ELSEVIER

Chemometrics and Intelligent Laboratory Systems 37 (1997) 215–239

Chemometrics and intelligent laboratory systems

Tutorial

An introduction to wavelet transforms for chemometricians: A time–frequency approach

Bjørn K. Alsberg *, Andrew M. Woodward, Douglas B. Kell

Institute of Biological Sciences, University of Wales, Aberystwyth, Ceredigion SY23 3DA, UK

Received 25 October 1996; accepted 20 February 1997

Abstract

One way to obtain an intuitive understanding of the wavelet transform is to explain it in terms of segmentation of the time–frequency/scale domain. The ordinary Fourier transform does not contain information about frequency changes over time and the short time Fourier transform (STFT) technique was suggested as a solution to this problem. The wavelet transform has similarities to STFT, but partitions the time–frequency space differently in order to obtain better resolutions along time and frequency/scales. In STFT a constant bandwidth partitioning is performed whereas in the wavelet transform the time–frequency domain is partitioned according to a constant *relative* bandwidth scheme. In this paper we also discuss the following application areas of wavelet transforms in chemistry and analytical biotechnology: denoising, removal of baselines, determination of zero crossings of higher derivatives, signal compression and wavelet preprocessing in partial least squares (PLS) regression.

Keywords: Wavelet transform; Short time Fourier transform (STFT); Frames; Time–frequency space; Baseline removal; Compression; Zero crossing

Contents

[*] Corresponding author. E-mail: bka|azw|dbk@aber.ac.uk.

PII S0169-7439(97)00029-4

1. Introduction

1.1. History

Wavelet is a word which means a 'small wave'. It is a theoretical formalism that was initiated by the French seismologist Jean Morlet [1–3] in 1980. He subsequently collaborated with the theoretical physicist Alex Grossmann [4–6] where wavelets were broadly defined in the context of quantum physics. This provided an understanding of wavelets based on physical intuition. But the mathematical theory leading up to wavelet theory can be traced much further back in time. In many ways it was initiated by Joseph Fourier (1807) with his theories of frequency analysis. After 1807 the first mention of mathematical objects that are wavelets appeared in an appendix to the thesis of Haar in 1909. The Haar wavelet has the property of being so-called *compactly supported* which means that it vanishes outside of a finite interval (see Section A.1). The Haar wavelets are not smooth functions (i.e. continuously differentiable) which of course limited their application. Gradually, mathematicians moved from the concept of frequency analysis to *scale analysis* which involves creating mathematical structures that vary in scale. Assume that a signal approximation is wanted. A function is created that is shifted by some amount and scaled. That structure is then used to approximate the signal. This process can be repeated by new shifts and scalings of the previous structures. At each step a new approximation of the signal can be accomplished. This type of scale analysis is less sensitive to noise because it measures the average fluctuations of the signal at different scales.

In the 1930's several groups (Levy, Littlewood, Paley, Stein and others) made contributions to the field of scale-varying basis functions which became fundamental for the theory of wavelets.

In 1985 Stephane Mallat discovered relations between filter-based signal compression methods and orthonormal wavelet bases. This work inspired Yves Meyer to construct the first nontrivial continuously differentiable wavelets. Meyer later collaborated with Grossmann and Ingrid Daubechies to find a discrete formulation of the continuous wavelet transform which was referred to as wavelet frames [7]. Meyer tried to prove that it was not possible to construct real orthonormal wavelets and was therefore quite surprised when he actually found an orthogonal basis from a regular wavelet [8,9]. Later Daubechies managed to construct orthogonal wavelets with compact support [10].

Wavelet theory can be divided into the following main categories:

(1) Continuous wavelet transforms

(2) Discrete wavelet transforms: (a) orthonormal bases of wavelets and (b) redundant discrete systems (frames).

Introductions to wavelet may be found in Refs. [11–20] and more advanced texts include Refs. [21–26]. A most helpful recent tutorial on wavelet packets appeared in this journal [27].

1.2. Classical Fourier transforms

Wavelets are best understood by first looking at some of the limitations of Fourier transform methods, in particular the short-time Fourier transform (STFT) which will be described below. The standard Fourier transform is defined as:

$$F(\omega) = \int_{-\infty}^{\infty} x(t) e^{-i\omega t} dt \tag{1}$$

A useful way of understanding the Fourier transform is to say that the signal $f(t)$ has been *projected* onto a set of *basis functions*. The basis functions in this case are the cosine and sine functions represented by complex exponential functions. All the basis functions are stored in a so-called *transformation kernel*, $K(\omega, t) = e^{-i\omega t}$ where $i = \sqrt{-1}$. For the continuous case the transformation kernel is a two-dimensional *function*. The ω variable is directly associated with the frequency of the corresponding basis function. For chemometricians it may be easier to look at the FT as a matrix equation (this is used in the *finite* implementations of the algorithm):

$$\boldsymbol{f} = \mathbf{K}\boldsymbol{x}. \tag{2}$$

Each column in **K** is the function $e^{-i\omega t/n}$ where $\omega = \{0, 1, \ldots, n\}$ and $t = \{0, 1, \ldots, n\}$. The matrix dimension of **K** is $[(n+1) \times (n+1)]$.

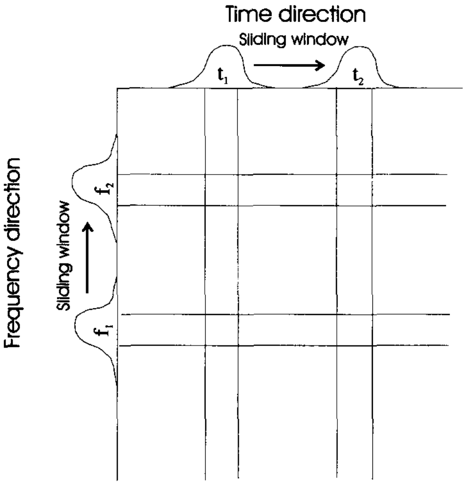
To get the main idea: Wavelets produce a different type of transformation kernel which has some desirable properties that classical FT does not have.

Using the standard discrete Fourier transform we can interpret the signal in terms of *either* the frequency domain *or* the time domain. Often we encounter problems where we would like to have information about both domains at the same time. The human ear for example uses information from both domains; we recognize regions in time with *localized* characteristic frequency distributions. By doing an FT over the whole time domain we are not able to focus on local frequency distribution variations. The FT assumes that the frequency content of the signal is constant throughout the entire signal and thus that it is effectively periodic.

1.3. Short time Fourier transform

In order to improve the time–frequency resolution for FT the short time Fourier transform (STFT) or *Gabor transform* was constructed [28–35]. The idea is very simple: use a window of finite length and move it along the signal in question. For each sliding step an FT on *that local region in time* is performed (see Fig. 1 for an illustration of how STFT is performed).

To avoid spurious frequency components due to the abrupt start and end of the window, we multiply



STFT(time,frequency)

Fig. 1. Here it is illustrated how the time–frequency domain is traversed in STFT. The algorithm can be seen either as a series of ordinary Fourier transforms of small segments along the time axis (here shown as vertical bars) or as a succession of bandlimited filters (horizontal bars).

the time signal $x(t)$ in the region with a Gaussian-like (apodizing) function $g(t)$:

$$y(t) = g(t)x(t). \tag{3}$$

Apodization is a standard technique also used in FT.

The sliding process is seen as several Gaussian-like g functions with different positions along the time axis. Each position of the g function can be controlled by the parameter s (s is the center of g):

$$\mathrm{STFT}(s, \omega) = \int x(t)g(t-s)\mathrm{e}^{-\mathrm{i}\omega t}\mathrm{d}t \tag{4}$$

This is effectively a *convolution* of the signal $x(t)\mathrm{e}^{-\mathrm{i}\omega t}$ with the function $g(t)$. The signal is decomposed in terms of a new set of basis functions, that are windowed versions of the original sine and cosine functions. By performing FT on each of these overlapping windows we are able to plot a time–frequency diagram which is a power spectrum for each time region. How the different STFT basis functions are located in the time frequency domain is illustrated in Fig. 2. Basis functions located in the bottom part in the figure have low-frequency components and those located in the upper part in the figure have high-frequency components. The results from an STFT analysis can be understood as a projection onto each of these basis functions located in time and frequency.

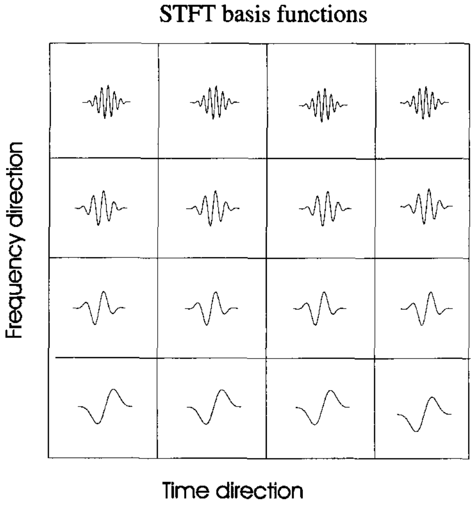


Fig. 2. Here we show the relative positions of localized cosine-like basis function for the STFT in the time–frequency domain.

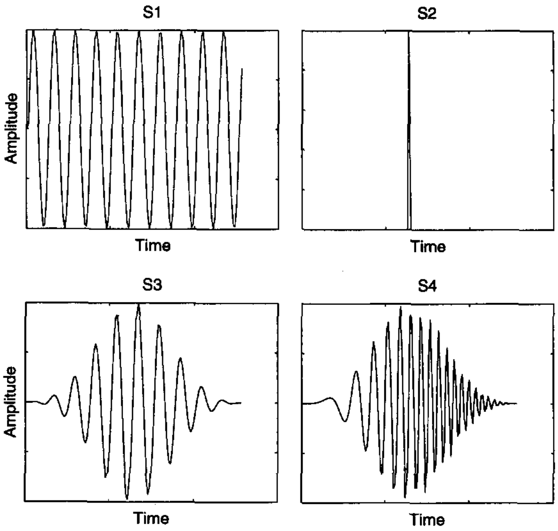


Fig. 3. The time domain representation for the four test signals S_1, ..., S_4 described in the text. S_1 is a sine wave (upper left). S_2 is the delta function (upper right). S_3 is a Gaussian apodized sinewave (lower left). S_4 is the 'chirp' signal which has a linear increase in frequency with respect to time (lower right).

The time–frequency diagrams are sometimes referred to as *signal spectrograms*. The properties of STFT are best understood by looking at some simple examples. We have chosen the following four test signals: (1) standard sine function (S_1), (2) delta function (S_2), (3) windowed sine wave (S_3) and (4) windowed sine wave with linear frequency increase (a 'chirp') (S_4).

These test signals are shown in Fig. 3. Each of these signals will be referred to as S_1, ..., S_4. In Fig. 4 we have computed the total power spectrum of S_1, ..., S_4. In these figures we cannot see the time evolution of the signals. Note that for signal S_1 we see a slight broadening of the signal compared to the windowed sine wave S_3 which is due to the periodization. In the upper right part of the figure the Fourier transform of S_2 is shown. Since the single peak in the delta function contains all frequencies it is not surprising that the spectrum is a single horizontal line, which means that all frequency bins are occupied. The lower right part of the figure shows the total power spectrum for the chirp. Again we see all the frequencies together without any of the time-dependent information retained.

Before actually analysing the STFT analyses of these four signals, it is worth considering how we

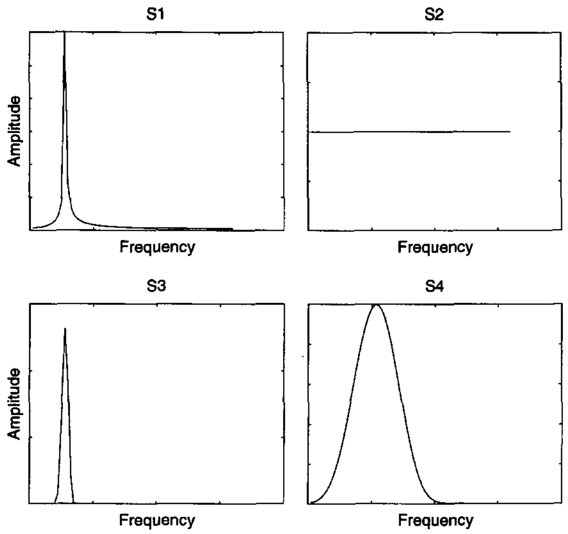


Fig. 4. The power spectra of the four test functions $S_1, \ldots, S_4$ described in the text.

