JOURNAL OF COMPUTATIONAL PHYSICS 120, 33–47 (1995)

A Multilevel Wavelet Collocation Method for Solving Partial Differential Equations in a Finite Domain

OLEG V. VASILYEV, SAMUEL PAOLUCCI, AND MIHIR SEN

Department of Aerospace and Mechanical Engineering, University of Notre Dame, Notre Dame, Indiana 46556

Received June 16, 1994; revised January 31, 1995

A multilevel wavelet collection method for the solution of partial differential equations is developed. Two different approaches of treating general boundary conditions are suggested. Both are based on the wavelet interpolation technique developed in the present research. The first approach uses wavelets as a basis and results in a differential-algebraic system of equations, where the algebraic part arises from boundary conditions. The second approach utilizes extended wavelets, which satisfy boundary conditions exactly. This approach results in a system of coupled ordinary differential equations. The method is tested on the one-dimensional Burgers equation with small viscosity. The solutions are compared with those resulting from the use of other numerical algorithms. The present results indicate that the method is competitive with well-established numerical algorithms.

1. INTRODUCTION

The nonlinear partial differential equations which describe physical phenomena, *e.g.*, the equations of fluid mechanics, are usually not susceptible to analytic solution. Conventional methods used for numerical solutions of partial differential equations mostly fall into three classes: finite difference methods, finite element methods, and spectral methods. Briefly, the finite difference method consists in defining the different unknowns by their values on a discrete (finite) grid and in replacing differential operators by difference operators using neighboring points. In the finite element method the equations are integrated against a set of linear independent test functions with small compact support, and the solution is considered as a linear combination of this set of test functions. In spectral methods, the unknown functions are developed along a basis of functions having global support. This development is truncated to a finite number of terms which satisfy a system of coupled ordinary differential equations in time. The advantage of using either of the first two numerical techniques is the simplicity in adapting to complex geometries, while the main advantage of spectral methods is the greater accuracy.

If the solution of a partial differential equation is regular, any of three above-mentioned numerical techniques can be applied successfully. However, singularities and sharp transitions in solutions can be observed in many physical phenomena such as the formation of shock waves in compressible gas flow, the formation of vortex sheets in high Reynolds number incompressible flow, and burst events in the wall region of the turbulent boundary layer. A characteristic feature of such phenomena is that the complex behavior occurs in a small region of space and possibly intermittent in time. This makes them particularly difficult to resolve numerically using the above-mentioned methods. Spectral methods are not easily implemented because the irregularity of the solution causes the loss of high accuracy. Moreover, the global support of the basis functions induces the well-known Gibbs phenomenon. Accurate representation of the solution in regions where singularities or sharp transitions occur requires the implementation of adaptive finite difference or finite element methods (see, for example, [1–5]). In these methods an automatic error estimation step determines locally whether the current resolution of the numerical solution is sufficient or if a finer grid is necessary. The main difficulty of adaptive methods is finding stable accurate difference operators at the interface between grids of possibly very different sizes.

Wavelet analysis is a new numerical concept which allows one to represent a function in terms of a set of basis functions, called wavelets, which are localized both in location and scale. As we noted earlier, spectral bases are infinitely differentiable, but have global support. On the other hand, basis functions used in finite difference or finite element methods have small compact support, but have poor continuity properties. As a result, spectral methods have good spectral localization, but poor spatial localization, while finite difference and finite element methods have good spatial localization, but poor spectral localization. Wavelet bases seem to combine the advantages of both spectral and finite difference (or finite element) bases. One can expect that numerical methods based on wavelet bases are able to attain good spatial and spectral resolution.

Recently much effort has been put into developing schemes based on properties of orthonormal wavelet bases introduced by Meyer [6] and Stromberg [7]. These bases are formed by the dilation and translation of a single function $\psi(x)$,

$$\psi_{j,k}(x) = 2^{-j/2}\psi(2^{-j}x - k), \tag{1}$$

0021-9991/95 $12.00

where $j, k \in Z$. Then for certain $\psi(x)$, the sequence of functions $(\psi_{j,k}(x))_{(j,k)\in Z^2}$ forms an orthonormal basis in $L^2(R)$. For more information we refer to [8].

In wavelet applications to the solution of partial differential equations the most frequently used wavelets are those with compact support introduced by Daubechies [8]. Exploration of the usage of Daubechies wavelets to solve partial differential equations has been undertaken by a number of investigators such as Beylkin *et al.* [9], Latto and Tenenbaum [10], Bacry *et al.* [11], Schult and Wyld [12], and Qian and Weiss [13]. Additional references can be found in the recent review by Jawerth and Sweldens [14].

Most of the wavelet algorithms for solving partial differential equations can handle periodic boundary conditions easily (see, for example, [13, 15]). The treatment of general boundary conditions is still an open question even though different possibilities of dealing with this problem has been studied (see [16–19]). The variational approach suggested by Glowinski *et al.* [16] is not applicable for some nonlinear problems; furthermore, it is impractical for higher dimensions. A different way of treating general boundary conditions is to use wavelets specified on an interval as suggested by Meyer [17] and Andersson *et al.* [18]. These wavelets are constructed satisfying certain boundary condition. The disadvantages of this approach are inconvenience of implementation and wavelet dependence on boundary conditions. A more satisfactory approach is to use antiderivatives of wavelets as trial functions, as done by Xu and Shann [19]. In this way the singularities in the wavelets are smoothened and the boundary conditions can be treated more easily.

Despite the relative success of the above-mentioned methods all of them have one intrinsic difficulty which is the treatment of nonlinearities in the equation. All of them to some extent first map the space of wavelet coefficients onto the physical space, compute the nonlinear term in physical space, and then project the result back to the wavelet coefficients space using analytical quadrature formulas or numerical integration (see, for example, [9, 12, 13]). This procedure is not quite acceptable since it is computationally expensive. It is desirable to have an algorithm which operates only in one space without going back and forth between the space of wavelet coefficients and physical space. One way of doing this is to use a collocation method analogous to the spectral collocation method.

The main objective of the paper is to present a new numerical approach for solving partial differential equations with arbitrary boundary conditions. In the presentation of the method we try to be as general as possible, giving only the main philosophy of the method and leaving some freedom for further exploration of its applications. Even though we do not try to predict what wavelet is the best for our algorithm (it is simply impossible, due to the fact that some wavelets work better for some problems and worse in others), we give some suggestions on wavelet applicability for the present algorithm. We illustrate the method using different wavelets; by doing so we illustrate the generality of the approach. The proposed method utilizes the classical idea of collocation to the wavelet approximation of partial differential equations. In an independent study, Bertoluzza et al. [20] use a different approach to incorporate the collocation idea with the auto-correlation function of Daubechies scaling functions to solve linear boundary value problems.

Three issues are addressed in this work. The first is how to incorporate the idea of collocation with wavelet bases to construct an effective algorithm of solving partial differential equations. The second is how to deal with boundary conditions. The last is how to construct a stable, accurate, and efficient numerical algorithm.

The rest of the paper is organized as follows. In Section 2 we discuss the wavelet interpolation technique. A brief description of the concepts of frames and Riesz bases are also given there. The numerical method of solving partial differential equations is described in Section 3. In this section we also discuss the efficient treatment of boundary conditions and the generalization of the method to higher dimensions. Finally, in Section 4 the numerical method is applied to the solution of Burgers equation. A comparison with other algorithms is given there as well.

2. WAVELET INTERPOLATION

In order to develop an algorithm which can utilize different wavelets, we try to be as general as possible. Thus we base our discussion on wavelet frames since the concept of frames is more general and includes Riesz and orthonormal bases. To illustrate the algorithm we use the correlation function of the Daubechies scaling functions of different orders (see Beylkin and Saito [21]) and the Gaussian family of wavelets which are given by

$$\psi^n(x) = (-1)^n \frac{d^n}{dx^n} \exp\left(-\frac{x^2}{2}\right) = \mathrm{He}_n(x) \exp\left(-\frac{x^2}{2}\right), \quad (2)$$

where $\mathrm{He}_n(x)$ is the modified Hermite polynomial. The choice of these two wavelets is rather arbitrary.

Let us briefly review the definition of frames and the definition of a Riesz basis. For details see Daubechies [8].

DEFINITION. A family of functions $(\phi_j)_{j\in Z}$ in a Hilbert space $H(R)$ with the inner product defined by $\langle f, g\rangle_R = \int_R f(x)g(x)dx$ is called a *frame* if there exist $A > 0$, $B < \infty$ so that for all $f \in H(R)$,

$$A\|f\|^2_{L^2(R)} \le \sum_{j\in Z} |\langle f, \phi_j\rangle_R|^2 \le B\|f\|^2_{L^2(R)}, \quad (3)$$

where $\|f\|^2_{L^2(R)} = \langle f, f\rangle_R$. A and B are called the frame bounds.

If the two frame bounds are equal, $A = B$, then it is a tight frame. Note if $(\phi_j)_{j\in Z}$ is a tight frame and if $\|\phi_j\|_{L^2(R)} = 1$ for all $j \in Z$, then the frame bound gives the "redundancy ratio," the measure of how much the frame is overdetermined. If this

redundancy ratio, as measured by A, is equal to 1, then the tight frame is an orthonormal basis.

DEFINITION. A set of functions $\{(\phi_k)_{k\in Z}\}$ in a Hilbert space $H(R)$ is called a *Riesz* basis if there exist constants $0 < A < B < \infty$ such that for every infinite square-summable sequences $\{c_k : k \in Z\}$ (that is, $\|\{c_k\}\|^2_{l^2(Z)} < \infty$)

$$A\|\{c_k\}\|^2_{l^2(Z)} \le \left\|\sum_{k\in Z} c_k\phi_k(x)\right\|^2_{L^2(R)} \le B\|\{c_k\}\|^2_{l^2(Z)} \tag{4}$$

and the linear span of $\{(\phi_k)_{k\in Z}\}$ is dense in $H(R)$. Here, A and B are called the Riesz bounds.

Note that the following two statements are equivalent (see [22]):

(i) $(\phi_j)_{j\in Z}$ is a Riesz basis of $L^2(R)$.

(ii) $(\phi_j)_{j\in Z}$ is a frame of $L^2(R)$ and is also a linearly independent family.

Furthermore, the Riesz bounds and frame bounds are the same.

