Appendix 2

Wavelets and Dilation Equations: A Brief Introduction

Siam Review 31 (1989) 613-627

Gilbert Strang

Abstract

Wavelets are new families of basis functions that yield the representation $f(x) = \sum b_{ij} W(2^j x - k)$. Their construction begins with the solution $\phi(x)$ to a dilation equation with coefficients c_k. Then W comes from ϕ, and the basis comes by translation and dilation of W. It is shown in Part 1 how conditions on the c_k lead to approximation properties and orthogonality properties of the wavelets. Part 2 describes the recursive algorithms (also based on the c_k) that decompose and reconstruct f. The object of wavelets is to localize as far as possible in both time and frequency, with efficient algorithms.

Wavelets are based on translation ($W(x) \to W(x+1)$) and above all on dilation ($W(x) \to W(2x)$). It is remarkable how long it has taken for "*dilation equations*" to be mentioned beside differential equations and difference equations. True, they are hardly in the same league. But ideas about wavelets are coming fast. The mathematics is attractive and several important applications seem to fit—I hope this survey will be helpful. You should know that its author is neither an expert nor an evangelist.

The goal is a new way to represent functions—especially functions that are local in time and frequency (or space and wave number). Compare with Fourier series. Sines and cosines are perfectly local in frequency, but global in x or t. A short pulse has slowly decaying coefficients that are hard to measure. To reconstruct the pulse, a Fourier series depends heavily on cancellation. The whole of Fourier analysis, relating properties of functions to properties of coefficients, is made difficult (some say interesting) by the nonlocal support of $\sin x$.

In achieving local support we lose the greatest property of the basis $\{e^{inx}\}$. With respect to a wavelet basis the differentiation operator is not diagonal. Wavelets are not eigenfunctions of $\partial/\partial x$, and frequencies are mixed up. The uncertainty principle imposes limits on what is possible in x and ξ together. The commutator $(\partial/\partial x)(\partial/\partial \xi) - (\partial/\partial \xi)(\partial/\partial x)$ is a multiple of the identity (since $(\partial/\partial x)(xu) - x(\partial u/\partial x) = u$), so we cannot diagonalize both operators. But a good "microlocalization" leaves $\partial/\partial x$ nearly diagonal, and at the same time nearly diagonalizes $\partial/\partial \xi$ (which is multiplication by x). To connect dilation with multiplication by x, differentiate $f(cx)$ with respect to c at $c = 1$.

The second important property of $\{e^{inx}\}$ is orthogonality. That can be saved. Wavelets can be made orthogonal to their own dilations (as well as their translations). Then $\int W(x)W(2^j x - k)dx = 0$ for all integers j and k. The wavelet basis has two indices, in which k is translation and j is dilation or compression. It suggests multigrid. A wavelet expansion $\sum b_{jk} W_{jk}(x)$ is a *multiresolution* of $f(x)$, in which b_{jk} carries information about f near $\xi = 2^j$ and $x = 2^{-j}k$. The sum on k is the detail at the scaling level $h = 2^{-j}$.

Orthogonality is not easy to achieve with local support. Truncated at zero and 2π, a sine wave $\phi(x)$ is orthogonal to $\phi(2x)$ but not to $\phi(4x)$. The "windowed Fourier transform" combines smoothness with local support by bringing $e^{i\xi x}$ gradually to zero, but it is not fully satisfactory. The price of orthogonality with compact support is irregular basis functions. We live with these wavelets by doing all computations recursively (this subject is recursion heaven). And it is important to recognize that orthogonality and even linear independence (!) are not essential in the representation of functions. Wavelets need not be orthogonal.

This brief introduction cannot do justice to the applications. Nor can we attempt a proper history—it would be mostly in French. The idea of wavelets grew out of seismic analysis. Their development has been led by Yves Meyer, whose book will describe a new chapter in harmonic analysis (connecting to work of Calderòn, Grossmann, Morlet, Coifman, Weiss, and many others). The interest in wavelets is both pure and applied—like the interest in splines.

Part 1 of this paper establishes the properties of wavelets—approximation through Condition A and orthogonality through Condition O. Since we never see wavelets as functions (only recursively), their properties have to be discovered indirectly. We absolutely need these properties in order to have any idea what the algorithms are producing. Then Part 2 begins with a piecewise constant example (ϕ is a box function, the wavelet is Haar's). The example reveals a lot with no deep analysis. You could go directly to Part 2, about algorithms, and then return to dilation equations.

1. Dilation equations: Construction of ϕ. *The basic dilation equation is a two-scale difference equation*:

$$\phi(x) = \sum c_k \phi(2x - k). \tag{A.1}$$

We look for a solution normalized by $\sum \phi \, dx = 1$. The first requirement on the coefficients c_k comes from multiplying by 2 and integrating:

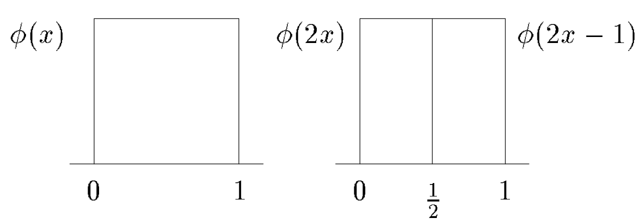
$$2\int \phi \, dx = \sum c_k \int \phi(2x-k)\, d(2x-k) \quad \text{yields} \quad \sum c_k = 2.$$

