2-D Wavelet-Based Adaptive-Grid Method for the Resolution of PDEs

J. C. Santos, P. Cruz, F. D. Magalhães, and A. Mendes
LEPAE—Chemical Engineering Dept., Faculty of Engineering, University of Porto, 4200-465 Porto, Portugal

An efficient interpolating wavelet-based adaptive-grid numerical method is described for solving systems of bidimensional partial differential equations. The grid is dynamically adapted in both dimensions during the integration procedure so that only the relevant information is stored, saving allocation memory. The spatial derivatives are directly calculated in a nonuniform grid using cubic splines. Numerical results for five typical problems presented illustrate the efficiency and robustness of the method. The adaptive strategy significantly reduces the computational times and the memory requirements, as compared to the fixed-grid approach.

Introduction

In chemical engineering, modeling processes and phenomena, for study, design, optimization, and control purposes, usually imply the solution of systems of partial differential equations (PDEs). The solution of these equations may not be possible analytically, and so numerical methods must be used. The computational effort required by these methods increases if the solution exhibits sharp moving fronts, which are very common when solving hyperbolic equations or parabolic equations with very strong sources, especially if they appear in bidimensional space. It is, therefore, important to have efficient and accurate tools to solve this type of problem.

PDEs can be solved in two ways: discretization in space and time, or discretization only in space. The first method consists in discretizing each PDE in space and time and then solving the resulting nonlinear system of equations. The second method, also called method of lines, consists in discretizing each PDE in space and then solving the resulting system of ordinary differential equations (ODEs) with an appropriate integrator (Finlayson, 1992). This is the method used in this work. The explicit schemes are more stable and faster than implicit schemes for small time-steps. Implicit schemes would allow larger time-steps but, because we are interested in following the moving front, small time-steps must be used (Finlayson, 1992).

The methods that are usually applied to the spatial discretization can be divided into two groups: finite differences methods (FDM) and weight-residual methods (WRM) (Finlayson, 1992). In the latter method, different weighting functions can be used. The most common are the collocation method, the Galerkin method, and the least-square method (Finlayson, 1980). The Galerkin method was proved to be the most accurate in the PDEs discretization, when the solution does not present sharp moving fronts. This method was extensively used in optimization and control content by Christofides' group (Antoniades and Christofides, 2001; Christofides and Daoutides, 1998). However, the method becomes unstable for problems involving sharp moving fronts, like the ones presented here (Finlayson, 1992).

The methods used for the calculation of the convective term in a PDE can be classified as unbounded (central schemes) or bounded [TVD, ENO, flux-corrected transport; Finlayson (1992)]. If an unbounded method is used for the resolution of quasi-hyperbolic PDEs (low viscosity, high Peclet or Reynolds number), the solution will present nonphysical oscillations (Finlayson, 1992), which can be reduced by refining the mesh in the vicinity of the sharp front. For hyperbolic PDEs, an artificial viscosity term must be added to handle shocks; this term has to be decreased until the solution remains unchanged. When bounded methods are used, nonphysical oscillations disappear; however, the solution presents artificial diffusion, which again can be avoided with mesh refinement.

Regardless of the method used, it is possible to conclude that a high density of mesh points must be located in the vicinity of the moving step-front. If a uniform grid is used, independently of the front, a fine grid is used throughout the domain, which is a waste of computational effort (Holmström, 1999) and memory requirements (Hesthaven and Jameson, 1997). Here is where wavelets, and their ability to compress a given set of data, play an important role (Cruz et al., 2002).

Correspondence concerning this article should be addressed to A. Mendes.

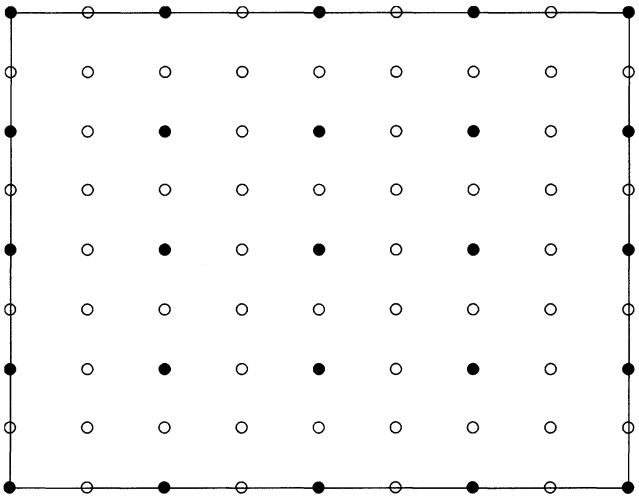


Figure 1. Example of points in a dyadic grid: • V^2 and ○ V^3.

The grid-reduction technique consists of transforming the discrete function data into the wavelet space (wavelet transform) and rejecting the wavelet coefficients that are below a small, predefined threshold. Since each wavelet is uniquely associated with one mesh point, this is omitted from the grid. Cruz et al. (2002) have extensively analyzed this grid-compression technique in one-dimensional chemical engineering problems. The purpose of the authors of the present work is to extend the grid-compression technique to bidimensional grids.

The remainder of the article is organized as follows: the wavelet-based adaptive grid method is presented, and then the calculation of the space derivatives in the adapted grid and the temporal integration are discussed. Finally, the strategy is applied to the solution of five typical two-dimensional (2-D) test cases. The article ends with a summary of the main conclusions.

Multiresolution Representation of Data

The treatment presented here is an extension to 2-D grids of the authors' previous work (Cruz et al., 2002). Consider a set of points in a dyadic grid of the form

$$V^j = \left\{ \left(x_k^j, y_l^j \right) \in \mathbb{R}^2 : x_k^j = 2^{-j}k, y_l^j = 2^{-j}l \right\}, \quad j,k,l \in \mathbb{Z} \quad (1)$$

where j identifies the resolution level, k is the spatial location for the x-direction, and l is the spatial location for the y-direction, as illustrated in Figure 1. Assume that the solution is known on the grid V^j, and one wants to extend it to the finer grid V^{j+1}, which has a quadruple number of points in relation to V^j (it would be double in 1-D). For the even-numbered grid points, the function values are already known, and given by

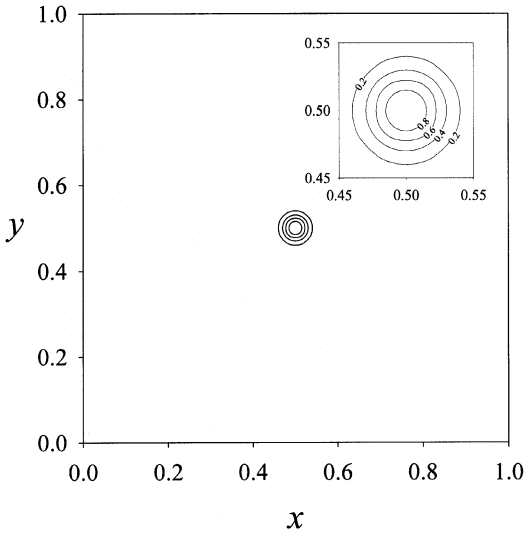
$$u_{2k,2l}^{j+1} = u_{k,l}^j \quad (2)$$

The values in the odd-numbered grid points of V^{j+1} in the x-direction, $u(x_{2k+1}^{j+1}, y_l^j)$, the y-direction, $u(x_k^j, y_{2l+1}^{j+1})$, or both, $u(x_{2k+1}^{j+1}, y_{2l+1}^{j+1})$, are computed by interpolating the known even-numbered grid points in the same direction (present in V^j). The difference between the interpolated and real values is called *wavelet coefficient*, and is expressed as

