INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
*Int. J. Numer. Meth. Fluids* 2006; **52**:749–784
Published online 6 March 2006 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/fld.1202

# Wavelet-based adaptive multiresolution computation of viscous reactive flows

Yevgenii A. Rastigejev[‡] and Samuel Paolucci[*,†]

*Department of Aerospace and Mechanical Engineering, University of Notre Dame, Notre Dame, IN 46556, U.S.A*

SUMMARY

We present a wavelet-based adaptive multiresolution algorithm for the numerical solution of multiscale problems. The main features of the method include fast algorithms for the calculation of wavelet coefficients and approximation of derivatives on nonuniform stencils. The connection between the wavelet order and the size of the stencil is established. The algorithm is based on the mathematically well-established wavelet theory. This allows us to provide error estimates of the solution resulting from the use of an appropriate threshold criteria. The algorithm is applied to a number of test problems as well as to the study of the ignition-delay and subsequent viscous detonation of a $H_2 : O_2 : Ar$ mixture in a shock tube. The simulations show the striking ability of the algorithm to adapt to a solution having different scales at different spatial locations so as to produce accurate results at a relatively low computational cost. The algorithm is compared with classic ENO and TVD schemes. It is shown that the algorithm, besides being significantly more efficient in terms of computational cost, is free from many numerical difficulties associated with those schemes. Copyright © 2006 John Wiley & Sons, Ltd.

KEY WORDS: wavelets; adaptive; multiresolution; multiscale; reacting flows

## 1. INTRODUCTION

Numerical simulations of many physical phenomena, such as the formation and propagation of reactive shock waves in compressible flow, present challenging computational difficulties because their solution, in general, possess a wide range of spatial and temporal scales. A large number of unknowns would be required to resolve the finest spatial scales if one were to use a uniform grid distribution. The solution to this difficulty is to use an adaptive grid strategy.

*Correspondence to: S. Paolucci, Department of Aerospace and Mechanical Engineering, University of Notre Dame, Notre Dame, IN 46556, U.S.A.
†E-mail: paolucci@nd.edu
‡E-mail: ye_rast@yahoo.com

Contract/grant sponsor: NSF
Contract/grant sponsor: AFOSR
Contract/grant sponsor: LANL

*Received 4 May 2004*
*Revised 28 December 2005*
*Accepted 31 December 2005*

The earliest adaptive algorithms have been based on finite volume formulations and are referred to as adaptive mesh refinement (AMR) algorithms [1–6]. These algorithms, while offering impressive advantages in computational speed and storage over uniform-grid algorithms, have a number of shortcomings. First, in general, monitor functions used to adapt the grid are based on *ad hoc* combinations of magnitudes of gradients of physical variables. While such adaption criteria makes sense intuitively, it is nevertheless heuristic in nature. A rigorously based criteria for mesh refinement is not available. As a result one cannot guarantee that the grid distribution is optimal in some sense. Second, for the above reason, it is difficult to estimate errors in AMR algorithms so as to directly control the solution accuracy. While Richardson extrapolation is used for error estimation and mesh refinement [3], as Quirk [5] points out, the use of this criteria is more costly and, while appearing to be more rigorous, yields no more accuracy than ones obtained using heuristic criteria since, in his words 'near flow discontinuities, the very types of flow features we are interested in, Richardson interpolation is invalid.' One should note that Jameson [7] uses wavelet analysis of amplitudes of dependent variables in a standard finite difference adaptive algorithm. Lastly, it is difficult to construct high-order schemes using finite volume formulations. This feature is particularly desirable in multiscale- or multiresolution-type algorithms.

In recent years, algorithms closely related to AMR methods, utilizing multiscale decomposition in the context of finite volume formulations for conservation laws, have been published [8–13]. These methods are referred to as multiscale representation (MR) methods. It should be pointed out that the majority of the works cited above, while they use a multilevel approach, they do not take full advantage of adaptivity. The multiresolution scheme is not used to compute and represent the solution in compressed form. Typically, the solution is represented on the finest grid and adaptive computations are only used for accelerating evolution of numerical fluxes, which are computed exactly on the finest mesh only in regions of largest gradients and are approximated in regions where the solution behaviour is smooth. This results in substantial limitations of potential gains in memory and computational time reductions which fully adaptive algorithms can achieve. Efforts have been made to construct a fully adaptive multiresolution scheme in which the solution is represented and computed on an evolving adaptive grid [11]. MR methods have been used almost exclusively for simulations of a variety of benchmark inert inviscid problems, a notable exception being Bihari and Schwendeman [13] who have used an MR scheme for simulations of a one-dimensional system of reactive Euler equations with possibly stiff source terms.

Recently, efforts have been made in the development of adaptive refinement algorithms using unstructured grids. For recent overviews and bibliography see Mackerle [14], Kita and Kamiya [15], Morgan and Peraire [16], Löhner [17], and Nithiarasu and Zienkiewicz [18]. Four different adaptive procedures have been considered in the finite element community. They are referred to as *r*-, *p*- and *h*-refinement and complete regeneration of a grid by adaptive remeshing. However, as pointed out by Zhu *et al.* [19], and Löhner and Baum [20], only the last two approaches lead to truly efficient and general algorithms, thus explaining their preference in this community. While the use of unstructured grids facilitates application of adaptive algorithms to complex geometries with or without moving boundaries, these methods have similar shortcomings as control-volume-based algorithms. Another apparent difficulty is that grid rearrangement is computationally expensive. This is due to the fact that every adaptive refinement involves not only the analysis of the current solution based on *a posteriori* error estimates to achieve a prescribed accuracy [19], but also a subsequent remeshing procedure

 

which is based on considerable use of search operations and mesh parameters recalculations [21]. As a consequence, it is difficult to rearrange the grid often during the process of numerical integration without degrading the efficiency of the algorithm. Thus, in most of these algorithms, grid rearrangement is only performed relatively few times. It should be pointed out that this expedient can result in an inefficient mesh distribution at any particular time, especially in problems where fast-moving small scales are present. Not surprisingly the majority of works dealing with adaption of unstructured grids are devoted to the numerical analysis of stationary problems. Very few works deal with rapidly developing small structures like shocks and boundary layers [21, 22].

Even more recently, wavelet-based multilevel or multiscale adaptive algorithms have been proposed. These methods are either based on Galerkin (or Petrov–Galerkin) [23–27] or collocation [7, 28–31] procedures. Analogous to boundary-element methods with *h*-refinement, some authors [32] have used wavelets to develop an adaptive Galerkin boundary-wavelet technique. Galerkin wavelet methods, though accurate, have difficulties with the treatment of general boundary conditions, general geometries, and nonlinear terms [33]. To overcome the difficulty associated with nonlinear terms most authors [34, 35] use a collocation procedure to treat these terms analogous to what is done in pseudo-spectral calculations. In this formulation linear terms are treated in wavelet space and nonlinear terms are computed in physical space. Nevertheless, as reported by Schneider *et al*. [34], the resulting adaptive wavelet simulation is still somewhat computationally expensive.

Wavelet collocation methods have been found to be just as accurate and at the same time are mostly free of the difficulties encountered with wavelet-Galerkin methods. A number of wavelet collocation methods have been developed [28–31, 34, 36–41].

Vasilyev and Paolucci, Cai and Wang, and Bertoluzza obtain the derivative approximation through direct differentiation of the wavelet approximation of the function. Unfortunately, this approach can lead to algorithms which are nearly as expensive as wavelet-Galerkin ones. Holmström uses a finite-difference derivative approximation on a uniform stencil of functional values. It can be shown that the latter approach, using a finite difference of an order consistent with the wavelet basis, is much more computationally efficient and equally accurate.

Vasilyev and Bowman [40] also use finite differences for derivative approximations on a nonuniform stencil. Their method is based on the construction of a piecewise polynomial locally approximating the solution. The derivatives are found by differentiating the polynomial. In the present approach we enumerate all possible nonuniform stencil patterns and store the vector of corresponding finite difference coefficients *a priori*. The local point patterns are matched with the stored data during grid rearrangement. Subsequently, the corresponding finite difference coefficients are retrieved and used for derivative calculations. In this paper we also propose an error control based on the Sobolev norm of the sparse wavelet representation (SWR) which is shown to account for the local error more accurately than the more traditional approach based on the wavelet decomposition of the solution.

In addition, some algorithms make use of external wavelets to simplify the application of boundary conditions on finite intervals (Vasilyev and Paolucci). This procedure can lead to much larger errors near boundaries than inside the domain. Lastly, we note that Donoho [42], among others, has considered basis functions which are modified near boundaries. Again, it can be shown that expansions using such basis result in errors, which are more uniform over the domain for low-order basis but become larger for high-order basis near boundaries. Here we suggest a simple procedure to overcome this difficulty. It should be noted that such basis has

 

the remarkable feature that it is associated with the Lagrange interpolation formula [43, 44], which enables fast computation of expansion coefficients of the wavelet approximation of a function.

In this work we present a wavelet-based adaptive multiresolution representation (WAMR) algorithm in one spatial dimension which exploits the above properties. Generalization of the algorithm to higher spatial dimensions will be discussed in a forthcoming article. WAMR is an efficient, accurate, and robust algorithm that appears to be ideal for solving problems which result in a large range of spatial scales and has the following properties. Time integration is accomplished with a linearized trapezoidal scheme. The boundary-modified correlation function of the Daubechies scaling function is used as the basis for spatial approximation. Wavelet coefficients of the approximating functions and their derivatives are obtained by consistent finite differences on a nonuniform stencil and are computed quickly and accurately. In general, it is desirable for the error distribution to be uniform. Errors arise from the SWR, the derivative approximation, and grid adaption. The analysis of these errors provides error estimates which are used for thresholding wavelet coefficients so that a near-uniform error distribution is achieved. As a result, the number of degrees of freedom for a prescribed accuracy is minimized. Lastly, the order of convergence of the method can be easily increased by appropriately using a higher-order basis. Special emphasis is placed on demonstrating the algorithm by solving problems of practical interest. Specifically, we apply the algorithm to the simulation of propagation and reflection of a shock wave in a tube as well as to the simulation of the ignition-delay mechanism resulting in a $H_2 : O_2 : Ar$ mixture.

The outline of the paper is as follows. In Section 2 we describe different elements of the wavelet approximation and adaptive algorithm for solving partial differential equations (PDEs). In Section 3 we provide a detailed discussion of different errors and their connection with criteria for grid rearrangement. Results of numerical simulations of physical problems are discussed in Section 4. Lastly, in Section 5 we summarize the main results of the work.

## 2. WAMR ALGORITHM

Below we describe the essential elements of the implementation of a wavelet-based adaptive algorithm for the solution of PDEs in one spatial dimension. More specifically, we give a brief introduction to wavelets and multiresolution analysis. We then elaborate on the approximation and differentiation of functions based on a SWR. Lastly, we discuss the time integration algorithm and dynamical adaption of the underlying grid.

### *2.1. Wavelets and bases*

The Daubechies autocorrelation functions and boundary-modified Daubechies autocorrelation functions of order $p$ have been used as elements for the construction of basis functions [44]. If $\varphi(x)$ is the Daubechies scaling function of order $p/2$ ($p$ even, see Reference [45]), which implies $p$ coefficients in the dilation equation

$$\varphi(x) = \sum_{i=0}^{p-1} a_i \varphi(2x - i) \tag{1}$$

 

then

$$\phi(x) = \int_{-\infty}^{+\infty} \varphi(y-x)\varphi(y)\,\mathrm{d}y \tag{2}$$

is the Daubechies autocorrelation function of order $p$.

The corresponding wavelet $\tilde{\psi}(x)$ is defined by

$$\tilde{\psi}(x) = 2\phi(2x) - \phi(x) \tag{3}$$

The functions $\phi(x)$ and $\tilde{\psi}(x)$ satisfy the following properties:

$$\int_{-\infty}^{+\infty} x^m \phi(x)\,\mathrm{d}x = \delta_{0m}, \quad \int_{-\infty}^{+\infty} x^m \tilde{\psi}(x)\,\mathrm{d}x = 0, \quad 0 \leqslant m \leqslant p \tag{4}$$

It may be easily shown now that $\phi_{l,k}(x) = \phi(2^l x - k)$ and $\tilde{\psi}_{l,k}(x) = \tilde{\psi}(2^l x - k)$ are elements of multiresolution biorthogonal bases.

Following Donoho [42], we use, instead, the following definition for the wavelet:

$$\psi(x) = \phi(2x - 1) \tag{5}$$

This choice is dictated by the fact that our analysis will be based on a collocation procedure (in which orthogonality of the basis does not play a role), and the computation for finding the wavelet expansion coefficients when using the above wavelet reduces to the efficient use of the Lagrange interpolation formula [43, 44]. Thus, in a different functional space, any function $f(x) \in C_0(R)$, where $C_0$ is the space of continuous functions, may be approximated within a prescribed accuracy (for $L$ large enough) by

$$f(x) \approx f^L(x) \equiv \mathbf{P}^L f(x) = \sum_k f_{0,k}\phi_{0,k}(x) + \sum_{l=0}^{L}\sum_k d_{l,k}\psi_{l,k}(x) \tag{6}$$

The sets of translated scaling functions and wavelets span the spaces $V_l$ and $W_l$, respectively, where

$$\begin{aligned} V_l &= \{\phi(2^l x - k), k \in Z\}, \quad l \in Z \\ W_l &= \{\psi(2^l x - k), k \in Z\}, \quad l \in Z \end{aligned} \tag{7}$$

Below, so as not to introduce additional notation, we will reuse the symbols for spaces $V_l$ and $W_l$, and their corresponding bases $\phi_{k,l}$ and $\psi_{k,l}$ to now apply to a finite interval. Thus, without loss of generality, we consider $f(x) \in C_0([0,1])$. Furthermore, we will overload the symbols $V_l$ and $W_l$ to also denote the sets of collocation points at which the scaling functions and wavelets are located. Specific meanings should be clear from the context. Thus, we define the sets of collocation points on the finite interval by

$$V_l = \{x_{l,j} : x_{l,j} = j/(2^{l+1}p), j = 0, \ldots, 2^{l+1}p\}, \quad l \in Z \tag{8}$$

and

$$W_l = \{x'_{l,k} : x'_{l,k} = (2k+1)/(2^{l+2}p), k = 0, \ldots, 2^{l+1}p - 1\}, \quad l \in Z \tag{9}$$

 

The sets of points are related so that

$$V_{l+1} = V_l \oplus W_l \tag{10}$$

Now, let us consider the approximation of a function on the finite interval. We outline a recursive interpolating subdivision scheme for constructing boundary-modified basis elements as well a fast algorithm for finding corresponding wavelet coefficients of the representation [44] of a function. In order to approximate a function on the interval, it is necessary to modify $p$ scaling functions and wavelets near the left and right boundaries. This suggests that $(2p+1)$ basis elements should be used as a minimum on the $V_0$ level. Then we construct the bases for the subspaces $V_l$, $l=1,\ldots,\infty$ by placing $2^l$-times compressed $p$ left-side boundary scaling functions at the first $p$ collocation points, and $p$ right-side boundary scaling functions at the last $p$ collocation points, and regular scaling functions at the remaining interior collocation points.