Uniqueness of ϕ is ensured by $\sum c_k = 2$. A smooth solution is not ensured. For a striking example, set $c_0 = 2$:

$$\text{The delta function} \quad \phi = \delta \quad \text{satisfies} \quad \delta(x) = 2\delta(2x).$$

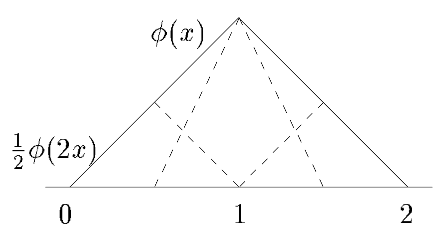
That dilation of δ is unfamiliar (but somehow very pleasing). For other c's, spline functions appear:

Box function:



$c_0 = 1$
$c_1 = 1$

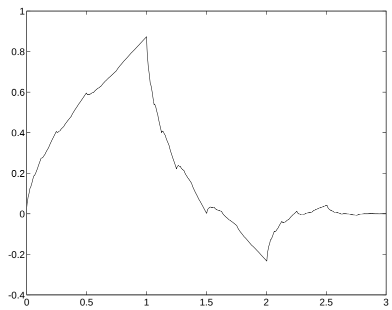
Hat function:



$c_0 = \frac{1}{2}$
$c_1 = 1$
$c_2 = \frac{1}{2}$

We now outline three constructions of the "scaling function" ϕ. Those constructions display very clearly the mathematics of dilation. Then we turn to wavelets, their properties and their purpose. A wavelet $W(x)$ is a second combination (involving the same recursion coefficients c_k) of the translates $\phi(2x - k)$.

Construction 1. Iterate $\phi_j(x) = \sum c_k \phi_{j-1}(2x-k)$ with the box function as $\phi_0(x)$. When $c_0 = 2$ the boxes get taller and thinner, approximating the delta function. For $c_0 = c_1 = 1$ the box is invariant: $\phi_j = \phi_0$. For $\frac{1}{2}$, 1, $\frac{1}{2}$ the hat function appears as $j \to \infty$, and $\frac{1}{8}, \frac{4}{8}, \frac{6}{8}, \frac{4}{8}, \frac{1}{8}$ yields the cubic B-spline. An example that will be important (an inspiration of Daubechies—we propose the notation D_4) has coefficients $\frac{1}{4}(1+\sqrt{3})$, $\frac{1}{4}(3+\sqrt{3})$, $\frac{1}{4}(3-\sqrt{3})$, and $\frac{1}{4}(1-\sqrt{3})$:



This scaling function D_4 leads to orthogonal wavelets. *It is not as smooth as it looks.* Note that the Weierstrass nowhere differentiable function, which is $\sum b^n \cos(3^n x)$, involves dilation by 3. So does de Rham's function, which has $c_k = \frac{2}{3}, \frac{1}{3}, 1, \frac{1}{3}, \frac{2}{3}$ adding to 3. Resnikoff has found a connection between Weierstrass functions and wavelets.

Construction 2. The second construction takes the Fourier transform of (1):

$$\begin{aligned}\widehat{\phi}(\xi) &= \sum c_k \int \phi(2x-k) e^{i\xi x}\, dx \\ &= \frac{1}{2}\left(\sum c_k e^{ik\xi/2}\right) \int \phi(y) e^{iy\xi/2}\, dy = P\left(\frac{\xi}{2}\right) \widehat{\phi}\left(\frac{\xi}{2}\right).\end{aligned} \tag{A.2}$$

The symbol $P(\xi) = \frac{1}{2}\sum c_k e^{ik\xi}$ is the crucial function in this theory. Note that $P(0) = 1$. Now repeat (2) at $\xi/2$, $\xi/4$, ... and recall $\widehat{\phi}(0) = \int \phi\, dx = 1$:

$$\widehat{\phi}(\xi) = \left[\prod_1^n P\left(\frac{\xi}{2^j}\right)\right] \widehat{\phi}\left(\frac{\xi}{2^N}\right) \quad \text{approaches} \quad \prod_1^\infty P\left(\frac{\xi}{2^j}\right). \tag{A.3}$$

For $c_0 = 2$ we find $P \equiv 1$ and $\widehat{\phi} \equiv 1$, the transform of the delta function. For $c_0 = c_1 = 1$ the products of the P's are geometric series:

$$P\left(\frac{\xi}{2}\right) P\left(\frac{\xi}{4}\right) = \frac{1}{4}\left(1 + e^{i\xi/2}\right)\left(1 + e^{i\xi/4}\right) = \frac{1 - e^{i\xi}}{4\left(1 - e^{i\xi/4}\right)}.$$

As $N \to \infty$ this approaches the infinite product $\left(1 - e^{i\xi}\right)/(-i\xi)$. This is $\int_0^1 e^{i\xi x}\, dx$, the transform of the box function. The hat function comes from squaring $P(\xi)$ which by (3) also squares $\widehat{\phi}(\xi)$. (Multiplication of P's is $\frac{1}{2}$ times convolution of c's). The cubic B-spline comes from squaring again.