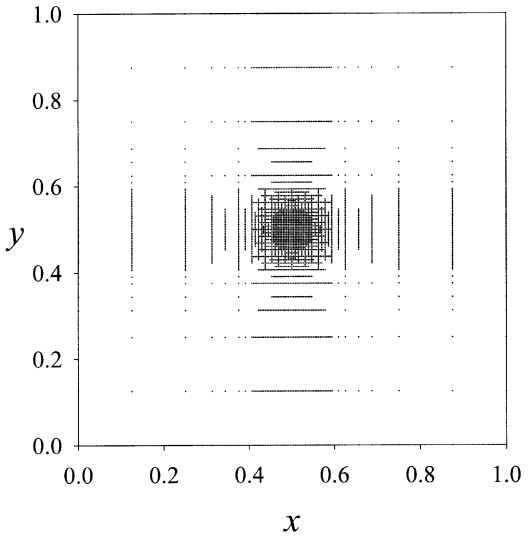
$$d_{1,k,l}^j = u\left(x_{2k+1}^{j+1}, y_{2l}^{j+1}\right) - I_x^j\left[u\left(x_{2k+1}^{j+1}, y_{2l}^{j+1}\right)\right], \quad k,l = 0 \text{ to } 2^j \quad (3)$$

for odd-numbered grid points in the x-direction and even-numbered in the y-direction. I_x^j is the interpolating operator based on the function values in the x-direction, for the corresponding y-coordinate

$$d_{2,k,l}^j = u\left(x_k^j, y_{2l+1}^{j+1}\right) - I_y^j\left[u\left(x_k^j, y_{2l+1}^{j+1}\right)\right], \quad k,l = 0 \text{ to } 2^j \quad (4)$$



(a)



(b)

Figure 2. Grid reduction example using the test function given by Eq. 7.

(a) Function profile (contour plot); (b) corresponding distribution of the grid points. The parameters used in the adaptation algorithm are $\epsilon = 10^{-4}$, $J\min = 3$, $J\max = 8$, and $m = 4$.

for odd-numbered grid points in the y-direction and even-numbered in the x-direction. I_y^j is the interpolating operator based on the function values in the y-direction, for the corresponding x-coordinate

$$d^j_{3,k,l} = u\left(x^{j+1}_{2k+1}, y^{j+1}_{2l+1}\right) - I^j_x\left[u\left(x^{j+1}_{2k+1}, y^{j+1}_{2l+1}\right)\right], \quad k,l = 0 \text{ to } 2^j \tag{5}$$

$$d^j_{4,k,l} = u\left(x^{j+1}_{2k+1}, y^{j+1}_{2l+1}\right) - I^j_y\left[u\left(x^{j+1}_{2k+1}, y^{j+1}_{2l+1}\right)\right], \quad k,l = 0 \text{ to } 2^j \tag{6}$$

for odd-numbered grid points in both directions. $d^j_{3,k,l}$ is the wavelet coefficient based on the I^j_x operator, and $d^j_{4,k,l}$ is the wavelet coefficient based on the I^j_y operator.

The interpolating operators used are based on the Lagrange interpolating polynomial. For this reason, this wavelet family is called *interpolating Lagrange wavelet* (Shi et al., 1999). The polynomial degree, n, is related to the wavelet order, $m = n + 1$, which means that the interpolation only involves m neighborhood points.

The wavelet coefficients in a given direction are a measure of the local "irregular" behavior of the analyzed function in the corresponding direction. If the absolute value of $d^j_{1,k,l}$ or $d^j_{2,k,l}$ is below a given (small) threshold ϵ, then the corresponding grid points, $(x^{j+1}_{2k+1}, y^{j+1}_{2l})$ or $(x^{j+1}_{2k}, y^{j+1}_{2l+1})$, are superfluous in the function representation, and so can be rejected without loss of significant information. Indeed, the function can be reconstructed from the preserved information on the coarser grid V^j. However, in order to reject the

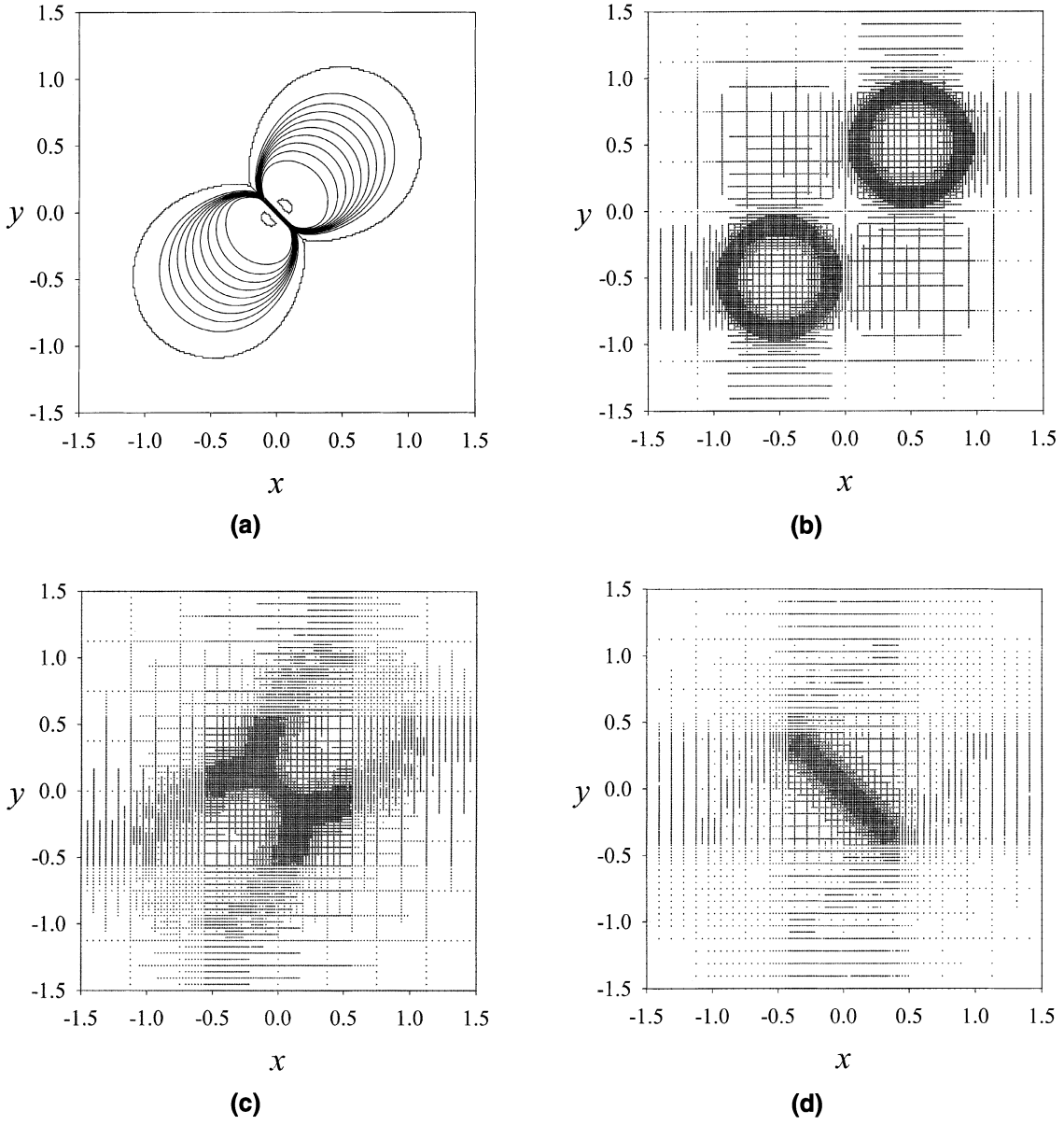


Figure 3. Solution of 2-D Burgers' equation with *Re* = 200.

(a) Dimensionless velocity profile for $\theta = 0.5$ (contour plot); distribution of the grid for (b) $\theta = 0$; (c) $\theta = 0.5$; (d) $\theta = 2$.

point $(x^{j+1}_{2k+1}, y^{j+1}_{2l+1})$, both coefficients $d^j_{3,k,l}$ and $d^j_{4,k,l}$ must be below the threshold.

A function that varies abruptly only in a narrow area of the domain, which is common in many chemical engineering problems, will have most of the $d^j_{i,k,l}$ coefficients close to zero, and so the information can be compressed with great efficiency without loss of accuracy. This 2-D multiresolution approach has been extensively studied in the context of image compression. The interested reader is referred to the work of Zhou (2000).

