerd door Werthen & Wolff [28]. Ze maakt gebruik van de operatoren in waveletbasis die we behandeld hebben in hoofdstuk 4. Van deze methode hebben we een prototype-versie geïmplementeerd in Matlab.

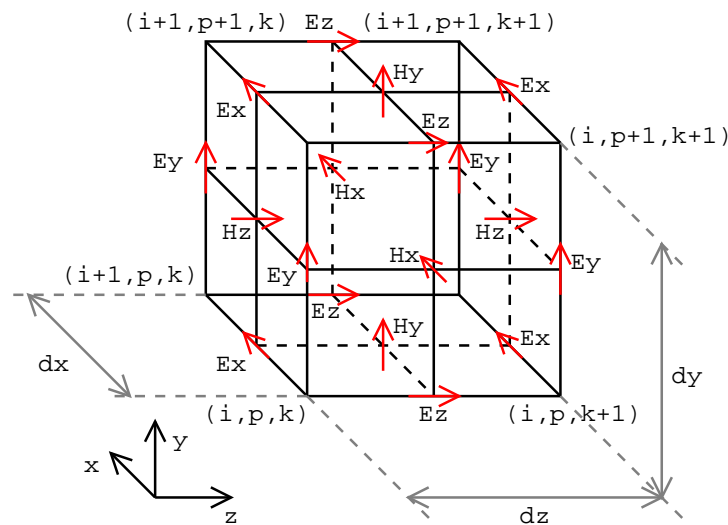
5.1 Wavelet-gebaseerde technieken voor numerieke elektromagnetische berekeningen

5.1.1 Methodes gebaseerd op de Green's functie

Veel elektromagnetische problemen, voornamelijk verstrooiingsproblemen (scattering), kunnen d.m.v. Green's functie-technieken geschreven worden als integraalvergelijkingen. Het discretiseren van deze vergelijkingen geeft aanleiding tot dense matrices (bijna geen elementen die nul of heel klein zijn). Daar bewerkingen met ijle matrices veel sneller zijn (complexiteit $O(N)$ tegenover $O(N^2)$ voor dense matrices) gebruikt men de compressie eigenschappen van wavelets om de dense matrices ijler te maken. Meer informatie kan gevonden worden in o.a. [29].

Eén van de grote voordelen van Green's functie gebaseerde technieken is dat men geen problemen heeft met straling die het simulatiegebied moet kunnen verlaten door middel van bv. 'absorbing boundary conditions'. Bij de Green's functie methoden wordt de waarde van het veld in alle discretisatiepunten nl. rechtstreeks uit de integraalvergelijking berekend, zonder iteraties (merk op: dit betekent niet dat het volledige algoritme niet iteratief is. Meestal wordt de integraalvergelijking nl. iteratief opgelost.).

Een nadeel van technieken zoals die uit [29] is dat voor het opstellen van de discretisatiematrix in waveletbasis meestal een groot aantal numerieke integraties nodig zijn (dit



Figuur 5.1: Yee's discretisatierooster [33]. We zien dat de roosters voor het \mathbf{E} - en \mathbf{H} -veld een half discretisatieinterval ten opzichte van elkaar verschoven zijn in alle coördinaatrichtingen. Ook in de tijd verschuift men \mathbf{E} en \mathbf{H} een halve stap ten opzichte van elkaar. Yee's rooster is het meest gebruikte rooster bij de eindige-verschillen tijdsdomein methode (FDTD).

komt omdat er voor de meeste wavelets geen analytische uitdrukking beschikbaar is). Een mogelijke oplossing hiervoor wordt beschreven in [30] en [31].

Het probleem bij de toepassing van deze methodes op vertical cavity surface emitting lasers (VCSEL's) is het vinden van de Green's functie van de structuur. Een elegante oplossing hiervoor wordt beschreven in [32].

De combinatie van [32] en [30] zou een interessante methode voor het numeriek simuleren van VCSEL's kunnen opleveren. Spijtig genoeg is [32] pas in april 2001 verschenen.

5.1.2 De multiresolutie tijdsdomein methode

Eindige-verschillen tijdsdomein methode

Een veelgebruikte techniek voor het numeriek berekenen van de elektromagnetische veldverdeling in een systeem is de eindige-verschillen tijdsdomein methode (finite-difference time-domain of FDTD methode) [33]. Hierbij gebruikt men meestal het discretisatierooster van Yee (Yee's mesh, zie figuur 5.1) voor het discretiseren van de vergelijkingen van Maxwell. Hoewel de FDTD methode al lang bestaat wordt ze nog steeds actief onderzocht (zie bv. [34, 35, 36]). Een interessante nieuwe manier om de numerieke stabiliteit van FDTD algoritmen te analyseren is te vinden in [37].

Multiresolutie tijdsdomein methode

De MRTD methode is een op multiresolutie analyse gebaseerde techniek die analoog is aan de FDTD methode [38, 39]. Ook de MRTD methode maakt gebruik van Yee's discretisatierooster. Er bestaan twee verschillende versies: de S-MRTD, die enkel met de schalingfuncties φ werkt, en de W-MRTD die ook de waveletfuncties ψ in rekening brengt. In W-MRTD simulaties moet men opletten voor niet-fysische oplossingen (spurious solutions).

Het voornaamste nadeel van de MRTD techniek is het grote aantal numerieke integraties die men moet uitvoeren voordat de simulatie kan gestart worden. Een voordeel is dat de praktische implementatie erg gelijkaardig is aan de implementatie van de standaard FDTD methode.

Meer informatie over MRTD is te vinden in o.a. [38, 39, 40].

5.2 De methode van Werthen & Wolff

De methode die we geïmplementeerd hebben is analoog aan die van Werthen & Wolff [28]. De methode op zich is eenvoudig maar de implementatie met behulp van wavelets zorgt voor specifieke moeilijkheden. Hoofdstuk 4 geeft ons echter de tools die we nodig hebben om de methode in praktijk te brengen.

We beginnen met de rot-vergelijkingen van Maxwell:

$$\begin{aligned}\frac{\partial \mathbf{E}}{\partial t} &= \frac{1}{\epsilon} \text{rot} \mathbf{H} \\ \frac{\partial \mathbf{H}}{\partial t} &= -\frac{1}{\mu} \text{rot} \mathbf{E}\end{aligned}$$

De tijdsafgeleiden hierin worden benaderd door eindige verschillen:

$$\begin{aligned}\frac{\mathbf{E}^n - \mathbf{E}^{n-1}}{\Delta t} &= \frac{1}{\epsilon} \text{rot} \mathbf{H}^{n-\frac{1}{2}} \\ \Downarrow \\ \mathbf{E}^n &= \mathbf{E}^{n-1} + \frac{\Delta t}{\epsilon} \text{rot} \mathbf{H}^{n-\frac{1}{2}}\end{aligned}\tag{5.1}$$

$$\begin{aligned}\frac{\mathbf{H}^{n+\frac{1}{2}} - \mathbf{H}^{n-\frac{1}{2}}}{\Delta t} &= -\frac{1}{\mu} \text{rot} \mathbf{E}^n \\ \Downarrow \\ \mathbf{H}^{n+\frac{1}{2}} &= \mathbf{H}^{n-\frac{1}{2}} - \frac{\Delta t}{\mu} \text{rot} \mathbf{E}^n\end{aligned}\tag{5.2}$$

Hierin zijn de indices bovenaan discrete tijdsindices: \mathbf{E}^n is de vector \mathbf{E} op tijdstip $n\Delta t$ (op eender welke plaats, de ruimte is nog niet gediscretiseerd). De tijdsdiscretisatie van het \mathbf{H} -veld is een halve tijdsstap verschoven ten opzichte van die van het \mathbf{E} -veld, net zoals bij de op Yee's discretisatierooster gebaseerde FDTD methoden en zoals bij de MRTD methode. Dit is nodig om de methode numeriek stabiel te maken.

Het kan handig zijn dat de elektrische en magnetische veldcomponenten dezelfde grootteorde en dimensie hebben (o.a. om numerieke fouten te verminderen). Daarom delen we het elektrisch veld \mathbf{E} door de karakteristieke impedantie van het vacuüm $Z_0 = \sqrt{\frac{\mu_0}{\epsilon_0}}$. Met $\mathbf{E}^n \rightarrow \frac{1}{Z_0}\mathbf{E}^n$ wordt (5.1) dan

$$\begin{aligned}
 \mathbf{E}^n &= \mathbf{E}^{n-1} + \frac{\Delta t}{\epsilon} \cdot \frac{1}{Z_0} \text{rot} \mathbf{H}^{n-\frac{1}{2}} \\
 &= \mathbf{E}^{n-1} + \frac{\Delta t}{\epsilon_r \epsilon_0} \cdot \frac{\sqrt{\epsilon_0}}{\sqrt{\mu_0}} \text{rot} \mathbf{H}^{n-\frac{1}{2}} \\
 &= \mathbf{E}^{n-1} + \frac{\Delta t}{\epsilon_r \sqrt{\epsilon_0 \mu_0}} \text{rot} \mathbf{H}^{n-\frac{1}{2}} \\
 &= \mathbf{E}^{n-1} + \frac{c \Delta t}{\epsilon_r} \text{rot} \mathbf{H}^{n-\frac{1}{2}}
 \end{aligned} \tag{5.3}$$

Hierin zijn $\epsilon = \epsilon_r \epsilon_0$ en $c = 1/\sqrt{\epsilon_0 \mu_0}$. Analoog vinden we voor (5.2)

$$\begin{aligned}
 \mathbf{H}^{n+\frac{1}{2}} &= \mathbf{H}^{n-\frac{1}{2}} - \frac{\Delta t}{\mu} Z_0 \text{rot} \mathbf{E}^n \\
 &= \mathbf{H}^{n-\frac{1}{2}} - \frac{c \Delta t}{\mu_r} \text{rot} \mathbf{E}^n
 \end{aligned} \tag{5.4}$$

waarin $\mu = \mu_r \mu_0$.

In cartesische coördinaten vormen (5.3) en (5.4) na splitsing in componenten het volgende

stelsel:

$$E_x^n = E_x^{n-1} + \frac{c\Delta t}{\epsilon_r} \left(\frac{\partial H_z^{n-\frac{1}{2}}}{\partial y} - \frac{\partial H_y^{n-\frac{1}{2}}}{\partial z} \right) \quad (5.5)$$

$$E_y^n = E_y^{n-1} + \frac{c\Delta t}{\epsilon_r} \left(\frac{\partial H_x^{n-\frac{1}{2}}}{\partial z} - \frac{\partial H_z^{n-\frac{1}{2}}}{\partial x} \right) \quad (5.6)$$

$$E_z^n = E_z^{n-1} + \frac{c\Delta t}{\epsilon_r} \left(\frac{\partial H_y^{n-\frac{1}{2}}}{\partial x} - \frac{\partial H_x^{n-\frac{1}{2}}}{\partial y} \right) \quad (5.7)$$

$$H_x^{n+\frac{1}{2}} = H_x^{n-\frac{1}{2}} - \frac{c\Delta t}{\mu_r} \left(\frac{\partial E_z^n}{\partial y} - \frac{\partial E_y^n}{\partial z} \right) \quad (5.8)$$

$$H_y^{n+\frac{1}{2}} = H_y^{n-\frac{1}{2}} - \frac{c\Delta t}{\mu_r} \left(\frac{\partial E_x^n}{\partial z} - \frac{\partial E_z^n}{\partial x} \right) \quad (5.9)$$

$$H_z^{n+\frac{1}{2}} = H_z^{n-\frac{1}{2}} - \frac{c\Delta t}{\mu_r} \left(\frac{\partial E_y^n}{\partial x} - \frac{\partial E_x^n}{\partial y} \right) \quad (5.10)$$

De bedoeling is nu om vergelijkingen (5.5)-(5.10) in waveletbasis op te lossen. De waveletexpansies van de veldcomponenten zullen nl. veel coëfficiënten bevatten die nul of zeer klein zijn, wat de uitvoeringssnelheid kan verhogen (mits men routines gebruikt die van het ijl zijn van de waveletvoorstellingen gebruik kunnen maken). De ontwikkeling van de veldcomponenten in waveletbasis maakt het ook mogelijk om ‘thresholding’ toe te passen: het nul stellen van heel kleine coëfficiënten. Dit kan tijdens de simulatie regelmatig gedaan worden (bv. steeds na een vast aantal stappen). De thresholding veroorzaakt in het algemeen maar een kleine fout, de winst in rekentijd kan daarentegen aanzienlijk zijn. Het thresholden zorgt er in feite voor dat de methode werkt met een discretisatiestrooster dat tijdens de simulatie lokaal fijner of groffer kan worden, naargelang de grootte van de detailcoëfficiënten. Het rooster past zich bij elke thresholding operatie automatisch aan zodat geen rekentijd verloren gaat met het nauwkeurig berekenen van de velden op plaatsen waar ze bv. nul zijn.

5.2.1 Eén dimensie

Theorie

We zullen beginnen met een ééndimensionale versie van het algoritme. We veronderstellen lineair gepolariseerd licht (met \mathbf{E} parallel met de x -richting) dat zich voortplant in de z -

richting. Vergelijkingen (5.5)-(5.10) worden dus gereduceerd tot

$$E_x^n = E_x^{n-1} - \frac{c\Delta t}{\epsilon_r} \frac{dH_y^{n-\frac{1}{2}}}{dz} \quad (5.11)$$

$$H_y^{n+\frac{1}{2}} = H_y^{n-\frac{1}{2}} - \frac{c\Delta t}{\mu_r} \frac{dE_x^n}{dz} \quad (5.12)$$

Deze formules moeten nog gediscrètiseerd worden in de ruimte (in dit ééndimensionale geval enkel in de z -richting). We stellen

$$E_{D_x}^n = \left\{ E_x^{n,k} = E_x^n|_{z=k\Delta z} \right\}_{k=1\dots 2^J}$$

Met $E_{D_x}^n$ bedoelen we de gediscrètiseerde versie van E_x^n met discretisatieinterval Δz . Anaaloog is $H_{D_y}^{n+\frac{1}{2}}$ de gediscrètiseerde versie van $H_y^{n+\frac{1}{2}}$ (met hetzelfde discretisatieinterval). Het aantal discretisatiepunten is telkens een macht van 2 zodat we de wavelettransformaties kunnen berekenen met het filterbankalgoritme (cf. 3.3.2).

Noem D_z de operator $\frac{d}{dz}$ in waveletbasis (in standaard vorm). Zie sectie 4.2.5 voor meer informatie over operator $\frac{d}{dz}$. D_z is opgesteld in de veronderstelling dat de sample- of discretisatieperiode genormaliseerd is: $\Delta z = 1$. Als $\Delta z \neq 1$ hebben we daarom (met f_W de waveletvoorstelling van f_D en f_D de discretisatie van f)

$$\left(\frac{df}{dz} \right)_W \approx \frac{1}{\Delta z} D_z f_W$$

We schrijven hier \approx omdat de operator $\frac{1}{\Delta z} D_z$ slechts een discrete benadering (in waveletbasis) is voor de afgeleide $\frac{d}{dz}$. Daar de operator $\frac{d}{dz}$ zoals beschreven in sectie 4.2.5 steeds een benadering van de afgeleide geeft anaaloog met een centrale eindige verschillen methode kunnen we Yee's rooster (zie figuur 5.1) niet gebruiken voor de ruimtelijke discretisatie. $\frac{d}{dz}$ in waveletbasis geeft nl. een benadering van de afgeleide in de discretisatiepunten zelf en niet in het midden tussen twee discretisatiepunten zoals men gebruikt bij Yee's discretisatiestrooster.

