EXPERIMENTS WITH SEMI-LAGRANGIAN WAVELETS

Antony Ware
Department of Mathematical Sciences
University of Durham

Summary: We review some basic material concerning wavelet analysis, highlighting aspects which are of importance in numerical analysis applications, and in particular the solution of partial differential equations. The proposed use of the semi-Lagrangian methodology introduces some further considerations which have to be taken into account when selecting which wavelets to use. The biorthogonal spline wavelets of Cohen, Daubechies and Feauveau have particularly favourable properties, and we make use of them as an adaptive spatial discretisation, in conjunction with a semi-Lagrangian timestepping procedure, for the solution of the Burgers equation.

1 INTRODUCTION

The basic idea of wavelet analysis is the representation of functions in terms of simple building blocks at different positions and scales. The ability of these building blocks to characterise *locally* the behaviour of a function f, and so to open up the possibility of efficient and reliable compression of the essential information contained in f, is one of the reasons that wavelet techniques have been seen to hold such promise in a wide variety of applications. Initially developed in the area of signal processing, wavelets have also been used in image compression and computer graphics, studies of turbulence and astronomical data, and the solution of integral and partial diffential equations. (see for example *Meyer*, 1993, *Bacry*, 1992 and *Qian*, 1993).

Wavelet-based techniques for solving partial differential equations are most often derived within a (Petrov-) Galerkin framework, sometimes mixed with a collocation treatment of nonlinear terms. Efficient algorithms exist for calculating derivatives of functions directly from their wavelet expansion coefficients, and wavelet decomposition can greatly simplify the inversion of simple linear differential operators, providing (with a diagonal preconditioner) a condition number independent of the size of the 'mesh' (*Beylkin*, 1993). (There is a relationship here between wavelets and multigrid, and this has been fruitfully explored (*Reider*, 1994).)

The objective of this paper is to investigate the feasibilty of using wavelets to provide a spatial discretisation of a function for use in conjunction with the semi-Lagrangian method for solving time-dependent partial differential equations. In the second section basic aspects of wavelet theory are reviewed within the framework of multiresolution analyses formulated by Mallat and Meyer (*Mallat*, 1989), leading up to an account of the biorthogonal spline wavelets of Cohen, Daubechies and Feauveau (*Cohen*, 1992). In Section 3 the implementation of a semi-Lagrangian wavelet method for solving the inviscid Burgers equation is described, and some related issues are discussed. The paper continues with an account of some experiments performed using the algorithms introduced, including simulations of a fully adaptive method which indicate that significant compression is possible, and ends with a brief indication of lines of future investigation.

2 MULTIRESOLUTION ANALYSES AND WAVELETS

The concept of grid refinement is fundamental to the numerical solution of differential equations. A sequence of grids is defined, and each grid can be associated with a finite-dimensional space of functions V_j. If the grids are nested, then so are the subspaces. The intention is that as we move further down the sequence of grids we capture more and more information about the solution, so that in the limit as $j \to \infty$ we would know the solution completely. Often we can have two equivalent representations of a function $f \in V_j$. One in terms of the values of f on the grid, and the other in terms of the coefficients of an expansion of f as a sum of 'trial' functions that span V_j. The definition of a multiresolution analysis formalises this situation the particluar case when the grids are *dyadic*, and when the trial functions are all translates of a single 'scaling function.'

Definition (*Mallat*, 1989). A multiresolution analysis of $L_2(\mathbf{R})$ is a nested sequence of closed subspaces $V_j \subset L_2(\mathbf{R})$ satisfying

1. $f(\cdot) \in V_j \Leftrightarrow f(2\cdot) \in V_{j+1} \qquad \forall j \in \mathbf{Z}$.
2. $\bigcup_{j\in\mathbf{Z}}$ is dense in $L_2(\mathbf{R})$, and $\bigcap_{j\in\mathbf{Z}} = \{0\}$.
3. $\exists \phi \in L_2(\mathbf{R})$ (a scaling function) such that $\{\phi(\cdot - l)\}_{l\in\mathbf{Z}}$ is a Riesz basis of V_0.

(A Riesz basis is such that the l_2-norm of the coefficients of a function is an equivalent norm to the L_2-norm.)

A simple but important consequence of the fact the $V_0 \subset V_1$ is that ϕ satisfies a *refinement equation*; there is a sequence (a filter) $\{h_k\} \in l_2$ such that

$$\phi(x) = \sum_k h_k \phi(2x - k). \tag{1}$$

Subject to some general conditions, ϕ is in fact completely specified by this equation, up to a constant multiple, which can be determined by specifying that

$$\int_{-\infty}^{\infty} \phi(x)\,\mathrm{d}\,x = 1.$$

Often it is by finding suitable sequences $\{h_k\}$ that scaling functions are contructed. Once such a sequence is known, values of ϕ at the integers may be found by solving an eigenvalue problem, and from there the refinement equation will give the value of ϕ at any dyadic point $x = 2^{-j}k$.