Let us introduce the Lagrange polynomials for $r \in [0,\ldots,q]$:

$$L_i^q(r) = \prod_{m=0, m\neq i}^{q} \frac{(r-m)}{(i-m)}, \quad i=0,\ldots,q \tag{11}$$

Now, for a given point $x'_{l,k} \subset W_l$ we introduce the subset of its $p$ nearest points at the coarser level $X_{l,k} \subset V_l$ where, for $k \in W_l$ and $j \in V_l$, we have

$$X_{l,k} \equiv \begin{cases} 0 \leqslant j \leqslant p-1 & \text{for } 0 \leqslant k \leqslant p/2-2 \\ k-p/2+1 \leqslant j \leqslant k+p/2 & \text{for } p/2-1 \leqslant k \leqslant 2^{l+1}p-p/2 \\ 2^{l+1}p-p+1 \leqslant j \leqslant 2^{l+1}p & \text{for } 2^{l+1}p-p/2+1 \leqslant k \leqslant 2^{l+1}p-1 \end{cases} \tag{12}$$

If we denote the values of $f(x)$ on $X_{l,k}$ by $f^j_{l,k}$, then the Lagrange interpolation projection at $x'_{l,k}$ can be written as

$$I_{l,k}(f) = \sum_j A^k_j f^j_{l,k} \tag{13}$$

where $A^k_j$ corresponds to an interpolating coefficient. Since we use Lagrange interpolation on the nearest $p$ points on the interval it follows that

$$A^k_j = \begin{cases} a^k_j, & 0 \leqslant k \leqslant p/2-2 \\ a^{p/2-1}_j, & p/2-1 \leqslant k \leqslant 2^{l+1}p-p/2 \\ a^{2^{l+1}p-k-1}_{p-1-j}, & 2^{l+1}p-p/2+1 \leqslant k \leqslant 2^{l+1}p-1 \end{cases} \tag{14}$$

where

$$a^k_j = L^{p-1}_j(k+1/2), \quad j=0,\ldots,p-1, \quad k=0,\ldots,p-1 \tag{15}$$

We note that the scaling function on $V_0$ has the following property:

$$\phi_{0,n}(x_{0,k}) = \delta_{k,n}, \quad n=0,\ldots,2p \tag{16}$$

 

where the collocation points $x_{0,n}$ for $n=0,\ldots,p-1$ and $n=p+1,\ldots,2p$ are associated, respectively, with the left and right boundary scaling functions and $n=p$ with the standard scaling function (the correlation of the Daubechies scaling function) [42].

The recursive procedure for reconstruction of the $2p+1$ scaling functions can then be expressed as

$$\phi_{0,n}(x'_{l,k}) = I_{l,k}(\phi_{0,n}), \quad l=0,\ldots,\infty, \quad k=0,\ldots,2^{l+1}p-1 \tag{17}$$

For a point on the interval we define the $n$th wavelet on the $W_0$ level of resolution as $\psi_{0,n}(x)=\phi_{0,n}(2x)$, $n=0,\ldots,2p-1$. Analogously, we refer to $\psi_{0,n}(x)$, $n=0,\ldots,p-2$ as left-side boundary wavelets, $\psi_{0,p-1}(x)$ and $\psi_{0,p}(x)$ as regular wavelets, and $\psi_{0,n}(x)$, $n=p+1,\ldots,2p-1$ as right-side boundary wavelets. We construct the subspaces $W_l$, $l=0,\ldots,\infty$ by locating $2^l$-times compressed $n=0,\ldots,p-2$ left-side boundary wavelets at the corresponding first $p-1$ collocation points, the mirror-reflected $p-1$ right-side boundary wavelets at the last $p-1$ points, and regular wavelets at the remaining interior points.

The left four boundary-modified scaling functions and the ordinary scaling function for $p=4$ are shown in Figure 1. The right boundary-modified scaling functions are mirror reflections of the left ones.

The multilevel wavelet bases constructed above utilize compressions and translations of modified Daubechies autocorrelation functions. Elements of these bases have compact support and cardinal (or interpolating) properties. A remarkable feature of these bases is that they enable the construction of a fast algorithm for obtaining wavelet coefficients. Let us briefly outline this algorithm for the wavelet approximation of a function $f(x)\in C_0([0,1])$. Consider the following representation of the function belonging to the $V_L$ subspace:

$$f^L(x) = \sum_k f_{0,k}\phi_{0,k}(x) + \sum_{l=0}^{L}\sum_k d_{l,k}\psi_{l,k}(x) \tag{18}$$

It can be shown [43] that the wavelet coefficients $d_{l,k}$ on $W_l$ can be obtained by the following interpolating subdivision scheme:

$$d_{l,k} = f_{l,k} - I_{l,k}(f), \quad l=0,\ldots,L, \quad k=0,\ldots,2^{l+1}p-1 \tag{19}$$

where $f_{l,k}\equiv f(x'_{l,k})$. In other words, a wavelet coefficient on the $W_l$ level of resolution is obtained by subtracting the Lagrange interpolation of the function (which uses values of the nearest $p$ points of the $V_l$ level of resolution) from the value of the function at that point. It is easy to see that any polynomial up to order $p-1$ can be reproduced exactly on $V_0$. Furthermore, it is also evident that the computation of all wavelet coefficients requires $O(N)$ operations, where $N=2^{L+2}p-2p$ denotes the total number of unknowns. Other subdivision interpolation schemes which possess certain properties have been used for multiresolution analysis. For example, Bihari and Schwendeman [13] use a subdivision interpolation scheme based on cell average representation, which retains the conservative property of the MR ENO algorithm constructed by them. We take the sets of points $V_l$ and $W_l$ as the sets of collocation points for the subspaces $V_l$ and $W_l$, respectively. By choosing the collocation points in this fashion, we establish the regular collocation grid $\mathcal{G}_0$.

 

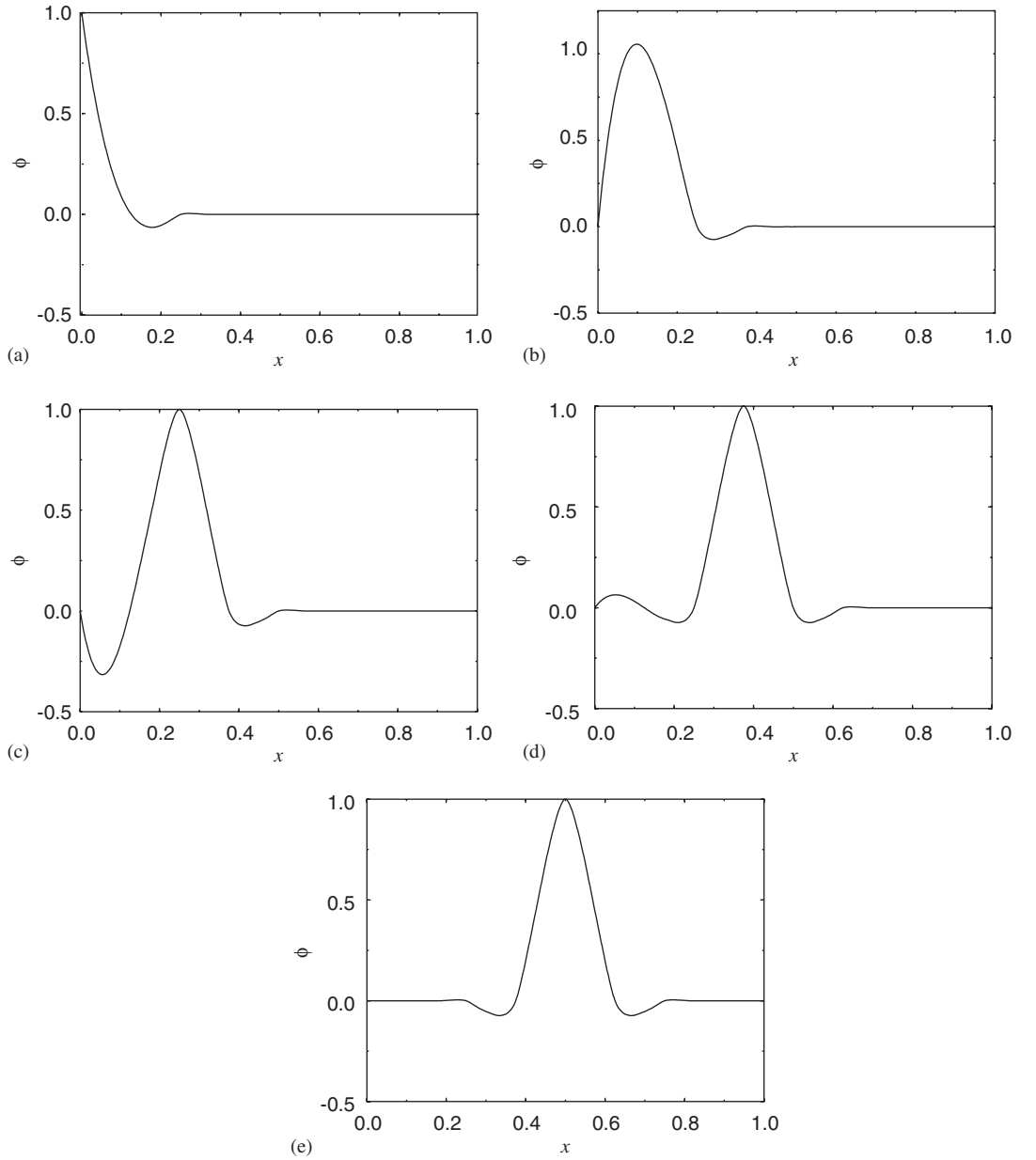


Figure 1. Left boundary scaling functions (a), (b), (c) and (d), and ordinary scaling function (e) for $p = 4$.

 

### 2.2. Sparse wavelet representation and construction of irregular grid

In order to reduce the number of unknowns while retaining the accuracy, we introduce the SWR by thresholding the wavelet coefficients. The approximation can be decomposed into two parts which represent wavelets whose amplitudes are above and below a specified threshold $\varepsilon$

$$f^L(x) = f^L_\varepsilon(x) + R^L_\varepsilon(x) \tag{20}$$

where

$$f^L_\varepsilon(x) \equiv \mathbf{P}^L_\varepsilon f(x) = \sum_k f_{0,k}\phi_{0,k}(x) + \sum_{l=0}^{L} \sum_{k,\,|d_{l,k}|\geqslant\varepsilon} d_{l,k}\psi_{l,k}(x) \tag{21}$$

and

$$R^L_\varepsilon(x) \equiv \sum_{l=0}^{L} \sum_{k,\,|d_{l,k}|<\varepsilon} d_{l,k}\psi_{l,k}(x) \tag{22}$$

As numerical results will later show, the actual error of the approximation is actually smaller than $\varepsilon$ for arbitrary $p$ in the interior of the domain. For $p \geqslant 6$ the error near boundaries becomes increasingly larger than $\varepsilon$ as $p$ is increased. This is due to the fact that the norms of the boundary scaling functions and wavelets are larger than unity and their norms grow with $p$. In order to account for this, we rescale the boundary-modified bases as follows:

$$\phi^+_{0,k}(x) \equiv \phi_{0,k}(x)/\|\phi_{0,k}(x)\|_\infty, \quad \psi^+_{0,k}(x) \equiv \psi_{0,k}(x)/\|\psi_{0,k}(x)\|_\infty \tag{23}$$

The bases at higher level of resolution are then obtained as before. Subsequently, the approximation $f^L_\varepsilon(x)$ is now given by

$$f^L_\varepsilon(x) = \sum_k c_{0,k}\phi^+_{0,k}(x) + \sum_{l=0}^{L} \sum_{k,\,|d^+_{l,k}|\geqslant\varepsilon} d^+_{l,k}\psi^+_{l,k}(x) \tag{24}$$

It can be shown that this renormalization is equivalent to the use of (21) with $\varepsilon_k$ as the thresholding parameter, where

$$\varepsilon_k = \varepsilon/\|\psi_{0,k}\|_\infty \tag{25}$$

and $\|f(x)\|_\infty \equiv \max_{x\in[0,1]} |f(x)|$. To simplify the notation, from now on we will omit the plus superscripts, but it should be understood that our approximation is based on (24). We now call $f^L_\varepsilon(x)$ the SWR of the function $f(x)$, and the corresponding wavelets used in the representation the *essential wavelets*. It is evident that if we omit a wavelet we should also omit the corresponding collocation point, and thus obtain an irregular collocation grid $\mathscr{G}_\varepsilon$. Any point in $\mathscr{G}_\varepsilon$ we call essential and the number of essential points given by $N_e$.

### 2.3. Derivative approximation

In order to solve PDEs we need to approximate derivatives of $f(x)$. Vasilyev and Paolucci [28, 29] approximate values of the derivative by direct differentiation of the SWR of the function

$$f'(x) \approx \sum_k f_{0,k}\phi'_{0,k}(x) + \sum_{l=0}^{L} \sum_{k,\,|d_{l,k}|\geqslant\varepsilon} 2^l d_{l,k}\psi'_{l,k}(x) \tag{26}$$

 

where $\psi'_{l,k}(x) \equiv \mathrm{d}\psi(x')/\mathrm{d}x'|_{x'=2^l x-k}$. The number of operations required for the approximation is $O(pLN)$. We note that while the number of operations increases linearly with the number of points $N$, the coefficient is typically large since it depends on the product of the order of the basis $p$ and the number of levels $L$.

As noted before, since the wavelet approximation for the present basis is related to Lagrange interpolation, we use this fact to construct a more efficient approximation for the derivative of the function. We approximate the derivative of the function at a particular location in the sparse grid by selecting a nonuniform stencil of nearest $m$ points (later we discuss how the number $m$ must be chosen to be consistent with the order of the basis) and applying a finite difference formula on this nonuniform stencil. Formally we can introduce the finite difference operator by

$$f'(x_i) \approx Df(x_i) = \sum_n a_n f(x_{i-n}) \tag{27}$$

where $a_n$ are finite difference coefficients obtained from the requirement that (27) be exact for approximating the Lagrange polynomials of order $m-1$. To compute the $r$th derivative of the function, $f^{(r)}$, to the same accuracy as given by the analogue of (26), we require that $m \geqslant p$. Note that using a stencil size larger than $m = p$ does not increase the order of accuracy of the scheme since the accuracy is controlled by the error associated with grid rearrangement (see error analysis in Section 3.2). Furthermore, in order to construct an efficient and robust algorithm, we impose the restriction that the ratio of largest to smallest distance between neighbouring points in the stencil should be less than or equal to 2. If the selected stencil does not satisfy the restriction, which occurs rarely in practice, then we interpolate the function by using the interpolation subdivision scheme described in Section 2.2. It is shown below that expensive calculations of the coefficients for derivative approximations on nonuniform stencils after every grid rearrangement can be avoided by exploiting the dyadic structure of the nonuniform grid which allows us to store interpolation coefficients *a priori*. The structure of $m$ points can be described by a set of $m+1$ integer values, which we refer to as the stencil structure

$$\mathbf{I}_m \equiv (\mathbf{J}_{m-1}, i_m, i_{m+1}) \tag{28}$$

where $\mathbf{J}_{m-1} = (i_1, i_2, i_3, \ldots, i_{m-1})$ defines the local stencil ordered from left to right, $i_m$ gives the position of the point within the stencil for which the derivative is desired (and can take integer values $1, \ldots, m$), and $i_{m+1}$ provides the local scale given by $2^{-i_{m+1}}$, which corresponds to the smallest distance between two points within the stencil. Since our collocation grid is dyadic, $i_{m+1}$ is a positive integer which can only take values of $0, \ldots, L$, where $L$ is the maximum number of levels. In the definition of $\mathbf{J}_{m-1}, i_j$ represents the ratio of the local distance between two points to the smallest distance between any two points within the stencil; note that $i_j$ can only take values of 1 or 2 due to our previous restriction. Since every element of $\mathbf{J}_{m-1}$ can take only two values, it itself can be represented by an integer written in binary form. Thus, we represent the stencil structure for every point by the three integers $(\mathbf{J}_{m-1}, i_m, i_{m+1})$. We note that the number of different stencil structures that we can have is finite and given by $2^{m-1} \times m \times L$, and thus the storage requirement for these structures involving $m$ coefficients is given by a matrix of $2^{m-1} \times m^2 \times L$ integers. This fact allows us to compute *a priori* all the structures and store them. Subsequently, during computations it is only necessary to identify the local structure at each essential point, and then the derivative approximation becomes very

 

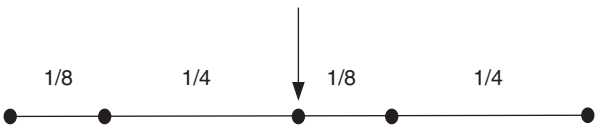


Figure 2. Illustration of a stencil structure.

inexpensive. We note that the storage required for the stencil structures is fairly small. For example, to obtain the derivative, if we use $p=4$ with 16 levels of resolution, then $m=5$ and $L=16$, thus requiring storage for 6400 integers.