Construction 3. This construction of ϕ works directly with the recursion. Suppose ϕ is known at the integers $x = j$. The recursion (1) gives ϕ at the half-integers. Then it gives ϕ at the quarter-integers, and ultimately at all dyadic points $x = k/2^j$. This is fast to program. *All good wavelet calculations use recursion.*

The values of ϕ at the integers come from an eigenvector. With the four Daubechies coefficients, set $x = 1$ and $x = 2$ in the dilation equation (1) and use the fact that $\phi = 0$ unless $0 < x < 3$:

$$\begin{aligned} \phi(1) &= \frac{1}{4}\left(3+\sqrt{3}\right)\phi(1) + \frac{1}{4}\left(1+\sqrt{3}\right)\phi(2) \\ \phi(2) &= \frac{1}{4}\left(1-\sqrt{3}\right)\phi(1) + \frac{1}{4}\left(3-\sqrt{3}\right)\phi(2). \end{aligned} \tag{A.4}$$

This is $\phi = L\phi$, with matrix entries $L_{ij} = c_{2i-j}$. Compare with c_{i-j} for an ordinary difference equation. The eigenvalues are 1 and $\frac{1}{2}$. The eigenvector for $\lambda = 1$ has components $\phi(1) = \frac{1}{2}\left(1+\sqrt{3}\right)$ and $\phi(2) = \frac{1}{2}\left(1-\sqrt{3}\right)$, which are the heights on our graph of D_4. The other eigenvalue $\lambda = \frac{1}{2}$ means that the recursion can be differentiated: $\phi'(x) = \sum c_k 2\phi'(2x-k)$ leads similarly to $\phi'(1)$ and $\phi'(2)$. In some weak sense, $\phi = D_4$ has a "dilational derivative." For the hat function, the recursion matrix (see below) again has $\lambda = 1, \frac{1}{2}$. For the cubic spline the eigenvalues are $1, \frac{1}{2}, \frac{1}{4}, \frac{1}{8}$.

To repeat for emphasis: From $\phi(1)$ and $\phi(2)$ the recursion gives everything.

In these constructions the properties of $P(\xi) = \frac{1}{2}\sum c_k e^{ik\xi}$ are decisive. The precise hypotheses are in flux, and infinitely many c_k can be allowed. One basic property will bring together the theory of dilation equations, before we go on to wavelets.

1.1. Dilation equations: Fundamental theorem. The accuracy of piecewise polynomial approximation, by splines or finite elements, depends on the answer to this question: To what degree $p-1$ can the polynomials $1, x, x^2, \ldots, x^{p-1}$ be reproduced exactly by the approximating functions? When the polynomials are "in the space," the approximation error is of order h^p. In our case, the approximating functions are $\phi(x)$ and its translates. Splines are the best at approximation, and finite elements have the narrowest support—but both are weeded out when we require orthogonality. There is already a theory of approximation by translates. It connects p with the properties of $\widehat{\phi}$. The link is the Poisson summation formula. When ϕ solves a dilation equation, that throws new questions into the theory—it is extremely satisfying that these new questions have the same answers.

For approximation with accuracy h^p, the Fourier transform $\widehat{\phi}$ must have zeros of order p at all points $\xi = 2\pi n$ (except at $\xi = 0$ where $\widehat{\phi} = 1$). Notice how easily that converts to a condition on the symbol P. According to (3), the transform $\widehat{\phi}$ is the infinite product of $P\left(\xi/2^j\right)$. At $\xi = 2\pi$ the first factor is $P(\pi)$. At $\xi = 4\pi$ the second factor becomes $P(\pi)$. At $\xi = 6\pi$ the first factor is $P(3\pi)$, which by periodicity is the same as $P(\pi)$. The zeros of P produce zeros of $\widehat{\phi}$:

Condition A. The symbol $P = \frac{1}{2}\sum c_k e^{ik\xi}$ has a zero of order p at $\xi = \pi$. Equivalently, the coefficients c_k satisfy the sum rules that yield $P^{(m)}(\pi) = 0$:

$$\sum (-1)^k k^m c_k = 0, \quad m = 0, 1, \ldots, p-1. \tag{A.5}$$

The box function has $P = \frac{1}{2}\left(1 + e^{i\xi}\right)$ and $p = 1$. The hat function has $p = 2$ and so does D_4. The cubic spline has $p = 4$.

A zero at $\xi = \pi/2$ (instead of π) would also produce the desired zeros in the product $\widehat{\phi}$. Thus Condition A is not strictly necessary in what follows. Choosing $c_0 = 1$ and $c_2 = 1$ and $P = \frac{1}{2}\left(1 + e^{2i\xi}\right)$ stretches out the box function—it becomes $\phi = \frac{1}{2}$ on the double interval $0 < x \le 2$. But $P(\pi/2) = 0$ produces instability and linear dependence—the alternating sum of stretched boxes is $\sum(-1)^k \phi(x-k) = 0$. With the added requirement of stability, the condition is exactly right.

The fundamental theorem states the consequences of Condition A:

1. The polynomials $1, x, \ldots, x^{p-1}$ are linear combinations of the translates $\phi(x-k)$.

2. Smooth functions can be approximated with error $O(h^p)$ by combinations at every scale $h = 2^{-j}$:
$$\left\| f - \sum_k a_k \phi\left(2^j x - k\right)\right\| \le C 2^{-jp} \left\| f^{(p)} \right\| \quad \text{for suitable } a_k.$$

3. The first p moments of the wavelet $W(x)$ (see below) are zero:
$$\int x^m W(x)\, dx = 0 \quad \text{for } m = 0, \ldots, p-1.$$

4. The wavelet coefficients of a smooth function decay like $\left|\int f(x) W\left(2^j x\right)\, dx\right| \le C 2^{-jp}$.

5. The recursion matrix M_N that determines ϕ at the integers has the eigenvalues $1, \frac{1}{2}, \ldots, \left(\frac{1}{2}\right)^{p-1}$.