An orthonormal scaling function ϕ is one that satisfies

$$\big(\phi(\cdot), \phi(\cdot - l)\big) = \delta_{0l} \qquad \forall l \in \mathbf{Z}. \tag{2}$$

Such a function always exists, as can be seen by noting that the orthonormality condition (2) is equivalent to

$$b(\omega) := \sum_{k\in\mathbf{Z}} |\hat{\phi}(\omega + 2k\pi)|^2 = 1;$$

thus if ϕ is not orthonormal, a scaling function that is orthonormal can be found via

$$\hat{\phi}_{\mathrm{orth}}(\omega) = \frac{\hat{\phi}(\omega)}{\sqrt{b(\omega)}}. \tag{3}$$

If we define

$$\phi_{j,k}(x) = 2^{j/2}\phi(2^j x - k), \tag{4}$$

then for any $j \in \mathbf{Z}$ the set $\{\phi_{j,k}\}_{k\in\mathbf{Z}}$ forms a (orthonormal) basis for V_j.

The canonical example of an orthogonal scaling function is the Haar basis function $\phi = \chi_{[0,1]}$ (Figure 1). This leads to spaces V_j consisting of functions that are piecewise constant on the intervals $[2^{-j}k, 2^{-j}(k+1)]$. Since their supports do not intersect, the functions $\{\phi_{j,k}\}_{k\in\mathbf{Z}}$ are plainly orthogonal, and the scaling factor in (4) ensures that they are in fact orthonormal with respect to the inner product on $L_2(\mathbf{R})$.

Cardinal B-splines also generate multiresolution analyses. Recall that these are defined as convolutions of Haar basis functions. Thus with $\beta^1 = \chi_{[0,1]}$, we define recursively $\beta^m = \beta^{m-1} * \beta^1$. All of these functions have compact support, vanishing outside the interval $[0, m]$, but in general they are not orthogonal. For example, β^2 is then the 'hat function',

$$\beta^2(x) = \begin{cases} x & \text{if } x \in (0,1] \\ 2 - x & \text{if } x \in (1,2] \\ 0 & \text{otherwise.} \end{cases}$$

Orthogonalising these according to the prescription in (3) leads to the Battle-Lemarie scaling functions, which have non-compact support, although they do decay rapidly as $x \to \infty$.

Given a function $f \in L_2(\mathbf{R})$, we can define its projection onto V_j in terms of its inner products with the functions $\phi_{j,k}$ if ϕ is orthogonal. These projections provide us with approximations to f that incorporate progressively finer levels of detail.

Let W_j be a space complementing V_j in V_{j+1}. Then W_j encapsulates the extra detail that is added when we progress from V_j to V_{j+1}. For example, in the case of the Haar basis, we can progress from a function constant on intervals of length 1 to a function constant on intervals of length $\frac{1}{2}$ by adding multiples of the integer translates of the *Haar wavelet* (see Figure 1)

$$\psi = \chi_{[0,\frac{1}{2}]} - \chi_{[\frac{1}{2},1]}.$$

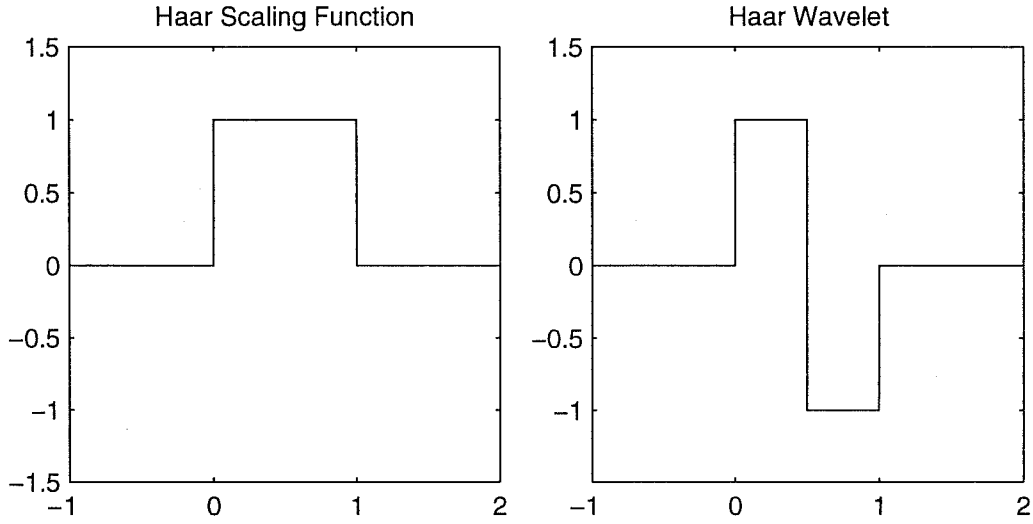

Fig. 1 The Haar scaling function and wavelet.

