CHAPTER-1

INTRODUCTION

1.1 INTRODUCTION TO WAVELET

The wavelet transform has been perhaps the most exciting development in the last decade to bring together , researchers in several different fields such as signal processing , image processing ,communications ,computer sciences and mathematics etc., It is well known form Fourier theory that a signal can be expressed as the sum of a , possibly infinite, series of sine's and cosines. This sum is also referred to as a Fourier expansion. The big disadvantage of a Fourier expansion however is that , it has only frequency resolution and no time resolution. This means that although we might be able to determine all the frequencies present in a signal, we do not know when they are present. To overcome this problem in the past decades several solutions have been developed which are more or less able to represent a signal in the time and frequency domain at the same time.

The idea behind these time-frequency joint representations is to cut the signal of interest into several parts and then analyze the parts separately. It is clear that analyzing a signal this way will give more information about the when and where of different frequency components, but it leads to a fundamental problem as well: how to cut the signal? Suppose that we want to know exactly all the frequency components present at a certain moment in time, we cut out only this very short time window using dirac pulse transform it to the frequency domain and something is very wrong.

The problem here is that cutting the signal corresponds to a convolution between the signal and the cutting window. Since convolution in the time domain is identical to multiplication in the frequency domain and since the Fourier transform of a dirac pulse contains all possible frequencies the frequency components of the signal will be smeared out all over the frequency axis. In fact this situation is the opposite of the standard Fourier transform since we now have time resolution but no frequency resolution whatsoever.

The underlying principle of the phenomenon just described is Heisenberg's uncertainty principle, which in signal processing terms, states that it is impossible to know the exact frequency and the exact time of occurrences of this frequency in a signal. In other words a signal can simply not be represented as a point in the time-frequency space. The uncertainty principle shows that it is very important now one cuts the signal.

The wavelet transform or wavelet analysis is probably the most recent solution to overcome the shortcomings of the Fourier transform. In wavelet analysis the use of a fully scalable modulated window solves the signal-cutting problem. The window is shifted along the signal and for every position the spectrum is calculated. Then this process is repeated many times with slightly shorter (or longer) window for every new cycle. In the end the result will be a collection of time and frequency representations of the signal, all with different resolutions. Because of this collection of representations we can speak multi-resolution analysis. In this case of wavelets we normally speaks about time-frequency representations but about time-scale representations, scale being in a way the opposite of frequency , because the term frequency is reserved for the Fourier transform.

1.2 WAVELET DEFINATION:-

A wavelet is a small wave which has its energy concentrated in time. It has oscillating wave like characteristic but also has the ability to allow simultaneous time and frequency analysis and it is a suitable tool for transient, non-stationary or time-varying phenomena.

Fig

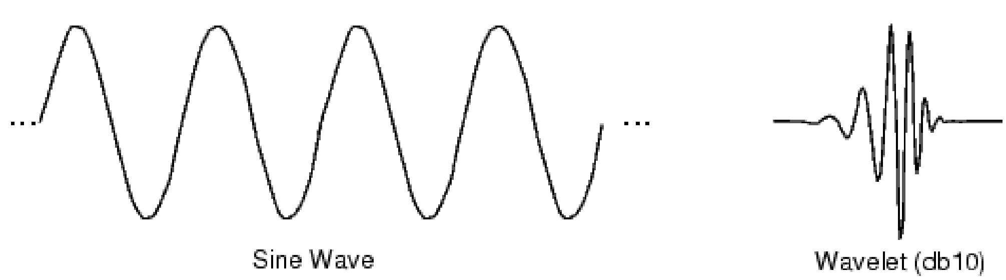


Fig 1 wavelet defination

1.3 WAVELET CHARACTERISTICS:-

The difference between wave and wavelet is shown in the above fig. waves are smooth, predictable and everlasting, whereas wavelets are limited duration, irregular and may be asymmetric. Waves are used as deterministic basis functions in Fourier analysis for the expansion of functions (signals), which are time-invariant, or stationary. The important characteristic of wavelets is that they can serve as deterministic or non-deterministic basis for generation and analysis of the most natural signals to provide better time-frequency representation, which is not possible with waves using conventional Fourier analysis.

1.4 WAVELET ANALYSIS:

The wavelet analysis procedure is to adopt a wavelet prototype function, called an 'analyzing wavelet' or 'mother wavelet' .Temporal analysis is performed with a contracted , high frequency version of the prototype wavelet , while frequency analysis is performed with a dilated, low frequency version of the same wavelet. Mathematical formulation of signal expansion using wavelets gives wavelet transform (WT) pair, which is an analogues to the Fourier transform (FT) pair. Discrete-time and discrete-parameter version of WT is termed as discrete wavelet transform (DWT) . DWT can be viewed in a similar framework of discrete Fourier transform (DFT) . DWT can be viewed in a similar framework of discrete Fourier transform (DFT) with its efficient implementation through fast filter bank algorithms similar to fast Fourier transform algorithms.

1.5 EVOLUTION OF WAVELET TRANSFORM :

The need of simultaneous representation and localization of both time and frequency for non-stationary signals (e.g. music, speech , images) led toward the evolution of wavelet transform form the popular Fourier transform. Different 'time-frequency representations' (TFR) are very informative in understanding and modelling of wavelet transform.

1.6 TRANSFORMS:-

First of all, why do we need transform? Mathematical transformations are applied to signals to obtain further information from that signal that is not readily available in the raw signal. Most of the signals in practice are time-domain signals in their raw format. Time domain representation is not always the best representation of the signal for most signal processing related applications. In many cases, the most distinguished information is hidden in the frequency content of the signal. The information that cannot be readily seen in the time-domain can be seen in the frequency domain.

Fourier Transform (FT) with its fast algorithms (FFT) is an important tool for analysis and processing of many natural signals. FT has certain limitations to characterize many natural signals, which are non-stationary (e.g. speech). Though a time varying , overlapping window based FT namely STFT (short time Fourier transform) is well known for speech processing applications ,a new time-scale based Wavelet Transform (WT) is a powerful mathematical tool for non-stationary signals.

Wavelet Transform uses a set of damped oscillating functions known as wavelet basis. Wavelet Transform in its continuous (analog) form is represented as CoWT (continuous wavelet transforms). Continuous wavelet transform with various deterministic or non-deterministic basis is a more effective representation of signals for analysis as well as characterization. Continuous wavelet transform is powerful in singularity detection. A discrete and fast implementation of continuous wavelet transform, (generally with real valued basis) is known as the standard DWT (discrete wavelet transforms).

With standard DWT (Discrete wavelet transform, signal has a same data size in transform domain and therefore it is a non-reluctant transform, Standard DWT can be implemented through a simple filter bank structure of recursive FIR filters. A very important property ; multi-resolution analysis (MRA) allows DWT to view and process different signals at various resolution levels .The advantages such as non-redundancy, fast and simple implementation with digital filters using micro-computers, and MRA capability popularized the DWT in many signal processing applications since last decade. Many researches have successfully applied and proved the advantages of DWT for signal de-noising and compression in a number of diverse fields.

1.6.1 FOURIER TRANSFORM

Fourier transform is used to find the frequency content of a signal. It allows going back and forwarding between the raw and processed (transformed) signals. However, only either of them is available at any given time. That is, no frequency information is available in the time-domain signal, and no time information is available in the Fourier transformed signal. Fourier transform of a time domain signal X(t) and inverse Fourier transform (IFT) of a frequency domain signal X(f) are given below.

$$X(f) = \int_{-\infty}^{\infty} x(t) e^{-2j\pi ft} dt \quad \text{...(1.6.1)}$$

$$x(t) = \int_{-\infty}^{\infty} X(f) e^{2j\pi ft} dt \quad \text{...(1.6.2)}$$

Though FT has a great ability to capture signal’s frequency content as long as X(t) is composed of few stationary components (e.g. sine waves) .However , any abrupt change in time for non-stationary signal X(t) is spread out over the whole frequency axis in X(f). Hence the time-domain signal sampled with dirac-delta function is highly localized in time but spills over entire frequency band and vice versa. The limitation of FT is that it cannot offer both time and frequency localization of a signal at the same time. To overcome the limitations of the standard FT, Gabor introduced the initial concept of short time Fourier transform(STFT).

1.6.2 SHORT TERM FOURIER ANALYSIS:

This is the revised version of the Fourier transform. There is only a minor difference between short term Fourier analysis (STFT) and FT. In STFT, the signal is divided into small enough segments, where these segments (portions) of the signal can be assumed to be stationary. For this purpose, a window function "w" is chosen. The width of this window must be equal to the segment of the signal where its stationarity is valid.

This window function is the first located to the very beginning of the signal. That is, the window function is located at t=0. Let's suppose that the width of the windows is "T" s. At this time instant (t=0), the window function will overlap with the first T/2 seconds. The window functions and the signal are then multiplied. By doing this , only the first T/2 seconds of the signal is being chosen, with the appropriate weighting of the window (if the window is a rectangle, with amplitude "I" , then the product will be equal to the signal) Assuming the product just as another signal, FT is taken. The result of this transformation is the FT of the first T/2 seconds of the signal. If this Portion of the signal is stationary, as it is assumed, the obtained result will be as true frequency representation of the first T/2 seconds of the signal. The next step would be shifting this window (for some t1 seconds) to a new location, multiplying with the signal and taking the FT of the product. This procedure is followed until the end of the signal is reached by shifting the window with "t1" seconds intervals. The following definition of the STFT summarizes all the above explanations in one line:

$$STFT_X^{\omega}(t,f)=\int_t [x(t).\omega^*(t-t^1)].e^{-j2\pi ft}dt \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(1.6.2.1)$$

In the above equation X(t) is the signal , w(t) is the window function, and * is the complex conjugate. As you can see from the equation, the STFT of the signal is nothing but the FT of the signal multiplied by a window function. Using STFT one cannot know the exact time-frequency representation of a signal, i.e., one cannot know what spectral components exist at what instances of times. What one can know are the time intervals in which certain band of frequencies exists, which is a resolution problem. This problem occurs because of width of window function used.

Narrow window ➔ good time resolution, poor frequency resolution

Wide window ➔ good frequency resolution, poor time resolution and violates the condition of stationary.

The selection of proper window is application dependent. Once a window has been chosen for STFT, the time-frequency resolution is fixed over the entire time-frequency plane because the same window is used at all frequencies. There is always a trade off between time resolution and frequency resolution in STFT.

1.6.3 CONTINUOUS WAVELET TRANSFORM

The continuous wavelet transform was developed as alternative approach to the short time Fourier transforms to overcome the resolution problem. The wavelet analysis is done in a similar way to the STFT analysis, in the sense that the signal is multiplied with a function (i.e. the wavelet) , similar to the window function in the STFT, and the transform is computed separately for different segments of the time-domain signal, however, there are two main differences between the STFT and the CWT.

1. The Fourier transforms of the windowed signals are not taken, and therefore are not computed.
2. The width of the window is changed as the transform is computed for every single spectral component, which is probably the most significant characteristic of the wavelet transform.

The wavelet transform (WT) in its continuous (CWT) form provides a flexible time-frequency window, which narrows when observing high frequency phenomena and widens when analysing low frequency behaviour. Thus time resolution becomes arbitrarily good at low frequencies. This kind of analysis is suitable for signals composed of high frequency components with short duration and low frequency components with long duration, which is often the case in practical situation.

The continuous wavelet transform is defined as follows

$$CWT_x^{\varphi}(\tau,s)=\psi_x^{\varphi}(\tau,x)=\frac{1}{\sqrt{s}}\int x(t)\varphi^{*}(\frac{t-\tau}{s})dt \quad \text{…………………(1.6.3.1)}$$

As seen in the above equation, the transformed signal is a function of two variables, τ and s , the translation and scale parameters, respectively. Ψ(t) is the transforming function, and it is called the mother wavelet.

The mother wavelet is a prototype for generating the other window functions. The term translation is related to the location of window, as the window is shifted through the signal. This term corresponds to the time information in transform. The scale parameter is defined as the inverse of frequency. High scales (low frequencies) correspond to global information of a signal (that usually spans the entire signal) whereas low scales (high frequencies) do not last for entire duration of signal but usually appear from time to time as short bursts and high scales (low frequencies) usually last for the entire duration of the signal.
The CWT is the correlation between a wavelet at different scales and the signal with the scale (or the frequency) being used as a measure of similarity. The continuous wavelet transform was computed by changing the scale of the analysis window, shifting the window in time, multiplying by the signal , and integrating over all times.

1.6.4 DESCRETE WAVELET TRANSFORM

The CWT has the drawbacks of redundancy and impracticability with digital computers. The discrete wavelet transform (DWT) provides sufficient information both for analysis and synthesis of the original signal, with a significant reduction in the computation time. The DWT is considerably easier to implement when compared to the CWT.
The DWT analyzes the signal at different frequency bands with different resolutions by decomposing the signal into a coarse approximation and detail information. DWT employs two sets of functions, called scaling functions and wavelet functions, which

are associated with low pass and high pass filters, respectively. The original signal x[n] is first passed through a half-band high pass filter g[n] and a low pass filter h[n]. After the filtering, half of the samples can be eliminated according to the nyquist's rule. The signal can therefore be sub sampled by 2 , simply by discarding every other sample. This constitutes one level of decomposition and can mathematically be expressed as follows:

$$y_{high}[n] = \sum x[k] \cdot g[2n - k] \qquad \text{……………………………..(1.6.4.1)}$$

$$y_{low}[n] = \sum x[k] \cdot h[2n - k] \qquad \text{……………………………..(1.6.4.2)}$$

$y_{high}[k]$ and $y_{low}[k]$ are the outputs of the high pass and low pass filters, respectively after sub sampling by 2. This decomposition halves the time resolution since only half the number of samples now characterises the entire signal. However, this operation doubles the frequency resolution, since the frequency band of the signal now spans only half the previous frequency band, effectively reducing the uncertainty in the frequency by half. The above procedure, which is also known as the sub-band coding can be repeated for further decomposition. At every level, the filtering and sub sampling will result in half the number of samples (and hence half the time resolution) and half the frequency band spanned (and hence half the frequency resolution). Hence the fig. illustrates this procedure, where x[n] is the original signal to be decomposed, and h[n] and g[n] are low pass and high pass filters, respectively. The bandwidth of the signal at every level is marked on the figure as " f ".

The frequencies that are most prominent in the original signal will appear as high amplitudes in that region of the DWT signal that includes those particular frequencies. The frequency bands that are not very prominent in the original signal will have very low amplitudes , and that part of the DWT signal can be discarded without any major loss of information , allowing data reduction. The difference of this transform from the Fourier transform is that the time localization of these frequencies will not be lost.

Fig

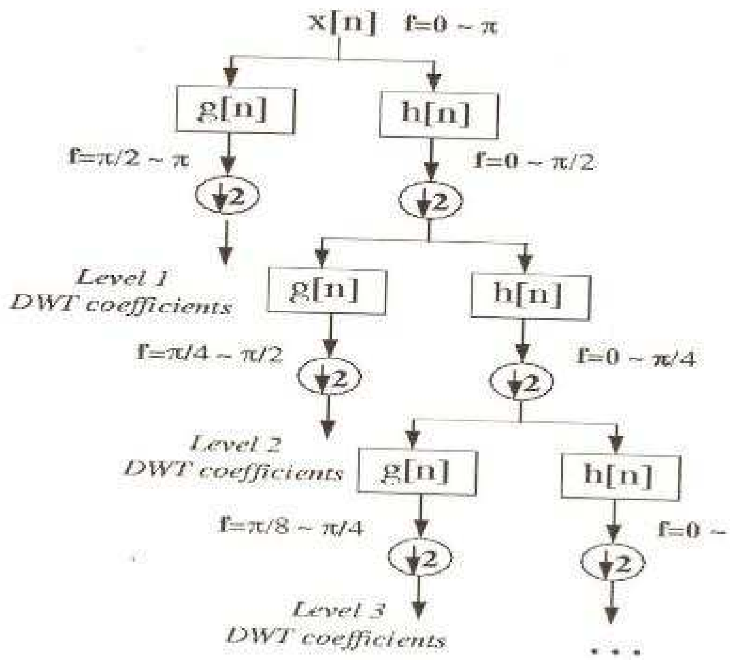


Fig 2 DWT coefficients at different levels

Four resulting sets of wavelet coefficients W_{LL}, W_{HL}, W_{LH} and W_{HH} are conventionally named according to the filtering types along rows and columns respectively (H: high pass filtering, L: for low pass filtering). These sets are also called wavelet sub bands (LL,LH,HL,HH). The perfect reconstruction is also obtained by applying the ID synthesis scheme on rows and columns successively.

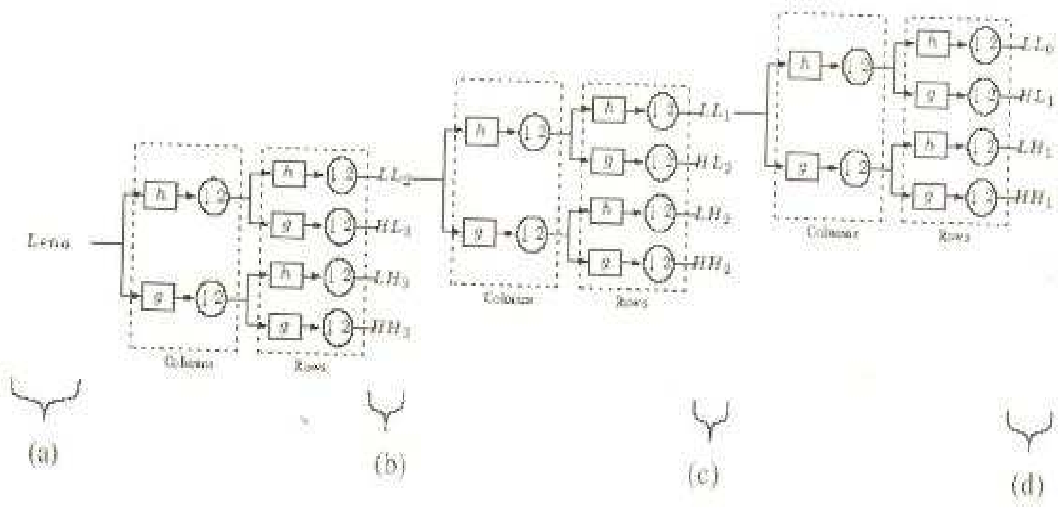


Fig 3 reconstruction using subbands

It is worth pointing out that the order in which rows and columns are processed at the analysis and synthesis sides has no importance since the global transformation is linear.

An advantage of wavelet transform is that the windows vary. In order to isolate signal discontinuities , one would like to have some very short basis functions. At the same time, in order to obtain detailed frequency analysis , one would like to have some very long basis functions. A way to achieve this is to have short high-frequency basis functions and long low-frequency ones. This happy medium is exactly what you get with wavelet transforms. One thing to remember is that wavelet transforms do not have a single set of basis functions like the Fourier transform, which utilizes just the sine and cosine functions. Instead , wavelet transforms have an infinite set of possible basis functions. Thus wavelet analysis provides immediate access to information that can be obscured by other time-frequency methods such as Fourier analysis.

1.7 COMPARATIVE VISUALIZATION

A comprehensive visualization of various time-frequency representation, shown in figure, demonstrates the time-frequency resolution for a given signal in various transform domains with their corresponding basis functions.

Figure describes the time and frequency responses in different domains

Here x- axis has time and y-axis has frequency

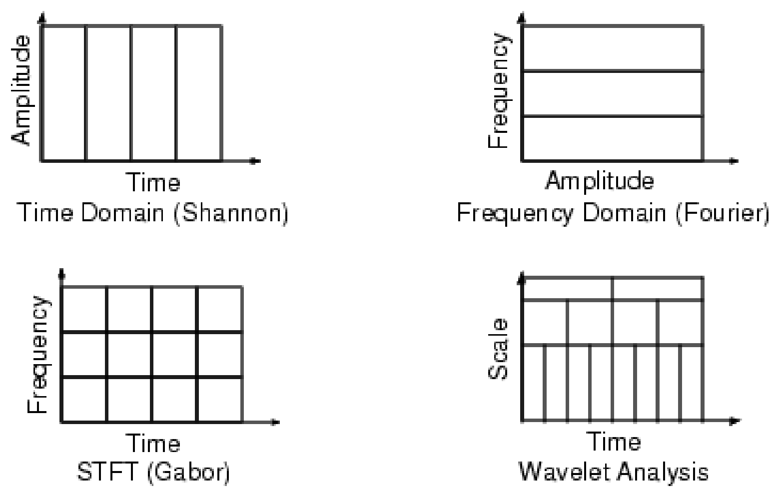


Fig 4 comparative visualization

1.8 WAVELET PROPERTIES

The most important properties of wavelets are the admissibility and the regularity conditions and these are the properties which gave wavelets their name. It can be shown that square integrable functions $\psi(t)$ satisfies the admissibility condition

$$\int \frac{|\psi(\omega)|^2}{|\omega|} d\omega \quad □$$

$+\infty$ ……………………………...……………(1.8.1)

can be used to first analyze and then reconstruct a signal without loss of information. $\psi(\omega)$ stands for the Fourier transform of $\psi(t)$. The admissibility condition implies that the Fourier transform of $\psi(t)$ vanishes at the zero frequency.

i.e.,

$$|\psi(\omega)|^2 \, |_{\omega = 0} = 0 \quad \text{……………………………………..(1.8.2)}$$

This means that wavelets must have a band-pass like spectrum. This is a very important observation, which we will use later on to build an efficient wavelet transform.