Noemen we $E_{W_x}^n$ de wavelettransformatie van $E_{D_x}^n$ (analoge conventie voor $H_y^{n+\frac{1}{2}}$) dan kunnen we (5.11) en (5.12) in waveletbasis schrijven als we stellen dat $\epsilon_r = 1$ en $\mu_r = 1$:

$$E_{W_x}^n = E_{W_x}^{n-1} - \frac{c\Delta t}{\Delta z} D_z H_{W_y}^{n-\frac{1}{2}} \quad (5.13)$$

$$H_{W_y}^{n+\frac{1}{2}} = H_{W_y}^{n-\frac{1}{2}} - \frac{c\Delta t}{\Delta z} D_z E_{W_x}^n \quad (5.14)$$

We zouden ook de non-standaard vorm van $\frac{d}{dz}$ kunnen gebruiken. Dan moeten we voor de velden wel redundante waveletvoorstellingen gebruiken en telkens herprojecteren nadat we

D_z hebben toegepast (cf. 4.2.4). Dit is ingewikkelder dan het gebruik van de standaard vorm maar heeft een lagere rekenkundige complexiteit ($O(N)$ tegenover $O(N \log N)$) [20]. Omdat we in het tweedimensionale geval echter voor de standaard vorm zullen kiezen doen we dat hier ook.

Met formules (5.13)-(5.14) kunnen we alleen elektromagnetische velden in vacuüm simuleren. We hebben ϵ_r en μ_r immers gelijk aan 1 gesteld. In het vervolg zullen we steeds $\mu_r = 1$ kiezen omdat dit voor de materialen waarin we geïnteresseerd zijn zo is. Als we willen dat de simulaties enig praktisch nut hebben moeten we kunnen werken met $\epsilon_r = \epsilon_r(z) \neq 1$. Dit betekent dat $1/\epsilon_r(z)$ in waveletbasis moet vermenigvuldigd worden met $D_z H_{W_y}^{n-\frac{1}{2}}$. We kunnen dat doen door aan de hand van $1/\epsilon_r(z)$ een materiaaloperator M_W in waveletbasis te construeren zoals beschreven in sectie 4.4. (5.13) en (5.14) worden dan

$$E_{W_x}^n = E_{W_x}^{n-1} - s \cdot M_W D_z H_{W_y}^{n-\frac{1}{2}} \quad (5.15)$$

$$H_{W_y}^{n+\frac{1}{2}} = H_{W_y}^{n-\frac{1}{2}} - s \cdot D_z E_{W_x}^n \quad (5.16)$$

waarin $s = \frac{c\Delta t}{\Delta z}$. s noemt men het getal van Courant-Friedrichs-Levy [39].

(5.11)-(5.12) vormen het ééndimensionale algoritme. Opdat het numeriek stabiel zou zijn moet $s < 1$. Dit legt een beperking op aan de maximale tijdsstap Δt die men kan gebruiken voor een gegeven ruimtediscretisatieinterval Δz .

Simulaties

Voor we praktische berekeningen kunnen doen moeten we eerste kiezen met welke wavelets we gaan werken. We hebben geopteerd voor de Daubechies wavelets om verschillende redenen:

- Het zijn wavelets met compacte drager (compact support). Voor praktische implementaties is dit een noodzaak. Als met wavelets zonder compacte drager zou gebruiken moeten deze in de praktische berekeningen toch beperkt worden tot een eindig gebied. Men legt dus een eindige draagwijdte (support) op aan een wavelet zonder compacte drager. Hoewel ze meestal klein zijn zal dit toch fouten in de berekening veroorzaken.
- De Daubechies wavelets zijn de wavelets met de kleinst mogelijke draagwijdte voor een gegeven aantal nulmomenten (vanishing moments). We hebben in sectie 4.2.5 gezien dat een kleine draagwijdte een ijle $\frac{d}{dz}$ operatormatrix oplevert.
- Voor de $\frac{d}{dz}$ operator gebaseerd op de Daubechies wavelets hebben we superconvergentie (cf. 4.2.5).

- De Daubechies wavelets worden in de praktijk vaak gebruikt en hebben hun kwaliteit reeds lang bewezen.

Om de operatormatrix van $\frac{d}{dz}$ in waveletbasis in praktijk op te stellen moeten we de coëfficiënten r_l kennen (zie (4.36)):

$$r_l = \int_{-\infty}^{+\infty} \varphi(z-l) \frac{d}{dz} \varphi(z) dz$$

We kunnen de r_l berekenen door het stelsel (4.40)-(4.41) op te lossen.

In figuur 5.2 laten we met een voorbeeld zien dat de voorstelling in waveletbasis economischer is (meer coëfficiënten die nul zijn of heel klein) dan die in natuurlijke basis. Dit betekent niet dat de wavelet-voorstelling de meest economische is. Voor snel oscillerende functies presteert de Fouriervoorstelling bv. meestal beter [18].

Om te zien of het algoritme werkt proberen we eerst lichtpropagatie in vacuüm. Zie figuur 5.3. Een lichtpuls die invalt op een oppervlak (= een stap in de brekingsindex) is afgebeeld in figuren 5.4 en 5.5. In figuur 5.6 is de weerkaatsing van een puls op een gedistribueerde Bragg reflector weergegeven. De samengestelde operatormatrix $M_W D_z$ die bij de berekening van deze figuur werd gebruikt (zie 5.11) is te zien in figuur 5.7.

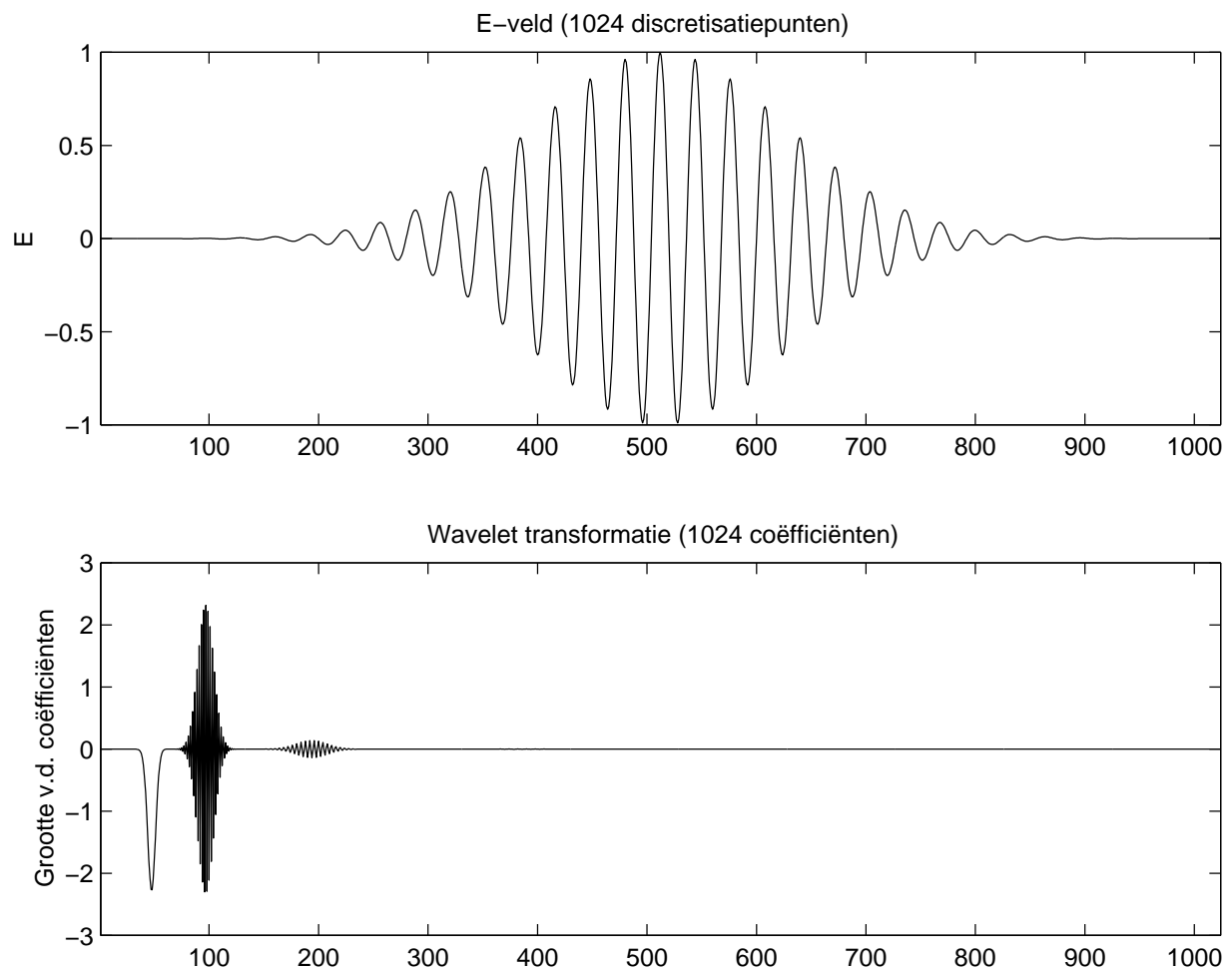
We hebben ook geprobeerd om een zogenaamde *absorbing boundary condition* (ABC) toe te passen zodat straling het simulatiedomein kan verlaten. De ABC die we getest hebben was de eerste-orde ABC van Mur [41]. Na inpassing in de wavelet gebaseerde simulatie bleek deze ABC echter maar gedeeltelijk te functioneren: de amplitude van pulsen die 'door de rand' gaan wordt door de ABC met ongeveer 50 tot 70 procent verlaagd. De voornaamste oorzaken hiervan zijn waarschijnlijk:

- De geïmplementeerde ABC is een eerste orde benadering terwijl de door ons gebruikte methode steeds met minimum vierde orde afgeleiden werkt (dit komt overeen met Daubechies wavelets met 2 nulmomenten).
- De Mur-ABC is eigenlijk ontworpen om gebruikt te worden met Yee's discretisatie-rooster. Zoals eerder uitgelegd kan dit discretisatierooster niet gebruikt worden in onze wavelet-gebaseerde simulatiemethode.

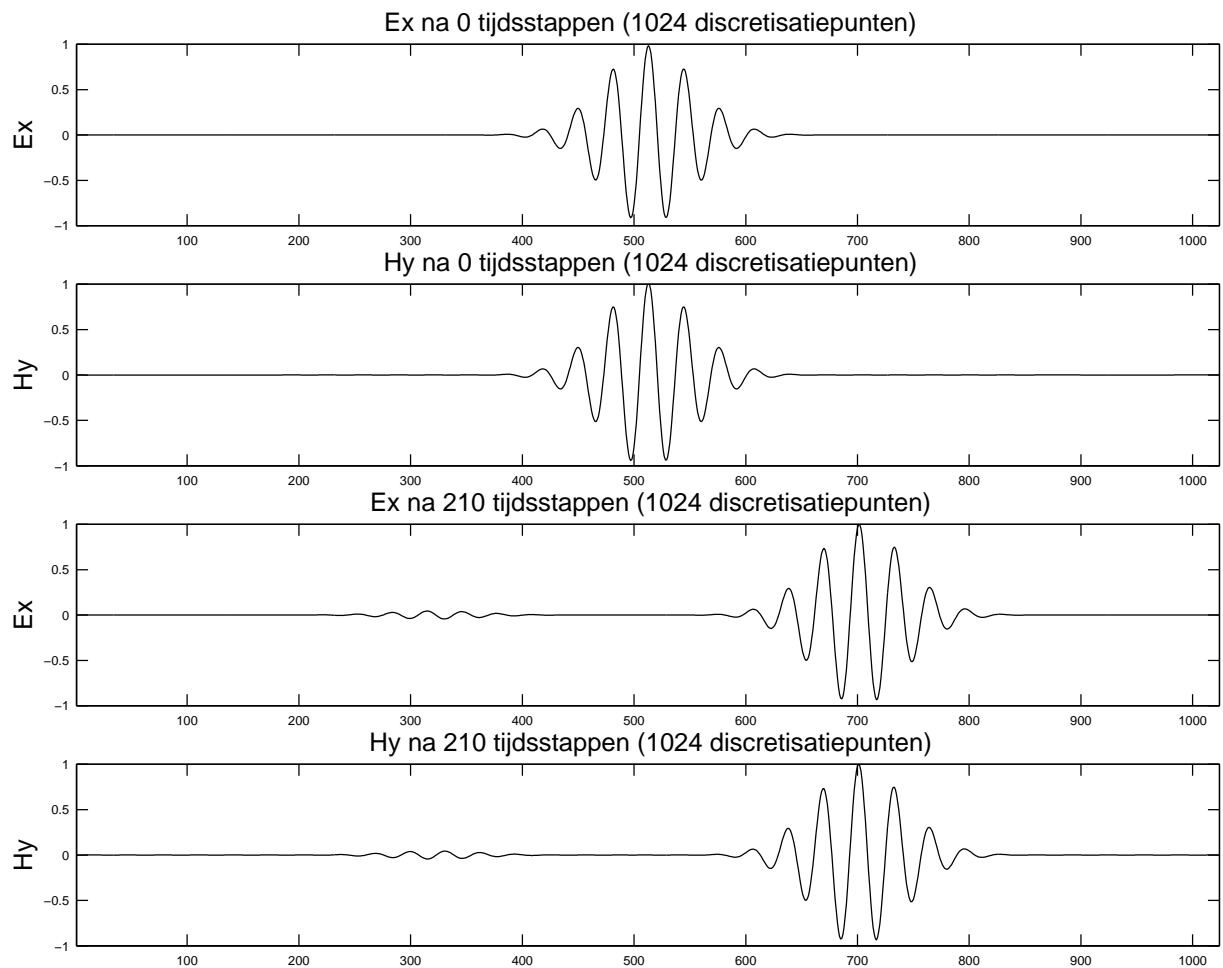
5.2.2 Twee dimensies

Theorie

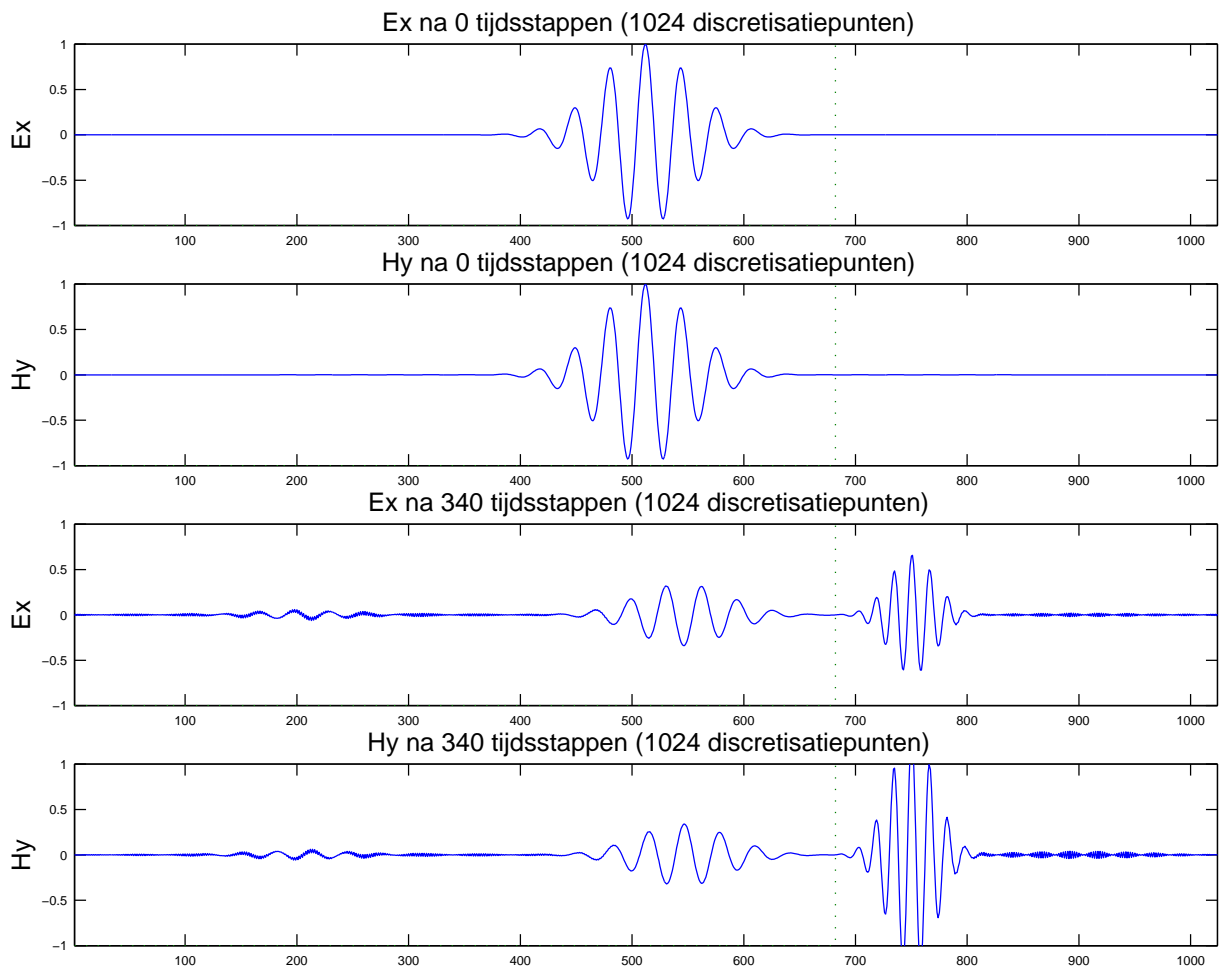
We kunnen een tweedimensionaal simulatiealgoritme bekomen door te veronderstellen dat de velden niet variëren in één richting, bv. de z -richting. We moeten dus alle afgeleiden



Figuur 5.2: Voorbeeld van een elektrische veldverdeling (boven) en de wavelettransformatie ervan (onder). De transformatie is berekend met de Daubechies wavelet met 6 nulmomenten. De coëfficiënten van de wavelettransformatie zijn als volgt afgebeeld: $(s_{7,1}; d_{7,1}; d_{6,1}, d_{6,2}; d_{5,1}, d_{5,2}, d_{5,3}, d_{5,4}; \dots; d_{1,1}, \dots, d_{1,1024})$. We zien duidelijk dat veel coëfficiënten nul of heel klein zijn.