In general, a function ψ is a wavelet if the collection $\{\psi(\cdot - l)\}_{l\in\mathbf{Z}}$ forms a Riesz basis of W_0. With a definition of $\psi_{j,k}$ similar to that of $\phi_{j,k}$ then $\{\psi_{j,k}\}_{k\in\mathbf{Z}}$ then forms a basis of W_j, and the complete set $\{\psi_{j,k}\}_{j,k\in\mathbf{Z}}$ is a basis of $L_2(\mathbf{R})$.

Since $W_0 \subset V_1$, the wavelet ψ will also satisfy a refinement equation:

$$\psi(x) = \sum_k g_k \phi(2x - k). \tag{5}$$

There will in general be many alternative choices of the spaces W_j for any particular multiresolution analysis. However, if ϕ is orthogonal, it is natural to take W_j as the orthogonal complement of V_j in V_{j+1}, and so to try to find a wavelet ψ such that

$$\big(\phi(\cdot - l), \psi\big) = 0, \qquad \forall l \in \mathbf{Z}.$$

It can be shown that if we take $g_k = (-1)^k \overline{h_{1-k}}$ then the refinement equation (5) serves to define just such a wavelet. Again the simplest example is the Haar wavelet. Here, the filters h_k and g_k are $(1,1)$ and $(1,-1)$ respectively. Note that as j increases, the functions $\psi_{j,k}$ become progressively more localised, and at the same time correspond in some sense to higher and higher frequencies.

2.1 The fast wavelet transform

A function $v_j \in V_j$ can be expressed as the sum of two functions: a coarse-scale version of v_j, v_{j-1} (in V_{j-1}), and the difference, w_{j-1}, in W_{j-1}. Thus

$$\begin{aligned} v_j(x) &= \sum_k \alpha_{j,k}\phi_{j,k}(x) = v_{j-1}(x) + w_{j-1}(x) \\ &= \sum_k \alpha_{j-1,k}\phi_{j-1,k}(x) + \beta_{j-1,k}\psi_{j-1,k}(x), \end{aligned}$$

where in the orthogonal case,

$$\alpha_{j-1,k} = \big(v_j, \phi_{j-1,k}\big) \qquad \text{and} \qquad \beta_{j-1,k} = \big(v_j, \psi_{j-1,k}\big).$$

We can therefore represent v_j either in terms of its coefficients $\{\alpha_{j,k}\}$, or in terms of the two sequences of coefficients $\{\alpha_{j-1,k}\}$ and $\{\beta_{j-1,k}\}$, each of half of the length of the original. This process can be continued down to some coarsest scale J_0, giving

$$v_j = w_{j-1} + w_{j-2} + \cdots + w_{J_0} + v_{J_0}.$$

How do we construct such a decomposition? Again in the orthogonal case, by (1), we have

$$\alpha_{j-1,k} = (v_j, \phi_{j-1,k}) = 2^{-1/2} \sum_l h_l (v_j, \phi_{j,l+2k}) = 2^{-1/2} \sum_l h_{l-2k} \alpha_{j,l}, \tag{6}$$

and, similarly, using (5) we obtain

$$\beta_{j-1,k} = 2^{-1/2} \sum_l g_{l-2k} \alpha_{j,l} = 2^{-1/2} \sum_l (-1)^l h_{1-l+2k} \alpha_{j,l}. \tag{7}$$

The recomposition of v_j from $v_{J_0} + w_{J_0} + \cdots + w_{j-1}$ is performed in a similar step-by-step manner. Making use of (1) and (5), and the orthogonality of ϕ, we have

$$(\phi(x-l), \phi(2x-k)) = h_{k-2l} \qquad \text{and} \qquad (\psi(x-l), \phi(2x-k)) = g_{k-2l}, \tag{8}$$

so that, generalising this to the $j-1$th and jth scales, the expansion of v_j may be reconstructed from those of v_{j-1} and w_{j-1} via

$$\alpha_{j,k} = (v_j, \phi_{j,k}) = 2^{-1/2} \sum_l \alpha_{j-1,l} h_{k-2l} + \beta_{j-1,l} g_{k-2l}. \tag{9}$$

Thus a function given in terms of scaling function coefficients at one level can be decomposed into its projections onto a sequence of detail spaces. The resulting coefficients capture information about the original function that is localised both in frequency and position, and which may be used to compress the amount of storage needed to capture the essential information about the function, for example by discarding those coefficients that fall below some tolerance level. The amount of work necesary to calculate them is proportional to the number of coefficients; the same is true of the reconstruction process, and hence the name 'fast wavelet transform.' The question remains of how, given a function f, the original coefficients $\alpha_{j,k} = (f, \phi_{j,k})$ are to be calculated. We shall return to this question below.

