



**Fermi National Accelerator Laboratory**

**FERMILAB-TM-2100**

## **Peak Finding Using Biorthogonal Wavelets**

Cheng-Yang Tan

*Fermi National Accelerator Laboratory  
P.O. Box 500, Batavia, Illinois 60510*

February 2000

## **Disclaimer**

*This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, expressed or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.*

## **Distribution**

*Approved for public release; further dissemination unlimited.*

## **Copyright Notification**

*This manuscript has been authored by Universities Research Association, Inc. under contract No. DE-AC02-76CH03000 with the U.S. Department of Energy. The United States Government and the publisher, by accepting the article for publication, acknowledges that the United States Government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this manuscript, or allow others to do so, for United States Government Purposes.*

# Peak Finding Using Biorthogonal Wavelets

Cheng-Yang Tan

*Beams Division/Tevatron*

ABSTRACT: We will show in this paper how we can find the peaks in the input data if the underlying signal is a sum of Lorentzians. In order to project the data into a space of Lorentzian like functions, we will show explicitly the construction of scaling functions which look like Lorentzians. From this construction, we can calculate the biorthogonal filter coefficients for both the analysis and synthesis functions. We then compare our biorthogonal wavelets to the FBI (Federal Bureau of Investigations) wavelets when used for peak finding in noisy data. We will show that in this instance, our filters perform much better than the FBI wavelets.

## INTRODUCTION

The transverse tune spectra consists of many peaks from synchrotron lines. The simplest way to find the peaks in the spectra is to perform a differential on the spectra and to look for zero-crossings in order to isolate the peaks. However, this simple method is inherently inefficient because the input data is usually quite noisy which results in zero-crossings which produces false peaks. It is also well known that differentiation in itself introduces more noise into the data and thus makes a bad situation even worse. Therefore what is usually done is the use of a low pass filter (LPF) which smooths out the signal before the differential process. Since this traditional method requires a filter, we can go one step better and think of a set of tailored filters which can be used to isolate the peaks.

It is well known that the Fast Wavelet Transform (FWT) when implemented as a multi-resolution analysis are sets of high pass filters (HPF) and low pass filters with down-sampling by 2 at each stage. Therefore, we propose a method where we use Lorentzian-like scaling functions with compact support to be the vector space where the input data will be projected. This will then allow us to easily see where the Lorentzian peaks are. Hence, the goal is to find the finite impulse response (FIR) coefficients of the scaling basis functions and wavelet basis functions.

It is immediately obvious that a Lorentzian

$$f(x) = \frac{1}{a(x^2 + b^2)} \tag{1}$$

is symmetric about  $x = 0$ , thus we expect the FIR filter to have coefficients of the form  $\{\dots, f_2, f_1, f_0, f_1, f_2, \dots\}$ , i.e. symmetric about  $f_0$ . There is a well known theorem in wavelet theory that a symmetric FIR filter cannot yield orthonormal wavelets.<sup>†</sup> Thus, we are left with the next simplest choice which are biorthogonal wavelets.

---

<sup>†</sup> For proof, see Rao<sup>1</sup> page 87. For proof that orthogonal wavelets with finite support cannot be symmetric, see Daubachies<sup>2</sup>, Theorem 8.1.4 on page 252.

## BIORTHOGONAL WAVELETS

Unlike orthogonal wavelets,<sup>†</sup> we have two vector spaces  $W$  and  $\tilde{W}$  where  $\psi \in W$  and  $\tilde{\psi} \in \tilde{W}$  have the property

$$\int_{-\infty}^{\infty} dx \psi(x)\tilde{\psi}(x-n) = \delta_{n0} \quad (2)$$

This property (2), defines biorthogonality.

It is clear from (2) that unlike orthogonal wavelets, we have two sets of wavelets  $\psi$ ,  $\tilde{\psi}$  and thus two sets of scaling functions  $\phi$ ,  $\tilde{\phi}$  which belong to the vector spaces  $V$  and  $\tilde{V}$  respectively. Therefore, we must have more than one independent set of FIR coefficients rather than the one independent set as in orthogonal wavelets.

In a multi-resolution analysis, we have embedded vector spaces  $V_{\infty} \subset \dots \subset V_k \subset V_{k-1} \subset \dots \subset V_{-\infty}$  which can be thought of as going from the finest resolution  $V_{-\infty}$  to the coarsest resolution  $V_{\infty}$ . Suppose  $y_{k-1}(t)$  is a  $L_2$ -function in the vector space  $V_{k-1}$ , then  $y_k(t) \in V_k$  is the approximation of  $y_{k-1}(t) \in V_{k-1}$  i.e.

$$y_k(t) = \sum_{n=-\infty}^{\infty} a(k, n)\phi(2^{-k}t - n) \quad (3)$$

where

$$a(k, n) = 2^{-k}\langle y_{k-1}(t), \tilde{\phi}(2^{-k}t - n) \rangle \quad (4)$$