A zero at the zero frequency also means that the average value of the wavelet in the time domain must be zero.

$$\int \psi(t) dt = 0 \quad \text{………………………………………………………..(1.8.3)}$$

And therefore it must be oscillatory. In other words $\psi(t)$ must be a wave. As from the above knowledge the wavelet transform of one dimensional function is two dimensional ; the wavelet transform of two-dimensional function is four-dimensional. The time-bandwidth product of the wavelet transform is the square of the input signal and for most practical applications this is not a desirable property. Therefore one imposes some additional conditions on the wavelet functions in order to make the wavelet transform decrease quickly with decreasing scale s. These are the regularity conditions and they state that the wavelet function should have some smoothness and concentration in both time and frequency domains.

If we expand the wavelet transform into the taylor series at t=0 until order n (let $\tau=0$ for simplicity) we get

$$\gamma(s,0) = \frac{1}{\sqrt{s}}[\sum_{p=0}^{n} f^{(p)}(0)\int \frac{t^p}{\underline{|p}}\psi(\frac{t}{s})dt + o(n+1)] \quad \text{..................(1.8.4)}$$

Hence $f^{(p)}$ stands for the p^{th} derivative of f and O(n+1) means the rest of the expansion. Now, if we define the moments of the wavelet by M_p ,

$$M_p = \int t^p \psi(t)\,dt, \quad \text{...(1.8.5)}$$

Then we can get the finite development

$$\gamma(s,0) = \frac{1}{\sqrt{s}}\left[f(0)M_0 s + \frac{f^{(1)}(0)}{\underline{|1}} M_1 s^2 + \frac{f^{(2)}(0)}{\underline{|2}} M_2 s^3 + ... + \frac{f^{(n)}(0)}{\underline{|n}} M_n s^{n+1} + 0(s^{n+2}) \right] \quad \text{.......(1.8.6)}$$

From the admissibility condition we already have that the 0^{th} moment M_0= 0 so that the first term in the right-hand side of above equation is zero. If we now manage to make the other moments up to M_n zero as well, then the wavelet transform coefficients $\gamma(s,\tau)$ will decay as fast as s^{n+2} for a smooth signal f(t). This is known in literature as the vanishing moments or approximation order. If a wavelet has N vanishing moments, then the approximation order of the wavelet transform is also N. The moments do not have to be exactly zero , a small value is often good enough. In fact experimental research suggests that the number of vanishing moments required depends heavily on the applications.

The admissibility condition gave us the wave, regularity and vanishing moments gave us the fast decay or the let, and put together they give us the wavelet.

1.9 A BAND-PASS FILTER

With the redundancy removed, we still have two hurdles to take before we have the wavelet transform in a practical form. We continue by trying to reduce the number of wavelets needed in the wavelet transform and save the problem of the difficult analytical solutions for the end.

Even with discrete wavelets we still needed an infinite number of scalings and translations to calculate the wavelet transform. The easiest way to tackle this problem is simply not to use an infinite number of discrete wavelets. Of course this poses the question of the quality of the transform. Is it possible to reduce the number of wavelets to analyze a signal and still have a useful result .

The translation of the wavelets are of course limited by the duration of the signal under investigation so that we have an upper boundary for the wavelets. This leaves us with the question of dilation how many scales do we need to analyze our signal? How do we get the lower bond ? it turns out that we can answer this question by looking at the wavelet transform in a different way.

The wavelet has a band-pass like spectrum. From Fourier theory we know that compression in time is equivalent to stretching the spectrum and shifting it upwards

$$F\left\{f(at)\right\} = \frac{1}{|a|} F\left(\frac{\omega}{a}\right) \quad \text{...(1.9.1)}$$

This means that a time compression of the wavelet by a factor of 2 will stretch the frequency spectrum of the wavelet by a factor of 2 and also shift all frequency components up by a factor of 2. Using this insight we can cover the finite spectrum of our signal with the spectrum of dilated wavelets in the same way as that we covered our signal in the time domain with translated wavelets. To get a good coverage of the signal spectrum the stretched wavelet spectra should touch each other, as if they were standing hand in had. This can be arranged by correctly designing the wavelets.

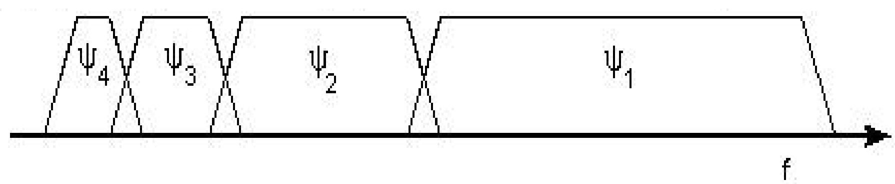


Fig 5 band pass filter

If one wavelet can be seen as a band-pass filter, then a series of dilated wavelets can be seen as a band-pass filter bank. If we look at the ratio between the center frequency of a wavelet spectrum and the width of this spectrum we will see that it is the same for all wavelets. This ratio is normally referred to as the fidelity factor Q of a filter and in the case of wavelets one speaks therefore of a constant-Q filter bank.

1.10 A CONSTRAINT

As a constraint we will now take a look at an important constraint on our signal, which has sneaked in during the last section the signal to analyze must have finite energy. when the signal has infinite energy it will be impossible to cover its frequency spectrum and its time duration with wavelets. Usually this constraint is formally stated as

$$\int \left| f(t)^2 \right| dt < \infty \quad \text{...(1.10.1)}$$

And it is equivalent to stating that the L^2 norm of our signal f(t) should be finite. This is where Hilbert spaces come in so we end our constraint by stating that natural signals normally have finite energy.

1.11 THE SCALING FUNCTION

The question arises how to cover the spectrum all the way down to zero ? Because every time we stretch the wavelet in time domain with a factor of 2, its bandwidth is halved. In other words, with every wavelet stretch we cover only half of the remaining spectrum, which means that we will need an infinite number of wavelets to get the job done.

The solution of this problem is simply not to try to cover the spectrum all the way down to zero with wavelet spectra, but to use a cork to plug the hole when it is small enough. This cork then is a low-pass spectrum and it belongs to the so-called scaling function. The scaling function was introduced by mallat . because of the low-pass nature of he scaling function spectrum it is sometimes referred to as the averaging filter.

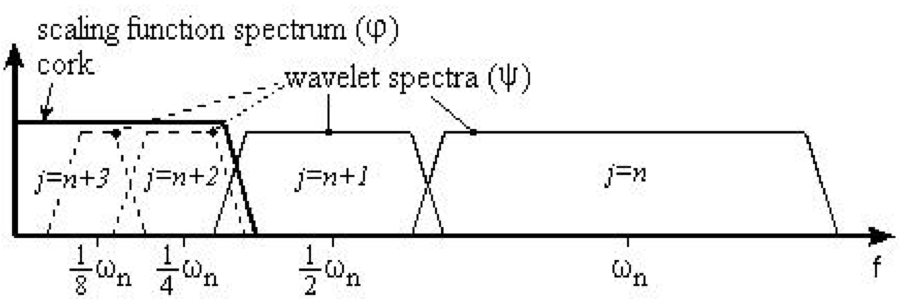


Figure 6 shows scaling function of wavelet.

If we look at the scaling function as being just a signal with a low-pass spectrum, then we can decompose it in wavelet components and express it as

$$\varphi(t) = \sum_{j,k} \gamma(j,k)\psi_{j,k}(t) \qquad \text{……………………………………….(1.11.1)}$$

Since we selected the scaling function $\varphi(t)$ in such a way that its spectrum neatly fitted in the space left open by the wavelets, the expression uses an infinite number of wavelets up to certain scale j. This means that if we analyze a signal using the combination of scaling function and wavelets, the scaling function by itself takes care of the spectrum otherwise covered by all the wavelets up to scale j, while the rest is done by the wavelets. In this way we have limited the number of wavelets form an infinite number to a finite number.

By introducing the scaling function we have circumvented the problem of the infinite number of wavelets and set a lower bound for the wavelet. Of course when we use a scaling function instead of wavelets we lose information. That is to say, from a signal representation view we do not loose any information, since it will still be possible to reconstruct the original signal but from a wavelet-analysis point of view we discard possible valuable scale information. The width of the scaling function spectrum is therefore an important parameter in the wavelet transform design. The shorter its spectrum the more wavelet coefficients, we will have and more scale information. But , as on , in the discrete wavelet transform this problem is more or less automatically solved.

The low-pass spectrum of the scaling function allows us to state some sort of admissibility condition similar to

$$\int \varphi(t)\,dt = 1 \quad \text{..........(1.11.2)}$$

Which shows that the 0^{th} moment of the scaling function can not vanish.

If one wavelet can be seen as a band-pass filter and scaling function is a loqw-pass filter, then a series of dilated wavelets together with a scaling function can be seen as a filter bank.

1.12 SUBBAND CODING

If we regard the wavelet transform as a filter bank, then we can consider wavelet transforming a signal as passing the signal through this filter bank. The outputs of the different filter stages are the wavelet-and scaling function transform coefficient. Analyzing a signal by passing it through a filter bank is not a new idea and has been around for many years under the name sub-band coding. It is used for instance in computer vision applications.

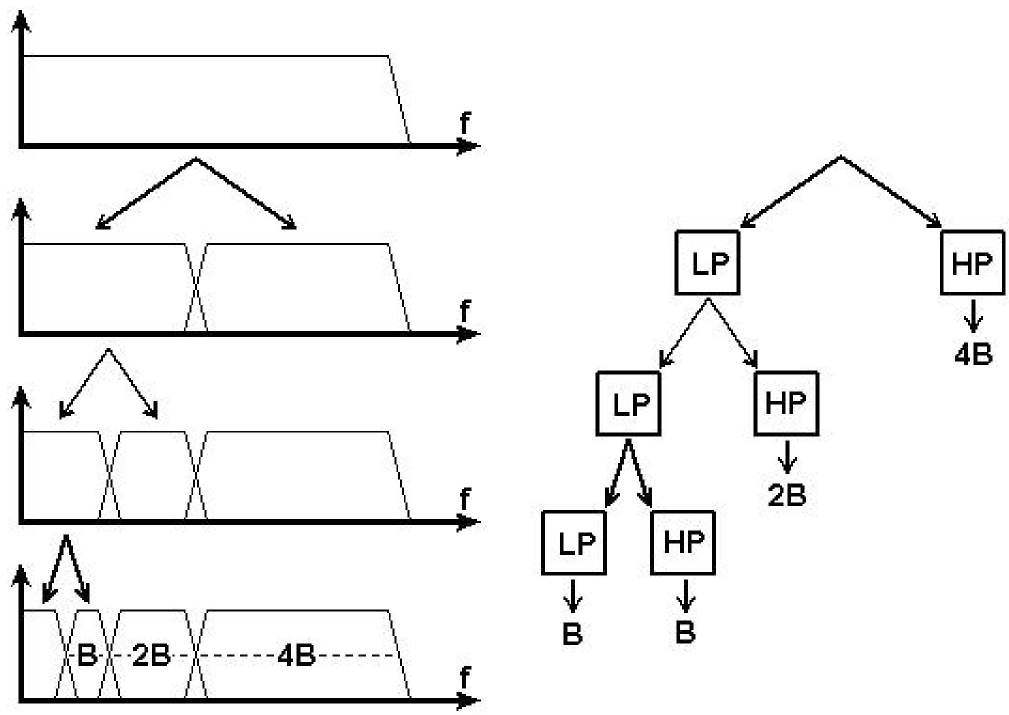


Fig 7 sub band coding function

The filter bank needed in subband coding can be built in several ways. One way is to build many band-pass filters to split the spectrum into frequency bands. The advantages is that the width of every band can be chosen freely, in such a way that the spectrum of the signal to analyze is covered in the places where it might be interesting. The disadvantage is that we will have to design every filter separately and this can be a time consuming process. Another way is to split the signal spectrum in two parts a low-pass and a high-pass part. The high-pass part contains the smallest details we are interested in and we could stop here. We now have two bands. However the low-pass part still contains some details and therefore we can split it again. And again, until we are satisfied with the number of bands we have created. In this way we have created an iterated filter bank. Usually the number of bands is limited by for instance the amount of data or computation power available. The process of splitting the spectrum is graphically delayed in fig. The advantage of this scheme is that we have to design only two filters , the disadvantage is that the signal spectrum coverage is fixed.

Looking in above fig. we see that what we are left with after the repeated spectrum splitting is a series of band-pass bands with doubling bandwidth and one low-pass band. (Although in first split gave us a high-pass band and a low-pass band, in reality the high-pass band is a band-pass band due to te limited bandwidth of the signal.) .The same can be done in another way by feeding the signal into a bank of band-pass filters of which each filter has a bandwidth twice as wide as his left neighbour (the frequency axis runs to the right here) and a low-pass filter. This is same as applying a wavelet transform to the signal. The wavelet gives us the band-pass bands with doubling bandwidth and scaling function provides us with the low-pass band. So we can conclude that a wavelet transform is the same thing as a sub-band coding scheme using a constant-Q filter bank. This analysis is referred to as a multi-resolution analysis.

1.13 ORTHOGONAL WAVELET

An orthogonal wavelet is a Wavelet where the associated wavelet transform is orthogonal. That is the inverse wavelet transform is the ad joint of the wavelet transform. If this condition is weakened you may end up with bi-orthogonal wavelet

The scaling function is a re-definable function. That is, it is a fractal functional equation, called refinement equation:

$$\varphi(x) = \sum_{k=0}^{N-1} a_k \varphi(2x - k) \quad ,\ldots\ldots(1.13.1)$$

where the sequence $(a_0, a_1, \ldots a_n)$ of real numbers is called scaling sequence or scaling mask. The wavelet proper is obtained by a similar linear combination,

$$\psi(x) = \sum_{k=0}^{M-1} b_k \varphi(2x - k) \quad \text{..(1.13.2)}$$

where the sequence $(b_0, b_1, \ldots, b_n)$ of real numbers is called wavelet sequence or wavelet mask.

A necessary condition for the orthogonality of the wavelets is, that the scaling sequence is orthogonal to any shifts of it by an even number of coefficients:

$$\sum_{n \in z} a_n a_{n+2m} = 2\delta_{m,0} \quad \text{..(1.13.3)}$$

In this case there is the same number M=N of coefficients in the scaling as in the wavelet sequence, the wavelet sequence can be determined as$b_n=(-1)^n a_{N-1-n}$. In some cases the opposite sign is chosen.

1.14 BI-ORTHOGONAL WAVELET

A bi-orthogonal wavelet is a wavelet where the associated wavelet transform is invertible but not necessarily orthogonal. Designing bi-orthogonal wavelets allows more degrees of freedoms than orthogonal wavelets. One additional degree of freedom is the possibility to construct symmetric wavelet functions.

In the bi-orthogonal case, there are two scaling functions $\varphi, \tilde{\varphi}$, which may generate different multi-resolution analyses, and accordingly two different wavelet functions $\psi, \tilde{\psi}$. So the numbers M,N of coefficients in the scaling sequences $a, \tilde{a}$ may differ. The scaling sequences must satisfy the following bi-orthogonality condition

$\sum_{n \in z} a_n \tilde{a}_{n+2m} = 2\delta_{m,0}$ .Then the wavelet sequences can be determined as

$b_n = (-1)^n \tilde{a}_{M-1-n}$, $\tilde{b}_n = (-1)^n \tilde{a}_{M-1-n}$, n=0,…,M-1 and ,n=0,…,N-1…..(1.13.4)

1.15 GENERATING SCALING FUNCTIONS AND WAVELETS FROM FILTER COEFFICIENTS

The following equation represents as

$$\Phi(2\omega) = H(\omega)\Phi(\omega) \quad \text{………………………………...…………….(1.15.1)}$$

$H(\omega)$ is the frequency response of H.

Rewriting this above equation as

$$\Phi(\omega) = H(\omega/2)H(\omega/4)\ldots H(0) \quad \text{………………………………….(1.15.2)}$$

Where we have set Φ(0)=1, we have the coefficients of the impulse response of a discrete-time filter h(n) satisfy the paraunitary conditions. The sequence 2h(n) can serve as the set of coefficients for the dilation equation to generate a potential scaling function □(t) for an orthonormal decomposition. If substitution of the frequency response H(ω) in the right-hand side of the equation leads to a function of Φ(ω), then its inverse Fourier transform is such a scaling function. There is a simple time-domain iteration method based on this result.

The steps of the algorithm are

1. Set c(n)=2h(n).
2. Let the initial scaling function be the haar scaling function

$$\phi_0 = \begin{cases} 1 & 0 \le t < 1 \\ 0 & \text{otherwise } 0 \end{cases} \quad \text{……………………………….……….…(1.15.3)}$$

3. At iteration n set

$$\phi_n(t) = \sum_t c(l)\phi_{n-1}(2t-l) \quad \text{……………………………………..…(1.15.4)}$$

4. Iterate until either divergence is established or the desired convergence is obtained. If there is convergence, the scaling function is given by

$$\phi(t) = \lim_{n \to 0} \phi_n(t) \quad \text{……………………………………………….....(1.15.5)}$$

Transforming the iteration to the frequency domain, at the end of the first iteration,

$$\Phi_1(\omega) = H(\omega/2)\Phi_0(\omega/2) \quad \text{……………………………..…(1.15.6)}$$

Where $\Phi_n(\omega)$ denotes the Fourier transform of the scaling function at the n^{th} iteration. The function $\Phi_0(\omega)$ is the Fourier transform of the Haar scaling function. Thus the scaling function become

$$\Phi_1(\omega) = H(\omega/2)H(\omega/4)\Phi_0(\omega/4) \qquad \text{..............................(1.15.7)}$$

1.16 WAVELET TRANSFORM AND DATA COMPRESSION

The most wide spread application of the wavelet transform so far has been for data compression. This is related to the fact that the DWT is tied closely to subband decomposition, and the latter was already being used for compression by the time the connection between the two was established by mallat.

Compression in the context of speech compression, image compression, and so forth, connotes the process of starting with a source of such data in digital form and creating a representation for it that uses fewer bits than the original. The aim is to reduce storage requirements or transmission time when such information is communicated over a distance. Ideally we want the compression process to be reversible. That is given the compressed data, we would like to get the original data. When it is possible to do so then compression is said to be lossless; otherwise we have lossy compression The error signal is represented as

$$e(n) = s(n) - \hat{s}(n). \qquad \text{...(1.16.1)}$$

For lossless compression, e(n) is identically zero. For lossy compression, an objective measure of distortion or figure of merit for the quality of the reproduction signal or image is the mean squared value of the e(n) . One might use the related measures of signal to noise ratio (SNR) or peak signal to noise ratio (PSNR) defined as

$$SNR(dB) = 10\log_{10}\frac{meansquared\ s(n)}{meansquared\ e(n)} \qquad \text{................................(1.16.2)}$$

1.17 WAVELET MATCHING

Generate a mother wavelet $\psi(t)$ such that the latter is as close as possible to the specified function in some optimal sense while generating an orthonormal basis. Now construct the meyer-type scaling function and wavelet since it is used in the derivation of the matching algorithm. Let $\phi(t)$ be a real value function of time and $\Phi(\omega)$ be its Fourier transform, let $\Phi(\omega)$ be bandlimited to $|\omega| \le {}^{2\pi}\!/_{3}$ such that $|\Phi(\omega)| = 1$

$0 \le \omega \le {}^{2\pi}\!/_{3}$

$$|\Phi(\pi+\omega)|^2 + |\Phi(\pi-\omega)|^2 = 1 \text{ .... (1.17.1)}$$

For $|\omega| \le {}^{\pi}\!/_{3}$. By virtue of the fact that $\phi(t)$ is real valued, it follows that $|\Phi(\omega)|$ is symmetric about origin.

It can be shown that the function $\Phi(\omega)$ is symmetric about the origin.

