# Optimization of biorthogonal wavelet filters for signal and image compression

Jabran Akhtar

February 2001

# Preface

This text is submitted as the required written part in partial fulfillment for the degree of Candidatus Scientarium at the Department of Informatics, University of Oslo.

The work was carried out from January 2000 till February 2001. The goal was to examine some properties of biorthogonal wavelet filters and to construct optimized wavelet filters for a given signal or image.

Professor Nils D. Christophersen and Associate Professor Hermann Lia have been the supervisors.

## Acknowledgements

All of the work in this study was performed using Matlab (http://www.mathworks.com) and the Matlab Optimization Toolbox. For wavelet specific work the Uvi Wave package was utilized (ftp.tsc.uvigo.es://pub/Uvi_Wave/matlab/).

I would also like to thank my supervisors for guiding me through this study.

# Contents

# Chapter 1

# Introduction

Over the last years the usage of wavelet transforms has increased many folds for different applications in signal and image processing. Wavelet techniques provide a new approach to many problems where one can combine time and frequency domain processing to some level. These ideas have made wavelets very suitable, especially, for image compression. Methods to design and to find the "best" wavelet filters for compacting purposes have been an active area of research and wavelets have also become the preferred choice for modern image compression. Compression of images have important applications in for example storage and transmission, and a wavelet transform of an image is generally the first step towards this compaction process.

Typical methods to design wavelet filters focus heavily upon the wavelet functions and their properties such as regularity, smoothness and symmetricalness. The purpose behind this study is to take a distinctive view of the biorthogonal wavelet filter design process, and experiments with new types of wavelet filters which are optimized for a particular signal or image. The word "best" or "optimized" can be interpreted in many different ways; our concentration will be upon the energy shifting properties of wavelet filters.

This text is divided into 8 chapters. Chapter 2 gives some insight into wavelet theory from a signal processing point of view. Today, many books are available on wavelets and this chapter obviously only covers a few general aspects.
The main consideration of chapter 3 is on selected orthogonal and biorthogonal wavelet filters which are used in signal and image processing. We examine these filters' energy shifting properties.
Starting with chapter 4 emphasis is given to the construction of optimized linear phase perfect reconstruction filters for a given one dimensional signal. This concept is further treated in chapter 5 where methods to increase stability are introduced.
Chapter 6 then expands the filter design optimization problem to the more interesting cases of two dimensional images. At the end, in chapter 7 we look at generating optimized filters for iterative decomposition of an image.

# Chapter 2

# Wavelets

The purpose of this chapter is to provide an intuitive understanding of what wavelets are and why they are used in signal and image processing. Generally [3], [5], [6] and [15] have been used as the main reference sources. Much of the theory behind wavelet analysis and filter banks is very well established and for more details and proofs the above references should be consulted.

## 2.1 Wavelet functions

Wavelets are continuous basis functions designed specially to also satisfy certain mathematical properties. A wavelet is defined as a function with zero mean value, hence it is both above and below the x-axis looking like a 'wave'. Some examples of wavelet functions are shown in the next figure.

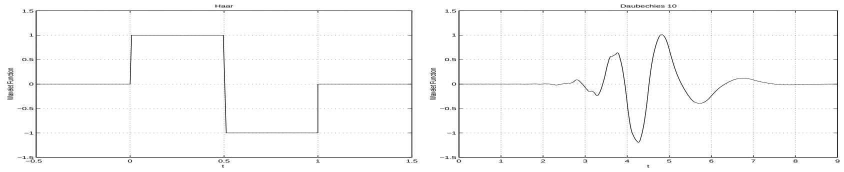

Figure 2.1: Example of wavelet functions

The function on left often goes under the name of Haar wavelet while the one on right is a variant of Daubechies wavelets. Wavelet functions can be of many other shapes as long as they have compact support. It is also straightforward to observe that the mean value is indeed zero; $\int_{-\infty}^{\infty} \psi(t)\ dt = 0$.

The idea behind wavelets is that by squeezing or stretching one such main wavelet function $\psi(t)$ (the mother wavelet) and by translating it on the x-axis one can represent finer parts of a function or a signal by simple linear combinations: $f(t) = \sum_{j,k}\ b_{j,k}\ \psi_{j,k}(t)$

The concept is similar to Fourier transform of a signal where a function is represented via trigonometrical basis functions; in wavelets analysis however, the functions are restricted in the time-domain. A whole family of wavelet functions can further be obtained by just shifting and scaling the main wavelet by:

$$\psi_{j,k}(t) = 2^{j/2}\psi(2^j t - k)$$

The normalization this way ensures energy preservation $||\psi_{j,k}(t)|| = ||\psi(t)||$, where one expects $||\psi(t)||^2 = \int_{-\infty}^{\infty} \psi(t)^2 \; dt = 1$. There are also other ways to define dilations and translations such as $\psi_{u,s}(t) = \frac{1}{\sqrt{|s|}}\psi(\frac{t-u}{s})$ where $u$ and $s$ are real numbers, though our first definition is more suitable for later expansion into discrete forms. We have so far assumed that $j$ and $k$ have been real integers and will continue to assume so also for $l$, $J$ and $K$.

Increasing $j$ dilates the wavelet by compressing it and more finer details of functions can then be represented accurately. The wavelets from figure 2.1 are shown dilated and shifted by one level in figure 2.2. $\psi_{1,0}(t)$ is on the left side on the plots while the translated wavelet $\psi_{1,1}(t)$ is on right in dashed print.

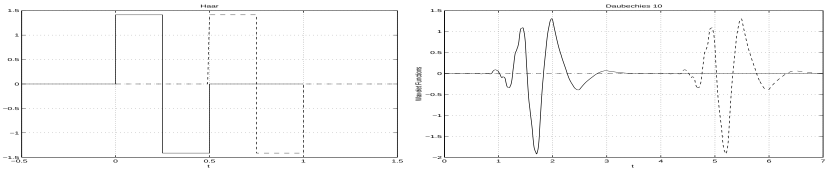

Figure 2.2: Example of dilated and translated wavelet functions

Another important property which is satisfied by many wavelets is that of orthogonality. The inner product between dilated and translated wavelet functions must be zero:

$$\int_{-\infty}^{\infty} \psi_{j,k}(t) \; \psi_{J,K}(t) \; dt = \delta(k - K) \; \delta(j - J)$$

A continuous wavelet transform of $f(t)$ at scale $j$ and position $k$ is defined as

$$F(j,k) = \int_{-\infty}^{\infty} f(t) \; \psi^*_{j,k}(t) \; dt$$

This equation shows how a function $f(t)$ is decomposed into a set of basis wavelet functions $\psi_{j,k}$. This may looks like a very redundant transform as it expands a one-dimensional function into two-dimensions. Nevertheless, starting with dilations and translations of wavelets we see that for a large $j$ $\psi_{j,k}(t)$ is short and of high frequency. Smaller values for $j$ on the other hand give long wavelet functions of low frequency. This makes it easy to analyze a function in different scales. The compact support of wavelet functions also show up in the transform giving good localization properties, in particular sharp time localization also at high frequencies which distinguishes it from other short-time Fourier transforms. Orthogonal wavelet transforms also preserve the energy of the original function as a formula similar to Parseval's theorem is available.

The approximation of a function/signal via wavelet functions depends upon the resolution of the basis functions. A larger $j$ for example gives us shorter functions which hence approximates the signal with increased accuracy. We define the space of functions generated via a particular $j$ as $W_j$; $W_j = sp\{\psi_{j,k}\}$.

Functions in $W_{j+1}$ have double resolution than those in $W_j$, due to the scaling by $2^j$. This can also be observed by directly comparing figures 2.1 and 2.2.

$$W_j \subset W_{j+1}$$

As $j$ is increased it should start incorporating the space of all functions with limited energy:

$$W_0 \subset W_1 \subset ... \subset W_j \subset W_{j+1} \subset W_{j+2} \subset ... \subset L^2$$

## 2.2 Scaling functions

In approximation theory wavelet functions go hand in hand with scaling functions $\phi(t)$; each wavelet function $\psi(t)$ is linked with a scaling function creating a pair. Together these two functions are used to approximate functions or signals. The scaling function gives an overall coarse approximation of the signal while the wavelet functions give the more finer details. The coarse signal along with the details combine into a multiresolution analysis of the signal.

Scaling functions related with wavelets in the previous figure are shown in figure 2.3.

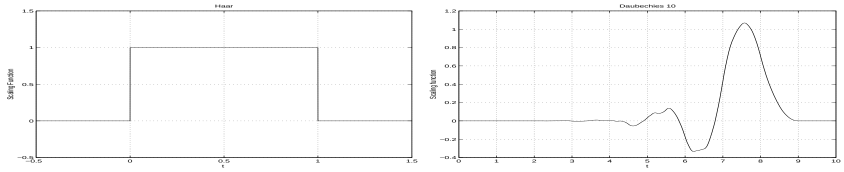

Figure 2.3: Example of scaling functions

The Haar scaling function can be seen on left and the Daubechies 8 scaling function is on right. Like wavelet functions scaling functions too are required to have compact support but they do not have zero mean value. On the contrary normalized scaling functions must cover an area of one unit: $\int_{-\infty}^{\infty} \phi(t)\ dt = 1$.

Shifting and dilation of scaling functions follows the same preceding methodology: $\phi_{j,k}(t) = 2^{j/2}\phi(2^j t - k)$ thus a basis of scaling functions can also be used to approximate a function: $f(t) = \sum_{j,k}\ a_{j,k}\ \phi_{j,k}(t)$.

The scaling functions used in the expansion have just like wavelets functions changing time-frequency properties.

Scaling functions too are required to satisfy the requirements for orthogonality but only on the same scale:

$$\int_{-\infty}^{\infty} \phi_{j,k}(t)\ \phi_{j,l}(t)\ dt = \delta(k-l)$$

On top of this the scaling functions must be orthogonal to the wavelet functions both on scale and translation:

$$\int_{-\infty}^{\infty} \phi_{j,k}(t)\ \psi_{J,K}(t)\ dt = 0$$

The last but important constrain makes it difficult to construct smooth orthogonal wavelet filters consequently many orthogonal wavelet system have functions with irregular shapes resembling fractals. Orthogonality is very important for numerical stability and also for energy preservation.

The space of functions generated via a particular $j$ by scaling functions is defined by $V_j$; $V_j = sp\{\phi_{j,k}\}$.

As previously we have $V_j \subset V_{j+1}$ and increasing $j$ leads us to:

$$V_0 \subset V_1 \subset ... \subset V_j \subset V_{j+1} \subset V_{j+2} \subset ... \subset L^2$$

The functions on the left sides in figure 2.4 are $\phi_{1,0}(t)$ which are just dilated versions of the main scaling functions $\phi(t)$. The other functions on right (dashed) are additionally shifted by one: $\phi_{1,1}(t)$. The original mother function can be obtained easily: $\phi(t) = \frac{1}{\sqrt{2}}\phi_{1,0}(t) + \frac{1}{\sqrt{2}}\phi_{1,1}(t)$.

The scaling by 2 always doubles the resolution when the basis is increased from $V_j$ to $V_{j+1}$, this means we can write a function $g(t)$ in $V_j$ as $g(2t)$ in $V_{j+1}$. Generalizing this an equation connecting the functions from these two different bases can hence be constructed:

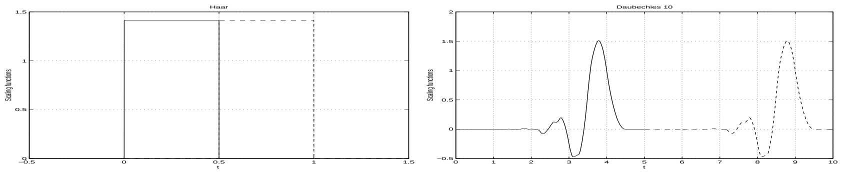

Figure 2.4: Example of dilated and translated scaling functions

$$\phi(t) = \sqrt{2}\sum_{k=0}^{N-1} h[k]\ \phi(2t-k)$$

This is the dilation equation for the scaling function.

Due to orthogonality all elements in $V_j$ must be perpendicular to $W_j$. It can be validated that $W_j$ makes up for the difference of elements between $V_j$ and $V_{j+1}$. This can be used as a new explicit method to construct wavelets out from scaling functions. As the wavelets are the differences of the scaling functions we get $\psi(t) = \frac{1}{\sqrt{2}}\phi_{1,0}(t) - \frac{1}{\sqrt{2}}\phi_{1,1}(t)$. This creates a direct relationship between scaling functions and wavelet functions.

We can now conclude with that a similar equation to the dilation equation can be implemented to link wavelets with scaling functions.

$$\psi(t) = \sqrt{2}\sum_{k=0}^{N-1} g[k]\ \phi(2t-k)$$

This equation is called the wavelet equation.

It turns out that $h[n]$ and $g[n]$ can in fact be regarded as coefficients of FIR filters.

## 2.3 Multiresolution analysis and Biorthogonality

The standard idea behind wavelet analysis is to represent a function or a signal via a combination of scaling and wavelet functions. The scaling functions are used to give an overview of the main function while wavelets represent the errors of the approximation done by the scaling function at different levels.

The approximation of a function given by the scaling function at a given scale is not perfect and it's here where wavelets come in. One can think of functions in $W_j$ as a means of representing the parts of a function in $V_{j+1}$ that cannot be represented in $V_j$. As $W_j$ is orthogonal to $V_j$ this does not pose any overlap and by using both $V_j$ and $W_j$ the accuracy is improved by one scale. We can therefore write $V_{j+1} = V_j \oplus W_j$. Repeating the argument we end up with

$$V_{j+1} = V_0 \oplus W_0 \oplus W_1 \oplus ... \oplus W_j$$

$V_0$ can be said to give the average behavior of a function/signal while $W_l$ gives the more finer details at various scales. This is multiresolution analysis.

If the approximation by the scaling function is good then one can expect good decorrelation as many of the wavelets coefficients will be close to zero.

So far we have seen orthogonal wavelets where $W_j$ must be the orthogonal complement of $V_j$ in $V_{j+1}$. Biorthogonal wavelets define a superset of the orthogonal wavelets and were first developed by Cohen et al. [4] for use in multiresolution analysis.

Biorthogonal wavelets as the name implies are no longer directly orthogonal to the scaling functions but they do satisfy certain orthogonal properties via the dual basis functions. Dual basis functions for $u_i$, $\tilde{u}_i$ are defined as:

$$\int_{-\infty}^{\infty} u_i(t)\ \tilde{u}_j(t)\ dt = \delta(i-j)$$

Dual basis functions are central to the definition of biorthogonal wavelet constructions. Using the duals for the scaling and the wavelet functions biorthogonal wavelets must satisfy the following conditions:

$$\int_{-\infty}^{\infty} \phi_{j,k}(t)\ \tilde{\psi}_{J,K}(t)\ dt = 0 \quad \text{and} \quad \int_{-\infty}^{\infty} \tilde{\phi}_{j,k}(t)\ \psi_{J,K}(t)\ dt = 0$$

Scaling and wavelet functions are now orthogonal to the duals: $\int_{-\infty}^{\infty} \phi_{j,k}(t)\ \tilde{\phi}_{j,l}(t)\ dt = \delta(k-l)$ and $\int_{-\infty}^{\infty} \psi_{j,k}(t)\ \tilde{\psi}_{J,K}(t)\ dt = \delta(j-J)\delta(k-K)$.

Furthermore the orthogonality $V_j \perp W_j$ takes a new general shape:

$$V_j \perp \tilde{W}_j \text{ and } \tilde{V}_j \perp W_j$$

where $\tilde{V}_j = sp\{\tilde{\phi}_{j,k}\}$ and $\tilde{W}_j = sp\{\tilde{\psi}_{j,k}\}$.

As an effect we now need two different hierarchies of spaces, $V_j$ and $\tilde{V}_j$. In addition to the standard dilation and wavelet equations we need two similar equations for the duals:

$$\tilde{\phi}(t) = \sqrt{2}\sum_{k=0}^{N-1} \tilde{h}[k]\ \tilde{\phi}(2t-k)$$
$$\tilde{\psi}(t) = \sqrt{2}\sum_{k=0}^{N-1} \tilde{g}[k]\ \tilde{\phi}(2t-k)$$

Expansions are now carried out by the standard functions but reconstruction is performed via the duals. Having a non-orthogonal basis does complicate things a bit as shown in figure 2.5.

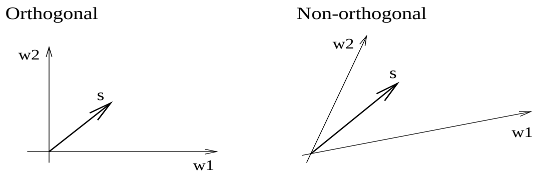

Figure 2.5: Orthogonal and non-orthogonal basis

In a (normalized) orthogonal basis the signal $s$ can be represented easily by linear combinations of the basis vectors; such as $s = c_1 w_1 + c_2 w_2$ where $c_i$ are the coefficients calculated by projecting $s$ on $w_i$.

In the non-orthogonal basis this will not be the situation. Projecting $s$ into the axes and later adding together as in the orthogonal basis will generally not equal $s$, instead it will be a different function. The non-orthogonal basis will allow us to reconstruct the original $s$ but to do so when projecting $s$ into $w_1$ we will have to be careful to make the projection parallel to $w_2$. To compute a single coefficient for $w_1$ will therefore involve using the other vectors for the basis as well as. This is here where some correlation come into play as for the biorthogonal case we have two- and two pairs of functions which are not orthogonal but in between linked via the biorthogonality relations.

## 2.4 Filter banks

The wavelet multiresolution analysis method can easily be extended and implemented on discrete data by means of filter banks. A filter bank is basically just a collection of many filters preferably with certain properties and with a common input signal. Figure 2.6 shows an example of a such two-channel filter bank.

As long as the two filters $H_0(z)$ and $H_1(z)$ are different the output subsignals $u[n]$ and $v[n]$ are also not equal to each other. The idea behind filter banks makes more sense if one of the filter for example $H_0(z)$ is a lowpass filter and the other a highpass filter. This way the filter bank splits the signal into different frequency bands. The length of these two new signals is the same

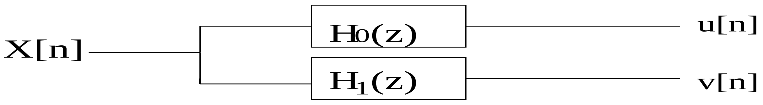

Figure 2.6: Two-channel filter bank

as the original signal, however, it should be possible to downsample $u[n]$ and $v[n]$ by two by removing every second sample, thus avoiding any increase of the data. It turns out that this is indeed possible if the filters satisfy certain mathematical requirements.
Repeating the process backward by upsampling and inverse filtering then recovers the original signal without any data loss.

The concept of filter banks can be expanded in many ways. One strategy is to iterate the process on the subsignals a few times. The ideas behind wavelet filter banks follow pretty much the same methodology: a pair of filters are used where a lowpass and a highpass filter participate, the filters are also constructed so that downsampling by two still later enable perfect reconstruction of the signal. Additionally the decomposition process is iterated a few times on the low frequency side. See figure 2.7 for an example of a wavelet filter bank with one level of decomposition and reconstruction.

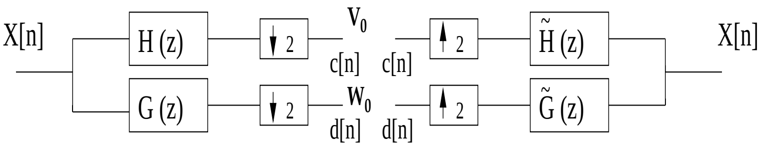

Figure 2.7: Wavelet filter bank

When displaying the decomposed signal it is standard custom to show all the sections in one plot where the left side displays the low frequency part while the high frequency subsignal is on right. See figure 2.8.

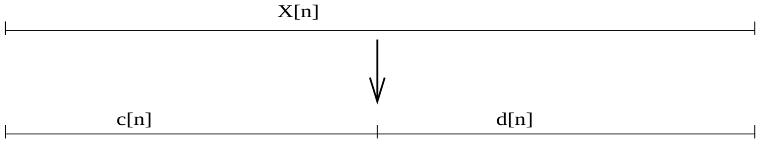

Figure 2.8: Plotting practice

The decomposing filters ($h[n]$ and $g[n]$) on the left are the analysis filters while the reconstruction filters ($\tilde{h}[n]$ and $\tilde{g}[n]$) go under the name of synthesis filters. A tilde sign is imposed to indicate the synthesis side.
$c[n]$ stands for the lowpass filtered signal downsampled by 2 while $d[n]$ is the highpass filtered equivalent. These conventions are used throughout the chapters.

Figure 2.9 shows how the iteration process can be repeated with wavelets.

The plotting follows the logarithmical decomposing style (figure 2.10).

To use wavelet techniques effectively on discrete data we need to treat wavelets in a discrete manner. This is accomplished by looking at the wavelet and scaling functions as impulse responses of a filter. The wavelet function generates the analysis highpass filter $g[n]$ while the

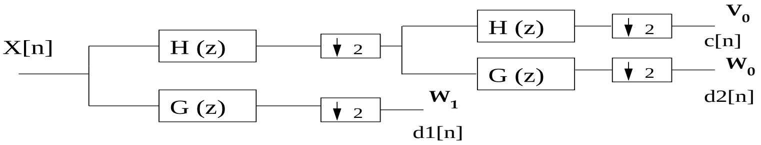

Figure 2.9: Wavelet filter bank - iterative decomposition

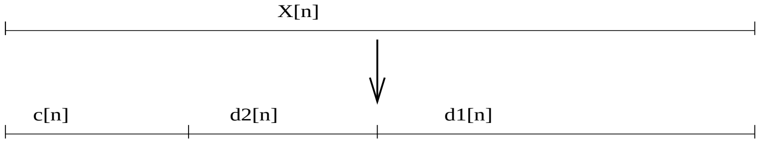

Figure 2.10: Plotting practice

scaling function builds the lowpass filter $h[n]$. Passing a signal $X[n]$ through the lowpass scaling filter removes the high frequencies from the signal, half of the samples can then be eliminated to obtain $c[n]$. Exactly the same procedure is also repeated with the highpass wavelet filter to acquire a high frequency subsignal $d[n]$. The lowpass averaging filter is there to capture the coarse behavior of the signal while filtering via $g[n]$ returns the finer details.

The signal is now decomposed in two parts; one with low frequencies and the other with high frequencies. On the negative side the time resolution is reduced by a factor of two, this process is also known as subband coding. At every level (scale), the filtering and down sampling results in half the number of samples along with half the time resolution, and half the frequency band spanned with doubled frequency resolution. The process is then normally repeated with the lowpass filtered subband, however, one is free to iterate the process even with the highpass filtered section; such decomposing goes under the name of wavelet packets. Figure 2.11 demonstrates this.

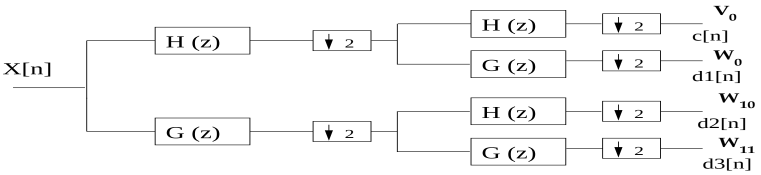

Figure 2.11: Wavelet packet two-scale decomposition

Having a filter bank can be of limited usage if one is not in a position to later fully recover the original signal. It can be shown that perfect reconstruction from a two-channel filter bank is possible only if the filters satisfy:

$$\tilde{H}(z)\ H(z) + \tilde{G}(z)\ G(z) = 2z^{-l}, \qquad \text{odd } l$$
$$\tilde{H}(z)\ H(-z) + \tilde{G}(z)\ G(-z) = 0$$

The first equation is required for reestablishment of the original signal with a possible time-delay while the second equation is there to cancel out any aliasing.

The desired property of orthogonality in filter banks additionally imposes some very strict relations on the filters. If the lowpass filter $H(z)$ of length $N$ has been selected then the analysis highpass filter must satisfy

$$G(z) = -z^{-N} H(-z^{-1})$$

Orthogonal scaling and wavelet filters are related with each other in a perfect way creating symmetry in the frequency response of the filters. Figure 2.12 shows such an example of the frequency responses of an orthogonal filter bank. $|H(\omega)|^2 + |G(\omega)|^2 = 2$ is satisfied and this is always the case with orthogonal wavelet filters.

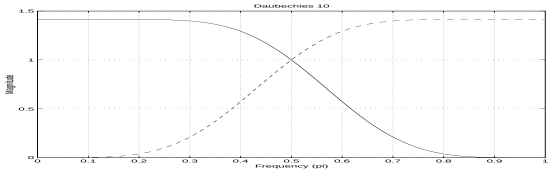

Figure 2.12: Frequency responses of low (solid) and highpass (dashed) filters

Reconstruction of the decomposed signal is later performed with synthesis filters $\tilde{H}(z)$ (lowpass) and $\tilde{G}(z)$ (highpass). The inverse filters must be related with the analysis filters to enable perfect reconstruction and cancel out any aliasing.

The synthesis filters are generated by alternating the signs and flipping the analysis filters:

$$\tilde{H}(z) = z^{-N}G(z^{-1})$$
$$\tilde{G}(z) = -H(-z)$$

Having set the lowpass filter coefficients there is therefore only one unique way to construct the highpass and the reconstruction filters. The filters are all also required to be of even length. If we for example start with $h[n] = [a \;\; b \;\; c \;\; d]$ then the other filters can be obtained via the asymmetries given in the mentioned equations, see figure 2.13.

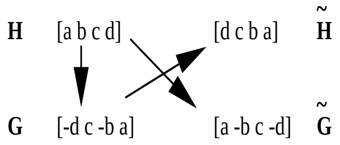

Figure 2.13: Relationship between coefficients

Orthogonal filters are little flexible due to their intimate relationships, biorthogonal filter banks give us more options in designing filter banks and can be related with biorthogonal wavelet theory. This removes the demand of orthogonality between $H(z)$ and $G(z)$, unfortunately frequency responses from biorthogonal filters may now not show any symmetry and the energy of the decomposed signal may also not equal the energy of the original signal.

The added flexibility shows up very clearly as the two filters which analyze and divide the signal into a low- and a highpass section $H(z)$ and $G(z)$ are allowed to be very different from each other. The filters, however, are required to show orthogonality to the quadrature filter bank pair which is used for the inverse transform.

Why do we need to use biorthogonal wavelets at all? The answer to this question lies behind some requirements which are often put on wavelet functions. One such condition is that wavelet functions should have compact support; looking at the functions as impulse responses compact support transforms over to a filter having finite impulse response. If a wavelet function does not have compact support we have to use an IIR filter to approximate the function, and IIR filters are more difficult to work with and they also require extra computing power.
The demand for compact support and orthogonality leads to filters which must be asymmetrical, henceforth a non-linear phase in FIR filters. The use of a non-linear phase filter can in some cases result in unwanted artifacts in the data.

If we would like to have symmetry in wavelet filters by asking for linear phase filter response then we must be willing to give up one of our requirements of either having orthogonality or compact support. Although non-compactly supported filters have been demonstrated [7] they have their own problems. It turns out that using biorthogonal wavelets we acquire both symmetry and compact support in our wavelets. They can also be made almost perpendicular and do not cause extra problems computationally, giving us linear phase filters rather cheaply. Due to the symmetry in filters the wavelet functions generated are also symmetrical, something which is not achievable with orthogonal filters. Symmetrical wavelet functions are often preferred in image processing.

Biorthogonal systems are also more easier to construct than orthogonal filters due to their flexibility. Scaling filters $H(z)$ and $\tilde{H}(z)$ now don't even need to be of the same length but they must both have either even or odd number of taps.

The synthesis filters can as for the orthogonal case be generated by alternating the signs and swapping the analysis filters. Figure 2.14 demonstrates the case for odd length symmetrical filters.

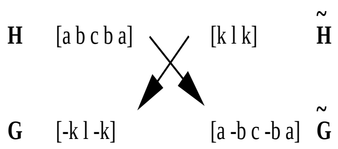

Figure 2.14: Relationship between coefficients

As long as the filters satisfy the two equations required for perfect reconstruction they build a legal PR filter bank. In the next section certain requirements on the filters are gone through which guarantee the convergence of filter banks to proper scaling and wavelet functions.

In figure 2.9 an example of an iterated filter bank was shown. Although the scaling and wavelet functions are not explicitly used, a link between the discrete-time filters $h[n]$ and $g[n]$ with the continuous-time scaling and wavelet functions can be generated via the recursive *cascade algorithm* :

$$\phi^{(j)}(t) = \sqrt{2}\sum_{n=0}^{N-1} h[n]\ \phi^{(j-1)}(2t-n)$$
$$\psi^{(j)}(t) = \sqrt{2}\sum_{n=0}^{N-1} g[n]\ \psi^{(j-1)}(2t-n)$$

Given initial starting values $\phi^{(0)}(t)$ and $\psi^{(0)}(t)$, the goal is of course that these functions converge to a fix point:

$$\lim_{j\to\infty} \phi^{(j)}(t) = \phi(t)$$
$$\lim_{j\to\infty} \psi^{(j)}(t) = \psi(t)$$

Twenty iterations of the above algorithm are used when plots of scaling and wavelet functions are displayed in this text.

## 2.5 Properties and Design

There are many different methods on how to construct wavelet filters for use in filter banks. The classical strategy goes on first creating analytical formulas for wavelet and scaling function who are then transformed into their discrete counterparts.

An alternative to the above strategy is to directly work upon the filters. Even here one can go more theoretical to work by building filters in the z-domain with the desired properties or one can directly generate discrete filters with numerical values. We will focus upon numerical construction of wavelet filter banks as this is one way to make filters which can be made signal specific.

If orthogonal wavelets are desired then one can start building with the analysis scaling filter $H(z)$ according to the required properties.

For biorthogonal wavelets we need different set of bases for $H(z)$ and $\tilde{H}(z)$. This way we start with the scaling filters and construct the wavelets out from that.

- The scaling filter $H(z)$ (and $\tilde{H}(z)$) are lowpass filters and in wavelet theory they are required to have at least one zero point at $z = -1$ in the complex plane ($|H(\pi)| = 0$). As the wavelet filters are designed by swapping and sign changing the scaling filters this implies that $G(z)$ (and $\tilde{G}(z)$) have at least one zero at $z = 1$ ($|G(0)| = 0$).

  In discrete form this transforms over to:

  $$\sum h[2n] \ = \ \sum h[2n+1]$$

- The requirement on the other end of the frequency response is $|H(0)| = \sqrt{2}$ for all scaling filters. This is the same as imposing:

  $$\sum h[n] \ = \ \sqrt{2}$$

  This is an essential condition for the dilation equation to converge to compactly supported basis functions.

  It can be shown that the effect of the last two equations on the wavelet filters leads to an average value of 0: $\sum g[n] = 0$. We also end up with $|G(\pi)| = \sqrt{2}$.

- Conditions for orthogonality or more general biorthogonality are very important and the discrete version takes the shape of:

  $$\sum h[n] \ \tilde{h}[n+2k] = \delta[k]$$

  If one is working with orthogonal filters then due to symmetry in the filters the equation obtains a very simple shape for $k = 0$: $\sum h[n]^2 = 1$. In other words the energy of the original signal is preserved by the filter.

It can be shown that the above conditions are all what we need to design proper discrete wavelet filter banks with perfect reconstruction and that the functions generated will satisfy all the main requirements set for them in the continuous case. There is, however, no guarantee that the functions generated via the cascade algorithm will really converge to a function with limited energy.

The above equations are also not enough to fully define a wavelet system. If both the filters are of length $N$ then there is $N/2 - 1$ degrees of freedom left in the system which can be used pretty much as pleased. In many cases this extra degree of freedom is used to make the scaling functions more smoother or to form the magnitude response of the filters as flat as possible at certain frequencies.

### 2.5.1 Regularity

Wavelet and scaling functions can be of many shapes but one often desired requirement is that the functions should be reasonable smooth. Regularity is concerned with differentiability of these functions which in one sense defines smoothness.

Regularity can be linked with the number of zeros at $z = -1$ of the scaling filter which are also known as the number of *vanishing moments* for the wavelet filter. The higher the number of vanishing moments the more tightly energy is concentrated among the scaling functions. With $M$ moments the wavelet function fulfills $\int_{-\infty}^{\infty} t^k \ \psi(t) \ dt = 0$ for $k = 0, 1, .., M$, and all polynomials up to degree $M - 1$ are captured by the space of $V_j$ in a decomposition. This property is important when a smooth function or signal needs to be decomposed using wavelets.

At least one zero at $z = -1$ is required from the scaling filter in the first place but with additional zeros the degree of smoothness can be increased. As a matter of fact regularity is the main point which distinguishes filter banks with perfect reconstruction from wavelet filter banks.

The discrete version of the above equation takes the form of:

$$\sum n^K g[n] = 0$$

which must be satisfied for $K = 0, 1, .., M$.

A regular filter through iterations lead to a scaling function with some degree of smoothness.

The number of vanishing moments can also be increased to $\hat{M}$ for the scaling filters them self by making sure they obey $\int_{-\infty}^{\infty} t^k \ \phi(t) \ dt = 0$ for $k = 0, 1, .., \hat{M}$. This can be of interest in certain circumstances such as when wavelet packets are to be implemented. Vanishing moments for scaling function can also be related to indicate the symmetry characteristics of the function.

### 2.5.2 Images

The wavelet concepts for one dimensional signals can trivially be extended to two dimensional images. Instead of creating wavelet and scaling functions of two variables one can treat a picture row-by-row and column-by-column. This gives a very cost effective and simple method to expand wavelet theory into several dimensions.

A wavelet decomposition of a digital image is performed by first going through an image row-by-row and decomposing each row like it was a standard one dimensional signal. After having gone through all the lines a new image can be build where the left side represents the low frequency part of each row while the right side shows the high frequency parts of each row. The same steps can next be repeated column-by-column which at the end gives an image which is arranged of four quadrats.

The square on upper-left only consists of low frequencies while the lower-right square only shows the very high frequency details. The other two subbands display a mixture of low and high frequency data. The decomposition can be iterated where one for example further divides the low frequency part into four new quadrats.

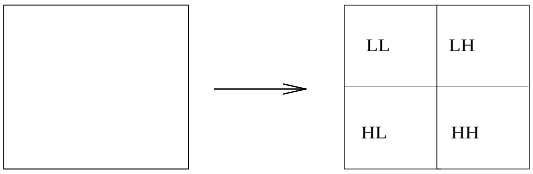

Figure 2.15: Image decomposition L: $f(x, y)$ R: $w(x, y)$

Over the years wavelets have become very popular for image compression. After a wavelet decomposition most of the energy of a picture is found in the low-low frequency zone while the other parts only contain some limited amount of it, as most images do not contain a lot of high frequency details. In case of lossy compression the high frequency parts can be quantized heavily before being entropy coded. Unlike standard JPEG compression where an image is treated and quantized in blocks, a wavelet decomposition smears out any negative effects over the whole image giving improved outcomes at the same compression rates. Wavelet decomposition also has other advantages due to the very nature of basis functions which generally give good decorrelation of data.

## 2.6 Filters

Using filter banks a signal can be decomposed into bands of frequencies and this is where the usefulness of filter bank lies. The wavelet decomposing process tries to approximate the signal with the aid of the scaling and wavelet functions. If the approximation done by scaling filter is good then much of the signal's energy is captured in the first low frequency zone and many of the high frequency coefficients obtained via filtering by $G(z)$ are close to zero. This decorrelation of frequencies in time-domain gives a big advantage for compression purposes as the separated parts can often each be entropy coded much more efficiently than the original signal in one go.

Before coding the subsignals it is possible to quantize the results if one is aiming for lossy compression. The exact coding gain depends upon the type of filters used and how well the scaling and wavelet functions approximates the signal. Generally smooth wavelet functions are preferred in wavelet analysis as these can then be used with all kinds of signals. Any noise or irregularities hence directly show up in the high frequency part. This also makes it easy to quantize the high frequency differently than the low frequency zones.

In the next chapters we will utilize a number of predesigned wavelet filters which are available from different sources. Both orthogonal and biorthogonal filters will be used.

The first number after the name of a filter always indicate the length of the analysis scaling filter $h[n]$, while the second figure the length of the synthesis scaling filter $\tilde{h}[n]$. For orthogonal filters the lengths are equal and the second number is dropped.