It can be shown (see [8]) that if $a_0 > 0$, $a_j = 2^{-j}a_0$, $b^j_k = a_j b_0 k$, and $\psi^j_k(x) = a_j^{-1/2}\psi((x - b^j_k)/a_j)$, where $\psi(x)$ is a wavelet, then there exists a range or a discrete set of spacing parameter $b_0 > 0$ such that the set of wavelets $\{\psi^j_k(x) : j, k \in Z\}$ constitutes either a frame (as in the case of the Gaussian wavelets) or an orthogonal basis in $H(R)$. Note that the wavelet $\psi(x)$ is a smooth and well-localized function, which means that $|\psi^{(m)}(x)| \le C \exp(-\alpha|x|)$ for $x \in R$, $m \le M$, and $\alpha > 0$. For clarity of discussion we will call wavelets corresponding to the same j as wavelets of the j level of resolution. For convenience of notation we use the superscript to denote the level of resolution and the subscript to denote the location in physical space (with the exception of a_j). Introducing $W^j(R) = \text{span}_{k\in Z, x\in R}\psi^j_k(x)$ it can be shown that $H(R) = \bigcup_{j=-\infty}^{+\infty} W^j(R)$. Note that for the correlation function of the Daubechies scaling function the set $\{\psi^j_k(x) : k \in Z\}$ is a Reisz basis for $W^j(R)$ (see [20]).

Up to this point we have been discussing frames on the real line, even though we are interested in applications to a finite domain. So one may ask how a frame, defined on the real line, can be applied to a finite domain. Let us consider the closed interval $\Omega \equiv [x_l, x_r]$. In this case we take $a_j = 2^{-j}a_0$, and $b^j_k = (x_r + x_l)/2 + a_j b_0 k$, where $a_0 = 2^{-L}(x_r - x_l)/b_0$, $L \in Z$; then the set of wavelets $\{\psi^j_k(x) : j, k \in Z\}$ still constitutes either a frame or an orthogonal basis in $H(R)$. Note that for any j there are values of k such that the wavelets $\psi^j_k(x)$ are centered outside of the domain Ω. For clarity of discussion, all wavelets whose centers are located within the domain, will be called internal wavelets; all other wavelets will be called external wavelets.

Let us consider a function $u(x) \in H(\Omega)$. We introduce the function $\tilde{u}(x) \in H(R)$, $\tilde{u}(x) \in C^M(R)$ in such a way that $\tilde{u}(x) = u(x)$ for all $x \in \Omega$ and $\tilde{u}(x)$ decays to zero "fast enough" outside of the domain Ω. Note that discontinuities of $\tilde{u}^{(M+1)}(x)$ may occur only at boundaries of the domain. Then due to the fact that the set of wavelets $\{\psi^j_k(x) : j, k \in Z\}$ constitutes a frame in $H(R)$ there exist c^j_k such that

$$\tilde{u}(x) = \sum_{j=-\infty}^{+\infty}\sum_{k=-\infty}^{+\infty} c^j_k\psi^j_k(x) \quad \text{for } x \in R. \tag{5}$$

Based on the construction of $\tilde{u}(x)$ we can write

$$u(x) = \sum_{j=-\infty}^{+\infty}\sum_{k=-\infty}^{+\infty} c^j_k\psi^j_k(x) \quad \text{for } x \in \Omega. \tag{6}$$

Equation (6) answers our question. This equation is the starting point for future discussions. Let $W^j(\Omega) = \text{span}_{k\in Z, x\in\Omega}\psi^j_k(x)$. Let us prove the following proposition.

PROPOSITION 1. *For any $\varepsilon > 0$ there exists an integer L such that for $j < 0$ and $k \in Z$ there exists a constant C such that $\|\psi^j_k(x) - C\|_{L^2(\Omega)} < \varepsilon$.*

Proof. Since the wavelet $\psi(x)$ is Hölder continuous with exponent α (see [8]), i.e., $|\psi(x) - \psi(x + t)| \le C_H|t|^\alpha$ for all x, $t \in R$, we can write $\|\psi^j_k(x) - \psi^j_k(b^j_0)\|^2_{L^2(\Omega)} \le 2^{(L+j)(2\alpha+1)}A$, where $A = 2^{-2\alpha}b_0^{2\alpha+1}(2\alpha + 1)^{-1}C^2_H$. From the above inequality the proposition follows for $L \le (2\alpha + 1)^{-1}\log_2(\varepsilon^2/A)$. ∎

In other words, Proposition 1 means that for a proper choice of L all the levels below $j = 0$ can be approximated by a constant, so for the numerical approximation all these lower levels can be excluded. Note that if $L + j - 1 < 0$ then there is only one wavelet inside of the domain; the rest are located outside. For $L + j - 1 \ge 0$ we have that $b^j_{-2^{L+j-1}} = x_l$ and $b^j_{2^{L+j-1}} = x_r$. In order not to complicate the notation, each time we use the index 2^{L+j-1} we mean the integer part of it, *i.e.*, $[2^{L+j-1}]$.

The next question which arises naturally is whether all wavelets have significant influence in the approximation (6); if not then we can keep only wavelets which are essential to the approximation. Note that any wavelet is either zero outside of its support (as in the case of Daubechies wavelets) or decays fast enough away from its center (as in the case of Gaussian wavelets). Thus there is always a finite number of wavelets that are located outside of the domain Ω and whose influence inside of the domain must be accounted. Thus at any level of resolution there is a subset of integers $\{Z^j_\Omega : -2^{L+j-1} - N_l, \ldots, 2^{L+j-1} + N_r\}$ such that for $k \in Z^j_\Omega$ the wavelet ψ^j_k affects the interior of the domain Ω.

PROPOSITION 2. *For any $\varepsilon > 0$ and $j \in Z$ there exists a finite integer set Z^j_Ω such that for $k \notin Z^j_\Omega$ $\|\psi^j_k(x)\|_{L^2(\Omega)} < \varepsilon$.*

Proof. Without loss of generality let us consider only wavelets which are to the right of the domain Ω. Since $b^j_{2^{L+j-1}+1} > x_r$, then $\|\psi^j_k(x)\|^2_{L^2(\Omega)} = \int_{x_l}^{x_r}(\psi^j_k(x))^2\,dx \le \int_{-\infty}^{b^j_{2^{L+j-1}+1}}(\psi^j_k(x))^2\,dx$. Let $\xi = (x - b^j_k)/a_j$ then $\|\psi^j_k(x)\|^2_{L^2(\Omega)} \le \int_{-\infty}^{b_0(2^{L+j-1}+1-k)}\psi^2(\xi)\,d\xi \le$

$\int_{-\infty}^{-b_0 N_r} \psi^2(\xi)\, d\xi$ for $k > 2^{L+j-1} + N_r$. Since $\psi \in L^2(R)$ then there exists an integer N_r such that $\int_{-\infty}^{-b_0 N_r} \psi^2(\xi)\, d\xi \leq \varepsilon^2$. Then it follows that $\|\psi_k^j(x)\|_{L^2(\Omega)} \leq \varepsilon$. ∎

The proof for the existence of N_l is analogous. Note that for symmetrical wavelets the number of wavelets centered on either side of the domain is the same, $N_l = N_r$.

Both Propositions 1 and 2 are essential for understanding why some wavelets are retained, while others can be omitted in the approximation (6). Even though in the proofs of the propositions we gave the estimates for L, N_l, N_r, these estimates are conservative. In numerical applications L can be taken much larger and N_l and N_r can be taken considerably less than the estimates. Note that the size of the domain Ω characterizes the largest scale which could possibly be present in $u(x) \in H(\Omega)$, while in approximation (6) for $j \geq 0$ the largest scale is determined by L. That is why in most practical applications $L = 1$, even though in some cases L can be taken larger. In numerical computations, resolution is limited by memory constraints and computational times. Thus we truncate the approximation (6) at the finest level of resolution J. Consequently levels 0 and J correspond respectively to the coarsest and finest scales present in the approximation.

Let $W_\Omega^j(\Omega) = \text{span}_{k \in Z_\Omega^j, x \in \Omega} \psi_k^j(x)$. We note that in general $W_\Omega^j(\Omega) \subseteq W^j(\Omega)$. The approximation of the function $u(x)$ can now be written as

$$u^J(x) = \sum_{j=0}^{J} \sum_{k \in Z_\Omega^j} c_k^j \psi_k^j(x), \tag{7}$$

where clearly $u^J(x) \in \bigcup_{j=0}^{J} W_\Omega^j(\Omega)$. Note that in general coefficients in (7) are unique only if all wavelets at different locations and levels are linearly independent. Nevertheless, the nonuniqueness of the coefficients c_k^j does not preclude the existence of a stable reconstruction algorithm. One way of finding the coefficients c_k^j is based on iterative reconstruction of the function from the inner product of the function itself with wavelets and making use of the frame bounds as suggested in [6]. Since we are interested in the collocation method, we cannot make use of the above-mentioned algorithm. Let us briefly discuss the essence of our method of determining coefficients c_k^j. We start from the coarsest level of resolution and progressively move to the finest level. On each level the coefficients of the lower levels are fixed, so we only obtain the coefficients corresponding to that level. Let us rewrite approximation (7) for any intermediate resolution level j $(0 \leq j \leq J)$ as

$$\begin{aligned} u^j(x) &= u^{j-1}(x) + \sum_{k \in Z_\Omega^j} c_k^j \psi_k^j(x), \qquad 1 \leq j \leq J, \\ u^0(x) &= \sum_{k \in Z_\Omega^0} c_k^0 \psi_k^0(x). \end{aligned} \tag{8}$$

Thus the approximation $u^j(x)$ is nothing but a refinement of the approximation $u^{j-1}(x)$. Details of the algorithm will be given later.