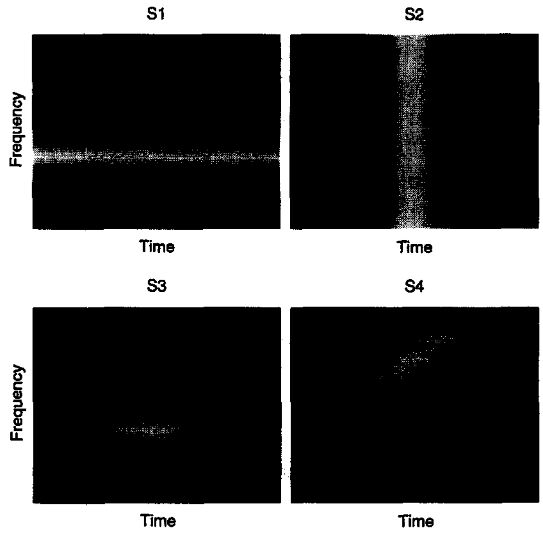


Fig. 5. The STFT results for the four test functions $S_1, \ldots, S_4$ described in the text. Light colour means high value and dark colour means low value.

would expect the time–frequency information to be in each of the four cases. For S_1 we have the same frequency for all time steps and therefore a horizontal line in the signal spectrogram is expected. S_2 has zero value everywhere except for a single point where the ordinate value is one. One should expect contributions from all frequencies close to the single peak and no contributions in all other regions giving a vertical line in the spectrogram. S_3 demonstrates the change in *amplitude* of the sine wave over time, but the frequency is constant in comparison with S_1. We will therefore again expect a horizontal region in time–frequency space, which has an intensity in proportion to the amplitude of the signal. In S_4 the frequency of the signal is increasing linearly with time. The region in time–frequency space should therefore be expected to be slanted. Since the signal is also windowed we expect a corresponding change in the amplitude in the time–frequency spectrum.

STFT's of the signals $S_1, \ldots, S_4$ are shown in Fig. 5. We observe that the results are in agreement with what we would expect the spectrograms would look like. In order to obtain such figures we must fix a certain length of the window and select what type of apodization function should be used (here it is a Gaussian function). Both will be damaging to the *resolution*, either in the time or frequency domain. Resolution is intuitively understood as the degree of detail we are able to see in each domain. A short window length will have a good time resolution i.e. we can see detailed changes happening in time. The resolution in the frequency domain is not so good, however and the number of discrete frequencies we can see is reduced. A large window will have opposite properties: poor resolution in the time domain and good in the frequency domain. It is useful here to imagine the window as a box containing sinusoidal waves. Since the box has a finite length, there must

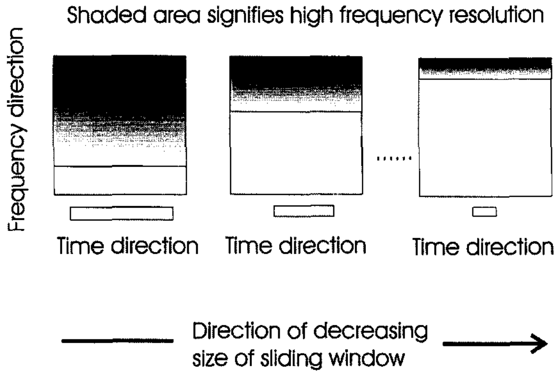

Shaded area signifies high frequency resolution
Frequency direction
......
Time direction
Time direction
Time direction
Direction of decreasing size of sliding window

Fig. 6. Illustration of the dependency of the frequency resolution on the size of the sliding window in STFT. This is a property which is signal-independent. Note that the lower limit of the frequency resolution increases with decreasing window size.

be a lower limit to the frequencies of the waves it can contain. If the wavelength of the wave is too large it cannot fit into the box. If we start out with a large window, there will be a lower limit ν_1 in the signal spectrogram to the resolving power along the frequency direction. The *upper limit* to the frequency resolution must correspond to the sampling frequency of the discrete signal at hand. If a new STFT is performed with a shorter window size, there will be a new lower limit to the frequency resolution, ν_2 where $\nu_2 > \nu_1$ (see Fig. 6). If we have a signal containing spikes in the middle, there will be problems with localizing in *time* those spikes with a large window (blurring). The resolution in the frequency domain, however is very good. Decreasing the window size will reduce the blurring along the time direction but worsen it in the frequency direction. One of the purposes with using the wavelet transform is to improve on the resolution problem. This will, in this case, correspond to selecting different sizes of the sliding window according to the frequency range we wish to investigate.

1.3.1. The uncertainty principle

The relationship between the resolution in time and frequency domains is referred to as the *uncertainty principle*. In Fig. 7 is a simple example which demonstrates this effect for *discrete* Fourier transform. In this figure the δt is the space between the points in time and δf is the equivalent spacing in frequency. This means there will be an ambiguity in the representation of the frequency f which will actually contain all frequencies within $f \pm \delta f$.

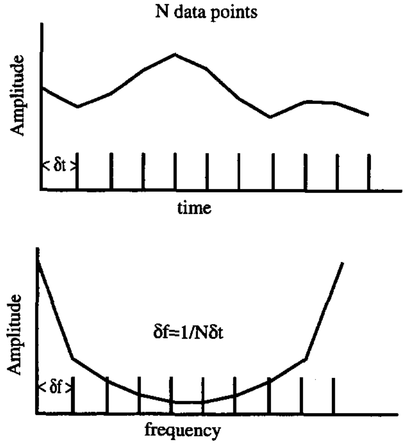


Fig. 7. The effect on the frequency resolution by changing the time resolution in Fourier transform.

Those who are familiar with the Heisenberg uncertainty principle between non-commutating operators in quantum mechanics will see that this is really the same formula:

$$(\Delta t)(\Delta \omega) \geq \frac{1}{4\pi}. \tag{5}$$

where ω is the angular frequency $\omega = 2\pi f$. This is the continuous case. Δt is the resolution in the time domain and $\Delta \omega$ is the resolution (or the 'bandwidth') in the frequency domain. $\Delta \omega$ can be expressed in terms of the window function $g(t)$ and its Fourier transform $\hat{g}(\omega)$:

$$\Delta \omega^2 = \frac{\int \omega^2 |\hat{g}(\omega)|^2 \, \mathrm{d}\omega}{\int |\hat{g}(\omega)|^2 \, \mathrm{d}\omega} \tag{6}$$

Two sinusoids will be discriminated if they are more than $\Delta \omega$ apart. Analogously we can express the time resolution:

$$\Delta t^2 = \frac{\int t^2 |g(t)|^2 \, \mathrm{d}t}{\int |g(\omega)|^2 \, \mathrm{d}t} \tag{7}$$

Two pulses in time can be discriminated if they are more than Δt apart.

For a certain selection of window size and apodizing function we get a resolution number $r_\mathrm{w} = (\Delta t)(\Delta \omega)$. Since the window size is fixed, the time–frequency resolution will be the same for the total time–frequency region. Another way of understanding this is to view the time–frequency domain as divided into regions of the same size. But this is not what we want. It would be better to have a different value of r_w for different positions in the time–frequency domain. This can be accomplished using several STFT analyses with different window sizes as discussed above. We can imagine this as the domain divided into a set of regions that become increasingly narrow in proportion to the frequency. This is similar to what the wavelet transform will be seen to be doing. In contrast to a constant bandwidth as in STFT, WT applies a constant *relative* bandwidth:

$$c = \frac{\Delta \omega}{\omega} \tag{8}$$

It is important to remember that the uncertainty relation is also valid for the wavelet transform.

2. The wavelet transform

There are several ways to look at the wavelet transform. In this tutorial we try to present more than one interpretation of the theory and at the same time try to show that they indeed converge to the same thing.

2.1. The continuous wavelet transform (CWT)

To understand the basic philosophy behind wavelets we will reinterpret the equations used to understand STFT. We stated earlier that a Fourier transform of the time-displaced windowed signal $x(t)g(t-s)$ was performed and that the basis was the complex functions $e^{-i\omega t}$. An alternative formulation is to say that our basis functions are time-displaced windowed complex functions: $g(t-s)e^{-j\omega t}$ and are applied to the full signal $x(t)$. Thus we reformulate our transformation kernel. We now use a simple index system to keep track of all the different functions. One efficient way to classify these functions is in terms of the position, s, of $g(t-s)$ and the frequency ω in $e^{-i\omega t}$. So for each pair of indices (s, ω) (which actually defines the time–frequency domain) we have a complex exponential function.

Let us define a function that uses these two indices: $k_{s,\omega}(t) = g(t-s)e^{-j\omega t}$ which inserted into our STFT equation gives:

$$\mathrm{STFT}(s, \omega) = \int x(t) k_{s,\omega}(t)\,\mathrm{d}t \tag{9}$$

The wavelet transform can be described in a similar fashion, but uses different functions than those described in $k_{s,\omega}$. In the WT the frequency is not used directly, but something that closely resembles it. This variable is referred to as the *scaling variable* a and the *position variable* b variable. In general we are actually using only one function (the scaling function ϕ) to build the different wavelet basis functions. We do this by *scaling* and *translating* ϕ:

$$\phi(ax-b) \tag{10}$$

A large a makes ϕ more narrow and small a makes it more broad. Positive translations b shift ϕ to the right.

The 'mother' wavelet function $\psi(t)$ can be expressed as a linear combinations of the scaling function (see Section 3 for details). In addition, every scaled and translated wavelet function $\psi_{a,b}(t)$ can be written in terms of the 'mother' wavelet:

$$\psi_{a,b} = \frac{1}{|\sqrt{a}|}\psi\left(\frac{t-b}{a}\right), \quad a, b \in \mathscr{R},\ a \neq 0 \tag{11}$$

where $\mathscr{R}$ is the domain of real numbers. The CWT is thus expressed as:

$$\begin{aligned}\mathrm{CWT}(a, b)[f(t)] &= \int_{-\infty}^{\infty} \psi^*_{a,b} f(t)\,dt \\ &= \langle \psi_{a,b} | f(t)\rangle \end{aligned}\tag{12}$$

where $\langle \psi_{a,b} | f(t)\rangle$ is a notation used for inner products or the projection of function $f(t)$ onto the wavelet function $\psi_{a,b}$ (see Section A.3 for more explanation).

We have given no restrictions here to the values of the indices (a, b) except that a cannot be zero. In the CWT the choice of (a, b) is *continuous* over the whole time-scale $((a, b))$ domain.

Given the continuous wavelet transform of a function it is possible to restore the signal perfectly. It has been shown [36] that $f(t)$ can be restored using the following formula:

$$f(t) = \frac{1}{C}\int_0^{\infty}\int_{-\infty}^{\infty} \mathrm{CWT}(a, b)\,\psi_{a,b}\frac{1}{a^2}\,\mathrm{d}a\,\mathrm{d}b \tag{13}$$

where C is

$$C = \int_0^{\infty} \frac{\hat{\psi}^*(\omega)\hat{\psi}(\omega)}{\omega}\,\mathrm{d}\omega \tag{14}$$

The reconstruction can take place only if this constant is defined.

The CWT is a continuous convolution of the wavelet function with the signal over the entire continuum of wavelet scales:

$$\mathrm{CWT}(t, a) = f(t) \otimes W(a, t) \tag{15}$$

It is important to stress this because it is crucial to the operation of the wavelet transform in all its forms (i.e. continuous, discrete and fast). Historically investigators have produced transformations similar to rudimentary wavelet transforms by convolving a basis function with the signal under investigation. The wavelet transform in its most general incarnation does exactly this with a series of scaled versions of the basis function. The transform coefficients produced by this process are merely the correlation of that scale of the basis function at that position with the signal i.e. it is a measure of how similar that basis at that scale and position in time is to the signal. In particular instances where the signal is known to contain features of a specific form, then there may be advantages to using this form as the basis. In general where no prior knowledge of the signal is available, an orthogonal basis is usually preferred in order to avoid redundancy in the transform. The wavelets are simply particular instances of basis functions which are chosen to satisfy certain criteria of benefit in analysis of general signals. One example of such a property is that the wavelet should be bandlimited in both time and frequency domain.

The CWT is a very redundant transform in that it maps a 1D function into a 2D function. The normalization $1/|\sqrt{a}|$ in Eq. (11) is to ensure that the norm for any translated or scaled version of the mother wavelet is the same as the mother wavelet itself i.e. to ensure that all wavelets at all scales have the same area and contain the same energy:

$$\|\psi_{a,b}(t)\| = \|\psi(t)\|. \qquad (16)$$

We have applied the CWT on the four test signals $S_1, \ldots, S_4$. The results from these analyses are shown in Fig. 8.