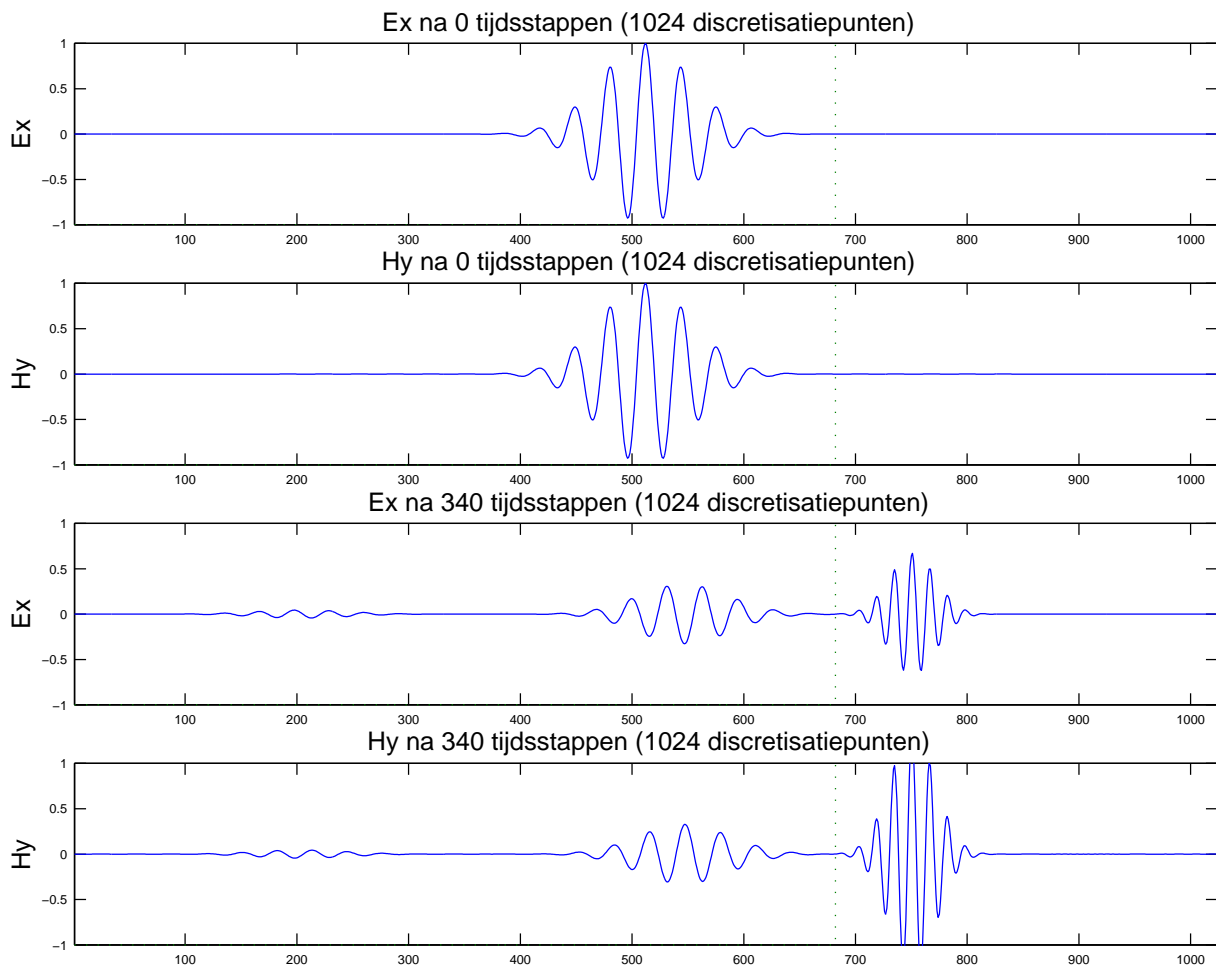
Figuur 5.3: Propagatie van een puls in vacuüm. De startvelden zijn weergegeven in de bovenste twee grafieken (E_x en H_y componenten) en de velden na 210 propagatiestappen in de onderste twee. Hoewel de veldcomponenten in waveletbasis berekend zijn worden ze hier in natuurlijke basis weergegeven (na inverse wavelettransformatie). We gebruikten de Daubechies wavelet met 6 nulmomenten. De kleine puls die naar links vertrekt wordt veroorzaakt door een lichte 'mismatch' tussen E_x en H_y in de beginwaarden. We hebben E_x en H_y daar immers gelijkgesteld terwijl ze in het algoritme over een halve tijdstap ten opzichte van elkaar verschoven zijn.



Figuur 5.4: Puls die invalt op een oppervlak, berekend met de Daubechies wavelet met 6 nulmomenten. De plaats van de brekingsindexstap is weergegeven in stippellijn. Links van deze stippellijn is de brekingsindex gelijk aan 1, rechts is hij gelijk aan 2.

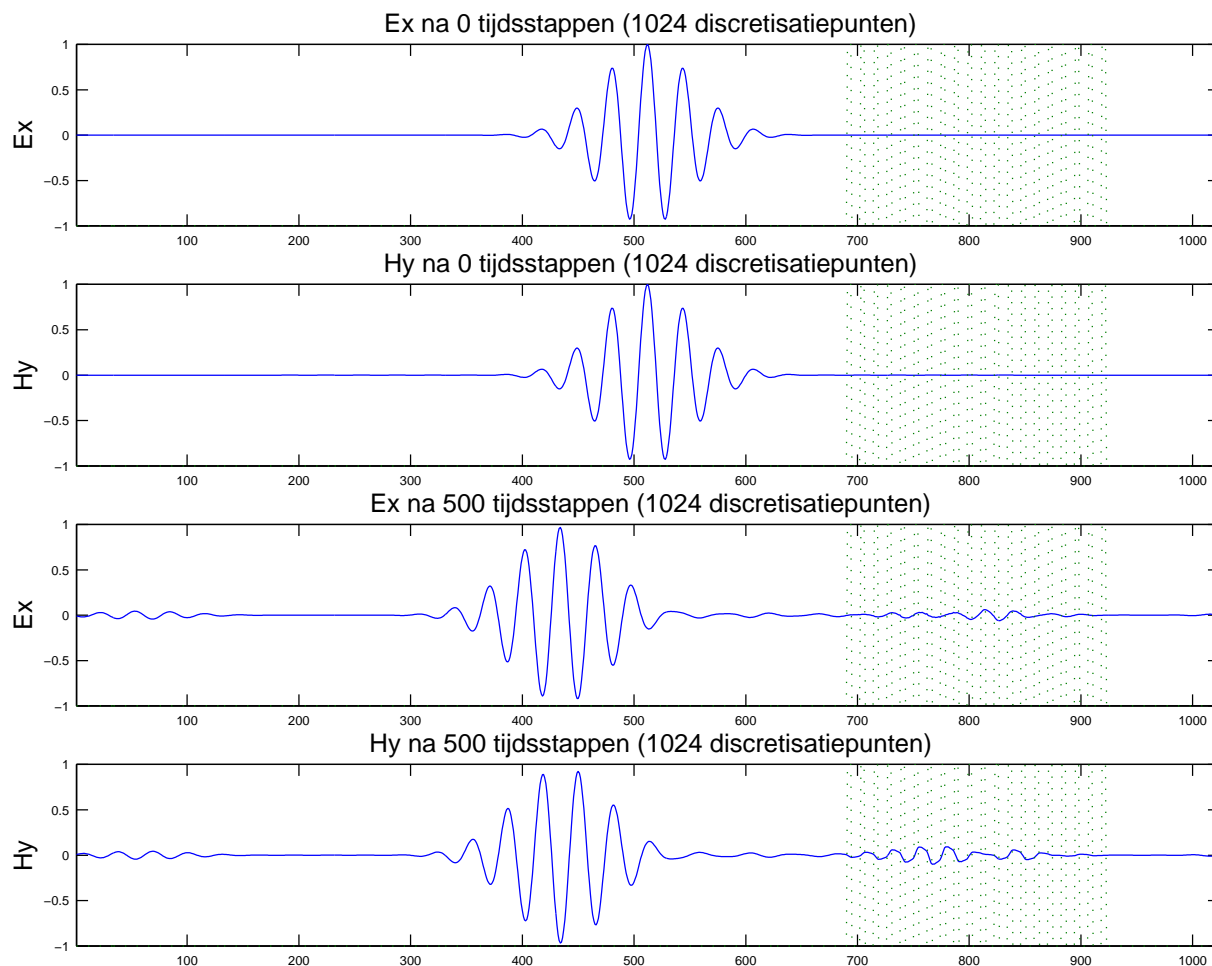
We zien dat er op sommige plaatsen hoogfrequente oscillaties op de velden gesuperponeerd zijn. Door het aantal nulmomenten van de gebruikte wavelet te verhogen kunnen de oscillaties verminderd worden. Ze worden veroorzaakt door de discontinuïteit van de brekingsindex. De oscillaties kunnen soms 'weggefilterd' worden door de detailcoëfficiënten op de fijnste schaal na het berekenen van elke stap nul te stellen. Zie hiervoor figuur 5.5.

Mogelijke verklaring voor het ontstaan van de oscillaties: aan het oppervlak verandert het elektrisch veld vrij plots. Dit geeft aanleiding tot scherpe overgangen (pieken) in de afgeleide. Deze worden via formule (5.16) in het magnetisch veld geïntroduceerd en via (5.15) in het elektrische veld. Door numerieke dispersie 'reizen' deze hoogfrequente componenten veel sneller door het systeem dan de gewone straling en verspreiden ze zich.

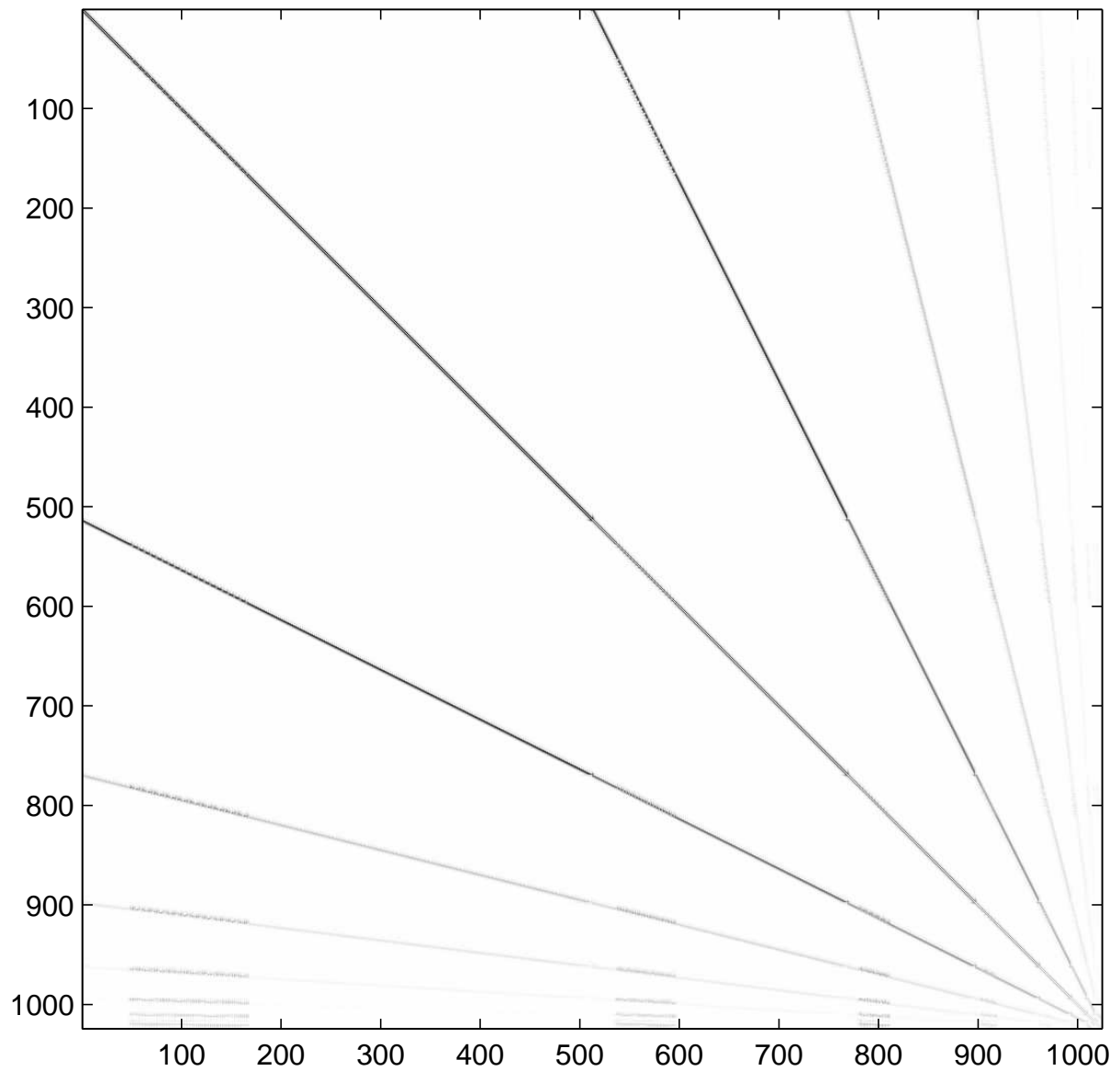


Figuur 5.5: Zelfde situatie als in figuur 5.4 maar nu hebben we na elke stap de detailcoëfficiënten op de fijnste schaal nul gesteld. Als het aantal punten per golflengte van het licht groot genoeg is kunnen we dit doen zonder significante fouten te maken. De wavelet- of detailcoëfficiënten van de velden zelf zullen op de fijnste schaal immers heel klein zijn (zoals te zien op figuur 5.2). Ze zouden tijdens een gewone thresholding operatie sowieso nul gemaakt worden. De numerieke fout veroorzaakt door de discontinuïteit van de brekingsindex zorgt er echter voor dat de detailcoëfficiënten op de fijnste schaal groter worden. Door ze nul te stellen filteren we de fout eruit zonder de velden noemenswaardig aan te tasten. Dit is duidelijk zichtbaar in de figuur: de veldprofielen zijn quasi dezelfde als in figuur 5.4, enkel de numerieke fouten zijn verdwenen.

De reden dat we genoeg punten per golflengte nodig hebben is de volgende: met weinig punten per golflengte zullen de detailcoëfficiënten op de fijnste schaal op sommige plaatsen niet heel klein zijn en zouden ze bij een thresholding operatie dus niet nul gemaakt worden. In dat geval maken we een serieuze fout door ze nul te stellen.