1 and **2** come from approximation theory. The combination of ϕ's at scale j is also a combination $\sum b_{jk} W\left(2^j x - k\right)$ down to scale j. **3** and **4** are easy once wavelets are defined. Mallat gives a sharp result, with properly stated requirements on the smoothness and decay of ϕ: The H^p norm of f is equivalent to the corresponding norm of its coefficients b_{jk}. Wavelets lead to unconditional bases, suitable for a wide range of function spaces.

It is **5** that makes $\phi(x)$ smoother as p increases and also makes the constructions successful. The smoothness is weaker than $\phi \in C^{p-1}$, but it is striking that "dilational derivatives" come at the same time as higher degrees of approximation. What remains to be studied is orthogonality—which imposes an entirely different condition on the c_k.

Remark 1. Suppose the basic recursion has coefficients $c_0, \ldots, c_N$. Then ϕ is zero outside the interval $[0, N]$. With continuity it follows that $\phi(0) = 0$ and $\phi(N) = 0$. Those were assumed in (4) when we determined $\phi = D_4$ at the integers. For the box function with $N = 1$, $\phi(0)$ and $\phi(N)$ cannot both be dropped. Our recursion matrix will be $(M_N)_{ij} = c_{2i-j}$ with $i, j = 0, \ldots, N-1$. For the box function $M_1 = [1]$ has eigenvalue $\lambda = 1$, as expected in **5** above.

The spectrum of the infinite matrix M (allowing all i, j) is an attractive problem in operator theory. Notice that M is *convolution followed by decimation*—multiplication by the matrix c_{i-j} followed by projection onto even-numbered coordinates. By contrast with the usual Toeplitz case, eigenfunctions can have compact support! Homogeneous difference equations with zero boundary conditions lead to $\phi = 0$, but not so for dilation equations.

Remark 2. The minimum requirement is $p = 1$. Then $P(\pi) = 0$, which means that $\sum c_{2k} = \sum c_{2k+1}$. Since $\sum c_k = 2$, the columns of M add to 1:

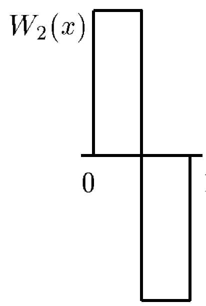
$$M_N = \begin{bmatrix} c_0 & & & \\ c_2 & c_1 & c_0 & \\ & c_3 & c_2 & c_1 \\ & & & c_3 \end{bmatrix} \qquad \begin{array}{l} \text{steps of 2 down columns} \\ \text{steps of 1 across rows} \\ \text{here } N = 4 \end{array}$$

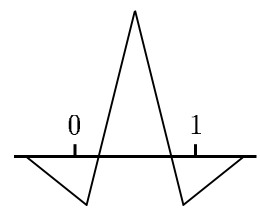
$(1,1,1,1)$ is a left eigenvector with $\lambda = 1$. The right eigenvector yields the values $\phi(0), \ldots, \phi(N-1)$ at the integers. The recursion determines ϕ at all dyadic points. Values at other points are never used.

1.2. Wavelets and orthogonality. Finally we define a wavelet. It comes from the scaling function ϕ by taking "differences":

$$W(x) = \sum (-1)^k c_{1-k} \phi(2x - k). \tag{A.6}$$

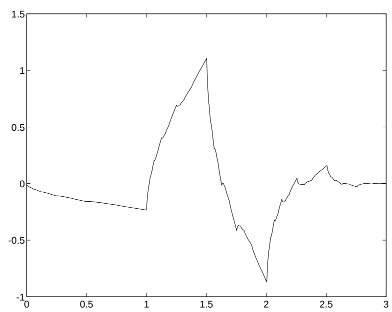
We write W in place of the usual ψ, to distinguish more clearly from ϕ. Notice $2x$ on the right, and especially $(-1)^k$. Examples show the effect of alternating signs:





Haar wavelet from box function $W_2(x) = \phi(2x) - \phi(2x-1)$

"Wavelet" from hat function $W = \phi(2x) - \frac{1}{2}\phi(2x-1) - \frac{1}{2}\phi(2x+1)$



$W_4(x)$ from $\phi = D_4$ Orthogonal wavelet

The wavelet from the hat function does not belong here. It is not orthogonal to $W(x+1)$. The point is that the other two do belong. The Haar function is orthogonal to its own translations and dilations. Historically it was the original wavelet (but with $p = 1$ and poor approximation). The orthogonal wavelet W_4 has $p = 2$ and second-order approximation.

Without formulas for D_4 and W_4, how is the orthogonality of their translates known? We need a test that applies to the recursion coefficients c_k, or to the symbol $P(\xi) = \frac{1}{2}\sum c_k e^{ik\xi}$.

Condition O.

$$|P(\xi)|^2 + |P(\xi+\pi)|^2 \equiv 1 \quad \text{or} \quad \sum c_k c_{k-2m} = 2\delta_{0m}.$$

With this condition, the infinite matrix L^*L in Part 2 is an orthogonal projection. To see now the role of Condition O, suppose the functions $\phi_0(2x-k)$ are orthogonal. Then so are the translates of $\phi_1(x) = \sum c_k \phi_0(2x-k)$:

$$\begin{aligned} \int \phi_1(x)\phi_1(x-m)\,dx &= \int \left(\sum c_k \phi_0(2x-k)\right)\left(\sum c_l \phi_0(2x-2m-l)\right)dx \\ &= \sum c_k c_{k-2m} \int \phi_0^2(2x)\,dx = 0 \quad \text{for } m \neq 0. \end{aligned} \tag{A.7}$$

Construction 1 creates ϕ by iteration from the box function, which is orthogonal to its translates. Therefore (as Daubechies observed) so is ϕ.