The error  $\Delta_k \in W_k$  of approximating  $y_{k-1}$  by  $y_k$  is

$$\begin{aligned} \Delta_k &= y_{k-1}(t) - y_k(t) \\ &= \sum_{n=-\infty}^{\infty} b(k, n)\psi(2^{-k}t - n) \end{aligned} \quad (5)$$

where

$$b(k, n) = 2^{-k}\langle y_{k-1}(t), \tilde{\psi}(2^{-k}t - n) \rangle \quad (6)$$

---

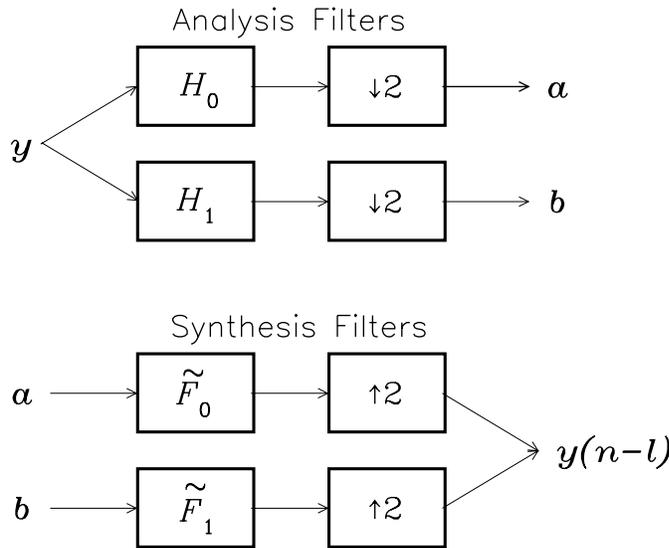
<sup>†</sup> In order not to unnecessary add length to this rather long paper, we refer the interested reader to Vidaković<sup>3</sup> and Sweldens<sup>4</sup> for some excellent tutorials on wavelets. Also Mathsoft<sup>5</sup> has an excellent wavelet resources webpage.

Notice that  $a$  is obtained from the projection of  $y$  onto the subspace  $\tilde{V}_k$  and *not*  $V_k$  as in orthogonal wavelets. Similarly,  $b$  is obtained from the projection of  $y_{k-1}$  onto the subspace  $\tilde{W}_k$  rather than  $W_k$ .

It has been shown by Daubechies that calculating  $a$  and  $b$  in a multi-resolution analysis can be done using FIR filters. In particular the FIR coefficients  $\{h_0, \dots, h_n\}$  are the coefficients of the polynomial

$$H(z) = \sum_0^n h_n z^{-n} \tag{7}$$

with normalizing condition  $H(1) = 1$ .



**Figure 1** The analysis filters  $H_0$  and  $H_1$  are filters which map the input  $y$  to  $\tilde{V}$  and  $\tilde{W}$  respectively. The synthesis filters  $\tilde{F}_0$  and  $\tilde{F}_1$  undo the the effect of the  $H_0$  and  $H_1$  filters to produce  $y(n - \ell)$ .

Let us write  $H_0$  to be the filter associated with the scaling function  $\phi$  and  $H_1$  to be the filter associated with the scaling function  $\psi$ . Similarly  $\tilde{F}_0$  and  $\tilde{F}_1$  are the filters associated with  $\tilde{\phi}$  and  $\tilde{\psi}$  respectively. Then, Figure 1 shows the one stage of the FWT and inverse FWT algorithm. Notice that although we have perfect reconstruction of the signal, there is a finite delay  $\ell$  between the input and output<sup>6</sup>.

## CONSTRUCTING $\tilde{\phi}$

We present here an algorithm for finding the scaling function  $\tilde{\phi}$  which is *close* to a Lorentzian. We will use the usual Euclidean measure to define *close*, i.e. we want to minimize

$$\epsilon = \sum_i \left( \tilde{\phi}(x_i) - \mathcal{L}(x_i) \right)^2 \quad (8)$$

with the Lorentzian  $\mathcal{L}$

$$\mathcal{L}(x) = \frac{1}{2\pi^2 \left( x^2 + \left( \frac{1}{2\pi} \right)^2 \right)} \quad (9)$$

where the normalizations of  $\mathcal{L}$  are chosen so that both  $\int_{-\infty}^{\infty} dx \mathcal{L}(x) = \int_{-\infty}^{\infty} dx \mathcal{L}^2(x) = 1$ .