The differentiation algorithm can be summarized as follows. After every grid rearrangement we identify the stencil structure $\mathbf{I}_m$ for each point on our sparse grid. Typically, only a small fraction of the points (those near the neighbouring region to be defined later) change their structure after grid rearrangement. Subsequently, by matching the local structure with the ones stored in the stencil structure matrix, we obtain the appropriate coefficients to compute the derivative approximation. To illustrate the algorithm, consider the sample stencil shown in Figure 2. The arrow indicates the collocation point at which the derivative is to be approximated. Numbers correspond to the distances between neighbouring points inside the stencil in terms of a spatial step on the $V_0$ level of resolution. From the above description $\mathbf{J}_4=(1,2,1,2)$, or in a binary form $\mathbf{J}_4=(0,1,0,1)=10$, $i_5=3$ and $i_6=3$, so that $\mathbf{I}_5=(10,3,3)$.

Since the present algorithm is based on a collocation procedure, we point out that spatial discretization results in a numerical scheme which does not enforce local conservation properties. Nevertheless, local conservation properties are satisfied within the error criteria imposed by the numerical method which ensures that errors at all spatial scales are below the imposed threshold. In an upcoming paper, we extend the algorithm for derivative calculations to the multi-dimensional case. It is shown there that the number of possible stencil patterns does not increase substantially. While the number of operations for the multi-dimensional case increases somewhat, the computational cost of the algorithm remain of the order of $pN$ as in the one-dimensional case. A comprehensive description of the multi-dimensional algorithm is beyond the scope of the present paper which is primarily devoted to the one-dimensional algorithm.

### *2.4. Time integration algorithm*

After application of the wavelet discretization in space, a system of PDEs

$$\frac{\partial \mathbf{u}}{\partial t}=\mathbf{F}(\mathbf{u},\mathbf{u}_x,\mathbf{u}_{xx},\ldots) \tag{29}$$

with $\mathbf{u}(t,x)=\{u_i(t,x)\}$ reduces to a system of ordinary differential equations (ODEs):

$$\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t}=\mathbf{F}(t,\mathbf{u}) \tag{30}$$

where now $\mathbf{u}=\{u_{i,\{l,k\}}\}$ is a vector of dependent variables defined on the nonuniform collocation grid.

 

To integrate this system, while a number of more efficient methods could be used, here we use the following 2nd order linearized (implicit) trapezoidal method:

$$\left(\mathsf{I} - \frac{\Delta t}{2}\mathsf{J}^{(n)}\right)\Delta\mathbf{u}^{(n+1)} = \frac{\Delta t}{2}[\mathbf{F}(t^{(n+1)}, \mathbf{u}^{(n)}) + \mathbf{F}(t^{(n)}, \mathbf{u}^{(n)})] \tag{31}$$

where $\Delta\mathbf{u}^{(n+1)} = \mathbf{u}^{(n+1)} - \mathbf{u}^{(n)}$, $\Delta t = t_{n+1} - t_n$, $\mathsf{I}$ is the identity matrix and $\mathsf{J} = \partial\mathbf{F}/\partial\mathbf{u}$ is the Jacobian matrix.

In order to take advantage of the sparseness of matrix $\mathsf{J}$ as well as the fact that $\Delta\mathbf{u}^{(n+1)} = O(\Delta t)$ is small, the system of algebraic equations (31) is solved by using the ILU preconditioned BiCGStab method [46]. For all calculations we used time step values sufficiently small to assure that the error associated with time discretization is much smaller than the error associated with spatial discretization.

### 2.5. *Dynamically adaptive algorithm*

In order to resolve all structures appearing in the solution as time evolves, we adapt the SWR in time. To accomplish this, at any time $t_n$ we need to save the amplitudes of essential wavelets and the amplitudes of wavelets in their *neighbouring region*, whose number we denote by $N_n$, i.e. those wavelets which are near the essential wavelets in location and scale [47]. We treat $\psi_{l',k'}$ as a member of the neighbouring region if there is an essential wavelet $\psi_{l,k}$ such that $(l' - l) \leqslant m$ and $|k' - k| \leqslant n$, where $m$ and $n$ are fixed positive integers. We would like to have $m$ and $n$ as small as possible to reduce the number of unknowns, but large enough so as not to lose the representation, or accuracy, of the solution. It has been found that the choice $m = n = 1$ is sufficient [29] for the class of problems considered here.

Essential and neighbouring wavelets constitute the group of *active wavelets*, whose total number is given by $N_a = N_e + N_n$. The irregular grid $\mathscr{G}^{(n)}$ is composed of collocation points corresponding to all active wavelets at any given time $t_n$.

The numerical algorithm for solving the system of algebraic equations (31) may now be summarized as follows:

1. advance (31) to obtain the approximate solution $\mathbf{u}^{(n+1)}$ on the irregular grid $\mathscr{G}^{(n)}$ by using $\mathbf{u}^{(n)}$ as initial condition;
2. obtain the new grid $\mathscr{G}^{(n+1)}$ based on the magnitudes of wavelet coefficients of $\mathbf{u}^{(n+1)}$ and accounting for the new neighbouring region;
3. if $\mathscr{G}^{(n+1)} = \mathscr{G}^{(n)}$ we increment time and go back to step 1, otherwise compute values of $\mathbf{u}^{(n+1)}$ at the new collocation points in $\mathscr{G}^{(n+1)}$, and then increment time and go back to step 1.

## 3. ERRORS

Estimation of discretization errors is a very important component of any adaptive method. Error estimates can be classified into two broad categories. The first class, *a priori* error estimates, are obtained without using any knowledge of a solution. These types of error estimates are generally very conservative, so they are mostly of theoretical value, although they do provide information that is sometimes useful in design methodologies. Error estimates are

 

given in terms of the unknown exact solution. The second class, *a posteriori* error estimates, are obtained from a computed numerical solution. These error estimates are very useful for grid adaption and solution error control.

*A priori* and *a posteriori* estimates generally are different in form and use different procedures. In wavelet adaptive approximations these two estimates are either identical or nearly so. In the present work we consider a method which is applied to essentially direct numerical simulations of viscous equations. In this approach we resolve all physical scales including viscous shocks. For the case where the problem contains very small spatial scales so that their resolution becomes impractical, we implement an artificial diffusion algorithm within the framework of our method [48, 49]. Recently, several such approaches have been successfully developed within the context of adaptive methods [50, 51]. Below we discuss different aspects of error estimates associated with wavelet collocation approximations and their use in the WAMR algorithm.

### 3.1. SWR

As shown by Donoho [42], for a sufficiently smooth function $f(x)$ (belonging to a Besov space, a Banach space of fractional order) and large enough $L$, we have from (20) that

$$E = \|e(x)\|_\infty \equiv \|f(x) - f_\varepsilon^L(x)\|_\infty = \|R_\varepsilon^L\|_\infty \leqslant \varepsilon(c_1 + c_2 \log_2 \varepsilon) \leqslant c\varepsilon \tag{32}$$

where $e(x)$ is the local error, $E$ is the maximum error, and $c_1, c_2$, and $c$ are $O(1)$ constants which depend on $p$ and $f(x)$. Furthermore, Donoho [42, Theorem 3.8], also proves that the rate of convergence of the multilevel SWR is given by

$$E < CN_c^{-p} \tag{33}$$

Note that if $f(x)$ is a polynomial of degree $d \leqslant p-1$, then we expect $E = 0$. Analogous estimates have been obtained for the more general error norm $\|e(x)\|_{L^q}$ by DeVore [52]. Here, however, we use $q = \infty$. This choise of norm for error control is the most consistent with the collocation approach we take. We estimate local errors, and relate them to derivative approximation errors which are also obtained locally. This allows us in turn to relate the order of the wavelet family to the stencil size on which derivatives are approximated. In addition, the choice of $L^\infty$ allows us to develop a simple criteria for error estimation based on the Sobolev norm of the solution.

Below, using different values of $p$ and $\varepsilon$, and different functions $f(x)$ such that $\|f(x)\|_\infty = 1$, we demonstrate numerically that generally $c \lesssim 1$. One such test function

$$f_1(x) = 0.5(x^2 - 1) - \sin(\pi x) + 0.25\sin(4\pi x) + 0.125\sin(8\pi x) + \exp(-10^4(x - 0.5)^2) \tag{34}$$

chosen to have multiscale character, is illustrated in Figure 3(a). The error distributions of the SWR for $p = 4$ and 10, are shown, respectively, in 3(b) and (c). Figure 3(d) demonstrates the effect of the renormalization (25), where it is clearly evident that the error is smaller than $\varepsilon$ globally. Estimate (33) has also been verified for the following functions:

$$\begin{aligned} f_2(x) &= \sin(2\pi x) \\ f_3(x) &= \sin(2\pi x) + \exp(-10^4(x - 0.5)^2) \\ f_4(x) &= \sin(2\pi x) + \exp(-10^4(x - 0.3)^2) + \exp(-10^5(x - 0.5)^2) \end{aligned} \tag{35}$$

 

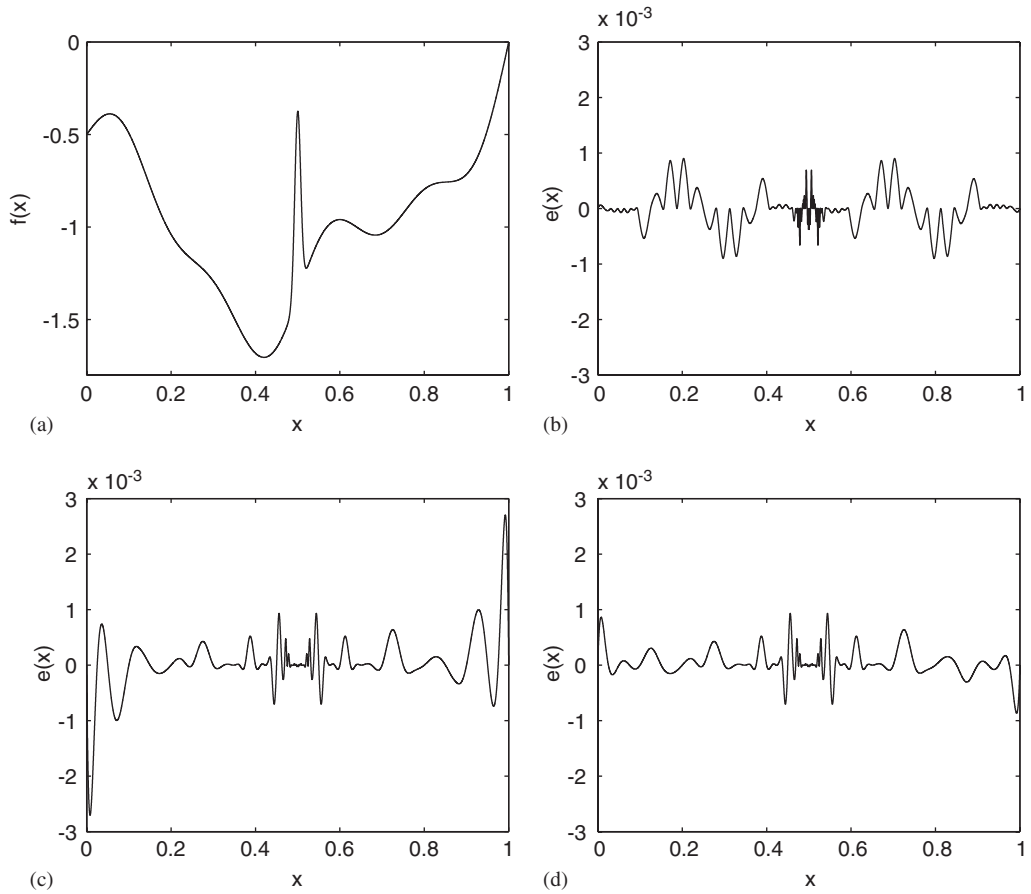


Figure 3. (a) Function $f_1(x)$ and local error distributions $e(x)$ for $\varepsilon = 10^{-3}$; (b) $p = 4$; (c) $p = 10$; and (d) with $p = 10$ and $\varepsilon_k$.

The approximation errors are shown in Figure 4(a) and (b) for $p = 6$ and 10, respectively. Very good agreement between the numerical results and estimate (33) is observed.

### 3.2. Derivative of SWR

As noted before, since our wavelet collocation approximation is connected with a local Lagrange interpolation, using (32) one can establish the following estimate for the local interpolation error at point $x$ based on the use of a nonuniform grid:

$$|e(x)| \approx |R_\varepsilon^L(x)| \approx C_1 |f^{(p)}(x) \Delta x^p| \approx C_1' \varepsilon \tag{36}$$

where $\Delta x \sim 2^{-l}$ is the local spacing of the nonuniform grid and $l$ is the local finest level of resolution. Now, it follows that the local error in the derivative approximation is given by

$$|e'(x)| \approx |[R_\varepsilon^L(x)]'| \approx C_2 |f^{(p)}(x) \Delta x^{p-1}| \approx C_2' \varepsilon / \Delta x \tag{37}$$

 

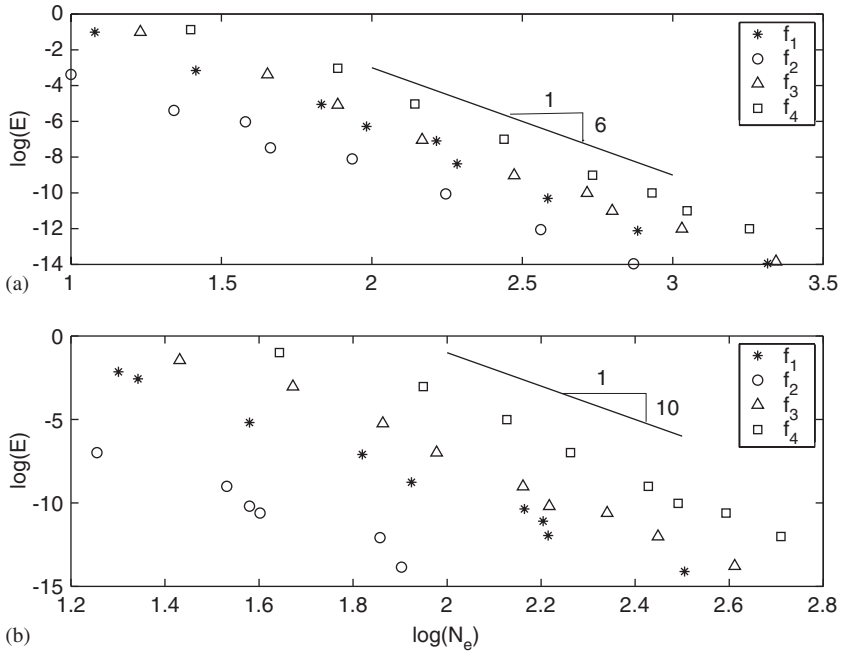


Figure 4. Approximation error $E$ vs number of essential wavelets $N_e$ for functions $f_1$, $f_2$, $f_3$ and $f_4$ (see text) with solid lines given by (33): (a) $p=6$; and (b) $p=10$.

An alternative approach is to approximate the derivative through finite differences on a nonuniform stencil of $m$ points, with the restriction that the ratio of the largest to the smallest distance between points in the stencil should not be greater than 2. As noted earlier, this restriction is violated very infrequently; when it is violated we add an appropriate number of points in the vicinity of the point where the derivative is being approximated such as to satisfy this restriction. This procedure is based on differentiation of the local Lagrange approximation. The local truncation error associated with this derivative approximation at the same point $x$ is given by

$$|e'(x)| \equiv |f'(x) - Df(x)| \approx C_3 |f^{(m)}(x) \Delta x^{m-1}| \tag{38}$$

By taking $m \geqslant p$ we assure that the error in the derivative approximation given by (38) is of the same order as that given by (37). However, the number of operations required to calculate the derivative in this case, say if $m = p$, is $O(pN)$ which is noticeably smaller than the number $O(pLN)$ required to compute (37). For this reason we use the Lagrange interpolation approach to approximate derivatives in this work. This approach is similar to the one suggested by Holmström [38] who, instead, approximates the derivative on a uniform stencil reconstructed from the local nonuniform grid with absent points obtained by interpolation using coarser levels of resolution.

An additional error occurs in the derivative approximation when the collocation grid is rearranged due to adaption. In this case, the error in the approximation at a new points

 

is again given by (36). If a derivative is approximated through finite differences on a stencil which contains at least one new point, the error is approximately given by

$$O(f^{(p)}(x)\Delta x^{p-1}) \sim \varepsilon/\Delta x \tag{39}$$

To retain the approximation accuracy, we choose a stencil size such that the truncation error is of the same order as the error associated with grid rearrangement. This can be easily achieved by taking $m \geqslant p$. It can be easily seen that since the error associated with grid rearrangement is larger than the truncation error for $m > p$, the use of a stencil size with $m > p$ for the derivative approximation will yield the same order of accuracy as one using $m = p$. Numerical experiments show that the present approach typically results in a reduction of computational time for computing derivatives by a factor of 2–5 compared to Holmstrom's procedure, and results in better accuracy. The above factor increases when the number of levels of resolution and the number of fine structures increase.