The orthogonal filters we will use are:

- Haar 2: This is the classical Haar filter which is also used for example plots in this chapter [3] [6].
- Daubechies: These are the popular filters designed by Daubechies, and various lengths are available. The frequency response of these filters have the property of being maximum flat at $\omega = 0$ and $\omega = \pi$ [6].
- Coiflet 6, 12 and Symmlet 6,12 and 16: Another group of orthogonal filters where the extra degrees of freedom are used to increase the smoothness/make wavelets as near-symmetrical as possible [6].

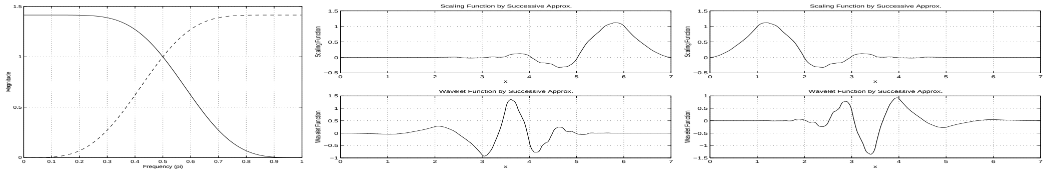

Figure 2.16: Daubechies 8, L: Frequency response, S/W functions: M: Analysis R: Synthesis

The frequency response of some of these orthogonal filters is shown on left (L) in figures 2.16 and 2.17. The analysis scaling filter is always shown in a solid print, the analysis wavelet filter

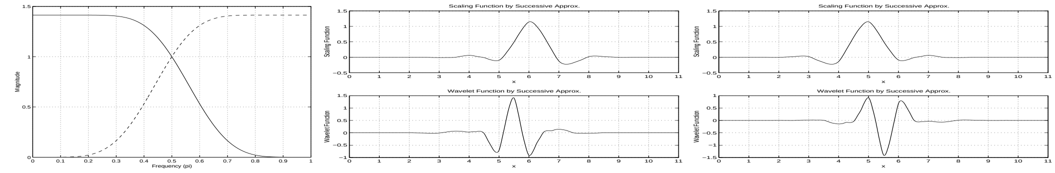

Figure 2.17: Symmlet 12, L: Frequency response, S/W functions: M: Analysis R: Synthesis

always in a dash line. The analysis scaling and wavelet functions them self are in the middle (M) while the synthesis scaling and wavelet functions are on right (R).

There is a wide variety of choices when it comes to the selection of biorthogonal filters. We have chosen some traditional filters often used in image processing along with a few non-standard filters:

- Crew 2-6: This biorthogonal filter was implemented by CREW [18]. Also elected by [1] to compress MR-images.
- 5-3, 9-3 and 6-10: These filters were chosen as the winners in a study done by Villasenor, Belzer and Liao [16].
- CDF 9-7: A filter designed by Cohen, Daubechies and Feauveau [4] which has become very popular for use in image compression and is also used by the FBI for fingerprint compression. Also selected as the winner by [16], additionally this filter is found in the JPEG 2000 compression standard [10].

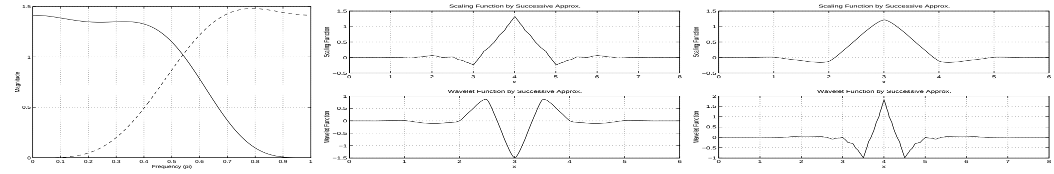

Figure 2.18: CDF 9-7, L: Frequency response, S/W functions: M: Analysis R: Synthesis

- EL 12-4: Proposed by Egger and Li [8].
- Bathlets 10-10: Filters designed at Bath university, with parameter value of either 0.7 or 1.4 [11].
- Coiflet 17-11: A modified variant of the orthogonal Coiflet filters [17].

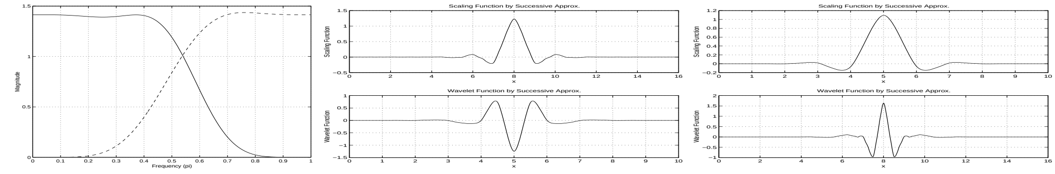

Figure 2.19: Coiflet 17-11, L: Frequency response, S/W functions: M: Analysis R: Synthesis

- B-spline 11-5: A biorthogonal filter generated via B-Spline functions by the Uvi Wave Matlab package [1].

[1] wspline(4,4)

## 2.7 Entropy

Entropy doesn't directly have anything to do with wavelets but as wavelet filter banks are often used to decompose a signal before coding it it is natural to mention it in this context.

Entropy is used to measure the amount of information stored in a collection of data via probabilistic theory. If an event $E_1$ occurs a number of times in the data then the probability of it happening $P(E_1)$ can be associated with *units of information* as $I(E_1) = -\log P(E_1)$. The negative sign ensures that only positive values are returned and the base of logarithm determines the unit which is used to measure information. In case of base 2 the measurements are directly in (digital) bits.

As long as there are a finite amount of different letters in the data a source alphabet can be constructed: $\{a_1, a_2, ..., a_J\}$. The probability that the source produces a letter $a_j$ is defined as $P(a_j)$; while the sum will add up to one, $\sum P(a_j) = 1$. Combining this with the units of information we get the average information per source $e$ which is defined as: $e = -\sum P(a_j) \log P(a_j)$, this is the *entropy* of the data set. In case of base 2 this indicates the average number of bits required to code a data set. A good coding algorithm should try to come as close to the entropy figure as possible; although it is not possible to code in fewer bits than the limit given by the entropy, a transformed data set may well have a much lower entropy. This is where wavelets come in.

# Chapter 3

# Biorthogonal Wavelet Filter Banks

Biorthogonal filters define a superset of orthogonal wavelet filters and have found their use in virtually all areas where wavelets are utilized.

There are, however, certain aspects of the biorthogonal wavelet transform which can be interesting to investigate. Since biorthogonal wavelet transforms obviously are not orthogonal, it should be possible to explore the aliasing and energy shifts which occur do to the non-orthogonality of the filters. The outcomes may give us some hints on why biorthogonal wavelets have been so successful till today and how they should be designed. As there is a big variety of available biorthogonal wavelets it is not unlikely that different filters have somewhat different properties.

We shall use a a synthetic signal of 5000 white noise samples (figure 3.1) for our experiments, as such a signal covers the whole frequency spectrum with the energy uniformly spread out.

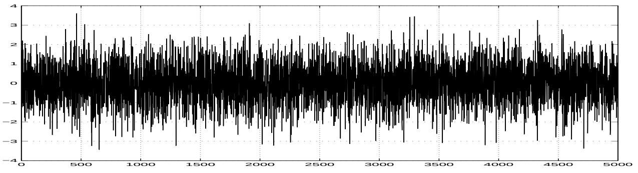

Figure 3.1: test signal

## 3.1 Energy shifts

One of the constrains put on orthogonal wavelet filters is that when a signal is decomposed in a low and a high frequency part then the total quantity of energy in both subbands should equal the total amount of energy in the original signal.

$$\sum y[k]^2 = \sum (c[k]^2 + d[k]^2)$$

Our experimental signal is built up of white noise and if we decompose it be using orthogonal wavelets then the signal's energy will hence be divided more or less equally between the low and high frequency subbands.

Biorthogonal filters on the other side are much more flexible at this and the sum of energy in low and high frequency areas may be very different from the original signal. The signal

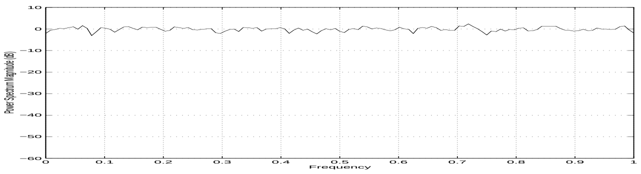

Figure 3.2: Power spectrum of test signal

decomposed using the famous CDF 9-7 biorthogonal wavelet filter is shown in figure 3.3, the white noise in the new signal should now have become somewhat colored.

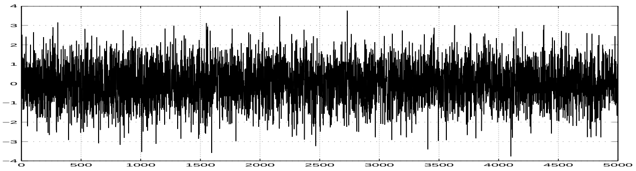

Figure 3.3: decomposed signal

Experimenting with different filters we decompose this signal and calculate the energy in the low and high frequency parts as well as the total energy of the signal. The outcomes are shown in tables 3.1 and 3.2 as percentage values relative to the total energy of the original signal.

Total energy of our original signal is: 5095.76.
Fourier transform reveals that 2609.97 (51.21%) is in the low frequency part and 2485.63 (48.77%) among the high frequencies.

| Orthogonal Filter | Energy L | Energy H | Sum |
|---|---|---|---|
| Haar 2 | 51.25% | 48.74% | 100% |
| Daubechies 8 | 50.79% | 49.20% | 100% |
| Daubechies 10 | 51.52% | 48.47% | 100% |
| Daubechies 12 | 51.79% | 48.20% | 100% |
| Daubechies 20 | 51.69% | 48.30% | 100% |
| Coiflet 6 | 51.36% | 48.63% | 100% |
| Coiflet 12 | 51.72% | 48.27% | 100% |
| Symmlet 6 | 51.74% | 48.25% | 100% |
| Symmlet 12 | 51.75% | 48.24% | 100% |
| Symmlet 16 | 51.76% | 48.23% | 100% |

Table 3.1: Energy divisions of original signal

For orthogonal filters the outcomes are pretty much as expected. The energy is more or less separated equally between the zones, and the total energy remains unchanged regardless which orthogonal filter we use.

For biorthogonal filters the results in table 3.2 turn out to be very different from that under orthogonal filters. Examining the table it is evident that biorthogonal wavelets increase the total energy of the signal as they decompose it into low and high frequency parts. This is the direct result of filters not being orthogonal and data gets separated in a redundanc way, albeit the total energy increase is modest for the majority of the filters.

| Biorthogonal Filter | Energy L | Energy H | Sum |
|---|---|---|---|
| Crew 2-6 | 51.25% | 50.45% | 101.71% |
| 5-3 | 74.00% | 36.58% | 110.58% |
| 9-3 | 73.58% | 36.58% | 110.16% |
| CDF 9-7 | 53.97% | 47.61% | 101.58% |
| 6-10 | 65.44% | 39.27% | 104.74% |
| EL 12-4 | 103.79% | 30.56% | 134.35% |
| Bathlets 10-10 0.7 | 54.30% | 46.06% | 100.37% |
| Bathlets 10-10 1.4 | 52.41% | 47.68% | 100.10% |
| Coiflet 17-11 | 56.11% | 45.78% | 101.89% |
| B-spline 11-5 | 172.95% | 26.67% | 199.62% |

Table 3.2: Energy divisions of original signal

Although biorthogonal filters generally seem to increase the total energy it is much more of interest to observe that the energy located in the high frequency band is quite a bit lower than in the case of orthogonal filters. All of the energy increase occurs in the low frequency domain; one can say that the energy is indeed more or less shifted towards the lower frequencies from higher frequencies. The 2-6 filter is the only filter which does not fit in with the general pattern, a more detailed study of the filters should be able to explain the differences between filters.

In spite of the fact that biorthogonal filters increase the total energy of the signal this does not automatically imply that the decomposed signal requires extra storage space. The number of coefficients remains the same regardless which filter we use. Yet, if a frequency zone contains relative few values (low energy) then compression may be performed easily and effectively. The opposite is true if a zone contains too much energy, compression may then not be very effective due to high degree of entropy. For compression purposes the ultimate goal is therefore to transform a signal in a stable way so the amount of energy in low/high frequency areas is as low as possible.

Decomposing using biorthogonal wavelets increases the total energy of an all frequency signal in a certain way special way, however, how does low/high frequencies take part in this increase ?

One possible strategy to find this is to use a signal which only consist of low frequencies. For example a chirp or a filtered signal so it is free from high frequencies. Opting for the second alternative we use the Remez method to design a filter which removes all high frequencies from our signal.

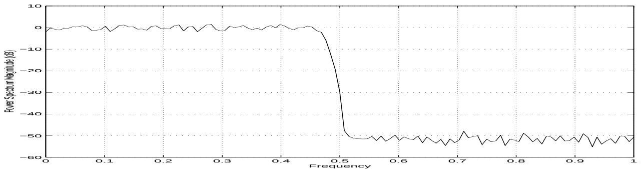

Figure 3.4: Power spectrum of lowpass filtered signal

The filtered signal hardly contains any high frequencies as one can observe in figure 3.4. The total energy of our lowpass filtered signal is calculated to be 2418.65. From the Fourier transform it is also straightforward to compute the energy located in low/high frequency parts. The total energy below $\frac{\pi}{2}$ is $2418.51$ (99.99%) and in the high frequency area: $0.14$, only 0.005% of the total signal's power.

If we now take wavelet decompositions of this signal using different filters and calculate the power located in the low/high frequency zones we obtain some interesting results shown in tables 3.3 and 3.4.

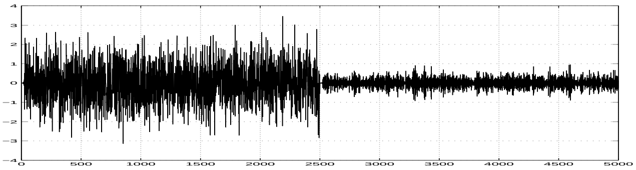

Figure 3.5: Orthogonal Daubechies 8 decomposition

| Orthogonal Filter | Energy L | Energy H | Sum |
|---|---|---|---|
| Haar 2 | 83.69% | 16.31% | 100% |
| Daubechies 8 | 93.15% | 6.84% | 100% |
| Daubechies 10 | 94.11% | 5.88% | 100% |
| Daubechies 12 | 94.83% | 5.16% | 100% |
| Daubechies 20 | 96.50% | 3.49% | 100% |
| Coiflet 6 | 89.46% | 10.53% | 100% |
| Coiflet 12 | 93.40% | 6.59% | 100% |
| Symmlet 6 | 91.68% | 8.31% | 100% |
| Symmlet 12 | 94.83% | 5.17% | 100% |
| Symmlet 16 | 95.83% | 4.16% | 100% |

Table 3.3: Energy divisions of lowpass filtered signal

In the case of orthogonality the total energy remains unchanged and most of the energy is found in the low frequency band. The longer the filter the less energy in the high frequency zone. Daubechies and Symmlet filters of same length have roughly the same performance, the Coiflet variants though are somewhat less effective at reducing the power in the high frequency area.

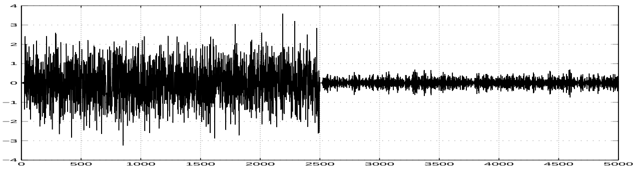

Figure 3.6: Biorthogonal CDF 9-7 decomposition

Biorthogonal wavelet filters on the other hand enlarge the total energy of the lowpass filtered signal in the decomposing process. One can also clearly see from table 3.4 that the energy quantity in the low frequencies is propagated while in the high frequency zone the energy is more or less reduced compared with orthogonal filters of the same length. One can even say that the signal's energy is concentrated tightly over to the low frequency band.

From our tests two biorthogonal filters, 2-6 and 9-7 stand out clearly from the rest as the total energy of the decomposed signal is in fact lower than the original signal.

| Biorthogonal Filter | Energy L | Energy H | Sum |
|---|---|---|---|
| Crew 2-6 | 83.69% | 8.01% | 91.70% |
| 5-3 | 114.83% | 4.61% | 119.44% |
| 9-3 | 122.40% | 4.61% | 127.01% |
| CDF 9-7 | 91.55% | 4.40% | 95.96% |
| 6-10 | 113.91% | 3.80% | 117.72% |
| EL 12-4 | 160.23% | 1.53% | 161.76% |
| Bathlets 10-10 0.7 | 97.98% | 7.19% | 105.17% |
| Bathlets 10-10 1.4 | 92.70% | 9.30% | 102.01% |
| Coiflet 17-11 | 97.78% | 3.04% | 100.83% |
| B-spline 11-5 | 208.57% | 0.55% | 209.12% |

Table 3.4: Energy divisions of lowpass filtered signal

One can't ignore to notice that the high frequency part accommodates quite a bit of the signal's power, even for orthogonal filters. This energy is more or less composed of aliasing components which are removed when the inverse transform is applied. The majority of the selected biorthogonal filters are more efficient at reducing the aliasing components in the high frequency zone.

The experiments done show how a signal's energy spreads out if a signal is only made up of low frequencies. The next step is to find out how this occurs the other way around; if a signal only consist of high frequencies. For orthogonal filters the aliasing should be similar to table 3.3 as the filters are symmetrical to each other, the outcomes from biorthogonal filters should be different.

As previously, we first filter our signal, this time with the aim of removing all low frequencies from it. The inverse filter as that used to filter high frequencies is applied and spectrum of the new signal can be seen in figure 3.7.

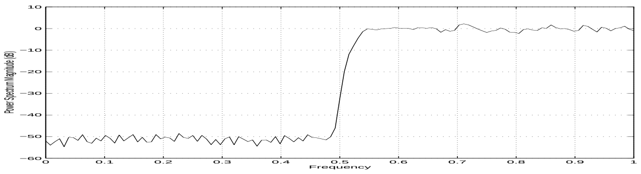

Figure 3.7: Power spectrum of highpass filtered signal

The low frequency area only accommodates about 0.01% of the total energy. As before all the values in the tables are relative to the total energy of the original signal.

The results in table 3.5 and 3.6 follow the same pattern we have already seen. Even here biorthogonal filters generally enlarge the energy in the low frequency zone while the quantity of energy in the high frequency part goes through a reduction. The energy transfer is very significant for many of the biorthogonal filters.

For orthogonal filters the aliasing is limited and follows the same symmetrical pattern we have seen before. The energy located in the low frequency domain is much lower than for biorthogonal filters. One can also witness that a larger orthogonal filter works obviously much better than a shorter filter, giving fewer aliasing components.

Comparing the three simulations it also comes out that certain biorthogonal wavelet filters don't simply shift the energy from high to low zones, but they also try to reduce any effect of

| Orthogonal Filter | Energy L | Energy H | Sum |
|---|---|---|---|
| Haar 2 | 16.34% | 83.56% | 100% |
| Daubechies 8 | 6.78% | 93.22% | 100% |
| Daubechies 10 | 5.77% | 94.22% | 100% |
| Daubechies 12 | 5.05% | 94.94% | 100% |
| Daubechies 20 | 3.37% | 96.62% | 100% |
| Coiflet 6 | 10.51% | 89.48% | 100% |
| Coiflet 12 | 5.04% | 94.95% | 100% |
| Symmlet 6 | 8.25% | 91.74% | 100% |
| Symmlet 12 | 5.04% | 94.95% | 100% |
| Symmlet 16 | 4.04% | 95.96% | 100% |

Table 3.5: Energy divisions of highpass filtered signal

| Biorthogonal Filter | Energy L | Energy H | Sum |
|---|---|---|---|
| Crew 2-6 | 16.34% | 95.36% | 111.70% |
| 5-3 | 25.35% | 71.89% | 97.25% |
| 9-3 | 15.21% | 71.89% | 87.11% |
| CDF 9-7 | 10.66% | 95.34% | 106.01% |
| 6-10 | 12.63% | 77.30% | 89.93% |
| EL 12-4 | 30.43% | 63.22% | 93.65% |
| Bathlets 10-10 0.7 | 8.29% | 86.98% | 92.28% |
| Bathlets 10-10 1.4 | 9.81% | 88.25% | 98.06% |
| Coiflet 17-11 | 7.79% | 93.12% | 100.91% |
| B-spline 11-5 | 89.98% | 56.66% | 146.64% |

Table 3.6: Energy divisions of highpass filtered signal

it with lower gain for the lowpass filters than for the highpass filter. This is noticeable for 2-6, 9-7 and the 10-10 1.4 Bathlet filters. From table 3.2 it is not evident that these filters increase the energy in the low frequency part at all. Table 3.6 on the other hand reveals that there is indeed shifting of energy to low zones which exceeds the aliasing of orthogonal filters of roughly the same size. The only explanation for why the energy increase pretty much goes unnoticed when a signal consist of both low and high frequencies must lay with the fact that the scaling filter for these filters have a lower gain. Low gain for filters can be very suitable if the aim is to compress signals.

From the experiments done so far it is already possible to make some conclusions at this point. For our selected biorthogonal wavelet filters:

- The amount of energy among the low frequencies is increased
- The high frequencies generally get a reduction in their energy compared with orthogonal filters
- As a corollary a signal with both low and high frequencies may get the total energy increased after decomposition as the energy increase in the low frequency zone is not necessary proportional to the energy decrease in the high frequency part

The energy of a signal can be regarded as a rough way of measuring the entropy. Low energy in one frequency zone can therefore imply effective compression of a particular area.

The low frequency zone of a signal decomposed using biorthogonal wavelets consists of a great deal of aliasing values which cause the energy increases. An important lesson to learn from this is that when processing signals with biorthogonal filters one needs to be careful to ensure that the values in the decomposed signal don't change very dramatically. In such case the aliasing components will not recover completely, and may distort a signal when the inverse transform is applied. For some filters the amount of aliasing can be very large (the 11-5 B-spline filter for instance) and when using such filters one should be aware of the negative effects.

To an lesser extent this problem is also valid for orthogonal wavelets filters as tables 3.3 and 3.5 show. For a filter with 8-10 coefficients will roughly 5% of the energy in a decomposed zone consist of aliasing values. Wavelets filters are simply unsuitable for precise filtering operations.

One notable aspect to point out is that our results depend heavily upon the filters we have used in our tests. For orthogonal wavelets the results represent the standard norm as the requirement for orthogonality does not leave much extra room for additional specialities. Biorthogonal wavelets follow a more generalized system and although our filters more or less increased the energy in low frequencies it is very well possible that other types of filters would do the opposite and reduce the energy in the low frequency zone. The conclusions for the high frequency part can then just as well be the opposite of what we reached.
Even though our results are restricted to certain filters they do represent the general line in image compression, one main area where biorthogonal wavelet filters are utilized. We could have reached other results by selecting other filters or by just reversing the analysis and synthesis filters, though, all non-orthogonal filters are likely to split the signal in a non-symmetrical way. Whether the shifting of energy occurs towards lower or higher frequencies is though only relevant for practical applications.

### 3.1.1 Non-orthogonality

The non-orthogonality of filters can directly be investigated using the standard innerproduct formula $\sum\ h[k]\ g[k]$ where $h[k]$ are the filter coefficients for the scaling filter while $g[k]$ are coefficients for wavelet filter. Orthogonal filters always return zero, pointing to an angle of $\frac{\pi}{2}$ between the filters. This formula can't be applied directly to biorthogonal filters as the low/high pass filters don't always have the same number of taps. This can be fixed by symmetrically zero extending the shortest filter. The angle spanned between the lowpass and highpass filters can be found via: $\theta = \cos^{-1} \frac{\sum\ h[k]\ g[k]}{\sum\ h[k]^2 \cdot \sum\ g[k]^2}$.
As it turns out in table 3.7 many of these filters can be considered as being nearly orthogonal to each other, which means that there should be little correlation between the decomposed zones of the signal. Highly non-orthogonal transforms can be unstable but this does not seem to be the problem with any of these filters. None of the filters have an angle above $\frac{\pi}{2}$.

A direct effect of non-orthogonality can also easily be observed. Assuming that $u[k]$ and $v[k]$ are the subsignals decomposed by the filters (but not downsampled) we get:

$$\begin{aligned}\sum(u[k] + v[k])^2 &= \sum(u[k]^2 + 2u[k]\ v[k] + v[k]^2)\\ &= \sum u[k]^2 + \sum v[k]^2 + 2\sum u[k]\ v[k]\end{aligned}$$

For orthogonal filters the cross sum $2u[n]\ v[n]$, ignoring any numerical discrepancies, be zero. Non-orthogonal filters should yield some interesting results which can be related with the energy in the signal filtered with low/high pass filters.

| Biorthogonal Filter | Angle in $\pi$ |
|---|---|
| Crew 2-6 | 0.500 |
| 5-3 | 0.346 |
| 9-3 | 0.374 |
| CDF 9-7 | 0.384 |
| 6-10 | 0.500 |
| EL 12-4 | 0.500 |
| Bathlets 10-10 0.7 | 0.500 |
| Bathlets 10-10 1.4 | 0.500 |
| Coiflet 17-11 | 0.408 |
| B-spline 11-5 | 0.418 |

Table 3.7: Non-orthogonality

Cross sum $\alpha = \sum u[k]\ v[k]$ relative to the total energy of the white-noise signal is in the next table 3.8.

| Biorthogonal Filter | $\alpha$ in percent |
|---|---|
| Crew 2-6 | 11.91% |
| 5-3 | -5.87% |
| 9-3 | 0.11% |
| CDF 9-7 | -3.00% |
| 6-10 | 4.94% |
| EL 12-4 | -4.78% |
| Bathlets 10-10 0.7 | 0.00% |
| Bathlets 10-10 1.4 | 0.00% |
| Coiflet 17-11 | 3.08% |
| B-spline 11-5 | 17.40% |

Table 3.8: Cross sum

Although many of these filters are almost orthogonal the effect of non-orthogonality is clearly visible in table 3.8.

## 3.2 Filter coefficients

The empirical evidence from our tests tells us quite a bit about how orthogonal and biorthogonal filters separate the energy of a signal in respectively low and high frequency parts. Seen from a more theoretical point of view the energy increases/decreases can directly be linked with the filter coefficients. For orthogonal filters we have the requirement $\sum h[k]^2 = 1$ which in frequency domain can be expanded into $|H(\omega)|^2 + |G(\omega)|^2 = 2$. When it comes to biorthogonal wavelets this condition is relaxed into $H(\omega)\tilde{H}^*(\omega) + H(\omega+\pi)\tilde{H}^*(\omega+\pi) = 2$ [3].

Ignoring downsampling, in the continuous case the energy of the decomposed signal can be calculated as

$$\int |Y(\omega)|^2\ d\omega = \int |X(\omega)|^2 |H(\omega)|^2\ d\omega + \int |X(\omega)|^2 |G(\omega)|^2\ d\omega$$

$$= \int |X(\omega)|^2 \;\; (|H(\omega)|^2 + |G(\omega)|^2) \; d\omega$$

The orthogonality relation therefore implies a doubling of the energy. In the biorthogonal case the sum $|H(\omega)|^2 + |G(\omega)|^2$ is non-constant and depends upon $\omega$. The gain may be arbitrarily distributed as long as the conditions for perfect reconstruction are satisfied. We can still find an upper limit in the biorthogonal case by using the maximum value:

$$\int |Y(\omega)|^2 \; d\omega \leq \int |X(\omega)|^2 \Big( (\; \max \; |H(\omega)|)^2 + (\max \; |G(\omega)|)^2 \;\; \Big) \; d\omega$$

$$= \int |X(\omega)|^2 \; ( \; \max |H(\omega)|^2 + \max |G(\omega)|^2 \; ) \; d\omega$$

By making use of $(\sum h[k]^2)^{\frac{1}{2}} \leq \max |H(\omega)|$ or $\sum h[k]^2 \leq \max |H(\omega)|^2$ [13] this can be related with the coefficients of the filters:

$$\int |Y(\omega)|^2 \; d\omega \leq \int |X(\omega)|^2 \; ( \; \sum h[k]^2 + \sum g[k]^2 \; ) \; d\omega$$

Different biorthogonal filters are likely to give different results for the sum as the energy quantity in low/high frequency parts varied quite a bit during our trials in the previous section.

| Biorthogonal Filter | $\sum h[k]^2$ | $\sum g[k]^2$ | Sum |
|---|---|---|---|
| Crew 2-6 | 1.00 | 1.03 | 2.03 |
| 5-3 | 1.43 | 0.75 | 2.18 |
| 9-3 | 1.41 | 0.75 | 2.16 |
| CDF 9-7 | 1.04 | 0.98 | 2.02 |
| 6-10 | 1.28 | 0.80 | 2.08 |
| EL 12-4 | 2.03 | 0.62 | 2.65 |
| Bathlets 10-10 0.7 | 1.06 | 0.94 | 2.00 |
| Bathlets 10-10 1.4 | 1.02 | 0.97 | 2.00 |
| Coiflet 17-11 | 1.07 | 0.94 | 2.01 |
| B-spline 11-5 | 3.28 | 0.54 | 3.82 |

Table 3.9: Filters' energy

The outcomes in table 3.9 agree pretty much with our previous results. Apart from one, all wavelet filters return a value below 1, reduction in energy; while $\sum h[k]^2$ is above unit value, increasing the energy in the lowpass subband. All the filters return a net sum above or equal to 2. A sum equal to 2 does not suggest any increase of the energy, a total above 2 therefore hints towards some redundanc in the decomposing. The values for nearly all of our filters are pretty close to 2, meaning that the amount of correlation and energy increase is small. The 11-5 B-spline filter is the only exception but then this filter is also a theoretical kind of filter rather than one made for practical intentions.

Comparing table 3.9 with table 3.2 we see that they nicely fit in with each other.

From the table it's now possible to give an explanation for the odd behavior of the 2-6 filter as results from this filter are not always in agreement with the other biorthogonal wavelet filters. As one can see in the table this filter does not reduce the energy in any way. The lowpass filter maintains the original power while the high frequency zone receives some propagation. Consequently no direct attempt is done to reduce the signal's power in any band.

### 3.2.1 Overlap

Filter coefficients directly influence on how a signal's energy is spread during decomposing, and virtually all biorthogonal filters in our experiments are designed so they decrease the energy in the high frequency part and increase the energy in the low frequency zone. This is probably what distinguishes between a "good" and a "bad" biorthogonal filter for compression purposes. A good filter should not increase the total energy too much while still reducing the energy in one frequency zone as much as possible.

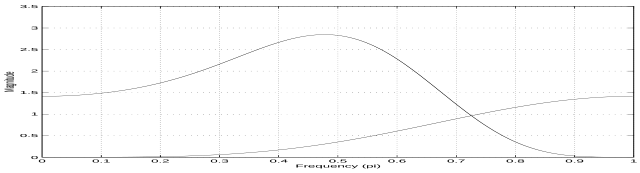

Figure 3.8: Frequency responses of B-spline 11-5

From classical algebra we know that since biorthogonal vector spaces plainly are not orthogonal there can be some correlation between vectors. As the filters are no longer orthogonal one major effect of this is from the overlap area between the low/high frequencies. For orthogonal filters the overlap period is perfectly adjusted, figure 2.12 gives precisely an example of this. Biorthogonal filters don't follow this ideology as one can see from table 3.9. If the overlap period for filters is rather large then certain frequencies are captured by both filters causing correlation and aliasing in the zones. This can be observed in figure 3.8 where the frequency response of the 11-5 B-spline biorthogonal filter is shown. This B-spline filter has so long given us the highest values in the tests and frequency response of this filter reveals the finer details. The wavelet filter has a relative small gain while the scaling filter dominates, covering much of the frequency spectrum.

Adding the frequency response of both analysis filters one ends up with a description of the overall treatment the filters give a signal. The sum of filters magnitudes $|H(\omega)| + |G(\omega)|$ is shown in figure 3.9.

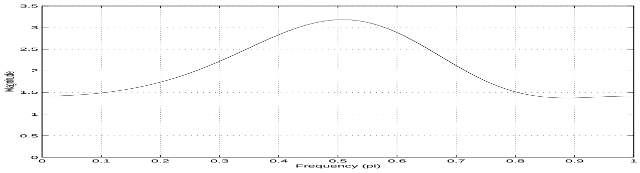

Figure 3.9: Additive frequency response of B-spline 11-5

Subtracting the frequency response of highpass with the lowpass, and the frequency response of the lowpass with the highpass, and making a new function out of this by only keeping the positive values one obtains the total frequency area $|N(\omega)|$ which does not have any overlap at all.

$$|N(\omega)| = \max(\ |H(\omega)| - |G(\omega)|, 0) + \max(\ |G(\omega)| - |H(\omega)|, 0)$$

From the total additive frequency response $|H(\omega)| + |G(\omega)|$ we subtract this non-overlap function and end up with the total frequency part $|O(\omega)|$ which does receive an overlap.

$$|O(\omega)| = |H(\omega)| + |G(\omega)| - |N(\omega)|$$

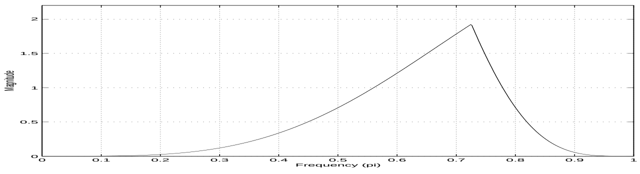

Figure 3.10: Overlap area of the B-spline 11-5

The overlap period of B-spline 11-5 can be seen in figure 3.10. Almost all the high frequencies and as well as low frequencies between $[0.2\pi, 0.9\pi]$ receive a dose of overlapping from this filter combination. Only the very low/high frequencies are exempted.

The 9-7 filter gives another kind of plot in figure 3.11, where the peak point is at $0.54\pi$.

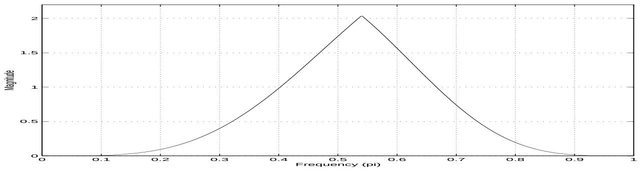

Figure 3.11: Overlap area of biorthogonal 9-7

If we repeat the process with the orthogonal Daubechies 8 filter the total overlap area has a slightly different shape which is symmetrical exactly in the middle at $0.50\pi$.

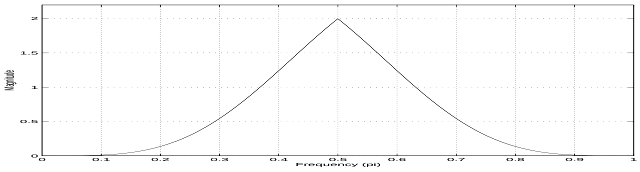

Figure 3.12: Overlap area of Daubechies 8

In the center a signal receives a unit gain from both the low and highpass filter which is also clear from the plot. For other orthogonal filters we will acquire the same results.

This is an alternative way to look at why some biorthogonal filters shift the energy of signals in special ways. For orthogonal filters the peak point is always in middle and frequencies under and above $\frac{\pi}{2}$ obtain an equal amount of aliasing. If the aim is to increase the energy in the low frequency subband then one can shift the overlap area towards higher frequencies. The scaling filter can then capture more energy from the higher frequencies while the wavelet filter is limited to certain high frequencies.

One can also observe that the B-spline 11-5 filter is not only shifted towards right but is also very unsymmetrical. A short overlap period on one side makes sure that few frequencies from that zone are aliased over to the other side. Additionally, the more the overlap period is towards right, the more the scaling filter captures from the higher frequencies causing aliasing and energy increase in the low frequency zone. The peak point of nearly all of the biorthogonal filters used in our trials is therefore likely to be shifted towards greater frequencies.