2.2 The biorthogonal case

The orthogonality property puts a strong limitation on the construction of wavelets. Orthogonal wavelets with compact support and arbitrarily high regularity were first constructed only in the last decade (*Daubechies*, 1988, where it is noted that the Haar wavelet is the only real-valued wavelet that is compactly supported, (anti-)symmetric and orthogonal)). The generalisation to the biorthogonal case relaxes this restiction. In this case, we actually have two multiresolution analyses, the second being generated by a dual scaling function $\tilde{\phi}$. There is a dual wavelet $\tilde{\psi}$, and, while ϕ is no longer orthogonal to its translates or to ψ, we have that for all $l \in \mathbf{Z}$,

$$(\phi, \tilde{\psi}(\cdot - l)) = 0, \qquad (\psi, \tilde{\phi}(\cdot - l)) = 0 \tag{10}$$

and

$$(\phi, \tilde{\phi}(\cdot - l)) = \delta_l \qquad (\psi, \tilde{\psi}(\cdot - l)) = \delta_l. \tag{11}$$

We now have two different pairs of sequences, h_k and g_k, and $\tilde{h}_k$ and $\tilde{g}_k$ for the refinement equations, and we use both of them in the fast wavelet transform: the dual sequences in the forward (decomposition) direction, and the primary sequences for the recontruction.

2.3 Biorthogonal spline wavelets

We have already defined the cardinal spline scaling functions β^m. They have been used to generated biorthogonal spline wavelets (*Cohen*, 1992) with many favourable properties, making them well suited to semi-Lagrangian calculations. In particular, being combinations of cardinal B-splines, the scaling function and wavelet have an explicit formulation, making it easy to evaluate them at arbitrary points.

The dual functions are also compactly supported. The scaling function and the dual scaling function are symmetric, and an additional property is that the coefficients in the refinement equations are all dyadic rationals, so that they may be computed efficiently.

The scaling functions $\phi^m = \beta^m$ are characterised by the parameter m, and are piecewise polynomials of degree $m-1$, with continuous $(m-2)$th derivatives. A second parameter, $\tilde{m} \geq m$ is used to characterise the other functions. As $\tilde{m}$ increases, the regularity of the dual wavelet increases.

Some examples are shown in Figure 2. The support of each of the scaling functions is equal to the support of the corresponding filter. The wavelets have support $[-\tilde{m}/2, 1+\tilde{m}/2]$. Notice the lack of regularity of the dual functions for low filter lengths. It can be shown (*Cohen*, 1992) that $\tilde{\phi}^{m,\tilde{m}} \in C^k$ if $\tilde{m} > 4.1653m + 5.1653(k+1)$.

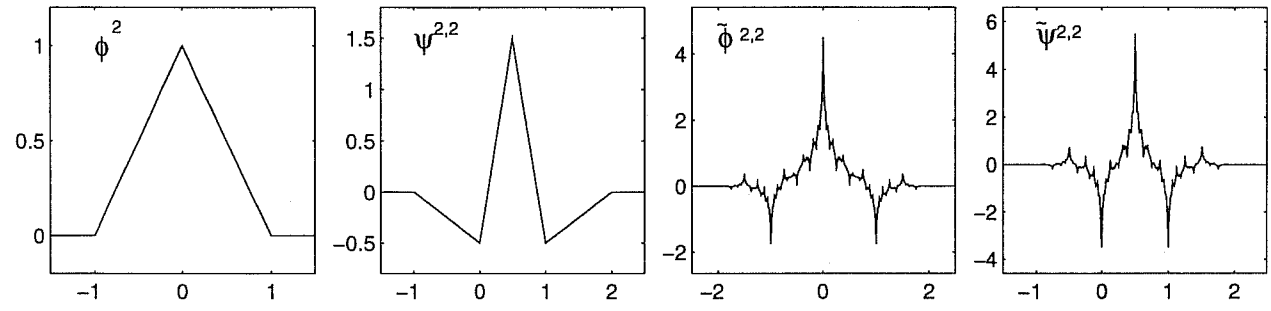

Fig. 2(a) The functions ϕ^2, $\psi^{2,2}$, $\tilde{\phi}^{2,2}$ and $\tilde{\psi}^{2,2}$.

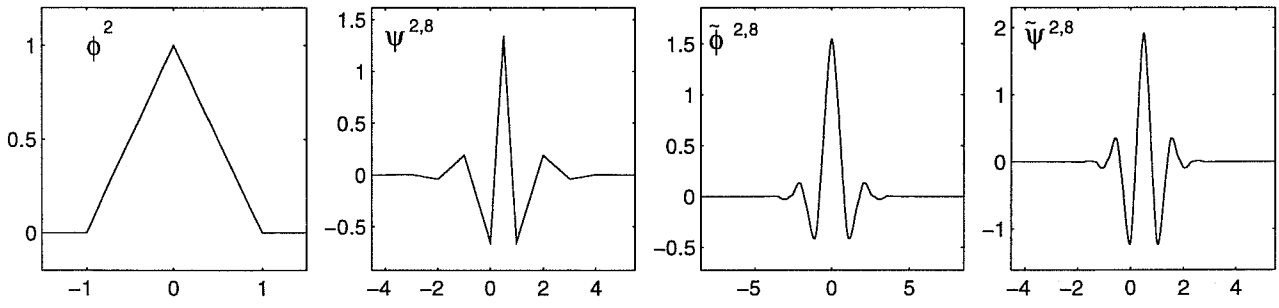

Fig. 2(b) The functions ϕ^2, $\psi^{2,8}$, $\tilde{\phi}^{2,8}$ and $\tilde{\psi}^{2,8}$.