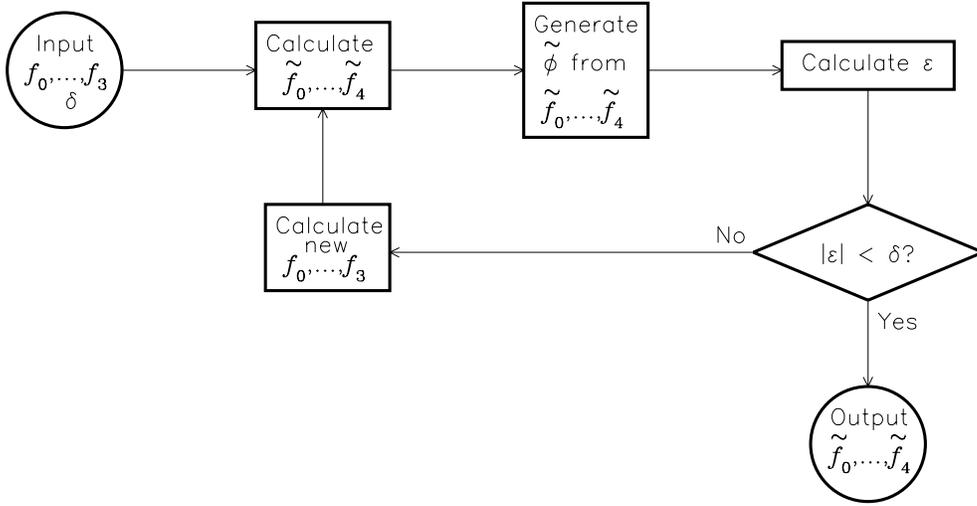
For the purposes of this paper, we will construct a length 9 filter associated with the scaling function  $\tilde{\phi}$ . As was discussed previously, the filter coefficients  $\{\tilde{f}_i | i = 0, \dots, 4\}$  are the coefficients of the expansion

$$\tilde{F}_0(z) = \tilde{f}_0 + \frac{\tilde{f}_1}{z} + \frac{\tilde{f}_2}{z^2} + \frac{\tilde{f}_3}{z^3} + \frac{\tilde{f}_4}{z^4} + \frac{\tilde{f}_3}{z^5} + \frac{\tilde{f}_2}{z^6} + \frac{\tilde{f}_1}{z^7} + \frac{\tilde{f}_0}{z^8} \quad (10)$$

From Strang, we know that the smoothness of the scaling function depends on the number of zeros  $\tilde{F}_0$  has at  $z = -1$ . In particular we choose  $\tilde{F}_0$  to have two zeros at  $z = -1$  which means that we reduce the number of degrees of freedom by two, i.e.

$$\tilde{F}_0(z) = \left( 1 + \frac{1}{z} \right)^2 \left( f_0 + \frac{f_1}{z} + \frac{f_2}{z^2} + \frac{f_3}{z^3} + \frac{f_2}{z^4} + \frac{f_1}{z^5} + \frac{f_0}{z^6} \right) \quad (10)$$

At this point, we use the computer algorithm shown in Figure 2, to find  $f_0, \dots, f_3$ . Then by expanding (10) we can calculate the filter coefficients  $\tilde{f}_0, \dots, \tilde{f}_4$ .



**Figure 2** This is the flow chart used to calculate  $\tilde{f}_0, \dots, \tilde{f}_4$ .  $\tilde{\phi}$  is generated from  $\tilde{f}_0, \dots, \tilde{f}_4$  using a time domain iteration method described by Rao<sup>1</sup>. If  $|\epsilon| > \delta$ , coefficients  $f_0, \dots, f_3$  are changed using Powell's method. We continue iterating until the required precision is reached.

### CONSTRUCTING $\phi$ FROM $\tilde{\phi}$

Once the  $\tilde{\phi}$  FIR filter coefficients  $\tilde{f}_0, \dots, \tilde{f}_4$  are found, we have to construct the  $\phi$  FIR filter coefficients from them. We know from our previous discussion that the  $\phi$  filter coefficients are  $h_0, h_1, \dots$  of the expansion

$$H_0(z) = h_0 + \frac{h_1}{z} + \frac{h_2}{z^2} + \dots + \frac{h_n}{z^n} \quad (11)$$

In order to find  $H_0(z)$ , we will use the method from Strang<sup>6</sup>. It has been shown by Strang that if  $P(z)$  is defined to be

$$P(z) = H_0(z)\tilde{F}_0(z) \quad (12)$$

then

$$P(z) - P(-z) = \frac{1}{z^\ell} \quad (13)$$

If we choose  $H_0$  to have three zeros at  $z = -1$ , then the highest power of  $H_0$  is  $1/z^{14}$  and  $\ell = 10$  which means that we can write  $H_0$  as

$$H_0(z) = \left(1 + \frac{1}{z}\right)^3 \left(\bar{h}_0 + \frac{\bar{h}_1}{z} + \dots + \frac{\bar{h}_5}{z^5} + \frac{\bar{h}_5}{z^6} + \frac{\bar{h}_4}{z^7} + \dots + \frac{\bar{h}_1}{z^{10}} + \frac{\bar{h}_0}{z^{11}}\right) \quad (14)$$

Using the condition given by (13), we have exactly six equations and six unknowns  $\bar{h}_0, \dots, \bar{h}_5$  which are