To find more about such properties of the filters new tests were performed where we found the extreme point of the overlap area as well as the energy below ($E_{P-}$) and above this peak point ($E_{P+}$). As the overlap area is not likely to be symmetrical around any particular point the energies above and below the peak point can be divided with each other to get an idea about the relationship with each other: $R_{LH} = E_{P-}/E_{P+}$. A high value involves that the filter pair captures most energy from the low frequencies; a small figure implies that majority of the energy in the overlap area is located towards higher frequencies.

| Orthogonal Filter | Peak ($\pi$) | Energy $\leq$ P | Energy $\geq$ P | $R_{LH}$ |
|---|---|---|---|---|
| Haar 2 | 0.50 | 1.46 | 1.46 | 1.00 |
| Daubechies 8 | 0.50 | 0.73 | 0.73 | 1.00 |
| Daubechies 10 | 0.50 | 0.65 | 0.65 | 1.00 |
| Daubechies 12 | 0.50 | 0.59 | 0.59 | 1.00 |
| Daubechies 20 | 0.50 | 0.46 | 0.46 | 1.00 |
| Coiflet 6 | 0.50 | 1.01 | 1.01 | 1.00 |
| Coiflet 12 | 0.50 | 0.71 | 0.71 | 1.00 |
| Symmlet 6 | 0.50 | 0.84 | 0.84 | 1.00 |
| Symmlet 12 | 0.50 | 0.59 | 0.59 | 1.00 |
| Symmlet 16 | 0.50 | 0.51 | 0.51 | 1.00 |

Table 3.10: Peak points

Looking at table 3.10 one can confirm that there is a perfect balance between the low and high-pass filters.

| Biorthogonal Filter | Peak ($\pi$) | Energy $\leq$ P | Energy $\geq$ P | $R_{LH}$ |
|---|---|---|---|---|
| Crew 2-6 | 0.50 | 0.82 | 1.46 | 0.56 |
| 5-3 | 0.63 | 1.25 | 0.72 | 1.72 |
| 9-3 | 0.60 | 1.00 | 0.43 | 2.31 |
| CDF 9-7 | 0.54 | 0.78 | 0.71 | 1.10 |
| 6-10 | 0.56 | 0.84 | 0.66 | 1.26 |
| EL 12-4 | 0.65 | 0.77 | 0.28 | 2.71 |
| Bathlets 10-10 0.7 | 0.51 | 0.82 | 0.78 | 1.05 |
| Bathlets 10-10 1.4 | 0.50 | 0.95 | 0.93 | 1.01 |
| Coiflet 17-11 | 0.53 | 0.64 | 0.51 | 1.24 |
| B-spline 11-5 | 0.72 | 0.87 | 0.29 | 2.94 |

Table 3.11: Peak points

Unlike orthogonal filters one can not observe any kind of symmetry in table 3.11 at all. The table shows that the 5-3 filter for example is slightly more shifted towards right than the 9-3 filter. The analysis scaling filter in the 5-3 pair should hence capture more of the high frequency energy and result in an increase in the low frequency zone when compared to the 9-3 variant. If we check this hypothesis with table 3.1 then we find that this is indeed the case.

## 3.3 Correlations

It is possible to clearly see the effects of biorthogonal filters by just examing the filters it can also be of interest to find the correlation properties of the decomposed signals. This can be done via statistical methods where autocorrelation and crosscorrelation functions are used on the decomposed signals. A crosscorrelation between the low and high frequency parts of a signal decomposed using wavelets transform describes the general dependence of low frequency values with high frequency values. An orthogonal wavelet transform should not yield any correlation but this may not be so for a biorthogonal transform according to the traditional theory and corresponding to our results from the previous sections for some filters.

Taking autocorrelation and crosscorrelations returns new vectors, however, it is much more convenient to have a simple numerical value as an indication of the average correlation. For autocorrelations the average correlation length can be calculated via [1]:

$$l_{c[n]} = \frac{1}{r_c(0)} \sum_{k=1} k \; r_c[k]$$

where $r_c[n] = \frac{1}{N} \sum c[k]c[n-k]$ is the autocorrelation of $c[n]$. A low value implies low correlation while a greater number indicates higher amount of correlation. Uncorrelated white noise signals should therefore ideally have zero average length.
The average correlation length although useful may not always give us a true picture; a long autocorrelation sequence with many small numbers can give the same result as a sequence mostly composed of zeros but a few large values.

For crosscorrelation $r_{cd}[n] = \frac{1}{N} \sum c[k]d[n-k]$ between $c[n]$ and $d[n]$ we need other methods as the formula above is not suitable. One possibility is to take the maximum value at any particular lag:

$$\max \{ \; |r_{cd}[n]| \; \}$$

The crosscorrelation coefficient can also be calculated via:

$$\zeta = | \sum (c[k] \; d[k]) \;\; / \;\; ( \;\; \sum \; (c[k] \; c[k]) \;\; \sum \; (d[k] \; d[k]) \;\; )^{\frac{1}{2}} |$$

As a starting point an orthogonal wavelet transform of our test signal is taken. The autocorrelation of both the low and high frequency zones along with a crosscorrelation can be used to see whether any correlations show up. A figure of merit can be defined $F = \frac{l_{d[n]}}{l_{c[n]}}$ which is the average correlation length of the high frequency part divided by the average correlation length of the low frequency part.

Our original signal only consist of 5000 white noise samples and as it turns out this is not adequate at all to numerically estimate autocorrelations and crosscorrelations with high degree of accuracy. A new signal with 50000 samples was therefore created for use in these statistically oriented experiments.

Taking the new white noise signal and decomposing it using the orthogonal Daubechies 8 filter and calculating the autocorrelation sequences of the low and high frequency zones we obtain figure 3.3 on a normalized scale.

The autocorrelation sequences are smoothed with a Bartlett window but as expected the sequences do not indicate any correlation within the high and low frequency areas. To reduce the effects of numerical errors all the values below 10% of the sequence average value are also truncated. Even so the average correlation length then amounts to 5733.05 and 6345.48 respectively for the low and high frequency sequences. Dividing the high frequency average length with the low frequency average length we acquire $F = 1.10$ which is very close to one. Noise and small correlations in the data can explain the slight off value. This figure we calculate after

[1] This is the discrete version of the ratio $\tau_c = \frac{1}{C(0)} \int_0^{\infty} C(\tau) \; d\tau$ found in for example [14]

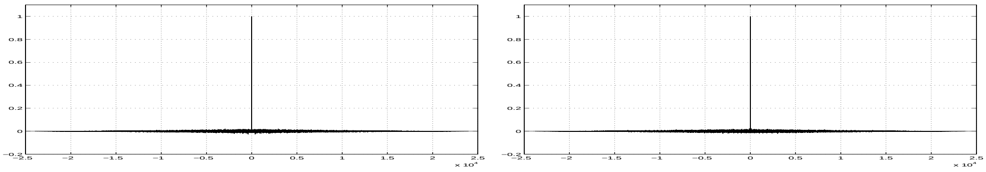

Figure 3.13: L: Autocorrelation low $r_c$; R: Autocorrelation high $r_d$

dividing the average length of the high frequency zone with the low frequency average length is the figure of merit which indicates the amount of average correlation. A low value close to one points to low correlation.

The crosscorrelation between the low and high frequency decomposed parts also does not further suggest any correlation in figure 3.14. Notice the different scaling used for crosscorrelation plots.

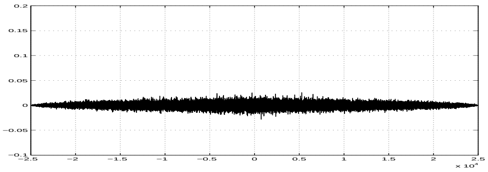

Figure 3.14: Crosscorrelation $r_{cd}$

The correlation coefficients comes at $\zeta = 0.005$; again not indicating any great degree of dependency.

Let us now turn our focus from orthogonal wavelet filters towards biorthogonal wavelet filters where more interesting results should be expected. In the first phase the 9-7 biorthogonal filter is tested.

Repeating the identical procedure for biorthogonal filters and taking autocorrelations between the subbands along with the crosscorrelation we obtain somewhat different results.

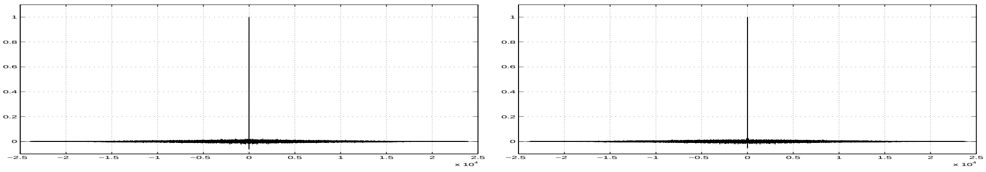

Figure 3.15: L: Autocorrelation low $r_c$; R: Autocorrelation high $r_d$ (9-7)

Figure of merit turns out to be $F = 1.19$, just slightly greater than 1.10 which we obtained under the Daubechies 8 filter.

From the plot of the crosscorrelation function one can notice a few above average high values in the middle of the plot. This indicates that there is indeed some correlation between the low and high frequency parts of the decomposed signal at lags around 0. The absolute correlation coefficient adds up to $\zeta = 0.031$. This value is somewhat larger than the figure obtained with the orthogonal filter and is also pointing towards more correlation in the biorthogonal case.

Results from further experiments are shown in tables 3.12 and 3.13. To get more accurate readings for the crosscorrelations coefficients these values are calculated by taking the average value

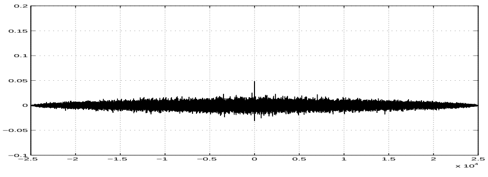

Figure 3.16: Crosscorrelation $r_{cd}$ (9-7)

over 1000 runs, where each run consisted of a different white noise sample of length 5000.

| Orthogonal Filter | Merit | $\max \|r_{LH}\|$ | $\zeta$ |
|---|---|---|---|
| Haar 2 | 1.12 | 0.022 | 0.001 |
| Daubechies 8 | 1.10 | 0.028 | 0.000 |
| Daubechies 10 | 1.12 | 0.023 | 0.000 |
| Daubechies 12 | 1.12 | 0.026 | 0.002 |
| Daubechies 20 | 1.12 | 0.024 | 0.002 |
| Coiflet 6 | 1.10 | 0.024 | 0.001 |
| Coiflet 12 | 1.11 | 0.025 | 0.000 |
| Symmlet 6 | 1.11 | 0.023 | 0.001 |
| Symmlet 12 | 1.13 | 0.023 | 0.001 |
| Symmlet 16 | 1.12 | 0.023 | 0.000 |

Table 3.12: crosscorrelations

| Biorthogonal Filter | Merit | $\max \|r_{LH}\|$ | $\zeta$ |
|---|---|---|---|
| Crew 2-6 | 1.10 | 0.122 | 0.000 |
| 5-3 | 2.11 | 0.060 | 0.003 |
| 9-3 | 2.10 | 0.024 | 0.002 |
| CDF 9-7 | 1.19 | 0.048 | 0.009 |
| 6-10 | 1.78 | 0.051 | 0.003 |
| EL 12-4 | 3.65 | 0.051 | 0.003 |
| Bathlets 10-10 0.7 | 1.26 | 0.023 | 0.003 |
| Bathlets 10-10 1.4 | 1.18 | 0.023 | 0.001 |
| Coiflet 17-11 | 1.28 | 0.031 | 0.007 |
| B-spline 11-5 | 6.59 | 0.130 | 0.052 |

Table 3.13: crosscorrelations

For biorthogonal filters our figure of merit for the most parts returns a larger value than for orthogonal filters. The maximum correlation amount can also be said to be higher for some biorthogonal filters compared with orthogonal filters. The B-spline filter gives the highest amount for the maximum correlation while the figure of merit is also very high. This is consistent with our result from for example table 3.2 and table 3.9 where this filter gave the highest amount of net energy after signal decomposition. Looking at the crosscorrelation coefficients one can observe that a few filters give slightly large values but even for many biorthogonal filters the figures are moderate. The 2-6 filter gives somewhat ample value for the maximum correlation just as in table 3.7 it also gave a large value for the cross sum.

The crosscorrelation taken between the low and high frequency part of the signal decomposed using the B-spline filter is shown in the next figure:

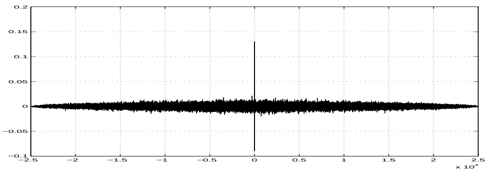

Figure 3.17: Crosscorrelation $r_{cd}$ (B-spline 11-5)

In many of our experiments biorthogonal filters did not increase the total energy of the signal by big ratios, for that reason the correlations we find here are indeed minimal. Two major exemptions from table 3.2 are the EL 12-4 and the B-spline 11-5 filters who both increase the energy by very noticeable figures. This shows up very clearly in this new table where the figure of merit is indeed much higher for these biorthogonal wavelet filters than for other filters. The 5-3 and 9-3 filters can also be regarded as filters who increase a white noise signal's energy by around 10%, in this case the merit does stand out from the rest by higher values.

## 3.4 Summary

The purpose of this chapter was to investigate some properties of certain known biorthogonal filters by means of practical experiments. Quite often regularity and the number of vanishing moments are regarded as the most interesting aspects behind wavelet filter banks, however, the energy shifts in biorthogonal filters also seem to play an important role. The correlation between the decomposed signal on the other hand turned out to be minimum for the majority of biorthogonal filters. The effects of correlations therefore do not seem to have any importance when biorthogonal wavelets are preferred over orthogonal variants, the energy shifting properties can depending upon the application have a much more noticeable function.

# Chapter 4

# Filter Optimization

In the preceding chapter certain properties of wavelets filters were examined and we found that many biorthogonal wavelet filters attempted to shift the energy of a signal into the low frequency subband. This compaction varied from filter to filter, however, the majority of filters only tried to shift a modest fixed amount of energy from higher frequencies to the low frequency zone. For optimal compression the ultimate goal should be to transfer as much energy as possible from the high frequency zone into the low frequency area. Additionally the total energy increase for the low frequency area should be small.

In case of biorthogonal filters it should be possible to explore the extra degrees of free parameters and construct filters optimized for moving as much energy of a signal to the low frequency zone after a wavelet filter decomposition. This chapter will be concerned with this problem, designing filters which would try to shift and shrink the energies of a signal.

Symmetrical biorthogonal filter banks must satisfy certain constrains placed upon them to be called wavelet filters in the first place and also to ensure properties like perfect reconstruction of signals. The main constrains from chapter 2 can be summed up as:

$$\sum h[n] = \sqrt{2} \tag{4.1}$$

$$\sum \tilde{h}[n] = \sqrt{2} \tag{4.2}$$

$$\sum h[2n] = \sum h[2n+1] \tag{4.3}$$

$$\sum \tilde{h}[2n] = \sum \tilde{h}[2n+1] \tag{4.4}$$

$$\sum h[n]\tilde{h}[n+2k] = \delta[k] \tag{4.5}$$

Equations (4.1) and (4.2) ensure the convergence of scaling and wavelets functions to a compactly supported basis. For even length symmetrical filters (4.3) and (4.4) are automatically satisfied and the sum also adds up to $\frac{1}{\sqrt{2}}$. Equation (4.5) is the key for biorthogonality, however, this also results in unlinear equations.

## 4.1 4-4 filters

Taking (4.1), (4.2) and (4.5) one can go ahead and try to construct parameterized biorthogonal wavelets filter with 4 taps for both the analysis and synthesis filters.

$$h[n] = [a\ b\ b\ a] \qquad \tilde{h}[n] = [c\ d\ d\ c]$$

Using the previous conditions one comes up with a non-linear system of equations:

$$\begin{aligned} 2a + 2b &= \sqrt{2} \\ 2c + 2d &= \sqrt{2} \\ 2ac + 2bd &= 1 \\ ad + bc &= 0 \end{aligned}$$

We have one degree of freedom in this system and it should be possible to use this to some good effect. In many wavelets systems such liberations are used to maximize the number of vanishing moments for wavelets filters and/or to make the frequency response as flat as possible. Smooth wavelets are generally considered very important but as we only have one degree of freedom it could be better to use this for other purposes than maximizing the numbers of vanishing moments. By fixing $a$ one can directly compute the figures for the other coefficients:

$$b = \tfrac{\sqrt{2}-2a}{2},\ \ c = \tfrac{a}{2\sqrt{2}a-1},\ \ d = \tfrac{\sqrt{2}}{2} - \tfrac{a}{2\sqrt{2}a-1}$$

The highpass filters $g[n]$ and $\tilde{g}[n]$ can easily be found by inverting the lowpass filters as they are crossrelated with each other. By varying $a$ one ends up with different biorthogonal wavelet filters each with different properties. It is also possible to use the free variable to place a zero point for the analysis scaling filter at an arbitrary place $z_0$ on the complex plane; taking z-transform of $h[n]$ gives: $a + bz_0^{-1} + bz_0^{-2} + az_0^{-3} = 0$. As we also have (4.2) to think about we end up with a linear system which must be solved to obtain coefficients for the analysis scaling filter:

$$\begin{aligned} (z_0^3 + 1)a + (z_0^2 + z_0)b &= 0 \\ 2a + 2b &= \sqrt{2} \end{aligned}$$

From the first equation in the system it is also easy to observe that the zero point $z_0$ can not be placed at any arbitrary position in the complex plane unless one is willing to have complex filter coefficients. To avoid this $z_0$ must be restricted to the real axis line alone or be placed somewhere on the unit circle. Some examples of the analysis scaling function and the wavelets generated by placing $z_0$ at different places computed using successive approximations are shown in figure 4.1. Setting $z_0$ to a large positive/negative value on the real axis line makes Haar-like filters, while by placing a zero on the left side on the unit circle very smooth analysis wavelets can be created. The synthesis wavelet function on the other side may not reflect any of this smoothness at all, as a matter of fact a smooth analysis wavelet function in many cases gives a very non-regular kind of synthesis wavelet function as the coefficients are related with each other in a negative way. Regularity, however, is not everything, for non-stationary signals a non-regular approximation might just as well turn out to be more suitable.

As we found in chapter 3 the filter coefficients play a major role in shifting the energies of a signal. By minimizing $\sum g[k]^2$ the amount of energy in the high frequency zone after a wavelet decomposition can be reduced quite heavily. The free variable in our system is $a = h[0]$ while we would prefer to directly minimize the wavelet filter $g[n] = [-c\ d\ -d\ c]$. Switching the role of the analysis/synthesis filters and moving the free variable to the synthesis scaling filter makes it easier to solve this problem.

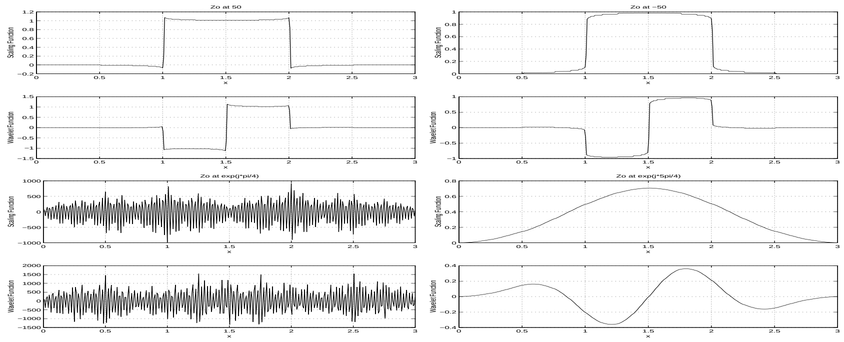

Figure 4.1: Examples of 4-4 scaling and wavelet functions

This transfers the problem into minimizing the energy of $\hat{g}[n] = [-a \ b \ -b \ a]$; $\min \sum \hat{g}[k]^2 = \min \quad a^2 + b^2 = a^2 + \frac{1}{4}(2 - 4\sqrt{2}a + 4a^2)$

Derivation by $a$ and setting the equation equal to 0 gives $8a - 2\sqrt{2} = 0$. The optimized value for the function is hence $a = \frac{1}{4}\sqrt{2}$. This gives the filter zeros at $e^{j\pi/2}$ and it's conjugate value $e^{j3\pi/2}$; albeit it is not possible to construct $h[n]$ out from $a = \frac{1}{4}\sqrt{2}$ due to divisions by zero. The total gain for $g[n]$ is indeed minimized but at the same time the gain for $h[n]$ is maximized to infinity. This shows the classical dilemma between filters which are related with each other. Low gain in one filter automatically relates over to high gain on the other.

Nevertheless, this does give us some hints on how the zero points can be placed to reduce the gain as much as possible. Due to the fact that the filters are related with each other the value $a = \frac{1}{4}\sqrt{2}$ is equivalent with a zero in the lowpass filter $h[n]$ at $e^{j0}$. A zero around $1$ ensures that the scaling filter captures as much energy as possible from the higher frequencies and the wavelet filter has a low gain as possible. In figure 4.2 the effect of placing a zero point relative close to $1$ on $e^{j0.1\pi}$ and at a distance apart at $e^{j0.25\pi}$ can be seen. The increase in the lowpass filter's total gain is quite large. For best results a compromise must be obtained which should ideally be suited to each particular signal.

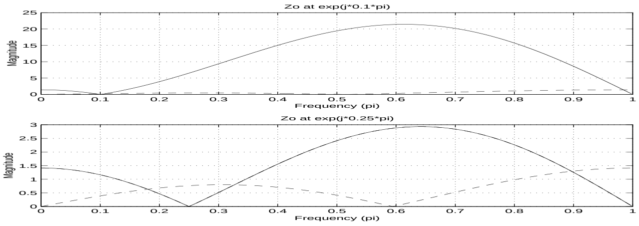

Figure 4.2: Frequency responses of low (solid) and highpass (dashed) filters

For $a = \frac{1}{4}\sqrt{2}$ we find $\sum \hat{g}[k]^2 = 0.5$, which is the lowest achievable coefficients energy for this kind of filter. This is though useless so rather than minimizing $\sum \hat{g}[k]^2$ another option is to minimize the sum of the filters coefficients energy $\sum(h[k]^2 + g[k]^2)$. As all the coefficients depend solely upon $a$ minimizing is straightforward: $\min \quad a^2 + b^2 + c^2 + d^2 : \quad (a^2 + b^2 + c^2 + d^2)' = 4a - \sqrt{2} - \frac{8\sqrt{2}a^2 + 4a}{(8a^2 - 4\sqrt{2}a + 1)^2} + \frac{2}{8a^2 - 4\sqrt{2}a + 1} = 0$

From the real extrema points $a = 0$ and $a = \frac{\sqrt{2}}{2}$ we get orthogonal Haar filters! This should not come as a total surprise as orthogonal wavelets obeying Parseval theorem preserve the energy of the signal after decomposition. The filter's energy sum likewise gives the standard result: $\sum(h[k]^2 + g[k]^2) = 2$.

The two procedures gone through above to minimize $\sum g[k]^2$ and $\sum(h[k]^2 + g[k]^2)$ are very general and both try to reduce the overall gain of the filters to achieve best possible outcomes for all kinds of signals.

If the aim is to transfer all of a given signal's energy to the low frequency subband then it should be possible to explore the structure of the signal and place the zero points of the wavelet filter at exactly the same frequency locations as the frequencies which dominate in the signal. The highpass wavelet filter is then not able to capture these elevating frequencies while the lowpass filter is forced to approximate them.

Wavelet filters with 10 taps have 4 degrees of freedom, two of them can for example be used to place zeros in the wavelet filter while the other free variables are used to dampen the gain of the scaling filter. Even longer filters give more flexibility but with this kind of setup one is definitively moving far away from the traditional multiresolution view of wavelet analysis.

For our 4-4 filter we only have one degree of freedom and this can be used to place a zero point in the wavelet filter around the dominating frequency in a signal. An artificial signal with one low frequency at $0.2\pi$ and an high frequency of $0.7\pi$ was created to test how this method of nulling certain frequencies can work. The signal and it's power spectrum are shown in figure 4.3 and 4.4 respectively.

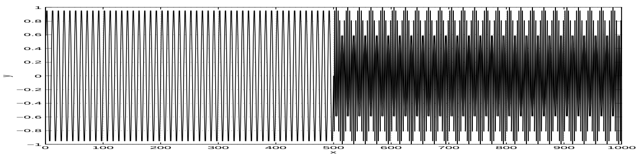

Figure 4.3: test signal P1

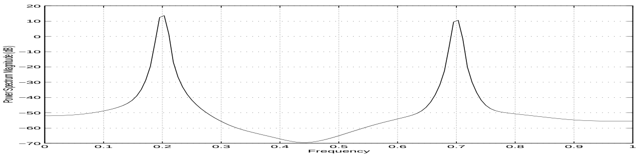

Figure 4.4: power spectrum of P1

For the analysis wavelet filter we decide to place a zero at $0.7\pi$ and as one can observe from the plots of frequency responses in figure 4.5 this has certain drawbacks. Although the wavelet filter does not capture much of the dominating high frequency it still grabs quite a lot from the lower frequencies. The overall net result can hardly be better than the traditional filters.

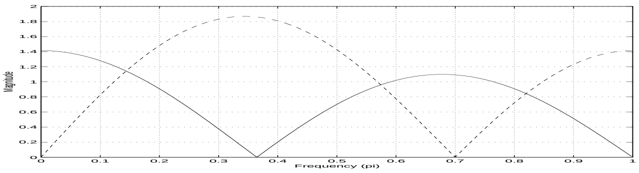

Figure 4.5: Frequency responses of low (solid) and highpass (dashed) filters

Calculating the energy of the decomposed signal's low and high frequency zones we find that this is indeed the case. Table 4.1 shows the result along with the outcomes from the more traditional filters.

| Filter | Energy L | Energy H | Sum |
|---|---|---|---|
| 4-4, $z = e^{j0.7\pi}$ | 50.48% | 54.72% | 105.20% |
| Daubechies 8 | 51.72% | 48.27% | 100% |
| CDF 9-7 | 48.99% | 50.97% | 99.96% |

Table 4.1: Energy divisions of test signal P1

The wavelet filter obviously needs more damping around the lower frequencies.

## 4.2 6-6 filters

The same methods as applied in the previous section can be extended to create a 6-6 biorthogonal wavelet system. With 6 filter coefficients two extra degrees of freedom is available, giving more flexibility. Learning from the previous results we this time place both the free variables at the synthesis lowpass filter rather than the analysis scaling filter.

$$h[n] = [k\ l\ m\ m\ l\ k] \qquad \tilde{h}[n] = [a\ b\ c\ c\ b\ a]$$

Keeping $a$ and $b$ fixed the value for $c$ can be calculated to be $c = \frac{\sqrt{2}}{2} - a - b$. Using equations (4.2) and (4.5) it is easy to find that we must solve the following linear system to obtain the coefficients for the analysis lowpass filter $h[n]$:

$$2\,k + 2\,l + 2\,m = \sqrt{2}$$
$$\tfrac{2}{b}(b^2 - a^2)\,l + 2c\,m = 1$$
$$\tfrac{c}{b}(b - a)\,l + (a + b)\,m = 0$$

Rather than experimenting with $a$ and $b$ it is easier to use the free variables to directly place two zero points $z_0$ and $z_1$ at given places on the complex plane. In this case the coefficients for filters also always end up being real if the location of zero points is conjugate symmetrical.

To place the zero points at $z_0$ and $z_1$ for the synthesis lowpass filter another linear system must be solved, which can return the coefficients for $\tilde{h}[n]$:

$$2a + 2b + 2c = \sqrt{2}$$
$$(z_0^5 + 1)a + (z_0^4 + z_0)b + (z_0^3 + z_0^2)c = 0$$
$$(z_1^5 + 1)a + (z_1^4 + z_1)b + (z_1^3 + z_1^2)c = 0$$

Since the zeros are placed at the synthesis scaling filter $\tilde{h}[n]$ they are negative conjugate symmetrical to the zero points for the analysis wavelet filter $g[n]$ as the filters are inverse related with each other.

We can now re-run our previous experiment with 6-6 filters where as well as nulling out $0.7\pi$ we also place a zero in the wavelet filter at $0.2\pi$; the dominating low frequency in the signal. This extra zero should improve the performance of the wavelet filter to some extent. Looking at the frequency response of the optimized filters (figure 4.6) the large gain in the scaling filter none the less is most noticeable.

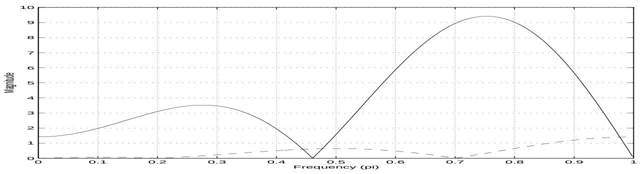

Figure 4.6: Frequency responses of low (solid) and highpass (dashed) filters

| Filter | Energy L | Energy H | Sum |
|---|---|---|---|
| 6-6, $z_0 = e^{j0.2\pi} z_1 = e^{j0.7\pi}$ | 2228.38% | 0.13% | 2228.51% |

Table 4.2: Energy divisions of test signal P1

In table 4.2 one can see that the wavelet filter is very successful at contracting the quantity of energy in the high frequency domain but the over 2000% energy increase in the low frequency subband doesn't looks so attractive. This at least does show that it is possible to construct adaptive wavelet filters which can reduce the total energy in one frequency zone to a bare minimum.

The large gain in the scaling filter can be reduced if we put more zero points around the middle frequencies. One can increase the length of the filters and try to use the extra degrees of freedom for the scaling filter. However, as we now already have an acceptable working wavelet filter all the focus can be turned over to the lowpass filter and the length of the scaling function alone can be increased.

One important desirable property which we have ignored so far concerns the smoothness of our wavelets. Highly non-smooth functions can distort a signal quite dramatically during numerical quantization processes.

Using the recursive equations the scaling and wavelet functions for both analysis and synthesis can be computed, for the optimized 6-6 filter pair they are shown in figure 4.7.

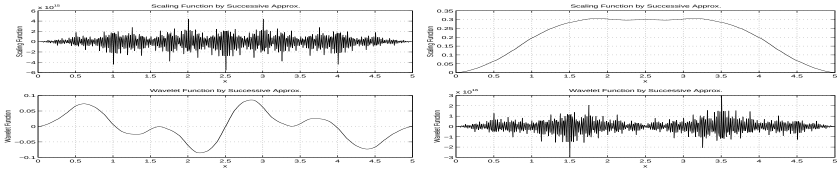

Figure 4.7: Scaling and Wavelet functions, L:Analysis R:Synthesis

The analysis wavelet and synthesis scaling functions are reasonably smooth but the counter functions are showing a totally different picture. We see a very high frequency pattern in them which is a direct result of our attempts to force the scaling function to approximate special frequencies. The biggest disadvantage with $\phi(t)$ and $\tilde{\psi}(t)$ is the instability, resulting in exceedingly enlarged values as the cascade algorithm iterates.

The functions generated via $h[n]$ can make use of some more regularity and the substantial values need to be reduced as well as. As the large values are related with the high gain in the filter's frequency response this can be reduced by placing more zero-points while regularity can be increased by demanding[3]:

$$\sum n^K \tilde{g}[n] = 0 \tag{4.6}$$

For $K = 0$ this equation must be satisfied in order to generate proper wavelet filters while for odd length filters it is automatically required to be satisfied for $K = 1$ due to (4.3) and (4.4). Increasing the regularity of filters and adding extra equations up to $K = 2$ or $K = 3$ should improve the regularity.

## 4.3 26-6 filters

Keeping the 6-part fixed we now increase the length of $h[n]$ directly to 26. This gives 5 extra degrees of freedom for the lowpass analysis filter and one can use them both to increase the regularity and set extra zero points at various positions. A filter of length 26 is quite large although the average numbers of coefficients for this system is no more than 16, which is a little less dramatical.

Assuming the analysis scaling filter has a format like

$$h[n] = [k\ l\ m\ n\ o\ p\ q\ r\ s\ t\ u\ v\ w\ w\ v\ u\ t\ s\ r\ q\ p\ o\ n\ m\ l\ k]$$

the linear system which must be solved to make this is a proper wavelet system is:

$$\begin{gathered}
2k + 2l + 2m + 2n + 2o + 2p + 2q + 2r + 2s + 2t + 2u + 2v + 2w = \sqrt{2} \\
2a\ u + 2b\ v + 2c\ w = 1 \\
a\ s + b\ t + c\ u + c\ v + (b + a)\ w = 0 \\
a\ q + b\ r + c\ s + c\ t + b\ u + a\ v = 0 \\
a\ o + b\ p + c\ q + c\ r + b\ s + a\ t = 0 \\
a\ m + b\ n + c\ o + c\ p + b\ q + a\ r = 0 \\
a\ k + b\ l + c\ m + c\ n + b\ o + a\ p = 0 \\
(c - \tfrac{bc}{a})\ k + b\ m + a\ n = 0
\end{gathered}$$

We have a total of 13 variables and 8 equations above, hence 5 free parameters. There are many different ways one can utilize the freedom of this system, the most simple method is to simply place zeros arbitrary on the unit circle and try to reduce the scaling filters high gain.

Placement of a zero at $z_0$ gives arise to a new equation:

$$(z_0^{25} + 1)\ k + (z_0^{24} + z_0)\ l + (z_0^{23} + z_0^2)\ m + (z_0^{22} + z_0^3)\ n + (z_0^{21} + z_0^4)\ o + (z_0^{20} + z_0^5)\ p + (z_0^{19} + z_0^6)\ q + (z_0^{18} + z_0^7)\ r + (z_0^{17} + z_0^8)\ s + (z_0^{16} + z_0^9)\ t + (z_0^{15} + z_0^{10})\ u + (z_0^{14} + z_0^{11})\ v + (z_0^{13} + z_0^{12})\ w = 0$$

We can use five forms of the above equation with different zero points and try to observe the effects of this. If we for example just place some zeros at $e^{j0.1\pi}, e^{j0.4\pi}, e^{j0.6\pi}, e^{j0.85\pi}$ and $e^{j0.9\pi}$ then one is covering pretty much of the frequency area and this may help reducing the filter's overall gain. This method, however, fails badly and we get a matrix which is very close to being singular. The coefficients we end up with are therefore very large and completely unusable.

### 4.3.1 Minimizing $\sum h[n]^2$

An alternative to manually positioning zeros is to use an optimization method and reduce the energy of $\sum h[n]^2$ to a minimum. There are many different optimization techniques available and variants of sequential quadratic programming [2] can be used for such problems. A filter with low coefficient energy is hence constructed.

Using the 8 previous equations we let the optimization routine select the free variables in the best possible way. Solving $\min \sum h[n]^2$ numerically with the 8 given constrains we obtain a scaling filter which for the most part does have a low overall gain:

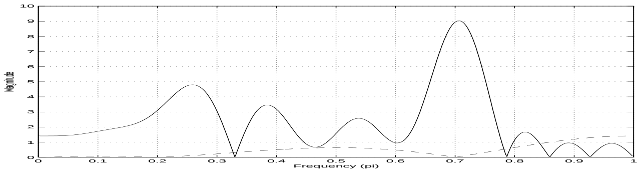

Figure 4.8: Frequency responses of optimized low (solid) and highpass (dashed) filters

This scaling filter still has a large gain at $0.7\pi$ which is a disadvantage as half of our signal consist of this frequency. After decomposing our signal with this optimized filter we find that the energy increase in the lowpass zone is still substantial although better than with our previous 6-6 filter.

| Filter | Energy L | Energy H | Sum |
|---|---|---|---|
| 26-6, Optimized$_1$ | 2203.50% | 0.13% | 2203.64% |

Table 4.3: Energy divisions of test signal P1

The problem with optimizing $\sum h[n]^2$ is that this is totally independent of the signal itself, and the gain at $0.7\pi$ has a magnitude of 9.02. For our purpose we would prefer a low gain at the dominating frequencies; at both $0.2\pi$ and $0.7\pi$.

### 4.3.2 Minimizing $\max |H(\hat{\omega})|$

As the filters return a very large gain at certain frequencies one can concentrate on this alone and attempt to reduce the maximum value $|H(\hat{\omega})|$ ever obtains. Changing to this optimization criteria and solving this numerically we find that the new analysis scaling filter is not so dramatically different though. The maximum magnitude gain is just a little lower at 8.92.

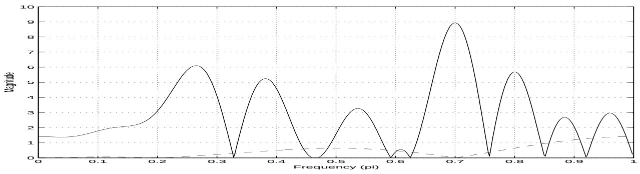

Figure 4.9: Frequency responses of optimized low (solid) and highpass (dashed) filters

Decomposing the artificial signal one finds a little improvement (table 4.4).

| Filter | Energy L | Energy H | Sum |
|---|---|---|---|
| 26-6, Optimized$_2$ | 2196.31% | 0.13% | 2196.45% |

Table 4.4: Energy divisions of test signal P1

### 4.3.3 Minimizing with $|H(\hat{\omega}_0)| \leq r$

So far the methods have only tried to minimize with regards to the filter coefficients without directly considering the signal itself. If one therefore adds an extra inequality to the system such as $|H(0.7\pi)| \leq 4$ then the gain at this dominating frequency can be decreased quite heavily. If necessary, one can even put such limits upon other frequencies such as $|H(0.2\pi)| \leq 4$ as well as. Nevertheless, the major drawback with this method can be severely increased gain at other frequencies.