It is important for applications to be able to recover an arbitrary function from a discrete set of sampled values. Let $\{x_i\}$ be a set of locations in Ω at which the function $u(x) \in H(\Omega)$ is sampled and $\{u_i \equiv u(x_i)\}$ the corresponding sampled values. We call the operator I the interpolation operator if

$$\begin{aligned} I(u)(x) &= \sum_i I_i(x) u_i, \\ I_i(x_k) &= \delta_{i,k}, \end{aligned} \tag{9}$$

where $\delta_{i,k}$ is the Kronecker delta symbol and there exists a small constant $\varepsilon > 0$ such that

$$\|u(x) - I(u)(x)\|_{L^2(\Omega)} \leq \varepsilon \|u(x)\|_{L^2(\Omega)}. \tag{10}$$

Let $\{x_i^j : i \in Z_\Omega^j\}$be a set of collocation points at the j level of resolution. Then let us evaluate equations (8) at x_i^j collocation points, so we have

$$\begin{aligned} u^j(x_i^j) &= u^{j-1}(x_i^j) + \sum_{k \in Z_\Omega^j} c_k^j \psi_k^j(x_i^j), \\ &\qquad 1 \leq j \leq J, 0 \leq l \leq J, i \in Z_\Omega^l, \\ u^0(x_i^l) &= \sum_{k \in Z_\Omega^0} c_k^0 \psi_k^0(x_i^l), \quad 0 \leq l \leq J, i \in Z_\Omega^l. \end{aligned} \tag{11}$$

For convenience of notation we introduce the operator

$$A_{i,k}^{l,j} = \psi_k^j(x_i^l), \quad 0 \leq l, j \leq J, i \in Z_\Omega^l, k \in Z_\Omega^j, \tag{12}$$

which measures the contribution of wavelet ψ_k^j at location x_i^l.

We choose the set of collocation points in such a way that for any j $(0 \leq j \leq J - 1)$ the following relation between the collocation points of different levels of resolution is satisfied

$$\{x_i^j\} \subset \{x_i^{j+1}\}. \tag{13}$$

This relation between collocation points of different levels enables us to have the same values of the function at different levels of resolution at the same collocation points. In other words, the relation between the values of the function at different levels of resolution is a simple restriction, *i.e.*, for $0 \leq l \leq j \leq J$, $u^l(x_i^l) = u^j(x_m^j)$ if $x_i^l = x_m^j$. Thus we can define the *restriction* operator $R_{i,m}^{l,j}$ as

$$R_{i,m}^{l,j} = \begin{cases} 1 & \text{for } x_i^l = x_m^j, \\ 0 & \text{otherwise.} \end{cases} \tag{14}$$

Then

$$u_i^l = \sum_{m \in Z_\Omega^j} R_{i,m}^{l,j} u_m^j, \quad 0 \le l \le j \le J, i \in Z_\Omega^l, \tag{15}$$

where $u_i^j = u^j(x_i^j)$. Note that $u^j(x_i^l) = u_i^l$ for $0 \le l \le j \le J$.

Once we have the approximation at a certain level, we can find its value at the higher level of resolution, although in general the value is not the same (with the exception of the collocation points corresponding to both levels). Thus based on the interpolation properties of the wavelet approximation, we can introduce the *prolongation* $P_{i,m}^{l,j}$ operator so that

$$u^j(x_i^l) = \sum_{m \in Z_\Omega^j} P_{i,m}^{l,j} u_m^j, \quad 0 \le j \le l \le J, i \in Z_\Omega^l. \tag{16}$$

We will define the *prolongation* operator later, since it requires some preliminary work. At this point we assume that it exists and it is known.

Taking $l = j$ in (11) and using (15) and (16), we can write

$$\sum_{m \in Z_\Omega^s} \Delta_{i,m}^{j,s} u_m^s = \sum_{k \in Z_\Omega^j} A_{i,k}^{j,j} c_k^j, \quad 0 \le j \le s \le J, i \in Z_\Omega^j, \tag{17}$$

where $A_{i,k}^{j,j}$ is a $(2^{L+j} + N_l + N_r + 1) \otimes (2^{L+j} + N_l + N_r + 1)$ matrix defined by (12) and the operator $\Delta_{i,m}^{j,s}$ is defined as

$$\Delta_{i,m}^{j,s} = \begin{cases} R_{i,m}^{j,s} - \sum_{p \in Z_\Omega^{j-1}} P_{i,p}^{j,j-1} R_{p,m}^{j-1,s}, & 1 \le j \le s \le J, i \in Z_\Omega^j, m \in Z_\Omega^s, \\ R_{i,m}^{0,s}, & j = 0, 0 \le s \le J, i \in Z_\Omega^0, m \in Z_\Omega^s. \end{cases} \tag{18}$$

Then for an appropriate choice of collocation points the matrix $A_{i,k}^{l,j}$ is not singular, so we can write that

$$c_k^j = \sum_{m \in Z_\Omega^s} \mathbb{C}_{k,m}^{j,s} u_m^s, \quad 0 \le j \le s \le J, k \in Z_\Omega^j, \tag{19}$$

where

$$\mathbb{C}_{k,m}^{j,s} = \sum_{p \in Z_\Omega^j} (A^{j,j})_{k,p}^{-1} \Delta_{p,m}^{j,s}, \quad 0 \le j \le s \le J, k \in Z_\Omega^j, m \in Z_\Omega^s. \tag{20}$$

Note that for any fixed j, $(A^{j,j})_{k,p}^{-1}$ denotes the (k, p) element of the inverse of the matrix $\mathbf{A}^{j,j}$, while operator $\mathbb{C}_{k,m}^{j,s}$ maps the set of functional values at the s level of resolution into the set of wavelet coefficients at the j level.

Now we are in position to define the *prolongation* operator. Let us rewrite Eqs. (11) as

$$\sum_{m \in Z_\Omega^l} P_{i,m}^{l,j} u_m^j = \sum_{m \in Z_\Omega^l} \sum_{p \in Z_\Omega^{j-1}} P_{i,p}^{l,j-1} R_{p,m}^{j-1,j} u_m^j + \sum_{k \in Z_\Omega^j} A_{i,k}^{l,j} c_k^j,$$

$$1 \le j \le l \le J, i \in Z_\Omega^l, \tag{21}$$

$$\sum_{m \in Z_\Omega^l} P_{i,m}^{l,0} u_m^0 = \sum_{k \in Z_\Omega^0} A_{i,k}^{l,0} c_k^0, \qquad j = 0, 0 \le l \le J, i \in Z_\Omega^l.$$

Substituting (19) into (21) the following expression for the *prolongation* operator can be obtained

$$\mathrm{P}_{i,m}^{l,j} = \begin{cases} \sum_{p \in Z_\Omega^{j-1}} P_{i,p}^{l,j-1} R_{p,m}^{j-1,j} + \sum_{k \in Z_\Omega^j} A_{i,k}^{l,j} \mathbb{C}_{k,m}^{j,j}, & \\ \qquad 1 = j \le l \le J, i \in Z_\Omega^l, m \in Z_\Omega^j, & \\ \sum_{k \in Z_\Omega^0} A_{i,k}^{l,0} \mathbb{C}_{k,m}^{0,0}, \qquad j = 0, 0 \le l \le J, i \in Z_\Omega^l, m \in Z_\Omega^0. & \end{cases} \tag{22}$$

Now we see that, since the restriction operator is known, then we have explicit form for $\Delta_{i,m}^{0,s}$ and, consequently, for both $\mathbb{C}_{k,m}^{0,0}$, and $P_{i,m}^{l,0}$. Then using (18), (20), and (22) all three operators $\Delta_{i,m}^{j,s}$, $\mathbb{C}_{k,m}^{j,s}$, and $P_{i,m}^{l,j}$ are obtained recursively.

Now we are in the position to start the discussion on interpolation. We define the interpolation operator as

$$u^j(x) = \sum_{i \in Z_\Omega^j} I_i(x) u_i^j, \tag{23}$$

where

$$I_i(x) = \sum_{j=0}^{J} \sum_{k \in Z_\Omega^j} \psi_k^j(x) \mathbb{C}_{k,i}^{j,j}, \qquad i \in Z_\Omega^j. \tag{24}$$

It can be observed that the success of the interpolation depends strongly on the invertibility of the matrix $A_{i,k}^{l,j}$. Let $|\lambda|_{\min}$ and $|\lambda|_{max}$ correspond to the minimum and maximum absolute values of eigenvalues of the matrix $A_{i,k}^{l,j}$. From the computational standpoint, if the condition number, defined as $C(A_{i,k}^{l,j}) = |\lambda|_{\max} / |\lambda|_{\min}$, is large then the inverse of the matrix $A_{i,k}^{l,j}$ is inaccurate. Thus one would like to have the condition number as small as possible. It can be shown that

$$|\lambda|_{\min} \|\{c_k^j\}\|_{l^2(Z_\Omega^j)}^2 \le \left\| \left\{ \sum_{k \in Z_\Omega^j} c_k^j \psi_k^j(x_i^j) \right\} \right\|_{l^2(Z_\Omega^j)}^2 \le |\lambda|_{\max} \|\{c_k^j\}\|_{l^2(Z_\Omega^j)}^2, \tag{25}$$

where $\|\{c_i\}\|_{l^2(Z)}^2 = \sum_{i \in Z} |c_i|^2$. Note that for any j, the set $\{\psi_k^j(x) : k \in Z_\Omega^j\}$ is a Riesz basis for $W_0^j(\Omega)$ if

$$A \|\{c_k^j\}\|_{l^2(Z_\Omega^j)}^2 \le \left\| \sum_{k \in Z_\Omega^j} c_k^j \psi_k^j(x) \right\|_{L^2(\Omega)}^2 \le B \|\{c_k^j\}\|_{l^2(Z_\Omega^j)}^2. \tag{26}$$

Due to the equivalency of the norms the two statements (25)

and (26) are equivalent. What is more, it can be shown that in the limit of $a_j \to 0$ the ratio $|\lambda|_{\max}/|\lambda|_{\min} \to B/A$. The last relationship is very important because, based on A and B in (26) one can predict the condition number of the matrix A^{jj}_{ik}. Even though we do not exploit the fact that A and B, and thus the condition number, can be analytically evaluated, this feature can be very important in some applications.

Once the collocation points are known and the resulting matrix is well conditioned the interpolation operator can be constructed. In some application the derivatives of the interpolation function is very important. In light of (7), (12), (23), and (24) the mth derivative of the approximate function can be written as

$$u^{J(m)}(x) = \sum_{i \in Z^J_\Omega} D^{(m)}_i(x) u^J_i, \tag{27}$$

where

$$D^{(m)}_i(x) = \sum_{j=0}^{J} \sum_{k \in Z^j_\Omega} \psi^{j(m)}_k(x) \mathbb{C}^{jJ}_{km}, \qquad i \in Z^J_\Omega. \tag{28}$$

Note that $D^{(0)}_i(x) = I_i(x)$.