$$\left. \begin{aligned} \text{Coeff} \left[ H_0, 1/z^0 \right] &= 10 f_0 \bar{h}_0 + 2 f_1 \bar{h}_0 + 2 f_0 \bar{h}_1 = 0 \\ \text{Coeff} \left[ H_0, 1/z^2 \right] &= 2 \left( 10 f_0 \bar{h}_0 + 10 f_1 \bar{h}_0 + 5 f_2 \bar{h}_0 + \dots \right) = 0 \\ \text{Coeff} \left[ H_0, 1/z^4 \right] &= 2 \left( f_0 \bar{h}_0 + 6 f_1 \bar{h}_0 + 15 f_2 \bar{h}_0 + \dots \right) = 0 \\ \text{Coeff} \left[ H_0, 1/z^6 \right] &= 2 \left( 5 f_0 \bar{h}_0 + 10 f_1 \bar{h}_0 + 11 f_2 \bar{h}_0 + \dots \right) = 0 \\ \text{Coeff} \left[ H_0, 1/z^8 \right] &= 2 \left( 10 f_0 \bar{h}_0 + 5 f_1 \bar{h}_0 + 10 f_0 \bar{h}_1 + \dots \right) = 0 \\ \text{Coeff} \left[ H_0, 1/z^{10} \right] &= 4 \left( f_0 \bar{h}_0 + 5 f_0 \bar{h}_1 + f_1 \bar{h}_1 + \dots \right) = 1 \end{aligned} \right\} \quad (15)$$

which we can easily solve. (The function  $\text{Coeff}[H_0, 1/z^i]$  just picks out the  $1/z^i$ th coefficient of  $H_0$ ). By substituting the solution  $\bar{h}_0, \dots, \bar{h}_5$  into (14) and expanding, we obtain the  $\phi$  filter coefficients  $h_0, \dots, h_7$ .

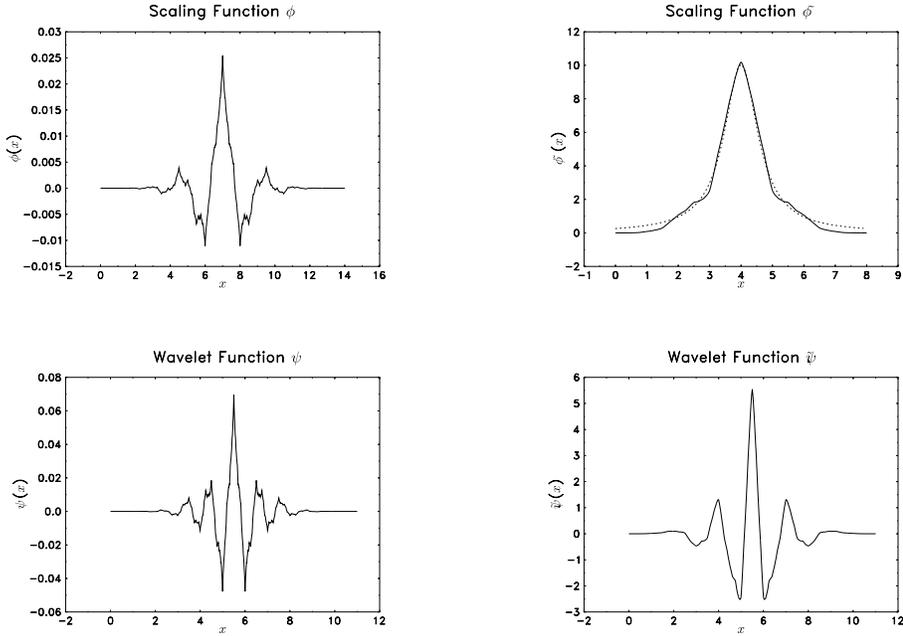
## THE FILTER COEFFICIENTS

With the  $\phi$  and  $\tilde{\phi}$  filter coefficients constructed from the previous sections, the  $\psi$  and  $\tilde{\psi}$  filter coefficients are trivially obtained by  $H_1(z) = \tilde{F}_0(-z)$  and  $\tilde{F}_1(z) = -H_0(-z)$ . The results are summarized in Table 1 where we list only the independent coefficients because  $h_8 = h_6$  etc., and  $\tilde{f}_3 = \tilde{f}_5$  etc. As a check that the coefficients are normalized correctly, we see that  $\sum_{i=0}^{14} \text{Coeff}[H_0, 1/z^i] = \sum_{i=0}^8 \text{Coeff}[\tilde{F}_0, 1/z^i] = 1$ . The  $\phi$  and  $\psi$  functions and  $\tilde{\phi}$  and  $\tilde{\psi}$  functions are shown in Figure 3 which are constructed from the filter coefficients of Table 1. Again, as a check, we see that the integral of  $\psi$  and  $\tilde{\psi}$  are zero as required.

**Table 1. The  $H_0$  and  $\tilde{F}_0$  Coefficients**

$i$	$h_i$	$\tilde{f}_i$
0	-0.01423519568	0.0357152
1	0.02101871946	0.0527346
2	0.07626104903	0.0151701
3	-0.04290477625	0.197265
4	-0.25612225073	0.398228
5	-0.1805498895	-
6	0.4440965418	-
7	0.9048721813	-

To see how well the filter coefficients work, we can construct a Lorentzian (Figure 4) and pass it through the  $\tilde{\phi}$  and  $\tilde{\psi}$  filters. We would expect that there will only be one large  $\phi$  coefficient with the rest small. This is indeed what we see in Figure 5 where the coefficients are plotted out.