Adding the extra $|H(0.2\pi)| \leq 4$ constrain and still trying to minimize the maximum gain we find that as expected despite low gain at certain frequencies the overall gain is very large.

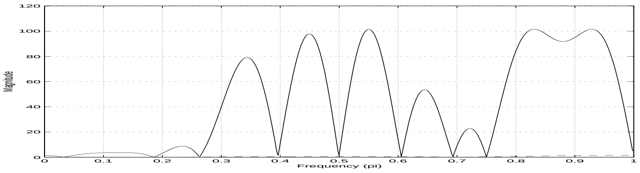

Figure 4.10: Frequency responses of optimized low (solid) and highpass (dashed) filters

This increased gain also delivers scaling and wavelet functions with infinite energy; making such an optimization rather unpalatable. In spite of low gain at the dominating frequencies the decomposition process gets influenced by the large gain as we see in table 4.5.

| Filter | Energy L | Energy H | Sum |
|---|---|---|---|
| 26-6, Optimized$_3$ | 3573.21% | 0.13% | 3573.35% |

Table 4.5: Energy divisions of test signal P1

Much of the increase can be related to the start of the signal where the filter uses zero padding for the signal and the noise which arises in the signal exactly at the frequency change.

There are many other variants over this contraction method. One interesting alternative is to change the minimization criteria to $\min\,(|H(0.2\pi)|^2 + |H(0.7\pi)|^2)$, even so one experiences the same problems as above; with very large gains generally dominating the frequency response.

### 4.3.4 Minimizing $\sum(|X(\hat{\omega})||H(\hat{\omega})|)^2$

The best possible minimization method should take into account all of the frequencies in the signal and construct a filter out from that which gives as low gain as possible so the decomposed signal's energy is minimized.

Such a filter can be created if we alter the objective function into $\sum(|X(\hat{\omega})||H(\hat{\omega})|)^2$ and hence try to minimize the energy of the signal, with discrete Fourier transform $X(\hat{\omega})$, multiplied with the filter's magnitude response $|H(\hat{\omega})|$. Computationally this can be a very demanding procedure as in each iteration of the optimization routine the Fourier transform of the filter must be computed and then multiplied, but the outcomes will definitively be the best possible for that particular signal.

If we solve the optimization problem with this technique we obtain a scaling filter with the magnitude response as shown in figure 4.11.

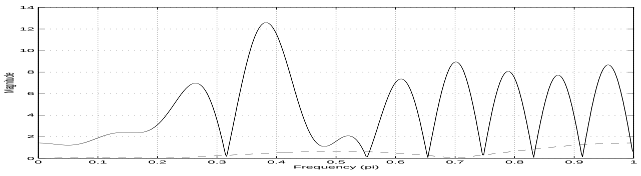

Figure 4.11: Frequency responses of optimized low (solid) and highpass (dashed) filter

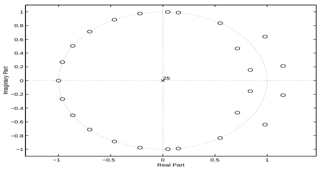

Figure 4.12: zero-points of scaling filter

The optimization algorithm has placed the zero points quite uniformly on the unit circle for high frequencies while the lower frequences are more damped by having pairs of zeros situated around the unit circle.

In our optimization context, this is the optimized analysis scaling filter suited for the test signal; unfortunately the large gain is still a problem. The scaling and wavelet functions generated from this filter therefore for all that have a very irregular shape and they do not converge as observerable in figure 4.13. The energy of the analysis wavelet filter is quite low $\sum g[k]^2 = 0.37$, on the other hand it is very high for the scaling filter: $\sum h[k]^2 = 28.1$. By zero extending the wavelet filter at both ends it is also straightforward to determine the angle between the two filters; and one finds that there is total orthogonality as the angle spans $0.50\pi$. Orthogonal projections give the best possible mean-square error reduction, which is precisely why one obtains this result.

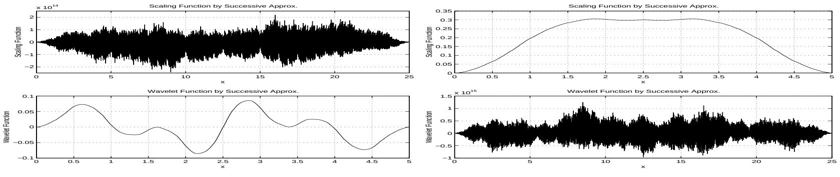

Figure 4.13: Scaling and Wavelet functions, L:Analysis R:Synthesis

Although the decomposition process (figure 4.14) does shift more or less all of the signal's energy into the low frequency zone with indeed the best result so far, the small quantity left in the high frequency zone plays a very crucial role.

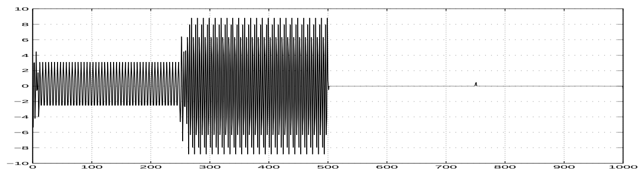

Figure 4.14: decomposed test signal

| Filter | Energy L | Energy H | Sum |
|---|---|---|---|
| 26-6, Optimized$_4$ | 2192.04% | 0.13% | 2192.18% |

Table 4.6: Energy divisions of test signal P1

If we for example truncate the high frequency zone and run the inverse wavelet transform then the new signal ($x_r[n]$) has deviations at the start / end of the signal as well in the middle which is apparent from figure 4.15.

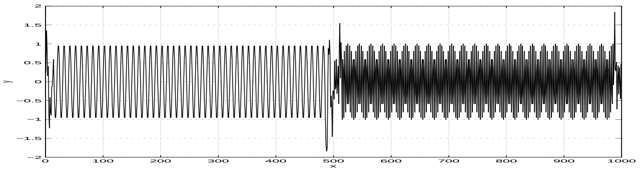

Figure 4.15: reconstructed signal, $\sum(x[n] - x_r[n])^2 = 17.08$

The noise at the beginning/end can easily be removed if we symmetrical extend the test signal before running the filter, however, the sudden noise arising from frequency change in the middle of the signal remains a problem. More realistical signals always have certain noise in them and a good wavelet system should be able to handle such small disturbances as we are witnessing here. The synthesis scaling function of the system is quite smooth, one can therefore try to increase the regularity of the analysis scaling filter.

Using equation (4.6) the regularity of the synthesis wavelet filter can be increased 4 levels by adding 4 extra equations of the form

$(-25^K)\ k + (24^K - 1)\ l + (2^K - 23^K)\ m + (22^K - 3^K)\ n + (4^K - 21^K)\ o + (20^K - 5^K)\ p + (6^K - 19^K)\ q + (18^K - 7^K)\ r + (8^K - 17^K)\ s + (16^K - 9^K)\ t + (10^K - 15^K)\ u + (14^K - 11^K)\ v + (12^K - 13^K)\ w = 0$

for $K = 1, 2, 3$ and $4$. Regularity up to $K = 4$ gives the analysis scaling filter four extra zeros at $z = -1$, on the negative side the optimization procedure is left with only one variable to play around with.

Extra regularity increases the filter's gain slightly (figure 4.16), still for our particular signal the effect is pretty moderate. The supplementary zeros nicely smooth out the frequency response at both ends.

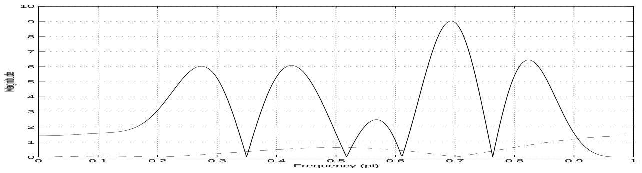

Figure 4.16: Frequency responses of optimized low (solid) and highpass (dashed) filter

Decomposing gives table 4.7.

Truncating the high frequency part and reconstructing gives back a signal where the mean square error is much smaller. Visually one can also observe that the approximation in the start/middle/end of the signal is much better (figure 4.17) compared to our previous filter with no regularity (figure 4.15).

| Filter | Energy L | Energy H | Sum |
|---|---|---|---|
| 26-6, Optimized$_5$ | 2202.35% | 0.13% | 2202.49% |

Table 4.7: Energy divisions of test signal P1

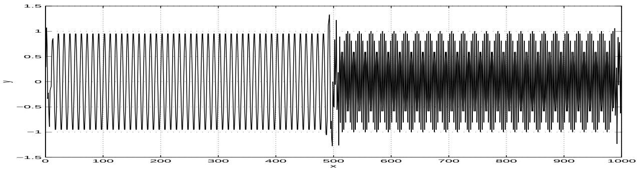

Figure 4.17: reconstructed signal, $\sum(x[n] - x_r[n])^2 = 5.98$

The scaling and wavelet functions generated from $h[n]$ (figure 4.18) are now more regular and have a somewhat different shape but they still do not show convergence to a stable basis.

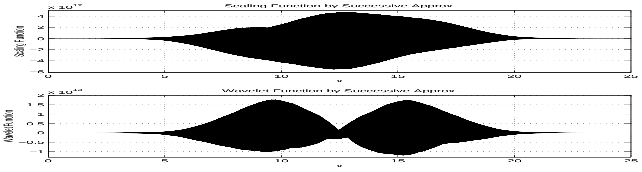

Figure 4.18: Scaling and Wavelet functions

The smoothness goes pretty much unnoticed in the plots as the high frequency structure totally dominates both the scaling and wavelet functions. In a practical situation one of course does not totally truncate one frequency area as we have done, but as it is known that only a limited amount of energy is found there it should be possible to quantize the data there much more strongly gaining noticeable entropy reductions on the high frequency side.

## 4.4 Generalized optimized filters - type A

To generate analysis wavelet filter we have, till now, manually placed zero points at the dominating frequencies. To increase the flexibility even the wavelet filter needs to be constructed in a more proper mathematical fashion. A similar procedure as used previously to design the scaling filter can be followed to construct wavelet filters which would grab as little energy as possible from the signal $x[n]$. In other words we minimize $\sum(|X(\hat{\omega})||G(\hat{\omega})|)^2$ to construct the wavelet filter.

A general two-fold procedure for the total algorithm can look like this:

- $\min \ \sum(|X(\hat{\omega})||G(\hat{\omega})|)^2$
  to create the wavelet filter subject to equations (4.2), (4.4) and any extra smoothness by a form of (4.6)

- $\min \ \sum(|X(\hat{\omega})||H(\hat{\omega})|)^2$
  to construct the scaling filter with respect to the wavelet filter and equations (4.1), (4.3), (4.5) along with the desired regularity (4.6)

The optimization routine needs to be run twice to find the solutions, first to design the wavelet filter and then the scaling filter. The constructed filters will be optimal in the sense of shifting as much of the signal's energy as possible to the low frequency subband, and the energy in high frequency zone is shrunk. In the above algorithm the Fourier transform of signal/filters needs to be computed which requires time-consuming complex arithmetic. An alternative optimization strategy is therefore to directly filter the signal with the coefficients and go straight towards computing the energy of the decomposed signal which is then used as the objective function. I.e.:

- $\min \;\; \sum d[n]^2$
  minimize energy in the high frequency zone to build $g[n]$ and then
- $\min \;\; \sum c[n]^2$
  find $h[n]$, so energy in the low frequency zone is contracted

An one-sided wavelet transform is a very fast operation, the main burden therefore comes on the optimization algorithm. Variant of Gauss-Newton's method can converge very rapidly and the end results are quite similar to the one we obtain via direct Fourier energy minimization, especially if the number of samples in time and frequency domain match each other. Starting values for the optimization can be chosen as perturbations of $-0.1$ and $0.1$ which gives reasonably quick convergence.

The analysis scaling and wavelet filters generated by the above routines can be shown to be orthogonal to each other:

The problem of minimizing the energy of $c[n]$ and $d[n]$ can be considered as two least square problems, assuming both $h[n]$ and $g[n]$ are of even length. First we want to reduce the norm of $d[n]$ which is basically the result of a convolution:

$d[n] = \sum_k g[k]x[n-k] = \sum_k x[k]g[n-k]$

By placing the signal $x[n]$ in a matrix, row-by-row where each row is translated, this can be formulated as: $\mathbf{d} = \mathbf{Xg}$. The goal is to reduce
$||\mathbf{r_1}|| = ||\mathbf{Xg} + \mathbf{x}||$. The least square solution $\mathbf{r_1}$ is orthogonal to the range of matrix $\mathbf{X}, \mathbf{X}^*\mathbf{r_1} = \mathbf{0}$.

For the analysis scaling filter we have a similar case:
$c[n] = \sum_k h[k]x[n-k] = \sum_k x[k]h[n-k]$ or $\mathbf{c} = \mathbf{Xh}$. The optimal solution which reduces the energy $||\mathbf{r_2}|| = ||\mathbf{Xh} - \mathbf{x}||$ is also orthogonal to the signal space. Consequently we get $\mathbf{r_1} \perp \mathbf{X}^*$ and $\mathbf{r_2} \perp \mathbf{X}^*$. Unless $\mathbf{r_1} = \mathbf{r_2}$ we can conclude that $\mathbf{r_1} \perp \mathbf{r_2}$ and this can only be the case when the filters them self $h[n]$ and $g[n]$ are uncorrelated.
The down-sampling has been ignored in the above but this can be incorporated in the matrix $\mathbf{X}$ itself. Odd length wavelet orthogonal filters are not possible [6] so in that case we end up with a contradiction, however, the best solution must be be as near orthogonal as possible. □

To investigate more about these optimized filters and see the practical effects of filters algorithms were designed to construct optimized filters of different lengths, given a signal.

We concentrate upon the following filters (descriptions relative to the analysis side):

- 10-6: 2 degrees of freedom for the wavelet filter and 1 free variable for the scaling filter
- 15-9: 3 degrees of freedom for the wavelet filter and 1 free variable for the scaling filter
- 26-6: Here we have 2 degrees of freedom for the wavelet filter and 5 degrees of freedom for the scaling filter, however, we increased the regularity of thescaling filter by one degree

- 26-6R: A regular filter variant of the 26 filter above. The regularity will be increased by 4 degrees, leaving only 1 free variable to the optimization routine
- 34-6: The wavelet filter is the same as in other cases, but we have 7 free variables to play with in $h[n]/\tilde{g}[n]$. We take up 3 of them to add extra regularity

All the free variables are used by the optimization algorithm to compact as much energy in the first coefficients of a wavelet decomposition.

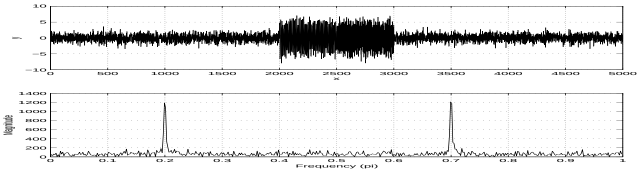

Figure 4.19: test signal P2

To test the filters a new artificial signal was created composed of the original testing signal from previous chapter (figure 3.3) which was combined with the two frequency signal P1 (figure 4.3). The signal and it's Fourier transform are in figure 4.19. This is a basically P1 where the whole frequency spectrum is covered to some level. 49.38% of the signal's energy is found in the lower frequency half while the rest 50.61% is in the higher frequency spectrum.

We decomposed this signal with the optimized filters, and found how much of the original signal's energy was distributed in different zones. The percentage results relative to the original signal's energy are in table 4.8. Additionally we also calculated the entropy of the decomposed signal assuming 3 digits after the decimal were stored; these are shown in brackets. The rightmost column shows the average entropy ($\bar{e}$) counts. For the original signal the entropy turns out to be 11.49.

| Biorthogonal Filter | Energy L | Energy H | Sum ($\bar{e}$) |
|---|---|---|---|
| 10-6 | 3229.89% (11.21) | 5.28%(10.39) | 3235.18% (10.80) |
| 15-9 | 27566.32% (11.24) | 3.54% (10.22) | 27569.86% (10.73) |
| 26-6 | 2196.06% (11.22) | 5.28% (10.39) | 2201.34% (10.80) |
| 26-6R | 2805.45% (11.21) | 5.28% (10.39) | 2810.74% (10.79) |
| 34-6 | 1688.66% (11.21) | 5.28% (10.39) | 1693.94% (10.80) |

Table 4.8: Energy divisions of signal P2

How does these signal specific filters compare with more standard filters ? The results of the same signal decomposed with three orthogonal and four biorthogonal variants are shown in table 4.9.

There is no doubt about that the filters we have designed are quite good at shifting as much energy as possible of a signal into the low frequency domain, however, the large increases in the power and entropy may look problematical. Longer scaling filters are also limited in helping reducing the gain of the frequency response. The scaling/wavelet functions furthermore are very irregular and adding extra regularity only helps on the micro level. Small disturbances in the high frequency zone can therefore have dramatical effects on the reconstructed signal.

Nevertheless, the minimization of energy on the high frequency part is substantial which also results in low figures for the entropy there. The average entropy results are therefore not so

| Filter | Energy L | Energy H | Sum ($\bar{e}$) |
|---|---|---|---|
| Daubechies 8 | 50.50%(10.78) | 49.49%(10.75) | 100.00% (10.77) |
| Daubechies 12 | 50.00%(10.77) | 49.99%(10.74) | 100.00% (10.76) |
| CDF 9-7 | 49.52%(10.82) | 51.04%(10.79) | 100.56% (10.80) |
| 6-10 | 64.33%(10.85) | 37.72%(10.74) | 102.05% (10.80) |
| Coiflet 17-11 | 50.65%(10.81) | 49.82%(10.78) | 100.48% (10.79) |

Table 4.9: Energy divisions of signal P2

bad, the best optimized filter 15-9 gives an average entropy of 10.73 which goes unmatched by other standard filters. A large wavelet filter is of course good for the energy damping in the H zone, however, this automatically enlarges the L area.

Repeating the same experiment with a signal (figure 4.20) where 99.66% of the signal's energy consists of low frequencies we get more or less similar results with large energy increases and also reductions.

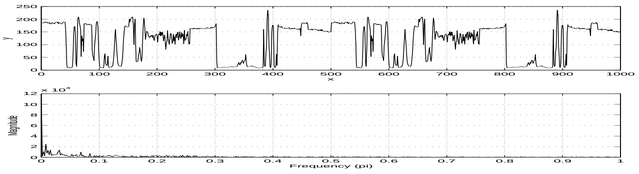

Figure 4.20: test signal P3

The entropy of this signal is 6.62 which is small as the signal is only composed of integers. Decomposing with the optimized designed filters gives table 4.10. The entropy counts in parenthesis are computed after including 1 decimal digit. The average entropy values are also in the table on right. The traditional filters used on signal P3 are in table 4.11.

| Biorthogonal Filter | Energy L | Energy H | Sum ($\bar{e}$) |
|---|---|---|---|
| 10-6 | 117.71% (7.86) | 0.19%(6.32) | 117.91% (7.09) |
| 15-9 | 948.43% (7.93) | 0.13% (6.39) | 948.56% (7.16) |
| 26-6 | 116.41% (7.91) | 0.19% (6.32) | 116.60% (7.12) |
| 26-6R | 118.59% (7.87) | 0.19% (6.32) | 118.78% (7.10) |
| 34-6 | 116.22% (7.88) | 0.19% (6.32) | 116.42% (7.10) |

Table 4.10: Energy divisions of signal P3

With quantization to 1 decimal digit after decomposition with the optimized filters, the quality of the reconstructed signal ($x_r[n]$) is somewhat acceptable and the maximum error at any position remains below 0.25 for the 10-6 filter. The mean square error from this filter $\sum(x[n]-x_r[n])^2$ comes at 11.33; the CDF 9-7 filter is much more stable though, and returns an error of only 0.81. Integer quantization with the optimized filters is not to recommend.

The P3 signal more or less does not contain high frequencies, yet the optimized filters have even further reduced any leakage which would come through the wavelet filter. The entropy numbers are also low in that zone compared with more standard filters. All the energy and entropy increase occur among the lower frequencies as a result of large gains in the analysis

| Filter | Energy L | Energy H | Sum ($\bar{e}$) |
|---|---|---|---|
| Daubechies 8 | 99.38%(7.79) | 0.61%(6.91) | 100.00% (7.35) |
| Daubechies 12 | 99.43%(7.78) | 0.56%(6.95) | 100.00% (7.37) |
| CDF 9-7 | 99.09%(7.76) | 0.55%(6.86) | 99.64% (7.31) |
| 6-10 | 100.48%(7.81) | 0.47%(6.80) | 100.96% (7.31) |
| Coiflet 17-11 | 99.43%(7.72) | 0.52%(6.85) | 99.96% (7.29) |

Table 4.11: Energy divisions of signal P3

scaling filter. The 10-6 filter for example for P3 gives $\sum h[k]^2 = 23.88$, the wavelet filter on the other hand gives a very low outcome: $\sum g[k]^2 = 0.36$. This filter though gives the best entropy results, showing that longer filters do not necessarily do a better job.

If one looks at the average entropy values then the optimized filters can give improved effective coding than the more typical filters. The low frequency zone indeed requires extra bits to be coded but the high frequency zone can compensate for this growth.

The optimization routine normally converges very rapidly, however, optimized filters do not necessarily need to be created on-line; by using average values over a particular class of signals filters can easily be designed and stored for future use.

The Karhunen-Loève transform (KL) [9] is a statistical based transform which is optimal when it comes to decorrelating a data set. The main disadvantage is that the basis becomes dependent upon the signal itself. Transforming a signal via KL, however, results in maximum energy being shifted over to the few first coefficients of the transformed signal.

The filters designed so far can therefore be regarded as approximation attempts to the KL transform, though, we have no stability at all, this makes it very unpractical to for example iterate the decomposition process on the low frequency zone or to only keep a few numer of coefficients as an approximation of the signal.

## 4.5 Generalized optimized filters - type B

In the previous sections the aim was to reduce the energy of a decomposed signal's high frequency zone to a minimum. The drawback with this is that the energy amount in the low frequency area is increased substantially and it is very difficult to depress this energy.

Although we succeeded in minimizing the energy in the high frequency zone there is not much to cheer about when it comes to the energy propagation and the unstability of such filters.

Our former methods have been based upon a design strategy where firstly the wavelet filter is constructed and then afterwards a scaling filter is put together. One can combine these two procedures into one and try a different minimization criteria:

$$\min \; \sum \Big( (|H(\hat{\omega})||X(\hat{\omega})|)^2 + (|G(\hat{\omega})||X(\hat{\omega})|)^2 \Big)$$

In other words the aim now is to minimize the total energy of the decomposed signal. This can also be formulated as:

$$\min \; \sum (c[n]^2 + d[n]^2)$$

The latter method, where one directly looks at the energy of the decomposed signal is a much effective way to calculate the objective function for signals of moderate length as one avoids complex arithmetic computations. We name filters constructed via the above technique for type B filters.

Regardless the method solving this kind of optimization problem demands a somewhat different approach than the one we have been using so far. The optimization routine is only called once and this time no attempts are made to reduce the energy amount in one particular zone or to shift energies as much as possible. The filters within the optimization routine are still designed in two stages; first constructing the synthesis scaling filter and thereupon creating the opposite counterpart. This effectively reduces the filter design problem into two linear systems. After the filter construction the value for the objective function can be found easily by a signal decomposition and energy calculation. Finding a solution to this unconstrained minimization problem requires an iteration process where either quasi-Newton methods or Nelder-Mead simplex search method[2] can be applied. The type B filter construction problem is computationally more demanding than the type A filter variant as both the scaling and wavelet filters are constructed in the same run, and the power minimization depends upon both filters.

The next section demonstrates how to construct optimized 17-11 filters, with extra regularity, to reduce the total energy of the decomposed signal to a minimum.

## 4.6 17-11 filters

Filters of odd lengths give symmetrical wavelet functions and another advantage is that one attains at least two vanishing moments automatically for both analysis and synthesis wavelet filters.

$$h[n] = [k\ l\ m\ n\ o\ p\ q\ r\ s\ r\ q\ p\ o\ n\ m\ l\ k]$$
$$\tilde{h}[n] = [a\ b\ c\ d\ e\ f\ e\ d\ c\ b\ a]$$

Such a system gives 5 free parameters to set as we please. If we start with construction of the synthesis scaling filter $\tilde{h}[n]$ then 4 of the free variables are available in $\tilde{h}[n]$ while only one component in $h[n]$ can be set arbitrary. $a, b, c, d$ and $k$ can for example be left over to the optimization algorithm and the end filters should be in a position to reduce the total energy of the decomposed signal to a small amount. Here, we will however, increase the number of vanishing moments for the filters and bring the number of free parameters down to 3.

Selecting $a, b$ and $c$ as the open elements and increasing the regularity by one degree, the equations for the synthesis scaling filter can be found to be:

$$2a + 2b + 2c + 2d + 2e + f = \sqrt{2}$$
$$2a - 2b + 2c - 2d + 2e - f = 0$$
$$-10^2a + (9^2 + 1)b - (8^2 + 2^2)c + (7^2 + 3^2)d - (6^2 + 4^2)e + 5^2 f = 0$$

This can be looked as a 3x3 linear system as $a, b$ and $c$ are to be preselected by the optimization routine.

Having constructed the synthesis scaling filter and the analysis wavelet filter :

$$g[n] = [-a\ b\ -c\ d\ -e\ f\ -e\ d\ -c\ b\ -a]$$

$h[n]$ can be build. The equations take the form of:

$$2k + 2l + 2m + 2n + 2o + 2p + 2q + 2r + s = \sqrt{2}$$
$$2k - 2l + 2m - 2n + 2o - 2p + 2q - 2r + s = 0$$
$$2a\ n + 2b\ o + 2c\ p + 2d\ q + 2e\ r + s\ f = 1$$

$$a\,l + b\,m + c\,n + d\,o + (e+a)\,p + (f+b)\,q + (e+c)\,r + d\,s = 0$$
$$b\,k + c\,l + d\,m + e\,n + f\,o + e\,p + d\,q + (c+a)\,r + b\,s = 0$$
$$d\,k + e\,l + f\,m + e\,n + d\,o + c\,p + b\,q + a\,r = 0$$
$$f\,k + e\,l + d\,m + c\,n + b\,o + a\,p = 0$$
$$(c - \tfrac{ad}{b})\,l + b\,m + a\,n = 0$$

$16^2k - (15^2+1)l + (14^2+2^2)m - (13^2+3^2)n + (12^2+4^2)o - (11^2+5^2)p + (10^2+6^2)q - (9^2+7^2)r + 8^2s = 0$

These 9 equations uniquely determine the analysis scaling and synthesis wavelet filters. The rest of the job can be left to the optimization algorithm which has to make a search to find suitable values for $a, b$ and $c$ relative to the input signal.

Running the algorithm on signal P1 with starting values $a = -0.1, b = 0.1$ and $c = 0.1$ we obtain convergence in 123 steps (309 function counts) using the Nelder-Mead routine. Gauss-Newton's algorithm is much quicker and converges in only 13 steps, with 118 function evaluations. The resulting optimized filters' magnitude responses are in figure 4.21 while table 4.12 shows the outcome of decomposing the signal.

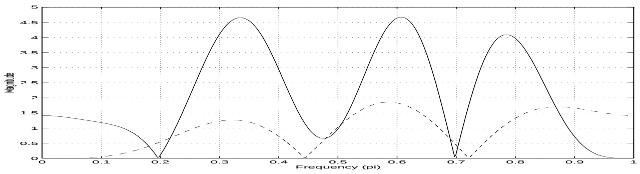

Figure 4.21: Frequency responses of low (solid) and highpass (dashed) filters

It is noteworthy that it is the scaling filter which has zeros placed very close to the dominating frequencies rather than the wavelet filter. The roles for analysis/synthesis filters can nevertheless always be reversed if necessary. Upon close examination it gets apparent that the scaling filter does not try to null out the dominating frequencies, instead the zero points are placed extremely close to $e^{j0.2\pi}$ and $e^{j0.7\pi}$. This is admittedly a wise strategy as we have already seen the negative effects of totally nulling out the most influential frequencies.

| Filter | Energy L | Energy H | Sum |
|---|---|---|---|
| 17-11, Optimized$_1$ | 3.21% | 12.41% | 15.63% |

Table 4.12: Energy divisions of test signal P1

The energy minimization result is no doubt a very good one considering we have had at least over 2000% energy increases in our previous attempts. The decomposed signal with this filter is observable in figure 4.22. The basic structure of the signal can be seen in both of the frequency zones.

The 17-11 filter is constructed to have extra regularity for functions generated via both $h[n]$ and $\tilde{h}[n]$. Figure 4.23 shows that there is little visible effect on the standard scale despite the fact that both $h[n]$ and $\tilde{h}[n]$ now end up having 4 zeros at $z = -1$. Although the energy of the decomposed signal has been shrunk the gain of scaling the filter is also very large at places, any form of quantization on the subsignals is hence a rather unstable operation.

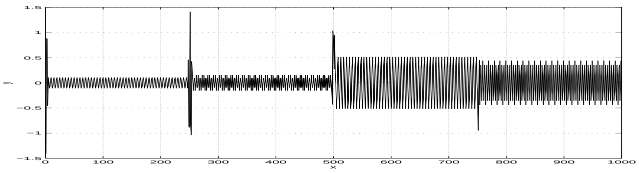

Figure 4.22: P1 decomposed

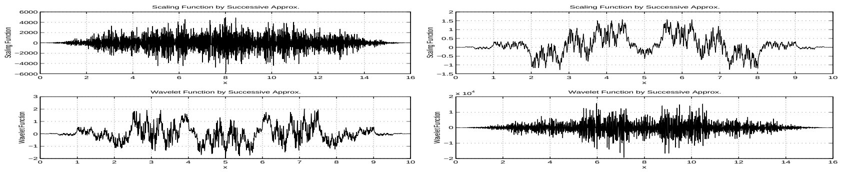

Figure 4.23: Scaling and Wavelet functions, L:Analysis R:Synthesis

The energy of the analysis wavelet filter is moderate $\sum g[k]^2 = 1.24$, but the scaling filter shows another side; $\sum h[k]^2 = 6.66$.

### 4.6.1 Weighted minimization

As mentioned previously the roles for analysis and scaling filters can be reversed if desired, however, there is then still left around 3% energy in the high frequency zone. Anyway, one can attempt to reduce the energy in the low frequency zone even further by modifying the optimization criteria to:

$$\min \;\; \sum(Q\; c[n]^2 + d[n]^2) \qquad Q \geq 1$$

A large $Q$ forces the algorithm to additionally contract the energy in the low frequency area as any energy there is given more weight than the energy in the high frequency zone $d[n]$.

Trying this generalized method with $Q = 2$ and decomposing the signal with the new filter obtained after optimization we find:

| Filter | Energy L | Energy H | Sum |
|---|---|---|---|
| 17-11, Optimized$_2$ | 1.27% | 16.02% | 17.29% |

Table 4.13: Energy divisions of test signal P1

Compared with table 4.12 the amount of energy in the low frequency zone is further reduced by half.

Increasing $Q$ to $10$ and after some adjustments of the initial values $a = 0.1, b = -0.1, c = -0.1$ the new optimized filters further reduce the energy in the low frequency zone (table 4.14).

| Filter | Energy L | Energy H | Sum |
|---|---|---|---|
| 17-11, Optimized$_3$ | 0.62% | 40.70% | 41.33% |

Table 4.14: Energy divisions of test signal P1

The decrease comes at the expense of the energy increases in the high frequency zone. One can go ahead and keep enlarging $Q$, none the less in that event we gradually start going back to our type A filters as with increasing $Q$ the energy amount grows very rapidly in the high frequency area.
If we set $N$ to a very large number or even make it go towards infinity $N \to \infty$

$$\min_{Q \to N} \sum (Q\ c[n]^2 + d[n]^2) \Rightarrow \min_{Q \to N} \sum Q\ c[n]^2 = \min \sum c[n]^2 \Leftrightarrow \min \sum (|H(\hat{\omega})||X(\hat{\omega})|)^2$$

This optimization cost function is identical to the one for type A filters. A benefit of this weighting method is that all the filters are constructed in the same process, this can be advantageous in certain circumstances.

## 4.7 Type B filters

Just like type A filters, (even length) type B filters are also perpendicular to each other, implying that there is not any correlation between the decomposed signal. The argumentation for this is similar to the one for type A filters. Within the optimization routine a scaling filter is fully constructed before we one starts generating the other scaling filter pair. The minimization can also be regarded as a least-square optimization.

To run some more experiments with filters who minimize the total energy of the decomposed signal we constructed algorithms to generate filters of various sizes and with different number of vanishing moments. None of these filters use any special weight consideration ($W = 1$); all references are to the analysis filters:

- 6-14: 2 free variables for the scaling filter and 2 free parameters for the wavelet filter
- 9-7: 2 free variables for the wavelet filter
- 10-6: 1 free variable for the scaling filter and 2 free variables for the wavelet filter
- 10-6R: increased regularity for both the scaling and wavelet filter; leaves 1 free variable
- 15-9: 1 free variable for the scaling filter and 3 free parameters for the wavelet filter
- 17-11: 1 free variable for the scaling filter and 4 free parameters for the wavelet filter
- 17-11R: in this filter the free parameter in the scaling filter is used to increase regularity and the same is the case for 2 of the free variables in the wavelet filter. This leaves 3 free variables to the optimization routine

Optimizing and decomposing the all-frequency signal P2 with the filters gives table 4.15.

Comparing the table with tables 4.8 and 4.9 we see that type B filters have done a fine job in removing the correlations as much as possible and have at least managed to reduce the energy of the decomposed signal. The more freedom in the system, the better the energy minimization. The entropy figures, however, are not taking part in any reduction.

It is important to point out that to really locate the global minimum value one needs to run the algorithm a few times with different starting points. Generally just perturbing the values between $-0.1$ and $0.1$ may be sufficient as we don't want our filter to be made up of large values but a foolproof algorithm should take somewhat larger starting numbers into account.

| Biorthogonal Filter | Energy L | Energy H | Sum ($\bar{e}$) |
|---|---|---|---|
| 6-14 | 36.41% (10.76) | 36.80%(10.88) | 73.21% (10.82) |
| 9-7 | 56.58% (10.83) | 42.47% (10.78) | 99.06% (10.80) |
| 10-6 | 39.13% (10.86) | 50.36% (10.78) | 89.49% (10.82) |
| 10-6R | 50.74% (10.82) | 48.49 % (10.78) | 99.24% (10.80) |
| 15-9 | 34.12% (10.84) | 33.07% (10.77) | 67.20% (10.80) |
| 17-11 | 32.36% (10.85) | 33.47% (10.75) | 65.83% (10.80) |
| 17-11R | 47.66% (10.94) | 31.54% (10.83) | 79.21% (10.88) |

Table 4.15: Energy divisions of signal P2

| Biorthogonal Filter | Energy L | Energy H | Sum ($\bar{e}$) |
|---|---|---|---|
| 6-14 | 95.63% (7.82) | 1.09%(7.17) | 96.73% (7.49) |
| 9-7 | 95.07% (7.82) | 2.05% (7.37) | 97.12% (7.60) |
| 10-6 | 95.04% (7.83) | 1.48% (6.87) | 96.53% (7.35) |
| 10-6R | 97.74% (7.74) | 1.37% (7.13) | 99.11% (7.43) |
| 15-9 | 93.95% (7.72) | 2.05% (7.43) | 96.01% (7.57) |
| 17-11 | 92.37% (7.84) | 1.97% (7.42) | 94.35% (7.63) |
| 17-11R | 96.38% (7.80) | 0.99% (7.30) | 97.38% (7.55) |

Table 4.16: Energy divisions of signal P3

The same experiments, this time with the more simple low frequency based signal P3 are in table 4.16.

Signal P3 virtually only consists of low frequencies and this is obviously creating problems for the type B optimized filters. There is already very little energy in the high frequency spectrum and hence little to reduce in a wavelet iteration. Still the filters have managed to decrease the total energy in all cases compared with the more typical filters (table 4.11) though the entropy values are not at all competitive. To achieve the energy reductions some quantity of energy has to be shifted acorss the different subbands, this gives relative enlarged entropy figures for both zones; increasing the average entropy values. Increased regularity for the scaling filter likewise enlarges the total energy and entropy in all cases.