Before proceeding any further we want to say a few words regarding the two different wavelets we use to illustrate the algorithm. The first is the correlation function of the Daubechies scaling function of order 5 and $b_0 = 1.0$ (see Beylkin and Saito [21]). We choose the order 5 as a compromise between the requirement on continuity of the second derivative (which we will need later) and the demand to have the support as small as possible. The second belongs to the Gaussian family of wavelets given by (2), where we use $n = 2$ and various values of b_0. This wavelet is also referred to in the literature as the "Mexican hat" because of its distinctive shape. Keeping in mind, that our goal is to illustrate the method, we will focus our discussion on these two functions, although we will discuss the peculiarities related to the use of wavelets of different orders. In addition, since both functions are symmetrical, we use $N_l = N_r = N$. Note that the correlation function of the Daubechies scaling function is not a wavelet, due to its non-zero mean; however, the present algorithm can be used with any suitable function which has local support in both physical and wavenumber spaces. In order not to cloud the discussion we will refer to these latter functions as wavelets, keeping in mind the difference.

Since every wavelet is characterized by the location b^j_k, then for internal wavelets these locations seem to be the most natural choice for collocation points, while for external wavelets the collocation points can be chosen differently. In the present method we locate them inside of the domain of interest. One may speculate on the best choice of placement strategy for collocation points. However, we emphasize once again that our goal is to present the general philosophy of the method which works for typical wavelets. Note that for a different choice of wavelets the placement strategy may be different, since one may exploit the specific properties of the wavelet. Let us describe the placement strategy which we adopt in this work. The collocation points for the external wavelets corresponding to the finest scale are taken to be distributed uniformly in the boundary intervals $[x_l, x_l + b_0 a_j]$ and $[x_r - b_0 a_j, x_r]$. The collocation points for $j < J$ are distributed in intervals $[x_l, x_l + b_0 a_j]$ and $[x_r - b_0 a_j, x_r]$; they are taken to be the collocation points of the higher levels of resolution. This is a very simple placement strategy. First we use the collocation points of the internal wavelets of the $j + 1$ level of resolution which belong to $[x_l, x_l + b_0 a_j]$ and $[x_r - b_0 a_j, x_r]$, then if all the collocation points of internal wavelets of the $j + 1$ level are used, we proceed to the $j + 2$ level and so on. When the procedure reaches the J level of resolution and we need additional collocation points, then we use the collocation points associated with the external wavelets of the J level of resolution. The placement strategy is illustrated in Fig. 1 for $N = 2$.

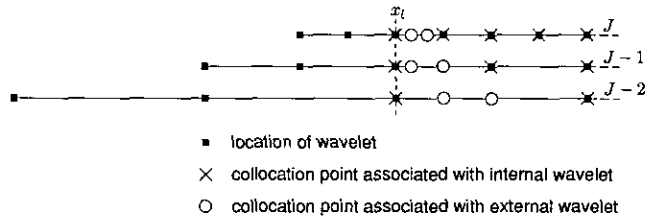

FIG. 1. The location of collocation points near x_l for $N = 2$.

It is important for the algorithm to be able to approximate functions which have non-zero mean, since the method in general may utilize wavelets whose mean is zero. Another required feature is to be able to resolve all the scales present in the approximated function. To test these features we choose to approximate the function

$$u(x) = \tfrac{1}{2}(x^2 - 1) - \sin(\pi x) + 2^{-J_s+3} \sin(2^{J_s-3}\pi x) + 2^{-J_s+2} \sin(2^{J_s-2}\pi x), \tag{29}$$

which has multiple scales as well as non-zero mean. For the test case x_l and x_r are taken to be -1 and 1, respectively.

Let us first discuss how the choice of N affects the approximation. Figure 2 shows the interpolation error for $J_s = 5$ using the Gaussian wavelet and $L = 1$, $J = 5$, $b_0 = 1.0$, $N = 0$. It can be seen that the error is not uniformly distributed within the interval. The largest error occurs close to the ends. Adding external wavelets decreases considerably the overall error of the interpolation as shown in Fig. 3. However, we cannot increase N without limit. By adding more wavelets outside of the domain whose support does not intersect with the domain of interest we make the matrix A^{jj}_{ik} ill-conditioned. In fact the condition number of the matrix increases with the increase of N, making the matrix inversion difficult and inaccurate. Thus for a specific

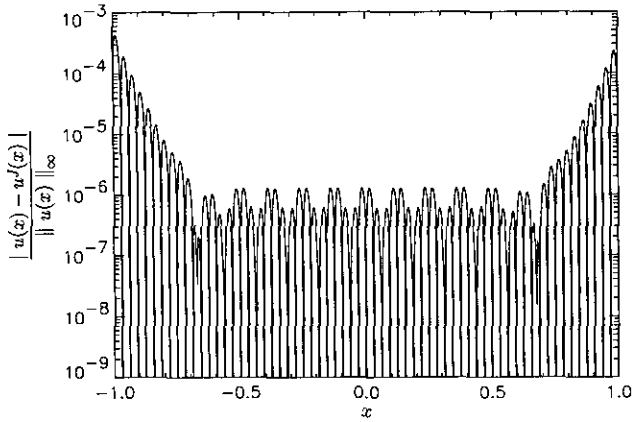

FIG. 2. Relative error of the interpolated function (29) with $J_s = 5$ using the Gaussian wavelet and $L = 1$, $J = 5$, $b_0 = 1.0$, $N = 0$.

choice of wavelet and all other parameters fixed, there is an optimum number of external wavelets. The effect of N on the interpolation error is illustrated in Tables I (cases 1–4) and II (cases 1–3). Since for some choices of the parameters the interpolation error close to the boundary of the domain is higher than in the middle of the domain, in Tables I, II we give the results for the interpolation error of the function and its derivatives not only for the whole domain $x \in [-1, 1]$ but also for the interior interval $x \in [-\frac{1}{2}, \frac{1}{2}]$. Note that for the Gaussian wavelets adding external wavelets makes the interpolation error nearly uniform, while for the case of the correlation function adding external wavelets decreases the error close to the boundary, but it is still higher than in the middle of the domain. In addition, for the same total number of wavelets N_w the interpolation based on the Gaussian wavelet results in a smaller error than using the correlation function (see Tables I, II). Nevertheless one cannot conclude that the Gaussian wavelet is better than the correlation function, simply because there is possibly another choice of collocation points which may work better for the latter.

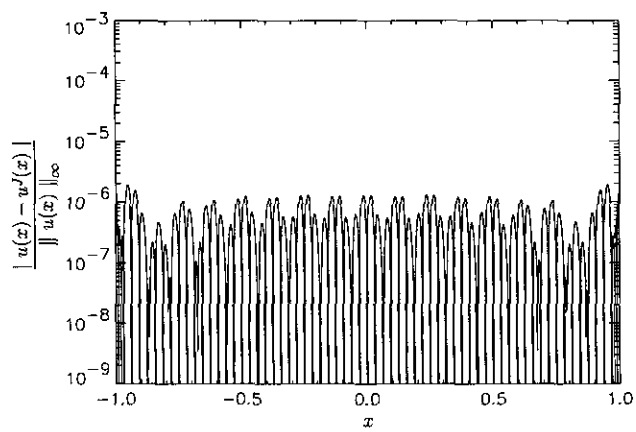

FIG. 3. Relative error of the interpolated function (29) with $J_s = 5$ using the Gaussian wavelet and $L = 1$, $J = 5$, $b_0 = 1.0$, $N = 2$.

As it was mentioned earlier L defines the largest scale present in the approximation, while J determines the finest scale of resolution. Let us discuss how the parameters L and J affect the performance of the wavelet interpolation. From (29) it can be seen that for $J_s \geq 2$ the minimum scale of the function (based on one-fourth the wavelength of the most oscillatory component is 2^{-J_s+1}. The distance between collocation points in the middle of the domain is 2^{-L-J+1}. So one would expect to have a poor approximation if $L + J < J_s$. This is confirmed by Fig. 4 and from the numerical results shown in Tables I (cases 1, 5–8) and II (cases 1, 4–7). These results indicate that the method converges uniformly with the increase of J. Unfortunately we are not able to prove this observation rigorously. Numerical results indicate that the interpolation error converges spectrally in the interior of the domain and in the whole domain for the appropriately chosen N. The convergence rate is affected by the choice of wavelets and other parameters. It should be noted that for small N the interpolation error converges algebraically. This is the reason why the choice of N is very important. Note that the interpolation functions on Fig. 4 for $J = 1$ and $J = 2$ look very similar, yet they are different. This visual similarity occurs due to the absence of the scale corresponding to $J = 2$ in the function (29).

The effect of the multilevel approach is illustrated in Tables I (cases 1, 9, 10) and II (cases 1, 8, 9). For the same number of collocation points ($N_w = 2^{L+J} + 2N + 1$), the interpolation error depends on the number of levels of resolution present in the approximation, *i.e.*, $J + 1$. For fixed N_w, regardless of the choice of wavelet, the interpolation error increases with the decrease of J. Note that if $J = 0$ we have a single level approximation as in the case of Bertoluzza et al. [20].

A distinctive feature of the multilevel approach is the possibility of resolving various scales using wavelets of different scales, while in the single level approach all scales must be represented using smaller scale wavelets. In typical applications one can estimate the smallest and the largest scales present in the problem. Based on this information L and J can be chosen appropriately. Note that (19) gives an inexpensive way of calculating the wavelet coefficients c_k^j. Thus in the multilevel approach we can obtain the energy spectrum for different scales present in the approximation, while for a single level approach this information is lost. Even though we did not take advantage of this property in the present multilevel approach, it can be exploited in a fully adaptive algorithm. Also note that in the multilevel approach the error at the boundary decays much faster than with the single level.