Finally, it should be noted that the multiresolution approach must select the relevant grid points in a treelike structure, so that it is possible to reconstruct the function value in the next adaptation step, and evaluate $d^j_{i,k,l}$.

In our algorithm, the user must specify the maximum resolution level, $J\max$, in order to avoid grid coalescence in problematic regions. The user also supplies the minimum level of resolution $J\min$. The grid points pertaining to this level of resolution are always conserved throughout the computations.

This grid-reduction technique was numerically tested with the function

$$f(x,y) = \exp\left\{-10^3\left[(x-0.5)^2+(y-0.5)^2\right]\right\} \qquad (7)$$

using $\epsilon = 10^{-4}$, $J\min = 3$, and $J\max = 8$.

Figure 2a shows the resulting approximate function in a contour plot and Figure 2b shows the location of the grid points.

In the resolution of PDEs, the grid should be continually adapted, so that it can automatically adjust to reflect modifications in the solution. The grid-adaptation strategy proposed here is as follows.

Given discrete function values, $f(x,y)$ at time $t = t_1$:

(1) Starting in $j = J\min$.

(2) Compute the wavelet transform for odd-numbered grid points in the x-direction and even-numbered grid points in the y-direction, to obtain the values of the wavelet coefficients, $d^j_{1,k,l}$, for $k,l = 0$ to $2^j - 1$.

(3) Compute the wavelet transform for odd-numbered grid points in the y-direction and even-numbered grid points in the x-direction, to obtain the values of the wavelet coefficients, $d^j_{2,k,l}$ for $k,l = 0$ to $2^j - 1$.

(4) Compute the wavelet transform for odd-numbered grid points in the x- and y-directions, to obtain the values of the wavelet coefficients, $d^j_{3,k,l}$ and $d^j_{4,k,l}$ for $k,l = 0$ to $2^j - 1$.

(5) Increase j by one, and repeat steps 2 to 4 until $j = J\max - 1$.

(6) Identify the wavelet coefficients, $d^j_{1,k,l}$, that fall above the predefined threshold ϵ. The grid points $(x^{j+1}_{2(k+i)+1}, y^{j+1}_{2l})$, with $i = -NL, NR^{(1)}$, and $(x^{j+2}_{2(2k+m)+1}, y^{j+1}_{2l})$, with $m = -NLU+1, NRU^{(2)}$, are included in an indicator.

(7) Identify the wavelet coefficients, $d^j_{2,k,l}$, that fall above the predefined threshold ϵ. The grid points $(x^{j+1}_{2k}, y^{j+1}_{2(l+i)+1})$, with $i = -NL, NR^{(1)}$, and $(x^{j+1}_{2k}, y^{j+2}_{2(2l+m)+1})$, with $m = -NLU+1, NRU^{(2)}$, are included in an indicator.

(8) Identify the wavelet coefficients, $d^j_{3,k,l}$, that fall above the predefined threshold ϵ. The grid points $(x^{j+1}_{2(k+i)+1}, y^{j+1}_{2l+1})$, with $i = -NL, NR^{(1)}$, and $(x^{j+2}_{2(2k+m)+1}, y^{j+1}_{2l+1})$, with $m = -NLU+1, NRU^{(2)}$, are included in an indicator.

(9) Identify the wavelet coefficients, $d^j_{4,k,l}$, that fall above the predefined threshold ϵ. The grid points $(x^{j+1}_{2k+1}, y^{j+1}_{2(l+i)+1})$, with $i = -NL, NR^{(1)}$, and $(x^{j+1}_{2k+1}, y^{j+2}_{2(2l+m)+1})$, with $m = -NLU+1, NRU^{(2)}$, are included in an indicator.

(10) Add to the indicator the grid points associated to the scaling function in the lower resolution level, $J\min$. These are the "basic" grid points that are always maintained throughout the integration.

(11) Remove all the columns and rows that are not pertinent for the function representation in the opposite direction. This procedure reduces significantly the number of grid points without loss of precision, because the function can be reconstructed with the values in the opposite direction.

(12) Beginning at resolution level $j = J\max - 1$, recursively extend the indicator so that all the grid points necessary for the calculation of the existing jth-level wavelet coefficients are included.

N.B.: Superscript (1) indicates the grid points at the same resolution level in each direction, where NL and NR are the

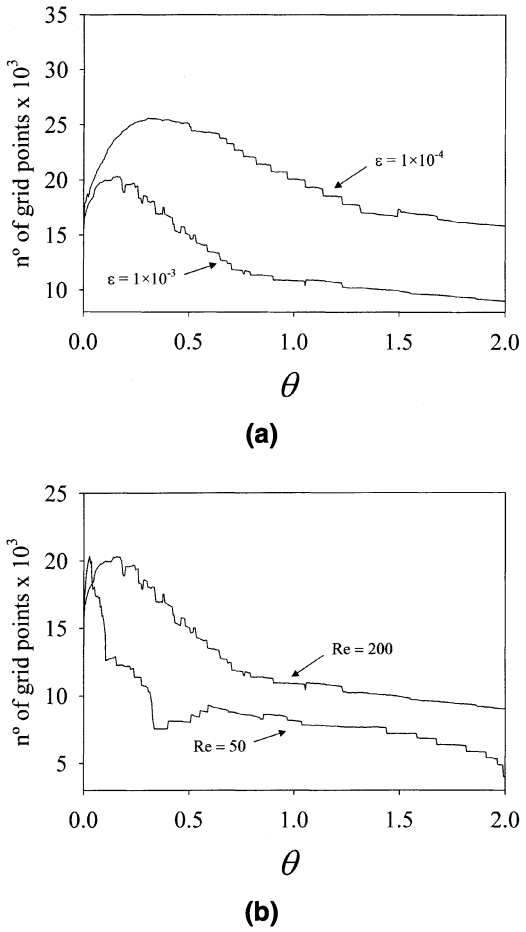


Figure 4. Number of grid points used by the adaptation algorithm for the solution of 2-D Burgers' equation.

(a) $\epsilon = 10^{-3}$ and $\epsilon = 10^{-4}$ and (b) $Re = 200$ and $Re = 50$. The other parameters used are the same as in Figure 3.

number of grid points added to the left and to the right, respectively. This is done in order to account for possible displacement of the sharp features of the solution in the next time-integration step. Superscript (2) indicates the grid points in the resolution level immediately above in both directions, where *NLU* and *NRU* are the number of grid points added to the left and to the right, respectively. *NLU* must be less than or equal to *NL*, and *NRU* must be less than or equal to *NR*. This accounts for the possibility of the solution becoming "steeper" in this region.

This strategy has shown to be efficient in the resolution of single-equation problems. For the resolution of systems of PDEs, the previous procedure must be performed for each PDE, in order to reflect the solutions' behavior of all equations.