## 4.8 Summary

In this chapter two different kinds of biorthogonal wavelet filters were designed, with the goal of either shifting a signal's energy to the low frequency zone or to minimize the total energy of a signal after a wavelet decomposition.

The first method of transferring all of the energy to one zone works quite well and it should even be possible to achieve useable results during entropy coding for certain signals without corrupting the reconstructed data too much. The second strategy of generating filters decorrelates a signal very effectively, though, the energy is spread out over both low and high frequency zones, restricting any entropy gains.

Type A and B filters manage to do their job, however, the biggest disadvantage with both filter types is the instability. This does not provide scaling and wavelet functions with limited energy and restricts the level of modifications which can be done to the decomposed signal.

# Chapter 5

# Further Improvements

In this chapter we investigate methods to expand and improve the performance of the biorthogonal filters from the previous chapter. Even though the filters constructed in the last chapter did do their job of either shifting all the energy to the low frequency zone or minimizing the total energy; unfortunately the unstability of such filters remained a big drawback.

## 5.1 Increasing Stability

The filters generated so far have the very objectionable feature that the scaling and wavelet functions keep reaching very large values, i.e. they do not have finite energy. This unstability of scaling functions can be looked upon via the cascade algorithm, $\phi^{(j)}(t) = \sqrt{2}\sum h[n]\ \phi^{(j-1)}(2t-n)$. We can define the inner product $\mathbf{a}^{(j)}(k) = \int \phi^{(j)}(t)\phi^{(j)}(t+k)\ dt$ for each iteration step and link $\mathbf{a}^{(j+1)}$ together with $\mathbf{a}^{(j)}$ with the help of a transition matrix, $\mathbf{a}^{(j+1)} = \mathbf{T_h}\mathbf{a}^{(j)}$. The matrix $\mathbf{T_h}$ can be formed with the help of $h[n]$, $\mathbf{T_h} = [(\downarrow 2)\ \mathbf{HH}^*]$ where $\mathbf{H}$ is the Toeplitz folding matrix of the filter, $\sqrt{2}\mathbf{H}_{i,j} = h[i-j]$. This transition matrix has at least one eigenvalue of $\lambda_1 = 1$, which can be related with the scaling filter constrain $|H(\pi)| = 0$. For a stable $L^2$ basis $\int |\phi(t) - \phi^{(j)}(t)|^2\ dt \to 0$ as $j \to \infty$, one should further ensure that the other eigenvalues are also limited, $|\lambda_i| < 1$. Additionally, the smaller these eigenvalues are the more smoothness can be expected from the generated scaling/wavelet functions. In the biorthogonal case a separate transition matrix $\mathbf{T_{\tilde{h}}}$ must be connected with the filter $\tilde{h}[n]$, and ideally it too should obey the same confinements [3][15]. The filters in the foregoing chapter do not consider the eigenvalues at all and it's evident that this is the cause behind the unstability.

One another alternative view to the stability problem is to look upon the filters' energy $\sum h[n]^2$ and $\sum \tilde{h}[n]^2$. Under the optimization of biorthogonal filters in chapter 4 no limit is posed upon these energies. By iterating the filters scaling functions are constructed but with no restrains on the filters the scaling functions easily end up reaching substantial values.

Orthogonal wavelet filters don't have these problems at all as the filters energy sums up to 1 giving full stability. What one can do is to add extra constrains to the optimization so the filters' energy is kept within desired limits:

$$p_1 \leq \sum h[k]^2 = \sum \tilde{g}[k]^2 \leq q_1 \tag{5.1}$$

$$p_2 \leq \sum \tilde{h}[k]^2 = \sum g[k]^2 \leq q_2 \tag{5.2}$$

It is known [13] that $( \sum h[k]^2 )^{\frac{1}{2}} \leq \max |H(\omega)|$ which is equivalent with $\sum h[k]^2 \leq \max |H(\omega)|^2$. Setting energy constrains is therefore identical to limiting the maximum gain of the frequency response. With these adjustments we avoid computations of eigenvalues and can set the restrains directly on filters with flexibility. This setting also ensures that the optimization algorithm for the most part is able to converge and find a solution, albeit it may not be mean-square stable in the strict sense.

One can obviously also make the functions fully $L^2$ stable by calculating the eigenvalues within the optimization and apply constrains such as:

$$\mathbf{T_h}; \qquad \max_{\mathrm{i}} |\lambda_{\mathrm{i}}| \leq 1 \tag{5.3}$$

$$\mathbf{T_{\tilde{h}}}; \qquad \max_{\mathrm{i}} |\lambda_{\mathrm{i}}| \leq 1 \tag{5.4}$$

We will later see that the latter method is not very appropriate for type A filter optimization, but it can work well under type B construction.

### 5.1.1 Type A-II filters

With (5.1) and (5.2) the more closer $p$ and $q$ are to $1$ the more stability can be expected from the wavelet system. Unfortunately this setting no doubt clashes with our optimization criteria of shifting and minimizing as much energy as possible. Table 3.9 can perhaps be used to get some hints on how we should select figures for $p$ and $q$, but the filters we experimented with were somewhat restrictive and maybe too stable than what is absolutely necessary. It should be possible to further relax these filters' limits without loosing too much on the stability part.

In type A filter design the analysis wavelet filter is first constructed and the extra constrain $p_2 \leq \sum g[k]^2$ will now generally increase the wavelet filter's energy to at least $p_2$ compared with the original algorithm. This also implies that the results for the high frequency zone will no longer be as good as we got under the previous chapter, to gain some stability one has no choice but to sacrifice optimal optimization. The second phase of the algorithm where the analysis scaling filter is constructed does not need to make use of equation (5.1) as this filter can be based upon the premisses already set by the wavelet filter, with the prime intention of minimizing the energy in the low frequency zone. Extra energy limits for $h[n]$ filter may easily return very large energy increases for $c[n]$, while having both filter energy constrains and total energy minimization criteria only lowers the chances of finding any optimized solution at all.
Consequently one basically only needs to add the lower part of the constrain to the optimization procedures; as the analysis wavelet filter tries to minimize the energy it captures the coefficients there will already be the small, removing the need of an upper limit.

We name type A filters with the additional energy constrain

$$p \leq \sum g[k]^2 \tag{5.5}$$

for type A-II filters.

The main question then comes up: how do we select $p$ for various classes of filters ? Table 5.1 shows how the 26-6 filter performs with different values of $p$ under the two frequency P1 signal.

The first thing we notice is that the roles for scaling and wavelet filters are reversed in regard to what we would prefer but this is only a minor nuisance. Full optimization with no energy constrain on the filters would give us $(p) = \sum \tilde{h}[k]^2 = 0.37$. As we keep increasing $p$ the low

| $p$ | $\sum h^2$ / $\sum \tilde{h}^2$ | $\max \lvert\lambda\rvert_h$ $/ \lvert\lambda\rvert_{\tilde{h}}$ | $\max \lvert\phi(t)\rvert$ / $\lvert\tilde{\phi}(t)\rvert$ | Energy L | Energy H | Sum |
|---|---|---|---|---|---|---|
| 0.70 | 3.12 / 0.70 | 3.94 / 1.00 | 20560.08 / 0.71 | 1.64% | 27.27% | 28.91% |
| 0.90 | 2.13 / 0.90 | 2.54 / 1.00 | 288.53 / 1.38 | 1.17% | 38.08% | 39.25% |
| 1.00 | 1.86 / 1.00 | 2.16 / 1.00 | 64.35 / 3.86 | 1.03% | 43.31% | 44.35% |
| 1.10 | 1.66 / 1.10 | 1.88 / 1.00 | 18.43 / 10.98 | 0.93% | 48.47% | 49.41% |
| 1.15 | 1.57 / 1.15 | 1.77 / 1.00 | 10.64 / 18.06 | 0.89% | 51.02% | 51.91% |
| 1.20 | 1.50 / 1.20 | 1.67 / 1.00 | 6.42 / 29.08 | 0.85% | 53.55% | 54.41% |
| 1.30 | 1.38 / 1.30 | 1.50 / 1.00 | 2.65 / 70.73 | 0.79% | 58.58% | 59.38% |
| 1.50 | 1.19 / 1.50 | 1.25 / 1.17 | 0.74 / 335.68 | 0.69% | 68.51% | 69.21% |
| 1.70 | 1.06 / 1.70 | 1.07 / 1.37 | 0.42 / 1259.39 | 0.62% | 78.30% | 78.93% |
| 2.00 | 0.92 / 2.00 | 1.00 / 1.66 | 0.31 / 6677.75 | 0.54% | 92.79% | 93.33% |

Table 5.1: Type A-II 26-6 filter on signal P1

frequency part gradually starts to loose energy while the energy in the high frequency zone is enlarged. The maximum values for the eigenvalues and scaling functions (20 iterations) follow the same pattern. A small $p$ gives a very large analysis scaling function while the synthesis scaling function is very small. A large $p$ on the other hand gives minute analysis scaling functions with small eigenvalues for $\mathbf{T_h}$, however, that results in larger eigenvalues for the synthesis scaling filter's matrix $\mathbf{T_{\tilde{h}}}$. Increasing $p$ further would simply lead us back to our type A kind of filters with very large values for the synthesis scaling filter. Extreme values are good at decorrelation but moderate values are more suitable for stability. As we see it is not possible to make both filter pairs steadfast at the same time by this method.

The method of putting energy constrain is still quite good at reducing the maximum value the scaling functions obtain under a limited number of iterations. Out from the table $p = 1.125$ points towards the optimum value where both scaling functions generated after 20 iterations reach lowest maximum values. It is though necessary to point out that even here (figure 5.1) one does not have stability in the full sense, the largest eigenvalue for $\mathbf{T_h}$ comes at 1.83 while for $\mathbf{T_{\tilde{h}}}$ it is 1.00. The scaling functions reaches up to $14 - 15$ and compared with more typical filters this is indeed somewhat ample.

Figure 5.1 shows the frequency response of 26-6 filters with $p = 1.125$ and one can see that the magnitude response of the scaling filter is like a linear line while the lowpass filter is nulling out the dominating frequencies. This gives hints about a novel idea on how to use optimized filters: One preconstructed static wavelet filters with somewhat linear magnitude response can be used to capture all the frequencies in a signal while the other filter designed specifically for the signal would null out the most influential frequencies, trying to catch as little energy as possible.

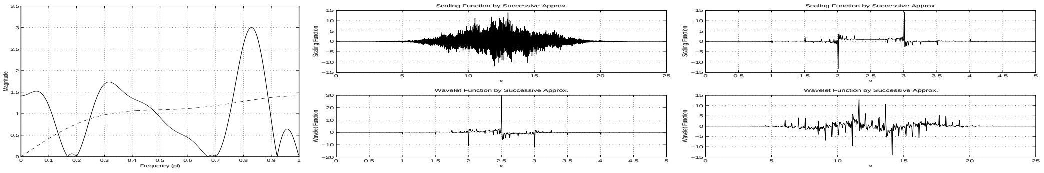

Figure 5.1: L: Frequency response, S/W functions: M: Analysis R: Synthesis

The scaling functions for 26-6 filter with $p = 1.125$ now do not grow wildly in the y-range but the high frequency approximation still gives very irregular shapes.

Finding a reasonable values for $p$ by fully computing $\phi(t)$ and $\tilde{\phi}(t)$ or by using the eigenvalues can be a very demanding procedure. The process can be speeded up by only running 3 or 4 iterations but as we are talking about an optimization within an optimization these procedures can hardly be recommended for quick processing. One can likewise question how much stability one really needs in the case of filter banks.

### 5.1.2 Type B-II filters

Energy limits on filters can also be placed for the type B class of filters. The main difference here is that one now need to use both equations (5.1) and (5.2) with upper and lower constrains as the design procedure for these types of filters is slightly different and limits on only one filter may not necessary have any direct impact on the other filter pair.

Type B-II filters further complicate the main issue; how do we find good values for $p_1, p_2, q_1$ and $q_2$ ? We should at least select $p_1 \approx p_2$ and $q_1 \approx q_2$ as this imposes equal restrictions on both filters. Otherwise the optimization routine may simply pick the filter with lowest $p_1$ or $p_2$, reduce the gain there; while increase it to maximum on the other.

Table 5.2 shows the results of the 17-11 filter on P1 where we have set $p = p_1 = p_2$ and $q = q_1 = q_2$.

| $p$ | $q$ | $\sum h^2$ / $\sum \tilde{h}^2$ | $\max|\lambda|_h$ /$|\lambda|_{\tilde{h}}$ | $\max|\phi(t)|/|\tilde{\phi}(t)|$ | Energy L | Energy H | Sum |
|---|---|---|---|---|---|---|---|
| 0.30 | 10.00 | 10.00 / 1.12 | 12.89/1.04 | $2.5 \cdot 10^{10}$ / 4.22 | 2.84% | 7.84% | 10.69% |
| 0.30 | 3.00 | 3.00 / 3.00 | 3.76/3.73 | 34605.6 / 26826.0 | 7.29% | 9.45% | 16.74% |
| 0.30 | 1.80 | 1.80 / 1.80 | 1.64/1.00 | 10.05 / 5.07 | 34.40% | 9.50% | 43.91% |
| 0.50 | 1.80 | 1.80 / 1.80 | 1.64/1.00 | 10.05 / 5.07 | 34.40% | 9.50% | 43.91% |
| 0.50 | 1.50 | 1.50 / 1.28 | 1.73/1.02 | 23.24 / 4.71 | 9.37% | 34.46% | 43.83% |
| 0.70 | 1.30 | 1.30 / 1.30 | 1.00/1.00 | 1.09 / 1.82 | 23.89% | 25.19% | 49.09% |
| 0.80 | 1.20 | 1.20 / 1.20 | 1.00/2.09 | 1.31 / 51.83 | 22.34% | 29.69% | 52.03% |
| 0.90 | 1.10 | 1.10 / 1.10 | 1.00/1.77 | 0.77 / 12.50 | 54.20% | 18.21% | 72.42% |
| 0.95 | 1.05 | 1.05 / 1.05 | 1.00/1.00 | 0.99 / 1.42 | 40.04% | 35.37% | 75.41 % |
| 0.97 | 1.03 | 1.03 / 1.03 | 1.00/1.00 | 1.02 / 1.37 | 43.31% | 37.14% | 80.45 % |

Table 5.2: Type B-II 17-11 filter on signal P1

It turns out that this time, unlike type A-II filters, the lower limit doesn't have much to say; the optimized filter generally limit them self as close as possible to the upper constrain $q$. For type B optimization setting boundaries on both filters helps reduce the large eigenvalues and one can easily acquire full stability.

Figure 5.2 show the frequency response and scaling/wavelet functions in the case of $p = 0.7$ and $q = 1.3$.

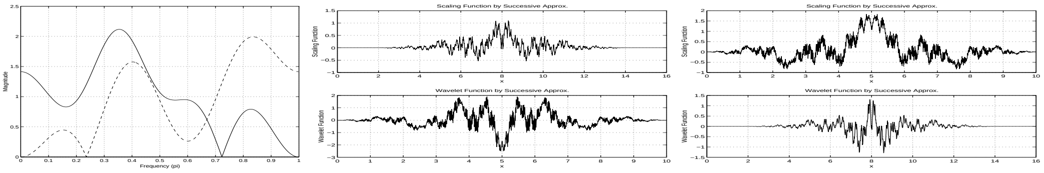

Figure 5.2: Scaling and Wavelet functions, L:Analysis R:Synthesis

The functions are now strictly restricted in the y range, though both of the pairs are still very irregular.

## 5.2 White noise signal

Having set stability conditions one can construct filters for the most general case: a white noise signal. The optimized filters can then be compared against traditional filters to observe the total improvement and the effectiveness of optimized filters.

### 5.2.1 A-II

Using the white noise test signal (figure 3.1) type A-II filters of various lengths can be designed with the stability criteria of, for example $p \leq 0.92$. The lower the limit the more effective the filters, however, a larger value gives better stability.

Table 5.3 displays the eigenvalues and decomposition case for some optimized filters build upon the white noise signal with $p \leq 0.92$.

| Biorthogonal Filter | $\max \lvert\lambda\rvert_h \,/ \lvert\lambda\rvert_{\tilde{h}}$ | Energy L | Energy H | Sum |
|---|---|---|---|---|
| 10-6 | 1.00/1.00 | 56.02% | 44.79% | 100.81% |
| 15-9 | 14.37/1.00 | 827.57% | 42.62% | 870.20% |
| 26-6R | 1.00/1.00 | 56.96% | 44.79% | 101.75% |

Table 5.3: Energy divisions of test signal

A 6-tap filter with the selected stability condition is capable of shifting about 5% more energy away from the high frequency subband than an orthogonal filter. The performance for the 26-6R filter is pretty much the same as the 10-6, however, the 15-9 variant still shows unstability.

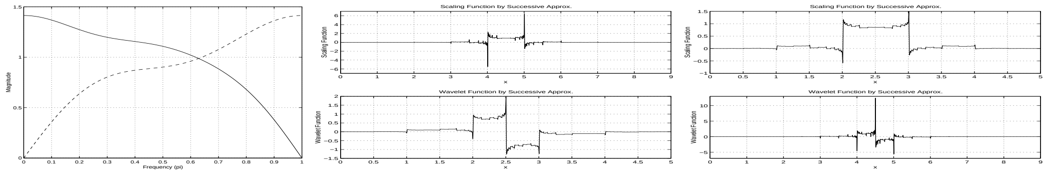

Figure 5.3: 10-6 L: Frequency response, S/W functions: M: Analysis R: Synthesis

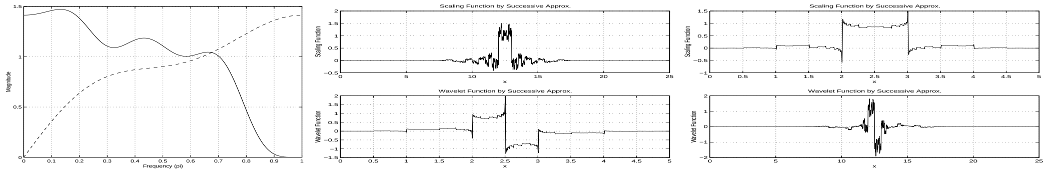

Figure 5.4: 26-6R L: Frequency response, S/W functions: M: Analysis R: Synthesis

The 10-6 filter's scaling and wavelet functions (figure 5.3) have a very squared non-smooth shape, not very different from Haar functions. Single peaks of the functions attain large values, however, this is not too problematical as the eigenvalues of $\mathbf{T_h}$ and $\mathbf{T_{\tilde{h}}}$ do not exceed 1.

In other words we only have mean-square convergence, and not pointwise or uniform convergence. Practically, these functions can give blocking-like effects if they are used for image compressing [15]. The extra regularity of the 26 tap filter gives the synthesis scaling function a refiner continuous shape than for the smaller 10 tap filter.

One can also attempt to follow the other full stabilization approach and demand that the filter's transition matrix do not contain any eigenvalues above 1. The constrain (5.4) can first be set for $\tilde{h}[n]$ when the wavelet filter is constructed, while in the second phase the condition (5.3) is required from the scaling filter $h[n]$.

| Biorthogonal Filter | Energy L | Energy H | Sum |
|---|---|---|---|
| 10-6 | NC | 16.19% | NC |
| 15-9 | NC | 11.17% | NC |
| 26-6R | NC | 16.19% | NC |

Table 5.4: Energy divisions of test signal

As table 5.4 shows the eigenvalue restrains only work for the construction of the first wavelet filter but they result in a convergence failure (NC) when one attempts to design the scaling filter. Forming eigenvalue constrains is therefore definitively not an applicable method for type A filters.

### 5.2.2 Weighted B-II

In the type B optimization case both the analysis and scaling filters are created in the same run; setting maximum bounds of 1 for the eigenvalues can therefore be expected to work well with this construction strategy. One can combine this with the weighting concept (section 4.6.1) and use heavier loads for the high frequency zone $d[n]$ to transfer as much energy as possible to the low frequency zone, $\min \sum (c[n]^2 + Q\ d[n]^2)$. One can in fact totally neglect the $c[n]$ part, thus removing the need of having any weights at all. By ignoring the low frequency part in computing the objective function all of the focus of the minimization can be turned to the high frequency zone alone. Both scaling filters are still made in the same procedure, however, we demand nothing from the analysis scaling filter but that is's transition matrix should not have any eigenvalues above unity (5.3). The analysis wavelet filter must accompany this condition (5.4) along with energy minimization of the high frequency zone: $\min \sum d[n]^2$.

New filters were created with this slightly modified method of weighted type B optimization and optimization / signal decomposition on the white noise signal gives table 5.5.

| Biorthogonal Filter | $\max \lvert\lambda\rvert_h\ /\lvert\lambda\rvert_{\tilde{h}}$ | Energy L | Energy H | Sum |
|---|---|---|---|---|
| 6-14 | 1.00/1.00 | 64.21% | 46.04% | 110.26% |
| 9-7 | 1.00/1.00 | 89.61% | 38.52% | 128.13% |
| 17-11 | 1.00/1.00 | 73.03% | 37.99% | 111.03% |

Table 5.5: Energy divisions of test signal

Finding the optimized solutions requires experimentations with different starting values to obtain the optimum filter. However, the filters are quite good at shifting as much energy as possible to the low frequency zone for this white noise signal. The optimized 9-7/17-11 filters transfer respectively 9% and 7% more energy away from the H zone than the CDF / Coiflet filters of equal lengths (table 3.2). The cumulative energy increase with the designed filters exceeds slightly what is normal under the traditional filters.

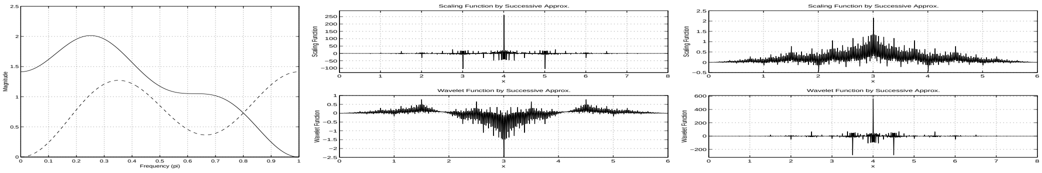

Figure 5.5: 9-7 L: Frequency response, S/W functions: M: Analysis R: Synthesis

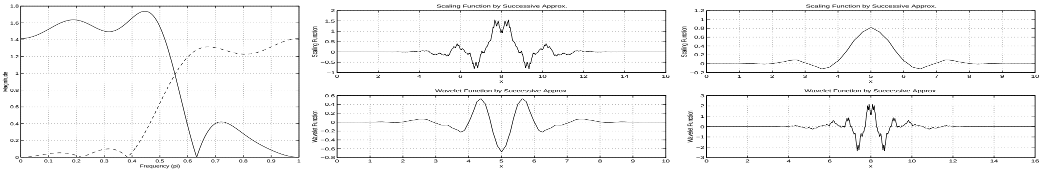

Figure 5.6: 17-11 L: Frequency response, S/W functions: M: Analysis R: Synthesis

The functions for the 9-7 variants shows a great deal of discontinuity. This irregularity does give the functions finite energy in $L^2$ but there is no smoothness and one can not expect uniform convergence. Functions for the 17-11 filter show a very different picture and are quite smooth.

Back in section 3.3 a figure of merit was defined, for the optimized 9-7 filter we obtain a noticeable higher merit value of $F = 2.51$ while the 17-11 returns $F = 1.99$.

## 5.3 P2 and P3

Along with the stability and weighted type B filter concept from section 5.2.2 one can now go back to the test signals P2 and P3. We are now in a position to shift maximum energy away from the H zone retaining full function stability, at least in $L^2$.

Optimizing P2/P3 under different filters with the eigenvalue constrains and then decomposing we obtain table 5.6 (P2) while table 5.7 shows the situation for the P3 signal.

| Biorthogonal Filter | Energy L | Energy H | Sum ($\bar{e}$) |
|---|---|---|---|
| 6-14 | 160.37% (11.01) | 11.88%(10.58) | 172.26% (10.80) |
| 9-7 | 111.57% (10.96) | 21.71% (10.73) | 133.28% (10.85) |
| 10-6 | 195.44% (11.06) | 17.62% (10.63) | 213.06% (10.85) |
| 10-6R | 102.30% (10.98) | 25.83 % (10.64) | 128.13% (10.81) |
| 15-9 | 271.40% (11.08) | 10.39% (10.56) | 281.79% (10.82) |
| 17-11 | 79.30% (10.90) | 21.45% (10.76) | 100.76% (10.83) |
| 17-11R | 75.64% (10.85) | 25.18% (10.79) | 100.83% (10.82) |

Table 5.6: Energy divisions of signal P2

Comparing with tables 4.8 and 4.10 we clearly see that the energy shifting away from the H zones is no longer as powerful as in chapter 4. On the plus side the problematical large energy increases for the low frequency part are now absent. Judging from the traditional orthogonal

| Biorthogonal Filter | Energy L | Energy H | Sum ($\bar{e}$) |
|---|---|---|---|
| 6-14 | 101.20% (7.75) | 1.00%(7.23) | 102.21% (7.49) |
| 9-7 | 101.56% (7.76) | 0.40% (6.66) | 101.96% (7.21) |
| 10-6 | 101.06% (7.82) | 0.39% (6.64) | 101.45% (7.23) |
| 10-6R | 100.87% (7.78) | 0.41 % (6.67) | 101.29% (7.22) |
| 15-9 | 110.49% (7.89) | 0.40% (6.94) | 110.90% (7.41) |
| 17-11 | 102.00% (7.80) | 0.46% (6.87) | 102.46% (7.33) |
| 17-11R | 99.72% (7.80) | 0.49% (6.69) | 100.22% (7.24) |

Table 5.7: Energy divisions of signal P3

and biorthogonal filters (tables 4.9 and 4.11) point of view the optimized stable filters generated still do a very noticeable better job of transferring energy to the L frequency zone. The entropy counts for the P2 signal are not any lower, although, for P3 the average entropy has gone through a reduction. Supplementary regularity for filters clearly helps in depressing the entropy at the expense of lesser energy shifting.

Figures 5.7 and 5.8 show how the 10-6R filter turns out to be for P2 and P3.

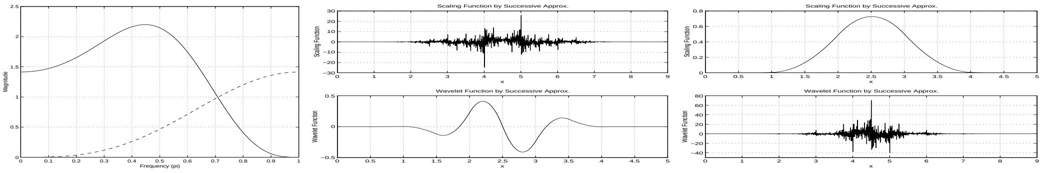

Figure 5.7: P2 L: Frequency response, S/W functions: M: Analysis R: Synthesis

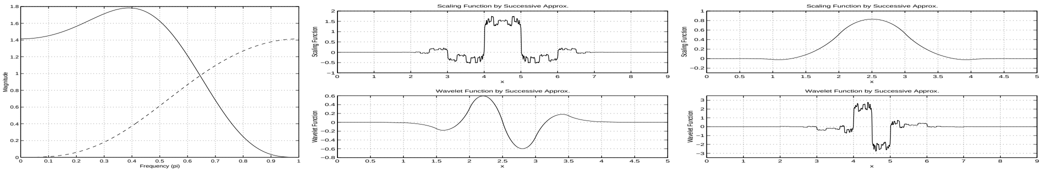

Figure 5.8: P3 L: Frequency response, S/W functions: M: Analysis R: Synthesis

Looking at the plots of the scaling and wavelet functions we see that the synthesis scaling functions are generally very smooth. On the other hand, there is more variety over the shape of the analysis scaling functions. The analysis scaling function and it's wavelet pair clearly show a high frequency pattern for the P2 signal, which is more or less an approximation of the P2 signal itself (figure 4.19). For the P3 signal we see a brick-like pattern which resembles the edges in P3 (figure 4.20).

## 5.4 Summary

This chapter introduced methods to create stable biorthogonal filters for transferring / minimizing the energy of a decomposed signal. Armed with the stability criterias it is possible to design filters with acceptable properties for practical usage. For type A filters, however, strict stability may not always be achievable. One can therefore move over to weighted type B filter concept and combine it with constrains on the eigenvalues, to acquire stable energy shifting wavelet filters.

# Chapter 6

# Images

Till now our main focus has been on constructing optimized wavelet filters for selected signals. In this chapter we will try to expand the same concepts to digital images.

Three selected gray-scale images of size 256x256 are used for testing purposes.

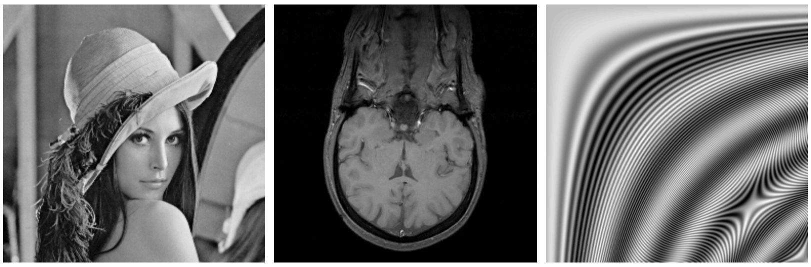

Figure 6.1: L: M1, M: M2, R:M3

The first image M1 is basically a low-frequency image (entropy 7.21 bits per pixel) while the second picture is from a MR scan of the human brain with low and some high frequency details in it, with an entropy of 6.03 b/p. The third image M3 is an artificial image where the high frequency pattern plays a dominating role, it also gives the largest rate of 8.44 b/p.
The difference between two images can be measured via several metrics, one alternative is the peak signal to noise ratio (PSNR) which can be computed as $PSNR = 10 \log_{10} \frac{255^2}{MSE}$ where $MSE = \frac{1}{MN} \sum \sum ((f(x,y) - f_r(x,y))^2$. M and N define the size of the image, in our case $M = N = 256$. A PSNR value above 32 is often regarded as if the image $f_r(x,y)$ is nearly error free from the original $f(x,y)$.

## 6.1 Type A filters, 2D extension

Traditional wavelet decomposition of two-dimensional data follows a separable method where an image is transformed row-by-row and then column-by-column using the same filters. We will stick to this approach, though, the filter optimization algorithms developed so far require an one-dimensional signal (or it's Fourier transform) as the main input data. For images the optimization criteria can easily be extended; assuming $F(\hat{\omega}_x, \hat{\omega}_y)$ is the (discrete) Fourier transform of the image, the two-fold strategy for type A filters becomes:

- construct the analysis wavelet filter by minimizing $\sum\sum |F(\hat{\omega}_x, \hat{\omega}_y)|^2 |G(\hat{\omega}_x)|^2 |G(\hat{\omega}_y)|^2$ and then
- fashion the analysis scaling filter by minimizing $\sum\sum |F(\hat{\omega}_x, \hat{\omega}_y)|^2 |H(\hat{\omega}_x)|^2 |H(\hat{\omega}_y)|^2$

Instead of running the procedure in Fourier domain a wavelet decomposed energy reduction method is a much better alternative in terms of speed, just as for the one-dimensional case. In this view the filter generation process takes the following shape:

- build $g[n]$ by minimizing the analysis wavelet filter decomposed image $\sum\sum w_{g[n]}(x, y)^2$
- generate $h[n]$ based upon $g[n]$ by minimizing the analysis scaling filter decomposed image $\sum\sum w_{h[n]}(x, y)^2$

$w(x, y)$ refers to the decomposed image and the subscript with lowercase letters indicates which one filter participates in the decomposition. The decomposed image constructed via this method will at least have as much energy as possible shifted away from the high frequency quadrat. Optimization via full two-dimensional wavelet decompositions is a slow and tedious process but nevertheless for off-line filter generating purposes it is fully executable.

Constructing optimized filters for the three test images, we decomposed them using the two-dimensional algorithmical versions of 10-6, 15-9 and 26-6R filters; and computed how they shifted the energies of the images, additionally the entropy values were also calculated which are given in parenthesis. As one can observe in table 6.1 the optimized filters have done a reasonable job and reduced the energy in the HH subimage to a minimum.

The bits per pixel values given in the right-most column $(\overline{b/p})$ are the average entropy values and this performance can theoretically be obtained if each zone is coded separately.

| Filter | Energy LL | Energy LH | Energy HL | Energy HH | Sum $(b/p)$ |
|---|---|---|---|---|---|
| M1 | | | | | |
| 10-6 | 469.84% (10.53) | 0.89% (6.43) | 1.25% (6.57) | 0.00% (2.60) | 472.00% (6.53) |
| 15-9 | 635.00% (10.77) | 0.82% (6.38) | 1.06% (6.47) | 0.00% (2.07) | 636.89% (6.43) |
| 26-6R | 343.23% (10.32) | 0.75% (6.33) | 1.02% (6.46) | 0.00% (2.60) | 345.02% (6.43) |
| M2 | | | | | |
| 10-6 | 570.01% (9.86) | 1.62% (5.89) | 1.72% (5.91) | 0.00% (2.12) | 573.37% (5.94) |
| 15-9 | 3184.95% (11.02) | 2.66% (6.29) | 2.48% (6.26) | 0.00% (1.59) | 3190.10% (6.29) |
| 26-6R | 498.51% (9.77) | 1.47% (5.82) | 1.57% (5.84) | 0.00% (2.12) | 501.57% (5.89) |
| M3 | | | | | |
| 10-6 | 3134945.3% (12.51) | 1730.5% (10.02) | 1730.5 % (10.02) | 0.79% (4.74) | 3138407.1 % (9.33) |
| 15-9 | 7375.10% (11.41) | 56.59% (7.83) | 56.59% (7.83) | 0.35% (3.76) | 7488.65% (7.71) |
| 26-6R | 1644802.9% (12.57) | 1242.6% (10.00) | 1242.6% (10.00) | 0.79% (4.74) | 1647289.0% (9.33) |

Table 6.1: Decomposition, energy divisions

The filters are successful in shifting energy from the HH subband but this is in fact the only positive aspect of these filters. The situation is not satisfactory at all for the LH and HL parts where the filters don't have any major impact. This shows up very clearly for the mixed-frequency M3 image which overall obtains a very poor performance.

The images decomposed using the 10-6 filters are also shown in figure 6.2. All the decomposed images in the text are shown in a logarithmical scale ($w_{disp}(x, y) = \log_{10}(1 + |w(x, y)|)$), on a normal scale only features from the LL quadrat would have come forth.

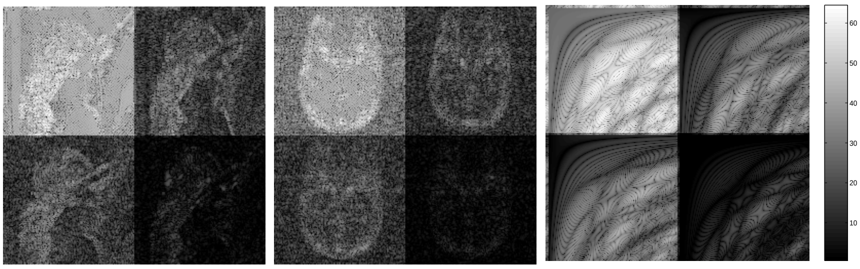

Figure 6.2: Decomposition using 10-6, L: M1, M: M2, R:M3

Variable scaling shows that despite the fact that little energy is concentrated on the high frequency parts they still contain a certain degree of structure. The entropy figures given in table 6.1 are founded upon integer roundups of the decomposed image, this roundup has some negative effects which are clearly visible (figure 6.3) where the reconstructed integer quantized images are displayed.

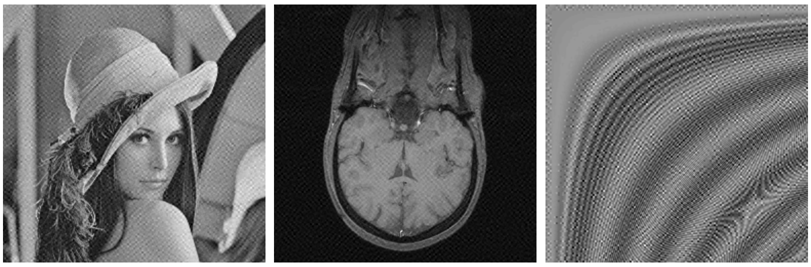

Figure 6.3: Integer quantization, L: PSNR=28.63, M: PSNR=30.85, R: PSNR=9.54

For all the images we observe that a high-frequency pattern have been overimposed on them, this is due to the quantization of the HH frequency zone. The integer roundoff in LL, LH and HL zones has very little effect and the errors in the reconstructed image can be attributed directly to the HH part alone. The quality of the reconstructed images can be increased many folds if we for example increase the accuracy and store 1 decimal digit for the HH zone, on the negative side this increases the entropy quite heavily.