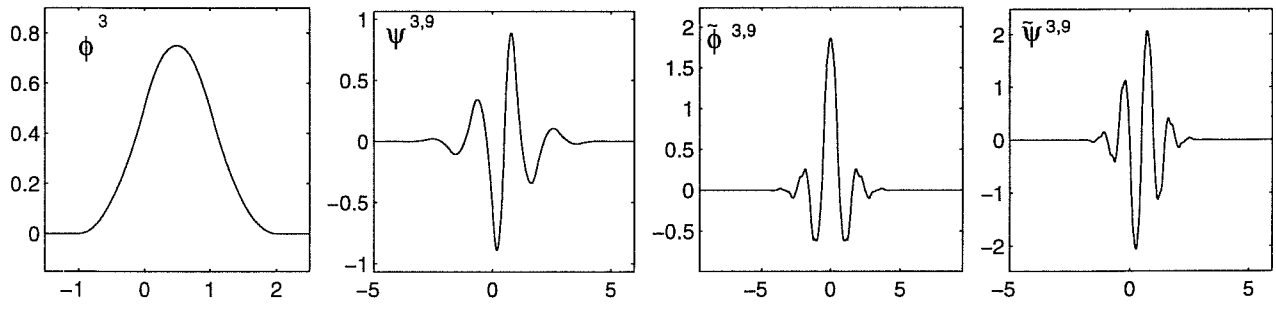

Fig. 2(c) The functions ϕ^3, $\psi^{3,9}$, $\tilde{\phi}^{3,9}$ and $\tilde{\psi}^{3,9}$.

3 A SEMI-LAGRANGIAN WAVELET METHOD

The novel feature of a semi-Lagrangian approach to the solution of a time-dependent partial differential equation, as opposed to an Eulerian approach, is the introduction of particle trajectories $X(x, t^*; t)$ and the necessity of evaluating functions at points such as $X(x_j, t)$, where x_j may lie on a regular grid, but $X(x_j, t^*; t)$ does not. With a Galerkin method, this arises in the calculation of inner products $\big(E(t^*;t)u, v\big)$, where $(E(t^*;t)u)(x) := u(X(x, t^*; t))$, and the integration is carried out by some form of numerical quadrature, with the quadrature points forming the 'grid'. It is this feature that determined the choice of wavelets to be used in this paper, since the biorthogonal spline wavelets are alone in having both compact support of the wavelets and their duals, and an explicit formula for the primary

scaling function and wavelet, thus making the exact evaluation of those functions at arbitrary points straightforward.

In the remainder of this paper we concentrate on this task. Other necessary components of the wavelet solution of (for example) convection-diffusion equations, such as the application of derivative operators, or the solution of Helmholtz problems, have been dealt with elsewhere, and will in due course be incorporated into the present work.

In the light of this we choose as a test case the inviscid Burgers equation,

$$u_t + uu_x = 0, \qquad x \in (0,1), \tag{12}$$

with initial data $u = f$, and with periodic boundary conditions. The particle trajectories X are solutions of the IVP

$$\frac{\mathrm{d}X(x,t^*;t)}{\mathrm{d}t} = u(X(x,t^*;t),t) \qquad \text{with} \qquad X(x,t^*;t^*) = x. \tag{13}$$

Then (12) can be written

$$\frac{\mathrm{d}u(X(x,t^*;t),t)}{\mathrm{d}t} = 0. \tag{14}$$

Thus u is constant along the space-time curves $(X(x,t^*;t),t)$, and then the fact that the rate of change of X with respect to t is in fact $u(X,t)$ gives the well-known result that these space-time curves are in fact straight lines.

With a semi-Lagrangian approach, coupled with a Petrov-Galerkin wavelet discretisation in space, as is the case when we are using biorthogonal wavelets, we seek a sequence of functions u^k, lying in V_j and approximating $u(\cdot, k\Delta t)$, such that

$$u^k = P_{V_j} E^k u^{k-1} = \sum_l (E^k u^{k-1}, \tilde{\phi}_{j,l})\phi_{j,l}, \qquad \text{with} \qquad u^0 = P_{V_j} f, \tag{15}$$

where we have written $E^k = E(t^k; t^{k-1})$.