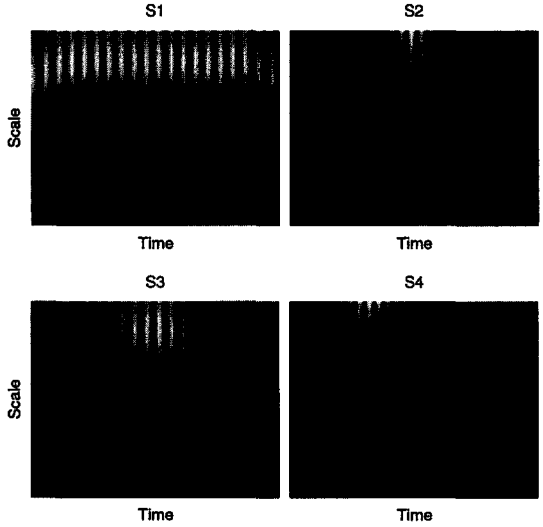


Fig. 8. The CWT results for the four test functions $S_1, \ldots, S_4$ described in the text. Light colour means high value and dark colour means low value.

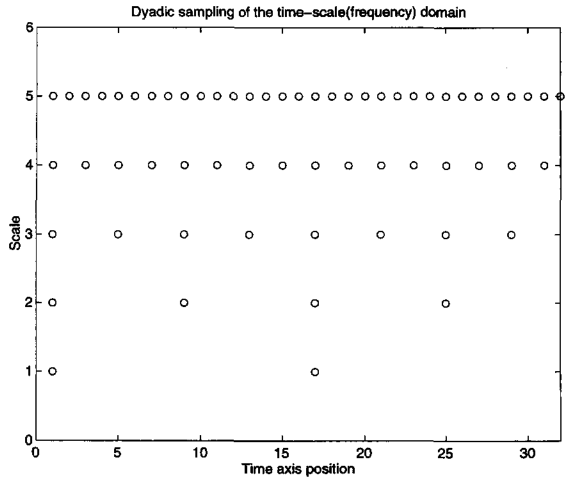


Fig. 9. Dyadic sampling of the time–frequency(scale) domain for wavelets. The dots indicate the center of the wavelet function $\psi_{j,k}(x)$.

2.2. The discrete wavelet transform

For the remainder of this tutorial we will be focusing on discrete rather than continuous methods. The main difference between the discrete and continuous methods is the choice of the possible values for the (a, b) variables. In CWT we do not put any constraints on the choice of these two coordinates and they can in principle map the whole (a, b) plane. In the discrete wavelet transform, however, we restrict the choice of possible (a, b) values as follows:

$$a = a_0^j,\ b = kb_0a_0^j \qquad (17)$$

where k, j are members in the set of all possible negative and positive integers (the set $\mathscr{Z}$). We will concentrate on orthonormal wavelet bases and choose $a_0 = 2$ and $b_0 = 1$. This is called *dyadic* dilations and translations. The dyadic sampling of the time–frequency (scale) domain is demonstrated in Fig. 9. The scaling variable will hereafter be referred to as j and the translation variables referred to as k. j can be interpreted as frequency, but is not identical to it:

$$\left.\begin{array}{l}\text{large } j\\ \text{short scale}\\ \text{high frequency}\end{array}\right\} \Leftrightarrow \text{small steps} \qquad (18)$$

$$\left.\begin{array}{l}\text{small } j\\ \text{long scale}\\ \text{low frequency}\end{array}\right\} \Leftrightarrow \text{large steps} \qquad (19)$$

In Eqs. (18) and (19) we follow the scale numbering by Donoho and collaborators in their WaveLab Toolbox [37].

In Eq. (15) represented the continuous wavelet transform as a continuum in both time and scale. The discrete wavelet transform is merely the discrete representation of this in both axes. This discretization allows the wavelet transform to be numerically computed as a series of discrete convolutions in time at discrete intervals of scale. However, this is a computationally cumbersome procedure and the popular algorithms achieve the same end by using the lower bandwidth of the longer scale wavelets to allow downsampling of the signal. There is no need to convolve a long-scale wavelet with every sample in the signal. The same information is obtained from a wavelet of size, say four times as long, by convolving it with every fourth sample as would be obtained by convolving it with every adjacent sample. This trick results in much shorter convolution sequences at long scales and hence a much faster transform [26].

We now look at the wavelets themselves and the motivation for how they are derived. In order to eliminate ambiguities in interpreting the transform it is desirable that each scale of wavelet covers a separate band of frequencies, such that information contained in any particular wavelet scale is prevented from blurring adjacent scales. Consequently it is necessary to *bandlimit* (limit the upper and lower frequencies covered by) the wavelet. How this is done is illustrated in Fig. 10. In the wavelet transform the time–frequency space is divided into *proportional* width bandpass filters, see Fig. 11. In the Fourier transform and the STFT, however, the time–frequency space is divided into sequences of equal width bandpass filters (previously illustrated in Fig. 2).

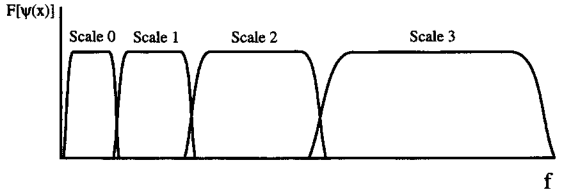


Fig. 10. Demonstration of how the frequency domain is divided in the wavelet transform. Each band corresponds to a scale and the width of the band doubles for each scale.

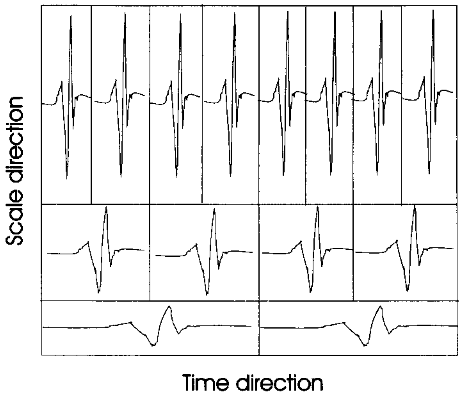


Fig. 11. This figure emphasises the resolution of the time and frequency (or scale) space in wavelet transforms. It is instructive to compare this figure with Fig. 6 which shows how the length of the sliding window in the time domain (of the STFT method) creates different resolutions in the frequency domain. For large windows in time, the frequency domain is well resolved to a certain lower frequency limit. This limit will *increase* with *decreasing* time window length. Assume that the window lengths were dyadically chosen and that we used the frequency region $[f_{n-1}, f_n]$ for the shortest time window (length L_n). f_n is the maximum resolved frequency and f_{n-1} is the minimum resolved frequency for window no. n. The next window will be double the size of the previous window and will have a frequency resolved region of $[f_{n-2}, f_n]$. We are not using the whole of this region but rather $[f_{n-2}, f_{n-1}]$. This is continued until the largest window is reached. In a similar way we can view the time-scale resolution as shown in the present figure. In other words: We want to use the best resolution from each of the sliding time windows.

How do we accomplish the octave division of the frequency space? It is convenient if the shape of these octave filters is symmetric in order to facilitate minimising leakage between them and to minimise phase distortion [38]. The standard way of achieving this symmetry is to use what are referred to as *quadrature mirror filters* (QMF), see Fig. 12. These are matching pairs of low- and high-pass filters whose cut-off characteristics are similar, such that when they are applied simultaneously the resulting bandpass filter has a symmetrical shape on either side of its proportional center frequency. In Fig. 13 we see the corresponding high and low-pass filters as they are applied in the wavelet transform. Note that the high-pass filter is applied to a signal that has been low-pass filtered on a *shorter* scale.

The bottom part of the figure shows the effect of applying both filters simultaneously on a signal. We

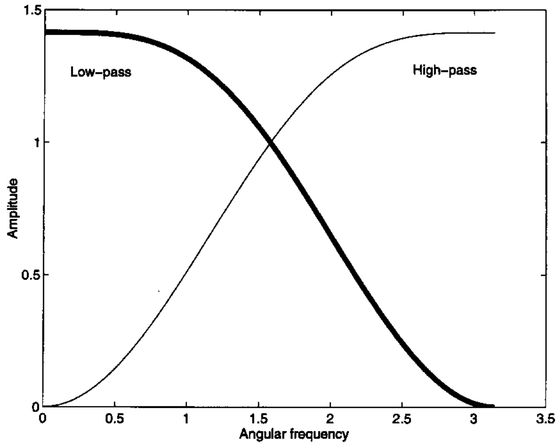


Fig. 12. A matching pair of quadrature mirror filters. Here we show the filters for the Daubechies 4 wavelet and scaling function [47]. The low-pass filter is shown in thick line. The low-pass filter corresponds to the scaling function and the high-pass filter corresponds to the wavelet function.

see that a section of the frequency space has been emphasized. Each of these QMF's are just the scaling and wavelet function in the Fourier domain. Thus, one way to understand the need for several types of wavelet functions is to realize that they correspond to different shapes of the QMF filters in the Fourier domain, i.e. different ways to bandlimit the signal. In the

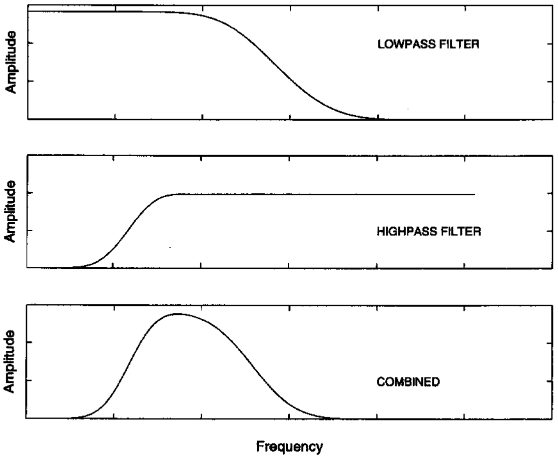


Fig. 13. The power spectrum of the high and low-pass filters which are also known as the quadrature mirror filters (QMF). The high-pass filters in each iteration (middle part) give rise to the wavelet coefficients. Note that when the high and low pass filters are combined they form a bandlimited filter. Here 'combination' is not an addition, but a multiplication.

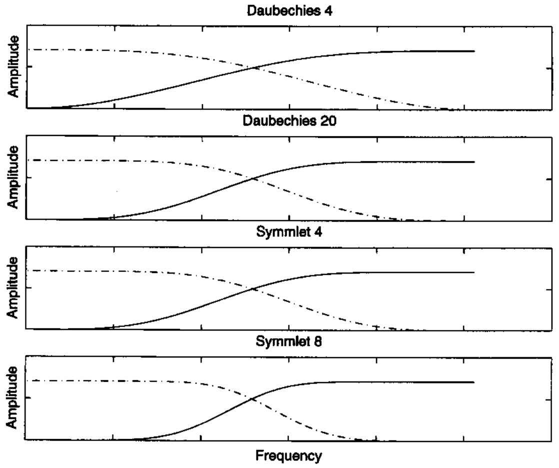


Fig. 14. The high and low-pass (dashed lines) filters for different types of wavelets (Daubechies and Symmlet).

Figs. 14 and 15 we have displayed the high and low-pass filters for Daubechies, Symmlet, Coiflet and Beylkin wavelets [39]. Due to strict conditions [10,40] about what is allowed to be defined as a wavelet there are not that many differences, so we note that the shifting and shape of the filters are only slightly different for the different wavelets. These conditions are related to accuracy in the reconstruction of the original signal, the orthogonality of the wavelets and the conservation of area of the scaling function.

If we can create a wavelet function (and associated scaling function) whose spectral content mimics

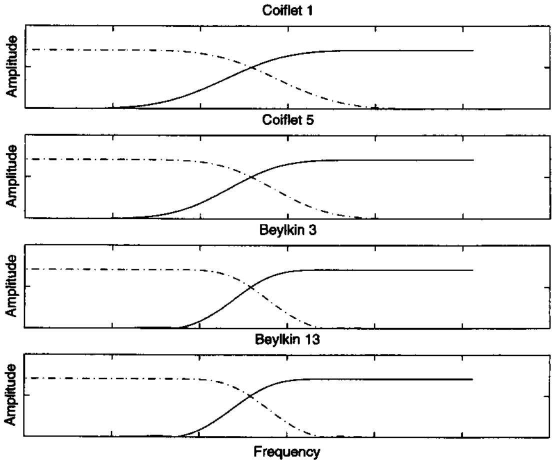


Fig. 15. The high and low-pass (dashed lines) filters for different types of wavelets (Coiflet and Beylkin).

these quadrature mirror filters and which also satisfy the additional requirements for valid wavelet functions then we can use these functions as a basis set for the wavelet transform. The STFT can also be described in terms of low- and high-pass filters but here they divide the frequency axis equally and not proportionally as for the wavelet quadrature mirror filters.