Calculation of the Space Derivatives in an Adapted Grid

The space derivatives were calculated directly in the adapted nonuniform grid as in the previous work in one dimension (Cruz et al., 2002). Another possibility is the interpolation of the solution to the maximum resolution level and calculation of the space derivatives in the generated uniform grid (Holmström, 1999). This second strategy is not recommended, since it involves too many unnecessary interpola-

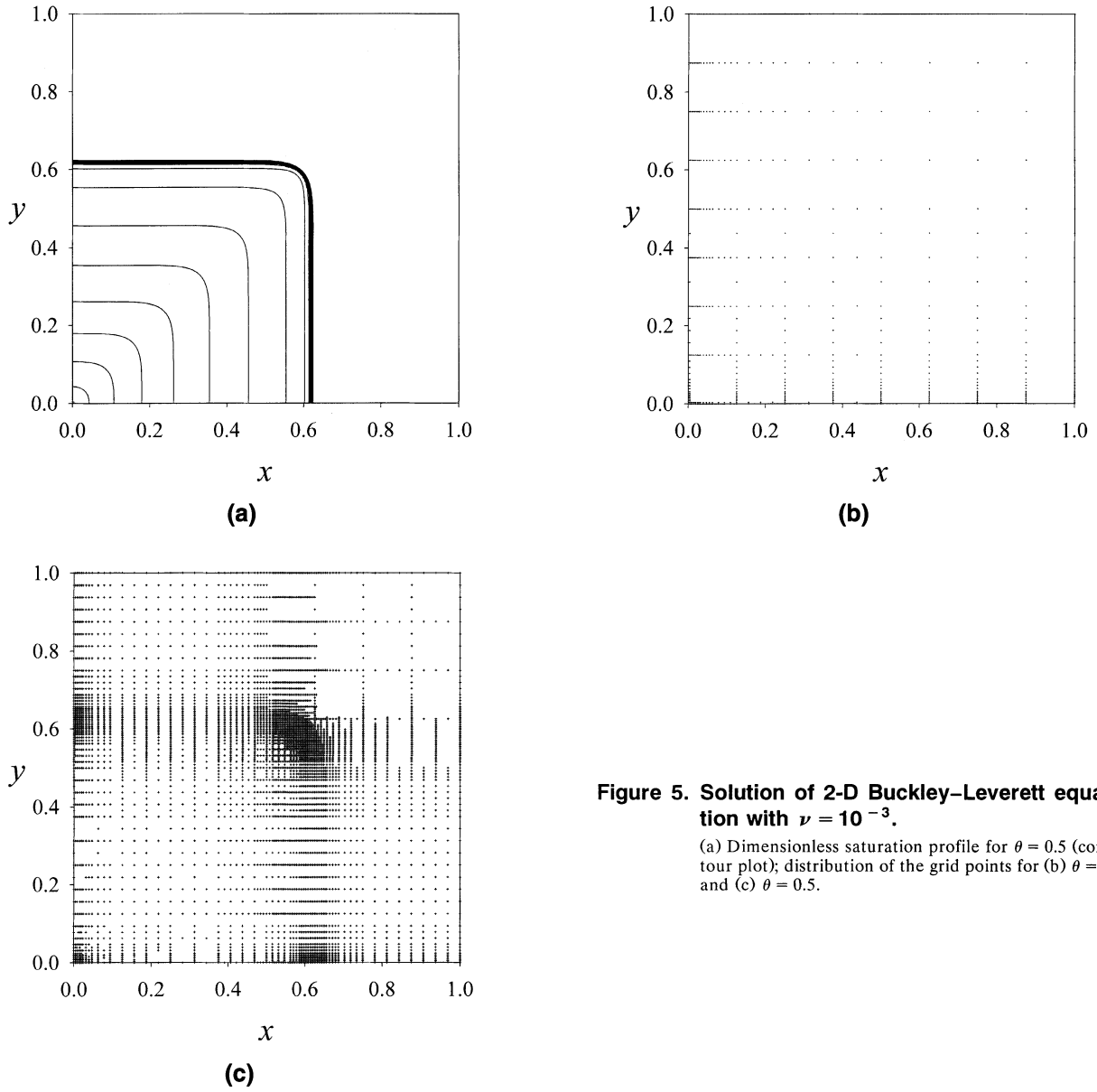


Figure 5. Solution of 2-D Buckley–Leverett equation with $\nu = 10^{-3}$.

(a) Dimensionless saturation profile for $\theta = 0.5$ (contour plot); distribution of the grid points for (b) $\theta = 0$ and (c) $\theta = 0.5$.

tions to obtain the values needed to calculate the space derivatives, thus leading to a significant slowdown in the integration process.

Temporal integration

In this work, the integration of the resulting system of ODEs (initial-value problem), one for each grid point, was done with the Fehlberg fourth–fifth-order Runge–Kutta method (Fehlberg, 1969).

The grid adaptation was performed dynamically throughout the integration. In order to save computation time and avoid superfluous grid adaptations, a criterion was implemented for adjusting the time interval along which the grid stays unchanged, dt_{out}. This is based on the amount of change the grid suffers between two consecutive adaptations and on the progress of the solution (in terms of the magnitude of the time derivatives). This capability is of great importance for the solution of problems where the time scale is unknown *a priori*.

Application examples

The examples presented next attempt to illustrate the effectiveness and robustness of the proposed method of dealing with common test problems: fluid flow (Burgers' equation, the Buckley–Leverett problem, and the compressible Euler system), heat-transfer (temperature wave propagation), and chemical reaction (scalar combustion model). All the results presented were calculated on a personal computer, Pentium IV 1.5 GHz with 256 Mb RIMM. Examples 1, 2, 3, and 4 refer to single-equation problems, while Example 5 deals with a system of PDEs.

In this work we adopted the following adaptation parameters: $\epsilon = 10^{-3}$, $NR = NL = 2$, $NRU = NLU = 1$, $J\min = 3$, $J\max = 8$, and $m = 4$. The space derivatives were computed in an irregular grid using cubic splines.

Example 1: Burgers' Equation. Burgers' equation results from the application of the incompressible fluid Navier–Stokes equation, with negligible pressure gradient. This equation has the special features of the Navier–Stokes equation, but without the complexity of the pressure gradient that has to be obtained from the continuum equation with special numerical algorithms (Ferziger and Perić, 1996). The 2-D Burgers' equation is expressed in conservation form as (Fletcher, 1991)

$$\frac{\partial u^*}{\partial \theta} = \frac{1}{Re}\left(\frac{\partial^2 u^*}{\partial x^2} + \frac{\partial^2 u^*}{\partial y^2}\right) - \left(\frac{\partial f(u^*)}{\partial x} + \frac{\partial f(u^*)}{\partial y}\right) \quad (8)$$

where $f(u^*) = (u^*)^2/2$, u^* is the dimensionless velocity, x and y are the dimensionless space coordinates, θ is the dimensionless time variable, Re is the Reynolds number $Re = Lu_{\text{ref}}/\nu$, u_{ref} is the reference velocity, L is the characteristic length, and ν is the fluid viscosity. This is a classic example that provides important tests for a numerical method, since the problem is nonlinear and consequently discontinuities can develop from smooth solutions (Carlson and Miller, 1998a).

In this specific example, if a sparse grid is used, the solution is erroneous. If the spatial derivatives calculation is performed with an unbounded method, the solution presents nonphysical oscillations (Finlayson, 1992). If a bounded method is used, the solution presents artificial diffusion. Therefore, a high density of grid points must be located in the moving step-front, and the grid must be adapted with a frequency proportional to the wave velocity.

The problem was solved in the intervals $-1.5 \le x \le 1.5$ and $-1.5 \le y \le 1.5$, subject to the following hypothetical initial and boundary conditions, as suggested by Kurganov and Tadmor (2000)

Initial condition: $u^*(x,y,0)$

$$= \begin{cases} -1 & (x-0.5)^2 + (y-0.5)^2 \le 0.4^2 \\ 1 & (x+0.5)^2 + (y+0.5)^2 \le 0.4^2 \\ 0 & \text{elsewhere.} \end{cases}$$

Boundary conditions: $u^*(0,y,\theta) = 0$, $u^*(1,y,\theta) = 0$,

$$u^*(x,0,\theta) = 0,\ u^*(x,1,\theta) = 0$$

For this specific type of initial condition, the cylinders' fronts (note that the cylinder centered in the negative xy-plane has negative velocity) move with a velocity equal to the mean of the velocities just before and after the shock front. The trailing edges do not move at all. The dimensionless velocity profile for $\theta = 0.5$, with $Re = 200$, is presented in Figure 3a, and the distribution of the grid points for $\theta = 0$, $\theta = 0.5$, and $\theta = 2$ in Figures 3b, 3c, and 3d, respectively.

Figure 4a shows the time evolution of the number of grid points used by the adaptation algorithm using two different threshold values, with $Re = 200$. It is interesting to note that when the two fronts touch each other the number of grid points needed to compute the solution, with the prescribed threshold, decreases significantly. This result is very important, since it demonstrates a virtue of the current strategy as compared to other approaches, such as the moving-mesh method, where a constant number of grid points are used throughout the computations. The proposed strategy uses only the grid points that are actually necessary to attain a given precision, and so is more versatile and efficient.