Note that u^k is exanded in terms of the spline scaling functions $\phi_{j,l}$. There are two components of the above algorithm to be clarified.

1. The evaluation of the inner products is carried out using the compound trapezium rule. Since we are on a periodic domain this is a sufficiently accurate procedure. Moreover, the quadrature points are all dyadic points, and so the dual functions may be readily evaluated. If the quadrature points are chosen to have spacing $h = 2^{-j-s}$ $(s \geq 0)$, then an alternative to calculating the projection onto V_j directly is to first calculate the projection onto V_{j+s} (using the same points for the quadrature), and follow that by a wavelet decomposition down to V_j. This procedure not only generates the *same* values for the required coefficients, but gives approximate values for wavelet coefficients at levels $j, \ldots, j+s-1$.
2. The location of the trajectory feet $X(x_l, t^k; t^{k-1})$. It is straightforward to show that these satisfy the equation

$$X(x_l, t^k; t^{k-1}) = x - \Delta t u(x_l, t^k) \approx x_l - \Delta t P_{V_j} E^k u^{k-1}(x_l).$$

Since X appears on both sides of this equation, we solve it by means of either fixed-point or Newton iteration (the latter is straightforward since the derivative of the spline scaling functions is as easy to calculate as the original functions).

3.1 Adaptivity

With the above algorithm, however, we have made no use of the capabilities of the wavelet decomposition in separating out the contributions to u from different scales and different locations. Once this has been done, those that are judged to be insignificant can be discarded. Moreover, if at any stage it is found that u has a significant component at the finest scale representable by the current approximation space then the approximation can be expanded to include the next level of refinement.

This amounts to an adaptive projection, not onto a fixed space V_j, but onto the union $V_{J_0} + W_{J_0} + \cdots + W_{J_{\max}}$, where the level of $J_{\max}$ may be set relatively high because most of the degrees of freedom are not used (the corresponding coefficients are zero to within the set tolerance). Thus the adaptive method follows the same pattern as the method (15), but with the fixed projection P_{V_j} replaced by the adaptive projection P_{tol} whenever it appears.

In the next section we present results which demonstrate the potential of this approach.

4 SOLUTION OF BURGERS' EQUATION

We take as our initial datum the function f defined by

$$f(x) = \begin{cases} 1 + \sin^4 2\pi(x - 0.25) & \text{for } x \in [0.25, 0.75] \\ 1 & \text{otherwise.} \end{cases} \tag{16}$$

With this initial datum the solution shifts to the right and develops a shock at time $t = \frac{2}{3\pi\sqrt{3}} \approx 1.225$, which continues to move in the same direction, but at an altered speed. Thus as time increases, the contributions from the highest frequencies increase rapidly, culminating in the formation of the shock which of course immediately brings in contributions from the entire range of scales.

The experiments below all took the final time to be $t = 0.2$, so that the shock first forms just over 3/5ths of the way through the calculation. The other constant parameter is the accuracy to which the feet of the trajectories are calculated: the iterations for each point were stopped when the iterates changed by less than 10^{-4}. This never took more than two iterations for more than 20% of the points, and for the remainder was almost always accomplished in 3 iterations. A fixed-point iteration was used.

4.1 Experiment 1

The first experiment involved only a global adaptation: i.e. fixed projections were used, but after each projection a single level of wavelet decomposition was performed to see if there was any requirement for increasing the resolution. This was decided according to whether the maximum wavelet coefficient exceeded the given tolerance, which in this case was 10^{-3}. The wavelet parameters were $m = 3$ and $\tilde{m} = 9$, and 10 time steps were taken. The solutions are plotted in Figure 3.

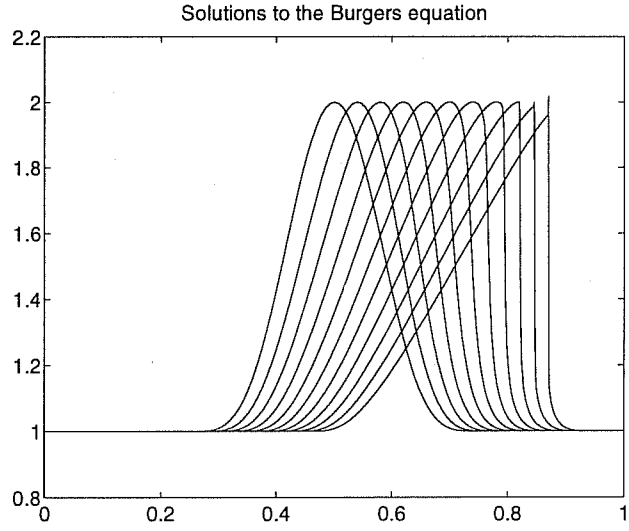

Fig. 3 The solutions at times $t = 0, 0.02, \ldots, 0.2$ using the globally adaptive method.