Figuur 5.6: Een puls voor en na reflectie op een gedistribueerde Bragg reflector (met alternerende brekingsindices $n_1 = 1$ en $n_2 = 2$). De puls op de twee bovenste figuren beweegt naar rechts, die op de twee onderste naar links. De figuren werden berekend met de Daubechies wavelet met 6 nulmomenten en na elke stap 'gefilterd' zoals beschreven bij figuur 5.5. De samengestelde operatormatrix $M_W D_z$ in standaard vorm die gebruikt werd bij de berekeningen is afgebeeld in figuur 5.7.



Figuur 5.7: Standaard vorm van de samengestelde operator $M_W D_z$ (zie formule (5.11)). De matrix werd berekend met de Daubechies wavelet met 6 nulmomenten. Grotere matrixelementen (in absolute waarde) zijn donkerder weergegeven. Het is duidelijk dat deze matrix ijl is.

naar z nul stellen in (5.5)-(5.10):

$$\begin{aligned} E_x^n &= E_x^{n-1} + \frac{c\Delta t}{\epsilon_r} \frac{\partial H_z^{n-\frac{1}{2}}}{\partial y} \\ E_y^n &= E_y^{n-1} - \frac{c\Delta t}{\epsilon_r} \frac{\partial H_z^{n-\frac{1}{2}}}{\partial x} \\ E_z^n &= E_z^{n-1} + \frac{c\Delta t}{\epsilon_r} \left(\frac{\partial H_y^{n-\frac{1}{2}}}{\partial x} - \frac{\partial H_x^{n-\frac{1}{2}}}{\partial y} \right) \\ H_x^{n+\frac{1}{2}} &= H_x^{n-\frac{1}{2}} - \frac{c\Delta t}{\mu_r} \frac{\partial E_z^n}{\partial y} \\ H_y^{n+\frac{1}{2}} &= H_y^{n-\frac{1}{2}} + \frac{c\Delta t}{\mu_r} \frac{\partial E_z^n}{\partial x} \\ H_z^{n+\frac{1}{2}} &= H_z^{n-\frac{1}{2}} - \frac{c\Delta t}{\mu_r} \left(\frac{\partial E_y^n}{\partial x} - \frac{\partial E_x^n}{\partial y} \right) \end{aligned}$$

Volledig analoog aan het ééndimensionale geval kunnen we dit omvormen tot

$$\begin{aligned} E_{W_x}^n &= E_{W_x}^{n-1} + s \cdot M_W D_y H_{W_z}^{n-\frac{1}{2}} \\ E_{W_y}^n &= E_{W_y}^{n-1} - s \cdot M_W D_x H_{W_z}^{n-\frac{1}{2}} \\ E_{W_z}^n &= E_{W_z}^{n-1} + s \cdot M_W \left(D_x H_{W_y}^{n-\frac{1}{2}} - D_y H_{W_x}^{n-\frac{1}{2}} \right) \\ H_{W_x}^{n+\frac{1}{2}} &= H_{W_x}^{n-\frac{1}{2}} - s \cdot D_y E_{W_z}^n \\ H_{W_y}^{n+\frac{1}{2}} &= H_{W_y}^{n-\frac{1}{2}} + s \cdot D_x E_{W_z}^n \\ H_{W_z}^{n+\frac{1}{2}} &= H_{W_z}^{n-\frac{1}{2}} - s \cdot (D_x E_{W_y}^n - D_y E_{W_x}^n) \end{aligned}$$

Hierbij is het discretisatieinterval in x - en y -richting gelijk, $\Delta x = \Delta y$, en $s = \frac{c\Delta t}{\Delta x} = \frac{c\Delta t}{\Delta y}$.

De $E_{W_\alpha}^n$ en $H_{W_\alpha}^{n+\frac{1}{2}}$ ($\forall n \in \mathbb{N}$ en met $\alpha = x, y, z$) zijn nu matrices en M_W , D_x en D_y zijn vierdimensionale operatortensoren in standaard vorm. De $E_{W_x}^n$ en $H_{W_x}^{n+\frac{1}{2}}$ zijn ontbindingen in standaard tensor-product waveletbasis. We kozen voor de standaard vorm om twee redenen:

1. De operatoren D_x en D_y zijn duidelijk separabel (ze werken immers maar op één variabele). Dat betekent dat we om bv. D_x toe te passen niet de volledige vierdimensionale tensor moeten opstellen en gebruiken maar dat we de operatormatrix van $\frac{d}{dx}$ in standaard vorm op elke rij van de matrix $E_{W_x}^n$ kunnen toepassen (cf. 4.3.3).
2. De tensorvoorstelling van een operator voor tweedimensionale gediscetiseerde functies blijft in standaard vorm een tensor van dezelfde afmetingen. Bij de non-standaard

Materiaaloperator	Ijlheidsgraad
$64 \times 64 \times 64 \times 64$, natuurlijke basis	$\approx 99,8\%$
$64 \times 64 \times 64 \times 64$, standaard vorm, Daubechies met 2 nulmomenten	$\approx 93\%$
$64 \times 64 \times 64 \times 64$, standaard vorm, Daubechies met 8 nulmomenten	$\approx 30\%$

Tabel 5.1: Percentage nulelementen in verschillende materiaaloperatoren. Dit percentage is sterk afhankelijk van de afmetingen van de operatortensor en het aantal nulmomenten van de gebruikte wavelet (een groter aantal nulmomenten betekent een gotere draagwijdte (support)). De ijleheidsgraad stijgt met stijgende afmetingen en daalt met stijgend aantal nulmomenten.

vorm is dat niet het geval en krijgen we een hele verzameling tensoren (cf. 4.3.1). Kiezen voor de non-standaard vorm impliceert ook dat we de $E_{W\alpha}^n$ en $H_{W\alpha}^{n+\frac{1}{2}}$ op een redundante manier ontwikkelen in een tweedimensionale waveletbasis gebaseerd op een multiresolutie analyse. Dat is vrij ingewikkeld daar het resultaat niet meer in een matrix van dezelfde grootte past (omwille van de redundantie).

De voorwaarde voor numerieke stabiliteit is dezelfde als bij de overeenkomstige FDTD methode [33, 37]:

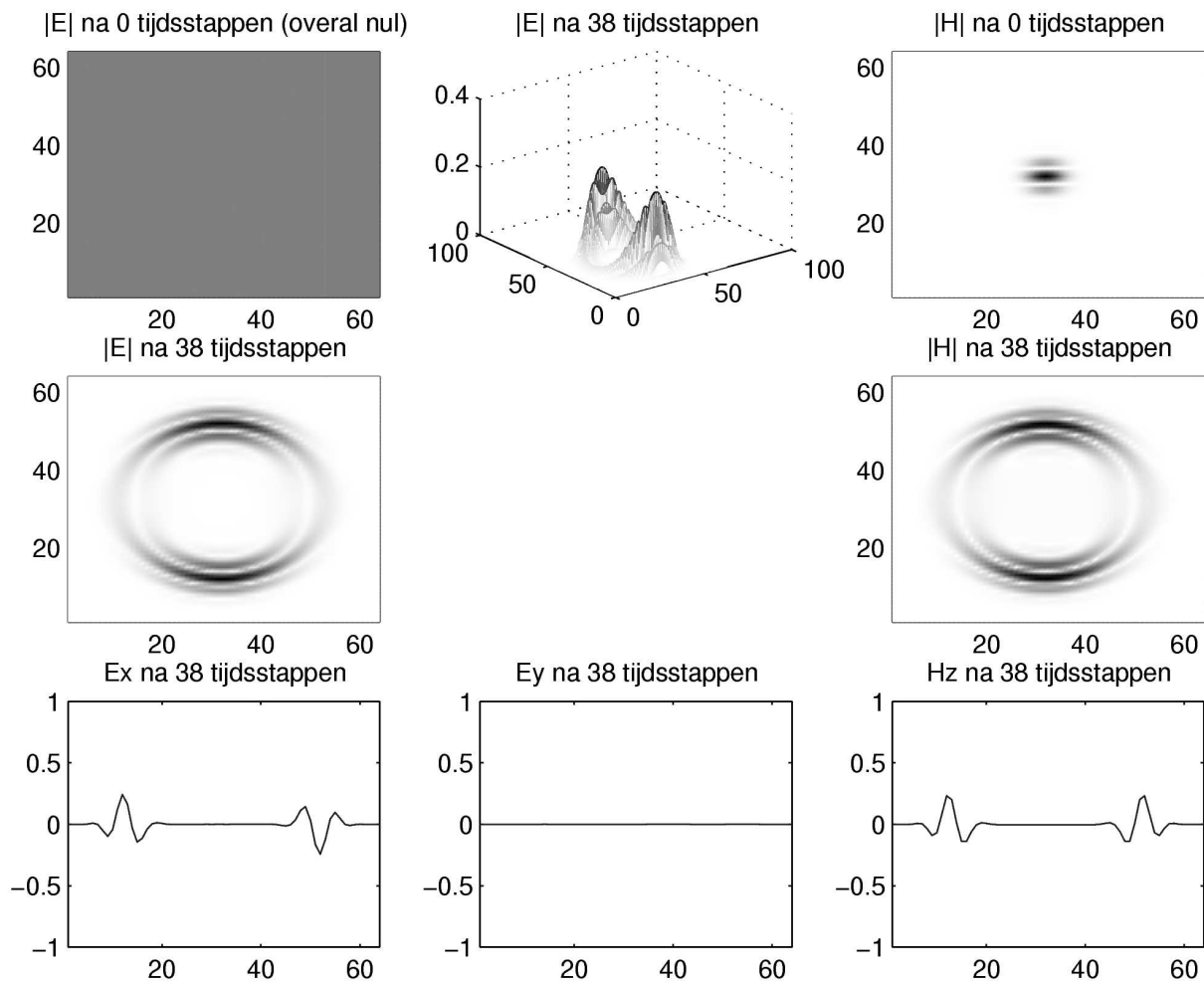
$$\Delta t \leq \frac{1}{c\sqrt{\Delta x^2 + \Delta y^2}}$$

Simulaties

Een probleem bij de praktische implementatie is het toepassen van de materiaaloperator M_W , een vierdimensionale tensor. M_W is meestal ijle zodat vermenigvuldigingen van M_W met matrices in theorie snel kunnen berekend worden. Matlab beschikt echter niet over routines die met ijle tensoren kunnen werken. Alleen ijle matrices worden ondersteund. Zelfs als M_W ijle is kost het dus toch $O(N^4)$ bewerkingen om de tensor met een matrix te vermenigvuldigen. In de implementatie hebben we het probleem kunnen verbeteren door (4.61) toe te passen (cf. 4.3.4). (4.61) lost echter niet alles op. Routines die met ijle tensoren kunnen werken zijn voor een snelle implementatie absoluut noodzakelijk, niet enkel om de rekentijd te beperken maar ook om de geheugenvereisten te verminderen. Een simulatiegebied van 64 bij 64 punten vereist bv. een $64 \times 64 \times 64 \times 64$ materiaaltensor. Om die op te slaan is 128 MB geheugen nodig (bij gebruik van 64 bits per getal).

Het percentage nulelementen in M_W is sterk afhankelijk van de afmetingen van M_W en van de draagwijdte (support) van de wavelet die gebruikt werd. Enkele voorbeelden staan in tabel 5.1.

Een simulatie van pulspropagatie in vacuüm is weergegeven in figuur 5.8. Het simulatiegebied is 64 bij 64 punten groot. Een groter gebied is niet haalbaar op een standaard



Figuur 5.8: Voortplanting van pulsen in vacuüm zoals berekend met de tweedimensionale methode. We gebruikten Daubechies wavelets met 8 nulmomenten. De diffractie is duidelijk zichtbaar op de figuren van de tweede rij. De figuren van de derde rij geven een doorsnede volgens de y -richting (kolom nr. 32) van enkele veldcomponenten (de niet weergegeven componenten zijn nul).

computer. Als we het aantal punten per golflengte verminderen merken we duidelijk dat de numerieke fouten toenemen. Dit is te zien in figuur 5.9. Als we ook beginwaarden geven voor één van de x - of y -componenten zien we dat er na een redelijk aantal propagatiestapen nog steeds resten van deze beginwaarden blijven bestaan. Dit is afgebeeld in figuur 5.10. De oorzaak van dit fenomeen is voorlopig nog niet gevonden.

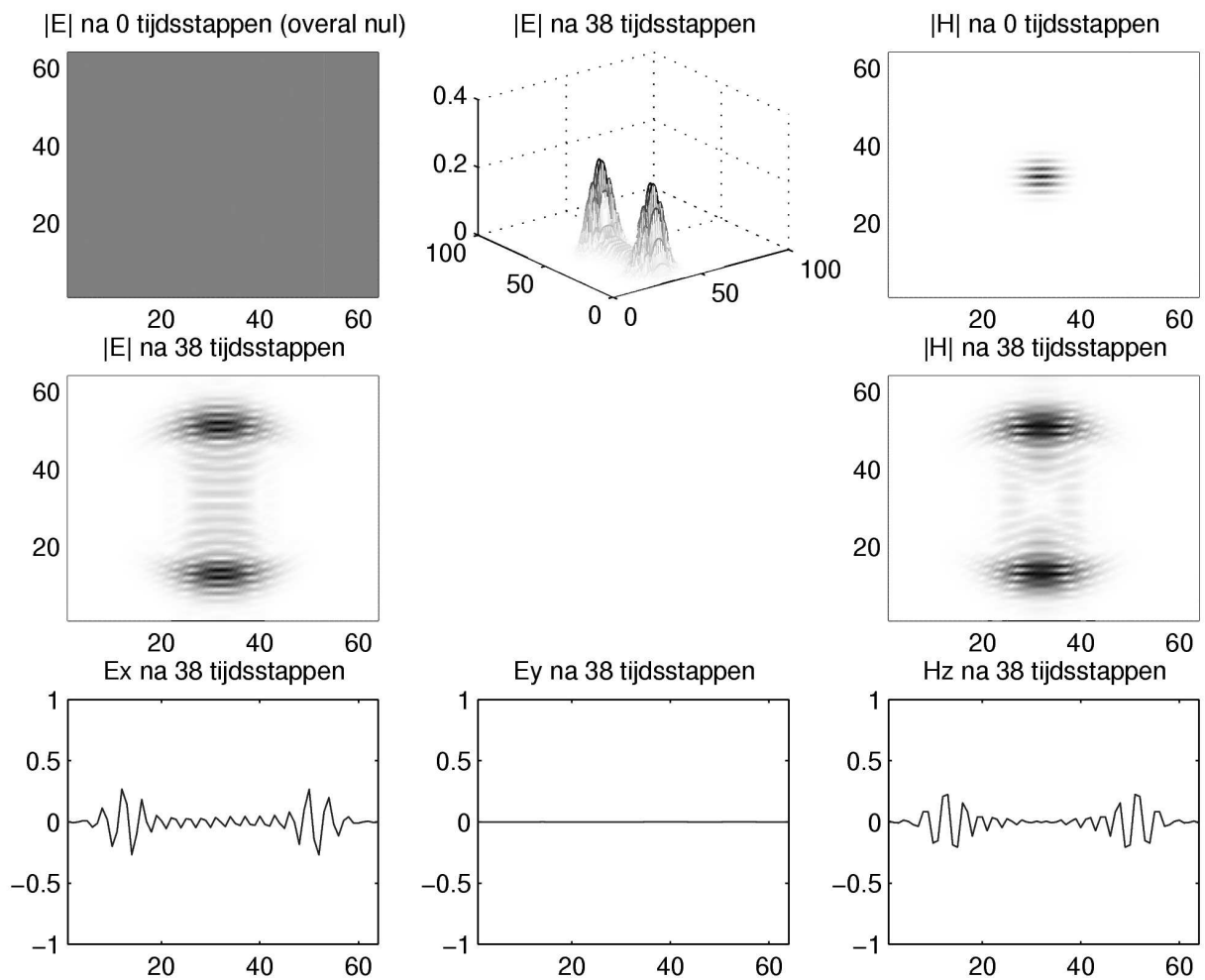
De diffractie van licht door een opening is te zien in figuur 5.11. Ook reflectie van een puls op een Bragg reflector werd uitgetoet. Dit werkt echter niet omwille van de beperkte resolutie. De tweedimensionale implementatie heeft immers net zoals de ééndimensionale versie last van het voorkomen van hoogfrequente oscillaties veroorzaakt door discontinuïteiten in de brekingsindex. Hoe lager het aantal punten hoe groter de fout die deze oscillaties veroorzaken. In het ééndimensionale geval hebben we deze weggefilterd door de detailcoëfficiënten op de fijnste schaal nul te maken. Als we maar 64 punten hebben mag dit echter niet (zie uitleg bij figuur 5.5): het aantal punten per golflengte is noodzakelijk laag, want als we de frequentie verlagen (en dus het aantal punten per golflengte groter maken) dan past de Bragg reflector niet meer in het simulatiegebied (de laagjes worden immers breder). Tests met de ééndimensionale implementatie bevestigen dat de numerieke fouten onaanvaardbaar groot zijn als we een Bragg reflector proberen te simuleren met slechts 64 discretisatiepunten.