### 3.3. Adaption

A drawback of algorithms for error estimation and grid rearrangement is that one cannot guarantee the accuracy of the solution *a priori*. The reason comes from the loss of accuracy when derivatives are approximated through finite differences. Moreover, the accuracy in the derivative approximation is worst in regions of large gradients. As a result, a very nonuniform error distribution is obtained over the computational domain. To overcome this difficulty, a different criteria is suggested for the construction of a nonuniform grid in evolution problems. We allow the threshold criteria, and thus the grid distribution, to depend not only on the solution, but also on the right-hand side of Equation (29). In arriving at this new criteria, we assume that the error associated with time discretization is much smaller than the error associated with space discretization.

Consider the system of nonlinear equations

$$\frac{\partial \mathbf{u}}{\partial t} = \mathbf{F}(\mathbf{u}, \mathbf{u}_x, \mathbf{u}_{xx}) \tag{40}$$

where $\mathbf{u}(t,x) \in C^1([0,\infty)) \times C^2([0,1])$, and $C^k$ is the space of functions that are $k$-times differentiable. Assume that the problem has been normalized so that, at any particular time $t$, $\|\mathbf{u}\|_\infty \equiv \max_{i,x} |u_i(x)| = O(1)$. Suppose that three types of terms appear in $\mathbf{F}(\mathbf{u}, \mathbf{u}_x, \mathbf{u}_{xx})$: convection type, $F_i^{\mathrm{c}} = c_j u_j (\partial u_i/\partial x)$, diffusion type, $F_i^{\mathrm{d}} = d^1(\partial^2 u_i/\partial x^2) + d_{ijk}^2(\partial u_j/\partial x)(\partial u_k/\partial x)$ and source type, $F_i^{\mathrm{s}} = S_i(u_j)$. If the local error in the approximation of $u_i(x)$ is $e_i(x)$, then the errors in the first and second derivatives, as shown before, are $O(e_i(x)/\Delta x) = O(2^l e_i(x))$, and $O(e_i x)/\Delta x^2) = O(2^{2l} e_i(x))$, respectively. The errors in the approximation of the right-hand side can now be estimated as follows:

$$E_i^{\mathrm{c}}(x) = O(2^l c_0 e_i(x)), \quad E_i^{\mathrm{d}}(x) = O(2^{2l} d_0 e_i(x)), \quad E_i^{\mathrm{s}}(x) = O(K_0 e_i(x)) \tag{41}$$

where $c_0 \equiv \|\mathbf{c}\|_\infty$, $d_0 \equiv \max(\|\mathbf{d}^1\|_\infty, \|\mathbf{d}^2\|_\infty)$, $K_0 \equiv \max(\|\mathbf{S}(\mathbf{u})\|_\infty, 1)$, and $l$ is the local resolution level.

Previously we considered a function with the thresholding criteria $|d_{l,k}| < \varepsilon$. Suppose that the thresholding for solution vector $\mathbf{u}(x)$ is $\varepsilon_0$. From (32) this assures that the SWR error is

$$E = \|\mathbf{e}\|_\infty = O(\varepsilon_0) \tag{42}$$

 

where $\mathbf{e} \equiv (e_1, \ldots, e_N)$. A better thresholding criteria should provide a similar estimate not only for errors in $\mathbf{u}$, but also for the other terms appearing on the right-hand side of (29). Consider $\mathscr{U} \equiv \{\mathbf{u}, \mathbf{u}_x, \mathbf{u}_{xx}\} \subset W^{2,\infty}$, where $W^{2,\infty}$ is the Sobolev space with corresponding norm

$$\|\mathscr{U}\|_{2,\infty} \equiv \max_{j=0,1,2} a_j \|\partial^j \mathbf{u}/\partial x^j\|_\infty, \quad a_j \geqslant 0 \tag{43}$$

From (41) we have $a_0 = K_0$, $a_1 = c_0$, and $a_2 = d_0$.

The error in the approximation of the right-hand side of system of Equations (40) is then given by

$$\|\mathbf{e}\|_{2,\infty} = E \cdot O(\max(2^l c_0, 2^{2l} d_0, K_0)) = \varepsilon_0 \cdot O(\max(2^l c_0, 2^{2l} d_0, K_0)) \tag{44}$$

If we require that $\|\mathbf{e}\|_{2,\infty} = O(\varepsilon)$, then the thresholding criteria becomes

$$|d_{l,k}| < \varepsilon_0 = \varepsilon / \max(2^l c_0, 2^{2l} d_0, K_0) \tag{45}$$

Numerical verification of this criteria is demonstrated in the next section.

## 4. RESULTS

### 4.1. Model problems

To illustrate essential features of the WAMR algorithm, we first solve three model problems.

(a) In the first example we investigate the propagation of a wave having two very different scales resulting from the solution of the following problem with $\alpha = 10^4$: the function $g(x)$ is used to prescribe both initial and boundary conditions

$$u_t = u_x, \quad t > 0, \quad x < 1 \tag{46}$$

$$u(0, x) = g(x), \quad u(t, 1) = g(t) \tag{47}$$

where

$$g(x) = \sin(2\pi x) + \exp(-\alpha(x - 0.5)^2) \tag{48}$$

The numerical solution at $t = 1$ is shown in Figure 5(a). The solution is obtained using bases with $p = 4$ and nonuniform stencil of size $m = 5$. The local error $e(x)$ at $t = 1$ is illustrated in Figure 5(b), while in Figure 5(c) we show the error for a reconstructed uniform stencil [38]. Figure 5(d) displays the error obtained when the threshold $\varepsilon_0 = \varepsilon/2^l$ resulting from the Sobolev norm, is used. The number of essential points $N_e$ as well as the total number of active points $N_a$ are shown in Figure 6(a) and (b) as functions of time. The maximum number of levels used $(L+1)$ for the solution in Figure 5(b) and (c) is 9, while 10 levels were required for the solution in Figure 5(d).

(b) To examine errors in the derivative approximation we look at the moving shock resulting from (46) and (47) with $\alpha = 2 \times 10^2$ where

$$g(x) = \tanh(\alpha(x - 0.5)) \tag{49}$$

 

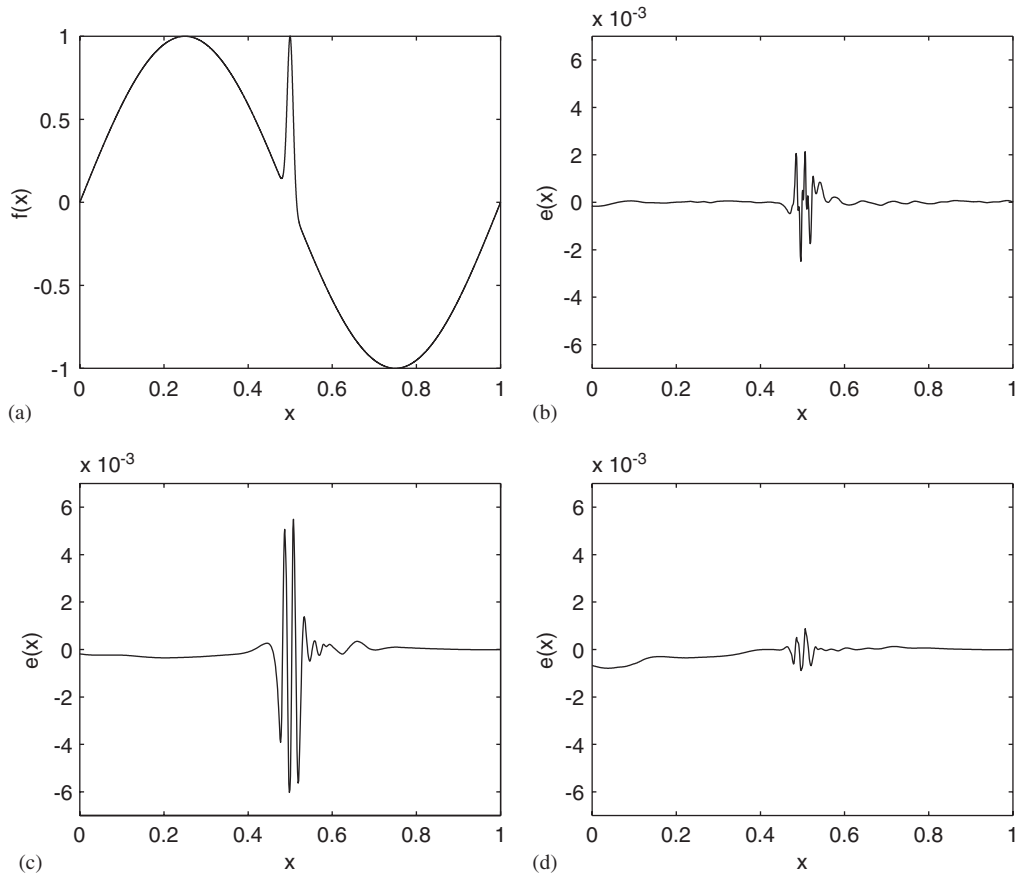


Figure 5. (a) Numerical solution and error distribution $e(x)$ vs $x$ for $t=1$, $p=4$; (b) $\varepsilon=10^{-4}$, derivatives approximated on nonuniform stencil; (c) $\varepsilon=10^{-4}$, derivatives approximated on uniform stencil; and (d) $\varepsilon_0=\varepsilon/2^l$, $\varepsilon=10^{-3}$, derivatives approximated on nonuniform stencil.

The results are presented in Figure 7. It is readily seen that the accuracy of the derivative approximation based on Lagrange interpolation is of the same order as that resulting from the more expensive approximation obtained from (26) [28, 29]. In this case, the maximum number of levels required for the prescribed accuracy is 8.

(c) In the third example we study the formation of a shock arising as the solution of Burgers equation with $\mu=2\times10^{-2}/\pi$:

$$u_t+uu_x=\mu u_{xx},\quad t>0,\quad 0<x<1 \tag{50}$$

and initial and boundary conditions

$$u(0,x)=\sin(2\pi x),\quad u(t,0)=u(t,1)=0 \tag{51}$$

The numerical solution at $t=0.8018/\pi$ (corresponding to the time resulting in the largest gradient at $x=0.5$) is shown in Figure 8(a). The solution is obtained using

 

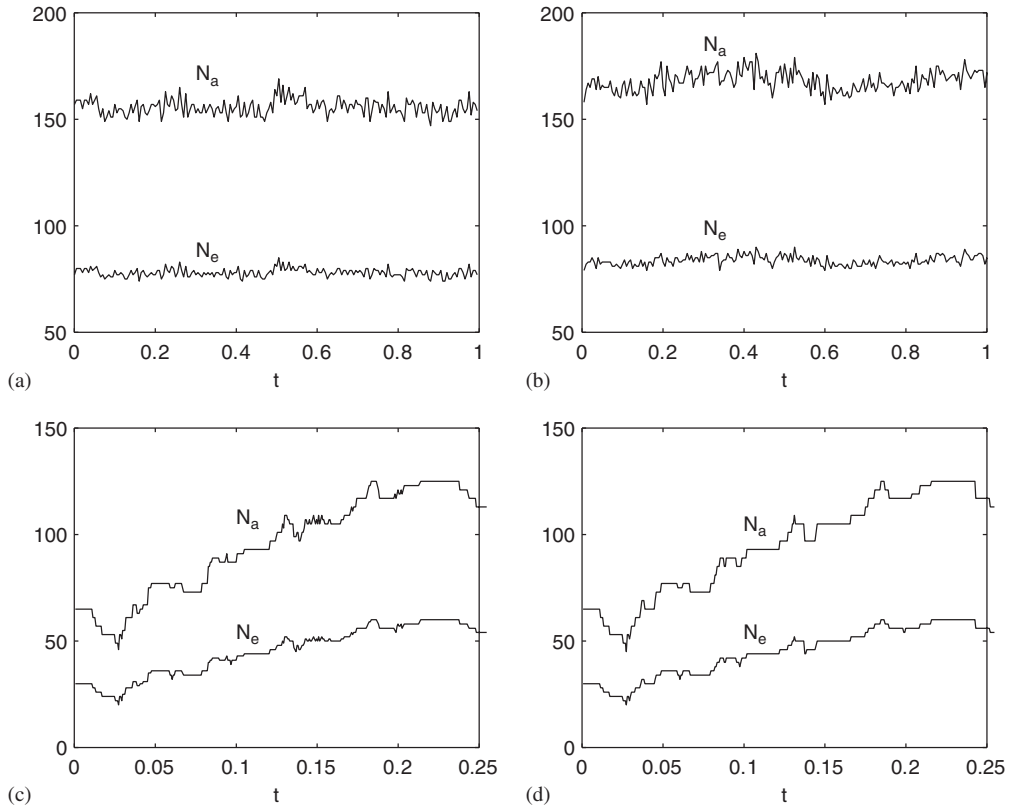


Figure 6. Number of essential points $N_e$ and total number of active points $N_a$ vs $t$ for $p=4$ and $\varepsilon=10^{-4}$. Linear wave propagation problem with derivatives approximated on: (a) uniform stencil; and (b) nonuniform stencil. Developing shock problem with derivatives approximated on: (c) uniform stencil; and (d) nonuniform stencil.

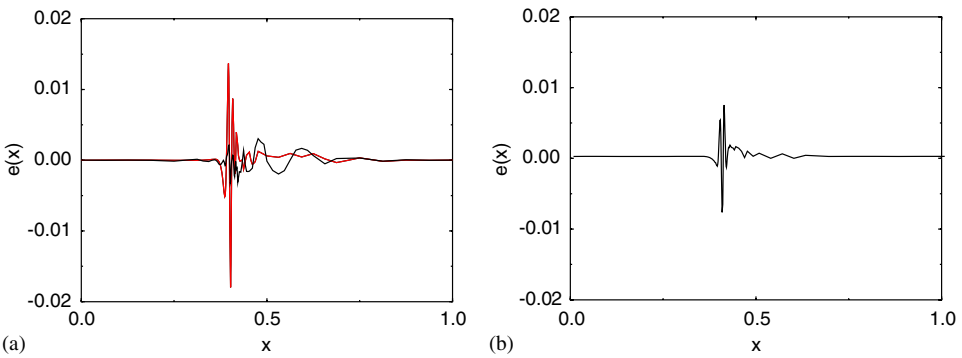


Figure 7. Error distribution $e(x)$ vs $x$ for $p=4$ and $\varepsilon=10^{-3}$ at $t=0.1$ with (a) spatial derivative approximated using the Lagrange procedure: solid line for $m=6$, dashed for $m=4$ and (b) spatial derivative approximated using (26).