It can be shown that the function $\Phi(\omega)$ satisfies the poisons summation formula,

$$\sum_{k} |\Phi(\omega + 2\pi k)|^2 = 1 \text{ .... (1.17.2)}$$

Thus indicating that {□(t-k) : k integer} is an orthogonal set. The function □(t) serves as a scaling function in an orthonormal MRA. The corresponding wavelet amplitude spectrum $|\psi(\omega)|$ is bandlimited to $2\pi/3 < |\omega| \le 8\pi/3$ and is given by

$$|\psi(\omega)| = \begin{cases} |\Phi(\omega - 2\pi)| & 2\pi/3 \le \omega \le 4\pi/3 \\ |\Phi(\omega/2)| & 4\pi/3 \le \omega \le 8\pi/3 \end{cases} \text{ .... (1.17.3)}$$

An interesting consequence is

$$|\psi(4\pi/3 - \omega)| = |\psi(4\pi/3 + 2\omega)| \text{ for } 0 \le \omega \ge 2\pi/3 \text{ .... (1.17.4)}$$

Now F(ω) be the Fourier transform of the specified signal. We address the problem of generating a function $|\psi(\omega)|$ such that it satisfies the conditions of a meyer wavelet while minimizing the squared norm of the function $|F(\omega)|^2 - |\psi(\omega)|^2$

$$|F(\omega)|^2 = |\psi(\omega)|^2 \quad \ldots\ldots(1.17.5)$$

And

$$|A(\omega)| = |\psi(\omega)|^2 \quad \ldots\ldots(1.17.6)$$

We minimize

$$j = \int [G(\omega) - A(\omega)]^2 d\omega \quad \ldots\ldots(1.17.7)$$

Subject to the condition that A(ω) is bandlimited to $2\pi/3 \le |\omega| \le 8\pi/3$ and

Where c is a constant $A(\pi-\omega)^2 + A(\pi+\omega)^2 = C$. ……(1.17.8)

The idea is first to minimize the cost function with C as a free parameter and then to minimize further with respect to C to get the minimum.

$$a = C/2 + (Fa - Fb - Fc + Fd)/2 \quad \ldots\ldots(1.17.9)$$

$$b = C/2 - (Fa - Fb - Fc + Fd)/2 \quad \ldots\ldots(1.17.10)$$

Extending this to the entire function yields

$$A(\omega) = C/2 + [G(\omega) - G(2\pi - \omega) - G(2\omega) + G(4\pi - 2\omega)]/2 \quad \ldots\ldots(1.17.11)$$

Now further minimizing of the cost function with respect to C yields

$$C = \frac{1}{2\pi}\left[\int_{2\pi/3}^{4\pi/3} G(\omega)d\omega + \frac{1}{2}\int_{4\pi/3}^{8\pi/3} G(\omega)d\omega\right] \quad \ldots\ldots(1.17.12)$$

1.18 AUDIO COMPRESSION

The DTWT or subband decomposition techniques are used for the compression of audio and wideband speech signals. There seems to be a perceptual basis in audio perception for using such decomposition. The frequency range for human hearing extends to 20 kHz. When an analog audio signal is sampled, the sampling frequency has to be greater than 40 kHz. In applications such as audio recording on conventional compact disk (CD), the sampling rate is 44.1 kHz. Since the sampled data are stored in digital form, only a finite number of levels, determined by the number of bits allotted per sample, can be used to represent the signal. In audio CDs, 16 bits per sample are used. This allows 65,536 levels. Then this levels are decoded is called pulse code modulation.

1.19 APPLICATIONS OF WAVELET TRANSFORMS

Finally, applications of widely used standard DWT implementations, utilizing its Multi-scale and Multi-resolution capabilities with fast filter bank algorithms are numerous to describe. Depending upon the application, extensions of standard DWT namely WP and SWT are also employed for improved performance at the cost of higher redundancy and computational complexity.

A few of such applications in data compression, de noising, source and channel coding , biomedical, non-destructive evolution, numerical solutions of PDE , study of distant universe, zero-crossing, fractals, turbulence, speckle removal, edge detection and object isolation, image fusion, scaling functions as signalling pulses, and finance are comprehensively covered in. wavelet applications in may diverse fields such as physics , medicine, and biology, computer graphics, communications and multimedia etc. can be found in various books on wavelets.

1.20 TYPES OF WAVELETS

HARR

Any discussion of wavelets begins with Haar wavelet, the first and simplest. Haar wavelet is discontinuous, and resembles a step function. It represents the same wavelet as Daubechies I.

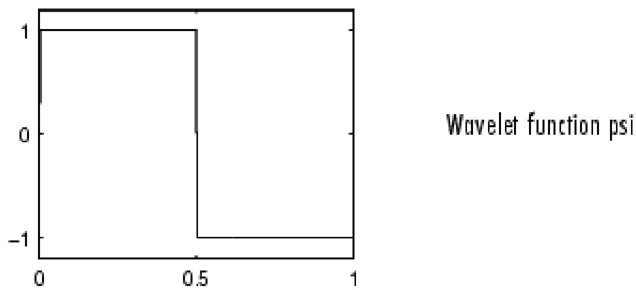


Fig.8

DAUBECHIES

Ingrid Daubechies, one of the brightest stars in the world of wavelet research, invented what are called compactly supported orthonormal wavelets thus making discrete wavelet analysis practicable.

The names of the Daubechies family wavelets are written , where is the order, and the "surname" of the wavelet. The wavelet, as mentioned above, is the same as wavelet. Here are the wavelet functions psi of the next nine members of the family:

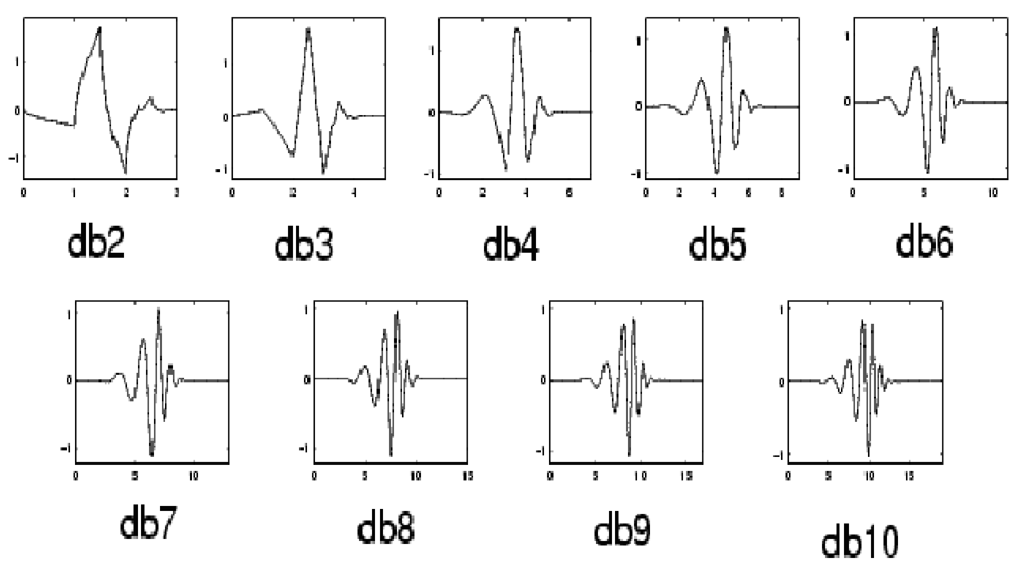
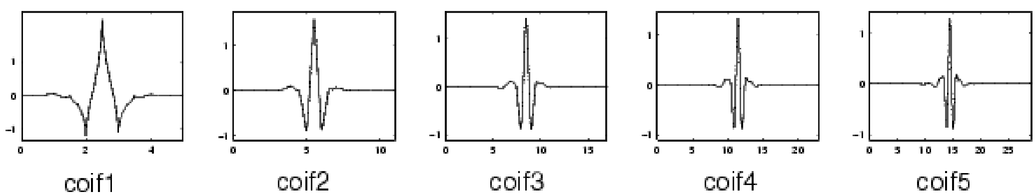


Fig 9

COIFLETS

Built by I. Daubechies at the request of R. Coifman. The wavelet function has 2*N* moments equal to 0 and the scaling function has 2*N*-1 moments equal to 0. The two functions have a support of length 6*N*-1. You can obtain a survey of the main properties of this family by typing from the MATLAB command line

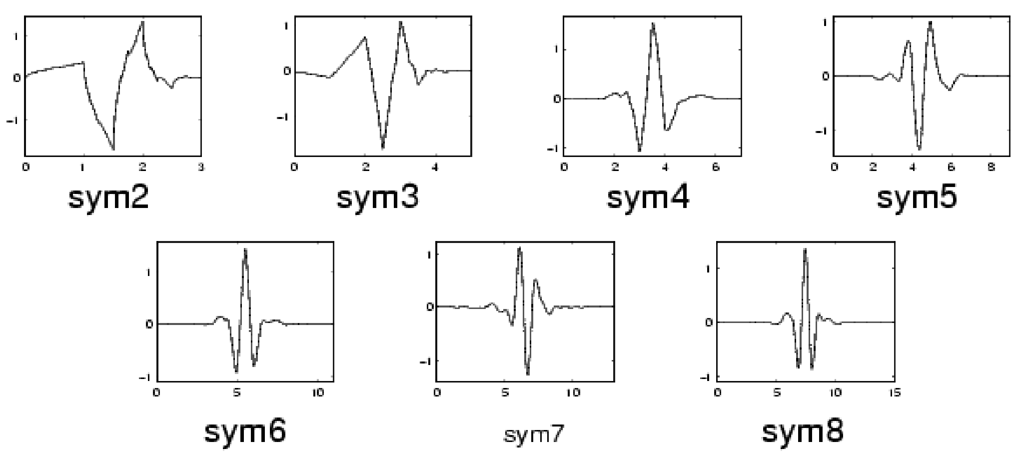
Fig 10



SYMLETS

The symlets are nearly symmetrical wavelets proposed by Daubechies as modifications to the family. The properties of the two wavelet families are similar. Here are the wavelet functions psi.

Fig 11



BIORTHOGONAL

This family of wavelets exhibits the property of linear phase, which is needed for signal and image reconstruction. By using two wavelets, one for decomposition (on the left side) and the other for reconstruction (on the right side) instead of the same single one, interesting properties are derived.

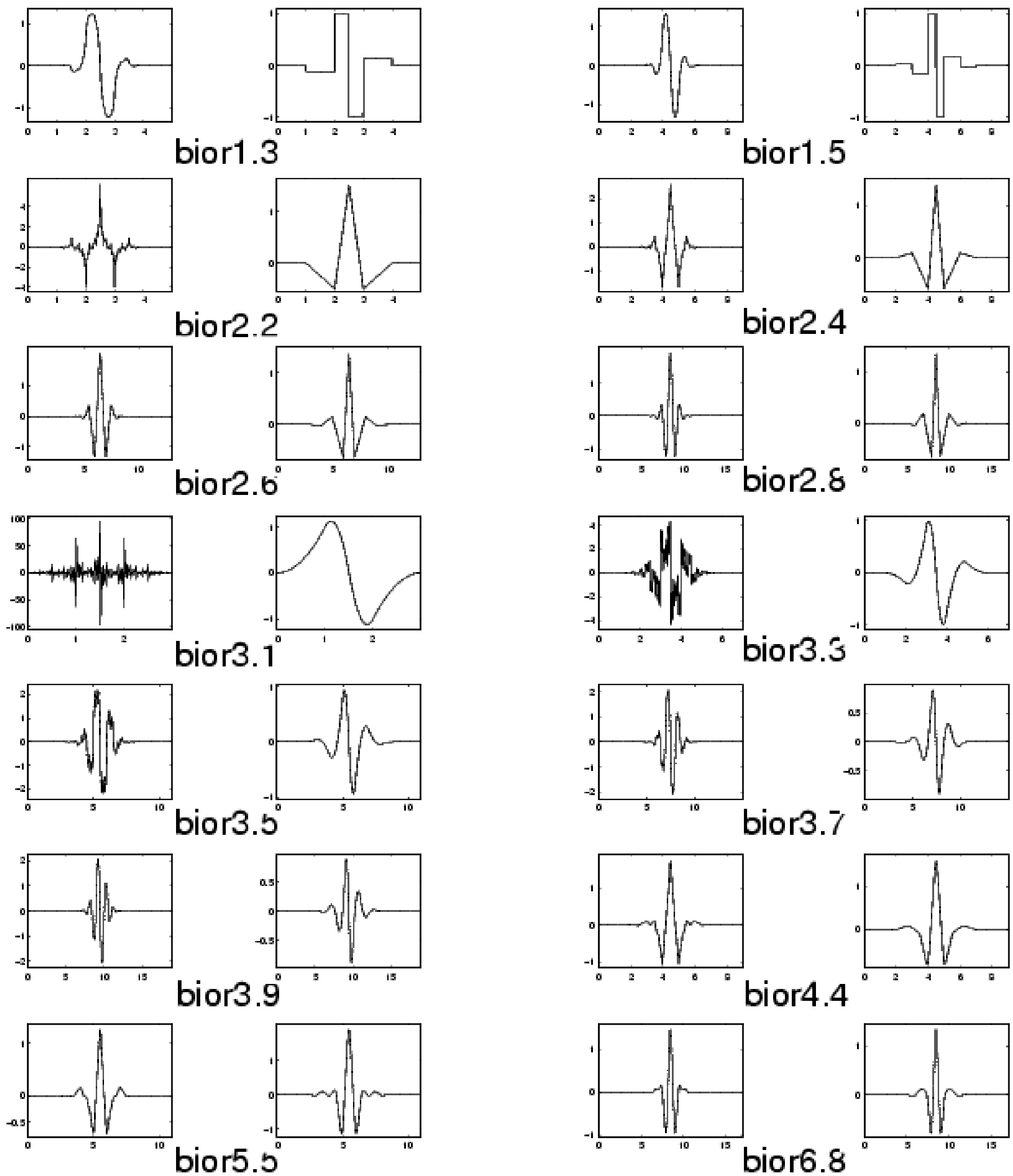


Fig 12

MORLET

This wavelet has no scaling function, but is explicit.

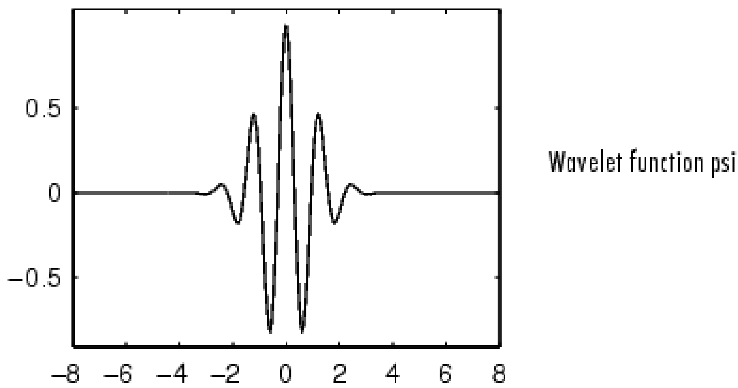


Fig 13

MEXICAN HAT

This wavelet has no scaling function and is derived from a function that is proportional to the second derivative function of the Gaussian probability density function.

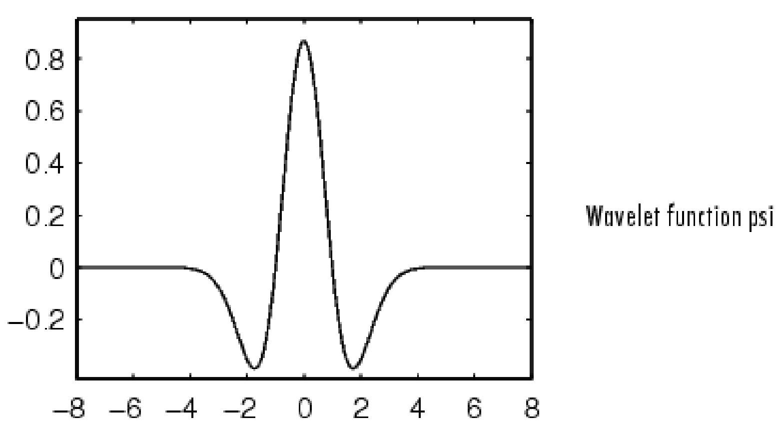


Fig 14

MEYER

The Meyer wavelet and scaling function are defined in the frequency domain.

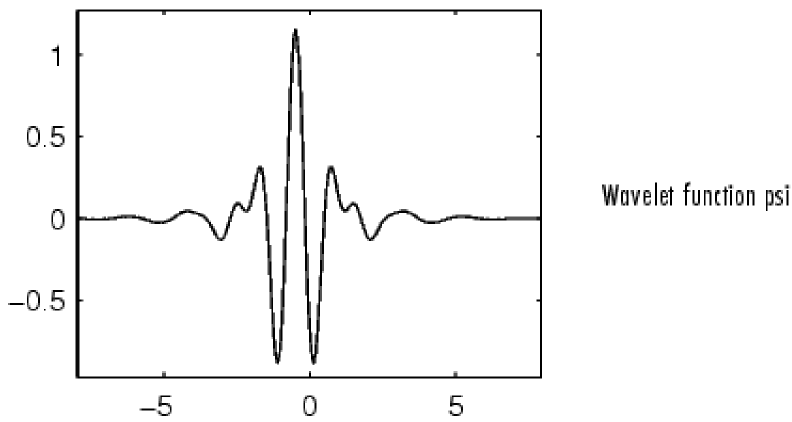


Fig 15

CHAPTER-2

LITERATURE SURVEY

2.1 ALGORITHMS FOR DESIGNING WAVELETS TO MATCH A SPECIFIED SIGNAL

Algorithms for designing a mother wavelet $\psi(x)$ such that it matches a signal of interest and such that the family of wavelets $\left\{2^{-(j/2)}\psi(2^{-j}x-k)\right\}$ forms an orthonormal Riesz basis of $L^2(R)$ are developed. The algorithms are based on a closed form solution for finding the scaling function spectrum from the wavelet spectrum. Many applications of signal representation, adaptive coding and pattern recognition require wavelets that are matched to a signal of interest. Most current design techniques, however, do not design the wavelet directly. They either build a composite wavelet from a library of previously designed wavelets, modify the bases in an existing multi-resolution analysis or design a scaling function that generates a multi-resolution analysis with some desired properties. In this paper, two sets of equations are developed that allow us to design the wavelet directly from the signal of interest. Both sets impose band-limitedness, resulting in closed form solutions. The first set derives expressions for continuous matched wavelet spectrum amplitudes. The second set of equations provides a direct discrete algorithm for calculating close approximations to the optimal complex wavelet spectrum. The discrete solution for the matched wavelet spectrum amplitude is identical to that of the continuous solution at the sampled frequencies. An interesting by product of this work is the result that Meyer's spectrum amplitude construction for an orthonormal band limited wavelet is not only sufficient but necessary. Specific examples are given which demonstrate the performance of\the wavelet matching algorithms for both known orthonormal wavelets and arbitrary signals.

Daubechies introduces the wavelet transform as "a tool that cuts up data or functions or operators into different frequency components, and then studies each component with a resolution matched to its scale". One of the exciting advantages of wavelets over Fourier analysis is the flexibility they afford in the shape and form of the analyzer, that which "cuts up" and "studies" the signal of interest. However ,with flexibility comes the difficult task of choosing or designing the appropriate wavelet or wavelets for a given application.

look at a multi resolution decomposition as the output of a bank of matched filters , we can see the need for the analyzing wavelet to “look” like the signal of interest. In signal detection applications, the decomposition of a signal in the presence of noise using a wavelet matched to the signal would produce a sharper or taller peak in time-scale space as compared to standard non matched wavelets. The design techniques developed to date do not specifically address the need for maximizing correlation in a signal decomposition. Daubechies’ classic technique for finding orthonormal wavelet bases with compact support is often used as the default in many wavelet applications. However, the wavelets produced are independent of the signal being analyzed. Tewfik, Sinha, and Jorgensen have developed a technique for finding the optimal orthonormal wavelet basis for representing a specified signal within a finite number of scales. Gopinath, Odegard, and Burrus extended the results of Tewfik, by assuming bandlimited signals and finding the optimal M-band wavelet basis for representing a desired signal, again within a finite number of scales. Both of these approaches seek to represent a signal over some number of scales. However, the desired output of a multiresolution decomposition of a bandpass signal using a matched wavelet is representation in one or at most two scales.

The wavelet design techniques developed Mallat and Zheng , and Chen and Donoho , build non orthonormal wavelet bases from a library of existing wavelets in such a way that some error cost function is minimized. These techniques are constrained by the library of functions used and do not satisfy the need for optimal correlation in both scale and translation. Sweldens developed the lifting scheme for constructing biorthogonal wavelets . Aldroubi and Unser match a wavelet basis to a desired signal by either projecting the desired signal onto an existing wavelet basis, or transforming the wavelet basis under certain conditions such that the error norm between the desired signal and the new wavelet basis is minimum. Both of these techniques are constrained by their initial choice of MRA.

Apart from being of mathematical interest, the problem of deriving orthonormal wavelets directly from a signal of interest has specific application to signal detection, image enhancement, and target detection, to name a few. In this paper, we will show that in the case of orthonormal MRA's with bandlimited wavelets, there is a solution to finding wavelets that "look" like a desired signal. The only additional constraints are the necessary conditions for an MRA and the signal of interest itself. While the matching algorithm is sub-optimal in the sense that it is performed on the spectrum magnitude and phase independent of one another, we will show by way of examples that it produces good matching wavelets.