The effect of applying the two filters simultaneously is to slice out a single section of the frequency space. This approach is applied iteratively in the fast wavelet transform, which is most often implemented as the Mallat algorithm [41]. The structure of the algorithm is shown as a diagram in Fig. 16 and can be explained as follows:

(1) The original signal $x(n)$ of length 2^n is first analysed with a high-pass filter (illustrated by a 'H' in Fig. 16). The results of this high-pass filter are the wavelet coefficients at the shortest scale $n-1$. The high-pass filter performs subsampling i.e. the signal size is halved in this process and the length of the wavelet coefficient vector $w(n-1)$ is 2^{n-1}.

(2) The original signal $x(n)$ is low-pass filtered (illustrated by an 'L' in Fig. 16) to produce the vector $x(n-1)$ of size 2^{n-1}.

(3) Goto item 1 and repeat the process for $n=n-1$.

(4) Stop.

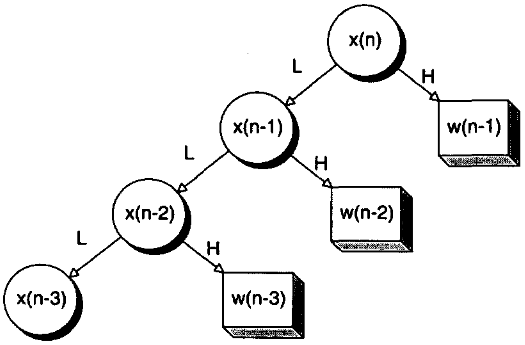


Fig. 16. A tree-diagram of Mallat's fast wavelet algorithm. The original signal (of length 2^n elements) is denoted $x(n)$ in the figure. The symbols H and L stand for high-pass and low-pass filters respectively. The first H filter downsamples the original by 2 and gives the wavelet coefficients $w(n-1)$ for the shortest scale n. The corresponding L filter applied to the original signal $x(n)$ subsamples and creates $x(n-1)$ which has 2^{n-1} elements. $w(n-1)$ also has 2^{n-1} elements. The process is continued until only one element remains.

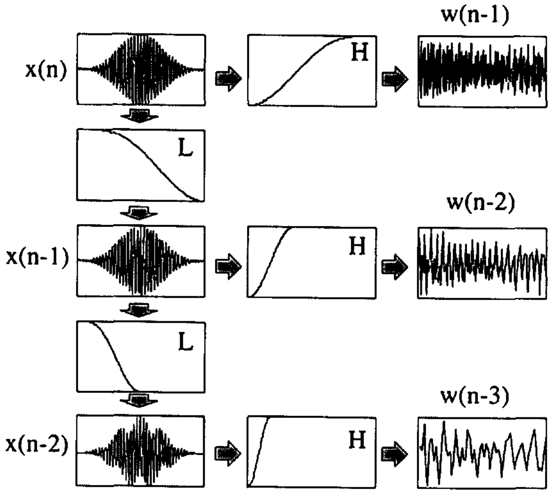


Fig. 17. The Mallat algorithm as described in the previous figure is here displayed in terms of the high (H) and low (L) pass filters (power spectra shown) as they appear when applied to a signal (here the decaying sine wave). The upper left figure is the $x(n)$ signal shown in Fig. 16. The upper middle figure is the shape of the high-pass filter (the wavelet function) for this scale. The upper right figure is the result of the filtering (here the power spectra of the wavelet coefficients). The single figure on the second row from the top shows the low-pass filter which converts $x(n)$ into $x(n-1)$ and subsamples. The remaining subplots are analogous for the lower scales. Here the three highest scales are shown.

We have also shown the algorithm in terms of the filters being used in the Fourier domain, see Fig. 17. Here the original signal (which is a windowed sinusoidal function, upper left part of figure) is first subjected to a high-pass filter (upper middle figure). Daubechies 4 wavelets are being used in this example. The upper right is the wavelet coefficients for the shortest scale $n-1$. In the second subfigure from the top, we have the low-pass filter (corresponds to the scaling function). These steps are continued until the number of elements in the subsampled signal is exhausted.

2.3. Wavelet packets

Once the Mallat algorithm tree has been introduced it is fairly straightforward to explain wavelet packets [27,42–46] (WP) in an intuitive way. Looking at Fig. 16 we see that the ordinary wavelet transform has a tree structure where only one path through the left branch has been taken. In WP, paths through

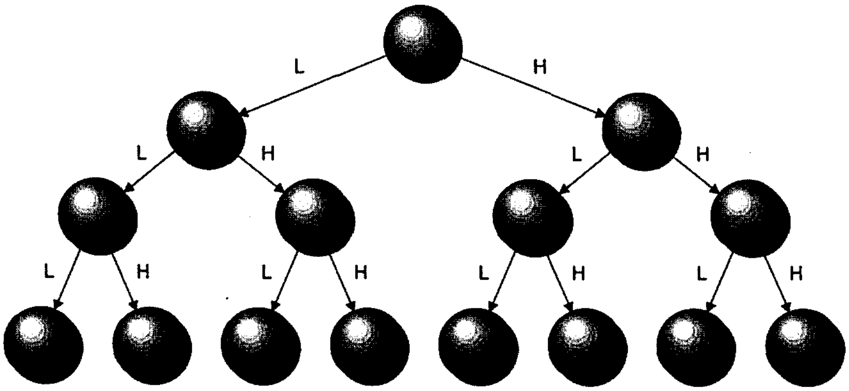


Fig. 18. The wavelet packet approach is a generalization of the Mallat algorithm. High- and low-pass filters can be used on the detail of signals (from high-pass filtering) as well as on the low-pass filtered signals.

the full tree are possible, which makes the wavelet transform a subset of the WP. The general tree structure for the WP transform is shown in Fig. 18. The number of possible wavelet forms is much larger and compared to the Mallat algorithm, WP is much more flexible. WP thus allows the user to zoom in on specific parts on the time–frequency domain in a more detailed way than is possible with the ordinary wavelet transform. Thus, wavelet packets can through arbitrary band splitting have frequency resolutions different from the octave band constant relative bandwidth scheme found in the ordinary wavelet transform. More importantly for general signal analysis is the fact that WP enables data-dependent partition of the time–frequency space.

3. The multiresolution approach to wavelets

3.1. Nested spaces

To obtain an intuitive understanding of what is meant by multiresolution we start with an example. Consider a landscape seen from an airplane high above the ground. Only the major features like mountains, rivers and large geological structures can be seen. We would call this a *coarse* or low level resolution of the landscape. Moving in closer more detail will enter the picture. The next we will see is houses, cars and roads. The next will be humans, chairs and animals. Each stage of resolution can be considered as a *space* which could be imagined to be represented as a linear combination of some suitable basis. Each resolution space is a subset of the resolution space which has a higher resolution. So we could say that ‘mountain space’ is a subset of ‘house–car space’ which is a subspace of ‘human–animal–chair space’ which is a subspace of ‘key–spoon–coin space’.... which is a subspace of ‘atomic space’. Calling each subspace V_j we can write this *nested space* as:

$$\ldots V_{-2} \subset V_{-1} \subset V_0 \subset V_1 \subset V_2 \ldots \quad (20)$$

For our finite approximations we always set the lowest detail space counter to zero.

For present purposes we will be interested in the multiresolution structure of curves or spectra. Intuitively, we view high-frequency noise differently from broad, low-frequency components due to e.g. baseline effects. By employing the multiresolution view, we can build and dismantle curves according to resolution level. So, the wavelet functions are constructed to focus on different resolution details in the signal at different positions. This is possible because of the special structure of the wavelet basis functions. We have seen that the wavelet transform corresponds to an octave partitioning of the frequency domain. Each of the filters correspond to the convolution of a wavelet function in the time domain. Even though we partition different parts of the frequency domain, the wavelet function is the same; it is just the scaling of the function that differs. This means that when we focus on low-frequency components in a spectrum, a stretched version of the wavelet function is used. For high-frequency components we need to use down-scaled wavelet functions. When we pre-

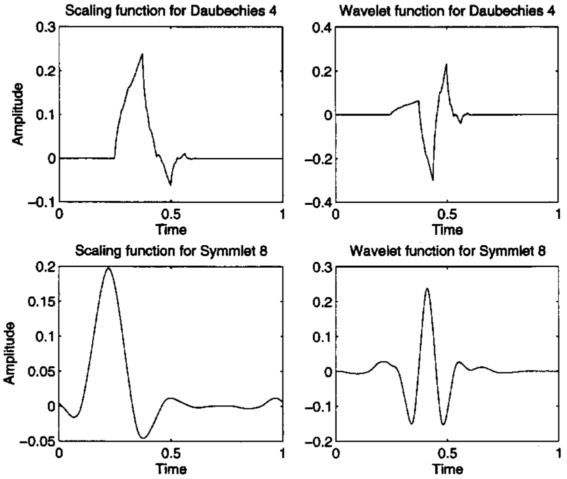


Fig. 19. The scaling functions and wavelets functions for Daubechies 4 and Symmlet 8 bases.

sented the Mallat algorithm, we needed two filters. The high-pass filter corresponds to the wavelet function whereas the low-pass filter corresponds to the scaling function. The nested multiresolution properties of these functions can be derived from their scaling relations. First, the scaling function in the time domain, $\phi(t)$ can be written in terms of itself on a smaller scale $\phi(2t)$ e.g.

$$\phi(t) = c_0\phi(2t) + c_1\phi(2t-1) + c_2\phi(2t-2) + c_3\phi(2t-3) \qquad (21)$$

where the c's are numerical constants that define the scaling function. Eq. (21) is called the *dilation* equation. For the Daubechies 4 scaling function (see Fig. 19 upper left corner) the coefficients are

$$c_0 = \frac{(1+\sqrt{3})}{4}$$
$$c_1 = \frac{(3+\sqrt{3})}{4}$$
$$c_2 = \frac{(3-\sqrt{3})}{4}$$
$$c_3 = \frac{-(\sqrt{3}-1)}{4}. \qquad (22)$$

Second, the corresponding wavelet function has a recursive relation to the scaling function also i.e.

$$\psi(t) = -c_3\phi(2t) + c_2\phi(2t-1) - c_1\phi(2t-2) + c_0\phi(2t-3) \qquad (23)$$

Please note the reversed order of the coefficients. These coefficients are the quadrature mirror filter coefficients discussed earlier. See Fig. 19 upper right corner for the Daubechies 4 wavelet function. The figure also contains the scaling and wavelet function for the Symmlet 8 wavelet which we have found to be useful for analysing vibrational spectra. Because of the recursive relations between the scaling and wavelet function, we can start the description of a function at a long scale and add subsequently more detail. This coarse-to-fine-structure approach is easily demonstrated by using the simplest wavelet, the Haar wavelet:

$$\psi(x) = \begin{cases} 1 & 0 \le x < \frac{1}{2} \\ -1 & \frac{1}{2} \le x < 1 \\ 0 & \text{otherwise} \end{cases} \qquad (24)$$

whose scaling function is:

$$\phi(t) = \begin{cases} 1 & 0 \le t < 1 \\ 0 & \text{otherwise} \end{cases} \qquad (25)$$

The Haar wavelet's relation to the scaling function is:

$$\psi(t) = \phi(2t) - \phi(2t-1). \qquad (26)$$

For the Haar basis an actual formula for the wavelet basis can be given, which unfortunately is not

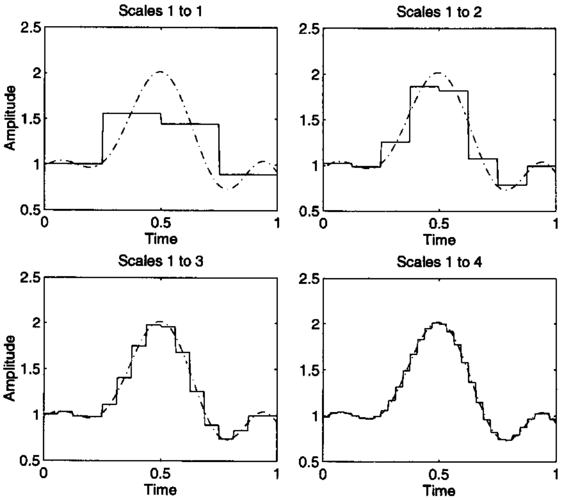


Fig. 20. Demonstration of multiresolution analysis (MRA) reconstruction of the smooth signal given in the dash-dotted line. We start with the coarsest level and add more detail by including new coefficients for new scales. Here the Haar basis is used.

the case in general for most wavelet function. Usually only a relation as in Eq. (23) can be given.