The formation of the shock and the onset of the collapse of the solution into the shock can be clearly seen. There is a slight overshoot, but the effect of this is contained within the immediate vicinity of the shock itself.

The maximum allowed level of refinementwas $J_{\max} = 10$, and the levels used at each time were as shown in Table 1.

time	0	0.02	0.04	0.06	0.08	0.1	0.12	0.14	0.16	0.18	0.2
level	7	7	7	8	8	9	10	10	10	10	10

Table 1: levels of refinement needed to resolve the solution to Burgers' equation in Experiment 1 (maximum allowed level was 10).

4.2 Experiment 2.

Here we introduce wavelet compression, by performing wavelet decompositions down to level 3 at every projection, and setting all wavelet coefficients falling below a tolerance of 10^{-3} to zero. The reduction in storage that this makes possible has not (yet) been implemented. The calculations were performed using the same parameters as Experiment 1, and the solutions can be seen in Figure 4.

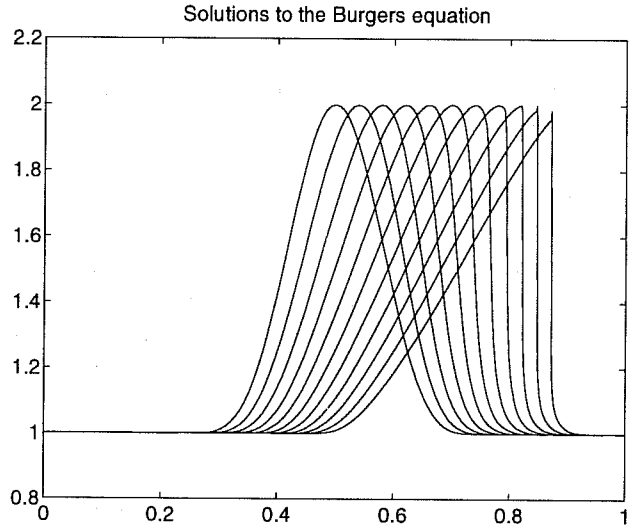

Fig. 4 The solutions at times $t = 0, 0.02, \ldots, 0.2$ using the fully adaptive method.

Table 2 contains information regarding the number of non-zero coefficients corresponding to each level of the wavelet decomposition for each time stage. The coarsest level was $J_0 = 3$, so there are two entries at that level corresponding to the scaling function and the wavelet coefficients. The finest two levels are unused until the shock starts to form, between $t = 1.2$ and $t = 1.4$. The total number of coefficients available is $2^{10} = 1024$, so it is evident that significant compression is possible. It is also evident from a comparison of Figure 3 and Figure 4 that the error incurred by performing the compression is minimal.

level	$t = 0$	0.02	0.04	0.06	0.08	0.1	0.12	0.14	0.16	0.18	0.2
9	0	0	0	0	0	0	1	4	5	5	4
8	0	0	0	0	0	0	1	3	4	5	5
7	0	0	0	0	0	2	3	5	4	4	5
6	0	0	0	1	4	5	4	4	5	4	5
5	4	4	4	5	5	4	4	5	5	4	4
4	8	8	8	8	7	6	9	6	7	7	6
3	6	5	5	5	6	5	6	7	5	6	6
3	8	8	8	8	8	8	8	8	8	8	8
total	26	25	25	27	30	30	36	42	43	43	43

Table 2: numbers of coefficients at each level of refinement at each time stage needed to resolve the solution to Burgers' equation in Experiment 2 (the min. level was 3, the max. allowed level was 10).

4.3 Experiment 3

This experiment was essentially the same as Experiment 2 except that the parameters of the scaling functions/wavelets were $m = 2$ and $\tilde{m} = 8$, as shown in Figure 2(b). The solutions are shown in Figure 5, and the numbers of coefficients used in Table 3.

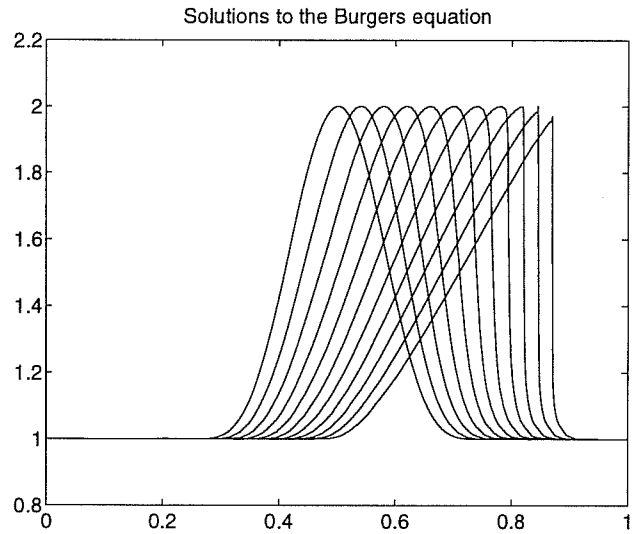

Fig. 5 The solutions at times $t = 0, 0.02, \ldots, 0.2$ from Experiment 3.