The decrease in the threshold parameter conduces to an increase in the number of grid points used by the adaptation algorithm. As was discussed, this parameter is a direct mea-

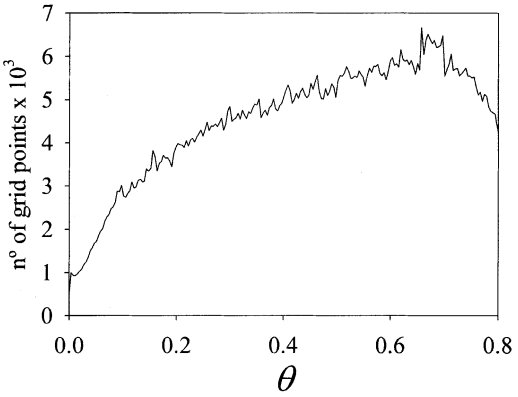


Figure 6. Number of grid points used by the adaptation algorithm for the solution of 2-D Buckley-Leverett equation; the parameters used are the same as in Figure 5.

sure of the error involved in the approximation of the solution by a reduced set of grid points. Higher ϵ values conduce to fewer equations to integrate, but with a lower accuracy.

The influence of the algorithm performance with the increase of the diffusion level is presented in Figure 4b. In this figure the number of grid points used by the adaptation algorithm is presented for two different values of the Reynolds number. Lower Reynolds number values conduce to a more dispersive front, and so the number of grid points required for obtaining the solution with the prescribed threshold decreases.

Example 2: Buckley–Leverett Equation. The 2-D Buckley–Leveret equation is a classic example that results from the application of the material-balance equations to two immiscible fluids, for example, water and oil, one being displaced by the other in a porous medium (Peaceman, 1977). Neither fluid completely fills the space either before or after the injection front, so the problem is solved in terms of a dimensionless saturation variable, s^* (fraction of the space filled with one phase). The 2-D equation that describes this problem, without a pressure gradient, is

$$\frac{\partial s^*}{\partial \theta} = -\left(\frac{\partial f(s^*)}{\partial x} + \frac{\partial f(s^*)}{\partial y} \right) \tag{9}$$

where θ is the dimensionless time variable, and x and y are the space coordinates, $f(s^*)$ is the ratio between the mobility of the two phases,

$$f(s^*) = \frac{s^{*2}}{s^{*2} + (1 - s^*)^2} \tag{10}$$

as suggested by Kurganov and Tadmor (2000), which does not include gravitational effects. The problem was solved in the intervals $0 \le x \le 1$ and $0 \le y \le 1$, subject to the following initial and boundary conditions (Carlson and Miller, 1998b)

Initial conditions: $s^*(x,y,0) = 0$

Boundary conditions: $s^*(0,0,\theta) = 1$

$$\frac{\partial}{\partial y} s^*(x,0,\theta) = 0, \quad \frac{\partial}{\partial x} s^*(0,y,\theta) = 0$$

Since there is no viscosity term in Eq. 9, a small artificial viscosity (diffusive term) was added to its right side to handle the shock wave

$$\nu \left\{ \frac{\partial^2 s^*}{\partial x^2} + \frac{\partial^2 s^*}{\partial y^2} \right\} \tag{11}$$

This highly nonlinear problem has a moving front with a variable velocity. It is a problem that suffers from the same numerical problems as the previous one. The simulation re-

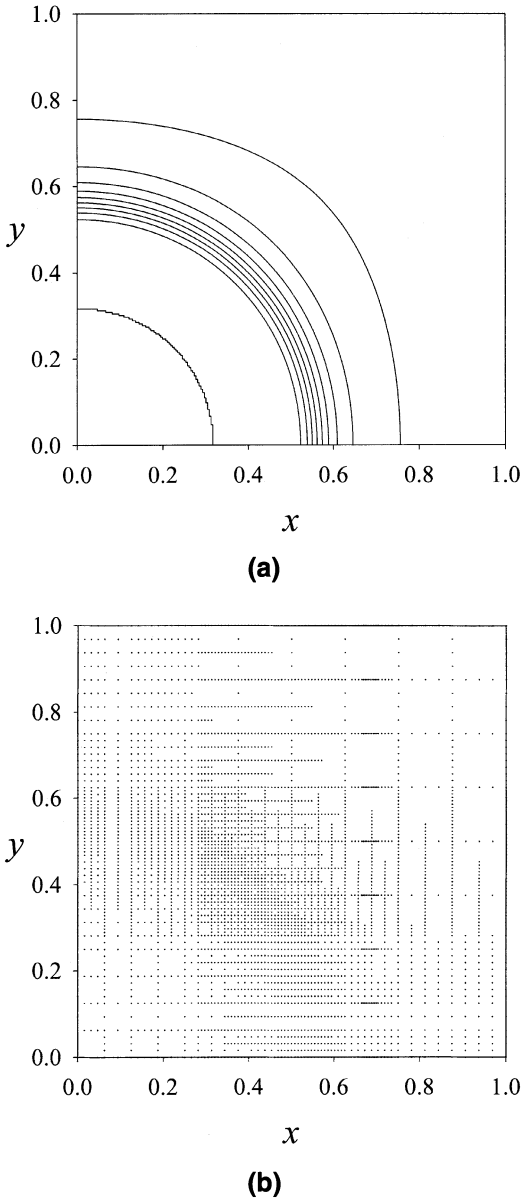


Figure 7. Solution of the scalar combustion model with $\delta = 20$, $R = 5$, and $\alpha = 1$.

(a) Dimensionless temperature profile for $\theta = 0.3$ (contour plot); (b) corresponding distribution of the grid points.

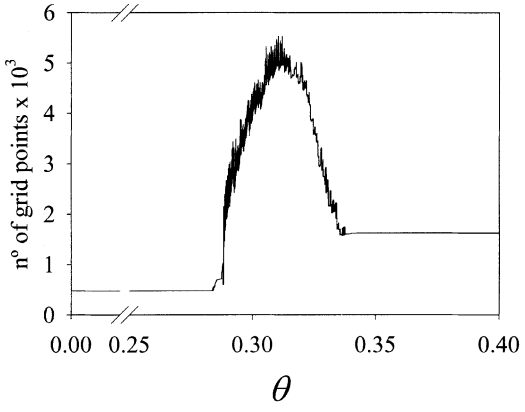


Figure 8. Number of grid points used by the adaptation algorithm for the solution of the scalar combustion model.

The parameters used are the same as in Figure 7.

sults for $\theta = 0.5$, with $\nu = 10^{-3}$, obtained in an adapted grid, are presented in Figure 5a. The adaptation strategy is clearly capable of dealing with highly nonlinear equations, as seen by the correct location of the grid points shown in Figures 5b and 5c for $\theta = 0$ and $\theta = 0.5$, respectively. The number of grid points used by the adaptive procedure is presented in Figure 6.