The aim of a wavelet transform is to decompose any signal f into a summation of all the possible wavelet bases at the different scales. This means we can use the following formula for reconstructing f given the wavelet coefficients $c_{mn} = \langle f | \psi_{mn} \rangle$ and the wavelet functions ψ_{mn} used:

$$f(t) = \sum_m \sum_n \langle f | \psi_{mn} \rangle \psi_{mn}. \tag{27}$$

Since the orthogonal wavelet and scaling functions discussed here follow a ladder-like structure, it is possible to reconstruct f gradually i.e. go from a coarse reconstruction and subsequently add more detail from shorter scales. In Fig. 20 we have done just this. Here the Haar wavelet is used to represent a very smooth function. By adding more detail, we will eventually approach the true form of the curve. However, the Haar wavelet is not suitable for description of smooth functions and the number of detail functions that we must add is high (slow convergence). By using a wavelet that is more similar to the function one tries to approximate, fewer scales are needed to get a satisfactory description. This is demonstrated in Fig. 21 where the same function has been approximated by the Symmlet 8 wavelet. In this case, we have an almost perfect representation after just three scales.

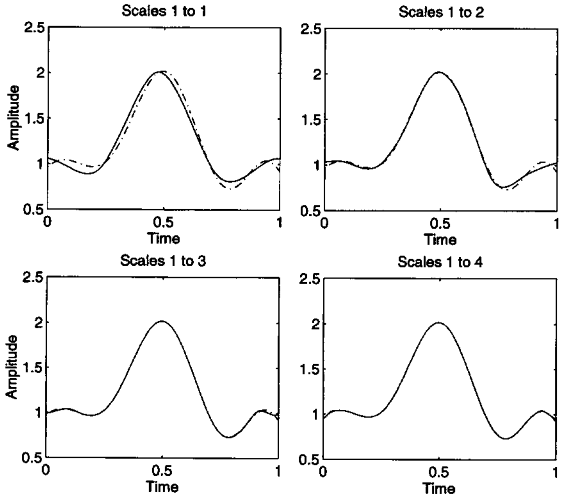


Fig. 21. Demonstration of MRA reconstruction. Same as previous figure, but here the Symmlet 8 basis is used. Note that this wavelet is better since it converges faster than Haar.

3.2. Frames

We showed in Section 3.1 that the discrete transform corresponds to wavelet functions with restrictions on the scaling and translations of the wavelet functions. So far, orthonormal wavelets have been discussed. Frames are a type of wavelet that are not orthonormal. Frames may be viewed as something between the nonredundant orthonormal wavelet transform and the very redundant continuous wavelet transform.

One important criterion that must be satisfied for selecting bases that represent functions is whether we can reconstruct the original function from the derived transform coefficients [47]. Assume we have wavelets that are defined by the following shifts and translations: $a = a_0^m$ and $b = nb_0 a_0^m$ where n, m are positive and negative integers. The space of available wavelet bases is therefore:

$$\psi_{m,n}(t) = a_0^{-m/2} \psi(a_0^{-m} t - nb_0) \tag{28}$$

It is important to note here that the functions $\psi_{m,n}$ are not orthonormal as for the functions described above.

In order to ensure that a wavelet transform using such functions is useful to us we must require that the set of inner products/projections $\langle f(t) | \psi_{m,n} \rangle$ must be able to reconstruct $f(t)$. In fact, we will require that the 'building blocks' $\psi_{m,n}$ can characterize *any* continuous function. We know from working with non-orthogonal or oblique vectors that a reconstruction of a vector v requires the introduction of *reciprocal* or *dual* vectors. We therefore assume that each of the functions (vectors in Hilbert space) $\psi_{m,n}$ have reciprocal functions $\tilde{\psi}_{m,n}$. The formula for the reconstruction is now:

$$f(t) = \sum_n \sum_m \langle \psi_{m,n} | f(t) \rangle \tilde{\psi}_{m,n} \tag{29}$$

Refer to Section A.4 for information about reciprocal/dual bases. To ensure stable reconstruction we must have a correspondence between the space of wavelet coefficients and the original function. If we have, say, two different functions f_1 and f_2 that are 'close' together then we should expect that this 'closeness' is also observed in the space of wavelet coefficients. Mathematicians express this as the two spaces having the same topology. The effect of this is that we can set upper and lower *bounds* on the sum of all inner products between the function and the wavelet frame functions. It is possible to show [47] that these bounds are:

$$A\|f\| \le \sum_m \sum_n |\langle f|\psi_{m,n}\rangle|^2 \le B\|f\| \qquad (30)$$

We have a frame when $A > 0$ and $B < \infty$. The parameters A and B are referred to as *frame bounds*. When they are equal $(A = B)$ it is called a *tight frame*. It is important to realize that not all choices for ψ, a_0, b_0 lead to frames of wavelets. First of all the mother wavelet must be *admissible* i.e if the wavelet $\psi_{m,n}(x) = a_0^{-m/2}\psi(a_0^{-m}x - nb_0)$ constitutes a frame with frame bounds A and B then we must have:

$$\frac{b_0 \ln a_0}{2\pi} A \le \int_0^{\infty} \omega^{-1} \mid \widehat{\psi(\omega)} \mid^2 d\omega \le \frac{b_0 \ln a_0}{2\pi} B \qquad (31)$$

and

$$\frac{b_0 \ln a_0}{2\pi} A \le \int_{-\infty}^{0} \mid \omega \mid^{-1} \mid \widehat{\psi(\omega)} \mid^2 d\omega \le \frac{b_0 \ln a_0}{2\pi} B \qquad (32)$$

In other words, it is related to the integral over the whole Fourier transformed region of the wavelet ($\widehat{\psi(\omega)}$ represents the Fourier transform of $\psi(t)$). This is a necessary, but not sufficient criterion. To obtain sufficient conditions for the functions to constitute a frame, assumptions about the *decay* of the wavelet must be fulfilled. Decay means here that the wavelet goes sufficiently fast towards zero in both the time and frequency domain.

3.2.1. Examples of frames

The second derivative of a Gaussian, also referred to as the Mexican hat:

$$\psi(t) = (1 - t^2)\exp(-t^2/2) \qquad (33)$$

is a frame.

Another example of a wavelet frame is the STFT basis mentioned early in the paper. The basis functions are:

$$g_{m,n} = g(t - ns_0)e^{im\omega t} \qquad (34)$$

If this is to be a frame we know that there must exist frame bounds $A > 0$, $B < \infty$ such that

$$A\int |f(t)|^2 \le \sum_m \sum_n |\langle f(t)|g_{m,n}\rangle|^2 \le B\int |f(t)|^2 \qquad (35)$$

In addition we must have that $\omega_0 t_0 \le 2\pi$ for this basis to be a wavelet frame.

4. Applications

We will briefly discuss some of the areas where the wavelet transform has been found effective and which have relevance to problems discussed in the field of chemometrics. We focus here on the following applications: (1) noise removal (denoising), (2) baseline removal, (3) zero crossing (finding second derivatives), (4) signal compression and (5) wavelet regression.

4.1. Denoising

Noise is a phenomenon that affects all frequencies. Since the signal will tend to dominate the low-frequency components it is expected that the majority of high-frequency components above a certain level are due to noise. This is the underlying philosophy for traditional Fourier filtering where low-pass filters cut off the high-frequency components. Similarly, we can expect small wavelet coefficients at short scales to be mainly due to noise components. The procedure for wavelet denoising will therefore be:

- Apply WT to a noisy signal and obtain the vector ***w*** wavelet coefficients.
- Reduce or remove those elements in ***w*** that are thought to be attributed to noise.
- Apply the inverse WT on ***w*** to obtain a denoised function.

Wavelet denoising methods in general use two different approaches:

- Hard thresholding.
- Soft thresholding.

The hard thresholding philosophy is simply to cut all the wavelet coefficients below a certain threshold

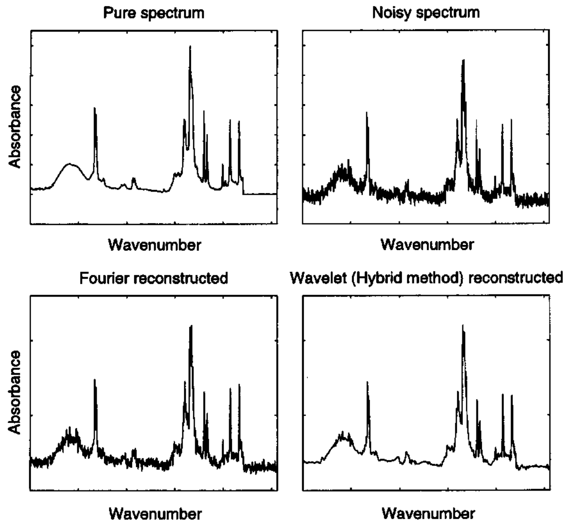


Fig. 22. Illustration of the wavelet denoising procedure applied to a diffuse reflectance infrared spectrum of sodium succinate (256 co-adds using a Bruker IFS28 FT-IR spectrometer). The performance of the wavelet denoising method 'HYBRID' is compared with Fourier denoising based on a smooth low-pass filter. The cut-off point for the Fourier denoising is partially based on manually determining the Fourier domain part dominated by noise (see the text for more details).

to zero. Soft thresholding on the other hand, reduces the value (referred to as 'shrinking') of wavelet coefficients towards zero if they are below a certain value. For a certain wavelet coefficient k at scale j we have:

$$w_k: = \mathrm{sign}|(|w_k| - \lambda)| \tag{36}$$

where sign returns the sign of the wavelet coefficient w_{k} and λ is the threshold value.

In Fig. 22 we have applied a wavelet denoising method referred to as 'HYBRID' [48] to an infrared spectrum (see Section A.5 for a description of the method).

The noise-free infrared spectrum is shown in the upper left corner of the figure. The noisy infrared spectrum is shown in the upper right corner and the wavelet denoised infrared spectrum is in the lower right corner of the figure. The method has been compared with the performance of the Fourier method by applying a low-pass filter to the spectrum (lower left corner). A region in the power spectrum of the signal is specified which most likely contains noise; this is usually located in the upper region of the power spectrum. The maximum amplitude value in this region is used as a cut-off level. At the located cut-off frequency a sigmoid function is used to implement a soft threshold. From visual inspection of the results it is obvious that the wavelet denoising method has performed better than the Fourier method.

4.2. Baseline removal

Whereas noise is mostly located in higher scales, unwanted baseline offsets are usually very-low-frequency components that can be found in the lower scales. Therefore, it is possible to obtain both denoising and 'debasing' by selecting the best middle scales that are dominated by the signal only. We decided to test removal of baselines and denoising on a Raman spectrum of a D-glucose crystal placed on a quartz cover slide. The spectrum was recorded by a Renishaw Raman Microscope with a 633 nm He–Ne laser in a 1 s collect (×50 magnification). In Figs. 23 and 24 we show how the wavelet transform can be used to remove baselines. First we removed the noise (here using a technique referred to as 'VISU', see Section A.6 for explanation of this method). The noisy Raman spectrum is shown in the upper part of Fig. 23 and the denoised spectrum is shown in the lower part of the figure. The signal in this example had $n = 4096$

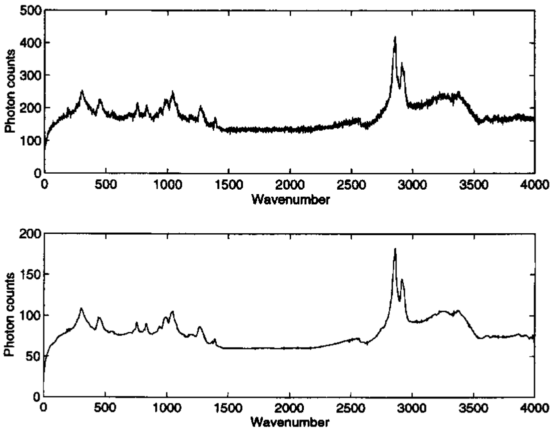


Fig. 23. The original (noisy) sample Raman spectrum of a D-glucose crystal on a quartz slide is shown in the upper part of this figure. The experiment originated from a 1 s collection experiment at ×50 magnification on a Renishaw Raman Microscope. The wavelet denoising technique 'VISU' (see Section A.6 for more details) on the signal produced the result shown in the lower part of the figure.