Tenslotte hebben we voor een simulatie in vacuüm na elke stap een getal evenredig met het totale vermogen in het systeem berekend (de som van $\|\mathbf{E}\|^2$ in elk punt). De resultaten zijn te zien in figuur 5.12. We zien dat het tijdsgemiddelde van het vermogen ongeveer constant blijft.

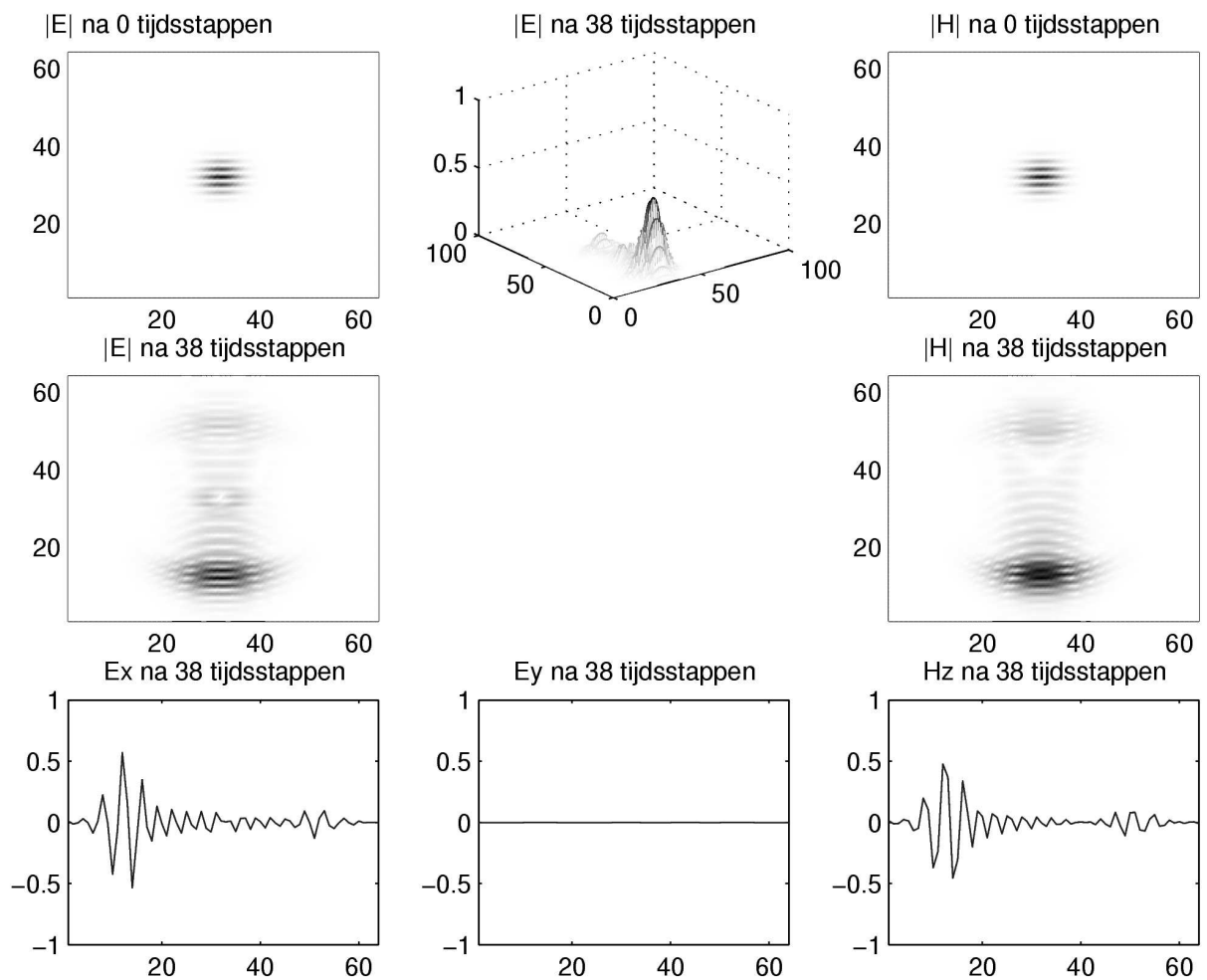
5.2.3 Drie dimensies

Het rechtstreeks in waveletbasis schrijven van (5.5)-(5.10) levert de driedimensionale methode:

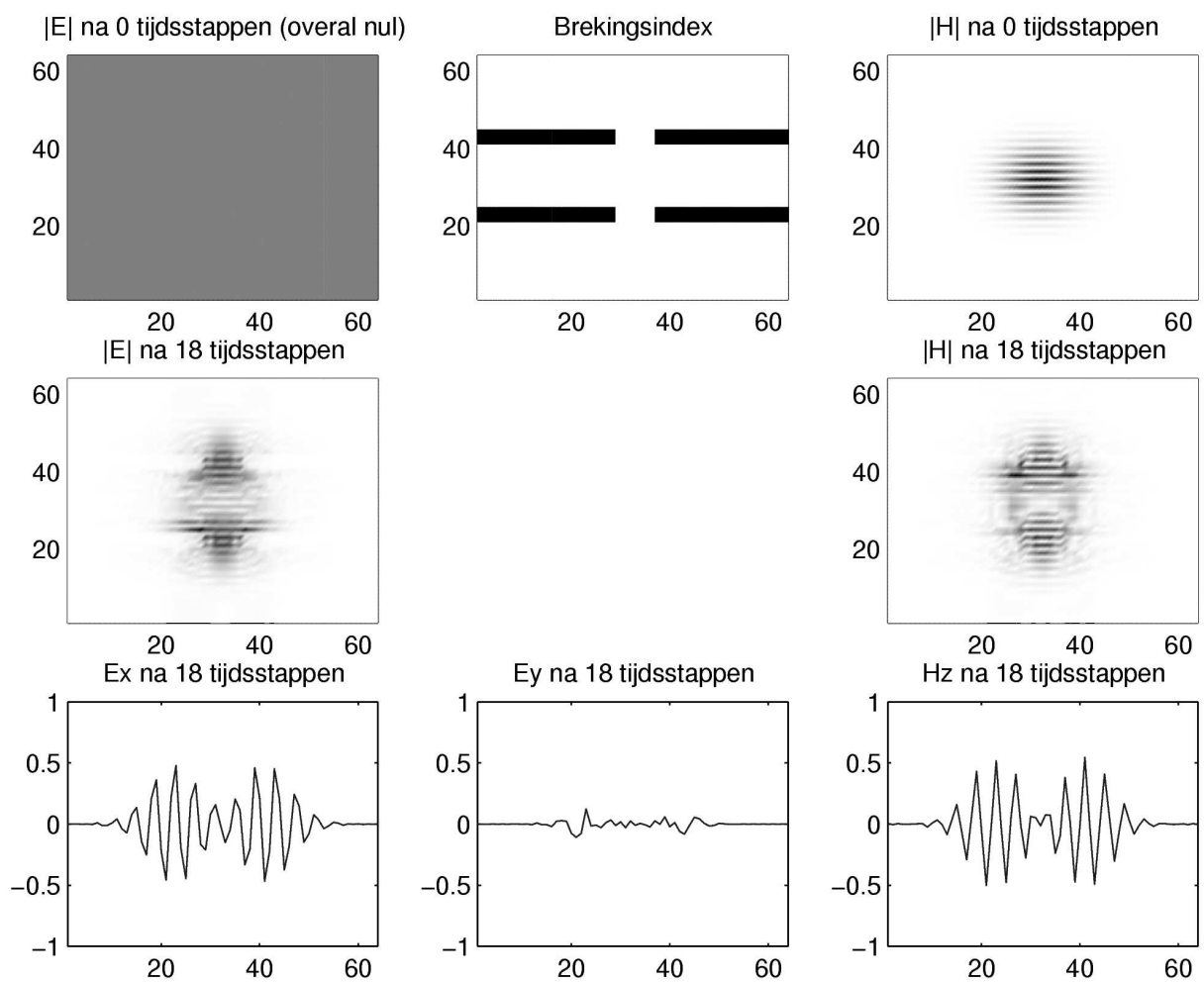
$$\begin{aligned}
 E_{W_x}^n &= E_{W_x}^{n-1} + s \cdot M_w \left(D_y H_{W_z}^{n-\frac{1}{2}} - D_z H_{W_y}^{n-\frac{1}{2}} \right) \\
 E_{W_y}^n &= E_{W_y}^{n-1} + s \cdot M_w \left(D_z H_{W_x}^{n-\frac{1}{2}} - D_x H_{W_z}^{n-\frac{1}{2}} \right) \\
 E_{W_z}^n &= E_{W_z}^{n-1} + s \cdot M_w \left(D_x H_{W_y}^{n-\frac{1}{2}} - D_y H_{W_x}^{n-\frac{1}{2}} \right) \\
 H_{W_x}^{n+\frac{1}{2}} &= H_{W_x}^{n-\frac{1}{2}} - s \cdot (D_y E_{W_z}^n - D_z E_{W_y}^n) \\
 H_{W_y}^{n+\frac{1}{2}} &= H_{W_y}^{n-\frac{1}{2}} - s \cdot (D_z E_{W_x}^n - D_x E_{W_z}^n) \\
 H_{W_z}^{n+\frac{1}{2}} &= H_{W_z}^{n-\frac{1}{2}} - s \cdot (D_x E_{W_y}^n - D_y E_{W_x}^n)
 \end{aligned}$$



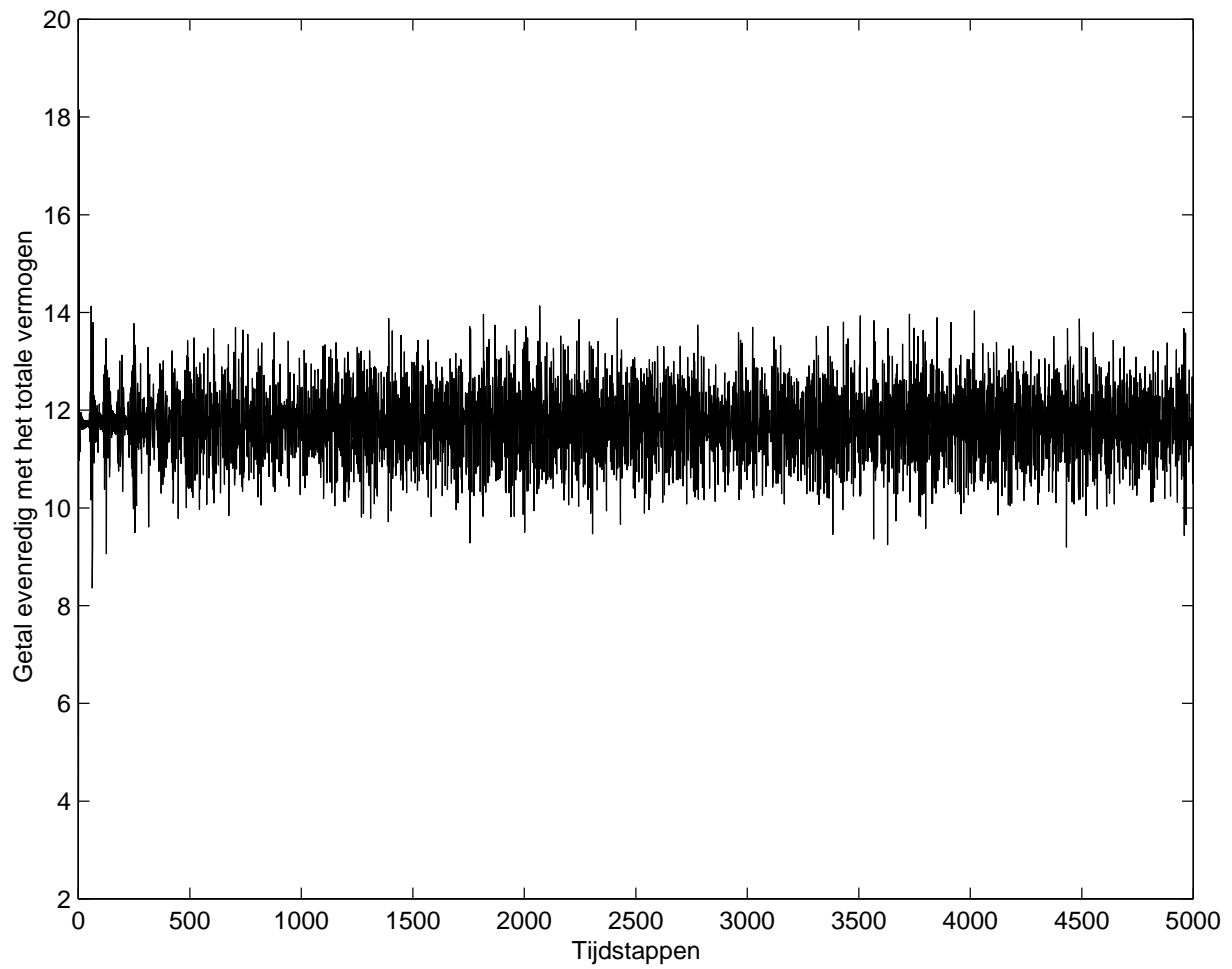
Figuur 5.9: Voortplanting van pulsen in vacuüm. Het aantal punten per golflengte is half zo groot als in figuur 5.8, de andere parameters zijn gelijk. We zien dat de grootte van de numerieke fouten is toegenomen.



Figuur 5.10: Als we ook beginwaarden geven voor E_x zijn er na 38 simulatiestappen nog steeds resten van deze beginwaarden zichtbaar in het elektrisch veld. De oorzaak hiervan is nog niet gevonden. Gebruikte wavelet: Daubechies met 8 nulmomenten.



Figuur 5.11: Diffractie van licht door een opening. De figuur werd gemaakt met de Daubechies wavelets met 8 nulmomenten. De donker afgebeelde delen in het brekingsindexprofiel hebben brekingsindex $n = 2$, voor de lichte delen geldt $n = 1$.



Figuur 5.12: Grafiek van de som van $\|\mathbf{E}\|^2$ in alle punten, berekend na elke stap. Deze som is evenredig met het totale vermogen aanwezig in het systeem. We zien dat het tijdsgemiddelde van het vermogen ongeveer constant is.

Nu hebben we $\Delta x = \Delta y = \Delta z$ en $s = \frac{c\Delta t}{\Delta x} = \frac{c\Delta t}{\Delta y} = \frac{c\Delta t}{\Delta z}$. Als we gebruik maken van de standaard vorm kunnen de operatoren D_x , D_y en D_z in hun matrix-standaard-vorm toegepast worden volgens de drie richtingen. M_W is echter een zesdimensionale tensor. Door dit hoge aantal dimensies kan de driedimensionale methode in de praktijk niet toegepast worden. Als we bv. 64 discretisatiepunten nemen volgens elke richting zullen we om de standaard vorm van M_W op te stellen $64^6 = 2^{36}$ of bijna 69 miljard ééndimensionale wavelettransformaties moeten uitvoeren. Hoewel het berekenen van deze transformaties zeer goed geschikt is voor uitvoering op een parallelle computer lijkt dit aantal toch te groot. Als we geen gebruik kunnen maken van een ijle voorstelling zou de tensor M_W bovendien 64 gigabyte geheugen innemen (bij gebruik van 64 bits dubbele precisie getallen). Om de driedimensionale methode praktisch bruikbaar te maken zouden we rechtstreeks een ijle voorstelling van M_W in waveletbasis moeten kunnen opstellen. Dat is niet evident en ik ken geen in de literatuur beschreven methodes om iets dergelijks te doen.