The wavelet $W(x)$ in (6) is also orthogonal to $\phi(x-m)$. This is simple but neat, not involving Condition O. The sum in (7) changes to

$$\sum (-1)^k c_{1-k} c_{k-2m} \quad \text{which is identically zero!} \tag{A.8}$$

Just replace k by $1-n+2m$. This identity is $HL^* = 0$ in Part 2. Then (6) makes $W(x)$ orthogonal to $W(2x-m)$. The orthogonality of $W(x)$ and $W(x-m)$ comes back to Condition O.

The goal in constructing wavelets is to satisfy Conditions A and O. The basic family W_2, W_4, W_6, ... was discovered by Daubechies, following Haar's W_2. The accuracies are $p = 1, 2, 3, \ldots$ and there are $2, 4, 6, \ldots$ nonzero coefficients c_k. The smoothness also increases with p—but only by about $\frac{1}{2}$ derivative each time. D_4 and W_4 are Hölder continuous with exponent $.550\cdots$. In Galerkin's method for solving differential equations, it is natural for these wavelets to be the trial functions—broader support than splines, nonsymmetric but orthogonal, multigrid built in, all computations based on recursion, difficulty to be expected at boundaries. The first experiments by Glowinski, Lawton, and Ravachol are particularly interesting for Burgers' equation.

2. Algorithms for wavelet expansions. Now comes a change of direction. Instead of discussing the properties of wavelets, we describe algorithms. The main question is how to *decompose* a signal into its wavelet coefficients, and how to *reconstruct* the signal from the coefficients. There is a "tree algorithm" or "pyramid algorithm" that makes these steps simple and fast. It does for the discrete wavelet transform what the Fast Fourier Transform (FFT) does for the discrete Fourier transform. The algorithm is fully recursive.

The user chooses a specific wavelet. We begin with the simplest choice, based on the box function. It satisfies the orthogonality property (Condition O), so all pieces of the decomposition are orthogonal. The approximation property (Condition A which preserves polynomials) determines how quickly the coefficients decay—for efficiency we want to stop the decomposition early. In that respect the box function is poor. Efficiency is the reason for working with higher wavelets W_4, W_6, $W_8, \ldots$, and simplicity is the reason for starting with W_2. This is Haar's wavelet $[1\ -1]$.

The discussion will be discrete—for vectors not functions. We are given $n = 2^j$ values $f_1, \ldots, f_n$. They may be equally spaced values of a function $f(x)$ on a unit interval. The goal is to split this vector f into its components at different scales, indexed by j. At each new level the meshwidth h is cut in half and the number of wavelet coefficients is doubled. The decomposition is

$$f = f^{\phi} + f^{(0)} + \cdots + f^{(J-1)}.$$

The "detail" $f^{(j)}$ is a combination of 2^j wavelets at scale 2^{-j}, and f^{ϕ} is a multiple of the scaling function ϕ. For a numerical example take $J = 2$. Then the finest detail $f^{(1)}$ is the sum of two terms, here with coefficients $b_{11} = 4$ and $b_{12} = 1$:

$$f = \begin{bmatrix} 9 \\ 1 \\ 2 \\ 0 \end{bmatrix} = 3\begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} + 2\begin{bmatrix} 1 \\ 1 \\ -1 \\ -1 \end{bmatrix} + 4\begin{bmatrix} 1 \\ -1 \\ 0 \\ 0 \end{bmatrix} + 1\begin{bmatrix} 0 \\ 0 \\ 1 \\ -1 \end{bmatrix}. \tag{A.9}$$

Notice that the four components are mutually orthogonal. There are $1 + 2 + \cdots + 2^{J-1}$ wavelet coefficients, and the one from f^{ϕ} makes 2^J.

How are the coefficients 3, 2, 4, 1 computed from f? *On the finest scale first.* As in the FFT, the decomposition begins with a "butterfly":

$$\begin{bmatrix} \frac{1}{2} & \frac{1}{2} & & \\ \frac{1}{2} & -\frac{1}{2} & & \\ & & \frac{1}{2} & \frac{1}{2} \\ & & \frac{1}{2} & -\frac{1}{2} \end{bmatrix}\begin{bmatrix} 9 \\ 1 \\ 2 \\ 0 \end{bmatrix} = \begin{bmatrix} 5 \\ 4 \\ 1 \\ 1 \end{bmatrix}. \tag{A.10}$$

This is followed by a permutation, in which high frequencies go to the bottom:

$$\begin{bmatrix} 1 & & & \\ & & 1 & \\ & 1 & & \\ & & & 1 \end{bmatrix} \begin{bmatrix} 5 \\ 1 \\ 1 \\ 4 \end{bmatrix} = \begin{bmatrix} 5 \\ 1 \\ 4 \\ 1 \end{bmatrix}. \qquad \text{(A.11)}$$

The next step is another butterfly, on low frequencies only:

$$\begin{bmatrix} \frac{1}{2} & \frac{1}{2} & & \\ \frac{1}{2} & -\frac{1}{2} & & \\ & & 1 & \\ & & & 1 \end{bmatrix} \begin{bmatrix} 5 \\ 1 \\ 4 \\ 1 \end{bmatrix} = \begin{bmatrix} 3 \\ 2 \\ 4 \\ 1 \end{bmatrix}. \qquad \text{(A.12)}$$