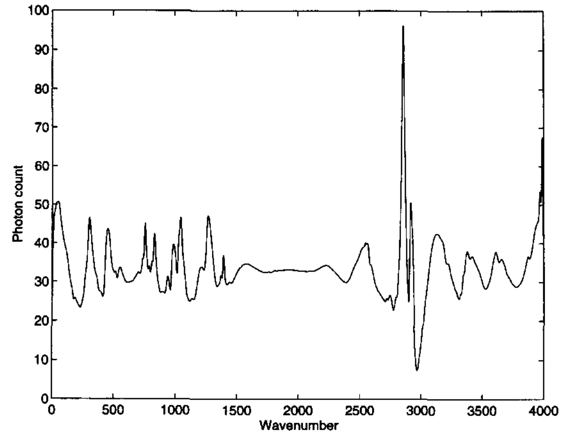


Fig. 24. The wavelet coefficient vector from the denoising process of the Raman signal in the previous figure is further used in the baseline removal step. In order to remove features that originate from baseline effects only, we inserted zero elements in the first four scales (i.e. the scales dominated by low-frequency information). This figure shows the result after inverse wavelet transform of that coefficient vector.

data points (where 1128 of these points are added as zeros to make the signal length a power of 2) and therefore 12 scales were constructed, [0, ..., 11]. The baseline information was found to be located in the four lower scales and these ([0, ..., 3]) were thus removed i.e. set to zero values. In this case the presence of baseline information was determined by visually inspecting the different scales using background information about Raman spectra for the system studied. We know that very broad spectral features were not to be expected to originate from the molecular structure and could therefore safely be ascribed to unwanted baseline and background fluorescence effects. The new wavelet coefficient vector was inverse transformed and we observe that the reconstructed signal (see Fig. 24) has significantly less baseline problems compared to the original signal.

4.3. Zero crossing

One important area of signal analysis is the finding of zero crossings of higher derivatives. This would be a trivial matter if the signals were noise-free. Unfortunately, this is not the case and some smoothing mechanism must be employed. Savitsky-Golay techniques are the most famous for these kinds of problems. The critical points of interest often exist at different scales and linear smoothing will have difficulties with finding the correct points and their locations. Witkin [49] demonstrated a scale-space method which can be used to estimate stable critical

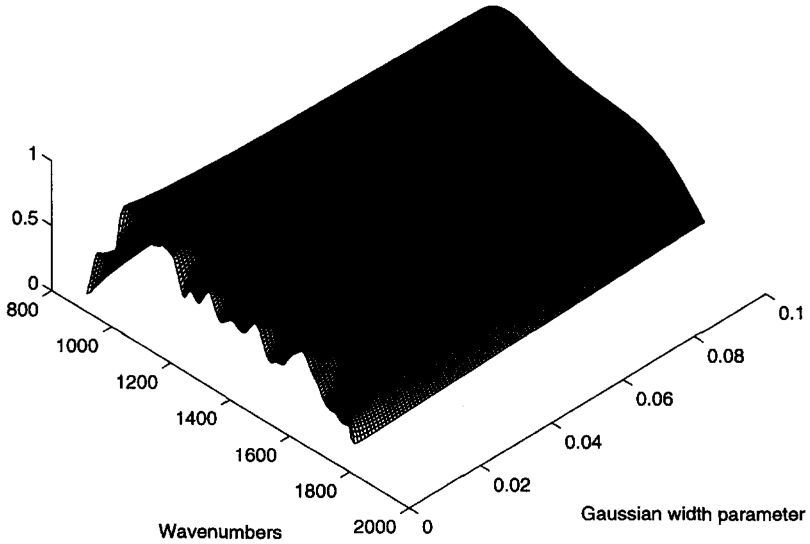


Fig. 25. The effect of convolving a Gaussian wavelet with increasing width to a signal (an FT-IR spectrum). When the width of the Gaussian is small most of the details of the original signal is still present. When the width of the Gaussian is increased, features are lost. The Gaussian function has one important property: The features at long Gaussian widths are not lost when we go to smaller widths.

points. The method is a continuous wavelet transform where the wavelet used is the Gaussian function. Thus, given a function $f(t)$ we convert this to a 2D function $G(t, \sigma)$ where σ is the scale parameter in the Gaussian i.e. *width* of the function:

$$G(t, \sigma) = f(t) \otimes g(t, \sigma) = \int_{-\infty}^{\infty} f(u) g(u, \sigma) \, \mathrm{d}u \tag{37}$$

where

$$g(u, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(\frac{-(t-u)^2}{2\sigma^2} \right) \tag{38}$$

The Gaussian is centered at u and has a standard deviation (width) of σ. The Fourier transform of a Gaussian is also a Gaussian. When we use various convolutions, we effectively remove more of the high-frequency components in proportion of the value of the σ parameter. In this way we will obtain a hierarchy of *abstractions* of this signal at different scales. In Fig. 25 we have plotted all the convolutions (within the region of selected widths) for a certain signal. Note that as we increase the width of the Gaussian wavelet, the convolved curve gets increasingly smooth. At any value of σ, the extrema of the nth derivative are given by the zero-crossings in the $(n+1)$th derivative. We can compute these derivative by:

$$\frac{\partial^n G}{\partial t^n} = f \otimes \frac{\partial^n g}{\partial t^n} \tag{39}$$

The derivatives of the Gaussian function can be obtained analytically. The method can be applied to any higher order derivative, but for most chemical problems, the second derivative, or the extrema of slope (inflection points), is the most important. The nth order zero-crossings in a signal $f(t)$ are thus the points that satisfy:

$$\frac{\partial^n G}{\partial t^n} = 0, \qquad \frac{\partial^{(n+1)} G}{\partial t^{(n+1)}} \neq 0. \tag{40}$$

In Fig. 26 we see the zero-crossings for the second derivative of an FT-IR spectrum. In order to extract qualitative information from such plots we make two assumptions:

(1) The extrema which are located on a common contour over several scales arise from a single underlying event (this is the *identity* assumption).

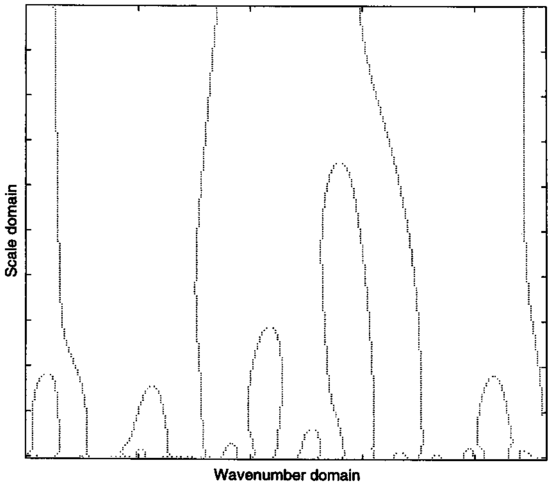


Fig. 26. The zero crossings of the set of convolved spectra shown in the previous figure. Note that the zero-crossing across scales form connected loops. The position within the loop *before* it is further split into subloops defines the stable region for that critical region. In a stable region the sign of the second derivative is constant.

(2) The true location of the event is where the zero-crossing contour approach $\sigma \rightarrow 0$ (the *localization* assumption).

It is thus possible to identify features that most likely correspond to true underlying processes by looking at the stability of an interval over scales.

Instead of using the Gaussian wavelet in a continuous manner, dyadic scale sampling in zero-crossing identification is also possible, as shown by Mallat [50] and Bakshi [51]. The major advantage is a significant reduction in the floating point operations (FLOPS) needed compared to the Gaussian wavelet approach. In addition, the Gaussian is not compactly supported in either domains and a large number of coefficients are required for the corresponding filters.

4.4. Signal compression

In a way, wavelet signal compression is very similar to the process of denoising. Since we want to represent the original signal with as few bits as possible, it is always assumed that high-frequency components are due to noise and can therefore be removed. There is a strict relationship between the number of bytes needed in storage of a function and its smoothness. By storing just the denoised wavelet *coefficients* instead of the original variables, a significant compression ration can be achieved. It should be noted that we restrict the number of possible wavelet coefficients by letting them vary over a finite (preferably small) set of possible values; this process is referred to as quantization.

One particular basis which has been found to be promising in this connection is the B-spline [52,53]. The B-spline is well suited for the development of wavelet bases [54–57]. It is necessary to perform an orthogonalization procedure to make the B-spline bases orthogonal. The B-spline bases are translates of each other, but they are not orthogonal. When the compression is for storage purposes only, the wavelet coefficients are further compressed using lossless methods such as that of Huffman [58]. An additional benefit from a linear compression, is that it is possible to apply the analysis (e.g. a principal components analysis or a partial least squares regression) on the compressed representation only [59–61]. This means one can get an approximate solution to the problem using fewer FLOPS. Since the representation is linear it has been demonstrated [60,61] that it is possible to use the results of the compressed representation and by simple multiplications of wavelet basis matrices be able to extract the results on the *uncompressed* representation. This is a general result and is valid for any type of linear transform matrices used on the data.

4.5. Wavelet regression

In *Fourier regression* [62] a regression model is formed between the frequency components determined in a Fourier analysis and a dependent variable $\boldsymbol{y}$. Let $\mathbf{X}$ be the original data matrix with N spectra (as rows) and M wavelengths. If $\boldsymbol{y}$ is the concentration vector for some chemical component, we need to estimate the regression coefficients $\mathbf{b}$ in the following equation:

$$\boldsymbol{y} = \mathbf{X}\boldsymbol{b} \tag{41}$$

In Fourier regression we replace the original data matrix $\mathbf{X}$ with the projections of each spectrum onto the Fourier basis matrix i.e. we are using the Fourier transform of each spectrum instead of the original spectrum. Assuming smoothness we cut off the highest frequency components. We usually use the power spectrum and not the full complex representation. If the matrix of power spectra is $\mathbf{P}$ we get:

$$\boldsymbol{y} = \mathbf{P}\boldsymbol{b} \tag{42}$$

which is solved in general by

$$\hat{\boldsymbol{b}} = \mathbf{P}^{+}\boldsymbol{y} \tag{43}$$

where the generalized inverse $\mathbf{P}^{+}$ can come from some regression method (e.g. partial least squares). Similarly, we can use the *wavelet transform* as a preprocessing step before doing a regression. There are other approaches to using wavelet in connection with regression [63–67]. It is important to realize that we are performing a *scale-dependent regression*. The advantage with this approach is that we form a regression model where specific scale features can be recognized as important for the prediction. The wavelet transform of a 1D signal is usually coded as a vector $\boldsymbol{w}$ where each scale is stored sequentially. The structure of $\boldsymbol{w}$ is Scale_0 with 1 element, followed by Scale_1 with 2 elements, followed by Scale_2 with 4 elements, ..., followed by Scale_j with 2^j elements. Thus, the total vector $\boldsymbol{w}$ of wavelet coefficients contains M elements i.e. as many as the original signal. As for Fourier regression we obtain the wavelet regression coefficients in general as:

$$\hat{\boldsymbol{b}} = \mathbf{W}^{+}\boldsymbol{y}. \tag{44}$$

There is one very interesting property of the estimated wavelet regression vector $\hat{\boldsymbol{b}}$: it can itself be interpreted as a wavelet transform of a signal. This means we can split up the regression vector into different scales as we do for wavelet coefficient vectors and *observe the regression coefficients at different scales*. It should be emphasized here that the $\boldsymbol{b}$ regression vector from the PLS analysis is one of several possible vectors that can be subjected to such an interpretation. For instance, each PLS loading vector can also be given the scale-space interpretation.

The process for giving PLS variable vectors a scale-space interpretation, is depicted in Fig. 27. To demonstrate the applicability of the approach we have performed wavelet regression to a set of Fourier Transform infrared spectra recorded in our lab (unpublished results). Each spectrum is associated with a different concentration of the antibiotic ampicillin added to a suspension of the bacterium *Staphylococcus aureus*. The wavelet coefficient vector for each spectrum was computed by using the Symmlet 8 wavelet. A PLS analysis of the wavelet transformed data set produced a seven-factor model with 94.6% prediction ability on a separate validation set. The corresponding *b*-regression vector was separated into different scales and interpreted as a wavelet coefficient vector. The results from this analysis are shown in Fig. 28. Scales 4 to 7 around 1250–1800 cm^{-1} on the abscissa seem to contain the largest regression coefficients. We also have quite large coefficients in a region around 2300 cm^{-1} which is due to interference from CO_2. This indicates that a localized region over several scales in the spectrum is very important for the prediction of the ampicillin concentration. Occurrences of a few spikes in the lowest scales suggest that we have some rather low-frequency components that play an important role in the prediction. This could be due to very broad peak features in the spectra or more likely to the *overall* absorbance due to the ampicillin.