The spacing parameter b_0 also affects the wavelet interpolation. The frame becomes too redundant if b_0 is too small, *i.e.*, $b_0 \lesssim 0.4$, so that the matrix A_{ik}^{lj} becomes ill-conditioned and, thus, difficult to invert. At the same time, as is shown in [8], the frame becomes looser with the increase of b_0, so one would expect the overall interpolation error to eventually increase

TABLE I

Maximum Normalized Error of the Interpolated Function and Its Derivatives for the Gaussian Wavelet ($J_s = 5$)

						$x \in [-1, 1]$			$x \in [-1/2, 1/2]$		
Case	N	L	J	N_w	b_0	$\frac{\|u-u^J\|_\infty}{\|u\|_\infty}$	$\frac{\|u'-u^{J'}\|_\infty}{\|u'\|_\infty}$	$\frac{\|u''-u^{J''}\|_\infty}{\|u''\|_\infty}$	$\frac{\|u-u^J\|_\infty}{\|u\|_\infty}$	$\frac{\|u'-u^{J'}\|_\infty}{\|u'\|_\infty}$	$\frac{\|u''-u^{J''}\|_\infty}{\|u''\|_\infty}$
1	2	1	5	69	1.00	1.90×10^{-6}	1.78×10^{-4}	6.13×10^{-3}	1.29×10^{-6}	2.63×10^{-5}	1.96×10^{-4}
2	0	1	5	65	1.00	4.13×10^{-4}	1.30×10^{-2}	7.37×10^{-2}	1.29×10^{-6}	2.62×10^{-5}	1.96×10^{-4}
3	1	1	5	67	1.00	8.61×10^{-5}	7.44×10^{-3}	1.47×10^{-1}	1.27×10^{-6}	2.59×10^{-5}	1.93×10^{-4}
4	3	1	5	71	1.00	2.08×10^{-6}	4.09×10^{-5}	3.39×10^{-4}	2.08×10^{-6}	4.09×10^{-5}	3.08×10^{-4}
5	2	1	3	21	1.00	7.57×10^{-2}	3.49×10^{-1}	8.25×10^{-1}	7.57×10^{-2}	3.49×10^{-1}	6.61×10^{-1}
6	2	1	4	37	1.00	1.08×10^{-4}	8.76×10^{-4}	1.03×10^{-2}	5.16×10^{-5}	6.76×10^{-4}	2.52×10^{-3}
7	2	1	6	133	1.00	7.25×10^{-8}	1.70×10^{-5}	1.17×10^{-3}	4.04×10^{-9}	1.71×10^{-7}	2.60×10^{-6}
8	2	1	7	261	1.00	3.93×10^{-9}	1.80×10^{-6}	2.49×10^{-4}	4.13×10^{-11}	3.35×10^{-9}	1.04×10^{-7}
9	2	4	2	69	1.00	3.06×10^{-6}	3.11×10^{-4}	1.08×10^{-2}	1.31×10^{-6}	2.65×10^{-5}	1.98×10^{-4}
10	2	6	0	69	1.00	2.17×10^{-4}	2.09×10^{-2}	7.09×10^{-1}	8.77×10^{-5}	4.55×10^{-4}	2.48×10^{-2}
11	2	1	5	69	0.50	3.75×10^{-6}	1.99×10^{-4}	6.77×10^{-3}	1.95×10^{-7}	3.49×10^{-6}	2.79×10^{-5}
12	2	1	5	69	0.75	5.05×10^{-6}	2.02×10^{-4}	6.77×10^{-3}	1.29×10^{-8}	2.08×10^{-7}	1.83×10^{-6}
13	2	1	5	69	1.25	4.29×10^{-3}	1.08×10^{-1}	4.17×10^{-0}	2.76×10^{-3}	4.03×10^{-2}	5.17×10^{-1}

when b_0 becomes too large. The results of varying b_0 are shown in Table I (cases 1, 11–13). The upper cutoff for b_0 can be understood by focusing on the interaction of two wavelets as shown in Fig. 5. One can easily see that with increasing b_0 the depth of a valley, which appears between the two wavelets, increases. This results in the loss of the interpolation. The appearance of the valley between two wavelets can be used as a criteria for the applicability of the wavelets for the present algorithm. Consequently, based on this criteria, as can be seen in Fig. 5 the upper cutoff for b_0 is about 1.5, which is in agreement with the loss of the frame indicated by Daubechies in [8].

Even though we do not present the results here, we note that for single level approximation the results using the Gaussian function ($n = 0$ in (2)) and the correlation function are not so different than those obtained with the multilevel approximation. This is not true for $n > 0$ for the Gaussian family of wavelets. What is more, it can be shown that for $n > 0$ the single level approach cannot be used to approximate a function whose scale is much larger than the scale of the wavelet. Note that this is always the case for a wavelet which has a zero mean. We also observed that for other wavelets of the Gaussian family the interpolation is better for lower order wavelets. Actually it is the best for $n = 0$, which corresponds to the Gaussian function.

TABLE II

Maximum Normalized Error of the Interpolated Function and Its Derivatives for the Correlation Function of the Daubechies Scaling Function ($J_s = 5$)

					$x \in [-1, 1]$			$x \in [-1/2, 1/2]$		
Case	N	L	J	N_w	$\frac{\|u-u^J\|_\infty}{\|u\|_\infty}$	$\frac{\|u'-u^{J'}\|_\infty}{\|u'\|_\infty}$	$\frac{\|u''-u^{J''}\|_\infty}{\|u''\|_\infty}$	$\frac{\|u-u^J\|_\infty}{\|u\|_\infty}$	$\frac{\|u'-u^{J'}\|_\infty}{\|u'\|_\infty}$	$\frac{\|u''-u^{J''}\|_\infty}{\|u''\|_\infty}$
1	1	1	5	67	7.56×10^{-5}	6.39×10^{-3}	1.24×10^{-1}	1.25×10^{-6}	1.19×10^{-5}	3.61×10^{-4}
2	0	1	5	65	8.08×10^{-4}	3.24×10^{-2}	2.89×10^{-1}	1.25×10^{-6}	1.19×10^{-5}	3.61×10^{-4}
3	2	1	5	69	1.28×10^{-4}	3.37×10^{-3}	1.21×10^{-1}	1.25×10^{-6}	1.19×10^{-5}	3.61×10^{-4}
4	1	1	3	19	7.91×10^{-2}	3.65×10^{-1}	1.19×10^{-0}	7.91×10^{-2}	3.65×10^{-1}	6.95×10^{-1}
5	1	1	4	35	1.68×10^{-3}	1.32×10^{-2}	8.57×10^{-2}	5.47×10^{-4}	5.77×10^{-3}	3.57×10^{-2}
6	1	1	6	131	6.79×10^{-6}	7.65×10^{-4}	4.34×10^{-2}	1.49×10^{-9}	5.02×10^{-8}	1.90×10^{-6}
7	1	1	7	259	9.13×10^{-7}	3.57×10^{-4}	3.15×10^{-2}	1.53×10^{-12}	3.64×10^{-8}	7.70×10^{-7}
8	1	4	2	67	8.90×10^{-5}	8.16×10^{-3}	1.72×10^{-1}	1.25×10^{-6}	1.19×10^{-5}	3.61×10^{-4}
9	1	6	0	67	2.40×10^{-4}	3.58×10^{-3}	1.23×10^{-1}	1.25×10^{-6}	1.19×10^{-5}	3.79×10^{-4}

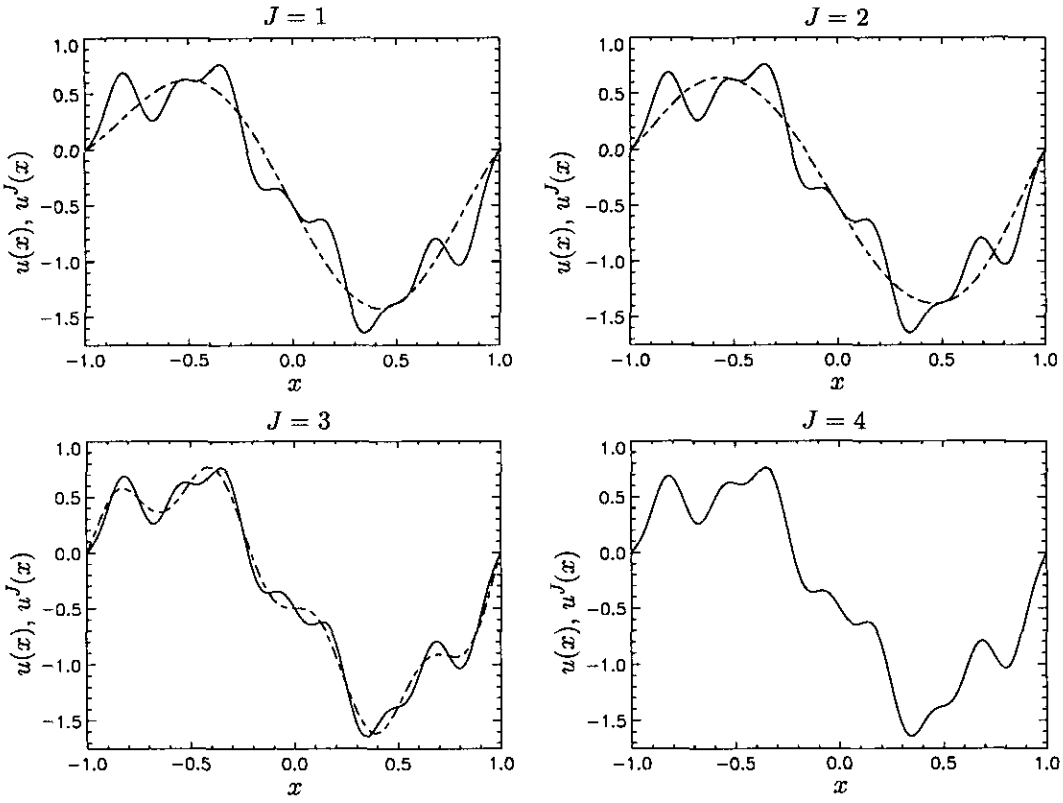

FIG. 4. Function (29) with $J_i = 5$ (———) and its approximation (23) (— — —) using the Gaussian wavelet with different J and $L = 1$, $b_0 = 1.0$, $N = 0$.

This phenomenon can be explained by the fact that the lower order Gaussian wavelets are less oscillatory, which makes them more appropriate to use in the present algorithm. It was also found that in the case of the correlation function of the Daubechies scaling function the interpolation error in the middle of the domain decreases spectrally with increasing order of the function. Nevertheless the error at the boundary is not affected at all.

The application of the wavelet interpolation algorithm to the solution of partial differential equations is described next.