 

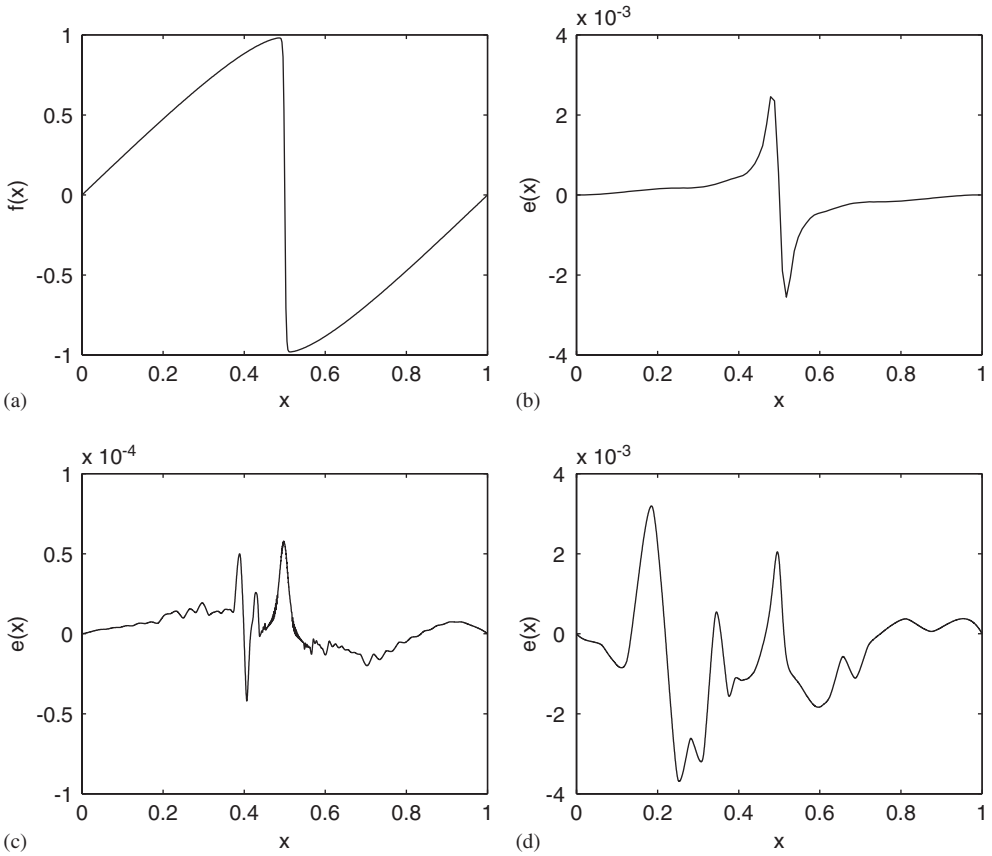


Figure 8. (a) Numerical solution, and error distribution $e(x)$ vs $x$, for $t = 0.8018/\pi$ with $p = 4$ and (b) $\varepsilon = 10^{-4}$, derivatives approximated on uniform stencil, (c) $\varepsilon = 10^{-4}$, derivatives approximated on nonuniform stencil, and (d) $\varepsilon_0 = \varepsilon/\max(2^l, 2^{2l}\mu)$, $\varepsilon = 10^{-2}$, derivatives approximated on nonuniform stencil.

bases with $p = 4$, and nonuniform stencils of size $m = 5$ for the first derivative and 6 for the second derivative. The local error $e(x)$ for $\varepsilon = 10^{-4}$ is illustrated in Figure 8(b) for a reconstructed uniform stencil [38], and in Figure 8(c) for a nonuniform stencil. Figure 8(d) displays the error obtained when the threshold parameter (45) is used with $c_0 = 1$, $d_0 = \mu$, $K_0 = 1$, and $\varepsilon = 10^{-2}$. The number of essential points $N_e$ as well as the total number of active points $N_a$ are shown in Figure 6(c) and (d) as functions of time. The maximum number of levels required to obtain the solution given in Figure 8(b) and (c) is 8, while that in Figure 8(d) is 7.

By comparing results of the above model problems, we conclude that the algorithm utilizing the approximation of derivatives on a nonuniform stencil based on Lagrange interpolation gives practically the same or better accuracy than that using a reconstructed uniform stencil. Furthermore, in all cases the algorithm based on the nonuniform stencil is noticeably faster

 

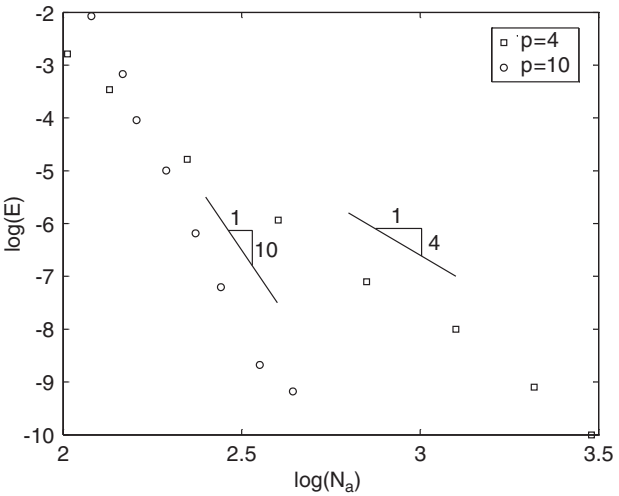


Figure 9. The $L^{\infty}$ error vs the number of active wavelets $N_{\mathrm{a}}$ at time $t = 0.8018/\pi$ for $p = 4$ and $p = 10$ for model problem 4.1(c).

than that based on the reconstructed uniform stencil and substantially faster than that based on the direct differentiation (26).

As seen from Figures 5(d) and 8(d), the use of the error criteria based on the Sobolev norm of the numerical solution results in a more uniform error distribution and guarantees that the solution is accurate to within $\varepsilon_0$. We use model problem (c) to demonstrate the order of convergence of the numerical method. A series of experiments is run where the threshold parameter $\varepsilon$ is successively reduced to take values between $10^{-3}$ and $10^{-11}$. Correspondingly, the accuracy and the number of active points $N_{\mathrm{a}}$ are increased. The dependence of the error on the number of active points is shown in Figure 9 for $p = 4$ and 10. The results demonstrate excellent convergence properties of the method.

### *4.2. Reacting flows in a shock tube*

The shock tube is a widely used experimental tool for studying different physico-chemical processes in gases at high temperature. It is a long tube which is initially separated by a thin diaphragm (or membrane) into two chambers: one at low pressure filled with a test gas, and the other at high pressure filled with a driver gas. At a given time the diaphragm is burst, a shock wave moves through the test gas, and at the same time a rarefaction wave travels through the driver gas. Eventually, the shock reflects from the end wall and the test gas behind the reflected shock comes to rest and is heated to a sufficiently high temperature to ignite it.

We consider $N$ species participating in the following $M$-step reversible chemical reactions:

$$\sum_{i=1}^{N} \nu'_{i,k} \mathscr{A}_i \rightleftarrows \sum_{i=1}^{N} \nu''_{i,k} \mathscr{A}_i \quad (k = 1, \ldots, M) \tag{52}$$

where $\nu'_{i,k}$ and $\nu''_{i,k}$ are the stoichiometric coefficients of chemical species $\mathscr{A}_i$.

 

Assuming that each of the chemical species behaves as an ideal gas, the dimensionless one-dimensional unsteady conservation equations for the chemically reacting flow are

$$\frac{\partial \rho}{\partial t} + \frac{\partial (\rho u)}{\partial x} = 0 \tag{53}$$

$$\rho \frac{\mathrm{D}u}{\mathrm{D}t} = -\frac{1}{\gamma}\frac{\partial p}{\partial x} + \frac{1}{Re}\frac{\partial}{\partial x}\left(\frac{4}{3}\mu \frac{\partial u}{\partial x}\right) \tag{54}$$

$$\rho c_p \frac{\mathrm{D}T}{\mathrm{D}t} = \left(\frac{\gamma - 1}{\gamma}\right)\frac{Dp}{Dt} + \frac{1}{Pr\,Re}\left[-\frac{\partial q}{\partial x} - \frac{1}{Le}\sum_{i=1}^{N} c_{p,i} j_i \frac{\partial T}{\partial x} + (\gamma - 1) Pr \left(\frac{4}{3}\mu \frac{\partial u}{\partial x}\right)\frac{\partial u}{\partial x}\right] + Q \tag{55}$$

$$\rho \frac{\mathrm{D}Y_i}{\mathrm{D}t} = -\frac{1}{Sc\,Re}\frac{\partial j_i}{\partial x} + \omega_i \quad (i = 1, \ldots, N-1) \tag{56}$$

where

$$\frac{\mathrm{D}}{\mathrm{D}t} = \frac{\partial}{\partial t} + u\frac{\partial}{\partial x} \tag{57}$$

In order to close the system we also have the mixture equation of state

$$p = \frac{\rho T}{m} \tag{58}$$

the heat flux

$$q = -k\frac{\partial T}{\partial x} \tag{59}$$

where $k$ is the thermal conductivity, the species flux

$$j_i = \rho \sum_{j=1}^{N} \frac{m_i m_j}{m^2} D_{ij} d_j - D_i^T \frac{\partial \ln T}{\partial x} \tag{60}$$

where

$$d_j = \frac{\partial Y_j^{\star}}{\partial x} + (Y_j^{\star} - Y_j)\frac{\partial \ln p}{\partial x} \tag{61}$$

the enthalpy of each component

$$h_i = h_i^0 + \int_{T_0}^{T} c_{p,i}(T)\,\mathrm{d}T \tag{62}$$

the rate of heat generation

$$Q = -\sum_{i=1}^{N} h_i \omega_i \tag{63}$$

where the rate of change of each component is given by

$$\omega_i = m_i \sum_{k=1}^{M} \nu_{i,k} r_k \tag{64}$$

 

and the reaction rate $r_k$ is

$$r_k = K_{f,k}(T)\left[\prod_{j=1}^{N} n_j^{\nu'_{j,k}} - K_{e,k}^{-1}\prod_{j=1}^{N} n_j^{\nu''_{j,k}}\right] \tag{65}$$

In the reaction rate we assume the Arrehenius model for the forward reaction

$$K_{f,k}(T) = B_k T^{\alpha_k} \mathrm{e}^{-\mathscr{E}_k/T} \tag{66}$$

and we have defined the equilibrium rate constant in terms of the equilibrium molar concentrations:

$$K_{e,k} = \frac{K_{f,k}(T)}{K_{b,k}(T)} = \prod_{j=1}^{N} (n_j)_e^{\nu''_{j,k}-\nu'_{j,k}} \tag{67}$$

In the above equations $T$, $p$, and $u$, are, respectively, the temperature, pressure, and mean velocity of the mixture. The mass concentration $\rho_i$, molar weight $m_i$, molar concentration $n_i$, molar fraction $Y_i^\star$ and mass fraction $Y_i$ of each component as well as the mixture average density $\rho$, the mean molar mass $m$, and the total number of moles $n$ are related by

$$\begin{gathered} \rho_i = m_i n_i, \quad Y_i = \frac{\rho_i}{\rho}, \quad Y_i^\star = \frac{n_i}{n}, \quad Y_i = \frac{m_i}{m} Y_i^\star \\ \sum_{i=1}^{N} Y_i = 1, \quad \rho = \sum_{i=1}^{N} \rho_i, \quad n = \sum_{i=1}^{N} n_i, \quad \rho = mn \end{gathered} \tag{68}$$

The equations have been rendered dimensionless by the use of reference length $L$, reference time $t_\mathrm{r}$, reference speed $u_\mathrm{r}$, reference temperature $T_\mathrm{r}$, and other obvious reference properties evaluated at $T_\mathrm{r}$. In addition, we use the following relations between the reference time and reference speed, $t_\mathrm{r} = L/u_\mathrm{r}$, the reference thermal and species diffusion coefficients, $D_\mathrm{r}^T = \rho_\mathrm{r} D_\mathrm{r}$, between enthalpy and temperature, $h_\mathrm{r} = c_{p,\mathrm{r}} T_\mathrm{r}$, and between the reference pressure and other reference mixture thermodynamic quantities, $p_\mathrm{r} = R_0 \rho_\mathrm{r} T_\mathrm{r}/m_\mathrm{r}$ ($R_0$ is the universal gas constant). Here, the reference length is chosen to be the tube length, the reference speed to be the initial sound speed of the driver gas mixture $c_\mathrm{r}$, the reference temperature to be the initial temperature of the mixtures, and all other properties defined by their initial driver-side mixture values. The resulting dimensionless parameters are, respectively, the ratio of specific heats, the Reynolds number, the Prandtl number, the Schmidt number, the Lewis number, and dimensionless activation energies:

$$\begin{gathered} \gamma = \frac{c_{p,\mathrm{r}}}{c_{v,\mathrm{r}}}, \quad Re = \frac{\rho_\mathrm{r} c_\mathrm{r} L}{\mu_\mathrm{r}}, \quad Pr = \frac{\mu_\mathrm{r} c_{p,\mathrm{r}}}{k_\mathrm{r}} \\ Sc = \frac{\mu_\mathrm{r}}{\rho_\mathrm{r} D_\mathrm{r}}, \quad Le = \frac{Sc}{Pr}, \quad \mathscr{E}_k = \frac{E_k}{R_0 T_\mathrm{r}} \end{gathered} \tag{69}$$

In all calculations that we perform below, for simplicity we use constant viscosity, thermal conductivity, and specific heats, i.e. $\mu = 1$, $k = 1$ and $c_{p,i} = 1$.

The imposed boundary conditions on the tube ends, $x = 0, 1$ are

$$u = 0, \quad q = 0, \quad j_i = 0 \tag{70}$$

 

The above conditions indicate that the ends are rigid, and that the fluxes of heat and mass are zero there. Note that the pressure gradient at the rigid end-walls needs to be consistent with the momentum equation evaluated there.

(a) *Simple reacting flow*. We first consider a simple one-step reaction ($M=1$, and thus drop the subscript $k$) consisting of three components ($N=3$): oxidizer O, fuel F and product P; thus, $i=(1,2,3) \Rightarrow (\mathscr{A}_1=\mathrm{O}, \mathscr{A}_2=\mathrm{F}, \mathscr{A}_3=\mathrm{P})$, and $v_1'=v_{\mathrm{O}}, v_2'=v_{\mathrm{F}}, v_3''=v_{\mathrm{P}}, v_3'=v_1''=v_2''=0$. Furthermore, we consider the solution corresponding to the following initial conditions:

$$\begin{aligned} &u(0,x)=0, \quad T(0,x)=1, \quad \rho(0,x)=2.25f(x) \\ &Y_{\mathrm{O}}(0,x)=0.067, \quad Y_{\mathrm{F}}(0,x)=0.044 \end{aligned} \tag{71}$$

where

$$f(x)=0.5625-0.4375\tanh((x-0.5)/\delta) \tag{72}$$

with $\delta=5\times10^{-3}$. In addition, we choose the following dimensionless parameters and physical properties:

$$\begin{aligned} &\gamma=1.4, \quad Re=2\times10^3, \quad Pr=0.7, \quad Sc=1, \quad \mathscr{E}=1.2, \quad h_{\mathrm{O}}^0=16, \quad h_{\mathrm{F}}^0=0, \quad h_{\mathrm{P}}^0=0 \\ &m_{\mathrm{O}}=1.5, \quad m_{\mathrm{F}}=1, m_{\mathrm{P}}=2.5, \quad v_{\mathrm{O}}=v_{\mathrm{F}}=v_{\mathrm{P}}=1, \quad K_e^{-1}=80, \quad B=30, \quad \alpha=0 \end{aligned} \tag{73}$$

Since we consider premixed conditions, species diffusion is not important, thus we take

$$j_i=0 \tag{74}$$

In order to be able to compare our results with other well-known accurate solutions [53, 54], we also solve the related inert problem. In this case the gas consists of only one component satisfying the following initial conditions:

$$u(0,x)=0, \quad T(0,x)=0.77+0.23f(x), \quad \rho(0,x)=f(x) \tag{75}$$

Let us briefly discuss the evolution of the solution of the inert problem. As a result of bursting of the diaphragm, a shock wave propagates into the low pressure chamber. A contact discontinuity moves right behind the shock but with lower speed. At the same time a rarefaction wave travels in the opposite direction towards the high pressure end. After a period of time, the shock wave reflects from the low-pressure side end-wall and moves in the opposite direction. The gas behind the shock comes to rest, and its temperature increases. Subsequently, the reflected shock wave collides with the incoming contact discontinuity. As a result of this collision, part of the shock is transmitted while another part is reflected and propagates in the same direction as the contact discontinuity, but at a faster speed. This process continues for some time. After each collision, the contact discontinuity slows down, and, in fact, it eventually stops and never reaches the low-pressure side end-wall before its speed is reversed. Figure 10 depicts the solution of the viscous problem obtained by the WAMR algorithm. In this case, since it is not too important and computationally expensive to maintain high accuracy inside shocks (here we are not interested in the shock structure), grid rearrangement is based on the

 

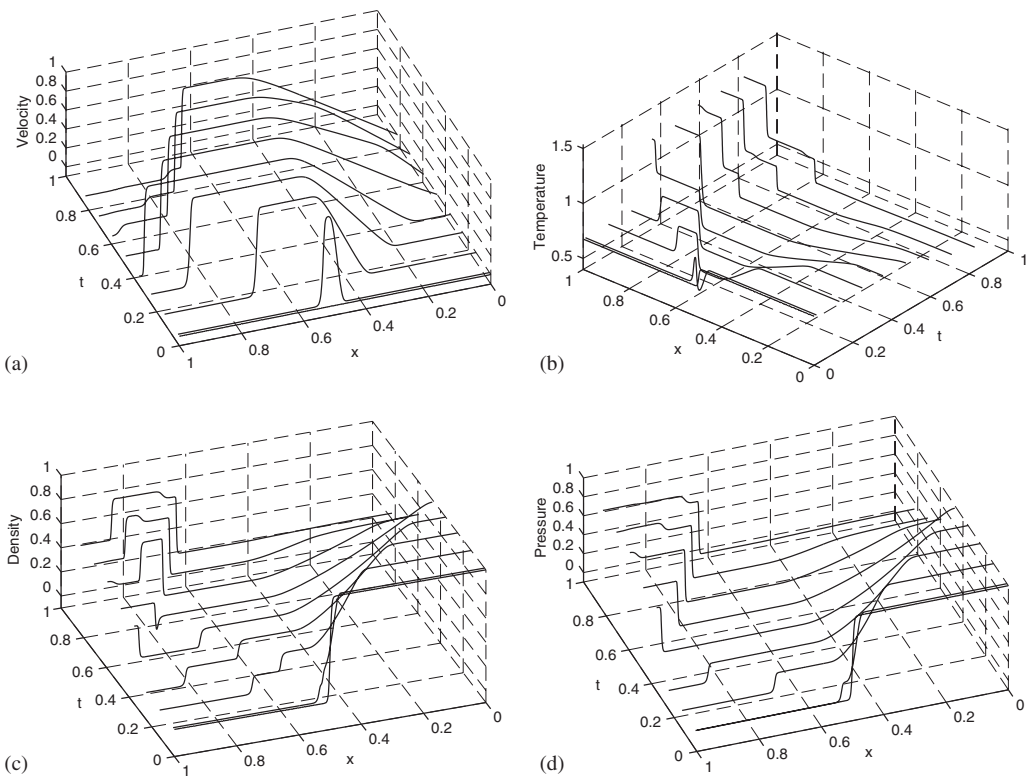


Figure 10. (a) Velocity; (b) temperature; (c) density; and (d) pressure distributions as functions of $x$ and $t$.