The result is the set of wavelet coefficients 3, 2, 4, 1. The product of the three matrices in (10–12) is the decomposition matrix D. Its inverse is the reconstruction matrix R:

$$D = \begin{bmatrix} \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} & -\frac{1}{4} & -\frac{1}{4} \\ \frac{1}{2} & -\frac{1}{2} & 0 & 0 \\ 0 & 0 & \frac{1}{2} & -\frac{1}{2} \end{bmatrix} \quad \text{has} \quad D^{-1} = R = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & -1 & 0 \\ 1 & -1 & 0 & 1 \\ 1 & -1 & 0 & -1 \end{bmatrix}.$$

The coefficients 3, 2, 4, 1 enter the vector $b = (b_\phi, b_{01}, b_{11}, b_{12})$. The wavelet expansion in (9) is $f = R\, b$. The coefficients are $b = R^{-1}f = Df$. This product Df was computed recursively, from two butterfly matrices with a permutation between. In general there will be J matrices with permutations between.

The reconstruction is also recursive. It inverts (12) then (11) then (10). The global matrix R is the product of these local inverse matrices.

Notice that the operation count is proportional to n. It is best possible (the FFT count is $n \log_2 n$). There are only $n - 1$ individual 2-by-2 matrix multiplications, since high frequency coefficients (here 4 and 1) are settled and not reused. The Walsh functions give a different piecewise constant representation, in which the last two basis vectors are $(1, -1, 1, -1)$ and $(1, -1, -1, 1)$. In that case 4 and 1 enter another butterfly to produce the Walsh coefficients $\frac{5}{2}$ and $\frac{3}{2}$. The Walsh basis is global. The wavelet basis is local, but scaled—its support has width $O(2^{-J})$ at the finest scale and $O(1)$ at the coarsest scale.

Notice also the normalizing factors $\frac{1}{2}$ in decomposition (and 1's in reconstruction). The alternative is to introduce $1/\sqrt{2}$ for both. This has the advantage of normalizing the wavelets $W_{jk} = 2^{j/2}W(2^j x - k)$ at every scale. The whole basis is orthonormal (when $\|W\| = 1$). In the discrete case R and D become orthogonal matrices:

$$\widehat{D} = \begin{bmatrix} 1/2 & 1/2 & 1/2 & 1/2 \\ 1/2 & 1/2 & -1/2 & -1/2 \\ 1/\sqrt{2} & -1/\sqrt{2} & 0 & 0 \\ 0 & 0 & 1/\sqrt{2} & -1/\sqrt{2} \end{bmatrix} \quad \text{has } \widehat{R} = \widehat{D}^{-1} = \text{transpose of } \widehat{D}.$$

Based on the Haar example, we now start on Mallat's beautiful *tree algorithm* for wavelets. The simple average from $\left[\frac{1}{2}\ \frac{1}{2}\right]$ is replaced by a discrete filter based on ϕ. The difference $\left[\frac{1}{2}\ -\frac{1}{2}\right]$ is replaced by a filter based on W. The filters use the same recursion coefficients c_k that led to ϕ and W in the first place.

Decomposition. The given n-vector f is on the finest scale $h = 2^{-J}$. The *fine-to-coarse filter* (the "restriction operator" in multigrid language, the lowpass filter in signal processing language) is L. It produces a vector with half as many entries:

$$(Lf)_i = \frac{1}{2} \sum c_{2i-j} f_j, \quad i = 1, \ldots, \frac{n}{2}. \tag{A.13}$$

In the Haar example with $c_0 = c_1 = 1$, the entries of Lf are $\frac{1}{2}(f_1 + f_2)$ and $\frac{1}{2}(f_3 + f_4)$. The recursion continues to coarser scales, and after J steps it reaches a single number—the coefficient b_ϕ in f^ϕ at the coarsest scale $h = 1$. Here $b_\phi = \frac{1}{4}(f_1 + f_2 + f_3 + f_4)$.

The dual to L is the *coarse-to-fine map* L^* (the "interpolation operator" in multigrid language). Notice the change of index and the disappearance of $\frac{1}{2}$:

$$(L^* g)_j = \sum c_{2i-j} g_i, \quad j = 1, \ldots, n. \tag{A.14}$$

In the Haar example L^*Lf has entries $\frac{1}{2}(f_1+f_2), \frac{1}{2}(f_1-f_2), \frac{1}{2}(f_3+f_4), \frac{1}{2}(f_3-f_4)$. It is the projection of f onto the subspace that is piecewise constant at scale $2h$. It gives a blurred picture, with details lost.

The decomposition picks out these details, orthogonal to the average. The projection onto the wavelet subspace is the high frequency component:

$$f^{(J-1)} = f - L^* L f. \tag{A.15}$$

This repeats at every stage. There is an "average" or "blurred picture" $a^{(j-1)} = La^{(j)}$, starting from $a^{(J)} = f$. The detail lost in that average is the component of f at that stage:

$$f^{(j-1)} = (I - L^* L) a^{(j)} = a^{(j)} - L^* a^{(j-1)}. \tag{A.16}$$

This is a first statement of the decomposition algorithm. We will see how Condition O simplifies the formula.

Reconstruction. To produce f from its details $f^{(j)}$, run the recursion (16) in reverse:

$$a^{(j)} = f^{(j-1)} + L^* a^{(j-1)}. \tag{A.17}$$

This starts from the coarsest detail $f^{(0)}$ and the totally blurred picture $a^{(0)} = f^\phi$. It returns to $f = a^{(J)}$.

Apply orthogonality. The most elegant part of the algorithm is still to come. It is not necessary to compute the detail vector $f^{(j)}$ from (16), and then to compute its wavelet coefficients b_{jk}. Those are the numbers we want (4 and 1 in the example at level $j = 1$). *These numbers can be found directly from* $a^{(j)}$.

