

**Lecture 14: Second Member of Daubechies Family**

*Prof. V.M. Gadre, EE, IIT Bombay*

## 1 Introduction

In this lecture the discussion of Daubechies filter bank is continued which was briefly introduced in the previous lecture. The salient feature of Daubechies filter bank is that construction of Daubechies filter depends on adding polynomial of higher and higher degree in filter transfer function. Specifically, more and more  $(1 - z^{-1})$  terms in high pass analysis filter bank.

### Impulse response of Daubechies analysis low pass filter

The first member of the Daubechies family is the Haar filter bank itself. In the second member of Daubechies family the analysis side high pass filter has factor of  $(1 - z^{-1})^2$ . Now high pass filter of analysis side is of form  $z^{-D}H_0(-z^{-1})$ , where  $H_0(z)$  is the analysis side low pass filter. So  $H_0(z)$  should have a factor of  $(1 + z^{-1})^2$ .

It can be recalled that in the Daubechies family the filter lengths are always even. So for the second member of the Daubechies family filter length will be 4 and a order will be 3. So  $H_0(z)$  has 3 zeros. Two of them are already specified to be at  $z = -1$ . The third zero is to be determined to get complete transfer function. The complete transfer is obtained as follows:

Let impulse response be,

$$h[n] = \begin{bmatrix} h_0 & h_1 & h_2 & h_3 \end{bmatrix}$$

So  $h[n]$  is orthogonal to its even shifts *e.g.* shifts of 2,4,6 etc. So the only non-trivial relation is obtained by the dot product of  $h[n]$  and  $h[n - 2]$ .

$$h[n] = \begin{bmatrix} h_0 & h_1 & h_2 & h_3 \end{bmatrix}$$

$$h[n - 2] = \begin{bmatrix} \dots\dots h_0 & h_1 & h_2 & h_3 \end{bmatrix}$$

Hence their dot product is  $(h_0h_2 + h_1h_3)$  and because of orthogonality of dot product with respect to even shifts,

$$h_0h_2 + h_1h_3 = 0 \tag{1}$$

Now system function can be expressed as,

$$H_0(z) = h_0 + h_1z^{-1} + h_2z^{-2} + h_3z^{-3} \tag{2}$$

Also as  $H_0(z)$  has a factor of  $(1 + z^{-1})^2$ , so in general  $H_0(z)$  can be written as,

$$H_0(z) = C_0(1 + z^{-1})^2(1 + B_0z^{-1})$$

where  $C_0$  is a constant.

Here two zeros are constrained at  $z = -1$ . If we neglect the constant for time being and then expanding the previous expression, we can write,

$$\begin{aligned} H_0(z) &= 1 + 2z^{-1} + z^{-2} + B_0z^{-1} + 2B_0z^{-2} + B_0z^{-3} \\ H_0(z) &= 1 + (2 + B_0)z^{-1} + (1 + 2B_0)z^{-2} + B_0z^{-3} \end{aligned} \quad (3)$$

Comparing the coefficients of powers of  $z^{-1}$  from equation 2 and 3, we get,

$$\begin{aligned} h_0 &= 1 \\ h_1 &= 2 + B_0 \\ h_2 &= 1 + 2B_0 \\ h_3 &= B_0 \end{aligned}$$

Putting this value in equation (1), we get,

$$\begin{aligned} (1 + 2B_0) + (2 + B_0)B_0 &= 0 \\ \Rightarrow B_0^2 + 4B_0 + 1 &= 0 \end{aligned}$$

Solving the above quadratic equation, we get,

$$B_0 = (-4 \pm 2\sqrt{3})/2 = -2 \pm \sqrt{3} \quad (4)$$

Now the implication of  $B_0$  is that the third zero of  $H_0(z)$  is at  $B_0$ . For  $B_0 = -2 - \sqrt{3}$  *i.e.*  $B_0 = -3.732$ , the zero is outside the unit circle in the  $z$ -plane. Because  $|B_0| > 1$ , this will not become the minimum phase implementation. The magnitude response being the same as required. So there will be more phase delay and group delay in the system which is not desired. But for  $B_0 = -2 + \sqrt{3}$  *i.e.*  $B_0 = -0.268$ , the zero is inside the unit circle in the  $z$ -plane and so the system remains the minimum phase system, because  $|B_0| < 1$ . So we choose  $B_0 = -2 + \sqrt{3}$  *i.e.*  $B_0 = -0.268$ .