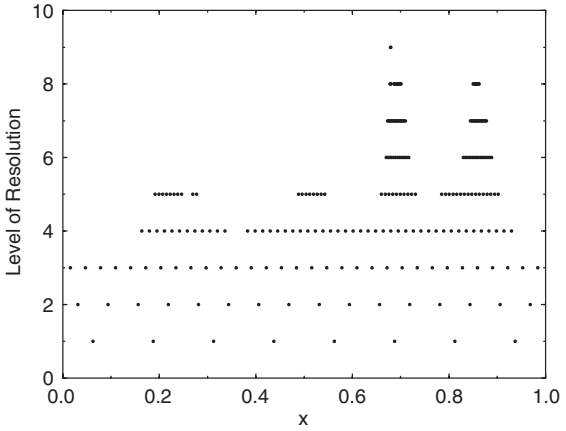


Figure 11. Irregular multilevel grid at time $t = 0.24$.

 

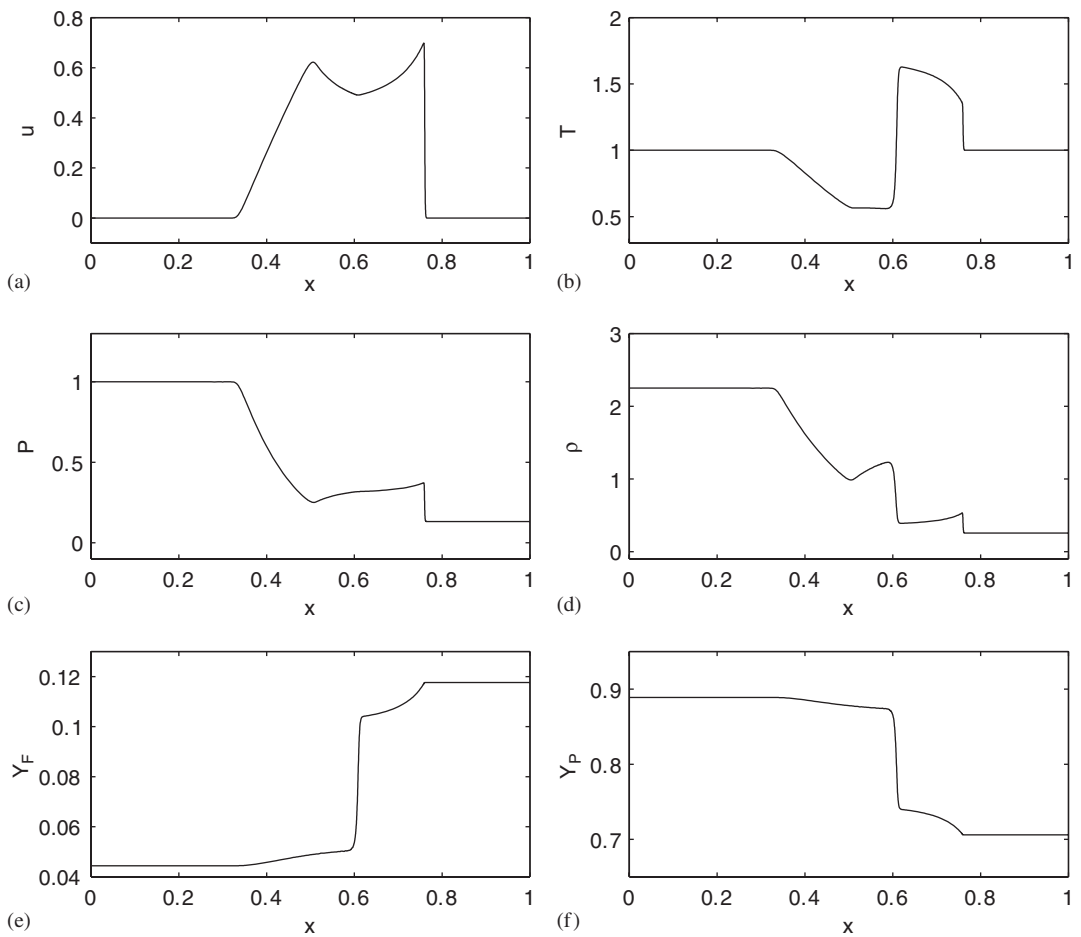


Figure 12. (a) Velocity; (b) temperature; (c) pressure; (d) density; (e) fuel; and (f) product mass fraction distributions vs $x$ at $t = 0.2$.

$L^\infty$ norm of the solution, rather than on the Sobolev norm (45). The threshold parameter used is $\varepsilon = 10^{-4}$. The total number of collocation points required to satisfy this threshold criteria varies between 200 and 280. The irregular multilevel collocation grid (which corresponds to wavelet locations) is shown in Figure 11 for $t = 0.24$. With the evolution of the solution, the multilevel grid adapts to the local structures appearing in the problem. There are four structures present at $t = 0.24$: the shock wave, the contact discontinuity, and the head and tail of the rarefaction wave. Quantitative comparisons show very good agreement between the WAMR results and those given by Sod [54].

When the shock propagates through a premixed gas consisting of oxidizer, fuel and product, the mixture is ignited after the shock reflects from the end wall. We limit our discussion to the examination of the structures existing near the shock wave. The species concentrations are chosen so that the gas is initially at chemical equilibrium.

 

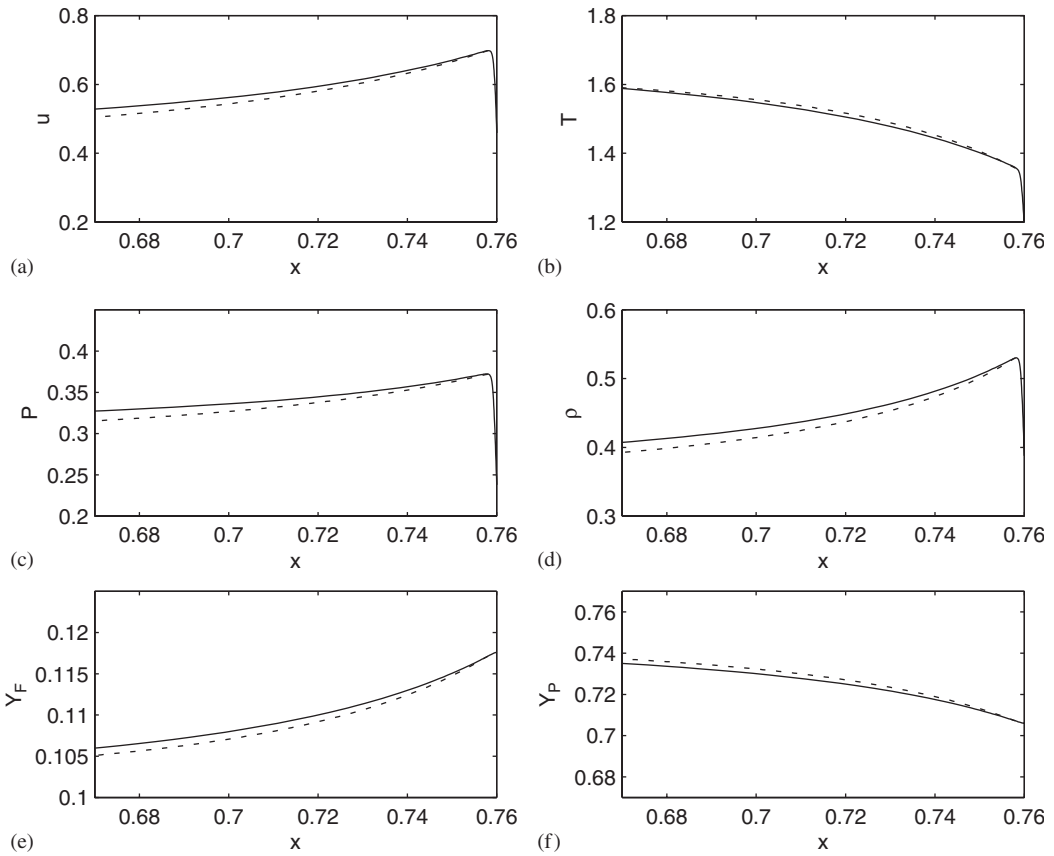


Figure 13. (a) Velocity; (b) temperature; (c) pressure; (d) density; (e) fuel; and (f) product mass fraction distributions vs $x$ at $t = 0.2$. Dashed lines are wavelet results, solid lines are ZND results.

Right next to the shock wave the gas quickly comes to a nonequilibrium state, but relaxes to chemical equilibrium far enough away. The nonequilibrium region, where the relaxation process occurs, is called the reaction zone. All structures present can be clearly seen in Figure 12 which represents the solution at $t = 0.2$. In order to verify the numerical results we also obtained the quasi-analytical profiles immediately behind the shock wave by using the Zeldovich–Von Neuman–Döring (ZND) theory. Comparisons between the wavelet and ZND results are shown in Figure 13. We can clearly see that the agreement between the results is very good considering the approximations made in the ZND theory. The reaction zone can be seen most clearly by examining the fuel mass fraction plot. From the figure we also see that the temperature increases in the reaction zone due primarily to the exothermic reaction. In this problem, the number of collocation points required remains between 200 and 280 for the same value of the threshold parameter.

(b) *Comparison test*. In order to compare WAMR with classical algorithms, we compute the viscous analogue of a previously considered detonation initiation problem [13, 55].

 

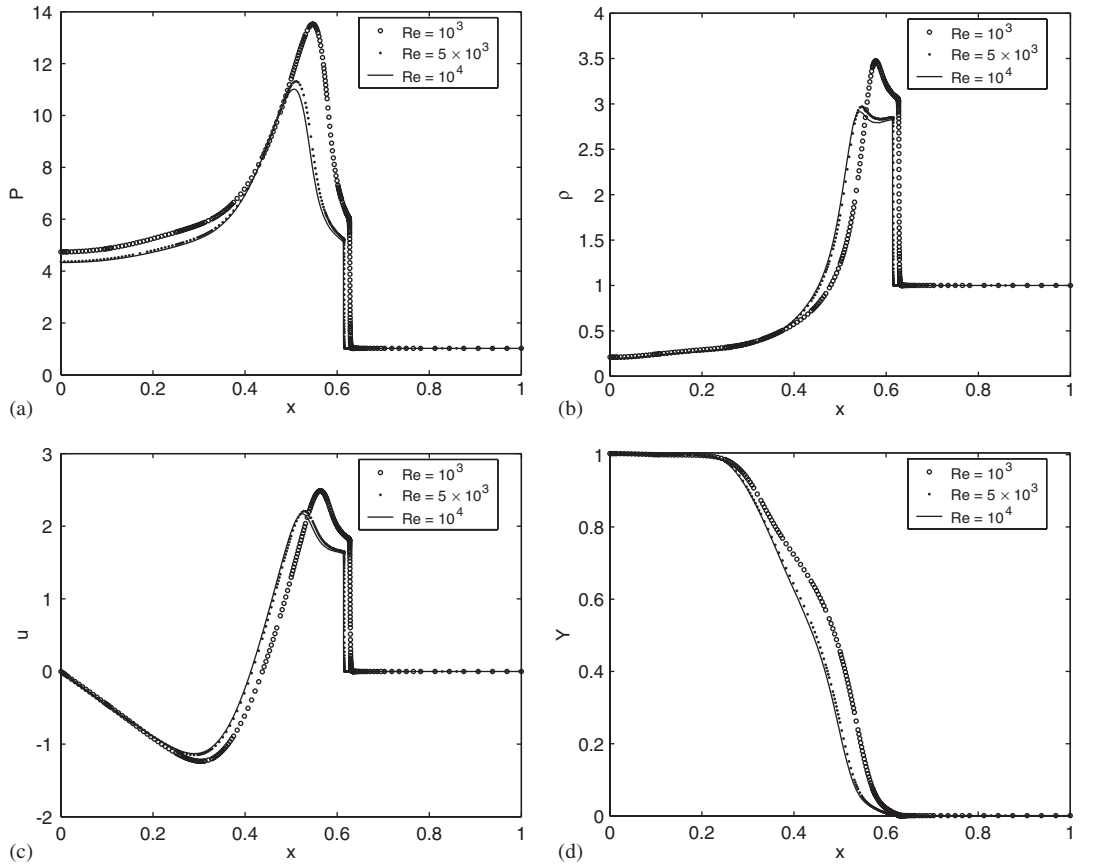


Figure 14. (a) Pressure; (b) density; (c) velocity; and (d) progress variable vs $x$ at $t = 0.25$ for $Re = 10^3$, $5 \times 10^3$, and $10^4$.