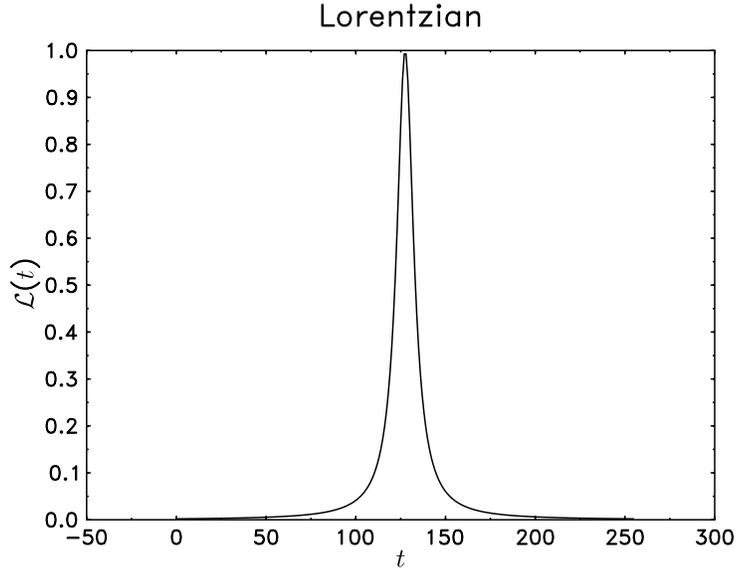
**Figure 3** The analysis filters produce the scaling and wavelet functions  $\phi$  and  $\psi$ . The analysis filters map the input data to the vector space where the scaling functions  $\tilde{\phi}$  look similar to Lorentzians.

## PEAK FINDING

The idea behind our peak finding method is to assume that the peaks come from Lorentzians. If the underlying basis are Lorentzians then we can express the input data  $y(t)$  as

$$y(t) = \sum_{i=0}^M a_i \mathcal{L}_i(t) \quad (16)$$

where  $a_i$  are constants. Therefore, if we can project the  $y(t)$  into Lorentzian space, we should be able to recover  $a_i$ 's. Thus by using the filters that we have discussed, we should be able to find the peaks in  $y(t)$  by looking for large  $a_i$ 's after it goes through our filters.