So the impulse response of the analysis side low pass filter of length 4 of Daubechies family is,

1	$(2 + B_0)$	$(1 + 2B_0)$	$B_0$
1	1.732	0.464	-0.268

In this derivation process we have neglected the constant  $C_0$ . To find  $C_0$ , let's recall:

$$\kappa_0(z) + \kappa_0(-z) = \text{constant} \quad (5)$$

where,

$$\kappa_0(z) = H_0(z)H_0(-z^{-1}) \quad (6)$$

In order to choose the constant  $C_0$  the easy thing to do is to make the norm of the impulse response of  $H_0(z)$  unity in the sense of  $L_2$  norm. Now the dot product of a sequence with itself

gives the square of its  $L_2$  norm. So the sequence corresponding to  $\kappa_0(z)$  at the  $0^{th}$  location is essentially the squared norm in the  $L_2$  of  $[h_0 \ h_1 \ h_2 \ h_3]$ . So  $C_0$  chosen such as,

$$\begin{aligned} C_0^2(h_0^2 + h_1^2 + h_2^2 + h_3^2) &= 1 \\ C_0^2 = 1/4.287 &= 0.233 \\ C_0 &= 0.4829 \end{aligned} \tag{7}$$

Hence the normalized impulse response is,

$h_0$	$h_1$	$h_2$	$h_3$
$0.4829*1=0.4829$	$0.4829*1.732=0.8364$	$0.4829*0.464=0.2241$	$0.4829*(-0.268)=-0.129$

## 1.1 Calculation of Scaling Function

Now our next job is to calculate  $\phi(t)$  and  $\psi(t)$  from the calculated impulse response. To calculate the scaling function  $\phi(t)$  we have to keep on comprising and convolving  $h[n]$  iteratively. This is done as follows:

Let us treat  $h[n]$  as the set of coefficients of an impulse train containing only 4 impulses in the continuous time domain such as the function in the continuous time domain is  $h(t)$  such as,

$$h(t) = h_0\delta(t) + h_1\delta(t - T) + h_2\delta(t - 2T) + h_3\delta(t - 3T) \tag{8}$$

Accordingly the shape of  $h(2t)$ ,  $h(4t)$ ,  $h(8t)$  etc are shown in figure 1. For the iterative convolution, first  $h(t)$  is convolved with  $h(2t)$  and the result is convolved with  $h(4t)$ . Then the result is convolved with  $h(8t)$  and so on. If this convolution process with compressed version of  $h(t)$  are carried on infinitely, we will get the scaling function  $\phi(t)$ .

There is an interesting conclusion of this iterative convolution process. Suppose  $h(t)$  has length  $L$  i.e.  $h(t) = 0$  for any  $t < 0$  and for any  $t > L$ . So  $h(2t)$  has a length  $L/2$ ,  $h(4t)$  has a length  $L/4$ ,  $h(8t)$  has a length  $L/8$  and so on. Now the convolution of  $h(t)$  with  $h(2t)$  gives result with length  $L + L/2$ . This result when convolved with  $h(4t)$ , gives a result with length  $L + L/2 + L/4$ . So going on this way we can get  $\phi(t)$ , which is of length  $L + L/2 + L/4 + \dots = 2L$ .