The governing equations consist of the reactive compressible Navier–Stokes equations with a single reaction progress variable $Y$, where $Y = 0$ and 1 correspond to unreacted and fully reacted gases. The reaction rate is given by

$$\omega = \rho\kappa(1 - Y)\exp(-E/(p/\rho)) \tag{76}$$

and the heat generation rate by

$$Q = \beta\omega \tag{77}$$

Values of parameters are taken to be the same as those used by Bihari and Schwendeman [13]:

$$\kappa = 84, \quad E = 10, \quad \beta = 50, \quad \gamma = 1.4 \tag{78}$$

 

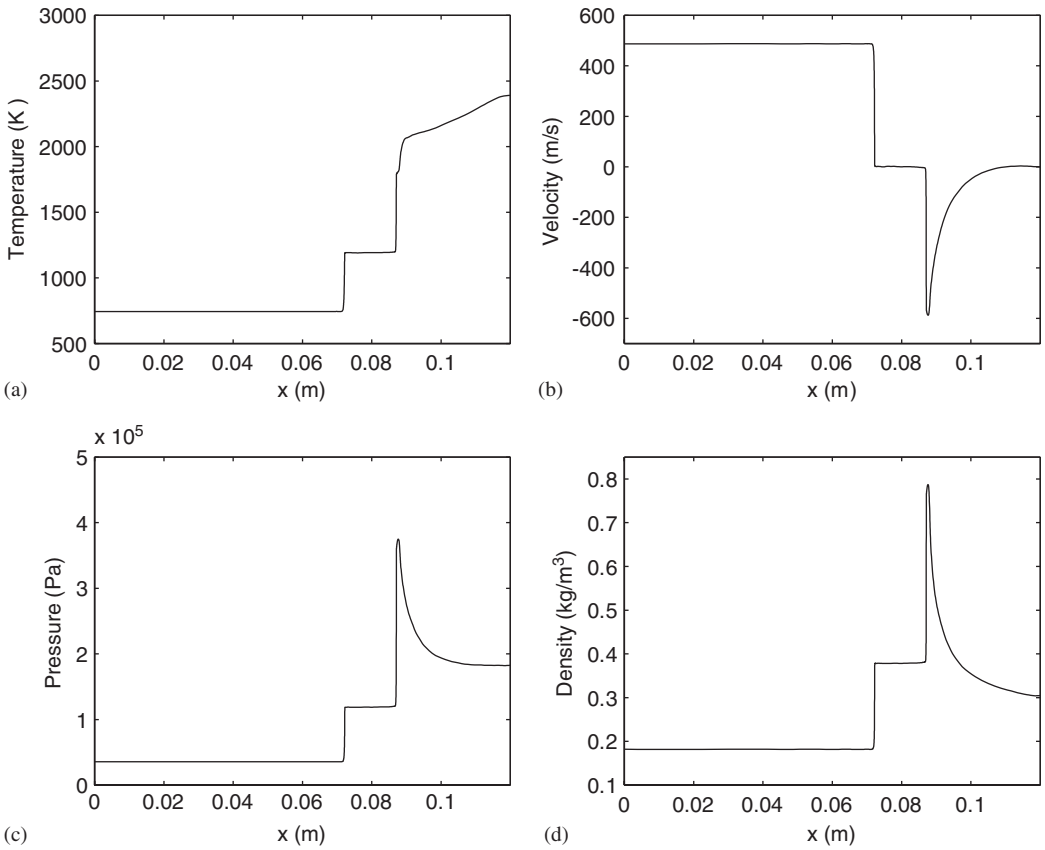


Figure 15. (a) Temperature; (b) velocity; (c) pressure; and (d) density vs $x$ at $t = 180\,\mu s$.

The initial conditions are taken to be

$$p = 1, \quad u = 0, \quad Y = 0, \quad \rho = 1/(1 + 3e^{-12x}) \tag{79}$$

At time $t \approx 0.05$ a shock wave forms with a reaction wave following behind. The heat released from chemical reaction increases the pressure behind the shock wave, forming a smooth bump, which eventually catches up with the shock forming a detonation wave. We perform calculations for a variety of Reynolds numbers in the range $(10^2, 5 \times 10^4)$ and found that the flow dynamics depends very strongly on the viscous dissipation for $Re \lesssim 10^4$. Therefore, it is essential to use Navier–Stokes equations for numerical simulations of shock propagation in relatively low Reynolds number flows. In Figure 14 we present the results of numerical simulations at time $t = 0.25$ for $Re = 10^3, 5 \times 10^3$, and $10^4$. The thresholding parameter used in all calculations is $\varepsilon = 3 \times 10^{-4}$. The maximum number of levels required in the three cases are 10, 12 and 13, respectively. A grid convergence study has been done to show that the numerical viscosity is much smaller than the physical one and does not influence the solution dynamics. Specifically, calculations

 

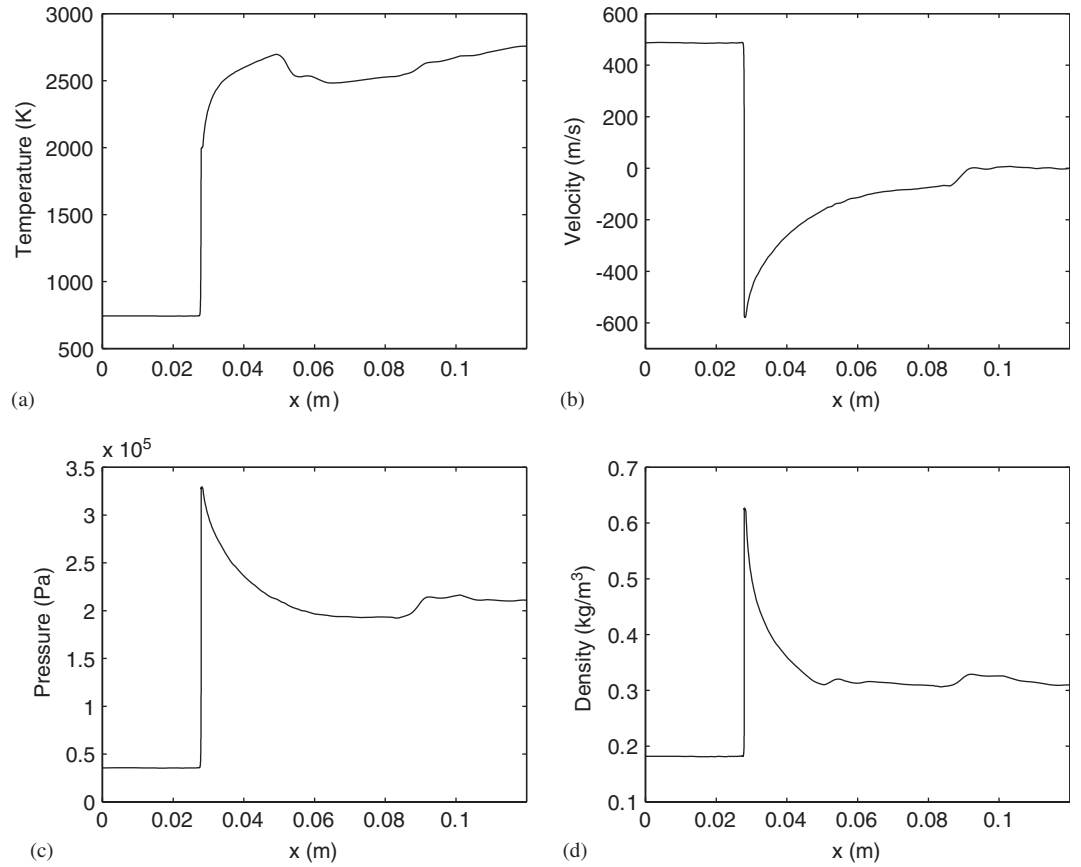


Figure 16. (a) Temperature; (b) velocity; (c) pressure; and (d) density vs $x$ at $t = 230\,\mu s$.

for the much smaller value of $\varepsilon = 6 \times 10^{-5}$ and for three selected values of Reynolds number have been performed. The smaller value of $\varepsilon$ demands a much finer nonuniform grid with at least one more level of resolution, i.e. at least twice the resolution. Obviously, the numerical viscosity is much smaller for this finer grid. Nevertheless, the solutions obtained are virtually indistinguishable from the solutions obtained on the coarser grids with $\varepsilon = 3 \times 10^{-4}$.

In their calculations, Bihari and Schwendeman solve the inviscid problem using both a second-order TVD scheme (TVD2), and a third-order ENO scheme (ENO3). Comparison with their results shows that TVD2 and ENO3 with 200 points, and TVD2 with 2000 points produce similar numerical results to ours for $Re = 10^3$, $5 \times 10^3$ and $10^4$, respectively. This indicates that the effect of artificial numerical viscosity introduced by these schemes is similar to that of physical dissipation in the Navier–Stokes model. Our simulations show that the results for TVD2 and ENO3 with 200 points are very inaccurate due to poor spatial resolution which results in very high numerical viscosity. To produce an accurate numerical solution using inviscid equations, the TVD2 scheme

 

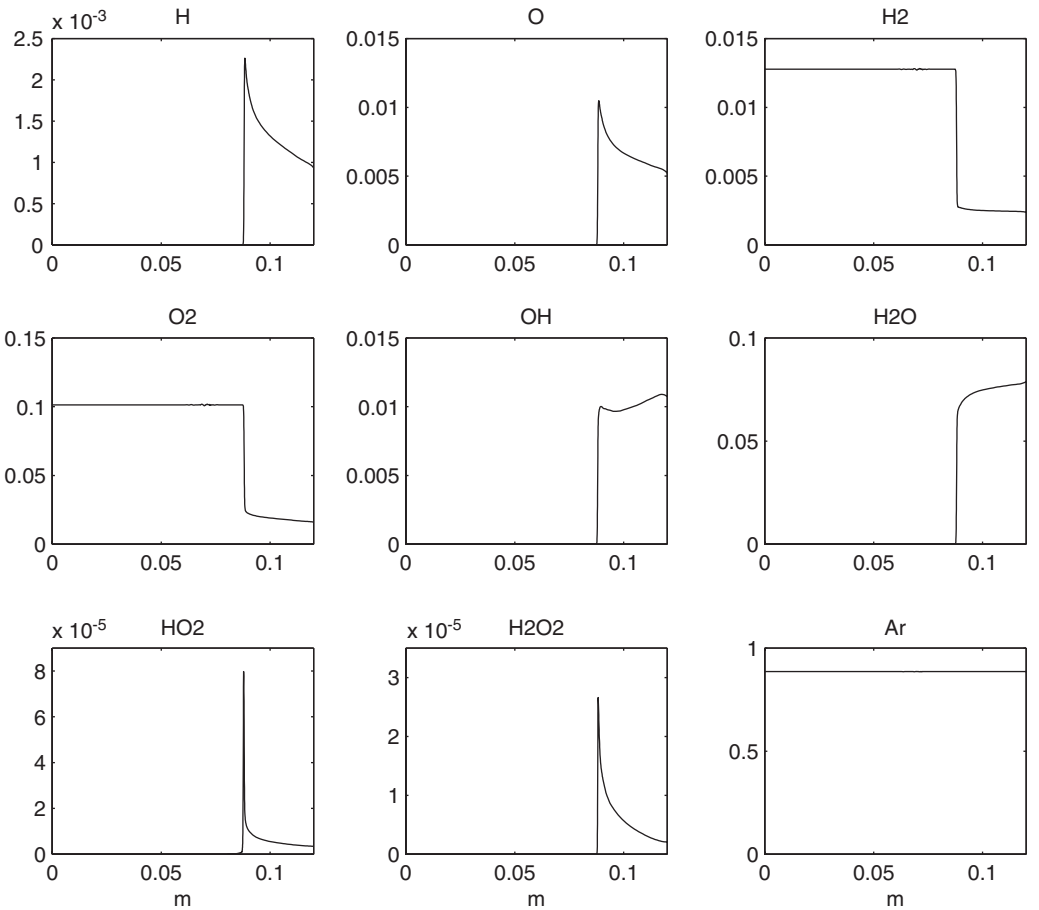


Figure 17. Species mass fractions $Y_i$ vs $x$ at $t = 180\,\mu s$.

requires approximately 2000 points. In order to obtain similar accurate results using a weighted ENO scheme (WENO), Xu *et al.* [55] required 3200 points. The WAMR algorithm, producing results of similar accuracy when $Re = 10^4$, requires between 50 and 300 collocation points; that is, we use approximately an order of magnitude fewer degrees of freedom than TVD/WENO schemes.

(c) *Viscous detonation in $H_2 : O_2 : Ar$ mixture*. We next consider the viscous analogue to the problem considered by Fedkiw *et al.* [56]. Initially a shock propagates in the tube filled with a mixture $H_2 : O_2 : Ar$. After reflection from the end wall, the pressure and temperature behind the shock rise significantly and trigger a chemical reaction which in turn results in a strong detonation wave. The detonation wave propagates in the same direction as the shock and eventually catches up with it. The kinetic model includes 37 reactions involving 9 species ($H, O, Ar, H_2, O_2, HO, H_2O, HO_2, H_2O_2$). Details regarding the physical problem are given by Singh *et al.* [57]. We consider a shock tube of length $L = 0.12\,m$ filled with a mixture of $H_2 : O_2 : Ar$ in a 2 : 1 : 7 ratio, and the following

 

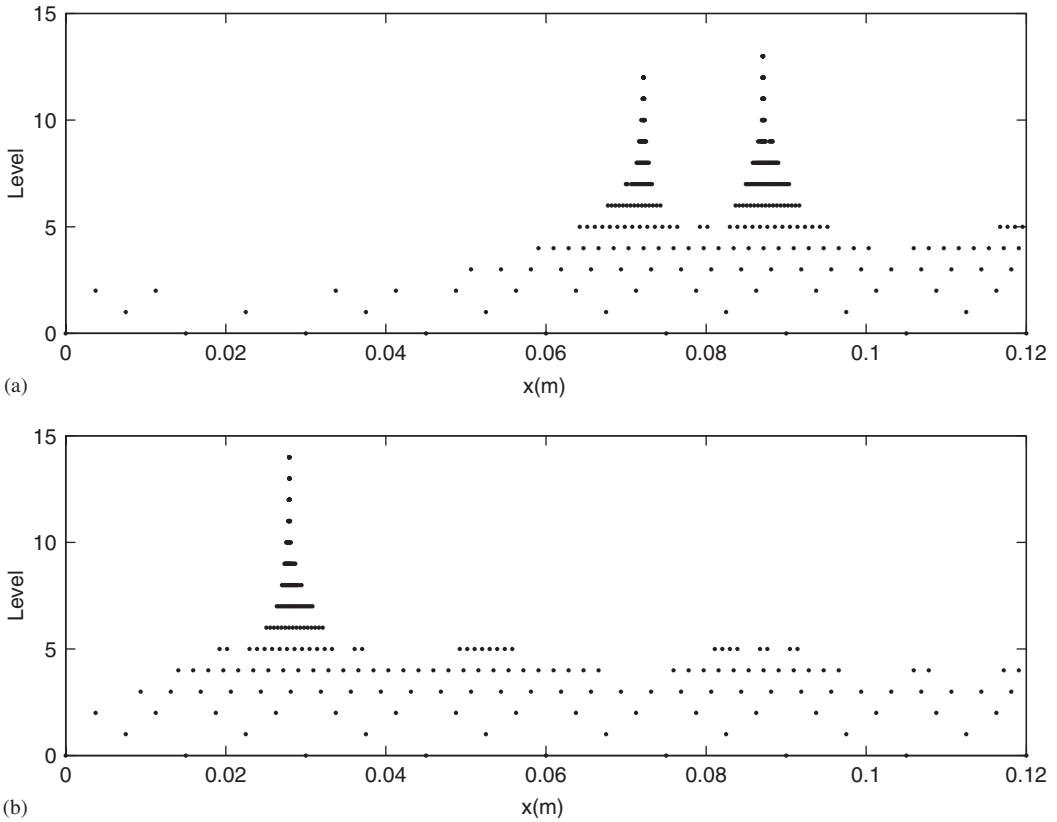


Figure 18. Spatial distribution of collocation points vs resolution level at: (a) $t = 180\,\mu s$; and (b) $t = 230\,\mu s$.

initial conditions:

$$\begin{aligned} &\rho_1 = 0.18075\,\mathrm{kg\,m^{-3}}, \quad \rho_2 = 0.072\,\mathrm{kg\,m^{-3}} \\ &P_1 = 35594\,\mathrm{Pa}, \quad P_2 = 7173\,\mathrm{Pa} \\ &u_1 = 487.34\,\mathrm{m\,s^{-1}}, \quad u_2 = 0\,\mathrm{m\,s^{-1}} \end{aligned} \tag{80}$$

Here subscript 1 refers to $0 \leqslant x \leqslant L/2$ and subscript 2 to $L/2 < x \leqslant L$. We model species fluxes by [58]

$$j_i = -\rho D \frac{\partial Y_i}{\partial x} \tag{81}$$

The following values of dissipation coefficients are used: $\mu = 1.0 \times 10^{-3}\,\mathrm{N\,s\,m^{-3}}$, $k$=8.3 $\mathrm{W\,m^{-1}\,K^{-1}}$, and $D = 5.6 \times 10^{-3}\,\mathrm{m^3\,s^{-1}}$. The threshold parameter is taken to be $\varepsilon = 10^{-3}$.