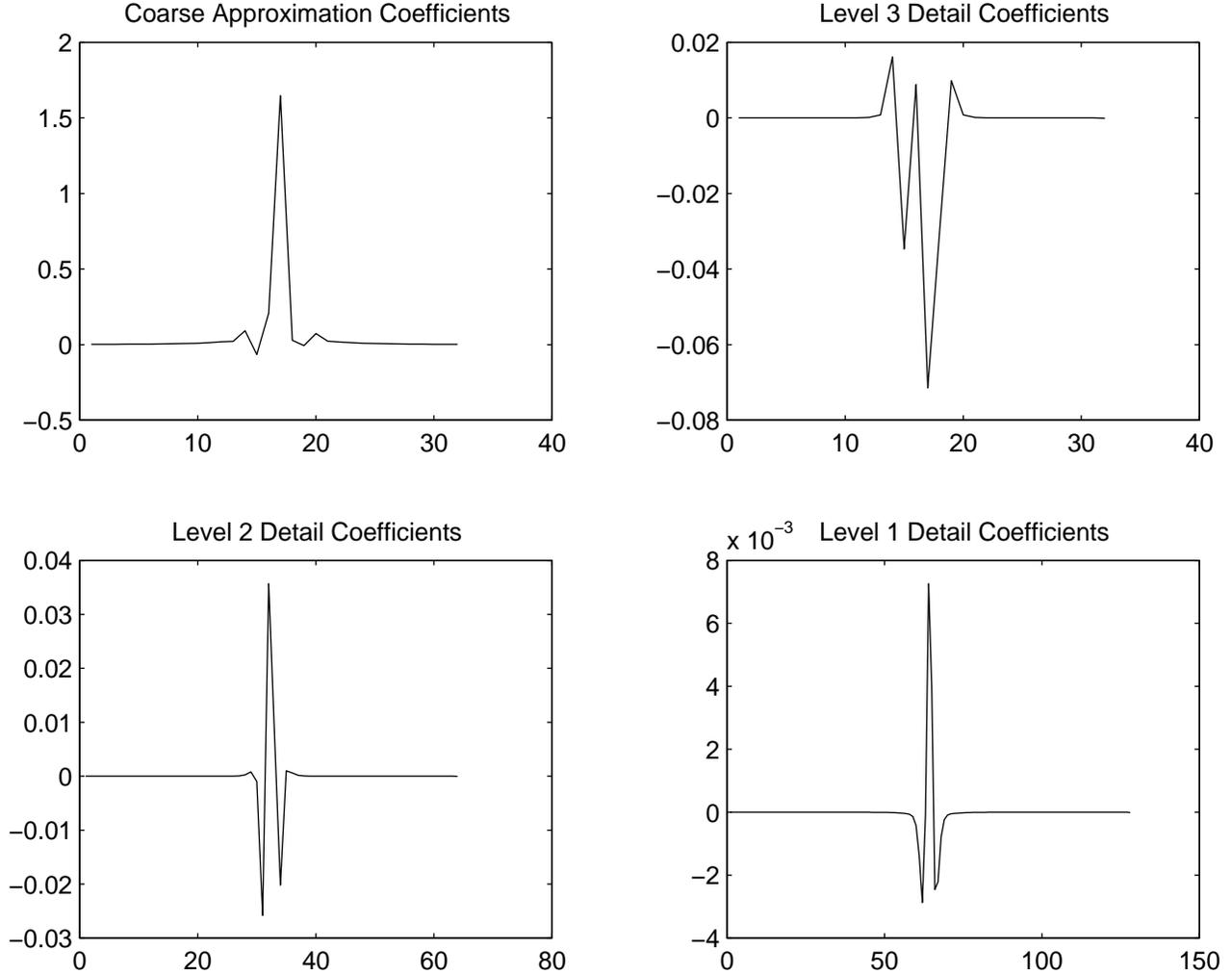


**Figure 4** This Lorentzian  $\mathcal{L}(t) = \frac{0.05}{t^2+0.05}$  is used as an example of how we project this into  $V$ .

**Table 2. The FBI  $H_0$  and  $\tilde{F}_0$  Coefficients**

$i$	$h_i$	$\tilde{f}_i$
0	0.03	-0.05
1	-0.02	-0.03
2	-0.08	0.3
3	0.27	0.56
4	0.6	0.3
5	0.27	-0.03
6	-0.08	-0.05
7	-0.02	-
8	0.03	-

As an illustration, let us see how well this idea works. We will compare our filters to the FBI filters<sup>6</sup>. The filter coefficients to two decimal places of the FBI 9/7 filters are



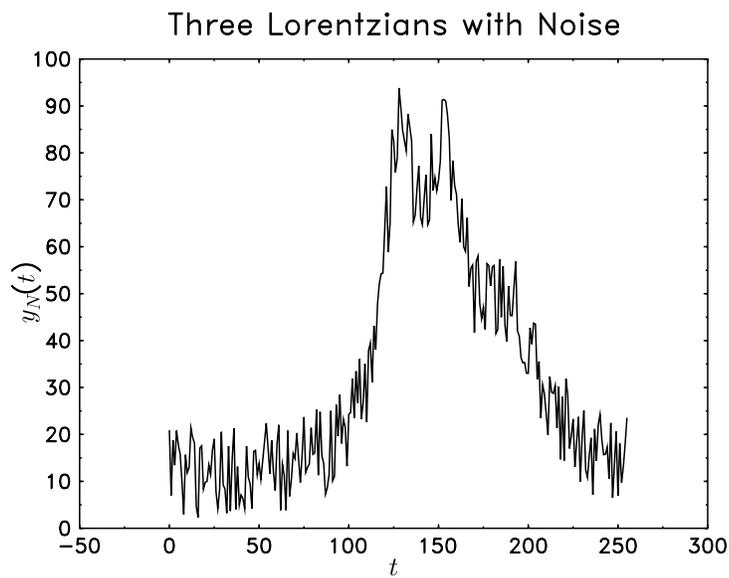
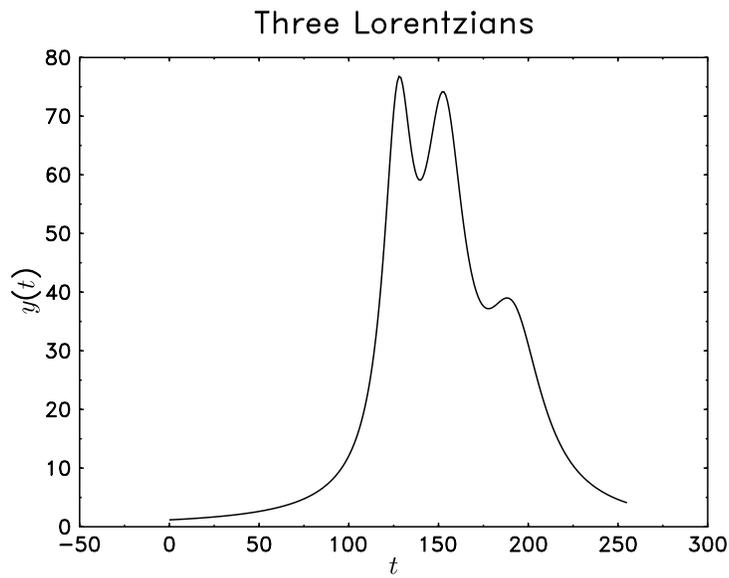
**Figure 5** By feeding the Lorentzian  $\mathcal{L}(t)$  shown in Figure 4 into the  $\tilde{\phi}$  and  $\tilde{\psi}$  filters to three levels, we see that  $\mathcal{L}(t)$  is projected into  $V_3$ . The coarse and detail coefficients are small everywhere except at one point. Thus we have successfully projected  $\mathcal{L}(t)$  into a Lorentzian like space described by  $V_3$ , i.e.  $\mathcal{L}(t) \approx 1.64 \phi(2^3 t - 17)$ .