➔In an orthonormal MRA(OMRA), a signal, $f(x) \in V_{-1}$, is decomposed into an infinite series of detail functions, $\{g_j(x)\}$ such that

$$f(x) = \sum_{j=0}^{\infty} g_j(x) \, , \qquad \text{(2.1.1)}$$

The first level decomposition is done by projecting $f(x)$ onto two orthogonal subspaces, V_0 and W_0, where $V_{-1} = V_0 \oplus W_0$ and $\oplus$ is the direct sum operator. The projection produces $f_0(x) \in V_0$, a low resolution approximation of $f(x)$, and $g_0(x) \in W_0$ .The detail lost in going from $f(x)$ to $f_0(x)$. The decomposition continues by projecting $f_0(x)$ onto V_1 and W_1 and so on. The orthonormal bases of W_j and V_j are given by

$$\psi_{j,k} = 2^{-j/2}\psi(2^{-j}x - k) \qquad \text{(2.1.2)}$$

$$\phi_{j,k} = 2^{-j/2}\phi(2^{-j}x - k) \qquad \text{(2.1.3)}$$

Where $\psi_{j,k}$ is the mother wavelet and $\phi_{j,k}$ is the scaling function. Where

$$\int \psi(x)dx = 0 \Leftrightarrow \Psi(0) = 0 \qquad \text{(2.1.4)}$$

$$\int \phi(x)dx = 1 \Leftrightarrow \Phi(0) = 1 \qquad \text{(2.1.5)}$$

And $\Phi(\omega)$ and $\Psi(\omega)$ are the Fourier transform of $\psi(x)$ and $\phi(x)$, respectively. The projection equations are

$$g_j(x) = \sum_{k=-\infty}^{\infty} d_j^k 2^{-j/2} \psi(2^{-j} x - k) \quad \text{.........(2.1.6)}$$

$$d_k^j = \langle f_{j-1}(x), \psi_{j,k} \rangle \quad \text{.........(2.1.7)}$$

$$f_j(x) = \sum_{k=-\infty}^{\infty} c_k^j \phi(2^{-(j/2)} x - k) \quad \text{.........(2.1.8)}$$

$$c_j^{k=} \langle f_{j-1}(x), \phi_{j,k} \rangle \quad \text{.........(2.1.9)}$$

Where d_j^k and c_j^k are the projection coefficient and $\langle .,. \rangle$ is the L^2 inner product. The nested sequence of subspaces, $\{V_j\}$, constitutes the multiresolution analysis. For the MRA to be orthonormal $\psi_{j,k}$ and $\phi_{j,k}$ must be orthonormal bases of W and V_j, respectively and $W_j \perp W_k$, for $j \neq k$, and $W_j \perp V_j$, which lead to the following conditions on ψ and ϕ.

$$\langle \phi_{j,k}, \phi_{j,m} \rangle = \delta_{k,m} \quad \text{.........(2.1.10)}$$

$$\langle \phi_{j,k}, \psi_{j,m} \rangle = 0 \quad \langle \psi_{j,k}, \psi_{l,m} \rangle = \delta_{j,l}, \delta_{k,m} \quad \text{.........(2.1.11)}$$

The Fourier transform gives the poisson summation which is 1 for all ω.

$$\sum_{m=-\infty}^{\infty} \left| \Phi(w + 2\pi m)^2 \right| = 1. \quad \text{.........(2.1.12)}$$

Since $\phi(x) \in V_0 \subset V_{-1}$ and $\psi(x) \in W_0 \subset V_{-1}$, they can be represented as linear combinations of the basis of V_{-1}

$$\phi(x) = 2 \sum_{k=-\infty}^{\infty} h_k \phi(2x - k) \quad \text{.........(2.1.13)}$$

$$\psi(x) = 2 \sum_{k=-\infty}^{\infty} g_k \phi(2x - k) \quad \text{.........(2.1.14)}$$

In the frequency domain

$$\Phi(\omega) = H(\frac{\omega}{2})\Phi(\frac{\omega}{2}) \qquad (2.1.15)$$

$$\Psi(\omega) = G(\frac{\omega}{2})\Phi(\frac{\omega}{2}) \qquad (2.1.16)$$

$$g_k = (-1)^{k+1} h_{1-k} \Leftrightarrow G(\omega) = e^{j\omega} H(\omega + \pi) \qquad (2.1.17)$$

2.2 SIGNAL DETECTION

Using a matched filter bank interpretation of wavelet transforms, we propose to design a wavelet that "matches" the signal of interest such that the output of the matched filter bank is maximized. The projection equation for the detail functions, given in (2.1.5), is an inner product integral and can be rewritten in the frequency domain by way of Parseval's Identity as

$$d_j^k = \langle f(x), \psi_{j,k} \rangle = \langle F(\omega), \psi_{j,k}(2^j \omega) \rangle \qquad (2.2.1)$$

Where $\Psi_{j,k}(2^j \omega) = 2^{-j/2} e^{-i2^j \omega k} \Psi(2^j \omega)$ is the Fourier transform of $\psi_{j,k}(x)$. The energy of d_j^k at a particular scale . j_0 ,and translation ,k_0 is given by its squared magnitude

$$\left| d_{k_0}^{j_0} \right|^2 = \left| \langle F(\omega), \Psi_{j_0,k_0}(2^{j_0} \omega) \rangle \right|^2 \qquad (2.2.2)$$

Applying the Cauchy-schwarz inequality to the right side gives

$$\left| \langle F(\omega), \Psi_{j_0,k_0}(2^{j_0} \omega) \rangle \right|^2 \le \langle F(\omega), F(\omega) \rangle \langle \Psi_{j_0,k_0}(2^{j_0} \omega), \Psi_{j_0,k_0}(2^{j_0} \omega) \rangle \qquad (2.2.3)$$

Where the equality holds if and only if

$$F(\omega) = K \Psi_{j_0,k_0}(2^{j_0} \omega) \qquad (2.2.4)$$

Where both F and Ψ are complex spectra. Therefore $\left|d_{k_0}^{j_0}\right|^2$, is maximized when the complex frequency spectrum of ψ_{j_0,k_0} is identical to that of . Therefore, we would like to develop a method for matching the complex spectrum of the wavelet to that of the desired signal while maintaining the conditions for an orthonormal MRA. However, because the conditions for orthonormality are on the spectrum amplitude (Poisson summation) only, our solution matches the spectrum amplitudes and group delays independently. While this approach is not ideal from an optimization standpoint, we will show that it One other difficulty in matching the wavelet spectrum directly to that of the desired signal is the fact that the conditions for an orthonormal MRA are on the scaling function and its frequency spectrum, not the wavelet specifically. If we were to construct a wavelet that satisfied its conditions for an orthonormal basis, it would not necessarily lead to a scaling function that generates an orthonormal MRA. Therefore, we must propagate the conditions for an orthonormal MRA from the 2-scale sequence and scaling function to the wavelet, match the wavelet to the desired signal under those conditions, and then calculate the scaling function and 2-scale sequence always guaranteeing that the conditions for an orthonormal MRA are satisfied still leads to good matching wavelets.

2.3 PROPERTIES OF A WAVELET IN AN OMRA

Most wavelet construction techniques first find a scaling function that satisfies (2.1.4) and (2.1.5), (2.1.12) , and (2.1.13) and then calculates the wavelet using (2.1.14) and (2.1.15) , (2.1.16) and (2.1.17).

2.3.1 Finding the scaling function from a Wavelet

The first step in deriving the OMRA conditions for the wavelet spectrum amplitude is providing a means of deriving the scaling function from the mother wavelet. Finding the wavelet from the scaling function is simple using (2.1.14), however, it is not invertible. To derive an expression for in terms of , the conditions provided in Section II will be applied directly. Conditions (2.1.4), (2.1.7) and (2.1.15) and (2.1.16) are required for to generate an orthonormal MRA, thereby satisfying (2.1.7). From (2.1.15) and (2.1.16), we get the following expression

$$\left|\Phi(\omega)\right|^2 = \left|\Psi(2\omega)\right|^2 + \left|\Phi(2\omega)\right|^2 \quad \text{……………….……………………(2.3.1)}$$

Repeated substitution of $\left|\Phi(2^k\omega)\right|^2$ for $k \geq 1$ into above equation gives the following closed form solution

$$\left|\Phi(\omega)\right|^2 = \sum_{j=1}^{\infty} \left|\Psi(2^j\omega)\right|^2 \text{ for } \omega \neq 0 \quad \text{……………………………………..(2.3.2)}$$

2.3.2 Properties of the Wavelet spectrum amplitude

Now that we have an expression for finding $|\Phi|$ from $|\Psi|$, we need to develop the constraints on $|\Psi|$ that are necessary and sufficient to guarantee $\phi_{j,k}$ is an orthonormal basis of V_j . Using (2.3.2), conditions (2.1.4), (2.1.7), and (2.1.15 and 2.1.16) can be transferred to conditions on $|\Psi(\omega)|$. To guarantee a closed form solution, we assume the scaling function spectrum is bandlimited with only a countable number of zeros. With this assumption, we can derive the following theorems for the properties of orthonormal bandlimited scaling function and wavelet spectra.

2.3.3 Properties of the Wavelet spectrum phase

It would be convenient if we could simply set the phase of Ψ to the phase of the desired signal spectrum, F, thereby cancelling the complex component of (2.2.4). However, just as in the previous section we showed that Ψ has specific constraints on its amplitude, here we will show that has specific constraints on the structure of its phase as well. First we will develop an expression for the group delay of $\Psi(\omega)$ in terms of the group delay of the scaling function spectrum, $\Phi(\omega)$. Substituting (2.1.17) into (2.1.15) gives

$$\Psi(2\omega) = e^{-i\omega} \frac{\Phi(2\omega + 2\pi)}{\Phi(\omega + \pi)} \Phi(\omega) \quad \text{……………………………………..……..(2.3.3)}$$

And so the phase of Ψ becomes

$$\theta_\Psi(\omega) = -\frac{\omega}{2} - \theta_\Phi(\omega + 2\pi) + \theta_\Phi(\frac{\omega}{2} + \pi) + \theta_\Phi(\frac{\omega}{2}) \quad \text{…………………..……..(2.3.4)}$$

Where $\theta_\Psi(\omega)$ and $\theta_\Phi(\omega)$ are the phases of Ψ and Φ, respectively. The negatives of the group delays are denoted as $\wedge_\Psi$ and $\wedge_\Phi$.

Setting $\Gamma_\Psi(\omega) = \wedge_\Psi(\omega) + 1/2$ gives

$$\Gamma_\Psi(\omega) = -\wedge_\Phi(\omega + 2\pi) + \frac{1}{2} \wedge_\Phi(\frac{\omega}{2} + \pi) + \wedge_\Phi(\frac{\omega}{2}) \quad \text{…………………...………(2.3.5)}$$

Next we develop an expression for the group delay of $\Psi(\omega)$ in terms of the group delay of $H(\omega)$,denoted as $\lambda(\omega)$. By repeated substitutions of the equations in (2.1.15) and (2.1.17), we get the following infinite products

$$\Phi(\omega) = \prod_{m=1}^{\infty} H(\frac{\omega}{2^m}) \quad \text{…………………………………………...……………..(2.3.6)}$$

$$\Psi(\omega) = e^{-i(\omega/2)} H(\frac{\omega}{2} + \pi) \prod_{m=2}^{\infty} H(\frac{\omega}{2^m}) \quad \text{…………………………...……………….(2.3.7)}$$

Where H(ω) is 2π-periodic.

2.4 Design Issues For Matched Wavelets

Wavelets and other methods of time-frequency analysis in their many practical applications require that the analyzing filter sequence have certain desired properties. Typical of these properties are good time-frequency localization, energy compaction, orthogonality, and regularity. In addition it may be desired to have an analyzing filter that resembles a given waveform or that statistically matches a process. These wavelets will be referred to as matched wavelets. The "matching" of'wavelet filters can be classified in two groups namely "waveform matching", and "statistical matching". When a given waveform is mapped into a function that possesses perfect reconstruction (PR) and regularity properties, this is called waveform matched wavelet. One technique for obtaining waveform matched wavelets is given by FK mapping. On the other hand, in the case of statistical matching the wavelet filter is designed to optimize its energy compaction or time-frequency resolution performance with respect to a given process, that is with respect to a given autocorrelation function.

2.5 Design trade offs in statistically matched wavelets:

Matched wavelet filters were designed for an autoregressive, AR(l), model, and the following tradeoffs were observed:

1.The number of vanishing moments beyond k > 4 **,**brings a penalty on the energy compaction, and therefore on the coding gain of a constraint matrix. This may not be however a significant limitation since for most processes encountered in practice, like images, vanishing moments only up to 2 are required.

2. When filters were designed with time-frequency localization as a performance criterion, it was observed that, increasing the number of vanishing moments leads to more compact functions in time while spreading the frequency support,. In total the resolution cell $\sigma_n^2\sigma_\omega^2$ is slightly decreased for larger k.

3. The relationship of interband correlation and G_{SBC} in filter banks is not as clear as in the block transforms. However at the highest value of G_{SBC}, the two band correlation is not necessarily zero, though the correlation coefficient remains quite small.

2.6 WAVEFORM MATCHED WAVELETS

Wavelet filter banks can also be designed by matching to a specific waveform. Here the goal is not to maximize the energy compaction with respect to a process model as was the case in statistically matched wavelet filters, but starting from a given reference waveform, to map it to a closest function that possesses wavelet properties. A mapping operator, which for a given reference function f, finds a wavelet filter Ψ that is closest to it in the mean square norm, is given by Frazier-Kumar (FK) technique. This design technique for 1-D wavelet filters has been detailed in "The Discrete Orthonormal Wavelet Transform", and its extension to 2-D filters has been provided by Alkin in "A study of 2-D Wavelet Transform, Technical Report". In the FK technique the wavelet function $\Psi(t)$ is related to the reference function as follows:

$$\Psi(z) = \frac{|F(z)|}{\sqrt{|F(z)|^2 + |F(-z)|^2}} \qquad \text{(2.6.1)}$$

where F(z) denotes the z-transform of the reference function f(n). The resulting filters are of infinite impulse response type, hence must be windowed. The windowing and truncation invariably results in some loss of orthogonality or perfect reconstruction property.

In summary, matched wavelet design using the FK method consists of the following steps:

1. Choice of a reference function
2.Spectral factorization of above equation and windowing of the response.

At this stage, the contribution of our work to the FK methodology is first to investigate the selection of a proper reference function. It is suggested that the reference function can be chosen as a combination of the eigen-spectra of a process. In other words the reference function in above equation can be chosen as

$$F(\omega) = \sum_{k=1}^{N/2} \alpha_k u_k(\omega) \qquad (2.6.2)$$

Where the k[th] eigen spectrum is given by

$$u_k(\omega) = \sum_{j=0}^{N-1} u_{kj} e^{2\pi j / N} \qquad (2.6.3)$$

Where u_k denoting an eigen vector of the N×N covariance matrix. Since we will deal with a two-band scheme, then the reference function is constituted with the eigenspectra corresponding to the N/2 largest eigenvalues. Finally in (2.1.14) $\alpha(k)'s$ denote the combiner coefficients. These coefficients can be determined in a variety of ways, such **as** being proportional to the eigenvalues λ_i **,** or to select them to maximize the energy compaction. Note that after the FK mapping operation, there is no guarantee that the energy compaction performance of $\Psi(n)$ will be optimum. This second method can be formulated as follows, (for N=4):

$$\max \int_{-\pi}^{\pi} \left| \sum \alpha_k u_k(\omega) \right|^2 S_{xx}(\omega) d\omega \qquad (2.6.4)$$

For

$$\alpha_1 + \alpha_2 = 1$$

$$\alpha_1 < (2\lambda_1 / \lambda_2)\alpha_2 \qquad (2.6.5)$$

where the inequality constraint is derived in the proof in**[4]** and $S_{xx}(\omega)$ is the power spectral density. For N larger the optimization problem is similar, except that there are additional inequality constraints on α_k 's.

2.7 A NEW APPROACH FOR CONSTRUCT

Compared with Fourier transform, wavelet transform has better ability to analyze the singularities and irregular signal because of a multi-resolution analysis, and we can obtain the, details of signal at different scales by applying a wavelet transform. A chronological development of efforts in wavelet analysis shows that WT is a good tool for analyzing the non-stationary signal. One of the exciting advantages of wavelets over Fourier analysis is the flexibility they afford in the shape and form of the analyzer, that which "cuts up" and "studies" the signal of interest. The given signal can be decomposed by a set frequency channel of equal bandwidth on a logarithmic scale, an analysis of using constant Q-filters. In other words, the signal projects into the basis function of wavelet, each of which is a dilation and translation of a function called mother wavelet $\Psi(t)$ at diffident scale. Unlike FT, WT do not have a unique basis. Using different basis function of wavelet to analyze signal will get different results. That means that the wavelet designed matches the signal to be analyzed so that hest representation of the signal can be resulted. Usually, one uses a wavelet to do signal decomposition; it is something like a blind man's walk. If we know the particular features of the signal and then design a wavelet to match the signal, it would be better. This is a reason that match wavelets are finding applications in diverse fields and is a topic of current research. Since Mallat "A Theory for Multi resolution Signal Decomposition," has introduced wavelet transform in 1989 and led to the discrete wavelet transform, many researchers proposed ***so*** many methods for construction basis function wavelet. Daubechies proposed method to find orthonormal and biorthonomal wavelet bases with compact support, where she gave regularity and decay conditions. Since Mallat has proposed tower algorithm and Daubechies has given regularity conditions, one found the relationship between the wavelet transform and filter banks. Many approaches to build bases function of wavelet based on filter banks, were proposed. But these approaches of designing wavelet were independent of the signals to he analyzed. In order to obtain the best signal representation, many researchers are designing wavelet to match signal. Tewfik first addressed an important problem in wavelet analysis, which is to find the best wavelet multiresoulation analysis that approximates a given signal in some norm. J.O.Chapa proposed suboptimal algorithm for designing match wavelet. Z.Chen described the best wavelet matched tree to represent a signal. Anubha gave a method for finding the maximal projection of the given signal on to the scaling subspace, but he does not give how to choice wavelet filter banks.

2.8 MATCH WAVELET TO SIGNAL DETECTION

DESIGNING ALGRITHM

The criterion of wavelet matching signal is to minimize an error between original and reconstruction signal only with coefficients in the scale space. In order to make the $h_0(n)$ and $h_1(n)$ be low-pass and high-pass filters respectively, the objective function Equ.(8) is modified as following

$$E = w_1 e^2(t) dt + w_2 \left({}_0^{\omega_p} (1 - H_0(e^{j\omega}))^2 d\omega) + {}_{\omega_s}^{\pi} H_0(e^{j\omega})^2 d\omega \right) \quad \ldots\ldots\ldots\ldots\text{..(2.8.1)}$$

where ω_p and ω_s are the stop frequency of the pass-band and stop-band respectively, and w_1 w_2 are weights. The design procedures for constructing a match wavelet can be presented as follows:

Step I: Give an original discrete signal x(n) **.**

Step 2: Random guess an initial coefficient α **.**

Step 3: Compute analysis and synthesis filters $h_0(n)$, $h_1(n)$, $f_0(n)$ and $f_1(n)$

Step **4:**Compute the coefficient of the wavelet basis
function by pyramidal algorithm, and reconstruct
signal $\hat{x}(n)$ only with coefficients in the scale space.

Step ***5:*** Compute the objective function in Equ.(22).

Step 6: Adjust the coefficient α by optimal algorithm, such as Nelder-Mead simplex method.

Step **7:** If the objective function **is** not reached at minimum, go to step 3 again; other than end the calculation.

2.9 A NEW APPROACH FOR ESTIMATION OF STATISTICALLY MATCHED WAVELET

It is well known that a number of natural and man-made phenomenon exhibit self-similar characteristics. Also known as fractal processes, these waveforms arise in natural landscapes, ocean waves, and distribution of earthquakes and have found profound applications in various engineering fields like image analysis, characterization of texture in bone radiographs, network traffic analysis etc.

These processes are in general non-stationary, and they exhibit self-similarity in the statistical sense. A class of these signals is called $1/f^{\beta}$ processes, which have measured power spectral density (psd) that decays by a factor of $1/f^{\beta}$. Wornell [7] emphasized the role of wavelet basis expansion as a Karhunen–Loève-type expansion for $1/f^{\beta}$ processes. Since processes simultaneously exhibit statistical scale invariance and time invariance, wavelet-like bases having both scaling and shifting can best represent these signals.

The wavelet transform has emerged as an alternative to traditional Fourier-based analysis techniques for the analysis of non-stationary signals. However, unlike Fourier methods, wavelet transforms do not have a unique basis, which is one of the reasons why wavelets are finding applications in diverse fields and is a topic of current research. Since the basis here is not unique, it is natural to seek a wavelet that is best in a particular context. Particularly, in the context of signal/image compression, an issue of great research interest is to find a wavelet that can provide the best representation for a given signal.