Review the Haar example first. The lowpass filter gave $a^{(1)}$ from $f = a^{(2)}$:

$$Lf = \frac{1}{2} \begin{bmatrix} c_1 & c_0 & & \\ & & c_1 & c_0 \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & & \\ & & \frac{1}{2} & \frac{1}{2} \end{bmatrix} \begin{bmatrix} 9 \\ 1 \\ 2 \\ 0 \end{bmatrix} = \begin{bmatrix} 5 \\ 1 \end{bmatrix}.$$

The blurred picture is $a^{(1)} = (5, 5, 1, 1)$. At the next level the low-pass filter leaves 3, the coefficient of $(1, 1, 1, 1)$. *We now want the orthogonal filter—the highpass filter H.* In the Haar example it produces

$$Hf = \frac{1}{2}\begin{bmatrix} c_0 & -c_1 & & \\ & & c_0 & -c_1 \end{bmatrix}\begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & -\frac{1}{2} & & \\ & & \frac{1}{2} & -\frac{1}{2} \end{bmatrix}\begin{bmatrix} 9 \\ 1 \\ 2 \\ 0 \end{bmatrix} = \begin{bmatrix} 4 \\ 1 \end{bmatrix}.$$

Those coefficients 4 and 1 represent the detail $f^{(1)} = (4, -4, 1, -1)$, which is lost when $a^{(2)}$ is blurred to $a^{(1)}$. At the next level H is applied to $a^{(1)}$. That produces $\frac{1}{2}(5) - \frac{1}{2}(1) = 2$. This is the coefficient b_{01}, representing the detail $(2, 2, -2, -2)$ lost when $a^{(1)}$ is blurred to $a^{(0)}$. We now put these pieces together into *Mallat's pyramid algorithm*:

Decomposition. Initialize $a^J = f$. For $j = J, \ldots, 1$ compute

$$a^{j-1} = La^j \quad \text{and} \quad b^{j-1} = Ha^j. \tag{A.18}$$

Reconstruction. Start with a^0 and $b^0, \ldots, b^{J-1}$. For $j = 1, \ldots, J$ compute

$$a^j = L^* a^{j-1} + H^* b^{j-1}. \tag{A.19}$$

The full decomposition is represented by a tree of filters:

$$\begin{array}{ccccccccc} a^J & \xrightarrow{L} & a^{J-1} & \xrightarrow{L} & a^{J-2} & \cdots & \xrightarrow{L} & a^0. \\ & {}_H\searrow & & {}_H\searrow & & & {}_H\searrow & \\ & & b^{J-1} & & b^{J-2} & & & b^0. \end{array}$$

The reconstruction goes from the branches of the tree back to the root:

$$\begin{array}{cccccccc} a^0 & \xrightarrow{L^*} & a^1 & \xrightarrow{L^*} & a^2 & \cdots & \xrightarrow{L^*} & a^J = f. \\ & \nearrow_{H^*} & & \nearrow_{H^*} & & & \nearrow_{H^*} & \\ b^0 & & b^1 & & & & & \end{array}$$

The next step is to identify these filter matrices L and H for examples other than "box and Haar."

Note. The filter matrices L and H have half as many rows as columns. By dropping the parentheses around j, we distinguish the vector a^j with only 2^j components from the vector $a^{(j)}$ with the full $w^J = n$ components. The vector a^j contains the expansion coefficients of $a^{(j)}$ with respect to the translates $\phi(2^j x - k)$. See the example above and the multiresolution below!

2.1. The filter matrices L and H. The matrix L is known from the first part of the paper. Its entries $L_{ij} = c_{2i-j}$ are the recursion coefficients for the scaling function. Rows 1, 2 and columns -1, 0, 1, 2 are displayed with $N = 3$:

$$L = \frac{1}{2}\begin{bmatrix} c_3 & c_2 & c_1 & c_0 & & \\ & & c_3 & c_2 & c_1 & c_0 \end{bmatrix}.$$

The beautiful thing is that the highpass filter (strictly speaking it is band-pass) uses the same coefficients. H is associated with the wavelet W just as L is associated with the scaling function ϕ. Equation (6) for W uses the same c_k, but with alternating signs and reversed order. The wavelet filter has

$$H_{ij} = (-1)^{j+1} c_{j+1-2i}. \tag{A.20}$$

Rows 1, 2 and columns 1, 2, 3, 4 are displayed:

$$H = \frac{1}{2}\begin{bmatrix} c_0 & -c_1 & c_2 & -c_3 & & \\ & & c_0 & -c_1 & c_2 & -c_3 \end{bmatrix}.$$

The indices were chosen to match the Haar example (variants are possible). The transposed matrices, without the factor $\frac{1}{2}$, represent the dual filters L^* and H^*. The important points now come quickly, and matrix multiplication is the best proof.

Theorem 1. *By their construction the filters are orthogonal:*

$$HL^* = 0. \tag{A.21}$$

This multiplication is the reason behind the construction of H—alternating signs, reversed order, index shifted by one. See equation (8).

We finally come to the reward for Condition O: $\sum c_k c_{k+2m} = 2\delta_{0m}$. The reason for that condition is in the reward. Remember that the box function and D_4 satisfied this requirement but not the hat function or the cubic spline. Condition O can be stated and understood in transform space, but I believe that the matrix interpretation is again the clearest.