Example 3: Scalar Combustion Model. This example presents a scalar combustion model that consists of a reaction–diffusion equation without a convective term. This is a typical parabolic problem with a moving step-front due to the strong reaction source. The 2-D extension model described by Adjerid and Flaherty (1986) is

$$\frac{\partial T^*}{\partial \theta} = \left(\frac{\partial^2 T^*}{\partial x^2} + \frac{\partial^2 T^*}{\partial y^2} \right) + D(1 + \alpha - T^*) e^{-\delta/T^*} \quad (12)$$

where $D = R \cdot e^{\delta}/(\alpha\delta)$ and R, α, and δ are constants. The problem was solved in the intervals $0 \le x \le 1$ and $0 \le y \le 1$, subject to the following initial and boundary conditions (Coimbra, 2001)

Initial condition: $T^*(x,y,0) = 1$

Boundary conditions: $\dfrac{\partial T^*}{\partial x}(x,0,t) = \dfrac{\partial T^*}{\partial x}(0,y,t) = 0$

$$T^*(x,1,t) = T^*(1,y,t) = 1$$

The solution represents the temperature of a reactant in a chemical-reaction system. For short periods, the temperature gradually increases from unity, with a "hot spot" forming at $x = y = 0$. Ignition occurs at a finite time, causing the temperature at $x = y = 0$ to rapidly increase to $1 + \alpha$. A front then

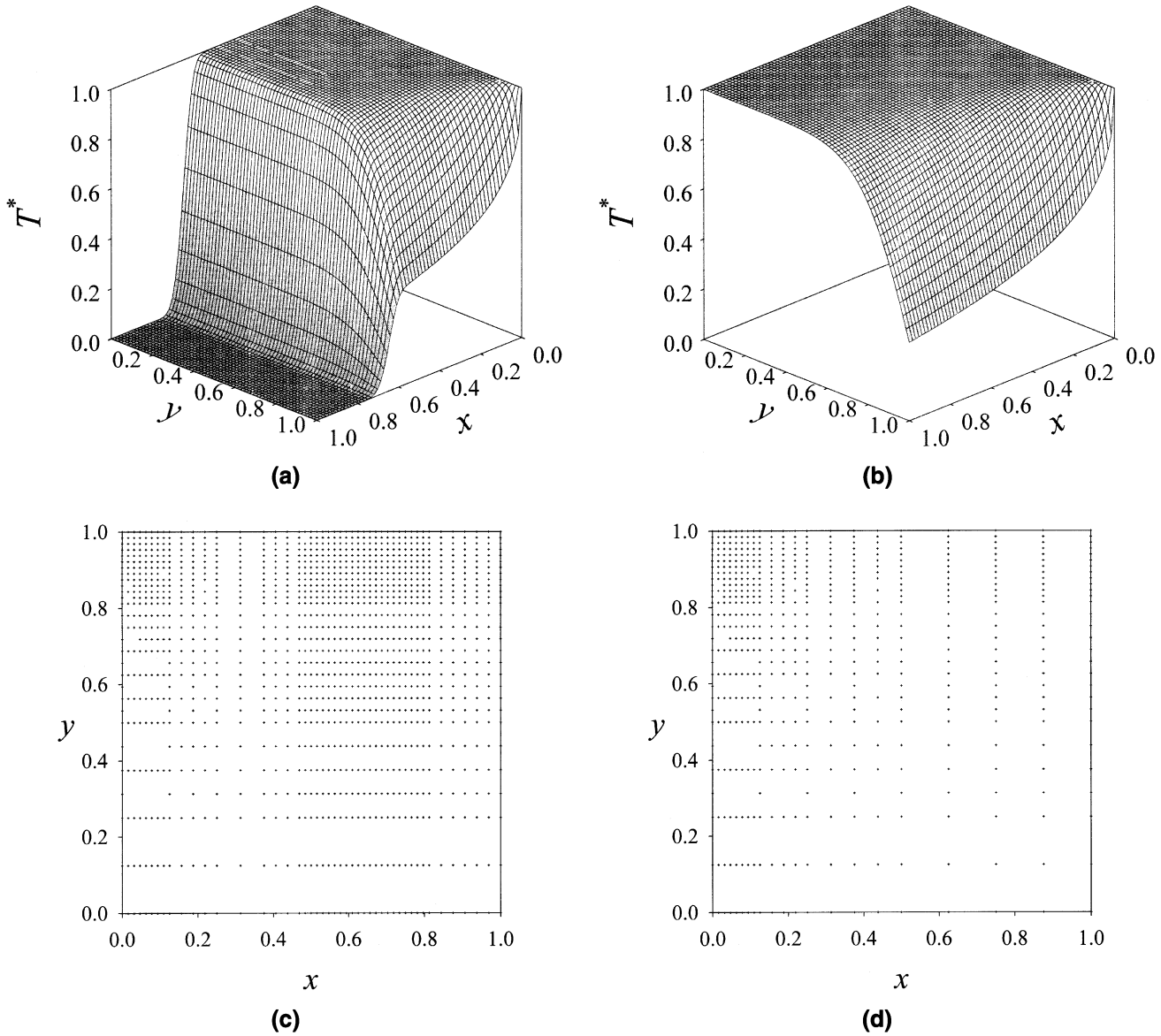


Figure 9. Solution of the temperature wave-propagation model with $Pe_a^H = 1 \times 10^3$, $Bi = 10$, $R^H = 2.3$, and $\gamma = 20$.

Dimensionless temperature profile for (a) $\theta = 1.5$ and (b) $\theta = 4$; distribution of the grid points for (c) $\theta = 1.5$ and (d) $\theta = 4$.

forms and propagates rapidly toward $x = y = 1$ (proportional to δ).

In this problem, a high density of grid points has to be located in the vicinity of the moving step-front, and the grid has to be continually adapted due to the high speed of the front. The numerical problem that frequently arises from the resolution of this type of equation is not concerned with nonphysical oscillations and/or artificial diffusion, as in the last problem, but with the front velocity. If a sparse grid is used, the front moves faster than it actually should move, which may be the reason for the same erroneous results that can be found in the literature.

Figure 7a shows the model solution for $\theta = 0.3$, with $\delta = 20$, $R = 5$, $\alpha = 1$, and Figure 7b shows the location of the grid points for $\theta = 0.3$. The number of grid points used by the adaptive procedure is presented in Figure 8.

Example 4: Temperature Wave Propagation. The following model describes the fixed-bed temperature wave propagation in a packed column. The main assumptions are axially thermal-dispersed plug flow, negligible pressure drop, constant interstitial velocity, thermal equilibrium between the stationary phase and the mobile phases, constant heat capacities, constant densities, and negligible heat accumulation in the column wall. The problem equation is (Coimbra, 2001)

$$R^H \frac{\partial T^*}{\partial \theta} = \frac{1}{Pe_a^H}\left[\frac{\partial^2 T^*}{\partial x^2} + \frac{\gamma}{y}\frac{\partial}{\partial y}\left(y\frac{\partial T^*}{\partial y}\right)\right] - \frac{\partial T^*}{\partial x} \quad (13)$$

where

$$Pe_a^H = \frac{LuCp\rho}{k_a}, \quad \gamma = \frac{L}{R_b}\frac{Pe_a^H}{Pe_r^H},$$

$$Pe_r^H = \frac{R_b u Cp\rho}{k_r}, \quad R^H = \frac{\epsilon_b Cp\rho + (1-\epsilon_b)Cp_s\rho_s}{\epsilon_b Cp\rho}$$

and T^* is the dimensionless temperature, θ is the dimensionless time variable, x is the dimensionless axial coordinate, y is the dimensionless radial coordinate, L is the column length, R_b is the column radius, u is the interstitial velocity, Cp and Cp_s are the fluid and solid heat capacities, ρ and ρ_s are the fluid and solid densities, k_a and k_r are the effective axial and radial thermal conductivity coefficients, and ϵ_b is the bed porosity.

This simple example is good for understanding the temperature radial profiles that are obtained in fixed-bed adsorption and reaction columns, and the concentration radial profiles in membrane separation and reaction processes (the permeation flux across the membrane can be seen as the heat exchange in the column wall) that are always assumed negligible. The problem was solved in the intervals $0 \le x \le 1$ and $0 \le y \le 1$, subject to the following initial and boundary conditions (Coimbra, 2001)

Initial condition: $T^*(x,y,0) = 1$

Boundary conditions: $T^*(0,y,\theta) = T^*_{in}, \quad \dfrac{\partial T^*}{\partial x}(1,y,\theta) = 0$

$$\frac{\partial T^*}{\partial y}(x,0,\theta) = 0, \quad \frac{\partial T^*}{\partial y}(x,1,\theta) = -Bi\,T^*$$

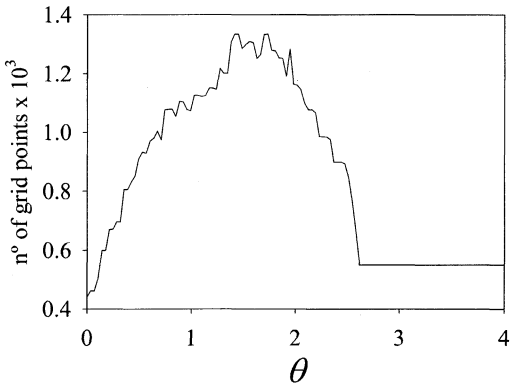


Figure 10. Number of grid points used by the adaptation algorithm for the solution of the temperature wave-propagation model.