M-Band Wavelets:

Similar to the two-band wavelet system, one can define a multi-resolution analysis (MRA) with a scaling factor of M to construct M-band wavelets. Motivation for a larger value of M comes from the desire to have a more flexible tiling of the time scale than that resulting from the M=2 wavelet, and it also comes from multirate filterbank theory. For the two-band wavelet system, the scaling function and wavelet function are defined by the two-scale difference equation as follows:

$$\phi(t) = \sum_n f_0 \sqrt{2\phi(2t-n)}, \forall n \in Z \quad \text{......(2.9.1)}$$

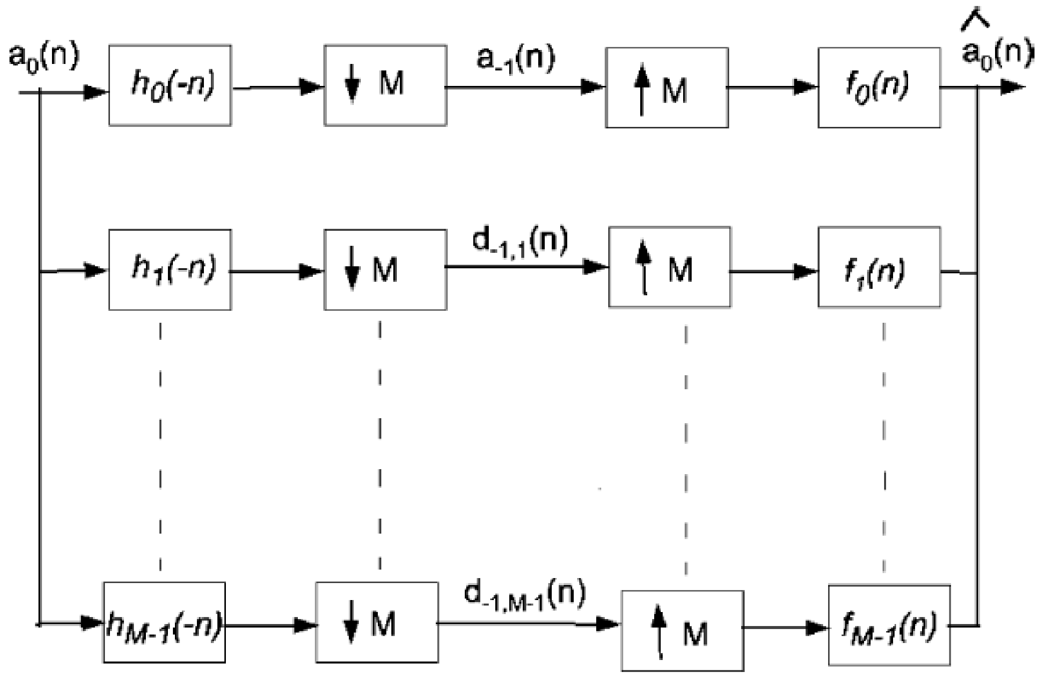


Fig 16 M-band wavelet

$$\psi(t) = f_1(n)\sqrt{2\phi(2t-n)}, \forall n \in Z \quad \text{......(2.9.2)}$$

For a two-band perfect reconstruction biorthogonal filterbank, the scaling filter f_0 and its dual h_0 , wavelet filter f_1, and its dual h_1 are required to satisfy the following conditions :

$$h_1(n) = (-1)^n f_0(N_1 - n) \quad \text{......(2.9.3)}$$

$$f_1(n) = (-1)^n h_0(N_1 - n), \quad \text{......(2.9.4)}$$

where N_1 is any odd delay.

The scaling function $\phi(t)$ and $\psi(t)$ wavelet function are related to f_0 and f_1 via (2.9.1) and (2.9.2). Dual scaling function $\phi'(t)$ and dual wavelet function $\psi'(t)$ are related to h_0 and h_1 via similar equations. The wavelet function is said to have vanishing moments of degree N if

$$\int t^k \psi(t)dt = 0,$$ for k=0,1,2,…,N…………………………………………..(2.9.5)

This equation can be transferred to discrete moments of f_1, where the k[th] moment of f_1 is defined as

$$m_1(k) = \sum_n n^k f_1(n)$$ ………………………………………………..……...(2.9.6)

Requiring the moments of $\psi(t)$ to be zero is equivalent to putting the discrete moments of f_1 to zero. For a more general multi-resolution formulation, consider an M-band uniformly decimated filter bank shown in Figure above to which the sampled version $(a_0(n))$ of the continuous time input signal $a(t)$ is applied as input.

$$\phi(t) = \sum_n f_0(n)\sqrt{M\phi(Mt-n)}, \forall n \in Z$$ ………………………………..……(2.9.7)

$$\psi_i(t) = \sum_n^n f_i(n)\sqrt{M\phi(Mt-n)}, \forall n \in Z$$ for i=1,2,3,…,M-1………..…..(2.9.8)

Here, f_0 is the synthesis lowpass filter, f_1 to f_{M-2} are synthesis bandpass filters, and f_{M-1} is the synthesis highpass filter. Unlike the M=2 case, there are M-1 wavelets associated with the scaling function governed by above mother wavelet equation. However, just as for the M=2 case, the multiplicity M scaling function and scaling coefficients are unique and are the solution of basic recursive equation defined in scaling function , and we can have multi-resolution approximation associated with the M-band scaling and wavelet functions. There are M-1 signal spaces spanned by the M-1 wavelets at each scale j.

2.10 BRIEF OVERVIEW OF THE THEORY OF SELF SIMILAR PROCESSES

A continuous-time random process is called self similar if its statistical properties are scale invariant. Symbolically, it is represented as

$$X(ct) \approx c^H X(t) \quad \text{..........(2.10.1)}$$

where the random process x(t) is self similar with self similarity index H (also called the Hurst exponent) for any scale parameter c > 0 . The equality in (2.1.14) holds in the statistical sense only. If, in addition to this, the process has stationary increments, it isdenoted H-sssi.

2.11 FRACTIONAL BROWNIAN MOTION:

An (H-sssi) Gaussian process x(t) with $0 < H < 1$ is called fractional Brownian motion (FBm) and is denoted as $B_H(t)$. For the value H=1/2 , the resulting process is the well-known Wiener process. Although an FBm process is a nonstationary process, Flandrin has shown, using time-frequency representation, that the averaged PSD of this process follows a power law and is directly proportional to $1/|f|^\beta$ with $\beta = 2H + 1$, where f is the frequency. Therefore, in general, these processes are also called $1/f^\beta$ processes. FBm has a generalized derivative and is termed fractional Gaussian noise (FGn). Corresponding to a discrete data set, discrete FBm is defined as

$$B_H[k] = B_H[kT_s] \quad \text{..........(2.11.1)}$$

Where T_s is the sampling period. Since the process is self-similar for any value of c > 0, therefore, can be taken to be equal to one without loss of generality. The mean value, variance, and autocorrelation function of the discrete Gaussian process $1/f^\beta$ are given by

$$E\{B_H[k]\} = 0 \quad \text{..........(2.11.2)}$$

$$\mathrm{var}\{B_H[k]\} = k^{2H}\sigma_H^2 \; \Gamma_{BH}(k_1, k_2) = \frac{1}{2}\sigma_H^2 (|k_1|^{2H} - |k_1 - k_2|^{2H} + |k_2|^{2H}) \quad \text{……………………………….(2.11.3)}$$

Where $\sigma_H^2 = \mathrm{var}\{B_H(1)\} = 1/\Gamma(2H+1)|\sin(H)|$, i.e., it is a zero mean, self similar, non stationary random process. Next, discrete FGn can be defined as

$$X_H[k] = B_H[k] - B_H[k-1] \quad \text{……………………………………….(2.11.4)}$$

2.12 M^th ORDER FRACTIONAL BROWNIAN MOTION (m-FBm):

FBm with 0 < H < 1 is called the 1-FBm, and the corresponding firstorder incremental process is called the 1-FGn. Similarly, the m-FBm process is denoted $B_{H,m}(t)$ with m-1 < H < m , and the corresponding mth-order incremental process is defined as m-FGn process. It is given as

$$X_{H,m}(t) = \Delta_l^{(m)} B_{H,m}(t) = \sum_{j=0}^{m} (-1)^{m-j} \binom{m}{j} B_{H,m}(t + jl) \quad \text{…………………(2.12.1)}$$

Where l is a real number, and is called a lag, and m is an integer

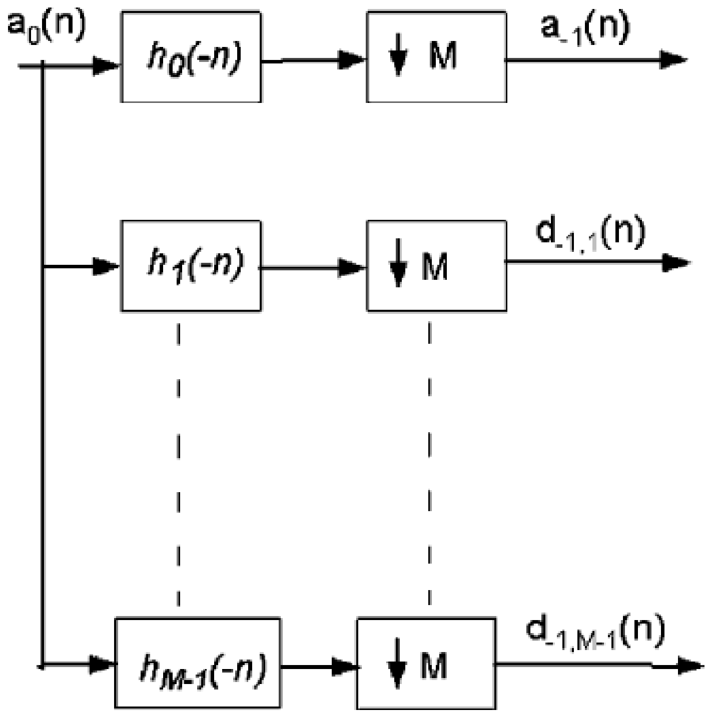


Fig:17 Analysis end of M-band wavelet system.

2.13 ESTIMATION OF H PARAMETER

The maximum likelihood estimation method presented in "Fractional Brownian motion: A maximum likelihood estimator and its application to image texture," can be used to estimate parameter H. In "Fractional Brownian motion: A maximum likelihood estimator and its application to image texture," the method is presented for a process with 0 < H < 1 that can be easily extended to m-FBm processes. If the input process is m-FBm, then its mth-order incremental process will be an m-FGn stationary process. Since it is stationary, maximum likelihood (ML) estimation is performed using a discrete m-FGn vector X and is denoted $\hat{H}$:

$$\hat{H} = \max_{m-1<H<m} (-N \log \frac{X^T R_X^{-1} X}{N} - \log |R_X|) \quad \text{.......................(2.13.1)}$$

Where R_X is the autocorrelation matrix of a discrete m-FGn process.

2.14 ESTIMATION OF STATISTICALLY MATCHED HIGHPASS WAVELET FILTER OF ANALYSIS FILTERBANK

Consider an analysis filterbank structure in above figure of the M-band wavelet system to which the sampled version of given continuous time signal a(t) is applied as input, i.e., $a_{0(n)} = a(n)$ sampled version of the input signal or approximation coefficients of the signal at scale j=0 . Here, h_0 is the lowpass filter, $h_1, h_2 ..., h_{M-2}$ are bandpass filters, and h_{M-1} is the highpass filter such that $a_{-1}(n)$ represents the approximation coefficients at scale j = -1 , and $d_{-1,1}(n), d_{-1,2}(n), ..., d_{-1,M-1}(n)$ represents the finer information in wavelet subspaces at scale j = -1 . Let us assume that the length of filter h_{M-1} is N=5 ; then, $d_{-1,M-1}(n)$ can be written in terms of filter weights as

$$d_{-1,M-1}(n) = h_{M-1}(0)a_0(Mn) + h_{M-1}(1)a_0(Mn+1) + h_{M-1}(2)a_0(Mn+2) + h_{M-1}(3)a_0(Mn+3) + h_{M-1}(4)a_0(Mn+4.) \quad \text{..........(2.14.1)}$$

The signal $d_{-1,M-1}(n)$ provides the detail or highpass information. Therefore, we would like to express this signal as smoothening error signal. Now if the center weight $h_{M-1}(2)$ of the highpass filter h_{M-1} is set to unity, then the above equation is rewritten as

$$d_{-1,M-1}(n) = a_0(Mn+2) - \{-[h_{M-1}(0)a_0(Mn) + h_{M-1}(1) \times a_0(Mn+1) + h_{M-1}(3)a_0(Mn+3) + h_{M-1}(4)a_0(Mn+4)]\} \quad \text{.........(2.14.2)}$$

$$= a_0(Mn+2) - \hat{a}_0(Mn+2) = e(n) \quad \text{.....................(2.14.3)}$$

Where

$$\hat{a}_0(Mn+2) = -[h_{M-1}(0)a_0(Mn) + h_{M-1}(1)a_0(Mn+1) + h_{M-1}(3)a_0(Mn+3) + h_{M-1}(4)a_0(Mn+4)] \quad \text{............(2.14.4)}$$

The interpretation of the above equation is as below : in fact , is the central idea of the present work. This equation has been put in the above form to derive an interesting interpolation for the same. This play a key role in the estimation of the matched wavelet . With the centre weight fixed to unity, from above equation. $\hat{a}_0(Mn+2)$ is the smoother estimate of $a_0(Mn+2)$ from the past as well as from future samples. Thus $d_{-1,M-1}(n)$ is the error in estimating $a_0(Mn+2)$ from its neighbourhood and, hence, represents additional finer information. This idea to estimate an analysis wavelet filter is similar to a sharpening filter used in image enhancement. Since $d_{-1,M-1}(n)$ represents error signal between the actual value of $a_0(Mn+2)$ and its estimated value $\hat{a}_0(Mn+2)$, we should minimize the mean square value of this error signal. Here , the resulting filter h_{M-1} is observed to be a high pass filter , which is in conformity with the result of the sharpening filter in image enhancement . from above equation this can also be represented as follows:

$$d_{-1,M-1}(n) = e(n) = a_0(Mn+J_1) - W_0^T A_0 \quad \text{……………………………(2.14.5)}$$

Where J_1 =index of centre weight of filter h_{M-1} and N= length of dual wavelet filter h_{M-1}

$$A_0 = -[a_0(Mn)a_0(Mn+1)....a_0(Mn+J_1-1)a_0(Mn+J_1+1)....\\ ....a_0(Mn+N-1)]^T \quad \text{…………..……(2.14.6)}$$

$$W_0 = [h_{M-1}(0)h_{M-1}(1)....h_{M-1}(J_1-1)h_{M-1}(J_1+1)....h_{M-1}(N-1)]^T \quad \text{……………...(2.14.7)}$$

$$E[e^2(n)] = E[a_0^2(Mn+J_1)] - 2E[a_0(Mn+J_1)W_0^T A_0] + E[W_0^T A_0 A_0^T W_0] \quad \text{……………(2.14.8)}$$

To minimize $E[e^2(n)]$, the derivation of $E[e^2(n)]$ with respect to W_0 is equal to zero.

$$\frac{\partial E[e^2(n)]}{\partial W_0} = -2E[a_0(Mn + J_1)A_0^T] + 2R_0W_0 = 0 \quad \text{..............(2.14.9)}$$

$$\Rightarrow E[a_0(Mn + J_1)A_0^T] = R_0W_0 \quad \text{...(2.14.10)}$$

Therefore, if statistics of the input signal are known ,then using above filter equation h_{M-1} can be computed. The wavelet structure is ideally suited for self-similar or, say, $1/f^{\beta}$ processes, and the wavelet basis acts like a K-L type basis for $1/f^{\beta}$ processes. Therefore, consider input signal a(t) as a self similar process with self similarity index H.

2.14.1 Algorithm 1

The algorithm to estimate statistically matched highpass analysis wavelet filter is explained below:

Step 1: First, find the self-similarity index H for a given input signal by the ML estimation method presented in Fractional Brownian motion. The procedure is

i) Form the mth-order incremental process (i.e., discrete m-FGN) from the given input signal starting from . Compute the autocorrelation matrix of the resulting m-FGn process with (2.14.11):

$$r^m_{(H,m)}(k) = \frac{\sigma_H^2}{2}(-1)^m \sum_{j=-m}^{m} (-1)^j \binom{2m}{m+j} |k+j|^{2H} \quad \text{.............(2.14.11)}$$

ii) Next, plot the graph of bracketed term for various values of H. If the graph is convex upward, the value of H corresponding to maxima in the graph is the correct value of H.

iii) If the graph is linear, increment m, and repeat steps i) and ii).

Step 2) Compute the autocorrelation matrix R of $a_0(Mn)$ with (2.14.11) for a fixed length N of the analysis wavelet filter.

$$r^{H}_{B,m}(Mn_1, Mn_2) = M^{2H-1}\sigma^{m}_{H}(-1)^{m}\{|n_1 - n_2|^{2H}$$
$$-\sum_{j=0}^{m-1}(-1)^{j}\binom{2H}{j}\left[\left(\frac{n_1}{n_2}\right)^{j}|n_2|^{2H} + \left(\frac{n_2}{n_1}\right)^{j}|n_1|^{2H}\right] \quad \text{.........(2.14.12)}$$

where $\sigma^{m}_{H} = \sigma^{2}_{H}$

Step 3) Estimate the analysis wavelet filter using (2.14.10) for the sufficiently high value of time index n. The resulting filter is the highpass analysis wavelet filter.

2.15 DESIGN OF FIR PERFECT RECONSTRUCTION BIORTHOGONAL FILTERBANK

The four filters h_0, h_1, f_0, f_1 of the two-band perfect reconstruction biorthogonal filterbank structure are related by (2.9.3)and (2.9.4).Here , all the filters are FIR filters. First,the highpass analysis wavelet filter h_1 is estimated as mentioned in above section.

Now compute the scaling filter f_0. Since the integer translates of $\phi(t)$ and $\psi(t)$ form the basis of V_0 and W_0 respectively, in L_2 , $f_0(2m-n)$ and $f_1(2m-n)$ form the basis of l^2 for integer values of m. similarly, $h_0(n-2m)$ and $h_1(n-2m)$ form the dual basis of l^2 for integer value of m. Therefore

$$\sum_{n} h_0(n-2m) f_0(n-2m_2) = \delta(m_1 - m_2), \forall m_1, m_2 \in Z \quad \text{..............(2.15.1)}$$

And $\sum_{n} h_0(n)h_1(n) = 0$ ....(2.15.2)

2.15.1 Algorithm 2

The complete algorithm to estimate a two-band compactly supported statistically matched wavelet with desired support and a desired number of vanishing moments from a given signal is as follows:

Step 1–3: Estimate the statistically matched analysis wavelet filter h_1 of order N_1 from a given input signal using steps 1 to 3 of above algorithm.

Step 4: If it is desired to design wavelet filter f_1 of order $N_2 > N_1$, then append extra zeros before and after h_1 such that its order is N_2.

Step 5: Use (2.9.4) to compute the synthesis scaling filter f_0.

Step 6: Use (2.15.1)and (2.15.2) and (2.9.5) and (2.9.14) to compute the analysis scaling filter h_0.

Step 7: Use (2.9.4) to compute the synthesis wavelet filter f_1.

Step 8: Design the scaling and wavelet functions from the scaling and wavelet filter using 2–scale recursive relations (2.9.1) and (2.9.2).

2.16 DESIGN OF STATISTICALLY MATCHED SEMIORTHOGONAL 2-BAND WAVELET SYSTEM

$$h_1(n) = (-1)^n h_0(N_1 - n) \quad \text{...(2.16.1)}$$

Where N_1 is odd delay.

$$E(z) = U(z)\Lambda(z)V(z) \quad \text{...(2.16.2)}$$

Where U(z) and V(z) are unimodular matrices, and $\Lambda(z)$ is a diagonal matrix.

Algorithm 3

Thus, the complete algorithm to estimate a semiorthogonal two-band statistically matched wavelet from a given signal is as follows.

Step 1–3: Estimate the statistically matched analysis wavelet filter h_1 from a given input signal using Steps 1 to 3 of above section.

Step 4: Use (2.16.1) to compute analysis scaling filter h_0 .

Step 5: Form the polyphase decomposition matrix E(z) from analysis filters h_0 and h_1. Carry out the Smith-McMillan form decomposition of as in (2.16.2). Find R(z) and using R(z) and $\det(\wedge(z))$, compute the synthesis filters and, hence, design the structure of the PR filterbank.

Step 6: Design the scaling and wavelet functions from the synthesis scaling and wavelet filter.

The resulting wavelet corresponding to the highpass synthesis filter is usually infinitely supported. However, a subclass of these wavelets have finite support when $\det(\wedge(z))$ is a monomial and results in a compactly supported wavelet.