level	$t=0$	0.02	0.04	0.06	0.08	0.1	0.12	0.14	0.16	0.18	0.2
9	0	0	0	0	2	2	6	7	6	4	4
8	0	0	0	1	3	4	9	4	5	4	4
7	0	0	2	3	6	5	7	5	6	4	6
6	16	15	16	11	12	10	7	8	3	2	5
5	12	12	12	11	11	10	9	9	8	7	6
4	8	7	7	8	7	8	6	8	6	8	8
3	4	4	5	5	5	6	5	5	6	6	5
3	8	8	8	8	8	8	8	8	8	8	8
total	48	46	50	47	54	53	57	54	48	43	46

Table 3: numbers of coefficients at each level of refinement at each time stage needed to resolve the solution to Burgers' equation in Experiment 3 (the min. level was 3, the max. allowed level was 10).

A cursory comparison with Table 2 indicates that the higher order wavelets are better at compressing smooth functions.

4.4 **Experiment 4**

This last experiment is a brief and cursory look at the effect of varying the tolerance in the wavelet compression. We consider the projection of the initial datum using a range of tolerances. The wavelets/scaling functions are those shown in Figure 2(c), so that the scaling functions are 3rd order cardinal B-splines. The results are shown in Table 4, and are consistent with third order accuracy.

level	8	7	6	5	4	3	3	total
$\texttt{tol} = 10^{-3}$	0	0	0	4	8	6	8	26
$\texttt{tol} = 10^{-4}$	0	0	10	16	8	8	8	50
$\texttt{tol} = 10^{-5}$	0	24	30	16	10	8	8	96

Table 4: numbers of coefficients at each level of refinement needed to resolve to within tol the initial data for Burgers' equation in Experiment 4 (the min. level was 3).

5 CONCLUSION

The results of Section 4 demonstrate the potential of the wavelet approximation for obtaining significant reduction of the amount of storage needed to retain the significant information regarding the solution to a PDE as it evolves in time, adapting itself to the changing requirements. Whether this results in a real reduction in the amount of work involved has not been addressed here; other authors have reported mixed results in this regard (*Wells*, 1992).

The work described here is at an early stage. Possibilities for the next step include:

- The use of other families of wavelets: for example, the popular Daubechies orthogonal wavelets (*Daubechies*, 1988), or some kind of interpolating wavelets (*Donoho*, 1992) could perhaps be used. Here one problem would be the efficient & accurate evaluation of these at arbitrary points.
- The incorporation of solvers for the elliptic problems that arise when diffusion terms are introduced.
- The extension to multidimensions and to non-periodic/non-rectangular domains.
- Analysis of the accuracy of the quadrature formulae used and the implications for stability.

The main conclusion of this paper is that the semi-Lagrangian method can be effectively implemented in the context of a wavelet approximation. The biorthogonal spline wavelets have been shown to perform well, even in the presence of a shock.

REFERENCES

Bacry, E., Stéphane Mallat and George Papanicolaou, 1992: A wavelet-based space-time adaptive numerical method for partial differential equations. Mathematical Modelling and Numerical Analysis, 26(7), 793–834.

Beylkin, G., 1993: On wavelet-based algorithms for solving differential equations. In: Wavelets: Mathematics and Applications, ed. J. Benedetto and M. Frazier. CRC Press, Boca Raton, 449–466.

Cohen, A., I. Daubechies and J. -C. Feauveau, 1992: Biorthogonal bases of compactly supported wavelets. Communications on Pure and Applied Mathematics, XLV(5), 485–560.

Daubechies, I., 1988: Orthonormal bases of compactly supported wavelets. Communications on Pure and Applied Mathematics, XLI, 909–996.

Donoho, D., 1992: Interpolating wavelet transforms. Preprint, Department of Statistics, Stanford University.

Mallat, S. G., 1989: Multiresolution approximations and wavelet orthonormal bases of $L^2(\mathbf{R})$. Trans. Amer. Math. Soc., 315(1), 69–87.

Meyer, Y., 1993: Wavelets: algorithms and applications. (Translated and revised by R. D. Ryan.) Society for Industrial and Applied Mathematics, Philadelphia.

Reider, A., R. O. Wells Jr. and X. Zhou, 1994: A wavelet approach to robust multilevel solvers for anisotropic elliptic problems. Applied and Computational Harmonic Analysis, 1, 355–367.

Qian, S., and J. Weiss, 1993: Wavelets and the numerical solution of partial differential equations. Journal of Computational Physics, 106, 155–175.

Wells Jr., R. O. and X. Zhou, 1992: Wavelet interpolation and approximate solutions of elliptic partial differential equations. Technical Report TR92-03, Dept. of Math. Rice Inversity, Houston, Texas, TX 77251-1892.