Theorem 2. *If condition* O *holds then*

1. $LL^* = I$ *and* $HH^* = I$. (A.22)
2. L^*L *and* H^*H *are mutually orthogonal projections with*

$$L^*L + H^*H = I. \tag{A.23}$$

Remember that L and H map into subspaces half as large as the original. L^* and H^* map back. The identity operators in (22) are on the half-sized subspaces.

The proof of (22) is by direct matrix manipulation. Condition O gives the result. Then it follows that $L^*LL^*L = L^*L$, so L^*L is a projection—and similarly for H^*H. The property $HL^* = 0$ in (21) yields $H(L^*L + H^*H) = H$. The transpose $LH^* = 0$ yields $L(L^*L + H^*H) = L$. The operator in (23) is the identity on both orthogonal components—the ranges of L and H—so it is the identity. We have an orthogonal decomposition by "*quadrature mirror filters*" L and H at every step.

2.2. Multiresolution of L^2. The last paragraphs changed quietly from functions to vectors. That was for the sake of algorithms, which use values of ϕ and W at dyadic points $k/2^J$. The Haar example began with f at equally spaced points on $(0, 1]$. But the filter matrices really apply to discrete values along the whole line—they are infinite matrices. More than that, the decomposition $f = \sum f^{(j)}$ is just as valuable for functions in L^2 as for vectors in l^2.

This multiresolution yields the details of f at all scalings 2^{-j}. On the whole line we take $j = 0, \pm 1, \pm 2, \ldots$. The decomposition develops an idea that was already present in approximation theory—to put frequencies together in "octaves." (Besov spaces combine frequencies $2^j \leq \xi < 2^{j+1}$. It seems that the ear also receives frequencies on a logarithmic scale.) For functional analysis the starting point is the subspace S_j spanned by the translates $\phi(2^j x - k)$. If a function $g(x)$ is in S_j, then $g(2x)$ is in S_{j+1}. The dilation equation writes $\phi(x)$ as a combination of $\phi(2x - k)$, which assures that $S_0 \subset S_1$. At all scales we have

$$\cdots S_{-1} \subset S_0 \subset S_1 \subset S_2 \cdots \text{ with } \cup S_j \text{ dense in } L^2 \text{ and } \cap S_j = \{0\}.$$

Now turn to the *wavelet subspace* W_j. It is spanned by the translates $W(2^j x - k)$. It is invariant under translation by multiples of 2^{-j}. If $g(x)$ is in W_j then $g(2x)$ is in W_{j+1}. The construction

$W(x) = \sum(-1)^k x_{1-k}\phi(2x-k)$ puts W and its translates into S_1, and makes them orthogonal to S_0. *In fact, W_0 and S_0 are orthogonal complements in S_1.* At every scale $W_j \bigoplus S_j = S_{j+1}$. *The spaces S_j give the "partial sums" of the differences W_j:*

$$\cdots \bigoplus W_{-1} \bigoplus W_0 \bigoplus \cdots \bigoplus W_j = S_{j+1} \quad \text{and} \quad \bigoplus_{-\infty}^{\infty} W_j = L^2.$$

The multiresolution of f is a splitting into components $f^{(j)} \in W_j$:

$$f = \sum_{-\infty}^{\infty} f^{(j)} \quad \text{or} \quad f = f^{\phi} + \sum_{0}^{\infty} f^{(j)}, \quad f^{\phi} \in S_0. \tag{A.24}$$

This is a very satisfying decomposition of L^2 functions, classical but with new subspaces. The coefficients b^j in Mallat's pyramid algorithm corresponded to $f^{(j)} \in W_j$, and a^j corresponded to $a^{(j)} \in S_j$.

The analogue of the discrete Fourier transform was in the algorithm. The analogue of ordinary Fourier series is (24). The analogue of the Fourier integral formula is the *integral wavelet transform*. Representations of different groups give rise to different transforms.

2.3. Applications. Image processing works with $F(x, y)$, so it is natural to look for *two-dimensional wavelets*. The simplest construction uses the products $\phi(x)\phi(y)$, $\phi(x)W(y)$, $W(x)\phi(y)$, $W(x)W(y)$. Orthogonality is clear. New constructions have been invented that are genuinely two-dimensional, but it is useful to start with the tensor products of "box and Haar." The given two-dimensional array F yields a two-dimensional array B of wavelet coefficients.

For pattern recognition, a major difficulty with the wavelet transform B is the lack of translation invariance. If the pattern is shifted by a fraction of h, its wavelet model is changed. A higher sampling rate is possible but expensive. Mallat studies instead the *zero-crossings* of the wavelet transform, which locate the signal edges. Now the difficulty is to make the reconstruction stable. In edge detection the first wavelets were Laplacians of shifted Gaussians, introduced by Gabor. The orthogonal wavelets of Meyer are C^{∞} with polynomial decay, the Battle-Lemarié wavelets based on splines are C^n with exponential decay, and the Daubechies wavelets are C^n (smaller n) with compact support.

In closing we recall the original problem—to localize in time and frequency. Geophysics needs to represent short high-frequency pulses. Physics needs to divide up phase space. The coherent states $g_{pq} = e^{ipx}g(x-q)$ give a "Weyl-Heisenberg" frame, with some redundancy—but still f can be reconstructed from $\iint (f, g_{pq})g_{pq}dp\,dq$. Mathematics needs (or wants) an orthogonal decomposition, better than g_{pq} at high frequencies and with no redundancy. The answer for now is wavelets.

It is a pleasure to thank Ingrid Daubechies and Howard Resnikoff for introducing me to wavelets.