### 6.1.1 Increasing stability

The methods introduced in chapter 5 provide opportunities to make the filters more stable at the loss of lesser energy movement away from the high-high subimage.

For type A filter the analysis wavelet filter is constructed at first and we can easily set any limit on the energy: $\sum g[k]^2 \leq p$. Setting $p = 1.10$ and generating optimized 10-6 filter for the images we obtain the results shown in table 6.2.

| Filter | Energy LL | Energy LH | Energy HL | Energy HH | Sum ($b/p$) |
|---|---|---|---|---|---|
| M1 | 98.62% (8.17) | 0.16% (5.12) | 0.27% (5.53) | 0.06% (4.33) | 99.12% (5.79) |
| M2 | 97.52% (7.19) | 0.35% (4.63) | 0.49% (4.84) | 0.11% (3.91) | 98.49% (5.14) |
| M3 | 73.80% (9.19) | 8.09% (7.51) | 8.09% (7.51) | 9.49% (6.73) | 99.49% (7.73) |

Table 6.2: Decomposition, energy divisions, $p = 1.10$

The energy increase is now not so dramatic, though, the stability problem has not been totally cured. Figure 6.4 shows the scaling and wavelet functions for the optimized 10-6 filter for the M2 image.

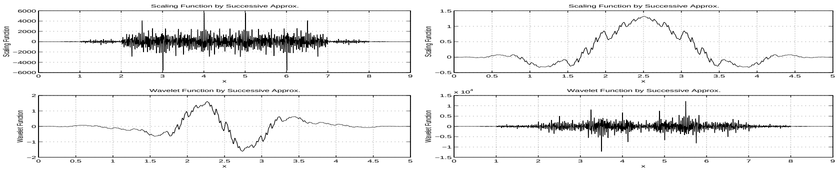

Figure 6.4: Scaling and Wavelet functions, L:Analysis R:Synthesis

Despite the extra constrain the long reaching values for one of the pair are still existent. They can be reduced further by enlarging $p$ but then the effect of energy shifting is depressed ever more.

Apart from the various degrees of unstability the other main hindrance in our optimization strategy can be attributed to the LH and HL zones where no direct attempts are done to minimize the energy. As we know from chapter 4 a wavelet filter which minimizes the energy forces a large gain on the scaling filter. The LH and HL zones in our image decomposition process therefore go through energy increases when the scaling filter is applied, effectively downgrading the full effect of the wavelet filter. Low energy in LH, HL and HH zones is what we should aim for in able to hope for any coding gain with these kinds of filters. A possible method to fix this problem is to construct all the filters in the same run, this can be accomplished by type B filters. One can also try to move away from the traditional image decomposition of four quadrats.

## 6.2 Type A filters, Three part decomposition

To reduce the contamination from the scaling filter on the LH and HL zones we propose a three part image decomposition and optimization with two sets of type A optimized filters as a substitute to the normal four parts quadratic image decomposition with one filter pair.

Filter optimization will now follow a new approach:

- treat the image row-by-row and construct an analysis wavelet filter $g_r[n]$ which transfers as much energy as possible to the left side of the decomposed image; $\min \sum \sum w_H(x, y)^2$
- generate analysis scaling filter $h_r[n]$ by $\min \sum \sum w_L(x, y)^2$, reduce the energy of the left zone

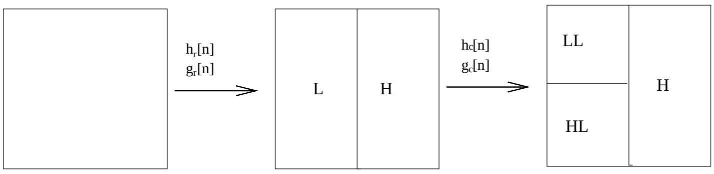

Figure 6.5: Three part image decomposition

- in the second phase only focus on the left side of the image column-by-column: design a wavelet filter $g_c[n]$ which shifts as much energy as possible to the upper-left quadrat; $\min \sum\sum w_{HL}(x,y)^2$
- construct the analysis scaling filter $h_c[n]$ pair by following $\min \sum\sum w_{LL}(x,y)^2$

$w(x,y)$ still refers to the decomposed image, while the subscript in uppercase points to different subimages. As a result of this optimization strategy we end up with two sets of filters; one optimized for row-by-row decomposition and the other for column-by-column separation for the low-frequency part of the row-by-row decomposed image. Extra set of filters give us more freedom and we also avoid contamination effects as the same filters are not executed on both columns and rows.

New algorithms were designed to solve this kind of problem, and the results from the test runs on the images can be seen in table 6.3. The firmness parameter $p$ was set at: $p \leq 1.10$ for both row and column wavelet filters.

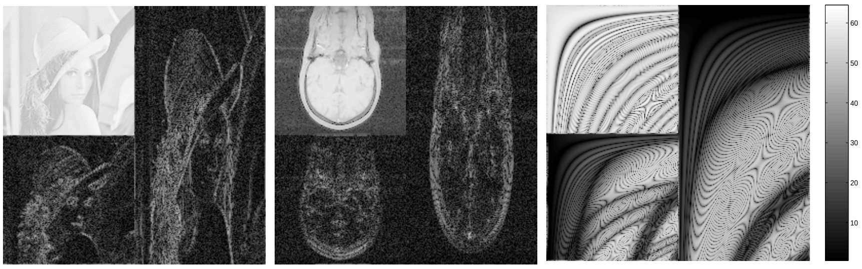

Figure 6.6: Three part decomposition using 10-6, L:M1, M:M2, R:M3

The average entropy values in the right-most column are calculated as $\overline{b/p} = (e_{LL} + e_{HL} + 2e_H)/4$ which yields the average number of bits required to code the integer quantized image if the three zones are coded independently. $e_X$ refers to the entropy of zone $X$ (see figure 6.5). The average entropy values are now somewhat better.

In table 6.3 the energy reduction in the H area is not as good as previously for the HH zone for the full four part decomposition but considering the LH and HH zones from table 6.2 together then it's apparent that this new algorithm is much better and shifts more power to the left side. A longer wavelet filter of length 9 with one extra degree of freedom is obviously more effective at minimizing energy in the H zone than a shorter 6-tap filter but as a disadvantage the scaling filter then enlarges the energy of HL and LL zones.

Having transferred as much energy as possible to the LL zone we can try to quantize the high frequency areas tightly. The images in figure 6.7 shows the case where in addition to integer quantization over the whole image both HL and H zones are quantized to an uniform integer interval of 5, i.e. figures in these zones get rounded up to $.., -5, 0, 5, 10, ..$ .

| Filter | Energy LL | Energy HL | Energy H | Sum ($b/p$) |
|---|---|---|---|---|
| M1 | | | | |
| 10-6 | 99.50% (8.18) | 0.06% (4.34) | 0.23% (4.69) | 99.80% (5.48) |
| 15-9 | 122.94% (9.06) | 0.26% (5.49) | 0.20% (4.64) | 123.40% (5.96) |
| 26-6R | 99.46% (8.17) | 0.07% (4.45) | 0.23% (4.69) | 99.77% (5.50) |
| M2 | | | | |
| 10-6 | 98.62% (9.45) | 0.22% (5.70) | 0.52% (3.30) | 99.37% (4.89) |
| 15-9 | 106.84% (7.15) | 0.98% (4.34) | 0.36% (4.06) | 108.18% (5.40) |
| 26-6R | 98.53% (6.87) | 0.23% (4.25) | 0.52% (4.32) | 99.29% (4.94) |
| M3 | | | | |
| 10-6 | 74.90% (9.21) | 7.02% (6.43) | 16.41% (6.79) | 98.34% (7.31) |
| 15-9 | 197478.83% (12.14) | 242.30% (7.98) | 15.90% (6.33) | 197737.04% (8.20) |
| 26-6R | 76.79% (9.24) | 8.57% (6.75) | 16.41% (6.79) | 101.78 % (7.39) |

Table 6.3: Decomposition, energy divisions

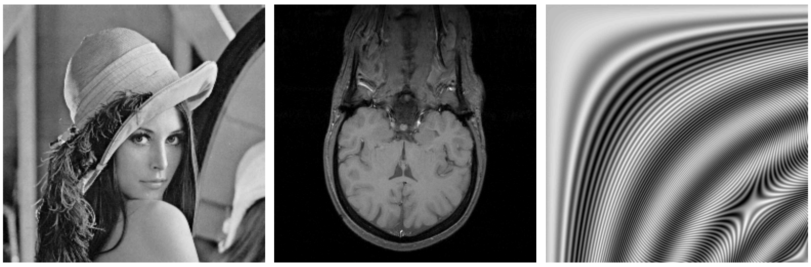

Figure 6.7: 10-6, 5-level quantization: L: M1 (PSNR= 45.40) M: M2 (PSNR=46.53) R: M3 (PSNR=46.76)

The quality of the images even after such a 5 level quantization remains very good (figure 6.7). The M2 image gives a PSNR of 46.53 at 3.21 $\bar{b}/p$, slightly above the CDF 9-7's 46.28 at 3.12 $\bar{b}/p$.

The stability of the filters is reflected in the scaling functions shown in figure 6.8 for the M2 image. The right hand side represents the functions for row-by-row filtering while functions on the left side are used to split the L subimage into HL and LL. The synthesis scaling functions look quite similar to each other, it may therefore be possible to combine them together.

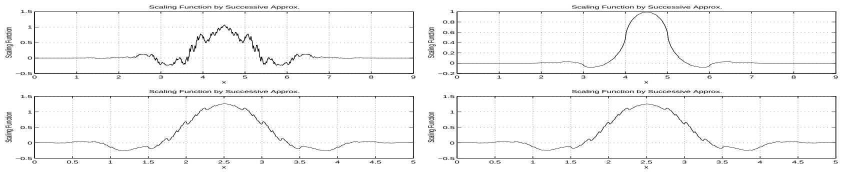

Figure 6.8: Scaling functions, T:Analysis B:Synthesis, L:Column R:Row

For comparation purposes the outcomes from CDF 9-7 filter on the test images with three level decomposition are in table 6.4

The results from the optimized filters compared with CDF 9-7 follow a mixed pattern. For the M1 image the special constructed 10-6 filters do shift more energy away from the H area

| Filter | Energy LL | Energy HL | Energy H | Sum ($b/p$) |
|---|---|---|---|---|
| M1 | 99.55% (8.18) | 0.06% (4.24) | 0.93% (4.54) | 100.54% (5.37) |
| M2 | 98.92% (6.77) | 0.20% (4.06) | 0.42% (4.17) | 99.55% (4.79) |
| M3 | 77.58% (9.23) | 7.10% (5.59) | 14.54% (5.99) | 99.22% (6.70) |

Table 6.4: Decomposition, energy divisions

than the CDF filter, for the M2/M3 images we notice the opposite pattern. The stability parameter $p$ was unchanged for all the test images and this flexibility has not been utilized in the optimization.

One can therefore try to vary $p$ and find how this influences the energy in H/HL zones and the average entropy. This can be seen in figure 6.9 where the total energy of the H and HL zones is shown, while figure 6.10 displays the average entropy of the decomposed images as $p$ varies.

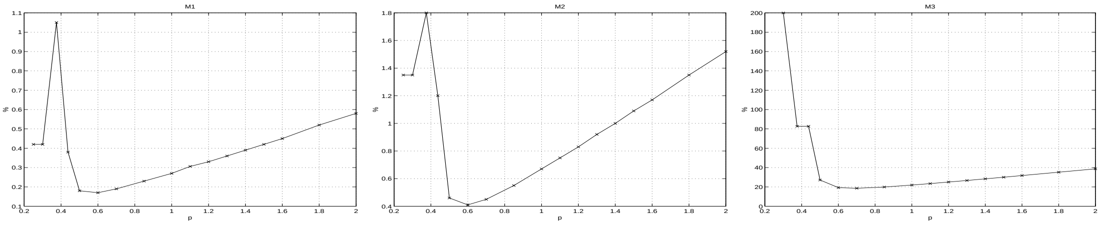

Figure 6.9: Energy H+HL, L:M1, M:M2, R:M3

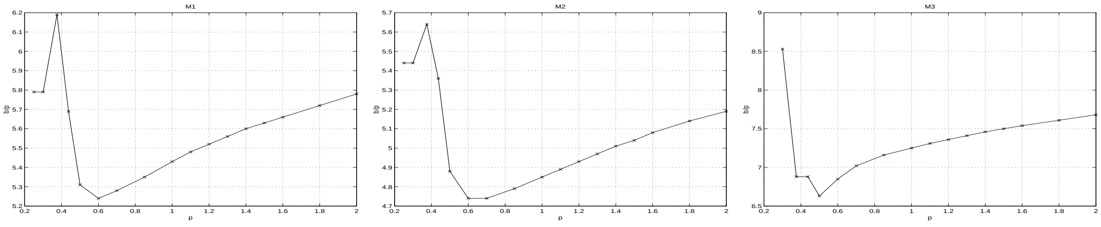

Figure 6.10: Average entropy, L:M1, M:M2, R:M3

The quantity of energy in the H/HL zones closely follows the average entropy, and for values of $p$ around $0.5-0.6$ a minimum point can be observed where both these figures are minimized. Such values of $p$, however, do not offer any stability. For that we need to increase $p$ to values at least nearby 1, unfortunately this then does not give favourable reductions for energy nor entropy.

The different kinds of type A filters generated either via a direct two dimensional extension of the one dimensional case or the three part decomposition with different filters for columns / rows are unfortunately quite useless for practical image compression. With energy constrains the situation can be improved, however, selecting the stability versus optimization becomes a problem of it's own. We will now end the discussion of type A filters and change focus to type B filters where it will turn out that one can almost ignore the stability conditions.

## 6.3 Type B filters, 2D extension

In chapter 4 type B filters were originally constructed to minimize the energy of a decomposed signal; however, by combining this with special weights the algorithm can be turned into a powerful energy shifting method suitable for digital images.

Just as for type A filters we can slightly modify our previous type B filter generating algorithms to make them suitable for images. Instead of minimizing $\sum(c[n]^2 + d[n]^2)$ we now aim for

$$\min \sum\sum w(x,y)^2 \quad = \min \sum\sum (w_{LL}(x,y)^2 + w_{LH}(x,y)^2 + w_{HL}(x,y)^2 + w_{HH}(x,y)^2)$$

which should yield filters who shrink the energy of a decomposed image as much as practicable.

In chapter 4 we found that filter designed via this method shift some of the energy across the subbands to achieve the desired goal; consequently this method was not found very well suitable for compression purposes. In spite of this it can be interesting to observe the effects of such energy minimization for images. Algorithmically this becomes a very time demanding operation as the optimization routine needs to decompose an image in every iteration. The energy of images can often reach large values, and on top of it the optimization criteria now demands many more iterations than for type A filters implying that finding the optimized filters will require some time.

To be able to acquire results within a decent time frame the algorithm was speeded up somewhat by slightly modifying the optimization objective function to $\min \frac{10000}{\gamma} \sum\sum w(x,y)^2$ where $\gamma = \sum\sum f(x,y)^2$ is the power of the original image. If the tolerance levels of the optimization routine are kept at the same level then this can be viewed as an approximation to the full optimization problem where the least significant digits are discarded. This is in fact also necessary to make sure that the large energy figures along with small filter coefficients / derivate approximations do not turn the optimization into a badly scaled problem [2]. The solutions should be accurate enough for experimental purposes at least.

Constructing optimized filters of size 6-14, 9-7 and 17-11 and thereafter running them on the test images gives table 6.5.

| Filter | Energy LL | Energy LH | Energy HL | Energy HH | Sum ($b/p$) |
|---|---|---|---|---|---|
| M1 | | | | | |
| 6-14 | 99.19% (8.13) | 0.08% (4.48) | 0.21% (5.05) | 0.17% (4.97) | 99.67% (5.66) |
| 9-7 | 98.05% (8.05) | 0.11% (4.82) | 0.30% (5.53) | 0.33% (5.55) | 98.81% (5.99) |
| 17-11 | 97.07% (8.00) | 0.16% (5.24) | 0.33% (5.78) | 0.34% (5.67) | 97.92% (6.17) |
| M2 | | | | | |
| 6-14 | 98.14% (6.69) | 0.26% (4.23) | 0.49% (4.59) | 0.18% (4.20) | 98.14% (4.93) |
| 9-7 | 94.76% (6.71) | 0.37% (4.64) | 0.69% (4.93) | 0.79% (5.27) | 96.61% (5.39) |
| 17-11 | 92.95% (6.99) | 0.48% (4.91) | 0.72% (5.14) | 0.78% (5.32) | 94.96% (5.59) |
| M3 | | | | | |
| 6-14 | 74.73% (9.21) | 7.36% (7.14) | 7.36% (7.14) | 9.50% (6.31) | 98.96% (7.45) |
| 9-7 | 72.07% (9.17) | 6.93% (6.02) | 9.93% (6.02) | 11.56% (5.24) | 97.49% (6.61) |
| 17-11 | 72.75% (9.19) | 6.99% (6.42) | 6.99% (6.42) | 11.22% (5.62) | 97.98% (6.91) |

Table 6.5: Decomposition, energy divisions

The first thing we notice is that the total energy reductions are by far impressive, none of the filters have been able to reduce the total energy below 90%. The justification to this once again lies with the LH and HL zones; a small gain on a wavelet filter results in a large gain on the scaling filter which in turn contaminates the results in the mixed frequency zones, LH and HL. The optimization routine is left with no other choice but to find a suitable compromise where no filter plays a leading minimizing role. One big benefit of this, however, is that the filters we obtain become very stable and the transistion matrix of the scaling filters does not return any eigenvalues above 1.

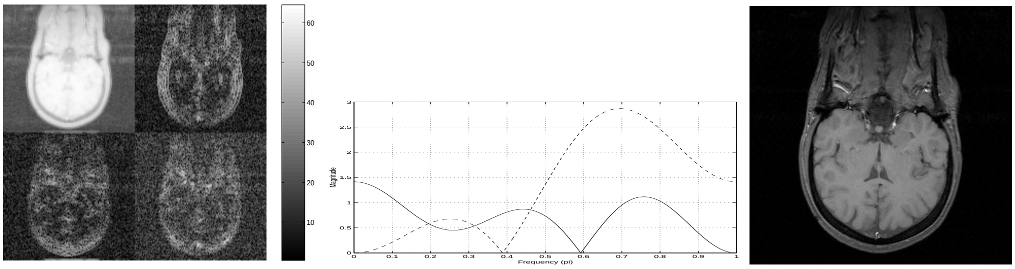

Figure 6.11: 9-7, L: M2 decomposed M: frequency response R: reconstructed image (PSNR=45.05)

The image M2 decomposed using the optimized 9-7 filter is shown on left in figure 6.11, one can witness how the HL and LH zones each capture horizontal and vertical aspects while the HH zone mainly consists of high frequency structure. This is very much similar to how traditional wavelet filters work. The image on right in figure 6.11 is reconstructed after quantizing the LH,HL and HH zones to an uniform interval of 5. The entropy comes at 3.69 $\overline{b/p}$, still quite a bit higher than the CDF 9-7 filter's 3.07 $\overline{b/p}$ under the same procedure, but it is much better than our type A filter variants.

## 6.4 Weighted type B filters

The good behavior of type B filters generated in the previous section makes them good candidate for further work. The goal of shifting as much energy as possible to one particular zone can hence be fulfilled if the above filters are combined with the weighted minimization concept from section 4.6.1.

The optimization objective function to achieve this aim can be formulated as:

$$\min \sum \sum (Q_1 w_{LL}(x,y)^2 + Q_2 w_{LH}(x,y)^2 + Q_3 w_{HL}(x,y)^2 + Q_4 w_{HH}(x,y)^2)$$

To make sure that most of the power falls into the LL zone we can set $Q_2, Q_3$ and $Q_4$ to some large values while retaining $Q_1$ at unity.

As a starting point setting $Q_2 = Q_3 = Q_4 = 100$ and generating filters under this new objective function we obtain table 6.6. The optimizations are performed using the approximation $\min \frac{10000}{\gamma} \sum \sum w(x,y)^2$ for reasons previously mentioned.

Looking at the table we are for the first time able to witness good results from optimized filters. With these filters the energy is increased in the LL subimage at the cost of reduction in the HH zone. The total energy increase is very modest and the average entropy levels are also low.

| Filter | Energy LL | Energy LH | Energy HL | Energy HH | Sum ($b/p$) |
|---|---|---|---|---|---|
| M1 | | | | | |
| 6-14 | 122.26% (9.09) | 0.12% (4.97) | 0.23% (5.30) | 0.00% (2.46) | 122.63% (5.45) |
| 9-7 | 100.92% (8.34) | 0.06% (4.34) | 0.15% (4.77) | 0.01% (3.40) | 100.73% (5.21) |
| 17-11 | 102.06% (8.44) | 0.05% (4.29) | 0.14% (4.74) | 0.01% (3.25) | 102.27% (5.18) |
| M2 | | | | | |
| 6-14 | 100.09% (7.59) | 0.29% (4.59) | 0.36% (4.70) | 0.02% (2.96) | 100.77% (4.96) |
| 9-7 | 101.71% (7.08) | 0.19% (4.13) | 0.27% (4.37) | 0.03% (3.11) | 102.22% (4.67) |
| 17-11 | 105.19% (7.27) | 0.18% (4.11) | 0.26% (4.37) | 0.02% (2.86) | 105.68% (4.65) |
| M3 | | | | | |
| 6-14 | 95.17% (9.42) | 10.10% (7.80) | 10.10% (7.80) | 2.67% (6.62) | 118.04% (7.91) |
| 9-7 | 93.54% (9.42) | 7.89% (7.06) | 7.89% (7.06) | 3.78% (5.82) | 113.11% (7.34) |
| 17-11 | 114.75% (9.56) | 7.56% (6.05) | 7.56% (6.05) | 2.78% (4.55) | 132.67% (6.55) |

Table 6.6: Decomposition, energy divisions

Apart from the 6-14 filter both 9-7 and 17-11 variants give us better entropy results for M1 and M2 than what we obtain under the standard filters (table 6.7). Comparing with the popular CDF 9-7 and the Coiflet 17-11 we can verify that the major earnings for the optimized filters come from the HH zone where the optimized filters outperform the standard variants.

| Filter | Energy LL | Energy LH | Energy HL | Energy HH | Sum ($b/p$) |
|---|---|---|---|---|---|
| M1 | | | | | |
| CDF 9-7 | 99.55% (8.18) | 0.06% (4.24) | 0.16% (4.73) | 0.04% (4.07) | 99.82% (5.31) |
| Coiflet 17-11 | 99.55% (8.21) | 0.05% (4.23) | 0.15% (4.72) | 0.04% (4.02) | 99.98% (5.30) |
| M2 | | | | | |
| CDF 9-7 | 98.92% (6.77) | 0.20% (4.06) | 0.32% (4.37) | 0.08% (3.71) | 99.54% (4.73) |
| Coiflet 17-11 | 99.39% (6.79) | 0.19% (4.05) | 0.31% (4.36) | 0.07% (3.65) | 99.98% (4.72) |
| M3 | | | | | |
| CDF 9-7 | 77.58% (9.23) | 7.10% (5.59) | 7.10% (5.59) | 7.65% (5.00) | 99.44% (6.35) |
| Coiflet 17-11 | 79.52% (9.23) | 7.04% (5.01) | 7.04% (5.01) | 7.08% (4.69) | 100.70% (5.99) |

Table 6.7: Decomposition, energy divisions

The artificial M3 image once again is in a special position where our filters are not able to directly compete with the traditional filters. The energy amount for optimized filters is low in the HH zone, though, the large high frequency content does not give any entropy savings.

Optimized 6-14 filters are also not doing well for any image at all if we only look at the average $b/p$ results in the table. As the 14-tap wavelet filter has the majority of free parameters in this system the wavelet filter gives good performance for the high frequency zone at the expense of the mixed frequency zones. This directly results in unstability, only for the 9-7 and 17-11 filter types the eigenvalues of $\mathbf{T_h}$ and $\mathbf{T_{\tilde{h}}}$ do not exceed 1. Optimization with 6-14 should therefore ideally have been performed with eigenvalue constrains.

How does these new optimized filters look like ? The frequency response of the 9-7 filter optimized for the images are shown in figures 6.12, 6.13 and 6.14 along with the scaling and wavelet functions. The filter coefficients are also provided in Appendix A. The goal of all these filter is the same and the 9-7 filter has only two degrees of freedom, hence the close resemblance to each others.

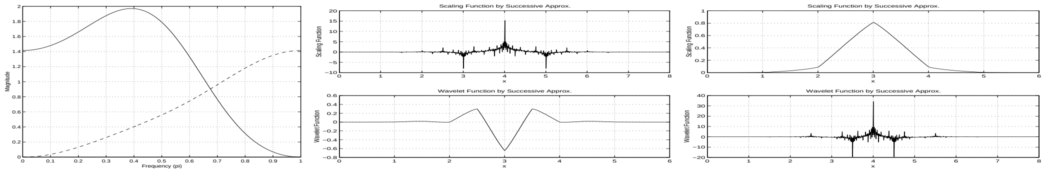

Figure 6.12: M1 L: Frequency response, S/W functions: M: Analysis R: Synthesis

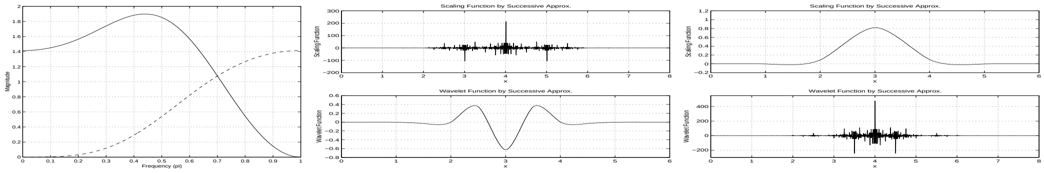

Figure 6.13: M2 L: Frequency response, S/W functions: M: Analysis R: Synthesis

Unlike many of our preceding filters the frequency response in the figures does not reach very large values and the same is valid for the scaling and wavelet functions, although, there are a few individual large peaks in the non-smooth functions we have $L^2$ convergence. The maximum eigenvalue for all these 9-7 filters comes at 1, quantization is hence a reasonable stable operation.

There are also similarities between the functions for M1 and M2 as both images contain a lot of low frequency structure. The irregular functions representing the high frequency details though are clearly different.

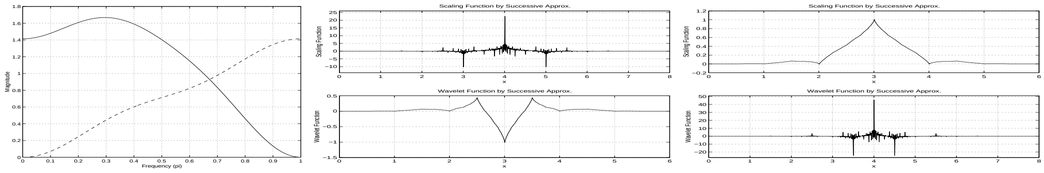

Figure 6.14: M3 L: Frequency response, S/W functions: M: Analysis R: Synthesis

The M2 image decomposed using the optimized filter is on left in figure 6.15 while on right the HL, LH and HH zones have been quantized to an uniform integer interval of 5 before reconstruction (entropy 3.01 $\bar{b}/p$). The image's quality remains pretty good.

One may also wonder whether it's possible to further enlarge $Q_2 = Q_3 = Q_4$ to for example 1000 or even 100000. It turns out that regarding the energy in the mixed/high frequency zones 100 times as worth as of that in the LL zone is enough for general experimental purposes and for our images a larger value would only have effect on the decimal level. A substitute to this weighting is to set all the values in the LL zone to zero ($Q_1 = 0, Q_2 = Q_3 = Q_4 = 1$), the effect of this on the results is also minor.

Looking at table 6.6 it is moreover clear that the energy reduction in the HL and LH subimages is non-existing, in these zones the optimized variants give very similar results to the CDF 9-7/Coiflet 17-11 filters. We can safely conclude: what distinguishes these optimized type B

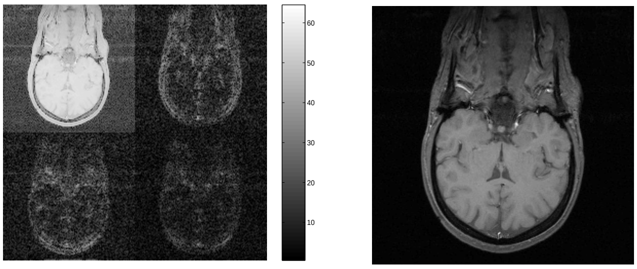

Figure 6.15: 9-7, L: M2 decomposed R: quantized reconstructed image (PSNR=43.73)

filters from the rest is that they shift more power only from the HH subimage to the other zones.

### 6.4.1 High frequency preservation

The movement of energy from the HH zone can be looked upon in different ways. Either simply as a lossless method to code the image more efficiently; the optimized filters constructed for a particular image or a class of images can give somewhat better coding performance as it turned out to be the case for the M1 and M2 images.

One other way to look at the energy shifts is to realize that after moving energy from the high frequency part one is in a position where the HH area can be quantized without that necessarily forcing a loss to all of the high frequency structure in the image. The effect of this will be very small on low frequency images like M1 and M2, however, the consequences should be directly noticeable for high frequency pictures like M3.

The reconstructed image on left in figure 6.16 was first decomposed using the optimized 9-7 filters, where the HH zone was truncated. The image on right shows the resulting image where the same procedure was performed using the CDF filter. Figure 6.17 shows a close up of the center parts.

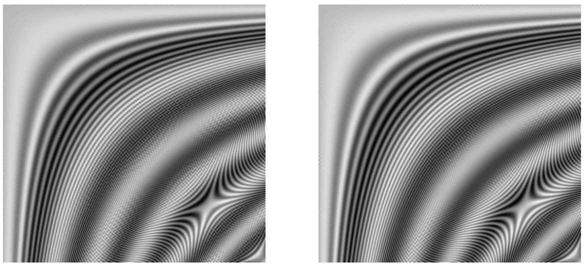

Figure 6.16: 9-7, HH truncation L: optimized filters (PSNR=18.89) R: CDF 9-7 (PSNR=18.83)

The PSNR values are similar, however, it is relative easy to observe that the high frequency pattern roughly in the middle of the image has been removed leaving back an empty ′patch′ which is clearly bigger in the CDF 9-7 reconstructed image than the optimized filter′s outcome. This, of course, comes at the expense of higher information content in the LL/LH/HL subbands (4.41 $\overline{b/p}$) for the image on left than the one on right (3.82 $\overline{b/p}$). An optimized 6-14 filter would give even higher entropy values but with additional preservation of high frequency patterns.

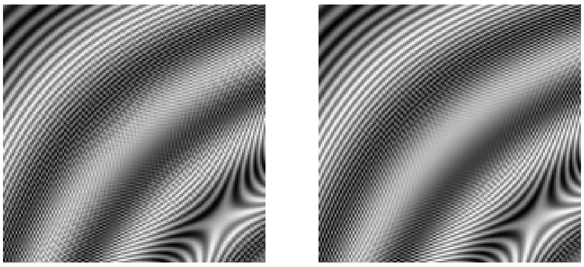

Figure 6.17: 9-7, HH truncation L: optimized filters R: CDF 9-7

The close correspondence of the PSNR figures for the images in this case demonstrates a well known fact that mean square estimates do not always portrait the differences very well.

## 6.5 Type B filters, Unequal weights

Starting with $\min \sum \sum (w_{LL}(x,y)^2 + Q_2 w_{LH}(x,y)^2 + Q_3 w_{HL}(x,y)^2 + Q_4 w_{HH}(x,y)^2)$ we notice that having equal loads $Q_2 = Q_3 = Q_4$ transfers energy away from the HH zone but this does not help much to restrict energy in the HL and LH zones. This is very apparent for the M3 image where the mixed frequency zones (table 6.6) store a great deal of energy. For such images we can use heavier weights for the HL and LH zones to additionally depress the power in these subbands. To experiment with this some test runs were there done utilizing different values for the cross weights (figure 6.18).

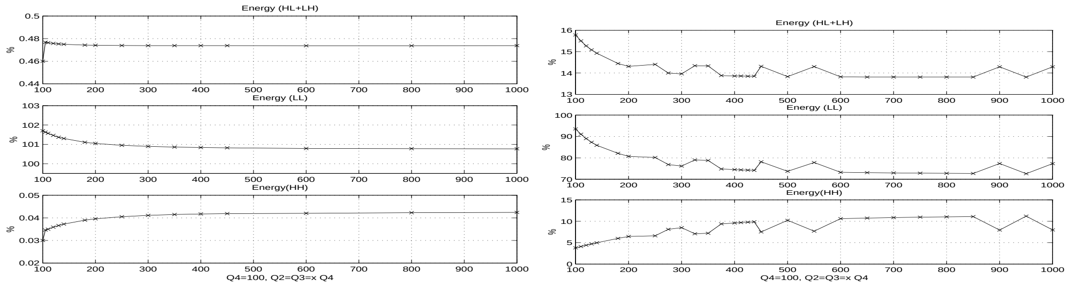

Figure 6.18: Different weights, L: M2 R: M3

The simulations formed 9-7 filters for both M2 and M3 images for various values of $100 \leq Q_2 = Q_3 \leq 1000$ with $Q_1 = 1$ and $Q_4 = 100$. As one can observe in figure 6.18, increasing $Q_2 = Q_3$ more or less helps reduce the amount of energy in the LH and HL zones (top row) for the M3 image. On the M2 image the effect is marginal. Further on, the LL zone also takes part in the decrease. The drawback of increasing weights for the cross zones is that the power of the high-high frequency part starts to raise (bottom row).

From the outcomes it is possible to conclude that for the M2 image the total energy of the decomposed image remains pretty much at the same level. Larger $Q_2/Q_3$ reduce the energy in the LL zone but any gain is destroyed by the energy/entropy augmentation in the HH subimage.

The situation is somewhat different for the M3 image as the HL and LH zone store a great deal of the image's power. Some enlarged energy in the HH zone can on that ground be tolerated, at the benefit of energy savings in the LH,HL and LL zones. From the plots though, it does look like that there may not exist any global minimum for any given value of $Q_2/Q_3$. Table 6.8 demonstrates the case for optimized 9-7/17-11 filters with weights set up as $Q_1 = 1, Q_2 = Q_3 = 400$ and $Q_4 = 100$.

| Filter | Energy LL | Energy LH | Energy HL | Energy HH | Sum ($b/p$) |
|---|---|---|---|---|---|
| M1 | | | | | |
| 9-7 | 100.10% (8.24) | 0.06% (4.27) | 0.14% (4.71) | 0.02% (3.69) | 100.34% (5.23) |
| 17-11 | 101.20% (8.37) | 0.05% (4.27) | 0.14% (4.73) | 0.01% (3.43) | 101.42% (5.20) |
| M2 | | | | | |
| 9-7 | 100.84% (6.99) | 0.19% (4.10) | 0.27% (4.36) | 0.04% (3.24) | 101.35% (4.67) |
| 17-11 | 103.15% (7.15) | 0.18% (4.09) | 0.26% (4.36) | 0.03% (3.03) | 103.63% (4.66) |
| M3 | | | | | |
| 9-7 | 74.50% (9.19) | 6.93% (5.56) | 6.93% (5.56) | 9.60% (5.10) | 97.98% (6.35) |
| 17-11 | 80.69% (9.25) | 6.88% (5.25) | 6.88% (5.25) | 7.35% (4.83) | 101.82% (6.15) |

Table 6.8: Decomposition, energy divisions

On M1 and M2, unequal weights do not bring any overall improvement which is only apparent for the M3 image. In this table some energy quantity has been shifted away from the HL and LH zones to the HH part, which does perform well for images with dominating high frequency structure as this lowers the total energy sum.

Often natural images do not contain high frequency details of large quantity; the schemes in the next two sections are anyway aimed primarily towards images such as M3: For high frequency pictures one may ask about whether one can directly let the optimization routine handle the optimization of $Q_2$ and $Q_3$. This is unfortunately not feasible as long as the cost functions are the same. A solution to the problem

$$\min \sum\sum (w_{LL}(x,y)^2 + Q_2 w_{LH}(x,y)^2 + Q_3 w_{HL}(x,y)^2 + 100 w_{HH}(x,y)^2)$$

with constrains on $Q_2$ and $Q_3$:

$$Q_2 \geq 100, \qquad Q_3 \geq 100$$

is for example obviously achieved when $Q_2 = Q_3 = 100$.