given in Table 2. We construct a noisy input  $y(t)$  with three Lorentzians

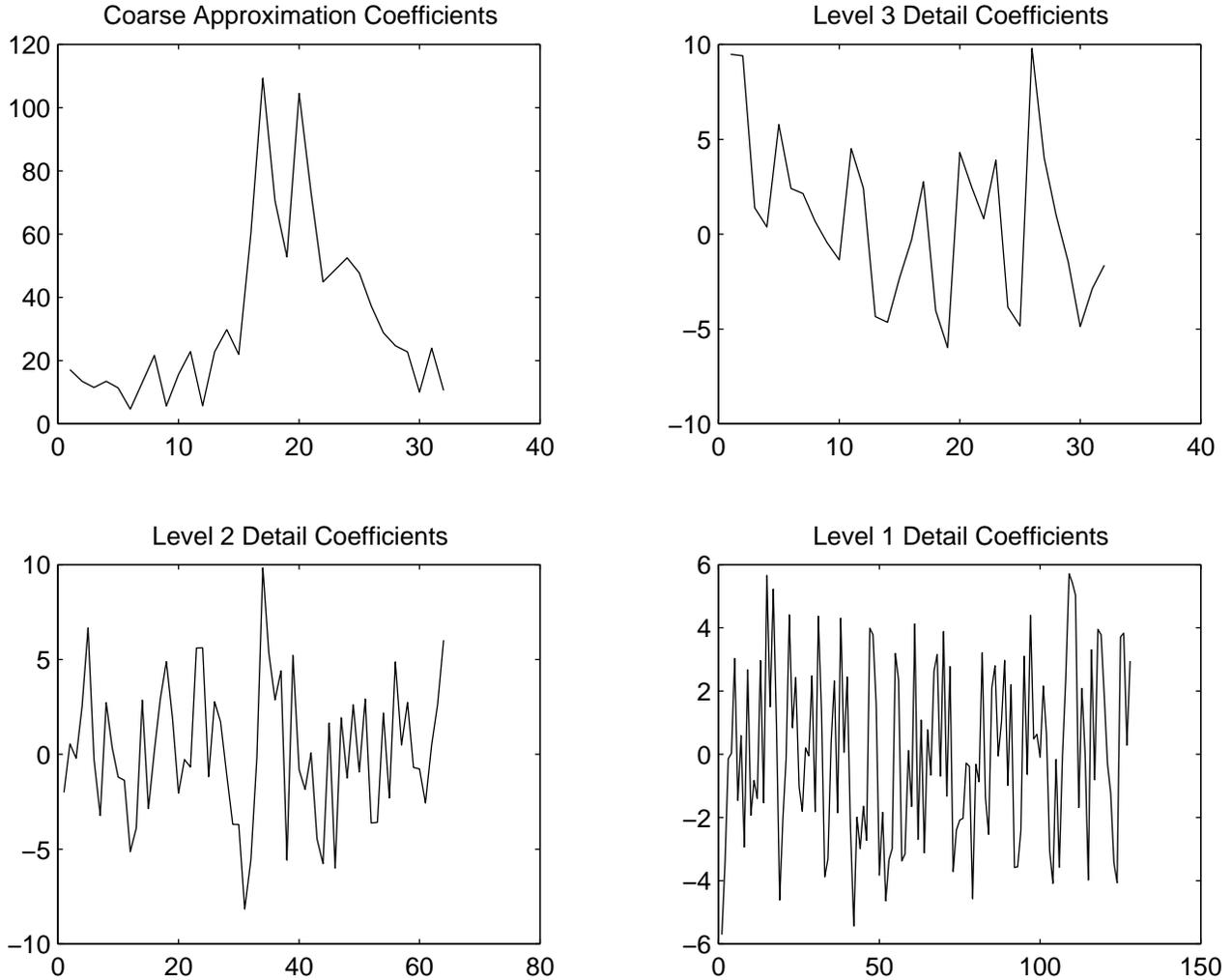
$$y(t) = \frac{9}{x^2 + 0.15} + \frac{18}{(x - 1)^2 + 0.3} + \frac{18}{(x - 2.5)^2 + 0.6} \quad (16)$$

with noise  $|n(t)| < 20$  added to it. The result is shown in Figure 6.

$y_N(t)$  is then passed through our filters and we clearly see that, above the noise floor, which we pick to be  $y = 20$ , there are three large coefficients relative to its neighbours in