It means  $\phi(t)$  is zero for any  $t < 0$  and  $t > 2L$  i.e. we converge towards a compactly supported scaling function. The independent variable region over which the scaling function is non-zero is finite. This is the most important contribution made by Daubechies. Before Daubechies came up with these set of filter banks idea of neatly constructing a family of compactly supported multiresolution analysis was not in the literature. So this is a very useful contribution in MRA.

## Interpretation for Daubechies filter banks

Actually the theories of wavelets and filter banks developed in parallel. But using filter banks effectively to generate compactly supported scaling function is an important contribution by Daubechies.

A different interpretation can be thought about the Daubechies filter banks. The high pass analysis filter bank essentially reduces the degree of a polynomial input. Suppose there is an input of the form  $x[n] = a + bn$ , where  $a$  and  $b$  are constants. So the factor  $(1 - z^{-1})$  in the

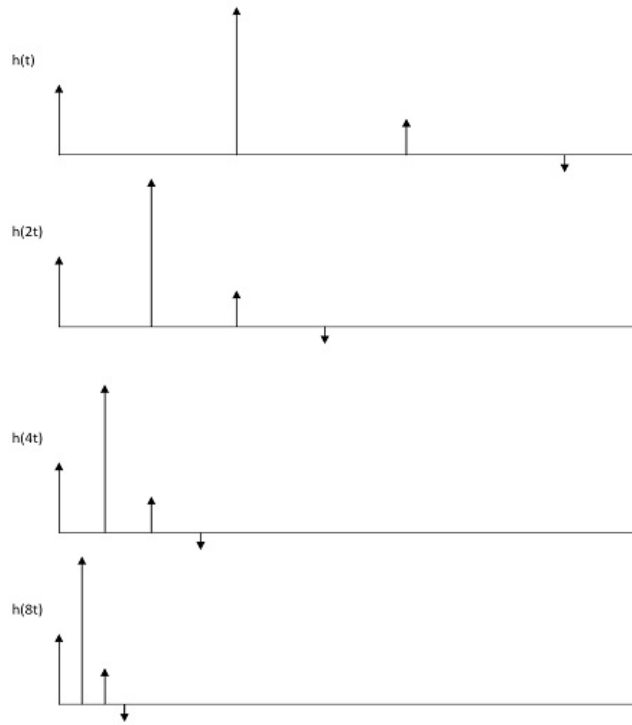


Figure 1: Shape of  $h(t)$ ,  $h(2t)$ ,  $h(4t)$  and  $h(8t)$

high pass filter reduces the degree of the input. If there had been only this term in the high pass filter, the output would have been in the form of  $y[n] = a + bn - a - b(n - 1) = b$ . If there had been another term of  $(1 - z^{-1})$  the output becomes 0. So in Daubechies length 4 (abbreviated as Daub-4) high pass filter bank the polynomial is annihilated. On the other side, in the low pass analysis filter, because of the  $(1 + z^{-1})$  term, the output becomes  $y[n] = a + bn + a + b(n - 1) = 2a + 2bn - b$ . It can similarly be extended for another  $(1 + z^{-1})$  term. This means that the polynomial form of expression remains in the low pass branch and the high pass branch contains some residual component, thereby retaining a few more smoother terms related to polynomial in the low pass branch and removing them from the high pass branch.

While calculating the iterative convolution we saw that the scaling function thus obtained has a compact support. But it is important to note that had it been taken any arbitrary values of  $h_0, h_1, h_2, h_3$  the iterative convolution process might not have converged to a function with finite number of discontinuities. But beauty of Daubechies family is that whatever be the filter length, the convolution always converges to a function with finite number of discontinuities. The specialty that makes the convolution to converge is denoted by a term 'regularity' in Wavelet literature, *i.e.* the filters need to obey regularity for the iterative convolution to converge. This regularity comes because of the presence of the zeros in the system function. One guaranteed way of forcing regularity is to introduce factors of  $(1 + z^{-1})$ , *i.e.* adding zeroes at  $z = -1$  *i.e.*  $\omega = \pi$  in the low pass analysis filters. In case of high pass analysis filter the zeros are added at  $z = 1$  *i.e.*  $\omega = 0$ . So the zeros are put at the extreme high frequency in low pass filter and at the extreme low frequency in high pass filter. In case of different filter banks, the number of zeros are:

Higher is the filter length more regular is the Daubechies filter. This means the function to which we converge by iterative convolution becomes more and more smooth *i.e.* they have

Haar	1 zeros
Daub-4	2 zeros
Daub-6	3 zeros

more and more derivative which are continuous. In Daubechies-4 there are some issues in the differentiability but in the higher order filters that is also taken care of.

## Next Daubechies family member Daub-6

The next member of Daubechies family is a length 6 filter of degree 5. So in that case  $H_0(z)$  can be written as,

$$H_0(z) = C_0(1 + z^{-1})^3(1 + \widetilde{B}_0 z^{-1})(1 + \widetilde{B}_1 z^{-1}) \quad (9)$$

Here  $C_0$  is a constant. Three zeros are constrained and two are free ( $\widetilde{B}_0$  and  $\widetilde{B}_1$ ). Let the impulse response be,

$$h[n] = \begin{bmatrix} h_0 & h_1 & h_2 & h_3 & h_4 & h_5 \end{bmatrix}$$

So  $h[n]$  is orthogonal to its even shifts *e.g.* shift of 2, 4, 6 etc. So the non trivial relations are obtained by the dot product between  $h[n]$  and  $h[n-2]$  or  $h[n-4]$ . Here,

$$h[n] = \begin{bmatrix} h_0 & h_1 & h_2 & h_3 & h_4 & h_5 \end{bmatrix}$$

$$h[n-2] = \begin{bmatrix} \dots\dots\dots h_0 & h_1 & h_2 & h_3 & h_4 & h_5 \end{bmatrix}$$

$$h[n-4] = \begin{bmatrix} \dots\dots\dots\dots\dots\dots h_0 & h_1 & h_2 & h_3 & h_4 & h_5 \end{bmatrix}$$

$$\Rightarrow h_2 h_0 + h_3 h_1 + h_4 h_2 + h_5 h_3 = 0 \quad (10)$$

$$\Rightarrow h_4 h_0 + h_5 h_1 = 0 \quad (11)$$

From equations 9, 10 and 11, we can find out the values of  $\widetilde{B}_0$  and  $\widetilde{B}_1$ . There from we can construct the impulse response in a way similar to Daub-4 case.

This type of filter banks are called Conjugate Quadrature filter bank. The reason for this nomenclature is that the low pass and the high pass filter frequency responses are  $\pi$  apart from each other. So the principal equation governing the conjugate quadrature filter is,

$$\kappa_0(z) + \kappa_0(-z) = constant \quad (12)$$

where,

$$\kappa_0(z) = H_0(z)H_0(z^{-1}) \quad (13)$$

$$\kappa_0(e^{j\omega}) = H_0(e^{j\omega})H_0(e^{j\omega}) \quad (14)$$

If the frequency response is real then,

$$|H_0(e^{j\omega})|^2 = \kappa_0(e^{j\omega}) \quad (15)$$

This means designing a conjugate quadrature filter bank is essentially designing  $\kappa_0(e^{j\omega})$  only.  $\kappa_0(z)$  corresponds to a real and even impulse response with the constraints that even samples of the impulse response are all '0' except at the 0th sample.

There are many ways to design such filter banks. Once we have  $\kappa_0(z)$  we find its roots. For each root there are pairs of reciprocal roots  $H_0(z)$  and  $H_0(z^{-1})$ . Out of each reciprocal root pair, one root is assigned to  $H_0(z)$  and the other automatically gets assigned to  $H_0(z^{-1})$ . The Daubechies filters are one class of conjugate quadrature filters.

Our next study will aims at what we are looking for out of these filter banks in both time and frequency domains.