2.17 Expected results of biorthogonal wavelet for a music clip is

Input as a music clip

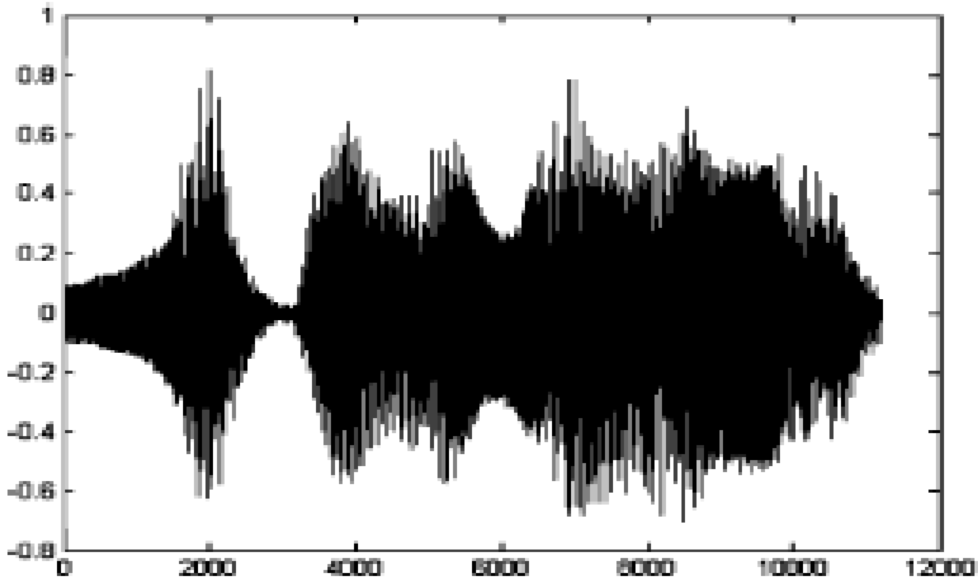


Fig 18

Scaling function

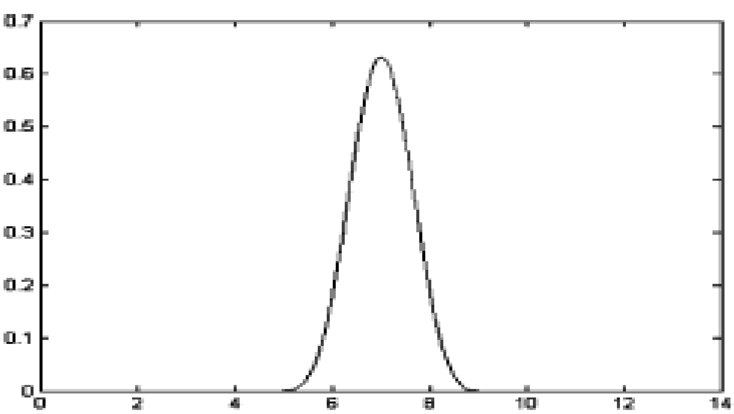


Fig 19

Wavelet function

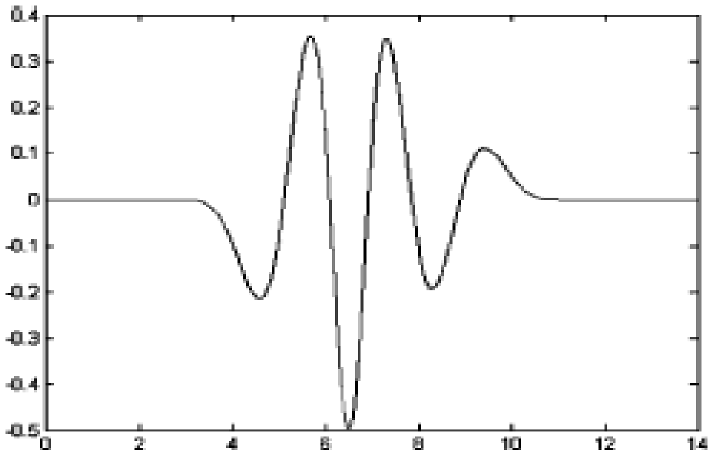


Fig 20

2.18 For semiorthogonal

Scaling function is

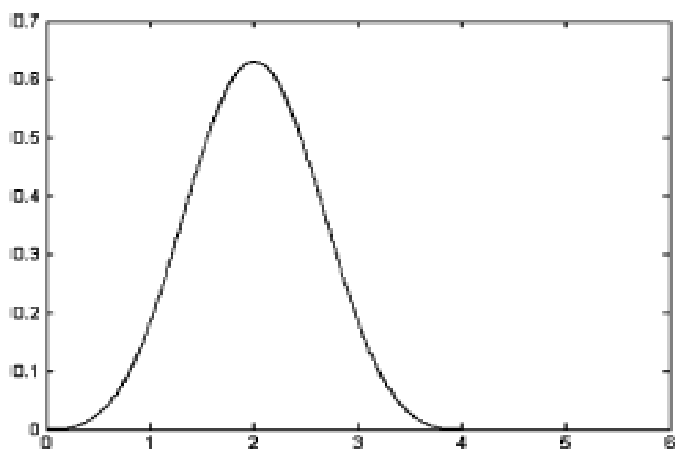


Fig 21

Wavelet function

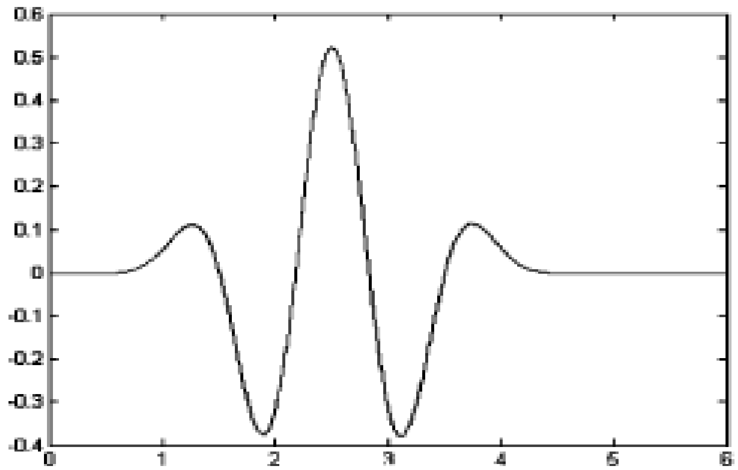


Fig 22

CHAPTER-3
LITERATURE SURVEY ON FRACTIONAL DELAY FILTER

3.1 INTRODUCTION TO FRACTIONAL DELAY FILTERS

A fractional delay filter is a device for band-limited interpolation between samples. It finds applications in numerous fields of signal processing, including communications, array processing, speech processing, and music technology and time-delay estimations, not only the sampling frequency but the actual sampling instants are of crucial importance. a comprehensive study of FIR and all-pass filter design techniques for band-limited approximation of a fractional digital delay. The sampling rate must satisfy the Nyquist criteria in order for a sample set to represent adequately the original continuous signal. The appropriate sampling rate is alone is not sufficient for many applications also the sampling instants must be properly selected.

Fractional delay filter design is used to approximate the delay filter $\exp^{(-j\omega D)}$ with a delay ***D*** for the full band $|\omega| < \pi$, using FIR filters or IIR allpass filters. Fractional delay filter design is necessary only when sampling is critical, i.e. Nyquist sampling. When the sampling rate is higher than the Nyquist rate, the ideal delay filter $\exp^{(-j\omega D)}$ for the base band $|\omega| < \frac{\pi}{r}$ only needs to be approximated, where *r* is the ratio of sampling rate over the Nyquist rate.

Fractional delay means, assuming uniform sampling, a delay that is a non integer multiple of the sample interval. Fractional delay filters are those filters which exhibit near unity magnitude response and a flat group-delay which is not necessarily an integer multiple of the sampling interval. FIR FD filters are discrete-time interpolators which approximate the signal in between sample points as a linear combination of sample values on either side of the desired signal value.

Designing FD filters involves determining the coefficients of an FIR filter such that its response best approximates the complex valued frequency response of the desired FD. One fundamental advantage of digital signal processing techniques over traditional analog methods is the easy implementation of a constant delay; the signal samples are simply stored in a buffer memory for the given time.

3.2 COMPARISON OF FIR AND IIR FILTER DESIGN

We can compare the above fractional delay filters in terms of their frequency response error (FRE) magnitude**.** The useful range of delay D is different for FIR and allpass filters. In the case of FIR filters, the best approximation is obtained when interpolating between the middle taps in the case of odd-length ones. The squared approximation error function are symmetric about the midpoint of the FIR filter. In the case of allpass filters, the error curves are asymmetric. The stability of allpass filters must be taken into account. Fractional delay filters yields the best approximation when the total delay D is close to $\frac{N}{2}$ for FIR filters and close to N for allpass filters.

CONCLUSION: The design of high quality FD filters is difficult if a very small delay is required. one possibility is to use a higher sampling rate if accurate and small fractional delays are needed.

3.3 LAGRANGES INTERPOLATOR

The fractional delay filters should have the following characteristics:

1. Lowpass characteristics with an almost flat magnitude response in the passband
2. Magnitude response less than unity at all frequencies, so as not to cause instability
3. Accurate model of the desired fractional delay
4. Easy and intuitive incorporation into the vocal tract model.

According to valimaki Lagrange’s interpolators are one type of FIR filter that are both easy to implement and have the desire properties listed above. Among IIR filters, thiran allpass filters are also considered suitable since they meet the listed requirements. In this work, only Lagrange's interpolators have been used because being FIR filters it is intuitively easier to understand how they work in a given application. Their design and characteristics are now declared.

While designing a digital filter, the ideal magnitude and frequency responses are always kept in mind. The response of an ideal fractional delay filter was described later in this thesis. If an FIR FD filter is being designed, the general form of an Nth order filter whose length is L=N+1;

$$H(z) = \sum_{n=0}^{N} h(n) z^{-n} \qquad \text{……………………………………………..(3.3.1)}$$

An error function $E(e^{j\omega})$ is defined as the difference between the actual and the ideal filters at a given frequency

$E(\omega)=H(\omega)-H_{id}(\omega)$

Frequency-domain filter design involves minimizing the above error metric according to criteria that lead to the filter design goals being met. It may be useful in certain applications to use a filter with zero error at ω=0. In other situations the squared error integrated over a range of frequencies, may be minimized. Different constraints on the error E(ω) lead to different types of filters.

Lagrange's interpolators belong to a class of filters called maximally flat filters they have a constant magnitude response around a particular frequency of interest. The response of Lagrange's interpolators is made identical to that of the ideal interpolators at zero frequency. The derivatives of the error function E(ω) are set to zero at the frequency of interest:

$$\left.\frac{d^n E(\omega)}{d\omega^n}\right|_{\omega=\omega_0} = 0 \qquad \text{for all n=0,1,2,… N………………………………..........(3.3.2)}$$

The N+1 linear equations that follows above equation ,and can be solved to obtain N+1 coefficients of the FIR filter. The resultant set of equations is of the form shown below, where D is as before, a positive real number representing the desired total delay:

$$\sum_{k=0}^{N} k^n h(k) = D^n \quad \text{or } n=0,1,2,\ldots,N \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \text{(3.3.3)}$$

On solving this equation , a closed form representation of the FIR filter coefficients can be obtained.

$$h_L(n) = \prod_{\substack{k=0 \\ k \neq n}}^{M} \frac{D-k}{n-k}, \quad \text{for} \quad n= 0,1,2,\ldots,M \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \text{(3.3.4)}$$

The ease of computing filter taps is an important feature of Lagrange's interpolators. By virtue of their design criterion, they exhibit a flat magnitude response at low frequencies with no ripples. The magnitude response and group delay characteristics of odd and even-length filtes are shown in result.

For fractional delay of D=0.5 , the point of interpolation is located mid-way between the two center filter taps. The filter impulse response for a third-order filter is shown in result. the filter is perfectly symmetric and the phase is linear in the entire frequency range of the interpolators. This is borne out in the group delay plot. For the values D=0.5 in (0,1) , the odd order filter are not symmetric. For an even length filter , the point of interpolation lies between the two central samples. In such a scenario, the delay characteristics are superior to odd-length interpolators.only odd order Lagrange's interpolators are used in this work. It is also important to analyze the magnitude response of the interpolators. In speech synthesis , the upper value of frequencies that are of interest is about 5 KHz. While the waveguide model being used in this work produces speed output at a sampling rate of 44.1KHz , the spatial resolution of the vocal tract is twice as much. This is because one segment length is 0.397cm m which is equivalent to a sampling rate of 88.2KHz. The interpolation method to be used for length variations can be visualized as “spatial interpolation” where the samples are 1/88200 s apart. The 0 – 5 kHz band thus corresponds to a maximum normalization frequency of about 0.06. even a first order Lagrange's interpolation has a very flat passband up to a normalized frequency of 0.1 , so the use of a sample first order filter is adequate for the highly over-sampled system being used. Linear interpolation is simply a two filter taps, [α, 1-α] where α is the desired fractional delay.

3.4 FRACTIONALLY DELAY FILTER DESIGN BASED ON TRUNCATED LAGRANGE'S INTERPOLATION

A new design method for fractional delay filters based on truncating the impulse response of the Lagrange interpolation filter is presented. The truncated Lagrange fractional delay filter introduces a wider approximation bandwidth than the Lagrange filter. However, because of truncation, a ripple caused by the Gibbs phenomenon appears in the filter's frequency response. Proper choices of filter order and prototype filter order allow adjusting the overshoot to a desired level and simultaneously reducing the overall frequency-response error. The design of the proposed filter is computationally efficient, because it is based on polynomial formulas, which have common terms for all coefficients.

Lagrange's interpolation is a common method in signal processing. It is used for the interpolation of band-limited signals, for instance, in sampling rate conversion and in fractional delay (FD) filters Lagrange interpolation is used to determine the coefficients of a finite impulse response (FIR) filter for a given fractional delayed Such a filter approximately produces a time delay of the Form $(D_{int} + d)T$, where D_{int} is an integer , d is a fractional Number $(0 < T < 1)$, and T is the sampling interval. Harmonics proposed the maximally-flat approximation of the ideal fractional delay and noted that when the point at which the approximation error and its N derivates are set to zero is chosen to be the zero frequency, the solution is equivalent to the Lagrange interpolation. Kootsookos and Williamson discovered that the coefficients of even-order Lagrange FD filters can be obtained from the truncated sinc function using the binomial window.

Välimäki showed that the same is true for odd-order Lagrange FD filters. Lagrange interpolation converges to sinc interpolation as the order N approaches infinity. The implementation cost of variable Lagrange interpolation can be reduced by using a structure based on the Taylor series. the popularity of Lagrange interpolation is due to its easy coefficient update rule, which uses closed-form formulae that are N th-order polynomials, where N is the filter order. Additional advantages of Lagrange FD filters include the accurate approximation of fractional delay at low frequencies and the fact that its magnitude response does not exceed unity. The latter property makes Lagrange FD filters a useful choice for feedback structures, in which it is necessary to restrict the loop gain to ensure stability. The main drawback of Lagrange FD filters is that the approximation bandwidth is narrow, and it is widened slowly as the filter order is increased .A number of coefficients at the beginning and the end of the coefficient vector of a prototype Lagrange FD filter are deleted to reduce the filter length. This way, the bandwidth of approximation can be extended with respect to that of a Lagrange FD filter of the same order. The truncation introduces a ripple in the frequency response, known as the Gibbs phenomenon, which typically appears in FIR filters, whose coefficients are samples of a truncated ideal impulse response. As the order of the prototype filter is increased, the response of the truncated Lagrange FD filter approaches that of a truncated sinc FD filter. The new technique can be interpreted as a hybrid method that combines properties of the Lagrange and the truncated sinc FD filters and allows mixing them in an appropriate proportion.

3.5 NEW DESIGN METHODS AND ITS PROPERTIES:

The new design method is based on discarding an equal number of coefficients from both sides of the impulse response of the Lagrange FD filter, which will be referred to as the prototype filter in the following. The closed-form formula to compute the coefficients of an M th-order Lagrange FD filter, $h_L(n)$, is given as

$$h_L(n) = \prod_{\substack{k=0 \\ k \neq n}}^{M} \frac{D-k}{n-k}, \quad \text{for } n = 0,1,2,\dots, M. \quad \text{……………………………………}(3.5.1)$$

where D is a real number that corresponds to the delay from the beginning (n=0)of the impulse response. An Nth-order truncated Lagrange FD filter is obtained by casting off K_1 coefficient from each end of the prototype filter as

$$h_L^{'}(n) = \begin{cases} 0, when 0 \le n \le K_1 - 1 \\ h_L(n), when K_1 \le n \le N + K_1 \\ 0, when N + K_1 + 1 \le n \le M \end{cases} \qquad \text{…………(3.5.2)}$$

where M > N is the prototype filter order, and K_1 is a positive integer($K_1 > \frac{M}{2}$).

Therefore, the truncated Lagrange interpolator can be represented as

$h_{TL}(n)= h_L(n+K_1)$, for n=0,1,2,…,N.

The explicit formula to compute the coefficients of the truncated Lagrange FD filter of order N is represented as follows

$$h_{TL}(n) = \prod_{\substack{k=0 \\ k \ne n+K_1}}^{M} \frac{D-k}{n+K_1-k} \quad \text{, for n=0,1,2,…,N…………………………….(3.5.3)}$$

In the designs the same number of prototype filter coefficients,K_1 , are deleted symmetrically from the beginning and end of the coefficient vector, that is, $M = N + 2K_1$. Although the truncation operation brings about the Gibbs phenomenon, the overshoot is small compared to the overshoot of the truncated sinc filter of the same order. the frequency-response error (FRE) for the same filters. It is seen that truncating the Lagrange interpolator results in an increased FRE at low frequencies. This is the price to be paid for widening the bandwidth. the Lagrange and truncated sinc FD filters have the largest and the smallest MSE values. The MSE of the truncated Lagrange filter decreases as the prototype filter order is increased. As the prototype filter order becomes larger, the MSE curves converge.

3.5.1 Coefficient update:

In FD filters, the coefficients depend on the delay value, which may change often. If updating the coefficients calls for heavy computations, the filtering algorithm becomes inefficient for high-speed applications. Therefore, it is necessary to find a way to reduce the computational complexity. An Nth-order truncated Lagrange FD filter with prototype filter order M has N + 1 coefficients, each of which is an M th-order polynomial in D. Direct calculation of coefficients along with one step of FIR filtering requires (M+1)(N+1)multiplications and M(N+1)=N additions. A technique to reduce the computational complexity is to use a look-up table, provided that there is enough memory available. Method for decreasing the computations is to implement the truncated Lagrange FD filter using the Farrow structure . Then the number of computations can be reduced to $(N+1)^2$ +M multiplications and N (N+1)+M additions. A modified Farrow structure has been proposed, which yields a further reduction in the number of multipliers. In Farrow structure, $\frac{(N+1)^2}{2+M}$ multiplications and N^2 +M+1 additions are required.

3.5.2 HOW TO CHOOSE M AND N:

The design of the truncated Lagrange filter consists of choosing the values of M and N such that its response best approximates the desired response. The overshoot of the magnitude response and the bandwidth are the main features according to which the design parameters can be determined. the behaviour of the overshoot and normalized bandwidth of the truncated Lagrange FD filter. The normalized bandwidth expresses the frequency at which the magnitude response reaches 3 Db for a given filter order N, as the prototype filter order M becomes larger, the normalized bandwidth becomes wider. The undesirable effect of the enlargement of the prototype filter order is the increase in the overshoot of the filter, which may be chosen to be small enough to be insignificant for parameter values are the smallest M and N that yield the desired response with sufficient accuracy.

3.6 IDEAL SOLUTION:

Assuming that the (real-valued) discrete-time signal represents a band-limited base band signal, the implementation of a constant delay can be considered as an approximation of the ideal discrete-time linear phase allpass filter with unity magnitude and constant group delay of the given value *D*. The corresponding impulse response is obtained via the inverse discrete-time Fourier transform

$$h(n) = \frac{1}{2\pi}\int_{-\pi}^{\pi} H(e^{j\omega})e^{j\omega n}d\omega \text{ for all n} \quad \ldots\ldots(3.6.1)$$

$$H_{id}(e^{j\omega}) = e^{-j\omega D} \quad \ldots\ldots(3.6.2)$$

$$h_{id}(n) = \frac{\sin[\pi(n-D)}{\pi(n-D)} = \sin c(n-D\pi) \text{ for all n} \quad \ldots\ldots(3.6.3)$$

which has the shape of the familiar sinc function defined as

$$\sin c(x) = \frac{\sin(\pi x)}{\pi x} \quad \ldots\ldots(3.6.4)$$

.

When the desired delay *D* assumes an integer value, the impulse response Eq. 03 reduces to a single impulse at *n* = *D*, but for non-integer values of *D* the impulse response is an infinitely long, shifted and sampled version of the sinc function **.** Unfortunately, the ideal impulse response is not only infinitely long but also non causal, which makes it impossible to implement it in real-time applications.

CHAPTER-4

LAGRANGE'S MULTIPLIER

Lagrange multipliers are a very useful technique in multivariable calculus. Lagrange's multipliers are useful in one of the most common problems in calculus is that of finding maxima or minima (in general, "extrema") of a function, but it is often difficult to find a closed form for the function being extremized. Such difficulties often arise when one wishes to maximize or minimize a function subject to fixed outside conditions or constraints. The method of Lagrange multipliers is a powerful tool for solving this class of problems without the need to explicitly solve the conditions and use them to eliminate extra variables. Lagrange multipliers are useful when some of the variables in the simplest description of a problem are made redundant by the constraints.

A example : the "**milkmaid problem**"

To give a specific, intuitive illustration of this kind of problem, we will consider a classic example which I believe is known as the "Milkmaid problem". It can be phrased as follows:

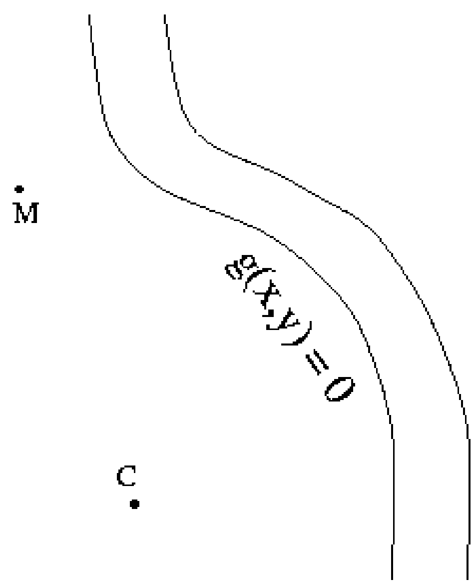


Fig 23

It's milking time at the farm, and the milkmaid has been sent to the field to get the day's milk. She's in a hurry to get back for a date with a handsome young goatherd, so she wants to finish her job as quickly as possible. However, before she can gather the milk , she has to rinse out her bucket in the nearby river. Just when she reaches point **M**, our heroine spots the cow, way down at point **C**. Because she is in a hurry, she wants to take the shortest possible path from where she is to the river and then to the cow. If the near bank of the river is a curve satisfying the function g(x, y) = 0, what is the shortest path for the milkmaid to take?