### 6.5.1 Energy contraction

What one can attempt at, however, is to select optimized $Q_2/Q_3/Q_4$ who fulfill an external cost function; $\min \sum\sum w(x,y)^2$. This can be run as an optimization outside the main filter forming optimization:

- find optimum values for $Q_2 = Q_3$ and $Q_4$ to minimize
  $\sum\sum w(x,y)^2 = \min \sum\sum (w_{LL}(x,y)^2 + w_{LH}(x,y)^2 + w_{HL}(x,y)^2 + w_{HH}(x,y)^2)$
  - generate filters to satisfy
    $\min \sum\sum (w_{LL}(x,y)^2 + Q_2 w_{LH}(x,y)^2 + Q_3 w_{HL}(x,y)^2 + Q_4 w_{HH}(x,y)^2)$

In this heavily time consuming procedure the inner loop is used to construct filters who minimize the energy of the decomposed image with special weights given to certain subimages, while the outer optimization just tries to minimize the energy of the decomposed image.

For the outer loop we also need to set constrains such as $Q_2 = Q_3 \geq Q_4$ to make sure that energy is really shifted off these zones. By selecting symmetrical weights for the LH/HL subimages

$Q_2 = Q_3$ we do loose some freedom, but we avoid situations where the optimization routine can pick a very large value for $Q_2$ in relation to $Q_3$. This can totally destroy the energy shifting intent behind the algorithm. To accelerate the convergence one can even preselect a value for $Q_4$ and end up with an one-variable optimization.

Under the M3 image and selecting the weight boundaries $100 \leq Q_2 = Q3 \leq 1000$ and $Q_4 = 100$ the double optimization for the 9-7 filters returns $Q_2 = Q_3 = 1000$. After many hours the routine simply converges to the largest upper limit, giving maximum emphasis to the mixed frequency zones. The optimized filter along with optimized weights $Q_2 = Q_3 = 1000$ and $Q_4 = 100$ executed on M3 gives table 6.9.

| Energy LL | Energy LH | Energy HL | Energy HH | Sum $(b/p)$ |
|---|---|---|---|---|
| 72.56% (9.17) | 6.90% (5.78) | 6.90% (5.78) | 11.24% (5.11) | 97.61% (6.46) |

Table 6.9: Decomposition, energy divisions

The upper limit does minimize the total energy but it also propagates the entropy. The reduction in energy is very small compared with table 6.8 where the cross weights were at only 400. It is quite clear that this double optimization is a somewhat restrictive method. The power of the decomposed image can indeed be dropped along with low energy for the cross subimages but the main disadvantage lies with energy growth in the HH subimage and entropy increases.

### 6.5.2 Entropy contraction

The concept from the previous section can be modified somewhat to design optimized 9-7 filters where the weights $Q_2 = Q_3$ and $Q_4$ are chosen under constrains $100 \leq Q_2 = Q_3 \leq 1000$, and $100 \leq Q_4 \leq 1000$, with the outer optimization criteria being the minimization of the entropy rather than the energy:

- find optimum values for $Q_2 = Q_3$ and $Q_4$ to minimize the average entropy of $w(x, y)$
  - generate filters to satisfy $\min \sum\sum(w_{LL}(x,y)^2 + Q_2 w_{LH}(x,y)^2 + Q_3 w_{HL}(x,y)^2 + Q_4 w_{HH}(x,y)^2)$

The entropy is a very image dependent measurement, nevertheless, low entropy should ensure that the approximation done by the scaling and wavelet functions is of good quality. Using the entropy as the cost function is generally not a good idea: As the optimization routine searches for an appropriate answer it may well come cross a solution which imposes large energy and entropy increases. This can make it impossible to compute the entropy values. However, in this algorithm one is only in concern with the entropy at the outer level, where optimized filters for particular weights have already been found, which makes it safe to use the entropy calculations.

Using the entropy in the outer optimization is also a much quicker alternative than the energy. The entropy figures are normally quite small $< 10$ which implies swift convergence.

Changing to the entropy contraction scheme and running the procedure over 9-7 filters and M3 image returns $Q_2 = Q_3 = 405.76$ as the optimized loads. The value for $Q_4$ was set constant at $100$ to get fast convergence and to make certain that some energy is also shifted away from

$M3 : Q_1 = 1, Q_2 = 405.76, Q_3 = 405.76, Q_4 = 100.00$:

| Energy LL | Energy LH | Energy HL | Energy HH | Sum ($b/p$) |
|---|---|---|---|---|
| 74.44% (9.19) | 6.93% (5.54) | 6.93% (5.54) | 9.65% (5.05) | 97.96% (6.33) |

Table 6.10: Decomposition, energy divisions

the high-high subimage. The optimized filters for these weights give as expected good average entropy results (table 6.10), slightly better than the CDF 9-7.

The mixed frequency zones are given very high consideration and the objectionable feature of course is that the entire energy amount in the HH subimage remains quite high, but then this is after all an energy shifting method with flexible emphasis on the LH/HL subimages.

The same algorithm under M1 and M2 returns respectively the weights $Q_2 = Q_3 = 100.43$ and $Q_2 = Q_3 = 100.00$. For M1 and M2 the most favorable weights are more or less identical to the preelected $Q_4(= 100)$. It follows that selecting equal weights for the LH,HL and HH zones gives more or less optimum results for low frequency images in regard to shifting as much energy as possible to the LL zone and also when it comes to reducing the average entropy of the decomposed image.

If one is primary interested in reducing the information content of a decomposed image then finding optimized weights for high frequency images is a useable option though this may not shift maximum energy away from the subzones. Finding good starting values for the algorithm can also become a slight problem.

## 6.6 White noise image

To test the optimized filters on as a general image as possible they can be designed over an white noise image (M4). Figure 6.19 displays the experimental white noise image scaled up to cover all the 8 available bits. The centered logarithmical scaled Fourier transform is displayed on right.

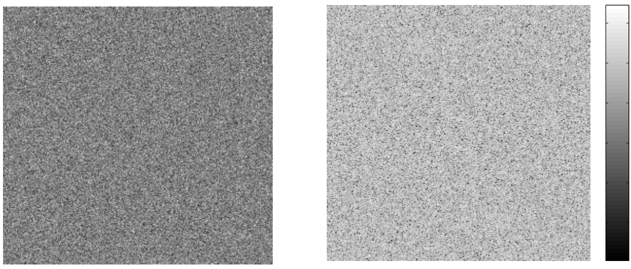

Figure 6.19: L: M4, R: Log Fourier transform of M4

The entropy of M4 comes at exactly 8.00 $\overline{b/p}$. Comparing the results given by optimized filters with more standard filters we should be able to obtain absolute figures on how much effective energy shifting filters can become.

Decomposing M4 with CDF 9-7 and Coiflet 17-11 we get table 6.11.

In accordance with our findings from chapter 3 both filters have attempted to shift some modest amount of energy to the LL frequency zone. The total energy is somewhat increased but does

| Filter | Energy LL | Energy LH | Energy HL | Energy HH | Sum ($b/p$) |
|---|---|---|---|---|---|
| CDF 9-7 | 26.83% (8.04) | 25.68% (8.01) | 25.40% (8.00) | 24.35% (7.97) | 102.27% (8.00) |
| Coiflet 17-11 | 28.76% (8.09) | 25.58% (8.00) | 25.34% (8.00) | 22.55% (7.91) | 102.25% (8.00) |

Table 6.11: Decomposition, energy divisions

not have any negative impacts upon the entropy. This stability has no doubt played a major role behind the success of these filters.

Running the optimization algorithm to construct effective filters to shift energy away from the LH,HL and HH zones by setting $Q_2 = Q_3 = Q_4 = 1000$ and thereafter running them on M4 gives table 6.12.

| Filter | Energy LL | Energy LH | Energy HL | Energy HH | Sum ($b/p$) |
|---|---|---|---|---|---|
| 6-14 | 92.14% (8.91) | 27.19% (8.05) | 26.84% (8.04) | 8.03% (7.18) | 154.21% (8.04) |
| 9-7 | 66.63% (8.68) | 26.98% (8.04) | 26.78% (8.04) | 10.85% (7.39) | 131.26% (8.04) |
| 17-11 | 86.53% (8.86) | 26.89% (8.04) | 26.76% (8.03) | 8.42% (7.21) | 148.61% (8.04) |

Table 6.12: Decomposition, energy divisions

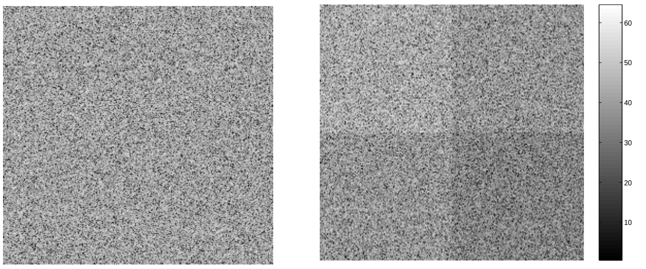

Figure 6.20: Decomposed images L: CDF 9-7, R: Optimized 9-7

The optimized 9-7 filter transfer about 14% more energy away from the HH subband than the CDF filter, the energy transfer is also visible on right in figure 6.19. All this comes at the expense of energy and entropy enlargement: A one percent energy reduction in the HH zone corresponds to around 2.5% enlargement in the low-low subimage.

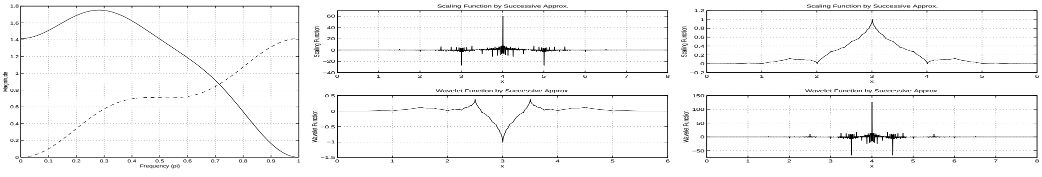

Figure 6.21: L: Frequency response, S/W functions: M: Analysis R: Synthesis

The frequency response and the scaling/wavelet functions are shown in figure 6.21 for this generalized 9-7 filter. The synthesis scaling function representing the irregularity of the image compared with figures 6.12 and 6.13 is not that smooth.

The other main extension we can perform on the algorithm is to find optimized weights to decrease the entropy levels. Following the procedure outlined in section 6.5.2 we acquire $Q_2 = Q_3 = 985.67$ and $Q_4 = 145.30$ as the loads for the 9-7 filter variant. The filters then provide us table 6.13.

M4: $Q_1 = 1, Q_2 = 985.67, Q_3 = 985.67, Q_4 = 145.30$:

| Energy LL | Energy LH | Energy HL | Energy HH | Sum ($b/p$) |
|---|---|---|---|---|
| 25.95% (8.01) | 25.11% (7.99) | 24.82% (7.98) | 24.16% (7.96) | 100.05% (7.98) |

Table 6.13: Decomposition, energy divisions

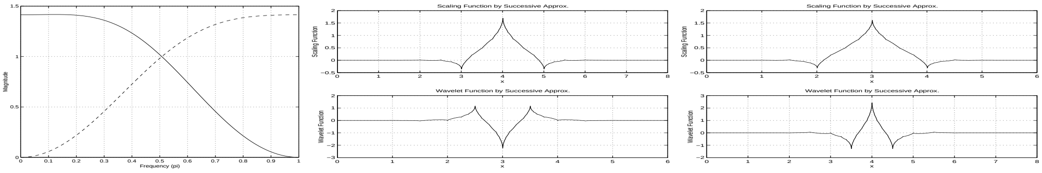

Figure 6.22: L: Frequency response, S/W functions: M: Analysis R: Synthesis

Although the average entropy can be decreased a little bit there is no energy shifting involved in the these filters (figure 6.22), the large number of $Q_2/Q_3$ diminish any effect the scaling/wavelet filter could have had on respectively LL or HH subimages. Both of the function pairs end up being continuous as no analysis/scaling filter plays any dominating role.

## 6.7 Generalized filters

The filters constructed so far have been optimized for one given image, their effectiveness can thus vary greatly on other images. Filters optimized for several images can generaed by taking account of many pictures in the optimization. Assuming we have $M$ images $f_0(x,y)$ ... $f_{M-1}(x,y)$ the optimization objective function can be modified to suit them all:

$$\min \sum\sum \Big( Q_1 w_{LL,0}(x,y)^2 + Q_2 w_{LH,0}(x,y)^2 + Q_3 w_{HL,0}(x,y)^2 + Q_4 w_{HH,0}(x,y)^2$$
$$+ \tfrac{\gamma_0}{\gamma_1}(Q_1 w_{LL,1}(x,y)^2 + Q_2 w_{LH,1}(x,y)^2 + Q_3 w_{HL,1}(x,y)^2 + Q_4 w_{HH,1}(x,y)^2)$$
$$+ \ldots.$$
$$+ \tfrac{\gamma_0}{\gamma_M}(Q_1 w_{LL,M}(x,y)^2 + Q_2 w_{LH,M}(x,y)^2 + Q_3 w_{HL,M}(x,y)^2 + Q_4 w_{HH,M}(x,y)^2)\Big)$$

$\gamma_i = \sum\sum w_i(x,y)^2$ refers to the energy of image $i$. The normalization above is linked with the first image ($w_{\ ,0}$) but any image can be selected, to ensure equal treatment is given to each image regardless the variations of energy. This strategy obviously means that all the images have to be decomposed in the optimization iterations and the time requirement also goes up. Anyway, the filters with best shifting properties for all the given images can be designed. For such general optimization it is probably not a good idea to for example run the procedure over M1,M2, M3 and M4 together as these images belong to different classes; mixing M1 and M2 on the other hand should give filters which can be used on many other images with similar frequency content.

Figure 6.23 shows the 9-7 filters optimized for both M1 and M2 images ($Q_1 = 1$ and $Q_2 = Q_3 = Q_4 = 100$).

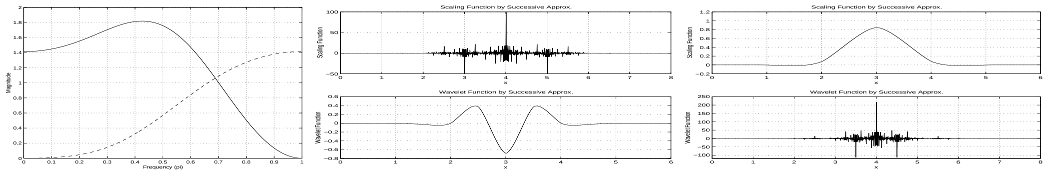

Figure 6.23: L: Frequency response, S/W functions: M: Analysis R: Synthesis

| Image | Energy LL | Energy LH | Energy HL | Energy HH | Sum $(b/p)$ |
|---|---|---|---|---|---|
| M1 | 100.50% (8.29) | 0.06% (4.31) | 0.15% (4.72) | 0.02% (3.51) | 100.73% (5.21) |
| M2 | 101.42% (7.04) | 0.19% (4.11) | 0.27% (4.36) | 0.03% (3.14) | 101.93% (4.67) |

Table 6.14: Decomposition, energy divisions

Comparing table 6.14 with the 9-7 filters in table 6.6 we remark that the results are not that different. The filters give a good compromise for both images.

It is very clear from the frequency response of the scaling filter that it is highly biased and covers a much wider frequency area than the wavelet filter. This also comes out very clear in the overlap area plot (section 3.2.1) in figure 6.24.

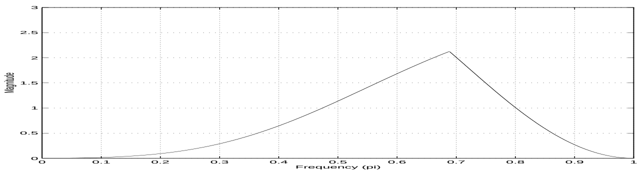

Figure 6.24: Overlap area of general optimized 9-7

The peak point occurs at $0.66\pi$, above the CDF 9-7's $0.54\pi$. The figure of merit also returns a much higher value of $F = 2.96$. The filters' energies come at $\sum h[k]^2 = 1.82$ and $\sum g[k]^2 = 0.68$, again showing a prominent difference.

## 6.8 Summary

The emphasis of this chapter has been on designing optimized separable filters for images which would transfer as much energy as possible to the low-low subimage. We investigated different models of type A filters and with stability limits one can obtain quite acceptable filters. For weighted type B filters on the hand other the stability problem more or less vanishes and the filters can manage to shift considerable energy away from the high-high frequency zone. For many natural images this can give extra gains for lossless entropy coding. The weights for high frequency images can also be set up in such a way to minimize the total energy or the average entropy by implementing a double optimization.

# Chapter 7

# Iterative decomposition

The preceding chapter was concerned with only one scale decomposition of images. The test images were decomposed into four quadrats where the optimization of filters and testing took place. Such one scale iteration is not used for practical image compression intentions, rather the iteration process is repeated with decompositions of the low-low subimages several times. In this chapter we concentrate on weighted type B filters, as they gave the most qualified results, and try to implement a 4 level dyadic iteration of the low frequency zone; this divides the image into a total of 13 different subzones.

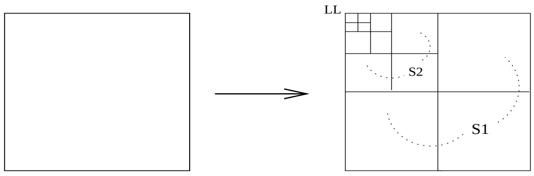

Figure 7.1: Image decomposition L: $f(x, y)$ R: $w(x, y)$

We denote the LH/HL/HH zones of the first iteration for S1, the subbands of the second iteration for S2 and so on till S4. The LL zone (included in S4) corresponds to the small quadrat on upper left. For the 256x256 test images the low-low subimage consists of 16x16 pixels.

## 7.1 Single scale optimized filters

The simplest possible method to design iterative suitable filters is to construct optimized weighted type B filters over a single iteration and use them to further iterate the low frequency zones. Taking basis in type B filters with equal weights for HL/LH/HH zones this implies that the energy should be shifted away from the HH zones also in the iterated decomposed image.

Using the same filters as those used to establish table 6.6 and calculating the energy / average entropy at each scale we obtain the next table (7.1).

The left-most column displays the energy and the average entropy of the upper-left 32x32 zone and most power of the images has definitively been shifted over there. The average entropy values are quite high for the S4 zone, however, the S4 zone only represents a small part of the decomposed image; in that regard the table is not linear.

| Filter | Energy S4 | Energy S3 | Energy S2 | Energy S1 | Sum ($b/p$) |
|---|---|---|---|---|---|
| M1 | | | | | |
| 6-14 | 153919.89% (7.99) | 613.14% (9.59) | 17.63% (7.93) | 0.36% (4.24) | 154551.04% (5.24) |
| 9-7 | 111.94% (7.70) | 2.45% (7.71) | 1.04% (6.18) | 0.24% (4.17) | 115.68% (4.77) |
| 17-11 | 135.24% (7.81) | 4.51% (8.05) | 1.27% (6.36) | 0.21% (4.10) | 141.24% (4.76) |
| M2 | | | | | |
| 6-14 | 1208.41% (7.89) | 23.99% (8.20) | 4.48% (6.36) | 0.68% (4.08) | 1237.58% (4.76) |
| 9-7 | 119.27% (7.32) | 5.40% (6.98) | 2.74% (5.77) | 0.51% (3.87) | 127.93% (4.42) |
| 17-11 | 160.70% (7.61) | 11.07% (7.38) | 3.94% (5.93) | 0.48% (3.78) | 176.19% (4.41) |
| M3 | | | | | |
| 6-14 | 185.61% (7.43) | 42.15% (8.71) | 28.26% (8.37) | 22.87% (7.40) | 278.90% (7.65) |
| 9-7 | 135.55% (7.42) | 17.77% (8.10) | 18.57% (7.53) | 19.57% (6.64) | 191.46% (6.89) |
| 17-11 | 590.74% (7.30) | 44.85% (7.60) | 27.15% (6.56) | 17.91% (5.55) | 680.67% (5.86) |

Table 7.1: Decomposition, energy divisions

The entropy figure on the right-most table can in a practical situation be obtained if each of the 13 zones are coded independently. These values are much lower than if no iteration is performed in the decomposition process (table 6.6). This is the whole advantage of iteration as the entropy is further depressed.

The 6-14 filter gives large energy increases in the S2/S3 zones than the other filters due to the wavelet filter having much more freedom than the 6-tap scaling filter, forcing a larger gain on the scaling filter. Inspection of the optimized filters with the standard filters (table 7.2) reveals that these optimized filters are, from an entropy point of view, not doing well after all. The specialized constructed filters only give improved performance for the S1 zones and that only under M1 and M2 images. As the filters are only optimized for one iteration this is not so unexpectable.

| Filter | Energy S4 | Energy S3 | Energy S2 | Energy S1 | Sum ($b/p$) |
|---|---|---|---|---|---|
| M1 | | | | | |
| CDF 9-7 | 97.44% (7.43) | 0.62% (6.81) | 0.40% (5.55) | 0.27% (4.35) | 98.75% (4.74) |
| Coiflet 17-11 | 98.43% (7.45) | 0.72% (6.91) | 0.43% (5.59) | 0.26% (4.33) | 99.85% (4.73) |
| M2 | | | | | |
| CDF 9-7 | 93.25% (6.76) | 1.86% (6.05) | 1.10% (5.06) | 0.61% (4.05) | 96.83% (4.37) |
| Coiflet 17-11 | 95.91% (6.88) | 2.10% (6.15) | 1.19% (5.11) | 0.58% (4.02) | 99.80% (4.37) |
| M3 | | | | | |
| CDF 9-7 | 51.64% (7.06) | 5.76% (6.28) | 9.68% (5.74) | 21.85% (5.40) | 88.94% (5.53) |
| Coiflet 17-11 | 61.29% (6.86) | 6.64% (5.47) | 10.33% (5.15) | 21.18% (4.90) | 99.45% (5.01) |

Table 7.2: Decomposition, energy divisions

The main feature of the selected optimized filters is the ability to shift energy away from the HH subimage at the expense of power increase in the LL subimage. While iterating, the energy in the newer LL subbands is enlarged all the times in each iteration. This also affects the cross zones LH and HL of the iterated areas with power increases, consequently the results for S2 and S3 demonstrate enlargements.

Figure 7.2 shows the distortion levels for M1, M2 and M3 at lossy coding for the optimized 9-7

filters; the dashed lines are from the CDF 9-7 filter. Such plots can be generated if we truncate different subzones of the decomposed image and reconstruct the image. This is the most effective method to test whether energy shifting really gives any advantages when the high frequency subzones are successively removed.
12 subbands (extreme right) in the plots indicate that only the HH zone in the image is truncated, with 10 zones the whole S1 is set to zero. This truncation of HH and HL/LH is repeated till only 1 zone (LL, extreme left) remains, which of course yields the lowest quality.

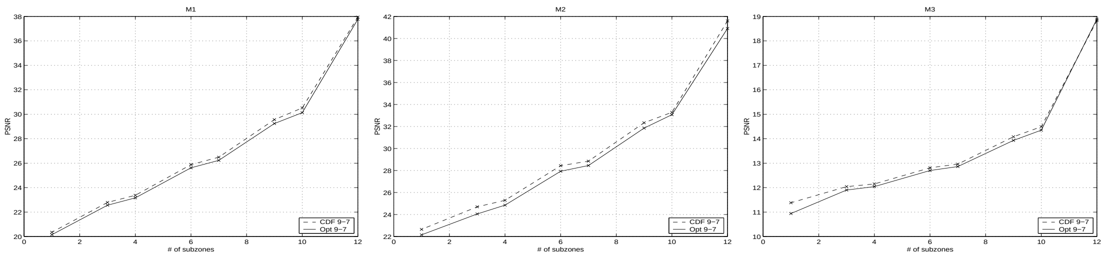

Figure 7.2: Subband truncation, L:M1 M:M2 R:M3

The plots show that the optimized filters perform worse than the popular CDF 9-7 filter. The higher entropy levels and energy shifting do not offer improve reconstruction quality if we look at the PSNR. It is easy to manifest that filters optimized for shifting energy over a single iteration are not very suitable for iterative work. Even at 12-10 subbands, where only the S1 zone is truncated, the quality of the reconstructed image is generally not better than under CDF 9-7. As we remarked in section 6.4.1 preservation of any high frequency structure has anyway little effect on the PSNR.

## 7.2 Iterative optimized filters

There are several methods one can use to make the filters take account of the iteration process. One extension of the weighted type B algorithm from section 6.4 is to find filters who minimize the energy of the iterated decomposed image.

The new optimization for a 4 level iteration can look like this:

$$\min \sum \sum (Q_1 S4_{LH/HL/HH}(x,y)^2 + Q_2 S3(x,y)^2 + Q_3 S2(x,y)^2 + Q_4 S1(x,y)^2)$$

All of the energy, but in the LL zone, is given higher weight with the aid of $Q_1, Q_2, Q_3$ and $Q_4$ in the optimization and this can help transfer the energy away from all the detail subbands.

### 7.2.1 Equal weights

Selecting equal weights $Q_1 = Q_2 = Q_3 = Q_4$ is a straightforward option and table 7.3 shows the outcomes in this circumstance; 9-7 filter with all the loads at $1000$.

Equal loads for S1, S2, S3 and S4 (LL excluded) can easily be seen as not being a reasonable choice for the optimization as with iterations the quantity of energy in the subbands increases, this must be allowed and tolerated by selecting decreasing loads for the iterative subzones. The optimization results in table 7.3 do not follow this strategy and one can observe that there is

| Image | Energy S4 | Energy S3 | Energy S2 | Energy S1 | Sum ($b/p$) |
|---|---|---|---|---|---|
| M1 | 93.58% (7.07) | 0.17% (6.26) | 0.23% (5.44) | 0.49% (4.96) | 94.48% (5.14) |
| M2 | 79.68% (6.67) | 0.49% (5.49) | 0.60% (4.86) | 1.16% (4.62) | 81.95% (4.74) |
| M3 | 32.19% (6.91) | 3.30% (6.48) | 7.05% (6.04) | 25.96% (5.89) | 68.51% (5.96) |

Table 7.3: Decomposition, energy divisions

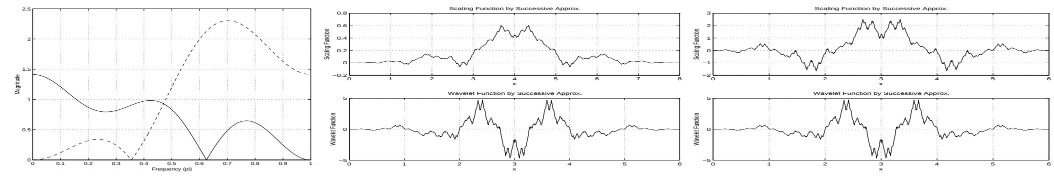

Figure 7.3: M1 L: Frequency response, S/W functions: M: Analysis R: Synthesis

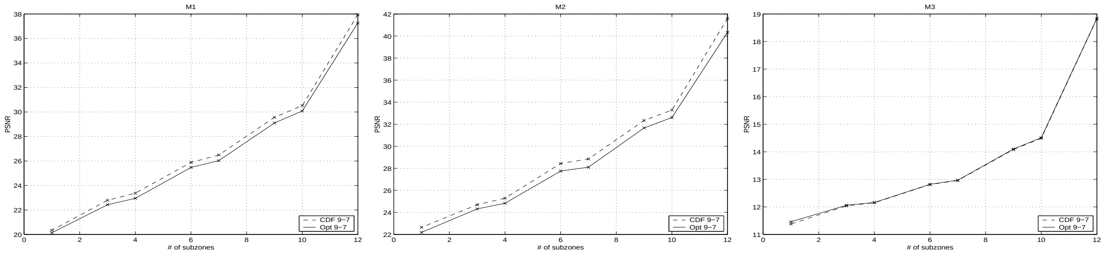

Figure 7.4: Subband truncation, L:M1 M:M2 R:M3

much energy still left in the S1 zone. There is likewise virtually no energy shifting to the S4/LL zones in the table.

The performance for these equal weight filters (solid line in figures 7.4) is also not any better. Only for the special M3 image the outcomes are comparable to the CDF filter.

### 7.2.2 Weights relative to the area

A different weight selection for $Q_1, Q_2, Q_3$ and $Q_4$ can be related with the number of pixels they cover. The outer zone S1 covers three quarter $\frac{3}{4}$ of the image while S2 only one fourth of this, $\frac{3}{4 \cdot 4}$. The S3 part $\frac{3}{4 \cdot 4 \cdot 4}$ while S4 (ignoring LL) conceals $\frac{3}{4 \cdot 4 \cdot 4 \cdot 4}$ part of the image. The weights can therefore be multiplied relative to the field. Starting with $Q_4 = 1000$ one obtains $Q_3 = 250$, $Q_2 = 62.5$ and $Q_1 = 15.6$.

#### 9-7 Filters

Selecting this optimization method we constructed optimized 9-7 filters for all three test images, the outcomes from the decompositions are in table 7.4.

The average entropy values are not any better than under CDF 9-7, however, together the total energy in S1,S2 and S3 compared with the CDF filter is lower for the optimized filters and likewise is the situation for the entropy in these zones. The artificial M3 image though is an exception. Still, one does not observe any large energy increases for the S4 zone and the effect

| Image | Energy S4 | Energy S3 | Energy S2 | Energy S1 | Sum ($b/p$) |
|---|---|---|---|---|---|
| M1 | 96.09% (7.33) | 0.44% (6.66) | 0.34% (5.50) | 0.28% (4.44) | 97.17% (4.79) |
| M2 | 88.49% (6.73) | 1.23% (5.84) | 0.89% (4.97) | 0.64% (4.16) | 91.27% (4.43) |
| M3 | 77.22% (7.34) | 9.00% (7.50) | 11.98% (6.89) | 20.35% (6.34) | 188.56% (6.51) |

Table 7.4: Decomposition, energy divisions

of energy shifting seems to be minimal, on the contrary one notices a power increase for the S1 area.

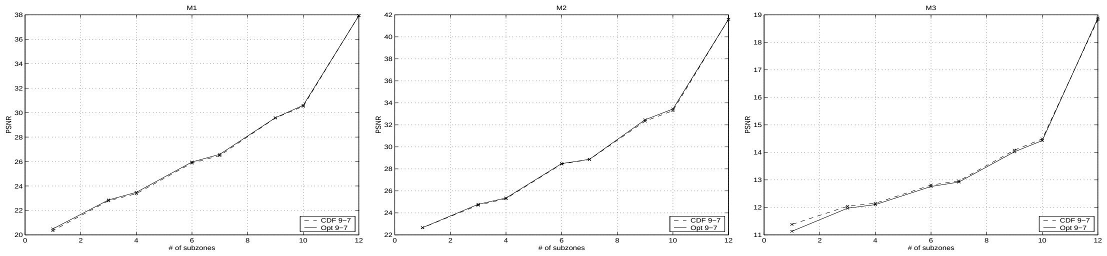

Figure 7.5: Subband truncation, L:M1 M:M2 R:M3

If we start to truncate different subzones and reconstruct the image then it's apparent (figure 7.5) that these optimized filters are capable of doing a little better job than the standard 9-7 filter. As the iterated subzones S2,S3 and S4 accommodate lower entropy, quantization of these areas result in lower $\overline{b/p}$ rates than what is the case under CDF 9-7. The PSNR values of the reconstructed image, however, remain competitive.

Figure 7.6 displays the PSNR against the average bit per pixel values and generally the optimized filters give a slightly better outcome, that is when we exclude M3.

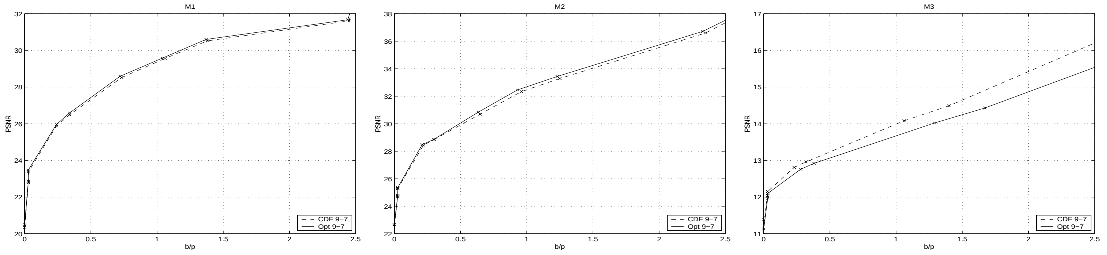

Figure 7.6: Bit rate coding, L:M1 M:M2 R:M3

The frequency response for M1 and M2 shows zero points for the optimized filters placed directly upon the unit circle which distinguishes the filters from for example the CDF 9-7 where the extra degrees are used to increase the number of vanishing moments.

The placement of zero points in the optimized filters is somewhat similar to the 9-7 filters designed in [12] where one also obtained better coding results with zero placement on the unit circle. The filter construction procedure in [12] is, however, vastly different from the one we have implemented.

The scaling and wavelet functions are now all rather smooth and the discontinuity of functions we observed in chapter 6 is gone. Constrains on filters are thus unnecessary. It follows that

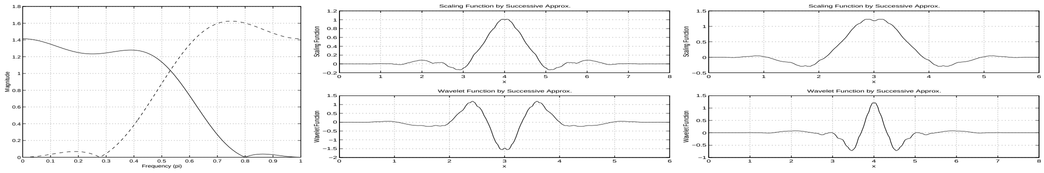

Figure 7.7: M1 L: Frequency response, S/W functions: M: Analysis R: Synthesis

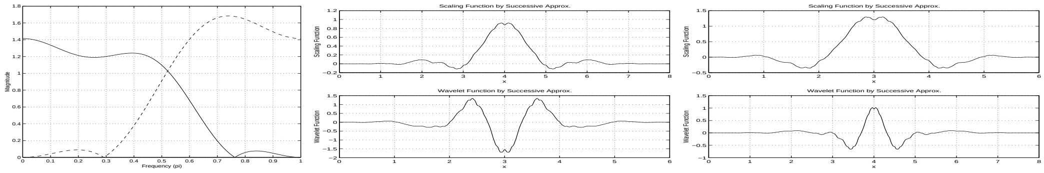

Figure 7.8: M2 L: Frequency response, S/W functions: M: Analysis R: Synthesis

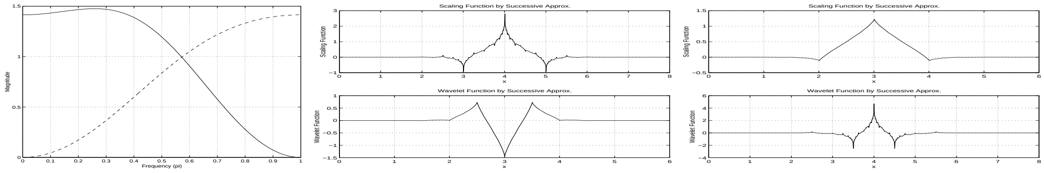

Figure 7.9: M3 L: Frequency response, S/W functions: M: Analysis R: Synthesis

smoothness for the analysis scaling function (and the synthesis wavelet filter) starts playing an increasingly important role when iterations are performed. For a single level decomposition one can manage even if one of the pairs of functions do not converge uniformly.

Figure 7.10 and 7.11 visually demonstrate the case of M1 and M2. The gains are not large, but they affirmatively are there.