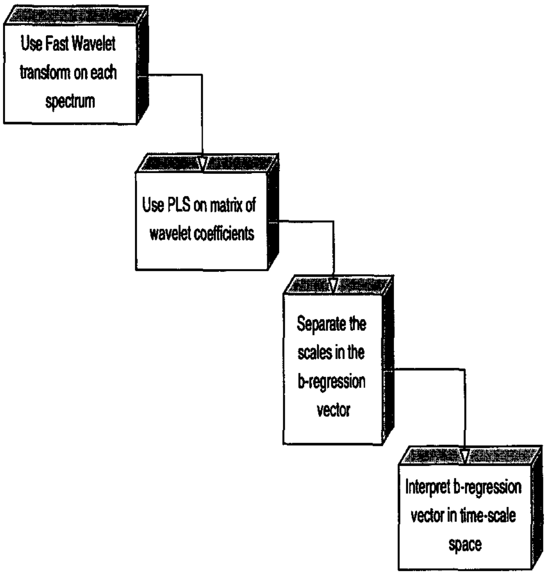

Use Fast Wavelet transform on each spectrum
Use PLS on matrix of wavelet coefficients
Separate the scales in the b-regression vector
Interpret b-regression vector in time-scale space

Fig. 27. A schematic overview of the wavelet regression approach. The process is started by converting all the spectra into wavelet coefficients. The new matrix of coefficients vector for each spectrum is then analysed by PLS (or another suitable regression method). Regression methods produce a *b*-regression vector which is used in the actual prediction from the wavelet coefficient space and into the dependent variable *y* space. This vector is interpreted as a wavelet coefficient vector which is separated into scale regions. It is now possible to interpret the regression coefficients at different scale levels. It is also possible to give loading vectors from PLS the same scale-space interpretation.

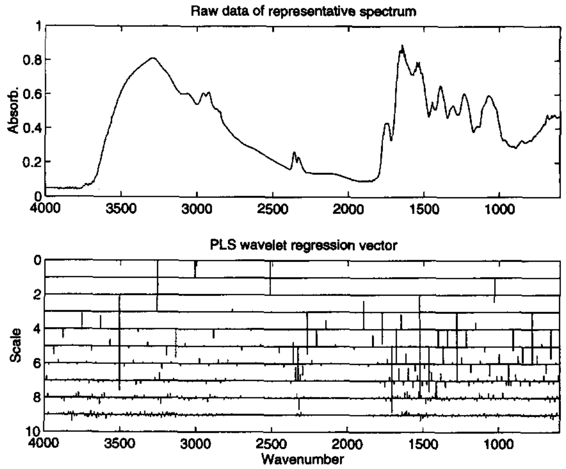


Fig. 28. This figure shows a wavelet regression coefficient vector from a PLS analysis. See a diagrammatical explanation of how this was constructed in previous figure. The upper part of this figure shows a representative infrared spectrum of the data set. The data set is a collection of infrared spectra of the bacterium *Staphylococcus aureus* in a culture to which are added different concentration levels of ampicillin. The lower part of the figure shows the PLS regression coefficients interpreted as a set of wavelet coefficients in a wavenumber-scale space. Using this diagram it is possible to see which of the wavelet basis functions at different scales and wavenumbers are the most important in the PLS regression vector. Different parts of the regression vector correspond to different scales. Note that scales 4 to 7 contain the largest coefficients and are located in the region 1250–1800 cm^{-1} on the abscissa. This region contains a peak which is known to have high selectivity for the ampicillin molecule (at 1767 cm^{-1}). We also have a few low-frequency scales with high regression coefficients in the same region.

It should also be mentioned that this type of analysis strongly suggests that the use of all the wavelet coefficients for the PLS analysis is not necessary; in other words we can perform variable selection in this wavelet regression coefficient space and still be able to retain a stable model.

5. Discussion

It is common practice to compare the fast wavelet transform to the fast Fourier transform (FFT). Since FFT managed to revolutionize the analysis of data in several areas of science it is probable that wavelets will do the same, but it remains to be seen to what degree. So far the majority of papers on wavelets have been focused on the theory and the development of efficient algorithms. The full implication of this theory for analytical chemistry and biochemistry will be evident when more interesting applications can be found. In spite, or perhaps because, of its promised power, the theory and particularly the use of wavelets is not particularly easy to understand for a newcomer to the field. So far the field has been dominated by mathematicians and computer scientists who have had very little interest in making the theory accessible to people in other fields who wish to use the wavelet transform simply as an analysis tool in the same way as they would use the FFT. This scenario is starting to change since we have observed that the number of papers using wavelets is increasing. Unfortunately, these papers are often written by the aforementioned mathematicians in the field and they often tend to see wavelets from the same point of view. This point of view is usually much too abstract to provide the newcomer with a feel for what wavelets do. We believe it is very important to give different interpretations and above all visual aids in the understanding of the results wavelet transforms generate. Even today, many people in chemistry and biology find the theory of Fourier series and Fourier transforms difficult to grasp, although they are beneficial in all areas of analysis [68]. Without intuitive understanding of the real-world effects of the complex mathematics and algorithms, it is in our opinion very difficult to apply any mathematical theory to real-world problems, and this application will be fraught with pitfalls due to ignorance of what the algorithm is actually doing in terms of concepts rather than equations. The correct use of the FFT is littered with these pitfalls such that the FFT cannot be used (as it often is in practice) as a black box without risk of serious errors and/or ambiguities in the analysis. The extra degrees of freedom in the wavelet transform allow extra flexibility but also extra freedom to make mistakes. Mistakes can only really be avoided by a comprehensive intuitive understanding of the basic concepts of the wavelet transform, in the same way that, in applying the FFT to real problems, it is far more important to know what it should look like than how to program the kernel.

Acknowledgements

We thank the Chemicals and Pharmaceuticals Directorate of the UK BBSRC, Glaxo Wellcome, Zeneca Life Science Molecules, Renishaw Transducer Systems and Bruker Spectrospin for financial support. Dr. Michael K. Winson and Dr. Naheed Kaderbhai are thanked for providing the data sets used in this article.

Appendix A

A.1. Compact support

A function $f(t)$ is said to be compactly supported if it is non-zero over a finite interval and zero elsewhere. For instance, a box function as in the Haar basis described in the text is compactly supported whereas the Gaussian function is not.

A.2. Bandlimited

A function $f(t)$ is bandlimited if its Fourier transform $F(f(t)) = \hat{f}(\omega)$ has compact support i.e. $\hat{f}(\omega) = 0$ for $|\omega| > Q$, where Q is some real number.

A.3. The bracket notation for inner products

The bracket notation used in the text between two functions e.g. f and g is defined as:

$$\langle f | g \rangle = \int f(t)^* g(t)\,\mathrm{d}t \tag{A.1}$$

and denotes the inner product between f and g. $f(t)^*$ indicates complex conjugation in case f and g are complex functions. It is customary to think of these functions as vectors in a infinite dimensional space, a so-called Hilbert space. The space of square integrable functions, $L^2(\mathscr{R})$ is an example of a proper Hilbert space.

A.4. Reciprocal = dual = biorthogonal bases

Reciprocal/dual/biorthogonal bases are much used in tensor algebra. It is simply another basis derived from the original non-orthogonal basis which simplifies expressions. Above we mention dual or reciprocal functions and we make use of the deep mathematical analogies between functions and vectors. It is often convenient to represent continuous functions as vectors in an infinite dimensional space (Hilbert space). To simplify the discussion and convey the basic ideas we will here consider ordinary vectors in 2 dimensional space only. Let $\boldsymbol{g}_1$ and $\boldsymbol{g}_2$ be two *basis vectors*. Two vectors $\boldsymbol{u}$ and $\boldsymbol{v}$ are described in relation to $\boldsymbol{g}_1$ and $\boldsymbol{g}_2$:

$$\boldsymbol{u} = u^1\boldsymbol{g}_1 + u^2\boldsymbol{g}_2, \qquad \boldsymbol{v} = v^1\boldsymbol{g}_1 + v^2\boldsymbol{g}_2 \tag{A.2}$$

The inner product $\langle \boldsymbol{u}|\boldsymbol{v}\rangle$ is:

$$\sum_{i=1}^{2}\sum_{j=1}^{2} u^i v^j \langle \boldsymbol{u}_i|\boldsymbol{v}_i\rangle \tag{A.3}$$

which quickly becomes complicated for bigger expressions. A more elegant method is by representing the second vector in another basis $\boldsymbol{g}^j$ (which is derived from the original one). Note that we use subscript for the original basis and superscript for the reciprocal basis. The two bases satisfy the following relations:

$$\langle \boldsymbol{g}_i|\boldsymbol{g}^j\rangle = \delta_{ij} \tag{A.4}$$

where δ_{ij} is the Kronecker delta. The new basis is referred to as the *reciprocal* or the *dual* basis and is orthonormal with respect to the old basis. We now use this to produce the components of a vector $\boldsymbol{v}$ to the basis $\boldsymbol{g}_i$. Each of these components are the inner products $\langle \boldsymbol{v}|\boldsymbol{g}_i\rangle$ (where i is running over all possible dimensions which is here just 2). The vector $\boldsymbol{v}$ can be represented in the reciprocal basis:

$$\boldsymbol{v} = v_1\boldsymbol{g}^1 + v_2\boldsymbol{g}^2 \tag{A.5}$$

Inserting this into the inner product formula above we get (in this example we investigate the inner product $\langle \boldsymbol{v}|\boldsymbol{g}_1\rangle$):

$$\langle \boldsymbol{v}|\boldsymbol{g}_1\rangle = \langle v_1\boldsymbol{g}^1 + v_2\boldsymbol{g}^2|\boldsymbol{g}_1\rangle = v_1\langle \boldsymbol{g}^1|\boldsymbol{g}_1\rangle + v_2\langle \boldsymbol{g}^2|\boldsymbol{g}_1\rangle = v_1 \tag{A.6}$$

Reconstructing $\boldsymbol{v}$ corresponds to $\boldsymbol{v} = v_1\boldsymbol{g}^1 + v_2\boldsymbol{g}^2$ and when we insert for v_i we get

$$\boldsymbol{v} = \sum_{i=1}^{2} \langle \boldsymbol{v}|\boldsymbol{g}_i\rangle \boldsymbol{g}^i. \tag{A.7}$$

This formula is analogous to the reconstruction formula described for frames in Eq. (29).

A.5. The 'HYBRID' method

This is a soft thresholding method that depending on the value

$$e = \frac{\|\boldsymbol{w}^{(j)}\|^2 - n}{n} \tag{A.8}$$

will choose one of two values for the soft threshold τ. Let us introduce $t_{\mathrm{A}} = \sqrt{2\log n}$, where n the number of data points at scale j and $t_{\mathrm{B}} = S(\boldsymbol{w}^{(j)})$ where S is a thresholding based on Stein's unbiased risk estimate (referred to as the 'SURE' method). The details of this method will not be discussed here. Further, let us define the soft thresholding function 'SOFT' which is here written as applied to a single wavelet coefficient $w_k^{(j)}$ at scale j:

$$\mathrm{SOFT}(w_k^{(j)}, \tau) = \mathrm{sign}|(|w_k^{(j)}| - \tau)| \tag{A.9}$$

where τ is the chosen threshold. When we indicate a soft thresholding of a whole wavelet coefficient vector at scale j we write SOFT($\boldsymbol{w}^{(j)}$, τ). The 'HYBRID' method uses one of two different soft thresholds depending on the value of the e parameter:

$$\begin{cases} \mathrm{SOFT}(\boldsymbol{w}^{(j)}, t_{\mathrm{A}}) & \text{if } e < J^{3/2}/n \\ \mathrm{SOFT}(\boldsymbol{w}^{(j)}, \min(t_{\mathrm{A}}, t_{\mathrm{B}})) & \text{otherwise} \end{cases} \tag{A.10}$$

n_j is the number of elements in the wavelet coefficient vector at scale j.

A.6. The 'VISU' method

Here the soft thresholding is applied only to the wavelet coefficient vectors in the index interval $[2^L + 1, J]$ where L is a long scale level which always must be smaller than J (the shortest scale). The threshold value is $t = \sqrt{2\log n}$.

A.7. Wavelet program packages available

There are several program packages available that contain the most popular wavelet methods. In particular, we have emphasized packages that run under MATLAB.

(1) MEGAWAVE: A program package of C-functions written by Stephane Mallat et al. Available for free.

URL: ftp://ftp.cs.nyu.edu/pub/wave/software/.

(2) WaveLab.701: A MATLAB toolbox created by Jonathan Buckheit, Shaobing Chen, David Donoho and Iain Johnstone. Available for free.

URL: http://playfair.stanford.edu/~wavelab/

(3) Rice Wavelet Toolbox for Matlab: A MATLAB toolbox created by Jan Erik Ødegård, Haitao Guo, Ramesh A. Gopinath, Markus Lang and Dong Wei. Available for free.

URL: http://www-dsp.rice.edu/edu/wavelets/

(4) WavBox: A MATLAB toolbox created by Carl Taswell. Available for free.

URL: ftp://simplicity.stanford.edu/pub/taswell/

(5) MathWorks Wavelet Toolbox: A MATLAB toolbox created by MathWorks. Commercial. Contact address/e-mail etc.:

The MathWorks, Inc.
24 Prime Park Way
Natick, MA 01760-1500
e-mail: info@mathworks.com
URL:http://www.mathworks.com/products/wavelettbx.shtml

(6) WaveLib: C program package by M. Bourges-Svenier. Available for free.