To put this into more mathematical terms, the milkmaid wants to find the point **P** for which the distance d(**M**,**P**) from **M** to **P** plus the distance d(**P**,**C**) from **P** to **C** is a minimum (we assume that the field is flat, so a straight line is the shortest distance between two points). It's not quite this simple, however: if that's the whole problem, then we could just choose **P** = **M** (or **P** = **C**, or for that matter **P** anywhere on the line between **M** and **C**): we have to impose the constraint that **P** is a point on the riverbank. Formally, we must minimize the function f(**P**) = d(**M**,**P**) + d(**P**, **C**), subject to the constraint that g(**P**) = 0.

4.1 Graphical inspiration for the method

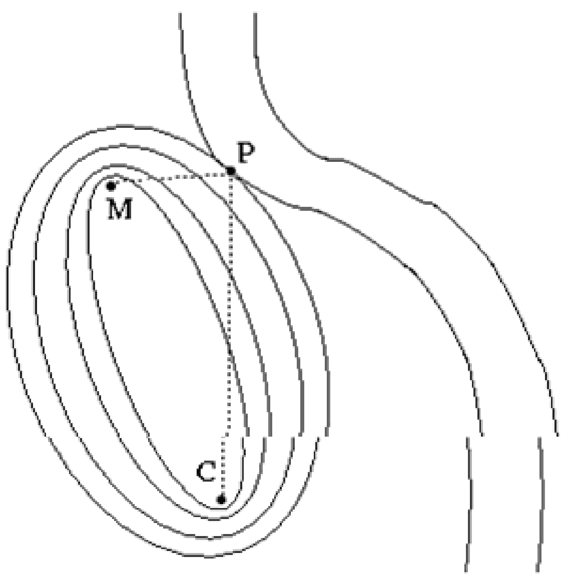


Fig 24

Our first way of thinking about this problem can be obtained directly from the picture itself. We'll use an obscure fact from geometry: for every point **P** on a given ellipse, the total distance from one focus of the ellipse to **P** and then to the other focus is exactly the same. In our problem, that means that the milkmaid could get to the cow by way of any point on a given ellipse in the same amount of time: the ellipses are curves of constant f(**P**). Therefore, to find the desired point **P** on the riverbank, we must simply find the smallest ellipse that intersects the curve of the river. Just to be clear, only the "constant f(**P**)" property is really important; the fact that these curves are ellipses is just a lucky convenience (ellipses are easy to draw). The same idea will work no matter what shape the curves happen to be. The image at right shows a sequence of ellipses of larger and larger size whose foci are **M** and **C**, ending with the one that is just tangent to the riverbank. This is a very significant word! It is obvious from the picture that the "perfect" ellipse and the river are truly tangential to each other at the ideal point **P**. More mathematically, this means that the normal vector to the ellipse is in the same direction as the normal vector to the riverbank. A few minutes' thought about pictures like this will convince you that this fact is not specific to this problem: it is a general property whenever you have constraints. And that is the insight that leads us to the method of Lagrange multipliers.

4.2 The mathematics of Lagrange multipliers

In multivariable calculus, the gradient of a function h is a normal vector to a curve (in two dimensions) or a surface (in higher dimensions) on which h is constant:
n = **grad**(h(**P**)). The length of the normal vector doesn't matter: any constant multiple of **grad**(h(**P**)) is also a normal vector. In our case, we have two functions whose normal vectors are parallel, so

grad(f(**P**)) = λ **grad**(g(**P**))…………………………………………………….(4.2.1)

The unknown constant multiplier λ is necessary because the magnitudes of the two gradients may be different. (Remember, all we know is that their directions are the same.) In D dimensions, we now have D+1 equations in D+1 unknowns. D of the unknowns are the coordinates of **P** (e.g. x, y, and z for D = 3), and the other is the new unknown constant λ. The equation for the gradients derived above is a vector equation, so it provides D equations of constraint. I once got stuck on an exam at this point: don't let it happen to you! The original constraint equation g(**P**) = 0 is the final equation in the system.

Thus, in general, a unique solution exists. As in many maximum/minimum problems, cases do exist with multiple solutions. There can even be an infinite number of solutions if the constraints are particularly degenerate: imagine if the milkmaid and the cow were both already standing right at the bank of a straight river, for example. In many cases, the actual value of the Lagrange multiplier isn't interesting, but there are some situations in which it can give useful information (as discussed below).

That's it: that's all there is to Lagrange multipliers. Just set the gradient of the function you want to extremize equal to the gradient of the constraint function. You will get a vector's worth of (algebraic) equations, and together with the original constraint equation they determine the solution.

A formal mathematical inspiration

There is another way to think of Lagrange multipliers that may be more helpful in some situations and that can provide a better way to remember the details of the technique (particularly with multiple constraints as described below). Once again, we start with a function f(**P**) that we wish to extremize, subject to the condition that g(**P**) = 0. Now, the usual way in which we extremize a function in multivariable calculus is to set **grad**(f(**P**)) = 0. How can we put this condition together with the constraint that we have?

One answer is to add a new variable λ to the problem, and to define a new function to extremize:

$$F(\mathbf{P}, \lambda) = f(\mathbf{P}) - \lambda\, g(\mathbf{P}) \qquad (4.2.2)$$

(Some references call this F "the Lagrangian function". I am not familiar with that usage, although it must be related to the somewhat similar "Lagrangian" used in advanced physics.)

We next set **grad**(F(**P**, λ)) = 0, but keep in mind that the gradient is now D + 1 dimensional: one of its components is a partial derivative with respect to λ. If you set this new component of the gradient equal to zero, you get the constraint equation g(**P**) = 0. Meanwhile, the old components of the gradient treat λ as a constant, so it just pulls through. Thus, the other D equations are precisely the D equations found in the graphical approach above. As presented here, this is just a trick to help you reconstruct the equations you need. However, for those who go on to use Lagrange multipliers in the calculus of variations, this is generally the most useful approach. I suspect that it is in fact very fundamental; my comments about the meaning of the multiplier below are a step toward exploring it in more depth, but I have never spent the time to work out the details.

Several constraints at once

If you have more than one constraint, all you need to do is to replace the right hand side of the equation with the sum of the gradients of each constraint function, each with its own (different!) Lagrange multiplier . This is usually only relevant in at least three dimensions (since two constraints in two dimensions generally intersect at isolated points). Again, it is easy to understand this graphically. Consider the example shown at right: the solution is constrained to lie on the brown plane (as an equation, "g(**P**) = 0") and also to lie on the purple ellipsoid ("h(**P**) = 0"). For both to be true, the solution must lie on the black ellipse where the two intersect. I have drawn several normal vectors to each constraint surface along the intersection. The important observation is that both normal vectors are perpendicular to the intersection curve at each point. In fact, any vector perpendicular to it can be written as a linear combination of the two normal vectors. (Assuming the two are linearly independent! If not, the two constraints may already give a specific solution: in our example, this would happen if the plane constraint was exactly tangent to the ellipsoid constraint at a single point.) The significance of this becomes clear when we consider a three dimensional analogue of the milkmaid problem. The pink ellipsoids at right all have the same two foci (which are faintly visible as black dots in the middle), and represent surfaces of constant total distance for travel from one focus to the surface and back to the other. As in two dimensions, the optimal ellipsoid is tangent to the constraint curve, and consequently its normal vector is perpendicular to the combined constraint (as shown). Thus, the normal vector can be written as a linear combination of the normal vectors of the two constraint surfaces. In equations, this statement reads

grad(f(**P**)) = λ **grad**(g(**P**)) + μ **grad**(h(**P**))……………………………………(4.2.3)

just as described above. The generalization to more constraints and higher dimensions is exactly the same.

4.3 The meaning of the multiplier

As a final note, I'll say a few words about what the Lagrange multiplier "means". In the more formal approach described two sections above, the constraint function g(**P**) can be thought of as "competing" with the desired function f(**P**) to "pull" the point **P** to its minimum or maximum.

The Lagrange multiplier λ can be thought of as a measure of how hard g(**P**) has to pull in order to make those "forces" balance out on the constraint surface. (This generalizes naturally to multiple constraints, which generally "pull" in different directions.) This analogy is inspired by the physics of potential energy.

WORK DONE

IMPLEMENTATION

1. DESIGN OF UNIT DELAY FILTER

2. DESIGN OF FRACTIONAL DELAY FILTER(Lagrange's interpolator (FIR))

3. COMPARISON OF THEIR CHARACTERISTIC

4. DESIGN OF LAGRANGE'S MULTIPLIER

5. DESIGN OF WAVELETS(orthogonal)

6. COMPARISON OF RESULTS USING FRACTIONALLY DELAYED FILTR

   AND UNIT DELAY FILTER.

CONCLUSION AND FUTURE SCOPE

To conclude this dissertation gives us the detailed knowledge of key issues in the field of communication named " Design of Matched Wavelet using Fractionally delayed filter". We introduced the theory and literature survey behind design of matched wavelet and design of fractionally delayed filter and discuss the basic design of matched wavelet and fractionally delayed filter, properties of matched wavelet, performance of matched and even properties of fractionally delayed filter , performance of fractionally delayed filter in various fields of their applications. We identified some factors that could result in the design of matched wavelet using fractional delay filters not performing to its potential. These factors includes randomness of signals, statistical signals etc., and the noise effect and issues of implementing them is crucial for proper functionality. We have discussed and reserved some for our discussion latter pursuits and we hope to carry that in our next work.

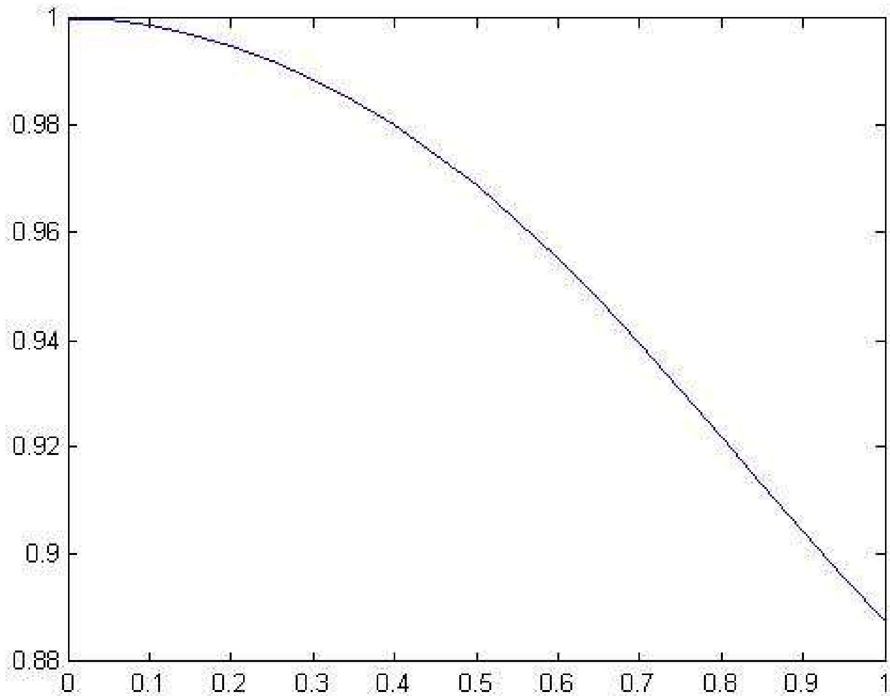
In this dissertation we focused our attention on the design of matched wavelet using fractionally delayed filter, using Lagrange's multiplier to get random ness of signal information and for non stationary signals. We used Lagrange's interpolator filter (FIR) and designed it for fractional delay, so our aim is to fine tune the signal and improve the performance of matched wavelet on signals. Here after discussion and the result we got , we can conclude that the matched wavelet designed by fractionally delayed filters are better than matched wavelets designed by unit delay filters.

To support my work, I have simulated the entire work on MATLAB 7.0.1. At this stage my work should be considered as a preliminary as it has plenty of scope for future investigation and analysis. Major work can be carry in the field of image processing and signal processing.

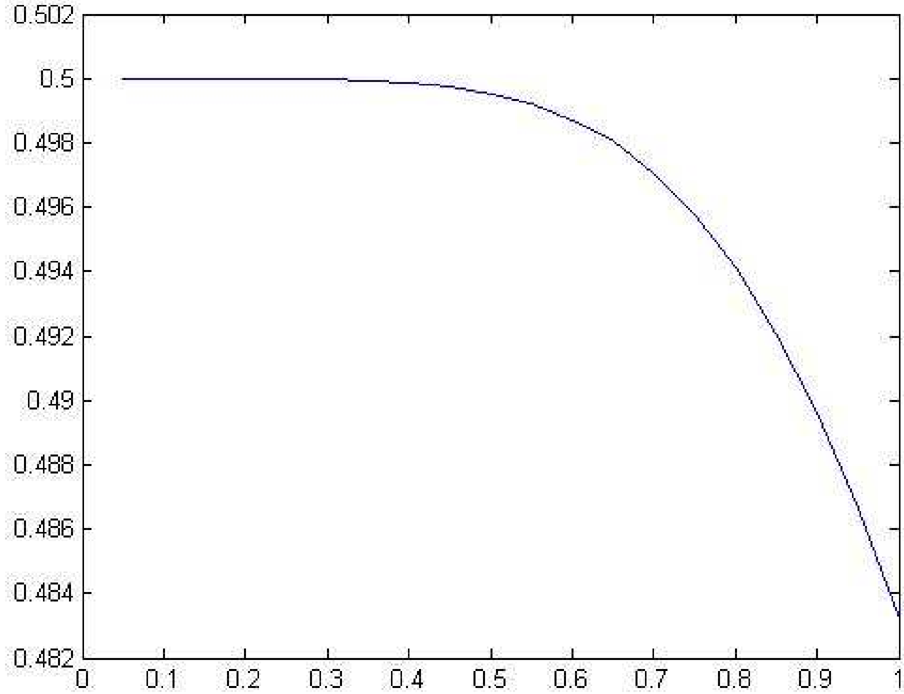
RESULTS

Results of fractional delay filter

Magnitude with normalized frequency

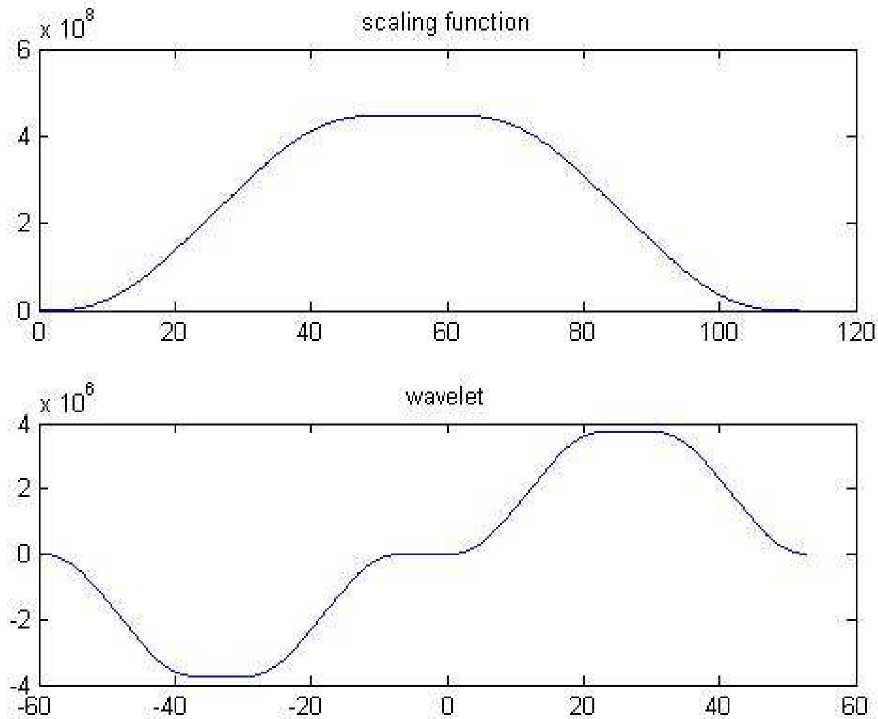


Phase with normalized frequency

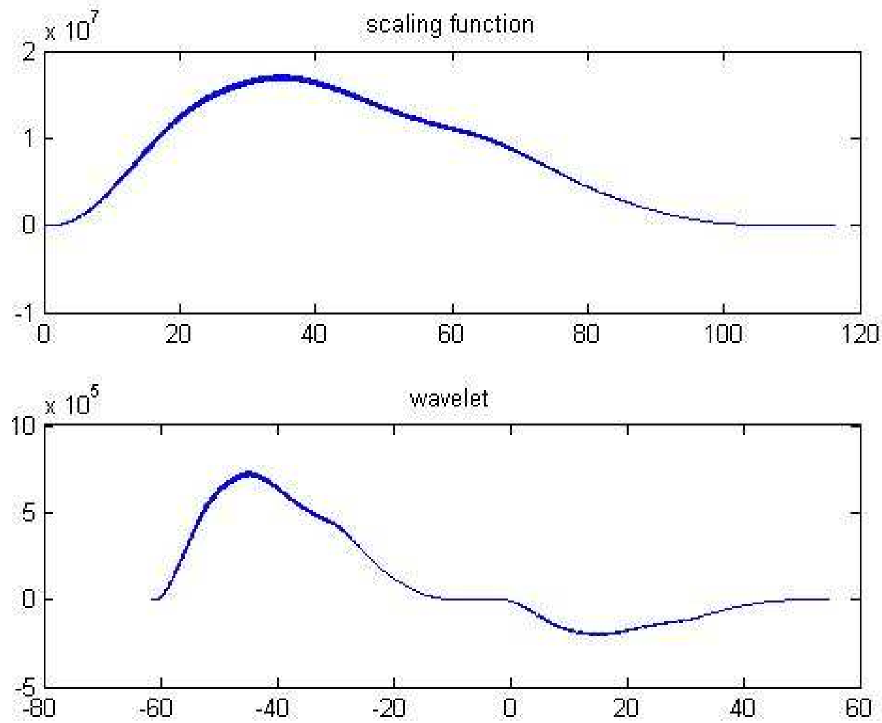


All this results are of orthogonal wavelet.

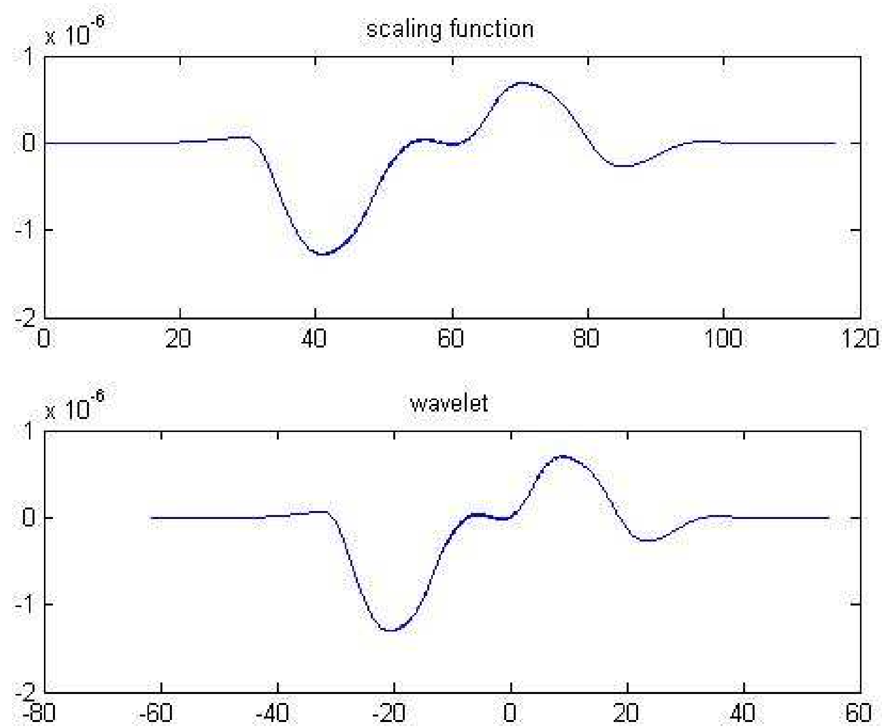
Results for unit delay filter when applied to images.