The parameters used are the same as in Figure 9.

where Bi is the Biot number, $Bi = hR_b/k_r$, and h is the overall heat-transfer coefficient.

The model solution for $\theta = 1.5$, with $Pe_a^H = 1\times10^3$, $Bi = 10$, $R^H = 2.3$, $\gamma = 20$, is presented in Figure 9. Figure 9a shows the simulation results for $\theta = 1.5$, and Figure 9b shows them for $\theta = 4$. The grid points' location is presented in Figures 9c and 9d. The number of grid points used by the adaptive procedure is presented in Figure 10. When the temperature profile leaves the column, and the steady state is reached, the number of mesh points rapidly decreases to 550, which means a compression of about 98% in memory and CPU time.

Example 5: Compressible Euler System. In this final example, we test the adaptive strategy in the 2-D compressible Euler system of conservation laws for gas dynamics, which is written in a conservative form as follows (Kurganov and Tadmor, 2002)

$$\frac{\partial}{\partial t}\begin{bmatrix}\rho\\ \rho u\\ \rho v\\ E\end{bmatrix} + \frac{\partial}{\partial x}\begin{bmatrix}\rho u\\ \rho u^2 + p\\ \rho uv\\ u(E+p)\end{bmatrix} + \frac{\partial}{\partial y}\begin{bmatrix}\rho v\\ \rho uv\\ \rho v^2 + p\\ v(E+p)\end{bmatrix} = 0 \quad (14)$$

where ρ, u, v, and E are the density, x- and y-velocities, and total energy of the gas per unit volume, respectively. The pressure, p, is given by

$$p = (\gamma - 1)\left[E - \frac{\rho}{2}(u^2 + v^2)\right] \quad (15)$$

where γ is the ratio of specific heats; $\gamma = 1.4$ is a good approximation for air. We solved the benchmark Riemann problem based on the standard 1-D Sod's tube shock problem, which consists of the initial data (Kurganov and Tadmor, 2002)

$$\begin{bmatrix}p\\ \rho\\ u\\ v\end{bmatrix}(x,y,0) = \begin{cases}[1.1\ \ 1.1\ \ 0\ \ 0]^T & x > 0.5,\ \ y > 0.5\\ [0.35\ \ 0.5065\ \ 0.8939\ \ 0]^T & x < 0.5,\ \ y > 0.5\\ [1.1\ \ 1.1\ \ 0.8939\ \ 0.8939]^T & x < 0.5,\ \ y < 0.5\\ [0.35\ \ 0.5065\ \ 0\ \ 0.8939]^T & x > 0.5,\ \ y < 0.5.\end{cases}$$

The gas is initially at rest. At $t = 0$, the diaphragm separating the four regions is removed. At $x = 0$, $x = 1$, $y = 0$, and $y = 1$, reflecting boundary conditions were imposed. This problem exhibits several interactions of nonlinear waves, shock reflection, shock merging, the interaction of a shock with a contact discontinuity, and the reflection of a rarefaction wave.

There is no viscosity term in Eq. 14, therefore, a small artificial viscosity (diffusive term) was added on the right side of each part of Eq. 14 to handle shocks

$$\nu\left\{\frac{\partial^2}{\partial x^2}[\rho \;\; \rho u \;\; \rho v \;\; E]^T + \frac{\partial^2}{\partial y^2}[\rho \;\; \rho u \;\; \rho v \;\; E]^T\right\} \quad (16)$$

Figure 11a shows the model solution for $\theta = 0.2$, with $\nu = 10^{-3}$, and the grid points are given in Figure 11b, $\theta = 0$, and 11c, $\theta = 0.2$. The number of grid points used by the adaptive procedure is presented in Figure 12.

Conclusions

The interpolating wavelet-based adaptive-grid method described here was shown to be very efficient and robust in the resolution of typical two-dimensional PDEs test problems. The method substantially reduces the computational time and allocation memory, because a higher density of mesh points is used only in the vicinity of the moving step-front and a sparse grid is used elsewhere.

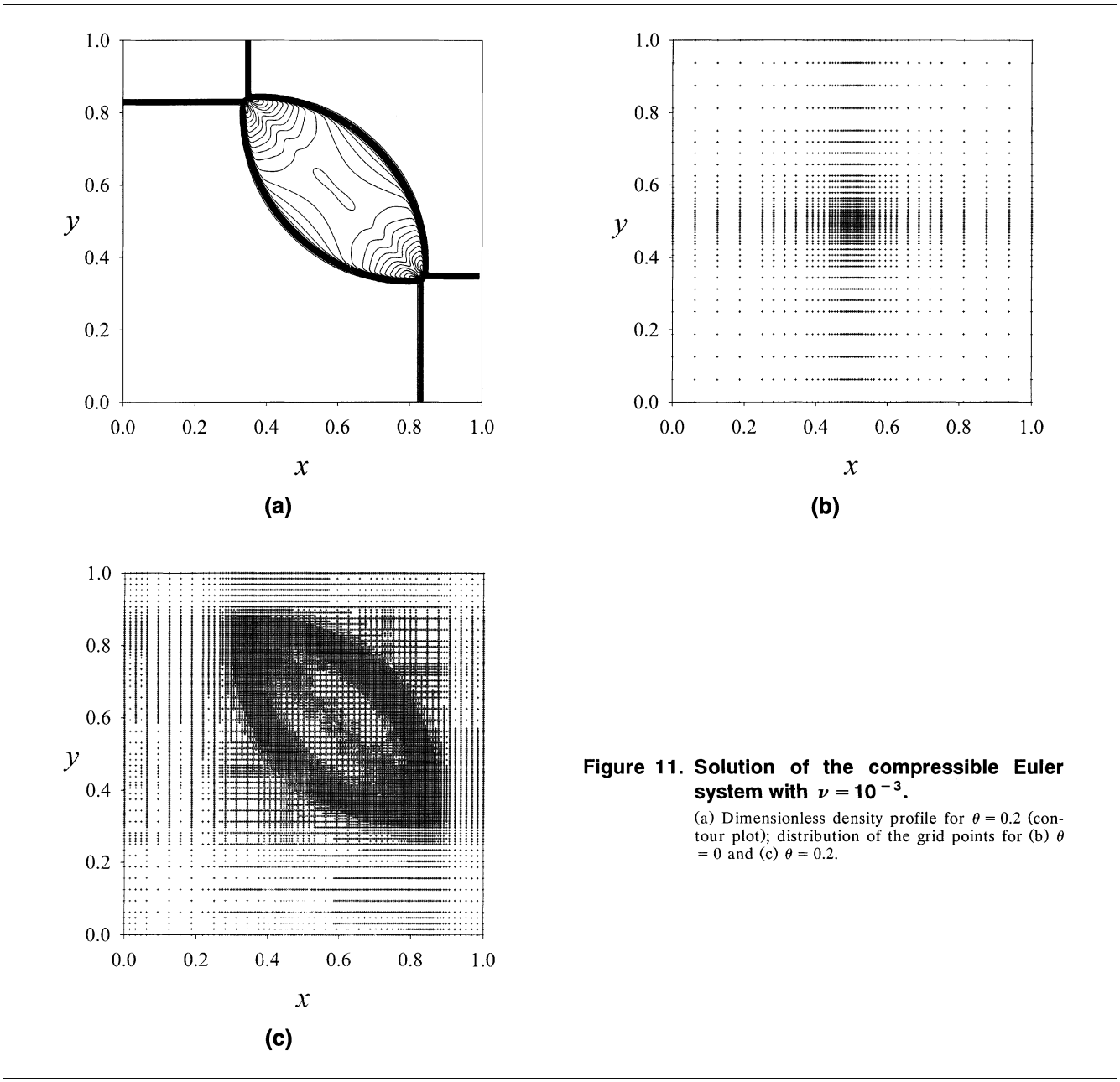


Figure 11. Solution of the compressible Euler system with $\nu = 10^{-3}$.