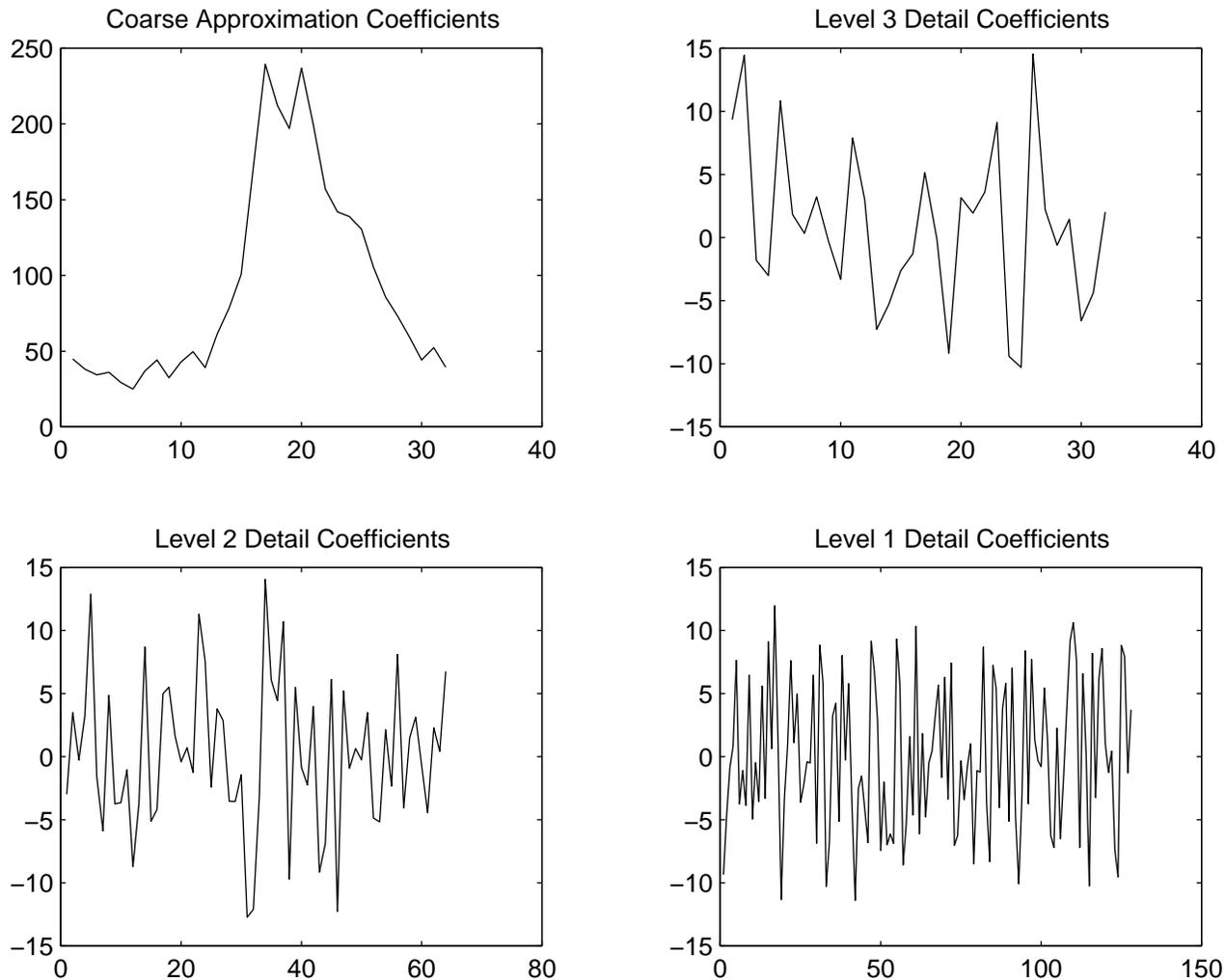


**Figure 6** The top figure shows three Lorentzians  $y(t) = \frac{9}{x^2+0.15} + \frac{18}{(x-1)^2+0.3} + \frac{18}{(x-2.5)^2+0.6}$ . Uniform noise  $-20 < n(t) < 20$  is then added to  $y(t)$  to produce  $y_N(t)$  which is shown in the bottom figure.



**Figure 7**  $y_N(t)$  shown in Figure 6 is fed through the  $\tilde{\phi}$  and  $\tilde{\psi}$  filters to three levels and we can clearly see that there are three peaks in the data.

Figure 7. We conclude, then, that there are three peaks in the original input  $y_N(t)$ . When  $\mathcal{L}_N$  is passed through the FBI filters, we see that the peaks are not as distinct although they perform the same type of noise filtering as our filters. In fact, although the FBI filter picks out two of the three peaks, the third one is barely visible.



**Figure 8** We apply the FBI filters to  $y_N(t)$  and compare the result with our filters. We see that the peaks are not as well defined as Figure 7.

## CONCLUSION

We have shown how we can construct biorthogonal scaling functions and wavelets which will enable us to easily find the peaks in the input data if the underlying vector space is Lorentzian like. We have also shown that compared to the FBI filters, our filters do a much better job in picking out the peaks in the input data.

## REFERENCES

- [1] *Wavelet Transforms, Introduction to Theory and Applications*, R.M. Rao and A.S. Bopardikar, Addison-Wesley, 1998.
- [2] *Ten Lectures on Wavelets*, I. Daubechies, Society for Industrial and Applied Mathematics, 1992.
- [3] *Wavelets for Kids*, B. Vidaković and P. Müller, <ftp://ftp.isds.duke.edu/pub/Users/brani/papers/wav4kidsA.ps.Z>.
- [4] *Building Your Own Wavelets at Home*, W. Sweldens and P. Schröder, <http://cm.bell-labs.com/who/wim/papers/athome.ps.gz>.
- [5] *Wavelet Resources*, Mathsoft, <http://www.mathsoft.com/wavelets.html>.
- [6] *Creating and Comparing Wavelets*, G. Strang, <http://www-math.mit.edu/~gs/papers/dundee.ps.gz>.
- [7] *Existence and Construction of Optimal Wavelet Basis for Signal Representation*, Y. Zhung and J.S. Baras, Center for Satellite and Hybrid Communication Networks, CSHCN T.R. 94-9.