5.3 Besluit

Na een korte inleiding over wavelet-gebaseerde methodes voor numerieke berekeningen in elektromagnetisme hebben we in dit hoofdstuk vooral de methode van Werthen & Wolff [28] behandeld. We hebben vastgesteld dat voorlopig alleen de ééndimensionale versie echt op punt staat. De tweedimensionale implementatie lijdt sterk onder het gebrek aan routines die met ijle tensoren kunnen werken. Om het driedimensionale algoritme praktisch uitvoerbaar te maken zijn de wiskundige tools momenteel niet beschikbaar.

Een belangrijk nadeel van de methode van Werthen & Wolff is dat ze niet toelaat om Yee's discretisatiestrooster te gebruiken. Dit rooster is bijzonder goed geschikt voor numerieke elektromagnetische berekeningen.

Daar veel interessante VCSEL's (bv. VCSEL's met oxide aperturen) niet goed in twee dimensies kunnen gesimuleerd worden is de hier besproken methode momenteel nog niet geschikt voor het simuleren van deze lasers. Hieruit mag echter niet geconcludeerd worden dat wavelets volledig ongeschikt zijn voor de numerieke simulatie van VCSEL's. De combinatie van [32] en [30] zou bv. een goede methode kunnen opleveren. Ook het tweedimensionale algoritme dat we geïmplementeerd hebben zou nuttig kunnen zijn als er routines voor ijle tensoren zouden ontwikkeld worden.

Hoofdstuk 6

Besluit

Het doel van dit afstudeerwerk was het bestuderen van de mogelijkheid om wavelets te gebruiken voor de numerieke simulatie van vertical cavity surface emitting lasers. De waveletontwikkeling van de elektromagnetische velden in de VCSEL caviteit is economischer (meer coëfficiënten die nul of heel klein zijn) dan de voorstelling in natuurlijke basis (hoewel de Fouriervoorstelling in het geval van VCSEL's mogelijk nog economischer is [18]). Bewerkingen (operaties) uitvoeren met waveletvoorstellingen is echter niet evident.

In dit afstudeerwerk hebben we ons voornamelijk geconcentreerd op het gebruik van operatoren in waveletbasis. Hierover is bijzonder weinig praktijkgerichte literatuur beschikbaar, zeker voor meerdere dimensies. Om praktische toepassingen te realiseren volstaat een globale indruk van de materie echter niet en moeten alle details bekend zijn. In hoofdstuk 4 hebben we deze details uitgewerkt voor standaard en non-standaard vorm operatoren in het één en tweedimensionale geval. Er is gebleken dat de standaard vorm (die minder ijle voorstellingen oplevert dan de non-standaard vorm) het gemakkelijkst te gebruiken is.

De prototype-implementatie van een wavelet-gebaseerde methode die in hoofdstuk 5 besproken werd heeft duidelijk gemaakt dat er nog een aantal praktische problemen bestaan. Het belangrijkste is het ontbreken van een snelle methode om te vermenigvuldigen in waveletbasis. We realiseerden vermenigvuldigingen met behulp van de operatoren uit hoofdstuk 5. Doordat we geen routines ter beschikking hadden die kunnen werken met ijle tensoren is dit in de praktijk te traag. Voor de non-standaard vorm bestaat er wel een snel vermenigvuldigingsalgoritme [27]. Deze vorm laat echter niet toe scheidbare operatoren in verschillende stappen toe te passen, wat de implementatie sterk bemoeilijkt.

Hoofdstuk 5 besprak één-, twee- en driedimensionale versies van één simulatiemethode. Om het driedimensionale algoritme praktisch bruikbaar te maken ontbreken voorlopig nog enkele wiskundige tools. Een tweedimensionale versie kon wel gerealiseerd worden, maar daarbij kregen we te kampen met de moeilijkheid van het vermenigvuldigen in waveletbasis. Het ééndimensionale algoritme is wel op een efficiënte manier gerealiseerd.

Voor de techniek uit hoofdstuk 5 geschikt kan zijn voor het simuleren van VCSEL's zijn dus nog een aantal verbeteringen nodig.

Toekomst

Er zijn verschillende richtingen die men zou kunnen uitgaan om de toepasbaarheid van wavelets voor het simuleren van VCSEL's verder te onderzoeken:

- Een eerste mogelijkheid is verdergaan met een techniek analoog aan degene die in hoofdstuk 5 is beschreven. Men zou een functiebibliotheek kunnen ontwikkelen voor het werken met ijle tensorvoorstellingen. Om een driedimensionaal algoritme mogelijk te maken kan er gezocht worden naar methoden om de materiaaloperator rechtstreeks in waveletbasis op te stellen.
- Een andere (veelbelovende) mogelijkheid is de Green's functie aanpak. Met behulp van een nieuwe methode van Paulus en Martin [32] kan misschien de Green's functie van een VCSEL caviteit worden opgesteld. De velden in de caviteit kunnen dan gevonden worden door het oplossen van een integraalvergelijking, een gebied waarin wavelets sterk staan. Technieken zoals die uit van Xiang en Lu [30] kunnen misschien gebruikt worden om het aantal numerieke integraties nodig voor het oplossen van deze vergelijking te beperken.
- Een derde domein dat kan bestudeerd worden is multiresolutie homogenisatie [42, 43]. Bij een VCSEL wordt het veldprofiel op de fijne schalen beïnvloed door gelokaliseerde structuren (bv. een oxide-apertuur). Die fijne schalen beïnvloeden op hun beurt de groffe schalen en dus het globale veldprofiel. Hoewel we de oplossing niet op een echt fijne schaal nodig hebben kunnen we de fijne schalen niet verwaarlozen als we de correcte oplossing willen vinden. Bij multiresolutie homogenisatie probeert men aan de hand van multiresolutie analyse benaderde vergelijkingen op te stellen voor het systeem die het gedrag op de fijne schalen zo goed mogelijk in rekening brengen zonder de berekeningen op die fijne schalen uit te voeren.

Bijlage A

Broncode

De prototype-implementaties uit hoofdstuk 5 werden gerealiseerd in Matlab met behulp van de WaveLab bibliotheek. We gebruikten versie 802 van WaveLab, die gratis kan binnengehaald worden via

```
http://www-stat.stanford.edu/~wavelab
```

WaveLab is een bibliotheek van functies waarmee allerlei wavelettransformaties en waveletgerelateerde bewerkingen kunnen uitgevoerd worden. De filosofie achter WaveLab is de *reproducible research* van David Donoho:

An article about computational science in a scientific publication is *not* the scholarship itself, it is merely *advertising* of the scholarship. The actual scholarship is the complete software development environment and the complete set of instructions which generated the figures.

In de geest van deze filosofie wordt in deze appendix een overzicht gegeven van de Matlab code.

- Implementatie van de ééndimensionale Werthen & Wolff methode. Informatie over de niet-Matlab functies die hierin gebruikt worden zal verder gegeven worden.

```
% 1D Werthen&Wolff methode  
  
% aantal punten  
N = 1024;  
  
% aantal vanishing moments
```

```

vmoments = 8;

% h-Filter
h = MakeOnFilter('Daubechies',2*vmoments);
Lcoarse = 3; % ontbinden tot er nog 2^3 punten over zijn

% Beylkin's r-coefficienten berekenen
r = ddxcoeff(h);
% index r_0
rzero = (length(r)-1)/2+1;

% Afgeleide matrix opstellen (in waveletbasis)
ddx = makeA0(N,r,rzero);
ddx = FTWT2_P0(ddx,Lcoarse,h);
ddx = sparse(ddx);

% Materiaalmatrix in waveletbasis;
eenoverepsilon(1:N) = 1;
eenoverepsilon(3*N/4:N) = 1/4;
wmat = FTWT2_P0(diag(eenoverepsilon),Lcoarse,h);
wmat = sparse(wmat);

% combinatie materiaal- een d/dx matrix
wmatddx = wmat*ddx;

% Courant-Friedrichs-Levy getal
s = 0.9;
dz = 1;
c = 3e8;
dt = s*dz/c;

% beginwaarden instellen voor E- en H-veld
E0(1:N,1) = 0;
H0(1:N,1) = 0;
% gaussische puls definieren
E0 = makegaussian(N,32,0.01)';

% wavelettransformeren
wE0 = FWT_P0(E0,Lcoarse,h);
wH0 = FWT_P0(H0,Lcoarse,h);

% itereren
while 1
    % H berekenen op nieuw tijdstip
    wH1 = wH0 - s*ddx*wE0;
    % E berekenen op nieuw tijdstip
    wE1 = wE0 - s*wmatddx*wH1;
    % updaten
    wE0 = wE1;

```

```

    wH0 = wH1;

    % eventueel "filteren"
    wE0(N/2+1:N) = 0;
    hE0(N/2+1:N) = 0;

    % plotten
    E = IWT_PO(wE1,Lcoarse,h);
    H = IWT_PO(wH1,Lcoarse,h);
    subplot(211);
    plot(E);
    axis([1 N -1 1]);
    subplot(212);
    plot(H);
    axis([1 N -1 1]);
    drawnow;
end

```

- Tweedimensionale Werthen & Wolff methode (vacuüm).

```

function ww2d
% 2D Werthen&Wolff methode (zonder materiaaltensor)
% Dit mag geen script zijn (moet dus een functie zijn)
% omdat we sub-functies gebruiken

% aantal punten (= NxN)
N = 64;

% aantal vanishing moments
vmoments = 8;

% h-Filter
h = MakeOnFilter('Daubechies',2*vmoments);
Lcoarse = 3; % ontbinden tot er nog 2^3 punten over zijn

% Beylkin's r-coefficienten berekenen
r = ddxcoeff(h);
% index r_0
rzero = (length(r)-1)/2+1;

% Afgeleide matrix opstellen (in waveletbasis)
ddx = makeA0(N,r,rzero);
ddx = FTWT2_PO(ddx,Lcoarse,h);
ddx = sparse(ddx);

% getal s
s = 0.5;

```

```

dx = 1;
dy = 1;
c = 3e8;
dt = s*sqrt(dx^2+dy^2)/c;

% beginwaarden instellen voor E- en H-veld
E0x = zeros(N);
H0x = zeros(N);
E0y = zeros(N);
H0y = zeros(N);
E0z = zeros(N);
H0z = zeros(N);

% gaussische puls definieren
H0z = make2Dgaussian(N,N/8,0.04)';

% wavelettransformeren
wE0x = FTWT2_PO(E0x,Lcoarse,h);
wH0x = FTWT2_PO(H0x,Lcoarse,h);
wE0y = FTWT2_PO(E0y,Lcoarse,h);
wH0y = FTWT2_PO(H0y,Lcoarse,h);
wE0z = FTWT2_PO(E0z,Lcoarse,h);
wH0z = FTWT2_PO(H0z,Lcoarse,h);

% itereren
while 1
    % H berekenen op nieuw tijdstip
    wH1x = wH0x - s*apply_y(ddx,wE0z);
    wH1y = wH0y + s*apply_x(ddx,wE0z);
    wH1z = wH0z - s*(apply_x(ddx,wE0y) - apply_y(ddx,wE0x));

    % E berekenen op nieuw tijdstip
    wE1x = wE0x + s*apply_y(ddx,wH1z);
    wE1y = wE0y - s*apply_x(ddx,wH1z);
    wE1z = wE0z + s*(apply_x(ddx,wH1y) - apply_y(ddx,wH1x));

    % updaten
    wE0x = wE1x;
    wH0x = wH1x;
    wE0y = wE1y;
    wH0y = wH1y;
    wE0z = wE1z;
    wH0z = wH1z;

    % plotten
    Ex = ITWT2_PO(wE1x,Lcoarse,h);
    Ey = ITWT2_PO(wE1y,Lcoarse,h);
    Ez = ITWT2_PO(wE1z,Lcoarse,h);
    Hx = ITWT2_PO(wH1x,Lcoarse,h);

```



```

Hy = ITWT2_P0(wH1y,Lcoarse,h);
Hz = ITWT2_P0(wH1z,Lcoarse,h);
E = sqrt(Ex.^2 + Ey.^2 + Ez.^2);
H = sqrt(Hx.^2 + Hy.^2 + Hz.^2);
p = sum(sum(E.^2));

subplot(331);
pcolor(E);
shading interp;
title(sprintf('%d tijdsstappen', i))

subplot(332);
mesh(E);
title(sprintf('P~%d', p));

subplot(333);
pcolor(H);
shading interp;

subplot(334);
plot(Ex(:,N/2));
title('Ex');
axis([1 N -1 1]);

subplot(335);
plot(Ey(:,N/2));
title('Ey');
axis([1 N -1 1]);

subplot(336);
plot(Ez(:,N/2));
title('Ez');
axis([1 N -1 1]);

subplot(337);
plot(Hx(:,N/2));
title('Hx');
axis([1 N -1 1]);

subplot(338);
plot(Hy(:,N/2));
title('Hy');
axis([1 N -1 1]);

subplot(339);
plot(Hz(:,N/2));
title('Hz');
axis([1 N -1 1]);

```

```

    drawnow;
end

function result = apply_x(op,mat)
% Past een operator 'op' toe op alle rijen van de matrix 'mat'
result = zeros(size(mat));
for i = 1 : size(mat, 1)
    result(i,:) = (op*mat(i,:))';
end

function result = apply_y(op,mat)
% Past een operator 'op' toe op alle kolommen van de matrix 'mat'
result = zeros(size(mat));
for i = 1 : size(mat, 1)
    result(:,i) = op*mat(:,i);
end

```

- Tweedimensionale Werthen & Wolff methode met materiaaltensor.

```

function ww2d
% 2D Werthen&Wolff methode (met materiaaltensor)
% Dit mag geen script zijn (moet dus een functie zijn)
% omdat we sub-functies gebruiken

% aantal punten
N = 64;

% aantal vanishing moments
vmoments = 8;

% h-Filter
h = MakeOnFilter('Daubechies',2*vmoments);
Lcoarse = 3; % ontbinden tot er nog 2^3 punten over zijn

% Beylkin's r-coefficienten berekenen
r = ddxcoeff(h);
% index r_0
rzero = (length(r)-1)/2+1;

% Afgeleide matrix opstellen (in waveletbasis)
ddx = makeA0(N,r,rzero);
ddx = FTWT2_P0(ddx,Lcoarse,h);
ddx = sparse(ddx);

% Materiaaloperator opstellen (waveletbasis)
mat_matrix = ones(N);
mat_matrix(N/4*3:N,:) = 1/4.5;

```

```

mat_tensor = tensor_from_matrix(mat_matrix);
wmat_tensor = FTWT4_P0(mat_tensor,Lcoarse,h);

% getal s
s = 0.5;
dx = 1;
dy = 1;
c = 3e8;
dt = s*sqrt(dx^2+dy^2)/c;

% beginwaarden instellen voor E- en H-veld
E0x = zeros(N);
H0x = zeros(N);
E0y = zeros(N);
H0y = zeros(N);
E0z = zeros(N);
H0z = zeros(N);

% gaussische puls definieren
H0z = make2Dgaussian(N,N/8,0.01)';

% wavelettransformeren
wE0x = FTWT2_P0(E0x,Lcoarse,h);
wH0x = FTWT2_P0(H0x,Lcoarse,h);
wE0y = FTWT2_P0(E0y,Lcoarse,h);
wH0y = FTWT2_P0(H0y,Lcoarse,h);
wE0z = FTWT2_P0(E0z,Lcoarse,h);
wH0z = FTWT2_P0(H0z,Lcoarse,h);

% itereren
while 1
    % H berekenen op nieuw tijdstip
    wH1x = wH0x - s*apply_y(ddx,wE0z);
    wH1y = wH0y + s*apply_x(ddx,wE0z);
    wH1z = wH0z - s*(apply_x(ddx,wE0y) - apply_y(ddx,wE0x));

    % E berekenen op nieuw tijdstip
    wE1x = wE0x + s*tensorprodukt(wmat_tensor,apply_y(ddx,wH1z));
    wE1y = wE0y - s*tensorprodukt(wmat_tensor,apply_x(ddx,wH1z));
    wE1z = wE0z + s*tensorprodukt(wmat_tensor,apply_x(ddx,wH1y)
        - apply_y(ddx,wH1x));

    % updaten
    wE0x = wE1x;
    wH0x = wH1x;
    wE0y = wE1y;
    wH0y = wH1y;
    wE0z = wE1z;
    wH0z = wH1z;