(a) Dimensionless density profile for $\theta = 0.2$ (contour plot); distribution of the grid points for (b) $\theta = 0$ and (c) $\theta = 0.2$.

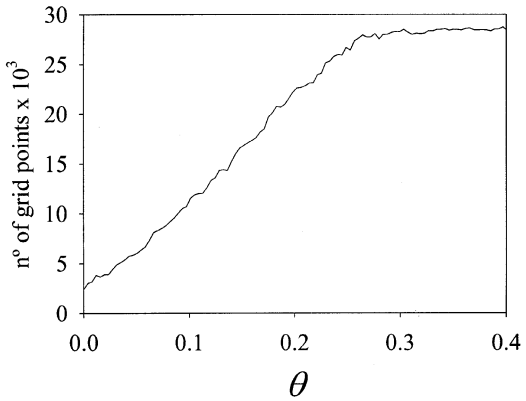


Figure 12. Number of grid points used by the adaptation algorithm for the solution of the compressible Euler system.

The parameters used are the same as in Figure 11.

The strategy applied to the spatial derivatives calculation in a nonuniform grid using cubic splines, was shown to be much more efficient and versatile than others proposed in the literature.

As in the 1-D case (Cruz et al., 2002), the maximum and minimum grid resolution levels usable by the algorithm are user defined, avoiding grid coalescence. The implementation of the algorithm is simple, modular, flexible, and problem independent, without demanding any prior knowledge of the problem's solution. Once the grid adaptation subroutine is available, it can be incorporated into any generic PDE solution package.

Acknowledgments

The work of João Carlos Santos and Paulo Cruz was supported by FCT under Grants SFRH/BD/6817/2001 and BD/21483/99, respectively. The research was supported by funds of Sapiens Project 38067/EQU/2001.

Notation

Bi = Biot number, $Bi = hR_b/K_r$
Cp = fluid-phase heat capacity, J/(K·kg)
Cp_s = solid-phase heat capacity, J/(K·kg)
d = wavelet coefficient
D = combustion model parameter, $D = R \cdot e^{\delta}/(\alpha\delta)$
dt_{out} = time interval along which the grid stays unchanged
E = total energy per unit volume (adimensional)
h = overall heat-transfer coefficient, W/(m²·K)
I = interpolating operator
$J\max$ = maximum resolution level
$J\min$ = minimum resolution level
k_a = effective axial thermal conductivity coefficient, J/(m·s·K)
k_r = effective radial thermal conductivity coefficient, J/(m·s·K)
L = characteristic length, m
m = wavelet order, $m = n + 1$
n = degree of the interpolation polynomial
NL = number of grid points added to the left in the same resolution level
NR = number of grid points added to the right in the same resolution level
NLU = number of grid points added to the left in the level immediately above
NRU = number of grid points added to the right in the level immediately above
p = pressure, $p = (\gamma - 1)[E - (\rho/2)(u^2 + v^2)]$
Pe_a^H = axial Peclet number, $Pe_a^H = Lu_b Cp\rho/k_a$
Pe_r^H = radial Peclet number, $Pe_r^H = R_b u_b Cp\rho/k_r$
R = combustion model parameter
R_b = column radius, m
Re = Reynolds number, $Re = Lu_{ref}/\nu$
R^H = adimensional parameter, $R^H = [\epsilon_b Cp\rho + (1 - \epsilon_b)\, Cp_s\, \rho_s]/\epsilon_b Cp\rho$
s = saturation (fraction of the space filled with one phase)
T = temperature, K
u = velocity (x-direction) or dependent variable
v = velocity (y-direction)
V = dyadic grid
x = dimensionless space coordinate
y = dimensionless space coordinate

Greek letters

α = combustion model parameter
δ = combustion model parameter
ϵ = threshold parameter
ϵ_b = bed porosity
γ = adimensional parameter, $\gamma = (L/R_b)(Pe_a^H/Pe_r^H)$ or ratio of specific heats
ρ = fluid-phase density, kg/m³
ρ_s = solid-phase density, kg/m³
ν = fluid viscosity, kg/(m·s)
θ = dimensionless time variable

Subscripts and superscripts

k = spatial localization for the x-direction
l = spatial localization for the y-direction
x = x-direction
y = y-direction
$*$ = adimensional
j = resolution level

Literature Cited

Adjerid, S., and J. E. Flaherty, "A Moving Finite Element Method with Error Estimation and Refinement for One-Dimensional Time Dependent Partial Differential Equations," *SIAM J. Numer. Anal.*, **23**, 778 (1986).

Antoniades, C., and P. D. Christofides, "Integrating Nonlinear Output Feedback Control and Optimal Actuator/Sensor Placement for Transport-Reaction Processes," *Chem. Eng. Sci.*, **56**, 4517 (2001).

Carlson, N. N., and K. Miller, "Design and Application of a Gradient-Weighted Moving Finite Element Code: I. In One Dimension," *SIAM J. Sci. Comput.*, **19**, 728 (1998a).

Carlson, N. N., and K. Miller, "Design and Application of a Gradient-Weighted Moving Finite Element Code: II. In Two Dimension," *SIAM J. Sci. Comput.*, **19**, 766 (1998b).

Christofides, P. D., and P. Daoutidis, "Robust Control of Hyperbolic PDE Systems," *Chem. Eng. Sci.*, **53**, 85 (1998).

Coimbra, M., *Método dos Elementos Finitos Móveis: Aplicação a Sistemas de Equações de Derivadas Parciais Bidimensionais*, PhD Thesis, Chemical Dept., Faculty of Engineering, Univ. of Porto (2001).

Cruz, P., A. Mendes, and F. D. Magalhães, "A Wavelet-Based Adaptive Grid Method for the Resolution of Nonlinear PDEs," *AIChE J.*, **48**, 774 (2002).

Fehlberg, E., "Low-Order Classical Runge-Kutta Formulas with Step Size Control and Their Application to Some Heat Transfer Problems," NASA Tech. Rep. R-315, NASA, Huntsville, AL (1969).

Ferziger, J. H., and M. Perić, *Computational Methods for Fluid Dynamics*, Springer-Verlag, Berlin (1996).

Finlayson, B., *Nonlinear Analysis in Chemical Engineering*, McGraw-Hill, New York (1980).

Finlayson, B., *Numerical Methods for Problems with Moving Fronts*, Ravenna Park, Seattle, WA (1992).

Fletcher, C. A. J., *Computational Techniques for Fluid Dynamics*, Vol. I, 2nd ed., Springer-Verlag, Berlin (1991).

Hesthaven, J. S., and L. M. Jameson, "A Wavelet Optimized Adaptive Multi-Domain Method," Tech. Rep. 97-52, NASA Langley Research Center, Hampton, VA (1997).

Holmström, M., "Solving Hyperbolic PDEs Using Interpolating Wavelets," *J. Sci. Comput.*, **21**, 405 (1999).
Kurganov, A., and E. Tadmor, "New High-Resolution Central Schemes for Nonlinear Conservation Laws and Convection-Diffusion Equations," *J. Comput. Phys.*, **160**, 241 (2000).
Kurganov, A., and E. Tadmor, "Solution of Two-Dimensional Riemann Problems for Gas Dynamics Without Riemann Problem Solvers," *Numerical Methods for Partial Differential Equations*, **18**, 584 (2002).
Peaceman, P. A., *Fundamentals of Numerical Reservoir Simulation*, Elsevier North–Holland, Amsterdam (1977).
Shi, Z., D. J. Kouri, G. W. Wei, and D. K. Hoffman, "Generalized Symmetric Interpolating Wavelets," *Comput. Phys. Commun.*, **119**, 194 (1999).
Zhou, H.-M., *Wavelet Transforms and PDE Techniques in Image Compression*, PhD Thesis, Dept. of Mathematics, Univ. of California at Los Angeles (2000).

Manuscript received Mar. 8, 2002, and revision received July 31, 2002.