URL: http://www-sim.int-evry.FR/bourges/libwave.html

(7) Wavelet Explorer: Program package in Mathematica. Created by Wolfram Research Inc. Commercial.

URL: http://www.wolfram.com

References

[1] J. Morlet, Sampling theory and wave propagation, Proc. 51st. Annu. Meet. Soc. Explor. Geophys., Los Angeles, 1981.

[2] P. Goupillaud, A. Grossmann, J. Morlet, Cycle-octave and related transforms in seismic signal analysis, Geoexploration 23 (1) (1984) 85–102.

[3] P. Goupillaud, A. Grossmann, J. Morlet, Cycle-octave representation for instantaneous frequency-spectra, Geophysics 49 (5) (1984) 669–669.

[4] A. Grossmann, J. Morlet, T. Paul, Transforms associated to square integrable group-representations. 1. General results, J. Math. Phys. 26 (10) (1985) 2473–2479.

[5] A. Grossmann, J. Morlet, T. Paul, Transforms associated to square integrable group-representations. 2. Examples, Ann. Inst. Henri Poincare 45 (3) (1986) 293–309.

[6] A. Grossmann, M. Holschneidėr, R. Kronland-Martinet, J. Morlet, Detection of abrupt changes in sound signals with the help of wavelet transforms, Adv. Electron. Electron Phys. S19 (1987) 289–306.

[7] I. Daubechies, A. Grossmann, Frames in the Bargmann space of entire-functions, Commun. Pure Appl. Math. 41 (2) (1988) 151–164.

[8] Y. Meyer, Wavelets and operators, Rapport CEREMADE, No. 8704, Univ. Paris-Dauphine, Paris, France, 1987.

[9] Y. Meyer, Orthonormal wavelets, Congr. Int. Phys. Math., Swansea, July, 1988.

[10] I. Daubechies, Orthonormal bases of compactly supported wavelets, Comm. Pure Appl. Math. 41 (7) (1988) 909–996.

[11] Y. Meyer, Wavelets. Algorithms and applications, SIAM, Philadelphia, 1993.

[12] Y.T. Chan, Wavelet basics, Kluwer Academic Publishers, Norwell, Massachusetts, 1995.

[13] G. Strang, Wavelets and dilation equations: A brief introduction, SIAM Rev. 31 (4) (1989) 614–627.

[14] O. Rioul, M. Vetterli, Wavelets and signal processing, IEEE Signal Proc. Mag. October (1991) 14–38.

[15] I. Daubechies, S. Mallat, A.S. Willsky, Special issue on wavelet transforms and multiresolution signal analysis introduction, IEEE Trans. Inf. Theory 38 (2 Pt2) (1992) 529–531.

[16] P. Duhamel, P. Flandrin, T. Nishitani, A.H. Tewfik, M. Vetterli, Wavelets and signal-processing; Introduction, IEEE Trans. Signal Proc. 41 (12) (1993) 3213–3215.

[17] J.R. Williams, K. Amaratunga, Introduction to wavelets in engineering, Int. J. Num. Methods Eng. 37 (14) (1994) 2365–2388.

[18] P.M. Bentley, J.T.E. Mcdonnell, Wavelet transforms: An introduction, Electron. Commun. Eng. J. 6 (4) (1994) 175–186.

[19] A. Graps, An introduction to wavelets, IEEE Comput. Sci. Eng. 2 (2) (1995) 50–61.

[20] M. Akay, Wavelet transforms in biomedical-engineering: Introduction, Ann. Biomed. Eng. 23 (5) (1995) 529–530.

[21] A. Antoniadis, G. Oppenheim (Eds.), Wavelets and Statistics, vol. 103, Lecture Notes in Statistics, Springer Verlag, New York, 1995.

[22] G.W. Wornell, in: A.V. Oppenheim (Ed.), Signal Processing with Fractals. A Wavelet Based Approach, Prentice Hall Signal Processing Series, Prentice Hall, Upper Saddle River, NJ, 1996.

[23] A. Cohen, R.D. Ryan, Wavelets and Multiscale Signal Processing, vol. 11, Applied Mathematics and Mathematical Computation, Chapman and Hall, London, 1995.

[24] M. Vetterli, J. Kovacevic, Wavelets and Subband Coding, Prentice Hall, Englewood Cliffs, NJ, 1995.

[25] A.N. Akansu, R.A. Haddad, Multiresolution Signal Decomposition. Transforms. Subbands. Wavelets, Academic Press, San Diego, CA, 1992.

[26] M.V. Wickerhauser, Adapted Wavelet Analysis from Theory to Software, A.K. Peters, Wellesley, MA, 1994.

[27] B. Walczak, D.L. Massart, Noise suppression and signal compression using the wavelet packet transform, Chemom. Intell. Lab. Syst. 36 (2) (1997) 81–94.

[28] M.G. Amin, K. Difeng, Short-time Fourier-transforms using cascade filter structures, IEEE Trans. Circuits Syst. II-analog Dig. Signal Proc. 42 (10) (1995) 631–641.

[29] M.S. Wang, Z. Bao, Modified short-time Fourier-transform, Opt. Eng. 34 (5) (1995) 1333–1337.

[30] S.H. Nawab, E. Dorken, A framework for quality versus efficiency tradeoffs in STFT analysis, IEEE Trans. Signal Proc. 43 (4) (1995) 998–1001.

[31] S. Wada, Generalized short-time Fourier-transforms based on nonuniform filter bank structure, IEICE Trans. Fundam. Electron. Commun. Comput. Sci. E78A (3) (1995) 431–436.

[32] F.T.S. Yu, G.W. Lu, Short-time Fourier-transform and wavelet transform with Fourier domain processing, Appl. Opt. 33 (23) (1994) 5262–5270.

[33] G.S. Cunningham, W.J. Williams, Fast implementations of generalized discrete time–frequency distributions, IEEE Trans. Signal Proc. 42 (6) (1994) 1496–1508.

[34] W. Chen, N. Kehtarnavaz, T.W. Spencer, An efficient recursive algorithm for time-varying Fourier-transform, IEEE Trans. Signal Proc. 41 (7) (1993) 2488–2490.

[35] P.J. Kootsookos, B.C. Lovell, B. Boashash, A unified approach to the STFT, TFDS and instantaneous frequency, IEEE Trans. Signal Proc. 40 (8) (1992) 1971–1982.

[36] A. Grossmann, J. Morlet, Decomposition of Hardy functions into square integrable wavelets of constant shape, SIAM J. Math. Anal. 15 (1984) 723–736.

[37] J. Buckheit, S. Chen, J. Crutchfield, D. Donoho, H. Gao, I. Johnstone, E. Kolaczyk, J. Scargle, K. Young, T. Yu. Wavelab. `http://playfair.Stanford.EDU/wavelab/`, 1996.

[38] M. Vetterli, C. Herley, Wavelets and filter banks: Theory and design, IEEE Trans. Signal Proc. 40 (1992) 2207–2231.

[39] G. Beylkin, On the representation of operators in bases of compactly supported wavelets, SIAM J. Num. Anal. 29 (6) (1992) 1716–1740.

[40] G. Strang, Wavelets and dilation equations: A brief introduction, SIAM Rev. 31 (4) (1989) 614–627.

[41] S. Mallat, A theory for multiresolution signal decomposition: The wavelet representation, IEEE Trans. Pattern Anal. Machine lntell. 11 (7) (1989) 674–693.

[42] R. Colfman, Y. Meyer, M.V. Wickerhauser, Wavelet analysis and signal processing, in: M.B. Ruskai, G. Beylkin, R. Coifman, I. Daubechies, Y. Meyer, L. Raphael (Eds.), Wavelets and their Applications, Jones and Bartlett, New York, 1992, pp. 153-178.

[43] R. Coifman, Y. Meyer, M.V. Wickerhauser, Size properties of wavelet packets, in: M.B. Ruskal, G. Beylkin, R. Coifman, I. Daubechics, Y. Meyer, L. Raphael (Eds.), Wavelets and their Applications, Jones and Bartlett, New York, 1992, pp. 453–470.

[44] M.V. Wickerhauser, Acoustic signal processing with wavelet packets, in: C.K. Chui (Ed.), Wavelets: A Tutorial in Theory and Applications, Academic Press, New York, 1992, pp. 679–700.

[45] B. Jawerth, W. Sweldens, An overview of wavelet-based multiresolution analyses, SIAM Rev. 36 (3) (1994) 377–412.

[46] N. Hessnielsen, M.V. Wickerhauser, Wavelets and time–frequency analysis, Proc. IEEE 84 (4) (1996) 523–540.

[47] I. Daubechies, Ten lectures on wavelets, vol. 61 of CBMS-NSF Regional Conference Series in Applied Mathematics, SIAM, Philadelphia, PA, 1992.

[48] D. Donoho, De-noising by soft-thresholding, IEEE Trans. Inf. Theory 41 (3) (1995) 613–627.

[49] A.P. Witkin, Scale-space filtering, in: A. Bundy (Ed.), Proc. Int. Joint Conf. Artificial Intell., Karlsruhe, 1983, pp. 1019-1022.

[50] S. Mallat, Zero crossing of a wavelet transform, IEEE Trans. Inf. Theory 37 (1991) 1019–1033.

[51] B.R. Bakshi, G. Stephanopoulos, Representation of process trends. III. Multiscale extraction of trends from process data, Comput. Chem. Eng. 18 (4) (1994) 267–302.

[52] B.K. Alsberg, O. M Kvalheim, Compression of *n*th-order data arrays by B-splines., Theory. J. Chemom. 7 (part 1) (1993) 61–73.

[53] B.K. Alsberg, E. Nodland, O.M. Kvalheim, Compression of *n*th-order data arrays by B-splines. Part 2. Application to second-order FT- or IR spectra, J. Chemom. 8 (2) (1994) 127–146.

[54] T. Berger, J.O. Stromberg, Exact reconstruction algorithms for the discrete wavelet transform using spline-wavelets, Appl. Comput. Harmonic Anal. 2 (4) (1995) 392–397.

[55] S.K. Yang, C.H. Cooke, Data compression based on the cubic B-spline wavelet with uniform 2-scale relation, Math. Comput. Model. 23 (7) (1996) 73–88.

[56] P.L. Ainsleigh, C.K. Chui, A B-wavelet-based noise-reduction algorithm, IEEE Trans. Signal Proc. 44 (5) (1996) 1279–1284.

[57] X.Q. Lu, J.Y. Mo, Spline wavelet multiresolution analysis for high noise digital signal processing in ultraviolet-visible spectrophotometry, Analyst 121 (8) (1996) 1019–1024.

[58] J.A. Storer, Data Compression. Methods and Theory, Computer Science Press, Rockville, MD, 1988.

[59] B.K. Alsberg, Representation of spectra by continuous functions, J. Chemom. 7 (1993) 177–193.

[60] B.K. Alsberg, O.M. Kvalheim, Speed improvement of multivariate algorithms by the method of postponed basis matrix multiplication. Part I. Principal component analysis, Chemom. Intell. Lab. Syst. 24 (1994) 31–42.

[61] B.K. Alsberg, O.M. Kvalheim, Speed improvement of multivariate algorithms by the method of postponed basis matrix multiplication. Part II. Three-mode principal component analysis, Chemom. Intell. Lab. Syst. 24 (1994) 43–54.

[62] A. Krzyzak, Identification of discrete Hammerstein systems by the Fourier-series regression estimate, Int. J. Syst. Sci. 20 (9) (1989) 1729–1744.

[63] J. Engel, A simple wavelet approach to nonparametric regression from recursive partitioning schemes, J. Multivariate Anal. 49 (2) (1994) 242–254.

[64] A.M. Herzberg, W.N. Traves, An optimal experimental-de-

sign for the Haar regression-model, Can. J. Stat. 22 (3) (1994) 357–364.

[65] J. Sjoberg, Q.H. Zhang, L. Ljung, A. Benveniste, B. Delyon, P.Y. Glorennec, H. Hjalmarsson, A. Juditsky, Nonlinear black-box modeling in system-identification: A unified overview, Automatica 31 (12) (1995) 1691–1724.

[66] T. Ogden, E. Parzen, Data dependent wavelet thresholding in non-parametric regression with change-point applications, Comput. Stat. Data Anal. 22 (1) (1996) 53–70.

[67] G.P. Nason, Wavelet shrinkage using cross-validation, J. R. Stat. Soc. Ser. B 58 (2) (1996) 463–479.

[68] D.B. Kell, The principles and potential of electrical admittance spectroscopy: An introduction, in: A.P.F. Turner, I. Karube, G.S. Wilson (Eds.), Biosensors: Fundamentals and Applications, Oxford University Press, 1987, pp. 427–468.