3. APPLICATION TO AN EVOLUTION EQUATION ON A FINITE DOMAIN

As mentioned earlier, the treatment of general boundary conditions on a finite domain is one of the difficulties for most wavelet algorithms. We suggest two different approaches of dealing with boundary conditions. The first is the derivative approach and the second is the integral approach. Since our main interest is in applications to fluid mechanics, we will demonstrate the method through its application to the solution of a second-order partial differential equation of the type

$$\frac{\partial u}{\partial t} = F(t, x, u, u_x, u_{xx}) \quad \text{for } t > 0, u(x, 0) = u_0(x), \tag{30}$$

where F is a linear or nonlinear operator. If the boundary conditions are inhomogeneous, the solution can always be written as a sum of a particular solution which satisfies the inhomogeneous boundary conditions and a complementary solution which satisfies homogeneous boundary conditions. Thus, without loss of generality, we consider the problem with homogeneous boundary conditions. We illustrate the method by solving (30), together with the Dirichlet boundary conditions

$$u(x_l, t) = u(x_r, t) = 0. \tag{31}$$

3.1. *Derivative Approach*

Following the classical collocation approach and evaluating (27), (28) at the collocation points x_i^J of the finest level of resolution we obtain

$$u_i^{J(m)}(t) = \sum_{k \in Z_\Omega^J} D_{i,k}^{(m)} u_k^J(t), \tag{32}$$

$$D_{i,k}^{(m)} = \sum_{j=0}^{J} \sum_{p \in Z_\Omega^j} \psi_p^{j(m)}(x_i^J) \mathbb{C}_{p,k}^{j,J}, \tag{33}$$

where $i \in Z_\Omega^J$ and $\mathbb{C}_{p,k}^{j,J}$ is given by (20). If we number the collocation points is such a way that

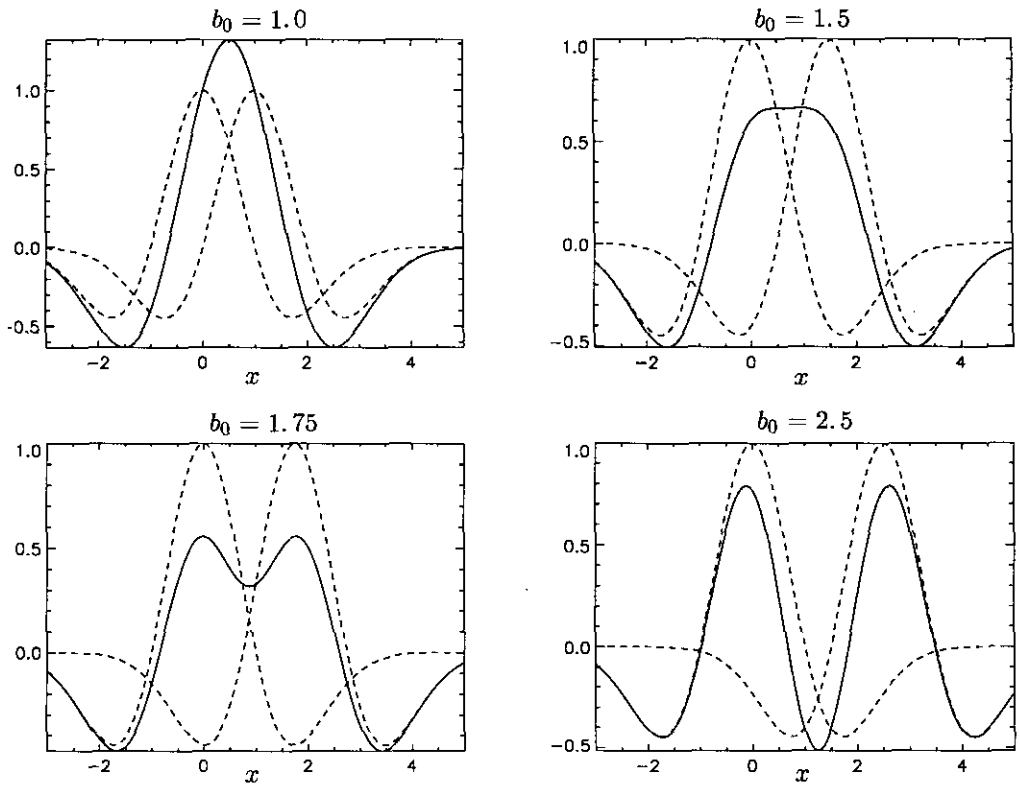

FIG. 5. Gaussian wavelets $\psi(x)$ and $\psi(x - b_0)$ (----) and their sum (———).

$$x^J_{-2^{L+J-1}-N_l} = x_l, \quad x^J_{2^{L+J-1}+N_r} = x_r, \tag{34}$$

then Eq. (30) reduces to a system of $2^{L+J} + N_l + N_r - 1$ nonlinear ordinary differential equations

$$\frac{d}{dt}u^J_i(t) = F(t, x^J_i, u^J_i(t), D^{(1)}_{ik}u^J_k(t), D^{(2)}_{ik}u^J_k(t)),$$
$$u^J_i(0) = u_0(x^J_i), \tag{35}$$

where repeated indices imply summation from $-2^{L+J-1} - N_l$ to $2^{L+J-1} + N_r$ and $i = -2^{L+J-1} - N_l + 1, \ldots, 2^{L+J-1} + N_r - 1$. The boundary conditions (31) become

$$u^J_{-2^{L+J-1}-N_l}(t) = u^J_{2^{L+J-1}+N_r}(t) = 0. \tag{36}$$

After solving (35) with boundary conditions (36), the solution on the interval is given by

$$u^J(x, t) = \sum_{i \in Z^J_\Omega} I_i(x)u^J_i(t). \tag{37}$$

Note that for Neumann or mixed boundary conditions, (36) is replaced by an algebraic relation between u^J_i, $i \in Z^J_\Omega$. Thus one has to solve a differential-algebraic system of equations, which can be rewritten as a system of coupled ordinary differential equations by expressing the values of the function at end points in terms of its values at the interior locations. In addition we note that even though the construction of the discrete derivative operator $D^{(m)}_{ik}$ requires some effort, this is only done once. Subsequently, the right-hand side is obtained by evaluation of matrix products, which can be performed very efficiently on vector computers.

3.2. *Integral Approach*

The essence of the integral method is to approximate the highest derivative appearing in the partial differential equation and then to integrate the approximation while incorporating the boundary conditions using integration constants. Writing the approximation for the second derivative as

$$u^J_{xx}(x, t) = \sum_{j=0}^{J} \sum_{k \in Z^j_\Omega} c^j_k(t)(\psi^j_k)''(x), \tag{38}$$

integrating twice with respect to x, and using the boundary conditions (31) we obtain the expression for $u^J(x, t)$ and $u^J_x(x, t)$,

$$u^J(x, t) = \sum_{j=0}^{J} \sum_{k \in Z^j_\Omega} c^j_k(t)\overline{\psi}^j_k(x), \tag{39}$$

$$u_x^J(x,t) = \sum_{j=0}^{J} \sum_{k \in Z_\Omega^j} c_k^j(t)(\overline{\psi}_k^j)'(x), \tag{40}$$

where

$$\overline{\psi}_k^j(x) = \psi_k^j(x) - \frac{x_r - x}{x_r - x_l}\psi_k^j(x_l) - \frac{x - x_l}{x_r - x_l}\psi_k^j(x_r). \tag{41}$$

Note that $(\overline{\psi}_k^j)''(x) = (\psi_k^j)''(x)$. We now have a set of functions $\overline{\psi}_k^j(x)$ satisfying the boundary conditions exactly. For clarity of discussion we will call these functions the extended wavelets. Note that the extended wavelets are not wavelets in the strict sense of wavelet definition, but if they are located far enough away from the boundaries they approach wavelets, due to the fact that the boundary terms become either numerically negligible (if the wavelets decay exponentially) or zero (if the wavelets have compact support). We have not presented interpolation results using extended wavelets since they are almost identical to those using the normal wavelets. The slight difference is that for the extended wavelets the same accuracy is achieved using one less boundary wavelet than with the normal wavelets. This can be explained by the fact that the extended wavelets satisfy the function at the boundaries exactly.

In this approach the collocation points are taken to be the same as in derivative approach, except that now the wavelets whose locations are on a boundary are considered to be external wavelets. This is necessary since boundary conditions are now satisfied automatically. Following the classical collocation approach and using (27) we can write an expression for derivatives. This expression is exactly the same as (33) except that $\overline{\psi}_k^j(x_i^j)$ is used instead of $\psi_k^j(x_i^j)$. The initial-boundary value problem (30), (31) reduces to the system of $2^{L+J} + N_l + N_r + 1$ nonlinear ordinary differential equations given by (35). After solving (35), the solution on the interval is then again given by (37) with $\psi(x)$ replaced by $\overline{\psi}(x)$ in the definition of $I_i(x)$.

3.3. *Generalization to Higher Dimensions*

It should be noted that both the derivative and integral wavelet collocation approaches can be easily generalized to rectangular two- and three-dimensional domains. Let us illustrate how the method can be applied in the two-dimensional case. We will illustrate the method for the partial differential equation of the type

$$\frac{\partial u}{\partial t} = F(t, x, y, D_x^{(n)} D_y^{(m)} u) \quad \text{for } t > 0,\ u(x, y, 0) = u_0(x, y), \tag{42}$$

where F is a linear or nonlinear operator and n and m denote the order of derivatives. Analogous to the one-dimensional case and without loss of generality, we consider the problem with homogeneous Dirichlet boundary conditions

$$u(x_l, y_l, t) = u(x_l, y_r, t) = u(x_r, y_l, t) = u(x_r, y_r, t) = 0. \tag{43}$$

Then introducing derivative operators in the x and y directions analogous to the one-dimensional case we can write the following system of equations:

$$\begin{aligned} \frac{d}{dt} u_{i,j}^{J,K}(t) &= F(t, x_i^J, y_j^K, D_{xi,k}^{(n)} D_{yj,l}^{(m)} u_{k,l}^{J,K}(t)), \\ u_{i,j}^{J,K}(0) &= u_0(x_i^J, y_j^K), \end{aligned} \tag{44}$$

Boundary conditions in the derivative method can be incorporated analogously to the one-dimensional case, while for the integral approach they are taken care of automatically. We will not elaborate further on the application of the method to higher dimensions here.