Figures 15 and 16 show the temperature, velocity, pressure and density profiles at time $t = 180\,\mu s$ and $t = 230\,\mu s$, respectively. Figure 17 displays the species mass

 

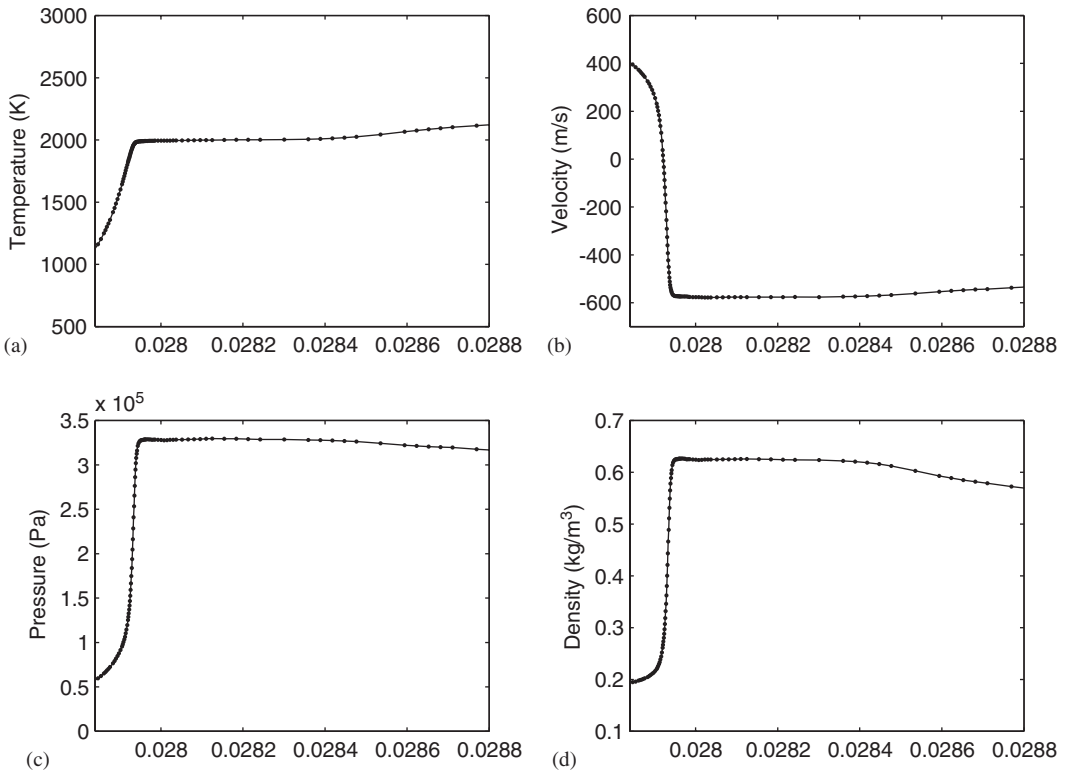


Figure 19. (a) Temperature; (b) velocity; (c) pressure; and (d) density vs $x$ at $t = 230\,\mu$s—magnified fine-scale structures (shock and induction zones) with dots denoting collocation points.

fractions at $t = 180\,\mu$s. The corresponding multilevel grid is shown in Figure 18. In the calculation we use up to 15 levels of resolution and a maximum of approximately 300 collocation points. To demonstrate the impressive ability of the WAMR algorithm to resolve small scales we show the 120-times spatially magnified view of the shock structure in Figure 19. It is clearly seen that both viscous and reactive zones are spatially resolved.

Finally, it is found that the algorithm does not experience any of the difficulties associated with artificial entropy production which occur in Godunov-type methods when a shock wave reflects from a wall or when different small-scale structures (like two shocks) collide [59, 60].

## 5. CONCLUSIONS

A new wavelet-based solution-adaptive multiresolution algorithm is proposed for solving multiscale evolution problems. The method is based on: an efficient wavelet spatial discretization which allows one to minimize the number of degrees of freedom for a prescribed accuracy, a fast algorithm for computing wavelet amplitudes, and efficient and accurate derivative

 

approximations on an irregular grid. A new procedure for error estimation and control based on a Sobolev norm of the sparse wavelet representation of the numerical solution is proposed. It is shown that this procedure results in a more uniform error distribution throughout the computational domain.

In order to demonstrate the versatility of the method we have applied the WAMR algorithm to the simulation of two realistic problems, specifically, to the propagation of a detonation wave in a shock tube, and to the simulation of the ignition-delay mechanism in a $H_2 : O_2 : Ar$ gas mixture. Usually, such problems are treated within the framework of inviscid equations and use of Godunov-type methods. In such case, shocks and other thin structures are considered as discontinuities. These methods can give rise to serious difficulties and possibly result in unphysical behaviour in numerical solutions. Two examples of such anomalies encountered when applying TVD- or ENO-type schemes to simulations of reactive flows have been reported by Bihari and Schwendemann [13]. One is that of wrong reaction front speed, varying widely with time step and the second is a 'numerical' density spike. Moreover, it was shown that numerical simulations of viscous equations are absolutely necessary for studying dynamics of shock propagation in relatively low Reynolds number flows since viscosity in such flows can significantly influence the shock dynamics. The WAMR algorithm, on the other hand, can be applied effectively for the direct numerical simulation of viscous equations, and has been shown to be free of these difficulties. The numerical results show that the collocation grid adapts very effectively to the local evolving structures. The scheme is found to be robust and to produce accurate solutions using a relatively small number of unknowns. Work is in progress in extending the algorithm to higher dimensions and incompressible flows, and we will report on these extensions as results become available.

ACKNOWLEDGEMENTS

The authors are grateful to Dr S. Singh and Dr J. M. Powers for their input and fruitful discussions, and NSF, AFOSR and LANL for their generous support.

REFERENCES

1. Berger MJ, Colella P. Local adaptive mesh refinement for shock hydrodynamics. *Journal of Computational Physics* 1989; **82**(1):64–84.
2. Berger MJ, Leveque RJ. Adaptive mesh refinement using wave-propagation algorithms for hypersonic systems. *SIAM Journal on Numerical Analysis* 1998; **35**(6):2298–2316.
3. Berger MJ, Oliger J. Adaptive mesh refinement for hyperbolic partial differential equations. *Journal of Computational Physics* 1984; **53**(3):484–512.
4. Quirk JJ. An alternative to unstructured grids for computing gas-dynamic flows around arbitrarily complex two-dimensional bodies. *Computational Fluids* 1994; **23**(1):125–142.
5. Quirk JJ. A parallel adaptive grid algorithm for computational shock hydrodynamics. *Applied Numerical Mathematics* 1996; **20**(4):427–453.
6. Chiang YL, van Leer B, Powell KG. Simulation of unsteady inviscid flow on an adaptively refined Cartesian grid. *AIAA Paper 93-0672*, 1993.
7. Jameson L. A wavelet-optimized, very high order adaptive grid and order numerical method. *SIAM Journal on Scientific Computing* 1998; **19**(6):1980–2013.
8. Harten A. Discrete multiresolution analysis and generalized wavelets. *Journal of Applied Numerical Mathematics* 1993; **12**(1–3):153–192.
9. Harten A. Adaptive multiresolution schemes for shock computations. *Journal of Computational Physics* 1994; **115**(2):319–338.
10. Harten A. Multiresolution algorithms for the numerical solution of hyperbolic conservation laws. *Communications on Pure and Applied Mathematics* 1995; **48**(12):1305–1342.

 

11. Cohen A, Kaber SM, Müller S, Postel M. Fully adaptive multiresolution finite volume schemes for conservation laws. *Mathematics of Computation* 2003; **72**(241):183–225.
12. Dahmen W, Gottschlich-Müller B, Müller S. Multiresolution schemes for conservation laws. *Numerische Mathematik* 2001; **88**(3):399–443.
13. Bihari BL, Schwendeman D. Multiresolution schemes for the reactive Euler equations. *Journal of Computational Physics* 1999; **154**(1):197–230.
14. Mackerle J. Error estimates and adaptive finite element methods. A bibliography (1990–2000). *Engineering Computations* 2001; **18**(5/6):802–914.
15. Kita E, Kamiya N. Error estimation and adaptive mesh refinement in boundary element method, an overview. *Engineering Analysis with Boundary Elements* 2001; **25**(7):479–495.
16. Morgan K, Peraire J. Unstructured grid finite-element methods for fluid mechanics. *Reports on Progress in Physics* 1998; **61**(6):569–638.
17. Löhner R. Automatic unstructured grid generators. *Finite Elements in Analysis and Design* 1997; **25**(1/2): 111–134.
18. Nithiarasu P, Zienkiewicz OC. Adaptive mesh generation for fluid mechanics problems. *International Journal for Numerical Methods in Engineering* 2000; **47**(1–3):629–662.
19. Zhu JZ, Hinton E, Zienkiewicz OC. Mesh enrichment against mesh regeneration using quadrilateral elements. *Communications in Numerical Methods in Engineering* 1993; **9**(7):547–554.
20. Löhner R, Baum JD. Adaptive *h*-refinement on 3D unstructured grids for transient problems. *International Journal for Numerical Methods in Fluids* 1992; **14**(12):1407–1419.
21. Hassan O, Probert EJ, Morgan K, Weatherill NP. Unsteady flow simulation using unstructured meshes. *Computer Methods in Applied Mechanics and Engineering* 2000; **189**(4):1247–1275.
22. Mayne DA, Usmani AS, Crapper M. *h*-adaptive finite element solution of unsteady thermally driven cavity problem. *International Journal of Numerical Methods for Heat and Fluid Flow* 2001; **1**(2):172–194.
23. Bacry E, Mallat S, Papanicolaou G. A wavelet based space-time adaptive numerical method for partial differential equations. *RAIRO Model. Math. Anal. Numer.* 1992; **26**(7):793–834.
24. Fröhlich J, Schneider K. An adaptive wavelet-Galerkin algorithm for one- and two-dimensional flame computations. *European Journal of Mechanics B/Fluids* 1994; **13**(4):439–471.
25. Beylkin G, Keiser JM. On the adaptive numerical solution of nonlinear partial differential equations in wavelet bases. *Journal of Computational Physics* 1997; **132**(2):233–259.
26. von Watzdorf R, Marquardt W. Fully adaptive model size reduction for multicomponent separation problems. *Computers and Chemical Engineering* 1997; **21**:811–816.
27. Venini P, Morana P. An adaptive wavelet-Galerkin method for an elastic-plastic-damage constitutive model: 1D problem. *Computer Methods in Applied Mechanics and Engineering* 2001; **190**(42):5619–5638.
28. Vasilyev OV, Paolucci S. A dynamically adaptive multilevel wavelet collocation method for solving partial differential equations in a finite domain. *Journal of Computational Physics* 1996; **125**(2):498–512.
29. Vasilyev OV, Paolucci S. A fast adaptive wavelet collocation algorithm for multidimensional PDEs. *Journal of Computational Physics* 1997; **138**(1):16–56.
30. Cai W, Wang JZ. Adaptive multiresolution collocation methods for initial boundary value problems of nonlinear PDEs. *Journal of Numerical Analysis* 1996; **33**(3):937–970.
31. Bertoluzza S. Adaptive wavelet collocation method for the solution of Burgers equation. *Transport Theory and Statistical Physics* 1996; **25**(3–5):339–352.
32. Abe K, Koro K, Itami K. An *h*-hierarchical Galerkin BEM using Haar wavelets. *Engineering Analysis with Boundary Elements* 2001; **25**(7):581–591.
33. Bertoluzza S, Naldi G, Ravel JC. Wavelet methods for the numerical solutions of boundary value problems on the interval. In *Wavelets*: *Theory*, *Algorithms*, *and Applications*, Chui CK, Montefusco L, Puccio L (eds). Academic Press, Inc.: New York, 1994; 425–448.
34. Schneider K, Kevlahan NK-R, Farge M. Comparison of an adaptive wavelet method and nonlinearly filtered pseudospectral methods for two-dimensional turbulence. *Theoretical and Computational Fluid Dynamics* 1997; **9**(3/4):191–206.
35. Fröhlich J, Schneider K. Computation of decaying turbulence in an adaptive wavelet basis. *Physica D* 1999; **134**(3):337–361.
36. Rastigejev YA, Paolucci S. The use of wavelets in computational fluid mechanics. In *Proceedings of the 3rd ASME/JSME Joint Fluids Engineering Conference,* FEDSM99-7162, San Francisco, 1999.
37. Cai W, Zhang W. An adaptive spline wavelet ADI (SW-ADI) method for two-dimensional reaction–diffusion equations. *Journal of Computational Physics* 1998; **139**(1):92–126.
38. Holmström M. Solving hyperbolic PDEs using interpolating wavelets. *SIAM Journal on Scientific Computing* 1999; **21**(2):405–420.
39. Prosser R, Cant RS. On the use of wavelets in computational combustion. *Journal of Computational Physics* 1998; **147**(2):337–361.
40. Vasilyev OV, Bowman C. Second-generation wavelet collocation method for the solution of partial differential equations. *Journal of Computational Physics* 2000; **165**:660–693.

 

41. Vasilyev OV, Kevlahan NK-R. Hybrid wavelet collocation-Brinkman penalization method for complex geometry flows. *International Journal for Numerical Methods in Fluids* 2002; **40**:531–538.
42. Donoho DL. Interpolating wavelet transforms. *Technical Report 408*, Department of Statistics, Stanford University, 1992.
43. Dubuc S. Interpolation through an iterative scheme. *Journal of Mathematical Analysis and Applications* 1986; **114**(1):185–204.
44. Saito N, Beylkin G. Multiresolution representations using the auto-correlation functions of compactly supported wavelets. *IEEE Transactions on Signal Processing* 1993; **41**(12):3584–3590.
45. Daubechies I. *Ten Lectures on Wavelets*. CBMS-NSF regional conference series in applied mathematics. Society for Industrial and Applied Mathematics, Philadelphia, PA, 1992.
46. van der Vorst H. Bi-CGStab: a fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems. *SIAM Journal on Scientific and Statistical Computing* 1992; **13**:631–644.
47. Liandrat J, Tchamitchian P. Resolution of 1D regularized Burgers equation using a spatial wavelet approximation. *Technical Report ICASE 90-83*, NASA, NASA Langley Research Center, Hampton VA 23665-5225, 1990.
48. Harten A. The artificial compression method for computation of shocks and contact discontinuities, iii self-adjoint hybrid schemes. *Mathematics of Computation* 1978; **32**:363–389.
49. Ducros F, Laporte F, Souleres T, Guinot V, Moinat P, Caruelle B. High-order fluxes for conservative skew-symmetric-like schemes in structured meshes: application to compressible flows. *Journal of Computational Physics* 2000; **161**(1):114–139.
50. Sjögreen B, Yee HC. Multiresolution wavelet based adaptive numerical dissipation control for high order methods. *Journal on Scientific Computing* 2004; **20**(2):211–255.
51. Gerritsen M, Olsson P. Designing an efficient solution strategy for fluid flows ii. Stable high-order central finite difference schemes on composite adaptive grids with sharp shock resolution. *Journal of Computational Physics* 1998; **147**(2):293–317.
52. DeVore RA. *Nonlinear Approximation*, vol. 7. Cambridge University Press: Cambridge, MA, 1998; 51–150.
53. Sanders R, Weiser A. High resolution staggered mesh approach for nonlinear hyperbolic systems of conservation laws. *Journal of Computational Physics* 1992; **101**(2):314–329.
54. Sod GA. A survey of several finite-difference methods for systems of nonlinear hyperbolic conservation laws. *Journal of Computational Physics* 1978; **27**(1):1–31.
55. Xu S, Aslam T, Stewart DS. High resolution numerical simulation of ideal and non-ideal compressible reacting flows with embedded internal boundaries. *Combustion Theory and Modeling* 1997; **1**:113–142.
56. Fedkiw RP, Merriman B, Osher S. High accuracy numerical methods for thermally perfect gas flows with chemistry. *Journal of Computational Physics* 1997; **132**(2):175–190.
57. Singh S, Rastigejev YA, Paolucci S, Powers JM. Viscous detonation in $H_2$–$O_2$–Ar using intrinsic low-dimensional manifolds and wavelet adaptive multilevel representation. *Combustion Theory and Modelling* 2001; **5**(2):163–184.
58. Merk HJ. The macroscopic equations for simultaneous heat and mass transfer in isotropic, continuous and closed systems. *Applied Scientific Research A* 1959; **8**:73–99.
59. Menikoff R. Errors when shock-waves interact due to numerical shock width. *SIAM Journal on Scientific Computing* 1994; **15**(5):1227–1242.
60. Fedkiw RP, Marquina A, Merriman B. An isobaric fix for the overheating problem in multimaterial compressible flows. *Journal of Computational Physics* 1999; **148**(2):545–578.

 