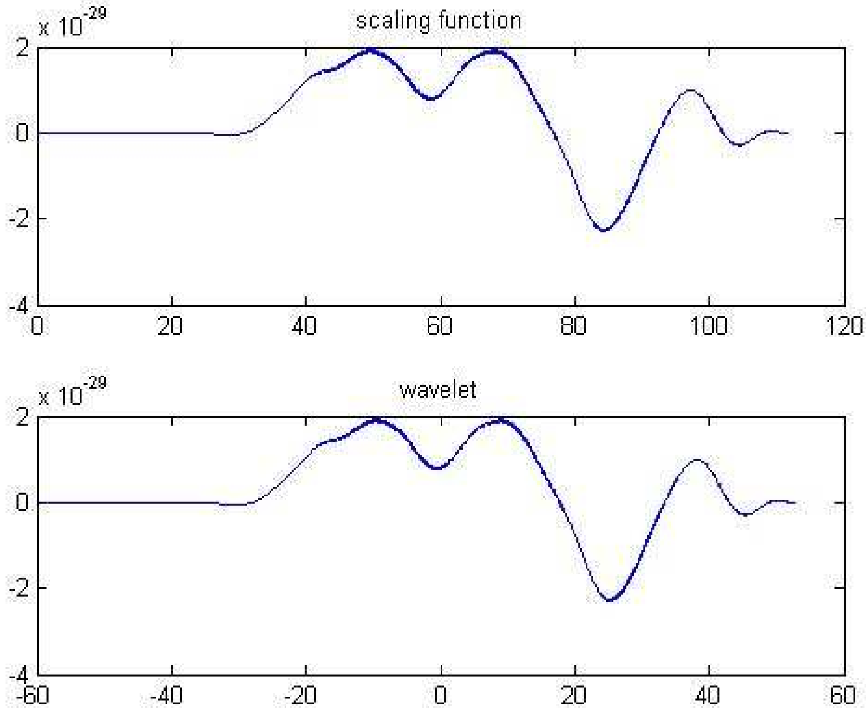
Results of fractional delay filter when applied to images



Results of fractional delay filter when applied to sound signal



Results for unit delay filter when applied to sound signals


scaling function
x 10^-29
2
0
-2
-4
0
20
40
60
80
100
120
wavelet
x 10^-29
2
0
-2
-4
-60
-40
-20
0
20
40
60

Results for the same sentence spoke by three different people

Lamda value for this three tests are

Test1	Test2	Test3
t =	t =	t =
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0.2983
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	1.2190	0
0	0	0
0	0	0
0	0	0.5361
0	0	0
0	0	0
0	0	0.0601
0	0	0
0	0	0
0	0	0
0	0	0.0621
0	0	0
0	0	0
0	0	0.0042
0.1640	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0.0893	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0

0	0	0
0	0	0
0	0	0
0	0	0
0	1.1142	0
0	0	0.1517
0	0	0
0	0	0.6520
0	0	0
0.6584	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0.9929	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0.3819	0	0
0	0	0
0	0.1660	0
0	0	0
0	0	0
0.3729	0.8870	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0.5449
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0.0685
0	0	0
0	0	0.1100
0	0	0
0	0	0
0	0	0
0.2585	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0

0.7333	0	0
0	0.3143	0
0	0	0
0.4907	0	0
0	0.0685	0
0	0	0
0	0	0
0	1.3587	0
0	0	0
0	0	0
0	0	0
0	0	0
0.5625	0	0
0	0.0878	0
0	0	0
0	0	0.5274
0.0664	0.1239	0
0	0	0
0	0	0.4650
0	0	0
0.5455	0.2472	0
0	0.0100	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0.4441	0
0	0	0
0	0	0
0	0	0
0	0	0.2202
0	0	0
0	0	0.3936
0	0	0
0	0.5424	0
0	0	0
0	0.5208	0
0	0	0
0	0	0.2541
0	0	0
0.2187	0	0
0.1789	0	0
0	0	0.3893
0	0	0
0	0	0
0.2406	0.1695	0
0	0	0
0	0	0

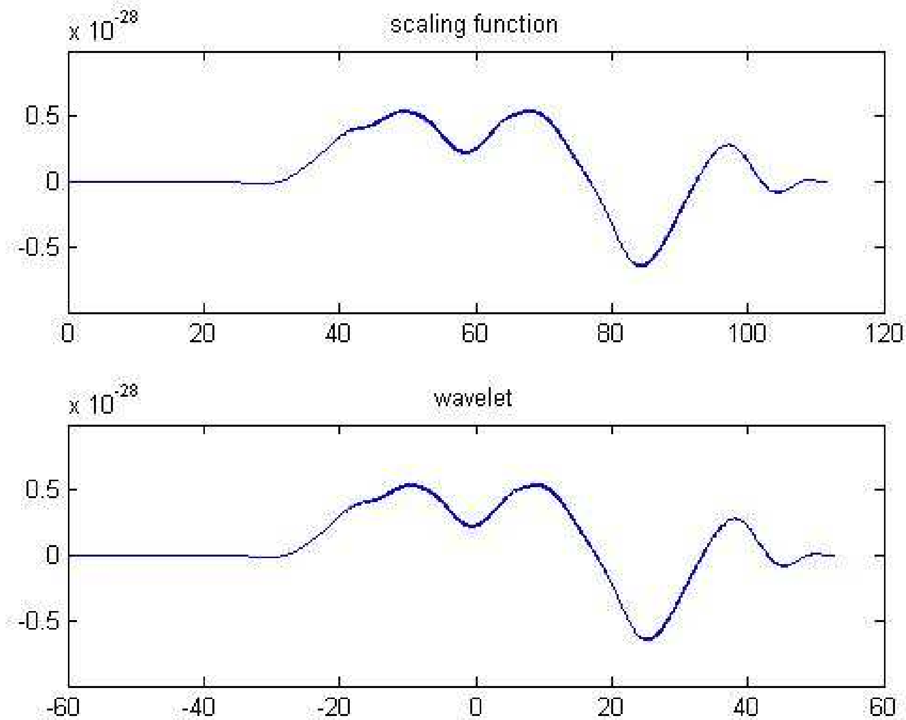
0.0149	0	0
0.5775	0	0
0	0	0
0.2143	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0.5734	0
0.3315	0	0
0	1.2321	0.3135
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0.1487	0.0717	0.2906
0	0	0
0	0	0
0.4870	0	0
0	0	0
0	0	0
0	0.7342	0
0	0	0
0	0	0.3817
0.0033	0	0
0	0	0
0	0	0
0	0	0
0	0	0.1989
0	0	0.4281
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0.2866	0
0	0	0
0	0	0.2145
0	0	0
0	0	0
0	0.5776	0
0	0	0
0	0.3028	0.3334

0	0	0
0	0	0
0	0	0
0	0	0.4086
0	0.0660	0
0	0	0
0	0	0
0.2026	0	0
0	0	0
0	0	0
0	0	0
0	0	1.2552
0.5050	0	0
0	0	0.0709
0	0	0
0	0	0
0	0.4828	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0.2879	0
0	0	1.1200
0.0784	0	0
0	0	0
0	0	0
1.5787	0	0.6734
0	0	0
0	0	0.0040
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0.3569
0.1622	0	0
0	0	0
0	0	0
0	0.8956	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0.4556	0
0	0	0
0	0	0
0	0	0
0	0	0

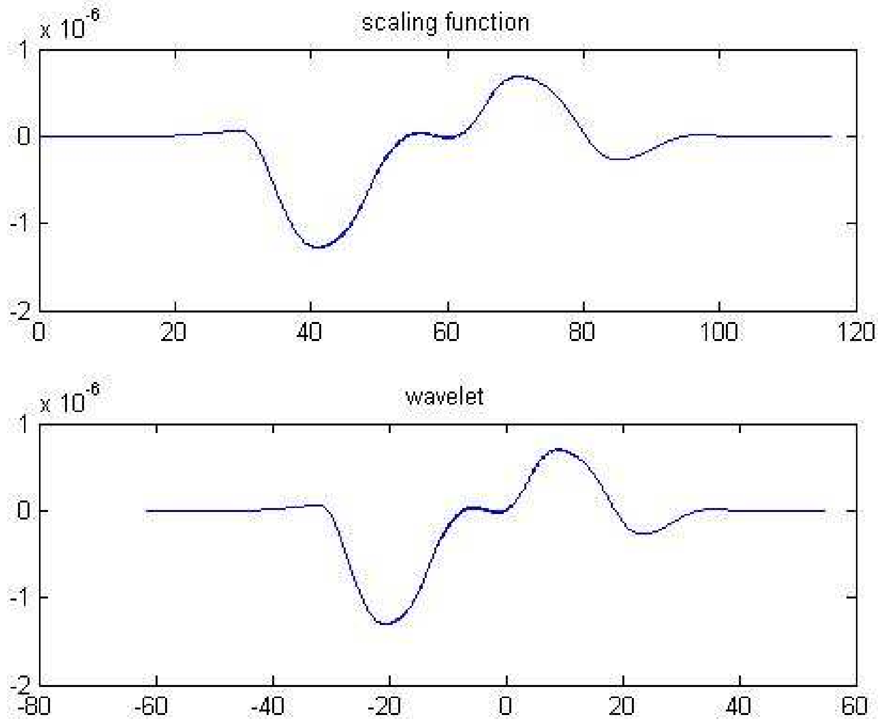
0	0	0

Results for unit delay filter and fractional delay filters are as below

Test1 for unit delay

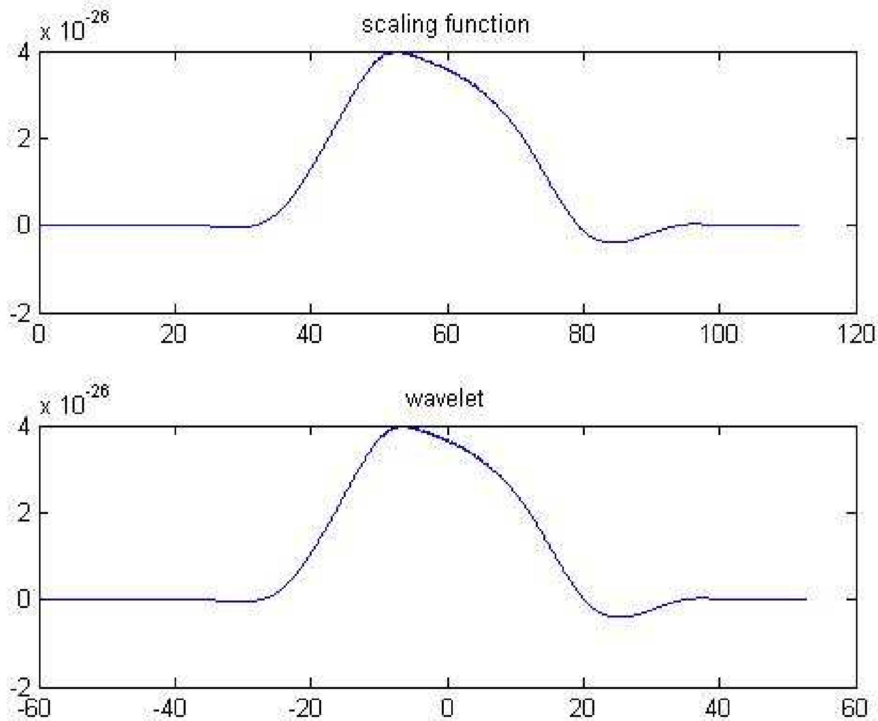


Test1 for fractional delay

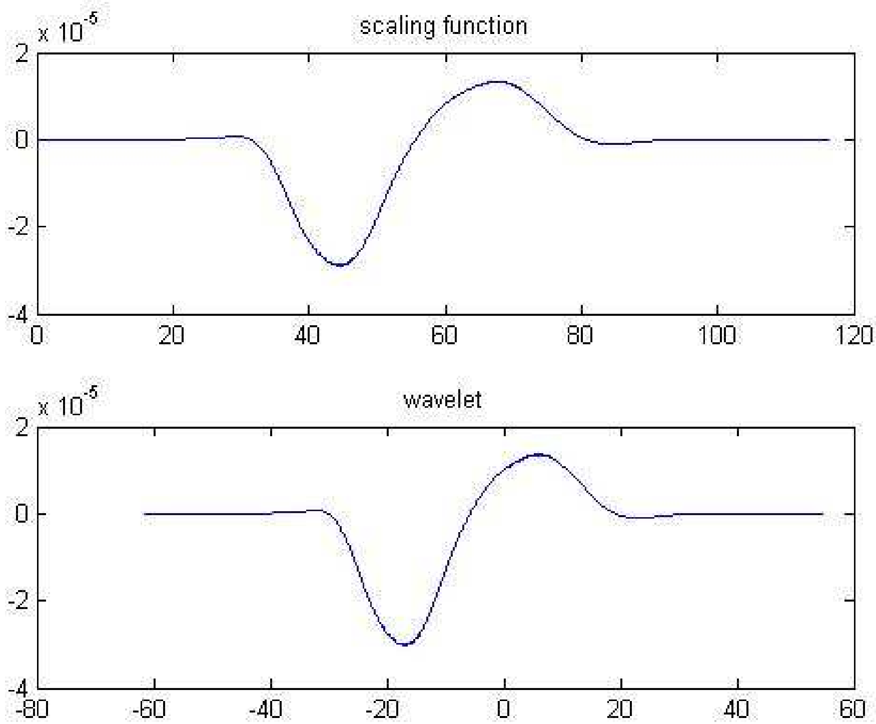


Result for test 2

Test 2 for unit delay

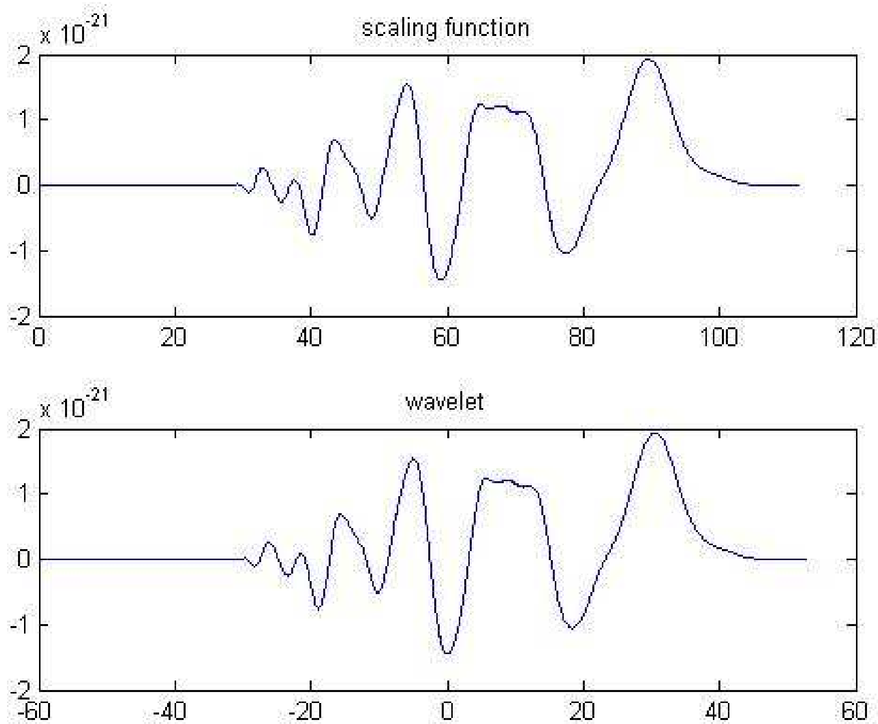


Fractional delay test2

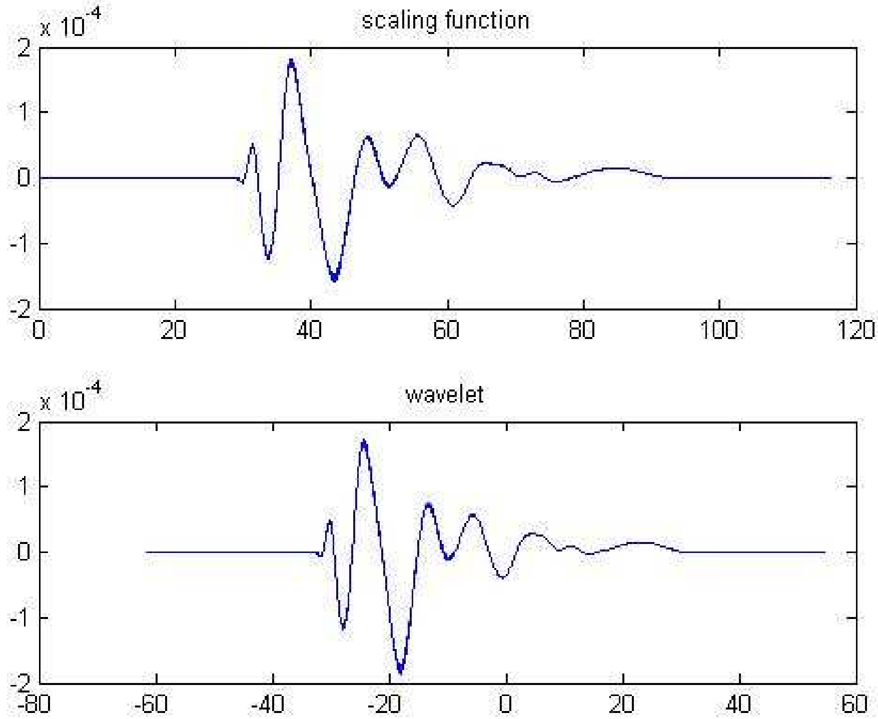


Results for test3

For unit delay filter



For fractional delay



LIST OF FIGURES

BIBLOGRAPHY

1. Vesa Välimäki, Azadeh Haghparast, "Fractional Delay Filter Design Based on Truncated Lagrange Interpolation", IEEE ,2007,Signal processing,page no. 816-819.vol.14, no.11.

2. C.W.Farrow , " a continuous variable digital delay elements" IEEE ,1998, digital communication, page no.2641-2645.

3.RONALD W. SCHAFER AND LAWRENCE R. RABINER , " A Digital signal processing approach to interpolation", IEEE, 1973,signal processing , page no.692 - 702, vol.16.no.14.

4.Saed samdi , M. Omair ahmad , and M.N.S Swamy, " results on maximally flat fractional delay system" , IEEE ,2004.,signal processing, electronics and computer science, page no.2271 -2286., vol.51, no.11.

5.Hakan Johansson and per lowenborg , "on the design of adjustable fractional delay FIR filters" ,IEEE,2003.,analog and digital signal processing, page no. 164-169.,vol.50, no.4.

6.Peter J. Kootsokos and Robert C. Williamson , " FIR approximation of fractional sample delay system" , IEEE,1996.,analog and digital signal processing ,page no. 269 -271.,vol.43.,no.3.

7.Chien-Cheng tseng , "design of variable fractional delay FIR filter using differentiator bank" , IEEE,2002 , communication, page no.421 -424

8.Gordana Jovanovic-Dolecek, Javier Diaz-Carmona , " one method for FIR fractional delay filter design" IEEE, 2002, circuits and systems , page no. c034-1 to c034-6.

9.Tian –bo Deng , “ variable fractional delay filter design using weighted least squares singular value decomposition”, IEEE conference , 2004.,communication, page no. 44 -47.

10.Nikzad babaii , AbdolReza Nabavi , “ design of fractional delay filters using least square error and fourier series” , IEEE conference ,2002.,electrical and computer engineering , page no. 1068 – 1072

11.William Putnam , Julius Smith , “design of fractional delay filters using convex optimization” , conference , music technology.

12.v.Valimaki and T.I.laakso , “ principle of fractional delay filters” ,conference , speech and signal processing, 2000.

13.Xiang-Gen Xia , “ fractional delay filter design when sampling rate is higher than nyquist rate” ,IEEE ,1997, signal processing, page no. 199-201.

14.Timo.i laakso, Vesa valimaki, matti karhalainen and Unto k. laine, “splitting the unit delay” ,IEEE signal processing magazine,1996, page no. 30 -60

15. Anubha Gupta, Shiv Dutt Joshi, and Surendra Prasad,Senior Member, IEEE. “A New Approach for Estimation of Statistically Matched Wavelet”, IEEE TRANSACTIONS ON SIGNAL PROCESSING, VOL. 53, NO. 5, MAY 2005.

16. Is11 Celasun**,** Biilent Sankur, Emin Anarim**,** Hakan Caglar ,Bogazisi University, Department of Electrical and Electronic Engineering,Bebek, 80815, Istanbul-TURKEY “Design Issues For Matched Wavelets”,IEEE 1994.

17. Joseph O. Chapa and Raghuveer M. Rao “Algorithms for Designing Wavelets to Match a Specified Signal” IEEE TRANSACTIONS ON SIGNAL PROCESSING, VOL. 48, NO. 12, DECEMBER 2000 .Signal processing.

18.Guangming Shi**(I)** School of Electronic Engineering,Xidian University, Xi’an, 710071,China Ailing Ding’.’, Licheng Jiao ’**(2)** School of Information Engineering, Changan University, Xi’an, 7 1005 1 ,China.
“A new Approach for Constructing Match Wavelet to Signal Detect ion”, IEEE, 2004 Signal processing.

19. Guangming Shi1 , Ailing Ding1, 2 and Ning Zhang1 School of Electronic Engineering, Xidian University Xi’ an, China, School of Information Engineering, Chang’an University Xi’an China “Design of wavelet based on waveform matching and its application to signal compression”,Signal Processing, IEEE, 2005.