4. RESULTS AND DISCUSSION

As a test problem for the numerical algorithm described in the previous section, we will consider Burgers equation

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = \nu \frac{\partial^2 u}{\partial x^2}, \quad x \in (-1, 1), t > 0 \tag{45}$$

with initial and boundary conditions

$$u(x, 0) = -\sin(\pi x), \quad u(\pm 1, t) = 0 \tag{46}$$

whose analytical solution is known (see [23]). For the sake of simplicity we will discuss only the formulation for the derivative approach, since essentially identical results are obtained when either the derivative or integral approaches are used with either wavelet. Also note that the boundary conditions are the same at the two ends and since the wavelets that we utilize are symmetric, we use the same number of external wavelets on each side of the domain, *i.e.*, $N_l = N_r = N$. In light of (35) and (36) the problem reduces to

$$\begin{aligned} \frac{d}{dt} u_i^J(t) &= \sum_{k \in Z_\Omega^J} [-u_i^J(t) D_{i,k}^{(1)} + \nu D_{i,k}^{(2)}]\, u_k^J(t), \\ u_i^J(0) &= -\sin(\pi x_i^J), \\ u_{\pm(2^{L+J-1}+N)}^J(t) &= 0, \end{aligned} \tag{47}$$

where $i = -2^{L+J-1} - N + 1, \ldots, 2^{L+J-1} + N - 1$. The system (47) is solved using a fifth-order Gear implicit integration algorithm implemented in the IMSL routine IVPAG [24] with the fixed integration step $\Delta t = 5 \times 10^{-4}/\pi$.

Basdevant *et al.* [23] presented a comparative study of spectral and finite difference methods for the solution of (45) and (46) and $\nu = 10^{-2}/\pi$. For such a small viscosity, the solution develops into a saw-tooth wave at the origin for $t \gtrsim 1/\pi$.

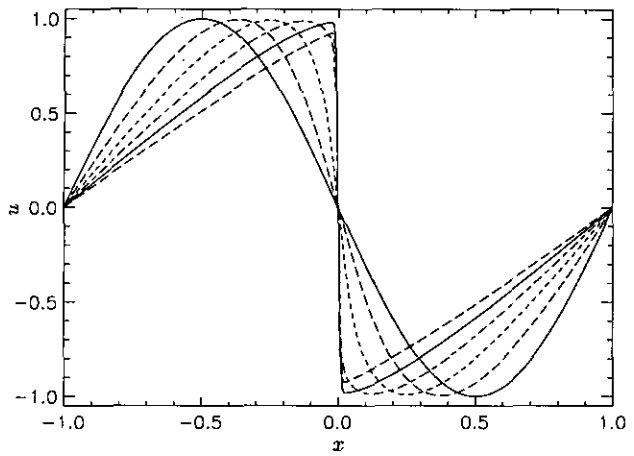

FIG. 6. Analytical solution of the Burgers equation at times $t = 2i/5\pi$, $i = 0(1)5$.

The gradient at the origin reaches its maximum value $|\partial u/\partial x|_{x=0}|_{\max} = 152.0051616$ at time $t_{\max} = 1.60369/\pi$. It appears from the study of Basdevant *et al.* [23] that the performance of a numerical method can be judged from its ability to resolve the large gradient region that develops in the solution, which is shown in Fig. 6.

The results obtained with the wavelet collocation method are given in Tables III and IV. In Table V our results are compared with those obtained using other methods (see [23]) as well as using a wavelet Galerkin approximation (see [15]). The numerical results show that the biggest error occurs in the neighborhood of $x = 0$, where the solution develops a shock. Due to the viscosity, the shock has a finite width. One would expect to resolve the shock properly if the smallest scale wavelets are such as to satisfy $b_0 a_J \leq |\partial u/\partial x|_{x=0}|_{\max}^{-1}$. Since for our particular problem $b_0 a_J = 2^{1-L-J}$, then the shock can be resolved properly for $L + J > 9$. Occurrence of localized oscillations is noted in cases 2–4 in Tables III, IV. Analogous observations were made by Liandrat and Tchamitchian [15] (see Table V): in the non-adaptive case they did not observe oscillations for $J \geq 8$. These oscillations, in contrast to those of Gibbs type observed in spectral methods, are localized in the neighborhood of the shock. This is illustrated in Fig. 7 corresponding to the derivative collocation method using the Gaussian wavelet with $L = 1$, $J = 5$, $N = 2$, $b_0 = 1.0$. (In Fig. 7 linear interpolation is used between collocation points to highlight the oscillations.) For $L + J \geq 8$ the oscillation are visually unnoticeable. The normalized errors of the solution and its derivative for the derivative approach using the Gaussian wavelet with $L = 1$, $J = 8$, $N = 2$, $b_0 = 1.0$ are shown in Fig. 8. It can be easily seen that the largest errors occur in the neighborhood of the large gradient.

Similar to the interpolation results the number of levels of resolution $J + 1$ present in approximation strongly affects the accuracy of the method. This is illustrated in Tables III and IV (cases 1, 5–7).

Since the local error is largest in the neighborhood of the shock one would expect results for this particular problem to be insensitive to N. However, this is not true. The situation is completely analogous to the interpolation example: addition of outside wavelets increases the accuracy of the solution near the boundary (see Tables III and IV (cases 1, 8)).

As it can be easily seen from Tables III, IV the choice of wavelet strongly affects the performance of the method. We can say that for essentially the same number of degrees of freedom, the Gaussian wavelet does better than the correlation function of the Daubechies scaling function in resolving the shock, while their accuracy is practically the same in the region where the solution is smooth.

As with interpolation, the accuracy of the method for the Gaussian wavelet depends strongly on b_0 (see Table III (cases 1, 9–11)). The method becomes more accurate for smaller b_0. Nevertheless with the decrease of b_0 the frame becomes more redundant; eventually the accuracy is completely lost for $b_0 \lesssim 0.4$. In addition the approximation is also lost when b_0 becomes too large, due to the loss of frame. We also note that with the increase of b_0 the approximation of derivatives is lost faster than for the function itself. That is why for case 11 in Table III, while there are no oscillations, the numerical solution is poor in the whole domain.

Even though the results are not shown here, the order of wavelets n affects the accuracy of the method as well. For the Gaussian wavelets with $b_0 = 0.5$ the accuracy of the method is slightly less for $n = 4$ than for $n \leq 2$, while for $b_0 = 1.0$ the method loses the approximation completely for $n = 4$. As far as the implementation of the collocation method for the correlation function of Daubechies scaling function is concerned, the accuracy increases with increasing order of the scaling function. Also note that for the Gaussian wavelets with $n > 0$ the method loses the approximation when a single level approach is used, while with $n = 0$ it does slightly worse when a multilevel approximation is used. The reason for this is discussed in Section 2.

A comparison of eight different methods for the solution of Burgers equation (45) with initial and boundary conditions (46) is made in Table V. The eight algorithms compared are: Fourier Galerkin method, Fourier pseudospectral method, Chebyshev collocation method, spectral element method, finite difference method with coordinate transformation, finite difference method on uniform grid, wavelet Galerkin method, and the present algorithm. Let us briefly comment on some of the above-mentioned algorithms. The Fourier pseudospectral method uses the tau projection method for the linear part and a pseudospectral scheme for the nonlinear term. The spectral element method decomposes the computational domain into properly chosen nonuniform subdomains and expands the unknown in each subdomain as a Lagrangian interpolant using Gauss–Lobatto Chebyshev collocation points. The method to which we refer as a finite difference method on a uniform grid is a method of lines with cubic

TABLE III

Numerical Results Using the Derivative Approach and the Gaussian Wavelet for the Solution of Burgers Equation

Case	N	L	J	N_w	b_0	πt_{max}	Numerical $-\frac{\partial u}{\partial x}(0, t_{max})$	$x_i^J \in [-1, 1]$ $\max\|u - u^J\|$	$x_i^J \in [2/3, 1]$ $\max\|u - u^J\|$	Remarks
1	2	1	8	517	1.00	1.6035	151.07	9.00×10^{-5}	5.74×10^{-7}	No oscillations
2	2	1	5	69	1.00	1.6050	57.03	1.83×10^{-1}	2.85×10^{-4}	Localized oscillations
3	2	1	6	133	1.00	1.6025	97.66	9.25×10^{-2}	6.01×10^{-6}	Localized oscillations
4	2	1	7	261	1.00	1.6010	136.76	1.18×10^{-2}	3.70×10^{-6}	Localized small oscillations
5	2	5	4	517	1.00	1.6035	151.07	9.00×10^{-5}	5.70×10^{-7}	No oscillations
6	2	7	2	517	1.00	1.6035	151.07	8.93×10^{-5}	2.86×10^{-6}	No oscillations
7	2	8	1	517	1.00	1.6035	150.62	2.11×10^{-3}	9.85×10^{-4}	No oscillations
8	0	1	8	513	1.00	1.6035	151.07	9.04×10^{-5}	8.54×10^{-5}	No oscillations
9	2	1	8	517	0.50	1.6035	151.48	4.30×10^{-5}	3.08×10^{-8}	No oscillations
10	2	1	8	517	0.75	1.6035	151.43	6.18×10^{-5}	3.13×10^{-8}	No oscillations
11	2	1	8	517	1.25	1.5780	144.10	3.08×10^{-2}	2.17×10^{-2}	No oscillations

Hermite polynomials which we implemented using the IMSL routine MOLCH [24]. With regard to the accuracy of the solution, Table V shows that for the same number of degrees of freedom, the present wavelet collocation method is competitive with spectral schemes and more accurate than finite difference methods. In comparing with the results obtained using a spline wavelet non-adaptive algorithm [15] we see that we require twice as many wavelets to resolve the shock properly. This is due to the fact that Liandrat and Tchamitchian [15] locate wavelets in a staggered manner, which effectively decrease the number of wavelets required to resolve the shock. This can be done in their Galerkin-type algorithm. However, the requirement in the present algorithm that collocation points at lower levels of resolution be a subset of those at the higher levels makes it impossible to locate wavelets in a staggered manner in our case. Nevertheless, while Liandrat and Tchamitchian provide no details, due to the collocation nature we expect that the present algorithm is more computationally efficient. Most importantly, we note that their method is limited to problems with periodic boundary conditions. This limitation, while acceptable for the present problem, precludes the use of their method in the solution of other problems with more general boundary conditions.