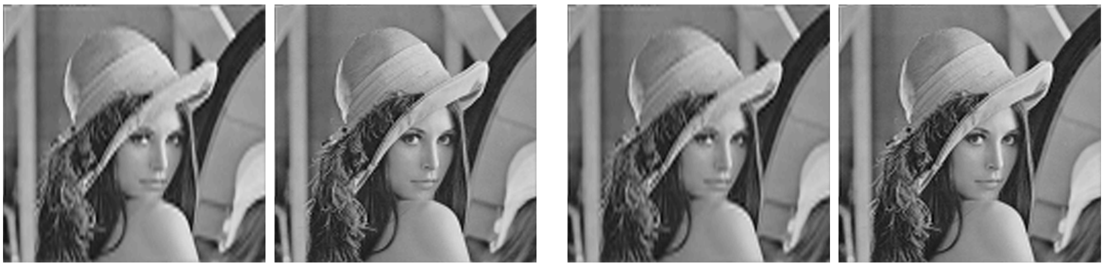

Figure 7.10: M1 CDF 9-7: LL: $\overline{b}/p = 0.73$ PSNR=28.52, L: $\overline{b}/p = 1.38$ PSNR=30.52, Optimized 9-7: R: $\overline{b}/p = 0.72$ PSNR=28.59, RR: $\overline{b}/p = 1.37$ PSNR=30.60

Is this higher gain really worth the optimization effort which must be performed on every image ? The answer to this question depends pretty much upon the type of images one is compressing. If the lossy compression process is limited to a certain class of images, for example MR scans, then designing a general optimized filter over a few images can be worthwhile. On the other hand, for general purpose image compressions this optimization process is most likely to be too time consuming. For lossless compression or very small quantization steps the CDF 9-7 filters still wins out by small margins.

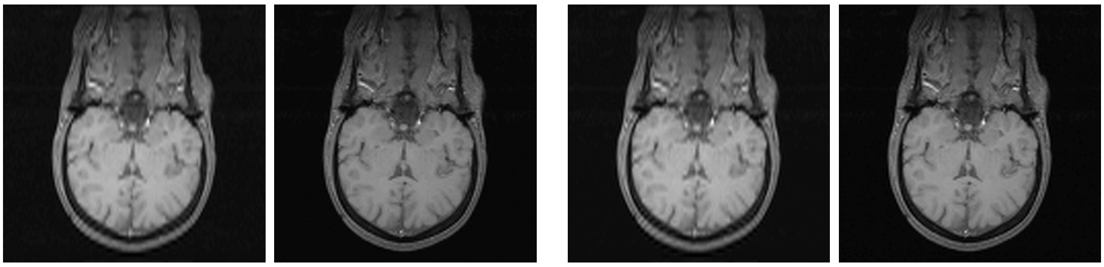

Figure 7.11: M2 CDF 9-7: LL: $\overline{b/p} = 0.64$ PSNR=30.70, L: $\overline{b/p} = 1.25$ PSNR=33.29, Optimized 9-7: R: $\overline{b/p} = 0.63$ PSNR=30.84, RR: $\overline{b/p} = 1.23$ PSNR=33.44

**17-11 Filters**

Increasing the filter lengths from 9-7 to 17-11 provides more free variables which can yield more advantageous to the optimization at the expense of extra computation time.

In table 7.5 one finds the decomposition results in case of 17-11 filters optimized using the same criteria of setting the weights for the iterated zones relative to the number of pixels they comprehend.

| Image | Energy S4 | Energy S3 | Energy S2 | Energy S1 | Sum $(b/p)$ |
|---|---|---|---|---|---|
| M1 | 95.55% (7.24) | 0.39% (6.55) | 0.33% (5.46) | 0.28% (4.45) | 96.56% (4.78) |
| M2 | 85.03% (6.65) | 0.83% (5.64) | 0.84% (4.94) | 0.64% (4.19) | 87.36% (4.44) |
| M3 | 117.92% (7.00) | 11.26% (6.00) | 12.79% (5.22) | 19.43% (5.03) | 161.42% (5.14) |

Table 7.5: Decomposition, energy divisions

Compared with Coiflet 17-11 the filters depress the total energy in S1,S2 and S3, one would expect this to enlarge the power in the S4 zone but that does not happen. Plots in figure 7.12 display the PSNR values against bit rates if the images are coded with only a limited number of subbands. The dashed lines tell the outcomes from the Coiflet 17-11 filter. The optimized filters, as in the 9-7 case, offer better coding rate for the same PSNR value. The optimized filters still do not work satisfactory for M3, for high frequency images the weights for the mixed frequency zones LH and HL should probably be selected differently.

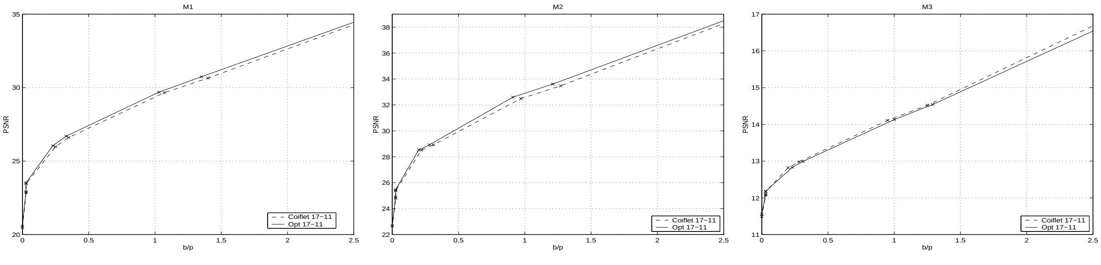

Figure 7.12: Bit rate coding, L:M1 M:M2 R:M3

The frequency response of the optimized 17-11 filter contains zero points on the unit circle which does not occur for the Coiflet 17-11 filter (figur 2.19). The optimized scaling and wavelet functions, however, exhibit a striking resemblance to the functions generated by the Coiflet

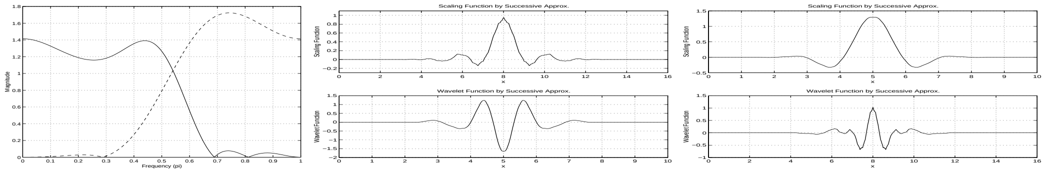

Figure 7.13: M1 L: Frequency response, S/W functions: M: Analysis R: Synthesis

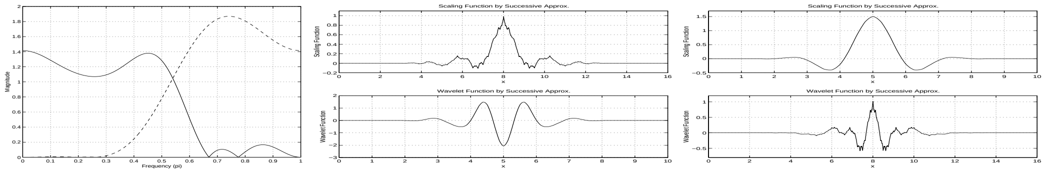

Figure 7.14: M2 L: Frequency response, S/W functions: M: Analysis R: Synthesis

variant. It is perhaps unnecessary to point to that we have no where used Bernstein basis and frequency domain design processing [17] to attain the optimized filters. The optimized functions have small irregularities in them which seems to be linked to image itself, as these minor features change for M1 and M2. The M2 image contains more high detail structure than M1, which is hence reflected by the functions.

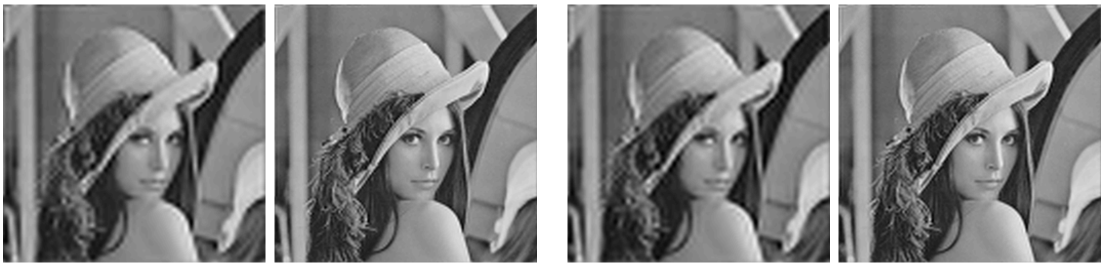

Figure 7.15: M1 Coiflet 17-11: LL: $\overline{b}/p = 0.35$ PSNR=26.61, L: $\overline{b}/p = 1.40$ PSNR=30.65, Optimized 17-11: R: $\overline{b}/p = 0.33$ PSNR=26.70, RR: $\overline{b}/p = 1.35$ PSNR=30.74

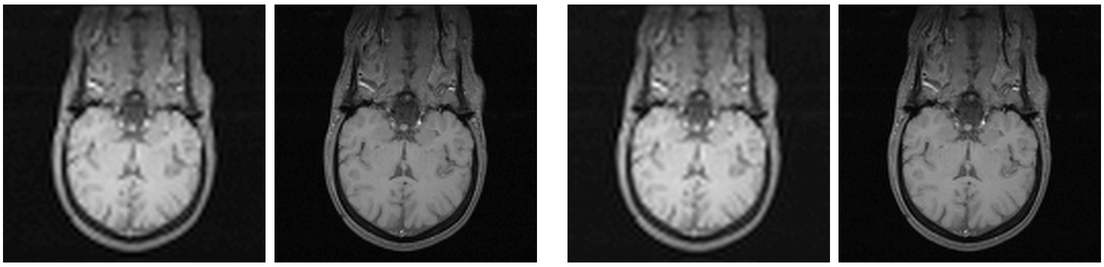

Figure 7.16: M2 Coiflet 17-11: LL: $\overline{b}/p = 0.31$ PSNR=28.92, L: $\overline{b}/p = 1.27$ PSNR=33.47, Optimized 17-11: R: $\overline{b}/p = 0.28$ PSNR=28.92, RR: $\overline{b}/p = 1.21$ PSNR=33.61

The main burden with longer lengths filters is related with the starting figures of the many free variables, and one needs to try many different values in order to locate a global minimum.

An alternative method to the optimization technique gone through in this section is to discard the LL zone altogether and select $Q_4 = 1$ for the S1 zone while the other loads can be set as $Q_3 = \frac{1}{4}, Q_2 = \frac{1}{4 \cdot 4}$ and $Q_1 = \frac{1}{4 \cdot 4 \cdot 4}$.

### 7.2.3 Optimized weights

The filters in the last section offer good performance, but the weights $Q_1, Q_2, Q_3$ and $Q_4$ are set constant and do not change. Optimized weights to shift maximum energy over to the LL zone can only be found via double-optimization procedures where the outer routine searches for the weights while the inner routine finds optimized filters given the weights. The criteria for the outer loop can as previously be set either as energy or entropy minimization.

Optimization over all 4 variables to find $Q_1$,$Q_2$,$Q_3$ and $Q_4$ is a demanding operation and may well give convergence problems if no good starting points are available. We can therefore simplify the problem by placing a linear relationship between the weights as: $Q_4 = 1000$, $Q_3 = Q_4(1 - \alpha)$, $Q_2 = Q_3(1 - \alpha)$ and $Q_1 = Q_2(1 - \alpha)$, $0 < \alpha < 1$. At the extreme points $\alpha = 0$ gives equal weights to all the subzones (except LL) while $\alpha = 1$ makes the algorithm only concentrate upon the energy minimization in S1. With $\alpha = 0.5$ the optimization routine for example accepts twice the amount of energy in the S2 zone than in S1 and so on.

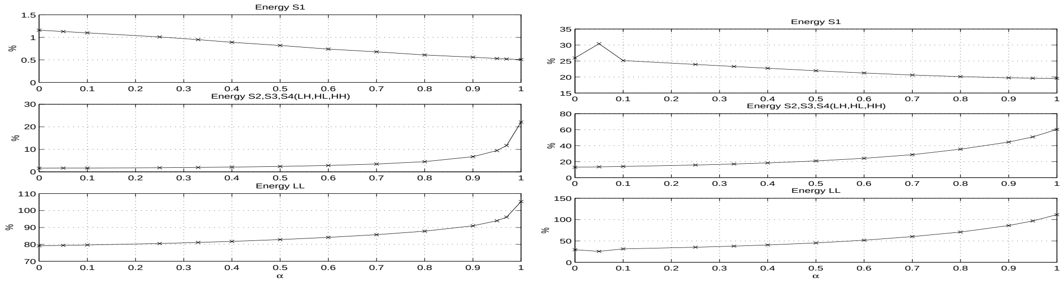

Figure 7.17: Varying $\alpha$, L: M2 R: M3

The left plot in figure 7.17 shows the effect of varying $\alpha$ on M2 and how this effects the energy in the S1 zone (top row), S2,S3 and S4$_{LH/HL/HH}$ zones (middle row) and the LL subimage (bottom row). The plot on right follows the same pattern for M3.

It is easy to observe that a large value for $\alpha$ shrinks the energy in S1 and results in an enlargement for the LL subimage. The total energy of the other zones is shown cumulative in the middle and as $\alpha$ propagates this sum grows as well. The S1 subzone is made up of three quarter of the image and maximum energy reduction there can only be obtained if we set $\alpha$ very close to 1. This, however, brings us back to the previous position in section 7.1 where the only focus is on the first iteration. If we set $\alpha$ to smaller values then we can reduce the energy in other zones but that does not yield any reduction in S1.

Sticking to the aim of shifting as much energy as possible to the LL zone one must find an $\alpha$ which minimizes the cumulative energy of the S1,S2 and S3 zones. Finding such an $\alpha$ requires anything but a double optimization procedure:

- find optimum $\alpha$ to minimize the total energy, S1+ S2+ S3
  - generate optimized filters to satisfy $\min \sum\sum Q\ S1(x,y)^2 + Q(1-\alpha)\ S2(x,y)^2 + Q(1-\alpha)^2\ S3(x,y)^2 + Q(1-\alpha)^3\ S4_{LH/HL/HH}(x,y)^2$

With $Q = 1000$ this optimization strategy returns $\alpha = 0.3214$ for M1, $\alpha = 0.3154$ for M2 and $\alpha = 0.2564$ for M3 and 9-7 filter pair. The value for M3 is very different from the others as this picture is clearly distinctive from the rest.

Using these respective values for $\alpha$ and decomposing with the optimized filters we acquire table 7.6.

| Image | Energy S4 | Energy S3 | Energy S2 | Energy S1 | Sum ($b/p$) |
|---|---|---|---|---|---|
| M1 $\alpha = 0.3214$ | 94.10% (7.10) | 0.22% (6.37) | 0.24% (5.41) | 0.40% (4.78) | 94.98% (5.01) |
| M2 $\alpha = 0.3154$ | 81.71% (6.60) | 0.61% (5.52) | 0.63% (4.83) | 0.96% (4.47) | 83.92% (4.62) |
| M3 $\alpha = 0.2564$ | 39.27% (6.90) | 4.12% (6.06) | 7.93% (5.75) | 23.88% (5.50) | 75.22% (5.60) |

Table 7.6: Decomposition, energy divisions

We notice that the cumulative sum of the energy in S1, S2 and S3 zones is lower than in tables 7.2,7.4 or 7.5, though, there is no power shrinking in the large S1 zone for the optimized filters and virtually no energy increase in the S4 subimage. These filters can not be said to be good at shifting power, but they dampen the energy with each iteration; this is quite similar to the filters in section 7.2.1. The energy and entropy content in the S4 zone is rather low, still, the performance from these filters (figure 7.18) is not better than the standard 9-7 nor does is it any enhanced than the method utilized in section 7.2.2.

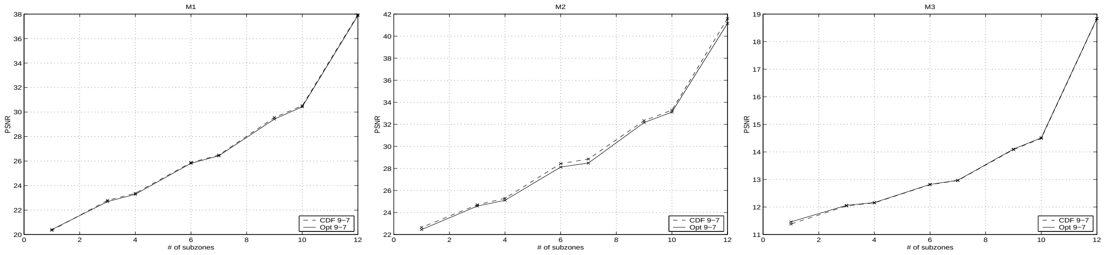

Figure 7.18: Subband truncation, L:M1 M:M2 R:M3

This suggest that maximum energy reduction for the S1,S2 and S3 zones; at the expense of power enlargement in S1 is not an appropriate method for wavelet filter design.

## 7.3 Summary

In this chapter we looked at methods to design optimized filters for iterative decomposition of images. Iterative usage of wavelets demand other features from the filters than purely the ability to shift energies, as it was shown this property alone does not perform well. By changing the optimization criteria and selecting weights relative to the total area covered by a subband, optimized filters can be designed with reasonable coding properties. Such algorithms can be used to design filters which can yield slightly higher PSNR values at the same or lower $\bar{b}/p$ coding rates compared with some traditional filters.

# Chapter 8

# Conclusions and future work

The purpose behind this study was to investigate new approaches to the wavelet filter design process, and to relate the extra degrees of freedom available in biorthogonal wavelet filter banks directly with the signal or image itself.

We started with examination of some current biorthogonal wavelet filters and found that these filters try to transfer energy from the higher frequencies to the low frequency zone. For the majority of filters this energy shift was limited and the filters only showed a small bias towards the high frequency spectrum.

Beginning with chapter 4 the attention was on construction of optimized wavelet filters for a given signal, which would move as much energy as possible to the low frequency subband. This turned out not to be a very difficult task, however, it was found that it is important to implement stability constrains in order to control the convergence of scaling functions. Stability has a negative efect on optimization, but even with stability setup the filters can manage to shift a noticeable amount of energy away from the high frequency side. For certain signals this can give lower average entropy results.

Expanding the same concepts to images we obtained similar outcomes, and for one level decomposition of natural images it is possible to transfer considerable quantity of energy away from the high-high subband and obtain lower average entropy results than many of the standard filters. Such filters alone do not perform satisfactory for iterative decomposition, as energy enlargements in the LL zones starts playing a negative role. By taking account of the iterative subbands and the number of pixels they cover, the energy shifting idea can still be employed for iterative decomposition and one can easily construct optimized filters which are able to compete with many standard accessible filters.

One of the main conclusion one can come up with is that this study provides some untraditional ways to design wavelet filter systems, based purely upon numerical optimization. The generated filters are also able to return acceptable performance and in some cases outperform the current variants. Although, there is no replacement for the theoretical framework the design process of wavelet filters for use in signal/image processing can be related with different energy shifting strategies with little emphasis given to for example maximum regularity constrains.

As it was demonstrated in section 7.2.2 a relative simple optimization procedure can return good filters for practical usage. Somewhat surprisingly, the resulting filters bear close resemblance to the wavelets filters others have proposed by various different methods. By designing optimized filters over a number of images one should therefore be in a position to attain filters with competent general performance. This subject, however, is very vast and and a whole new study can be devoted to this topic alone, where one further investigates different properties and the advantages / disadvantages of such optimized filters.

It may also be possible to slightly modify the weights in such a way which can provide optimal coding. The weights can thus be set relative to the area and to some other selected criteria. One possibility is not to minimize energy but to directly aim for entropy reductions of the decomposed image. This would require a closer look at the optimization process and the energy minimization can be used to find good starting points for entropy reduction. A more detailed look at the quantization and coding process for the filter designed in the study can be of much interest.

The optimization of one dimensional signals was not given any practical thoughts and this too is an area where more work can be done. The effects of energy shifts in for example compression of audio signals can be interesting to track.

# Appendix A

# Filter coefficients

This appendix provides filter coefficients for some of the more interesting filters generated in this study. References are made to tables in the text, and only the scaling filters are provided. The wavelet filters follow by relationships given in chapter 2.

10-6 from table 5.3:
$h = [0.0081149241839353 \quad 0.00446696552391771 \quad 0.0462180167230959 \quad -0.0845502470871813$
$0.73285712184278 \quad 0.73285712184278 \quad -0.0845502470871813 \quad 0.0462180167230959 \quad 0.00446696552391771$
$0.0081149241839353]$
$\tilde{h} = [0.0769297508395753 \quad -0.0423469815582817 \quad 0.672524011905254 \quad 0.672524011905254$
$-0.0423469815582817 \quad 0.0769297508395753]$

26-6R from table 5.3:
$h = [0.000125231293128392 \quad 6.89353006265115e-05 \quad 0.00308361322971405 \quad 1.35796101230516e-10$
$0.0139147934709254 \quad -0.0193848333909051 \quad -0.0369598090837434 \quad 0.0243907880725944$
$-0.0467582149942421 \quad 0.0595550195269377 \quad 0.0210502002777024 \quad -0.0498970160502079 \quad 0.737918073465225$
$0.737918073465225 \quad -0.0498970160502079 \quad 0.0210502002777024 \quad 0.0595550195269377 \quad -0.0467582149942421$
$0.0243907880725944 \quad -0.0369598090837434 \quad -0.0193848333909051 \quad 0.0139147934709254$
$1.35796101230516e-10 \quad 0.00308361322971405 \quad 6.89353006265115e-05 \quad 0.000125231293128392]$
$\tilde{h} = [0.0769297508395753 \quad -0.0423469815582817 \quad 0.672524011905254 \quad 0.672524011905254$
$-0.0423469815582817 \quad 0.0769297508395753]$

9-7 from table 5.5:
$h = [-0.130931417586072 \quad -0.0176481089799225 \quad -0.116864596701424 \quad 0.371201499573196$
$1.20269880976154 \quad 0.371201499573196 \quad -0.116864596701424 \quad -0.0176481089799225 \quad -0.130931417586072]$
$\tilde{h} = [0.267588909631267 \quad -0.0360680295536157 \quad 0.0859644809620065 \quad 0.779242840293779$
$0.0859644809620065 \quad -0.0360680295536157 \quad 0.267588909631267]$

17-11 from table 5.5:
$h = [0.0211997942192092 \quad -0.00553754081796288 \quad -0.0868175502898043 \quad 0.00719973460776095$
$0.134923434358389 \quad -0.178521741418566 \quad -0.125130648343527 \quad 0.530412938222042 \quad 0.818756721298013$
$0.530412938222042 \quad -0.125130648343527 \quad -0.178521741418566 \quad 0.134923434358389 \quad 0.00719973460776095$
$-0.0868175502898043 \quad -0.00553754081796288 \quad 0.0211997942192092]$
$\tilde{h} = [0.0636917449451208 \quad 0.0166367481567973 \quad -0.116787522070408 \quad 0.0159946075632493 \quad 0.406649167718561$
$0.641844069746454 \quad 0.406649167718561 \quad 0.0159946075632493 \quad -0.116787522070408 \quad 0.0166367481567973$
$0.0636917449451208]$

10-6R from table 5.6:
$h = [-0.00259813512265031 \quad 0.0788109197918546 \quad -0.209625449219418 \quad -0.195666863751791$
$1.03618630948855 \quad 1.03618630948855 \quad -0.195666863751791 \quad -0.209625449219418 \quad 0.0788109197918546$
$-0.00259813512265031]$

$\tilde{h} = [0.00602641332303701 \quad 0.182803108619674 \quad 0.518277259243837 \quad 0.518277259243837 \quad 0.182803108619674$
$0.00602641332303701]$

10-6R from table 5.7:
$h = [0.00499310536498627 \quad 0.0383956201924282 \quad -0.165614277490092 \quad -0.0556426118483388$
$0.884974944967564 \quad 0.884974944967564 \quad -0.0556426118483388 \quad -0.165614277490092 \quad 0.0383956201924282$
$0.00499310536498627]$
$\tilde{h} = [-0.0203431802697728 \quad 0.156433515026864 \quad 0.571016446429456 \quad 0.571016446429456 \quad 0.156433515026864$
$-0.0203431802697728]$

9-7, M1 from table 6.5:
$h = [0.238657181089884 \quad -0.0841514752230178 \quad -0.019653556112844 \quad 0.437704865816292$
$0.269099531232468 \quad 0.437704865816292 \quad -0.019653556112844 \quad -0.0841514752230178 \quad 0.238657181089884]$
$\tilde{h} = [-0.401023255740114 \quad -0.141402401617065 \quad 0.754576646333388 \quad 0.989911584420678$
$0.754576646333388 \quad -0.141402401617065 \quad -0.401023255740114]$

9-7, M2 from table 6.5:
$h = [0.262560103355673 \quad -0.100325191619466 \quad -0.00688022363096841 \quad 0.45387858221274$
$0.195747021737139 \quad 0.45387858221274 \quad -0.00688022363096841 \quad -0.100325191619466 \quad 0.262560103355673]$
$\tilde{h} = [-0.427977377417235 \quad -0.163531747014957 \quad 0.781530768010508 \quad 1.03417027521646$
$0.781530768010508 \quad -0.163531747014957 \quad -0.427977377417235]$

9-7, M2 from table 6.5:
$h = [0.07479242225809 \quad -0.0407958429382374 \quad -0.0790014839556439 \quad 0.394349233531511$
$0.715524904581655 \quad 0.394349233531511 \quad -0.0790014839556439 \quad -0.0407958429382374 \quad 0.07479242225809]$
$\tilde{h} = [-0.123888778291883 \quad -0.0675756579131128 \quad 0.477442168885156 \quad 0.842258097012773$
$0.477442168885156 \quad -0.0675756579131128 \quad -0.123888778291883]$

9-7, M1 from table 6.6:
$h = [0.0261056339807951 \quad -0.0452366592097134 \quad -0.270565469529881 \quad 0.398790049802987$
$1.19602645228472 \quad 0.398790049802987 \quad -0.270565469529881 \quad -0.0452366592097134 \quad 0.0261056339807951]$
$\tilde{h} = [0.0213884718297301 \quad 0.037062613070031 \quad 0.332164918763544 \quad 0.632981555046486$
$0.332164918763544 \quad 0.037062613070031 \quad 0.0213884718297301]$

9-7, M2 from table 6.6:
$h = [0.0133877323692417 \quad 0.0275274083367311 \quad -0.286526327836927 \quad 0.326025982256543$
$1.25338397212192 \quad 0.326025982256543 \quad -0.286526327836927 \quad 0.0275274083367311 \quad 0.0133877323692417]$
$\tilde{h} = [-0.0203659488344288 \quad 0.041875784058719 \quad 0.373919339427703 \quad 0.62335521306911$
$0.373919339427703 \quad 0.041875784058719 \quad -0.0203659488344288]$

9-7, M3 from table 6.6:
$h = [-0.0241301567960936 \quad -0.000675942236116889 \quad -0.174092887734909 \quad 0.354229332829391$
$1.10355287024855 \quad 0.354229332829391 \quad -0.174092887734909 \quad -0.000675942236116889$
$-0.0241301567960936]$
$\tilde{h} = [0.0482704360605875 \quad -0.00135216802629404 \quad 0.305282954532686 \quad 0.709811117239136$
$0.305282954532686 \quad -0.00135216802629404 \quad 0.0482704360605875]$

9-7, M4 from table 6.12:
$h = [-0.045781118855036 \quad -0.000339301092777133 \quad -0.175424642942453 \quad 0.353892691686051$
$1.14951830478152 \quad 0.353892691686051 \quad -0.175424642942453 \quad -0.000339301092777133$
$-0.045781118855036]$
$\tilde{h} = [0.0915648232999242 \quad -0.000678621348333776 \quad 0.26198856729335 \quad 0.708464023883215$
$0.26198856729335 \quad -0.000678621348333776 \quad 0.0915648232999242]$

9-7, M4 from table 6.13:
$h = [0.000859175631050722 \quad -0.0118649074894742 \quad -0.0777909575050763 \quad 0.365418298082748$
$0.860970344934599 \quad 0.365418298082748 \quad -0.0777909575050763 \quad -0.0118649074894742 \quad 0.000859175631050722]$
$\tilde{h} = [-0.00497750851024253 \quad -0.0687376083163294 \quad 0.358530899103516 \quad 0.844581997819206$
$0.358530899103516 \quad -0.0687376083163294 \quad -0.00497750851024253]$

9-7 from table 6.14:
$h = [0.011337006969397 \quad 0.0213998899846802 \quad -0.262863566612625 \quad 0.332153500608594 \quad 1.210159900473$

$0.332153500608594 \quad -0.262863566612625 \quad 0.0213998899846802 \quad 0.011337006969397]$
$\tilde{h} = [-0.0183379738992656 \quad 0.0346150112675717 \quad 0.371891364492539 \quad 0.637876758651404$
$0.371891364492539 \quad 0.0346150112675717 \quad -0.0183379738992656]$

9-7, M1 from table 7.4:
$h = [0.0637759701342507 \quad -0.0235541748351232 \quad -0.107271542870431 \quad 0.377107565428397$
$0.794097926658909 \quad 0.377107565428397 \quad -0.107271542870431 \quad -0.0235541748351232 \quad 0.0637759701342507]$
$\tilde{h} = [-0.117122164631056 \quad -0.043256354030849 \quad 0.47067555522433 \quad 0.793619489248246$
$0.47067555522433 \quad -0.043256354030849 \quad -0.117122164631056]$

9-7, M2 from table 7.4:
$h = [0.0730292822756296 \quad -0.0282417930404378 \quad -0.0976056823824546 \quad 0.381795183633712$
$0.756259581400197 \quad 0.381795183633712 \quad -0.0976056823824546 \quad -0.0282417930404378 \quad 0.0730292822756296]$
$\tilde{h} = [-0.131898471665481 \quad -0.0510076126048637 \quad 0.485451862258754 \quad 0.809122006396275$
$0.485451862258754 \quad -0.0510076126048637 \quad -0.131898471665481]$

9-7, M3 from table 7.4:
$h = [0.0010217775871718 \quad -0.00987782662572342 \quad -0.121958126996232 \quad 0.363431217218997$
$0.948979480004667 \quad 0.363431217218997 \quad -0.121958126996232 \quad -0.00987782662572342 \quad 0.0010217775871718]$
$\tilde{h} = [-0.0033555353867028 \quad -0.0324389546242395 \quad 0.356908925979977 \quad 0.771984690435026$
$0.356908925979977 \quad -0.0324389546242395 \quad -0.0033555353867028]$

17-11, M1 from table 7.5:
$h = [0.00323239109327755 \quad 0.00882230121114521 \quad -0.0409170204115113 \quad 0.0301927523836962$
$0.110712233819226 \quad -0.0714068111642326 \quad -0.0985340281841358 \quad 0.385945148162665 \quad 0.758119628552834$
$0.385945148162665 \quad -0.0985340281841358 \quad -0.0714068111642326 \quad 0.110712233819226 \quad 0.0301927523836962$
$-0.0409170204115113 \quad 0.00882230121114521 \quad 0.00323239109327755]$
$\tilde{h} = [0.00925570944187871 \quad -0.0252619977789099 \quad -0.141193887964795 \quad -0.0208656722917339 \quad 0.48549156911619$
$0.799362121327835 \quad 0.48549156911619 \quad -0.0208656722917339 \quad -0.141193887964795 \quad -0.0252619977789099$
$0.00925570944187871]$

17-11, M2 from table 7.5:
$h = [0.00369138916002243 \quad 0.0194110401306916 \quad -0.0594515142386153 \quad 0.0464937805788446$
$0.123015123831897 \quad -0.0705189357723379 \quad -0.0842241676884034 \quad 0.358167505656076 \quad 0.741045119056745$
$0.358167505656076 \quad -0.0842241676884034 \quad -0.0705189357723379 \quad 0.123015123831897 \quad 0.0464937805788446$
$-0.0594515142386153 \quad 0.0194110401306916 \quad 0.00369138916002243]$
$\tilde{h} = [0.00899686455733635 \quad -0.0473096959985112 \quad -0.162975616429307 \quad -0.0182600853366955$
$0.507532142465245 \quad 0.838246343856961 \quad 0.507532142465245 \quad -0.0182600853366955 \quad -0.162975616429307$
$-0.0473096959985112 \quad 0.00899686455733635]$

9-7, M1 from table 7.6:
$h = [0.133955211970994 \quad -0.0575809071840287 \quad -0.0519266802464524 \quad 0.411134297777302$
$0.543049717737464 \quad 0.411134297777302 \quad -0.0519266802464524 \quad -0.0575809071840287 \quad 0.133955211970994]$
$\tilde{h} = [-0.224502999757239 \quad -0.0965030490516252 \quad 0.578056390350513 \quad 0.900112879289798$
$0.578056390350513 \quad -0.0965030490516252 \quad -0.224502999757239]$

9-7, M2 from table 7.6:
$h = [0.153871529466597 \quad -0.0739640388239883 \quad -0.0352283673932729 \quad 0.427517429417262$
$0.469820457039899 \quad 0.427517429417262 \quad -0.0352283673932729 \quad -0.0739640388239883 \quad 0.153871529466597]$
$\tilde{h} = [-0.245539391385284 \quad -0.118027585351208 \quad 0.599092781978558 \quad 0.943161951888963$
$0.599092781978558 \quad -0.118027585351208 \quad -0.245539391385284]$

9-7, M3 from table 7.6:
$h = [0.0609662870539563 \quad -0.0342647337280331 \quad -0.0900037953332786 \quad 0.387818124321307$
$0.765181797745192 \quad 0.387818124321307 \quad -0.0900037953332786 \quad -0.0342647337280331 \quad 0.0609662870539563]$
$\tilde{h} = [-0.102305136964679 \quad -0.0574983067937579 \quad 0.455858527557953 \quad 0.822103394774063$
$0.455858527557953 \quad -0.0574983067937579 \quad -0.102305136964679]$

# References

[1] V. Bekkedal. *Compression of MR-images using wavelets.* (Original title in Norwegian: *Kompresjon av MR-bilder ved hjelp av wavelets.*) Master thesis; Department of Informatics, University of Oslo, 1997.

[2] D. P. Bertsekas. *Nonlinear programming.* Athena Scientific, Belmont, Massachusetts, 1995.

[3] C. S. Burrus, R. A. Gopinath and H. Guo. *Introduction to wavelets and wavelets transforms.* Precntice Hall, New Jersey, 1998.

[4] A. Cohen, I. Daubechies and J.C.Feauveau. "Biorthogonal bases of compactly supported wavelets." *Communications on Pure and Applied Mathematics,* 45(5):485-500, June 1992.

[5] A. Cohen and Robert D. Ryan. *Wavelets and multiscale signal processing.* Chapman & Hall, London, 1995.

[6] I. Daubechies. *Ten lectures on Wavelets.* Society for Industrial and Applied Mathematics, Philadelphia, 1992.

[7] T. D. Derose, D. H. Salesin and E. J. Stollnitz. *Wavelets for computer graphics: theory and applications.* Morgan Kaufmann Publishers, San Francisco, 1996.

[8] O. Egger and W. Li. "Subband coding of images using asymmetrical filter banks." *IEEE Transactions on Image Processing,* 4(4):478-485, April 1995.

[9] R. C. Gonzalez and R. E. Woods. *Digital Image Processing.* Addison-Wesley Publishing Company, Inc. Reading, Massachusetts, 1993.

[10] JPEG 2000 Image Coding System, JPEG 2000 Committee draft version 1.0, 9. December 1999. http://www.jpeg.org.

[11] D. M. Monro and B. G. Sherlock. "Space-frequency balance in biorthogonal wavelets." *IEEE International Conference on Image Processing, 1997, Vol.1: 624-627.*

[12] J. E. Odegard and C. S. Burrus. "Smooth biorthogonal wavelets for applications in image compression." *Proceedings of IEEE Digital Signal Processing Workshop*, Loen, Norway, 1996

[13] A. V. Oppenheim and R.W. Schafer. *Discrete-Time Signal Processing.* Prentice-Hall International, New Jersey, 1989.

[14] A. Papoulis. *Probability, random variables and stochastic processes, 3rd ed.* McGraw-Hill, New York, 1991.

[15] G. Strang and T. Nguyen. *Wavelets and filter banks.* Wellesley-Cambridge Press, Wellesley, 1996.

[16] J. D. Villasenor, B. Belzer and J. Liao. "Wavelet filter evaluation for image compression." *IEEE Transactions on Image Processing,* vol. 4, no.8, August 1995.

[17] L. W. Winger and A. N. Venetsanopoulos. "Biorthogonal Modified Coiflet Filters for Image Compression" *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing.* Seattle, 1998.
(CR-DOM: http://www.causalproductions.com/TEMP/INDEX/IC98_TOC.HTM).

[18] A. Zandi, J. D. Allen, E. L. Schwartz and M. Boliek. *Compression with reversible embedded wavelets*. http://www.crc.ricoh.com/CREW ,1996.