```

```
% plotten
Ex = ITWT2_PO(wE1x,Lcoarse,h);
Ey = ITWT2_PO(wE1y,Lcoarse,h);
Ez = ITWT2_PO(wE1z,Lcoarse,h);
Hx = ITWT2_PO(wH1x,Lcoarse,h);
Hy = ITWT2_PO(wH1y,Lcoarse,h);
Hz = ITWT2_PO(wH1z,Lcoarse,h);
E = sqrt(Ex.^2 + Ey.^2 + Ez.^2);

subplot(331);
pcolor(E);
shading interp;

subplot(332);
mesh(E);

subplot(334);
plot(Ex(:,N/2));
title('Ex');
axis([1 N -1 1]);

subplot(335);
plot(Ey(:,N/2));
title('Ey');
axis([1 N -1 1]);

subplot(336);
plot(Ez(:,N/2));
title('Ez');
axis([1 N -1 1]);

subplot(337);
plot(Hx(:,N/2));
title('Hx');
axis([1 N -1 1]);

subplot(338);
plot(Hy(:,N/2));
title('Hy');
axis([1 N -1 1]);

subplot(339);
plot(Hz(:,N/2));
title('Hz');
axis([1 N -1 1]);

drawnow;
end
```

```

function result = apply_x(op,mat)
% Past een operator 'op' toe op alle rijen van de matrix 'mat'
result = zeros(size(mat));
for i = 1 : size(mat, 1)
    result(i,:) = (op*mat(i,:)')';
end

function result = apply_y(op,mat)
% Past een operator 'op' toe op alle kolommen van de matrix 'mat'
result = zeros(size(mat));
for i = 1 : size(mat, 1)
    result(:,i) = op*mat(:,i);
end

```

- Tweedimensionale Werthen & Wolff methode die enkel de E_x , E_y en H_z veldcomponenten gebruikt.

```

function ww2d
% 2D Werthen&Wolff methode (zonder materiaaltensor)
% Dit mag geen script zijn (moet dus een functie zijn)
% omdat we sub-functies gebruiken
%
% Deze versie gebruikt enkel Ex,Ey en Hz en is dus
% sneller

% aantal punten
N = 64;

% aantal vanishing moments
vmoments = 8;

% h-Filter
h = MakeOnFilter('Daubechies',2*vmoments);
Lcoarse = 1; % ontbinden tot er nog 2^3 punten over zijn

% Beylkin's r-coefficienten berekenen
r = ddxcoeff(h);
% index r_0
rzero = (length(r)-1)/2+1;

% Afgeleide matrix opstellen (in waveletbasis)
ddx = makeA0(N,r,rzero);
ddx = FTWT2_P0(ddx,Lcoarse,h);
ddx = sparse(ddx);

% getal s

```

```

s = 0.5;
dx = 1;
dy = 1;
c = 3e8;
dt = s*sqrt(dx^2+dy^2)/c;

% beginwaarden instellen voor E- en H-veld
E0x = zeros(N);
E0y = zeros(N);
H0z = zeros(N);

% gaussische puls definieren
H0z = make2Dgaussian(N,N/8,0.04)';

% wavelettransformeren
wE0x = FTWT2_PO(E0x,Lcoarse,h);
wE0y = FTWT2_PO(E0y,Lcoarse,h);
wH0z = FTWT2_PO(H0z,Lcoarse,h);

% itereren
while 1
    % H berekenen op nieuw tijdstip
    wH1z = wH0z - s*(apply_x(ddx,wE0y) - apply_y(ddx,wE0x));

    % E berekenen op nieuw tijdstip
    wE1x = wE0x + s*apply_y(ddx,wH1z);
    wE1y = wE0y - s*apply_x(ddx,wH1z);

    % updaten
    wE0x = wE1x;
    wE0y = wE1y;
    wH0z = wH1z;

    % plotten
    Ex = ITWT2_PO(wE1x,Lcoarse,h);
    Ey = ITWT2_PO(wE1y,Lcoarse,h);
    Hz = ITWT2_PO(wH1z,Lcoarse,h);
    E = sqrt(Ex.^2 + Ey.^2);
    H = sqrt(Hz.^2);

    subplot(331);
    pcolor(E);
    shading interp;
    title(sprintf('%d\tijdsstappen', i))

    subplot(332);
    mesh(E);

    subplot(333);

```

```

    pcolor(H);
    shading interp;

    subplot(334);
    plot(Ex(:,N/2));
    title('Ex');
    axis([1 N -1 1]);

    subplot(335);
    plot(Ey(:,N/2));
    title('Ey');
    axis([1 N -1 1]);

    subplot(339);
    plot(Hz(:,N/2));
    title('Hz');
    axis([1 N -1 1]);

    drawnow;
end

function result = apply_x(op,mat)
% Past een operator 'op' toe op alle rijen van de matrix 'mat'
result = zeros(size(mat));
for i = 1 : size(mat, 1)
    result(i,:) = (op*mat(i,:))';
end

function result = apply_y(op,mat)
% Past een operator 'op' toe op alle kolommen van de matrix 'mat'
result = zeros(size(mat));
for i = 1 : size(mat, 1)
    result(:,i) = op*mat(:,i);
end
end

```

- Functie die Beylkin's r -coëfficiënten berekent door het stelsel (4.40)-(4.41) op te lossen.

```

function result = ddxcoeff(h);
% function result = ddxcoeff(h);
%
% Berekent de coëfficiënten  $r_1$ , nodig voor het
% opstellen van de  $d/dx$  operatormatrix in waveletbasis
% (zie Beylkin).
%
%  $h$  =  $h$ -filter coëfficiënten
% (opstellen met Wavlab functie MakeOnFilter)

```

```

% result = array met de r-coefficienten
%         Index van element nr. 0 is
%         (length(result)-1)/2+1
%
% lengte filter
L = length(h);

% de autocorrelatiecoeff. van h berekenen
a(1:L-1) = 0;
for n = 1 : L-1
    for i = 1 : L-n
        a(n) = a(n) + h(i)*h(i+n);
    end
end
a=2*a;

% stelsel RX=values opstellen
R = zeros(2*L-4+1);
Rzero = (2*L-4)/2+1;
N = 2*L-4+1;
for l = -L+2 : L-2
    R(1+Rzero,l+Rzero)=-1;
    if 2*l+Rzero <= N & 2*l+Rzero > 0
        R(1+Rzero,2*l+Rzero) = R(1+Rzero,2*l+Rzero) + 2;
    end
    for k = 1 : L/2
        if 2*l-2*k+1+Rzero <= N & 2*l-2*k+1+Rzero > 0
            R(1+Rzero,2*l-2*k+1+Rzero) = R(1+Rzero,2*l-2*k+1+Rzero) + a(2*k-1);
        end
        if 2*l+2*k-1+Rzero <= N & 2*l+2*k-1+Rzero > 0
            R(1+Rzero,2*l+2*k-1+Rzero) = R(1+Rzero,2*l+2*k-1+Rzero) + a(2*k-1);
        end
    end
end
end
for i = 1 : N
    R(N+1,i) = i-Rzero;
end
values = zeros(N+1,1);
values(N+1,1)=-1;

% stelsel oplossen (N+1 vgln., N onbek.)
result = (R\values)';

```

- Functie die de matrix $\{r_{il}^0\}$ opstelt (cf. 4.2.5).

```
function A0 = makeA0(N, r, rzero);
```



```

% function A0 = makeA0(N, r, rzero)
%
% Maakt de matrix A0 van de afgeleide operator in
% waveletbasis.
%
% N: grootte matrix (macht van 2!)
% r: array met r-coefficiënten (zie papers Beylkin)
% rzero: elementnummer van r met index 0
%
A0 = matrix_from_stencil(-r,rzero, N);

```

- Functie die een circulaire matrix maakt aan de hand van een 'stencil'.

```

function A=matrix_from_stencil(stencil, nulelem, N)
% function A=matrix_from_stencil(stencil, nulelem, N)
%
% Genereert een NxN matrix door verhuivingen van een stencil.
% De index van het nulde element van stencil wordt gegeven
% door 'nulelem'.

A=zeros(N);
s=zeros(1,N);
s(1:length(stencil)) = stencil;
for j = 0 : N-1
    index = mod(nulelem-1-j,N)+1;
    A(j+1, 1 : (N - index)+1) = s(index : N);
    A(j+1, (N - index)+2 : N) = s(1 : index-1);
end

%Werkt ook, maar veel trager:
%for j = 0 : N-1
%    for i = 0 : N-1
%        A(j+1,i+1) = s(mod(i+nulelem-1-j,N)+1);
%    end
%end

```

- Functie die een gemoduleerde Gaussische puls maakt.

```

function g = makegaussian(N,freq,width);
% function g = makegaussian(N,freq,width);
%
% Maakt gaussische bundel (1D)
% N: aantal punten (even!)
% freq: aantal oscillaties van de carrier binnen N punten

```

```

% width: breedte van de puls. Hoe hoger hoe breder.
%         typische waarde: 0.05
%
% carrier maken (= 1 op punt N/2)
carrier(1:N+1) = -freq*pi : freq*2*pi/N : freq*pi;
carrier = cos(carrier);
carrier = carrier(2:N+1);
%
% gaussiaan maken (= 1 op punt N/2)
y = -pi/2 : pi/N : pi/2;
z = exp(-y.*y./(pi*width));
z = z(2:N+1);
%
% produkt van beide geeft bundel met gaussische envelope
g = z.*carrier;

```

- Functie die een tweedimensionale Gaussische puls teruggeeft. De puls is gemoduleerd volgens de y -richting (kolom-richting).

```

function g = make2Dgaussian(N,freq,width);
% function g = make2Dgaussian(N,freq,width);
%
% Maakt gaussische bundel (2D)
% N: er wordt een NxN gaussiaan aangemaakt (N = even!)
% freq: aantal oscillaties van de carrier binnen N punten
% width: breedte van de puls. Hoe hoger hoe breder.
%         typische waarde: 0.05
%
% Opm.: plotten met bv. meshc(g), of met pcolor(g); shading interp;
%
% carrier maken (= 1 op punt N/2)
carrier(1:N+1) = -freq*pi : freq*2*pi/N : freq*pi;
[X,Y] = meshgrid(carrier);
carrier = cos(X);
carrier = carrier(2:N+1,2:N+1);
%
% gaussiaan maken (= 1 op punt N/2)
y = -pi/2 : pi/N : pi/2;
[X,Y] = meshgrid(y);
z = exp(-(X.^2 + Y.^2)./(pi*width));
z = z(2:N+1,2:N+1);
%
% produkt van beide geeft bundel met gaussische envelope
g = z.*carrier;

```

- Functie die aan de hand van een matrix een vierdimensionale operator-tensor opstelt waarmee puntsgewijze vermenigvuldigingen kunnen uitgevoerd worden (cf. 4.4).

```
function R = tensor_from_matrix(A);
% function R = tensor_from_matrix(A);
%
% Maakt een 4D-tensor R aan de hand van de
% matrix A, zodanig dat 'tensorprodukt(R,B)'
% een matrix geeft gelijk aan de puntsgewijze
% vermenigvuldiging van A en B.
%
% A moet NxN zijn
%

N = size(A,1);
R = zeros(N,N,N,N);

for i = 1 : N
    for j = 1 : N
        R(i,i,j,j) = A(i,j);
    end
end
```

- Functie die het produkt van een vierdimensionale tensor met een matrix berekent d.m.v. formule (4.61).

```
function R = tensorprodukt(A,B)
% function result = tensorprodukt(A,B)
%
% Berekent het tensor-inprodukt tussen de
% 4D tensor A en de 2D tensor B.
%
% Formule:
% R(i1,i2) = Som_over_j1_j2_van(A(i1,j1,i2,j2)*B(j1,j2))
%
% Voor de eenvoud van implementatie wordt gesteld
% dat A NxNxNxN is en B NxN.
%

N = size(B,1);
R = zeros(N);
Btrans = sparse(B');

for i = 1 : N
    for j = 1 : N
```

```

        R(i,j) = trace( sparse(squeeze(A(i, :, j, :))) * Btrans );
    end
end

```

- Functie om ééndimensionale Bragg reflectoren te maken.

```

function bragg = make1Dbraggmirror(n1, n2, ppp, npairs)
% function make1Dbraggmirror(n1, n2, ppp, nperiods)
%
% Maakt 1D bragg spiegel - geeft array van 1/epsilon
% waarden terug.
%
% n1,n2: brekingsindices van de spiegellaagjes
% ppp (points per period): aantal punten per periode
%                of golflengte (de design
%                golflengte van de bragg
%                spiegel)
% npairs: aantal lagenparen (n1,n2).
%

l1 = round(ppp/(4*n1));
l2 = round(ppp/(4*n2));
pointer = 1;
for i = 1 : npairs
    bragg(pointer : pointer + l1 - 1) = n1;
    pointer = pointer + l1;
    bragg(pointer : pointer + l2 - 1) = n2;
    pointer = pointer + l2;
end

bragg = (1./bragg).^2;

```

- Functie om de wavelettransformatie van een vierdimensionale tensor te berekenen (in standaard tensor-produkt basis, cf. 4.1.1).

```

function wc = FTWT4_PO(x,L,qmf)
% FTWT4_PO -- 4-d tensor wavelet transform (periodized, orthogonal)
% Usage
%   wc = FTWT4_PO(x,L,qmf)
% Inputs
%   x      4-d tensor (n by n by n by n array, n dyadic)
%   L      coarsest level
%   qmf    quadrature mirror filter
% Outputs
%   wc     2-d wavelet transform

```

```

%
% Description
%   A two-dimensional Wavelet Transform is computed for the
%   array x. To reconstruct, use ITWT4_PO.
%
% See Also
%   ITWT4_PO, MakeONFilter
%

[n,J] = quadlength(x);

for i = 1 : n
    for j = 1 : n
        for k = 1 : n
            row = squeeze(x(i,j,k,:));
            wrow = FWT_PO(row,L,qmf);
            x(i,j,k,:) = wrow;
        end
    end
end

for i = 1 : n
    for j = 1 : n
        for k = 1 : n
            row = squeeze(x(i,j,:,k));
            wrow = FWT_PO(row,L,qmf);
            x(i,j,:,k) = wrow;
        end
    end
end

for i = 1 : n
    for j = 1 : n
        for k = 1 : n
            row = squeeze(x(i,:,j,k));
            wrow = FWT_PO(row,L,qmf);
            x(i,:,j,k) = wrow;
        end
    end
end

for i = 1 : n
    for j = 1 : n
        for k = 1 : n
            row = squeeze(x(:,i,j,k));
            wrow = FWT_PO(row,L,qmf);
            x(:,i,j,k) = wrow;
        end
    end
end

```

```

end

wc = x;

```

Om bovenstaande code te kunnen begrijpen is nog informatie nodig over de gebruikte WaveLab functies:

- Functie die de filtercoëfficiënten van verschillende wavelets teruggeeft.

```

function f = MakeONFilter(Type,Par)
% MakeONFilter -- Generate Orthonormal QMF Filter for Wavelet Transform
% Usage
%   qmf = MakeONFilter(Type,Par)
% Inputs
%   Type    string, 'Haar', 'Beylkin', 'Coiflet', 'Daubechies',
%           'Symmlet', 'Vaidyanathan','Battle'
%   Par     integer, it is a parameter related to the support and vanishing
%           moments of the wavelets, explained below for each wavelet.
%
% Outputs
%   qmf     quadrature mirror filter
%
% Description
%   The Haar filter (which could be considered a Daubechies-2) was the
%   first wavelet, though not called as such, and is discontinuous.
%
%   The Beylkin filter places roots for the frequency response function
%   close to the Nyquist frequency on the real axis.
%
%   The Coiflet filters are designed to give both the mother and father
%   wavelets 2*Par vanishing moments; here Par may be one of 1,2,3,4 or 5.
%
%   The Daubechies filters are minimal phase filters that generate wavelets
%   which have a minimal support for a given number of vanishing moments.
%   They are indexed by their length, Par, which may be one of
%   4,6,8,10,12,14,16,18 or 20. The number of vanishing moments is par/2.
%
%   Symmlets are also wavelets within a minimum size support for a given
%   number of vanishing moments, but they are as symmetrical as possible,
%   as opposed to the Daubechies filters which are highly asymmetrical.
%   They are indexed by Par, which specifies the number of vanishing
%   moments and is equal to half the size of the support. It ranges
%   from 4 to 10.
%
%   The Vaidyanathan filter gives an exact reconstruction, but does not
%   satisfy any moment condition. The filter has been optimized for

```

```

%   speech coding.
%
%   The Battle-Lemarie filter generate spline orthogonal wavelet basis.
%   The parameter Par gives the degree of the spline. The number of
%   vanishing moments is Par+1.
%
% See Also
%   FWT_PO, IWT_PO, FWT2_PO, IWT2_PO, WPAnalysis
%
% References
%   The books by Daubechies and Wickerhauser.
%

```

- Functie die een ééndimensionale voorwaartse wavelettransformatie uitvoert.

```

function wcoef = FWT_PO(x,L,qmf)
% FWT_PO -- Forward Wavelet Transform (periodized, orthogonal)
% Usage
%   wc = FWT_PO(x,L,qmf)
% Inputs
%   x    1-d signal; length(x) = 2^J
%   L    Coarsest Level of V_0; L << J
%   qmf  quadrature mirror filter (orthonormal)
% Outputs
%   wc   1-d wavelet transform of x.
%
% Description
%   1. qmf filter may be obtained from MakeONFilter
%   2. usually, length(qmf) < 2^(L+1)
%   3. To reconstruct use IWT_PO
%
% See Also
%   IWT_PO, MakeONFilter
%

```

- Functie die een tweedimensionale voorwaartse wavelettransformatie uitvoert. Ze gebruikt de standaard tensor-produkt basis (cf. 4.1.1).

```

function wc = FTWT2_PO(x,L,qmf)
% FTWT2_PO -- 2-d tensor wavelet transform (periodized, orthogonal)
% Usage
%   wc = FTWT2_PO(x,L,qmf)
% Inputs
%   x    2-d image (n by n array, n dyadic)
%   L    coarse level

```

```

%   qmf   quadrature mirror filter
%   Outputs
%   wc    2-d wavelet transform
%
%   Description
%   A two-dimensional Wavelet Transform is computed for the
%   array x. To reconstruct, use ITWT2_PO.
%
%   See Also
%   ITWT2_PO, MakeONFilter
%

```

- Functie die een tweedimensionale inverse wavelettransformatie uitvoert (omgekeerde van FTWT2_PO)

```

function x = IWT2_PO(wc,L,qmf)
% IWT2_PO -- Inverse 2-d tensor wavelet transform (periodized, orthogonal)
% Usage
%   x = IWT2_PO(wc,L,qmf)
% Inputs
%   wc    2-d wavelet transform [n by n array, n dyadic]
%   L     coarse level
%   qmf   quadrature mirror filter
% Outputs
%   x     2-d signal reconstructed from wc
%
% Description
%   If wc is the result of a forward 2d wavelet transform, with
%   wc = FTWT2_PO(x,L,qmf), then x = IWT2_PO(wc,L,qmf) reconstructs x
%   exactly if qmf is a nice qmf, e.g. one made by MakeONFilter.
%
% See Also
%   FTWT2_PO, MakeONFilter
%

```

- Functie die teruggeeft aan welke macht van 2 de afmetingen van een vierkante matrix gelijk zijn.

```

function [n,J] = quadlength(x)
% quadlength -- Find length and dyadic length of square matrix
% Usage
%   [n,J] = quadlength(x)
% Inputs
%   x     2-d image; size(n,n), n = 2^J (hopefully)
% Outputs

```



```
%      n      length(x)
%      J      least power of two greater than n
%
% Side Effects
%      A warning message is issue if n is not a power of 2,
%      or if x is not a square matrix.
%
```

Bijlage B

Afkortingen

ABC	Absorbing boundary condition
DBR	Gedistribueerde Bragg reflector (Distributed Bragg reflector)
EEL	Edge emitting laser
FDTD	Finite difference time domain
MRA	Multiresolutie analyse
MRTD	Multiresolution time domain
VCSEL	Vertical cavity surface emitting laser

Bijlage C

Engelstalige termen

Absorbing boundary condition	Randvoorwaarde die toelaat dat straling het simulatiedomein verlaat.
Compactly supported	Met compacte drager
Current confinement	Ruimtelijke opsluiting van de stroom
Decay	Verval
Grid	Rooster
Mesh	Rooster
Optical confinement	Ruimtelijke opsluiting van licht
Partition of unity	Partitie van de eenheid
Refinement equation	Verfijningsvergelijking
Regularity	Regelmatigheid
Separable	Separabel
Smooth	Glad
Support	Draagwijdte
Thresholding	Het nul maken van coëfficiënten kleiner dan een bepaalde waarde
Vanishing moments	Momenten die nul zijn, nulmomenten

Bijlage D

Notaties

$\langle f, g \rangle$	Inwendig produkt, $\int_{-\infty}^{+\infty} f(x) \overline{g(x)} dx$
$\ f\ $	Norm, $\ f\ ^2 = \int_{-\infty}^{+\infty} f(x) ^2 dx$
$L^2(\mathbb{R})$	Ruimte van kwadratisch integreerbare functies, $\int_{-\infty}^{+\infty} f(x) ^2 dx < +\infty$
$L^1(\mathbb{R})$	Ruimte van absoluut integreerbare functies, $\int_{-\infty}^{+\infty} f(x) dx < +\infty$
$U \oplus V$	Directe som van twee vectorruimten
$U \otimes V$	Tensorprodukt van twee vectorruimten
$\hat{f}(\omega)$	Fouriertransformatie, $\hat{f}(\omega) = \int_{-\infty}^{+\infty} f(t) e^{-i\omega t} dt$
$\widetilde{\mathcal{W}}f(u, \omega)$	Gevensterde Fouriertransformatie, zie (3.7)
$\mathcal{W}f(s, u)$	Continue wavelettransformatie, zie (3.8)
$d_{j,k}$	Detail- of waveletcoëfficiënt in de waveletexpansie van f , $d_{j,k} = \langle f, \psi_{j,k} \rangle$
$s_{j,k}$	Schaal- of schalingcoëfficiënt in de waveletexpansie van f , $s_{j,k} = \langle f, \varphi_{j,k} \rangle$

Bibliografie

- [1] Kenichi Iga, "Surface-Emitting Laser - Its Birth and Generation of New Optoelectronics Field," *IEEE Journal of Selected Topics in Quantum Electronics*, vol. 6, pp. 1201–1215, Nov. 2000.
- [2] Connie J. Chang-Hasnain, "Tunable VCSEL," *IEEE Journal of Selected Topics in Quantum Electronics*, vol. 6, pp. 978–987, Nov. 2000.
- [3] A. Karim, P. Abraham, D. Lofgreen, Y.J. Chiu, J. Piprek, and J. Bowers, "Wafer-bonded 1.55 μm vertical cavity laser arrays for wavelength division multiplexing," *Electronics Letters*, vol. 37, pp. 431–432, 29 Mar. 2001.
- [4] G. Hadley, K. Lear, W. Warren, K. Choquette, J. Scott, and S. Corzine, "Comprehensive numerical model for cw vertical-cavity surface-emitting lasers," in *Physics and Simulation of Optoelectronic Devices III*, vol. 2399 of *Proceedings SPIE*, pp. 336–347, 1995.
- [5] J. Berenger, "A perfectly matched layer for the absorption of electromagnetic-waves," *Journal of Computational Physics*, vol. 114, no. 2, pp. 185–200, 1994.
- [6] G. Mur, "Total-field absorbing boundary conditions for the time-domain electromagnetic field equations," *IEEE Transactions on Electromagnetic Compatibility*, vol. 40, no. 2, pp. 100–102, 1998.
- [7] G. Van Der Sande, "Analyse en synthese van theoretische modellen voor polarisatieschakelen in vertical cavity surface emitting lasers," Master's thesis, Vrije Universiteit Brussel, 2001.
- [8] G. Hadley, "Effective index model for vertical-cavity surface-emitting lasers," *Optics Letters*, vol. 20, pp. 1482–1485, 1995.
- [9] M. Nobel, J. Loehr, and J. Lott, "Analysis of microcavity vcsel lasing modes using a full-vector weighted index method," *IEEE Journal of Quantum Electronics*, vol. 34, no. 10, pp. 1890–1903, 1998.

- [10] P. Bienstman and R. Baets, "Optical modelling of photonic crystals and vcsels using eignemode expansion and perfectly matched layers," *Optics and Quantum Electronics*, 2001. to be published.
- [11] G. Bava, P. Debernardi, and L. Fratta, "Three-dimensional model for vectorial fields in vertical-cavity surface-emitting lasers," *Physical Review A.*, vol. 63, p. 23816, 2001.
- [12] E. Ahlers, S. Helfert, and R. Pregla, "Modeling of vcsels by the method of lines," in *Proceedings on Integrated Photonics Research Technical Digest*, vol. 6, pp. 340–343, 1996.
- [13] B. Klein, L. Register, K. Hess, D. Deppe, and Q. Deng, "Self-consistent green's function approach to the analysis of dielectrically apertured vertical-cavity surface-emitting lasers," *Applied Physics Letters*, vol. 73, no. 23, pp. 3324–3326, 1998.
- [14] R. Lemeire, "Bijdrage tot de numerieke beschrijving van vertical cavity surface emitting lasers," Master's thesis, Vrije Universiteit Brussel, 1998.
- [15] G. Hadley, "Finite difference time domain methods on a triangular grid for vcsel's," in *Proceedings of the workshop on Numerical Modelling*, (Paderborn, Germany), April 2001.
- [16] Stéphane Mallat, *A Wavelet Tour of Signal Processing*. Academic Press, second ed., 1999.
- [17] Gerald Kaiser, *A Friendly Guide to Wavelets*. Birkhäuser, 1994.
- [18] Ingrid C. Daubechies and Anna C. Gilbert, *Harmonic Analysis, Wavelets and Applications*. Graduate Summer School Lectures, American Mathematical Society, June 1995.
- [19] Björn Jawerth and Wim Sweldens, "An overview of wavelet based multiresolution analyses," *SIAM rev.*, vol. 36, pp. 377–412, 1994.
- [20] G. Beylkin, R. Coifman, and V. Rokhlin, "Fast wavelet transforms and numerical algorithms I," *Comm. Pure Appl. Math.*, vol. 44, pp. 141–183, 1991.
- [21] G. Beylkin, "Wavelets and Fast Numerical Algorithms." Lecture notes for short course, AMS-93, 1993.
- [22] Mihai Dorobantu, *Wavelet-based Algorithms for Fast PDE Solvers*. PhD thesis, Royal Institute of Technology, Stockholm, Sweden, Sept. 1995.
- [23] G. Beylkin, J. Dunn, and D. Gines, "Order N Static and Quasi-Static Computations in Electromagnetics using Wavelets," PAM report 189, 1994.

- [24] G. Beylkin, "On the representation of operators in bases of compactly supported wavelets," *SIAM Journal on Numerical Analysis*, vol. 29, no. 6, pp. 1716–1740, 1992.
- [25] G. Beylkin, "Wavelets, Multiresolution Analysis and Fast Numerical Algorithms." A Draft of INRIA lectures, May 1991.
- [26] L. Jameson, "On the daubechies-based wavelet differentiation matrix," tech. rep., Institute for Computer Applications in Science and Engineering, NASA Langley Research Center, Hampton, VA, 1993.
- [27] G. Beylkin, "On the fast algorithm for multiplication of functions in wavelet bases," PAM report 152, 1992.
- [28] M. Werthen and I. Wolff, "A Novel Wavelet Based Time Domain Simulation Approach," *IEEE Microwave and guided wave letters*, vol. 6, pp. 438–440, Dec. 1996.
- [29] R. L. Wagner and W. C. Chew, "A study of wavelets for the solution of electromagnetic integral equations," *IEEE Transactions on Antennas and Propagation*, vol. 43, pp. 802–810, Aug. 1995.
- [30] Zhonggui Xiang and Yilong Lu, "An Effective Wavelet Matrix Transform Approach for Efficient Solutions of Electromagnetic Integral Equations," *IEEE Transactions on Antennas and Propagation*, vol. 45, pp. 1205–1213, Aug. 1998.
- [31] Ning Guan, Ken'ichiro Yashiro, and Sumio Ohkawa, "On a Choice of Wavelet Bases in the Wavelet Transform Approach," *IEEE Transactions on Antennas and Propagation*, vol. 48, pp. 1186–1190, Aug. 2000.
- [32] Michael Paulus and Oliver J. F. Martin, "Light propagation and scattering in stratified media: a Green's tensor approach," *Journal of the Optical Society of America A*, vol. 18, pp. 854–861, Apr. 2001.
- [33] Kane S. Yee, "Numerical solution of Initial Boundary Value Problems Involving Maxwell's Equations in Isotropic Media," *IEEE Transactions on Antennas and Propagation*, vol. AP-14, pp. 302–307, May 1966.
- [34] John B. Schneider and Robert J. Kruhlak, "Dispersion of Homogeneous and Inhomogeneous Waves in the Yee Finite-Difference Tim-Domain Grid," *IEEE Transactions on Microwave Theory and Techniques*, vol. 49, pp. 280–287, Feb. 2001.
- [35] Takefumi Namiki, "3-D ADI-FDTD Method - Unconditionally Stable Time-Domain Algorithm for Solving Full Vector Maxwell's Equations," *IEEE Transactions on Microwave Theory and Techniques*, vol. 48, pp. 1743–1748, Oct. 2000.

- [36] Cynthia M. Furse, Daniel H. Roper, Dale N. Buechler, Douglas A. Christensen, and Carl H. Durney, "The Problem and Treatment of DC Offsets in FDTD Simulations," *IEEE Transactions on Antennas and Propagation*, vol. 48, pp. 1198–1201, Aug. 2000.
- [37] José A. Pereda, Luis A. Vielva, Angel Vegas, and Andrés Prieto, "Analyzing the Stability of the FDTD Technique by Combining the von Neumann Method with the Routh-Hurwitz Criterion," *IEEE Transactions on Microwave Theory and Techniques*, vol. 49, pp. 377–381, Feb. 2001.
- [38] Michael Krumpholz and Linda P.B. Katehi, "MRTD: New Time-Domain Schemes Based on Multiresolution Analysis," *IEEE Transactions on Microwave Theory and Techniques*, vol. 44, pp. 555–571, Apr. 1996.
- [39] C.D. Sarris, L.P.B. Katehi, and J.F. Harvey, "Application of multiresolution analysis to the modeling of microwave and optical structures," *Optical and Quantum Electronics*, vol. 32, pp. 657–679, 2000.
- [40] Guillaume Carat, Raphaël Gillard, Jacques Citerne, and Joe Wiart, "An Efficient Analysis of Planar Microwave Circuits Using A DWT-Based HAAR MRTD Scheme," *IEEE Transactions on Microwave Theory and Techniques*, vol. 48, pp. 2261–2270, Dec. 2000.
- [41] Gerrit Mur, "Absorbing Boundary Conditions for the Finite-Difference Approximation of the Time-Domain Electromagnetic-Field Equations," *IEEE Transactions on Electromagnetic Compatibility*, vol. EMC-23, pp. 377–382, Nov. 1981.
- [42] Anna C. Gilbert, "A comparison of classical and multiresolution one-dimensional homogenization schemes," *Applied and Computational Harmonic Analysis*, vol. 5, pp. 1–35, 1998.
- [43] M.E. Brewster and G. Beylkin, "A Multiresolution strategy for numerical homogenization," *Applied and Computational Harmonic Analysis*, vol. 2, pp. 327–349, 1995.