5. CONCLUSIONS

A wavelet collocation method based on the wavelet interpolation technique is developed for the solution of partial differential

TABLE IV

Numerical Results Using the Derivative Approach and the Correlation Function of the Daubechies Scaling Function for the Solution of Burgers Equation

Case	N	L	J	N_w	πt_{max}	Numerical $-\frac{\partial u}{\partial x}(0, t_{max})$	$x_i^J \in [-1, 1]$ $\max\|u - u^J\|$	$x_i^J \in [2/3, 1]$ $\max\|u - u^J\|$	Remarks
1	1	1	8	515	1.6030	149.28	1.72×10^{-3}	6.44×10^{-8}	No oscillations
2	1	1	5	67	1.6670	59.49	3.25×10^{-1}	2.85×10^{-6}	Localized oscillations
3	1	1	6	131	1.6110	97.93	1.59×10^{-1}	5.85×10^{-7}	Localized oscillations
4	1	1	7	259	1.6015	132.95	2.01×10^{-2}	8.74×10^{-8}	Localized small oscillations
5	1	5	4	515	1.6030	149.28	1.72×10^{-3}	3.30×10^{-7}	No oscillations
6	1	7	2	515	1.6030	149.28	1.74×10^{-3}	5.69×10^{-6}	No oscillations
7	1	9	0	515	1.6030	149.59	3.01×10^{-3}	3.04×10^{-4}	No oscillations
8	0	1	8	513	1.6030	149.28	1.72×10^{-3}	1.77×10^{-4}	No oscillations

TABLE V

Comparison of Different Methods for the Solution of Burgers Equation (45) with Initial Condition (46) ($\nu = 10^{-2}/\pi$) (analytical values $t_{max} = 1.60369/\pi$ and $(\partial u/\partial x)(0, t_{max}) = -152.0051616$)

Method	πt_{max}	Numerical $-\frac{\partial u}{\partial x}(0, t_{max})$	Degrees of freedom	Integration step $\pi\Delta t$	Remarks
Fourier Galerkin Spectral [23]	1.6035	151.94	682	5×10^{-4}	No oscillations
	1.60	142.67	682	10^{-2}	Spread oscillations
	1.603	148.98	170	5×10^{-4}	Spread oscillations
	1.60	142.31	170	10^{-2}	Spread oscillations
Fourier Pseudospectral [23]	1.60	142.61	256	10^{-2}	Spread oscillations
	1.60	144.24	128	10^{-2}	Spread oscillations
Chebyshev Collocation [23]	1.60	145.88	512	5×10^{-3}	Spread oscillations
Non-Uniform Spectral Element [23]	1.6033	152.04	64	$10^{-2}/6$	No oscillations
Finite Difference [23] (second order + stretching)	1.63	150.1	81	10^{-2}	No oscillations
Finite Difference (uniform grid)	1.557	61.69	129	5×10^{-4}	No oscillations
	1.593	105.93	257	5×10^{-4}	No oscillations
	1.601	136.07	513	5×10^{-4}	No oscillations
Wavelet Galerkin [15] (periodic boundary conditions, spline wavelet of order $m = 6$, smallest scale $J = 7$)	1.63	135.0	128	10^{-3}	Localized oscillations
($m = 6$, $J = 8$)	1.64	150.3	256	10^{-3}	Localized oscillations
($m = 6$, $J = 8$, adaptive)	1.64	150.3	≤ 104	10^{-3}	No oscillations
Derivative Collocation Method using Mexican hat wavelet $L = 1$, $J = 8$, $b_0 = 0.5$, $N = 2$	1.6035	151.48	517	5×10^{-4}	No oscillations

equations in a finite domain. Two different approaches of treating general boundary conditions are suggested. The method is tested on the one-dimensional Burgers equation with small viscosity. Comparison with other numerical algorithms shows that method is competitive and efficient.

Future areas of further development include the application to two- and three-dimensional rectangular domains and the development of a fully adaptive algorithm. This work is currently under way.

ACKNOWLEDGMENTS

The research reported in this paper has been partially supported by National Science Foundation under Grant No. CTS-9201152 and the Center of Applied Mathematics of the University of Notre Dame.

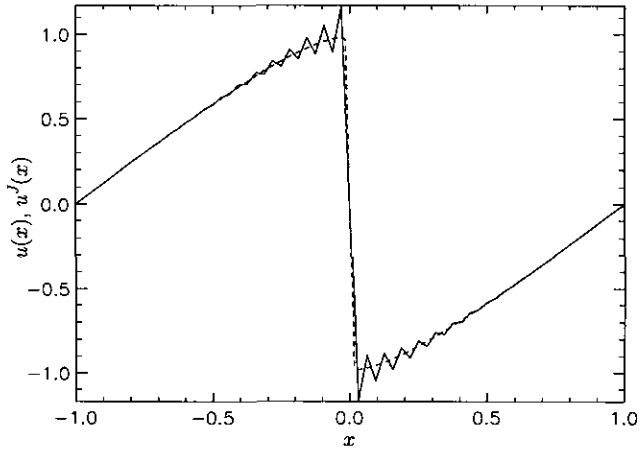

FIG. 7. Comparison of the solution at $t = t_{max}$ using the derivative approach and the Gaussian wavelet (———) with the analytical solution (– – –) for $L = 1$, $J = 5$, $N = 2$, $b_0 = 1.0$.

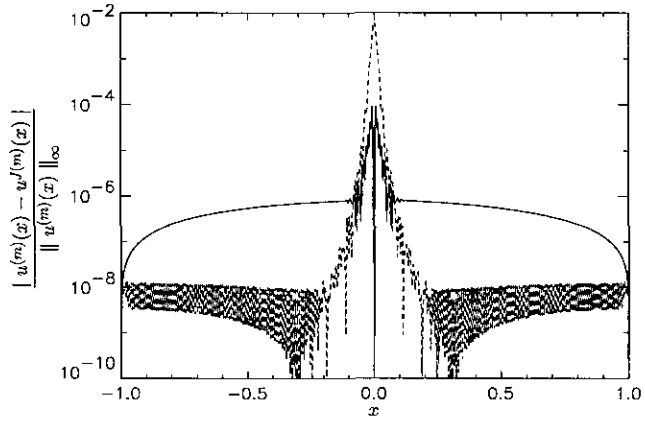

FIG. 8. Normalized error of the solution $u^J(x)$ at $t = t_{max}$ using the derivative collocation approach with the Gaussian wavelet ($m = 0$: ———) and its derivative ($m = 1$: – – –) for $L = 1$, $J = 8$, $N = 2$, $b_0 = 1.0$.

REFERENCES

1. D. Hawken, J. Gottlieb, and J. Hansen, *J. Comput. Phys.* **95,** 254 (1991).
2. D. Hawken, J. Hansen, and J. Gottlieb, *Philos. Trans.* **341,** 373 (1992).
3. J. Dow and I. Stevanson. *Partial Differential Equations* **8,** 537 (1992).
4. M. Berger and J. Oliger, *J. Comput. Phys.* **53,** 484 (1984).
5. T. Tezduyar and J. Liou, *Comput. Methods Appl. Mech. Eng.* **78,** 165 (1990).
6. Y. Meyer, *Ondelettes et opérateurs* (Hermann, Paris, 1990).
7. J. Stromberg, "A Modified Franklin System and Higher-Order Systems of Rn as Unconditional Bases for Hardy Spaces," in *Conference in Harmonic Analysis in Honor of Zygmund,* Vol. 2, edited by W. Beckner *et al.* Math. Series, (Wadsworth, Belmont, CA, 1982), p. 475.
8. I. Daubechies, *Ten Lectures on Wavelets,* CBMS-NSF Ser. Appl. Math. (SIAM, Philadelphia, 1992).
9. G. Beylkin, R. Coifman, and V. Rokhlin, Tech. Rep. YALEU/DCS/RR-696, Yale University, August 1989 (unpublished).
10. A. Latto and E. Tenenbaum, *C.R. Acad. Sci. Paris* **311,** 903 (1990).
11. E. Bacry, S. Mallat, and G. Papanicolau, *Math. Modell. Numer. Anal.* **26,** 793 (1992).
12. R. Schult and H. Wyld, *Phys. Rev. A* **46,** 7953 (1992).
13. S. Qian and J. Weiss, *J. Comput. Phys.* **106,** 155 (1993).
14. B. Jawerth and W. Sweldens, *SIAM Review* **36,** 377 (1994).
15. J. Liandrat and P. Tchamitchian, NASA Contractor Report 187480, ICASE Report 90-83, NASA Langley Research Center, Hampton, VA 23665-5225, 1990 (unpublished).
16. R. Glowinski, W. Lawton, M. Ravechol, and E. Tenenbaum, "Wavelet Solutions of Linear and Nonlinear Elliptic, Parabolic and Hyperbolic Problems in One Space Dimension," in *Computing Methods in Applied Sciences and Engineering,* edited by R. Glowinski and A. Lichnewsky (SIAM, Philadelphia, 1990), p. 55.
17. Y. Meyer, *Rev. Mat. Iberoamericana* **7,** 115 (1992).
18. L. Andersson, N. Hall, B. Jawerth, and G. Peters, "Wavelets on a Closed Subset of the Real Line," in *Recent Advances in Wavelet Analysis* (Academic Press, San Diego, 1993).
19. J.-C. Xu and W.-C. Shann, *Numer. Math.* **63,** 123 (1992).
20. S. Bertoluzza, "Wavelet Methods for the Numerical Solution of Boundary Value Problems on the Interol," in *Wavelets: Theory, Algorithms, and Applications,* edited by C. K. Chui, L. Montefusco, and L. Puccio (Academic Press, San Diego, 1994), p. 425.
21. G. Beylkin and N. Saito, "Wavelets, their Autocorrelation Functions, and Multiresolution Representations of Signals," in *Proceedings of SPIE–The International Society for Optical Engineering,* Vol. LB26 (Int. Soc. for Optical Engineering, Bellingham, WA, 1993), p. 39.
22. C. Chui, *An Introduction to Wavelets* (Academic Press, San Diego, 1992).
23. C. Basdevant, M. Deville, P. Haldenwang, J. M. Lacroix, J. Ouazzani, R. Peyret, P. Orlandi, and A. T. Patera, *Comput. & Fluids* **14,** 23 (1986).
24. *IMSL Mathematical Library,* Verson 1.1, 1989.