ADAPTIVE GRID SIMULATION OF HYPERBOLIC EQUATIONS

HAOJUN LI[1] AND MYUNGJOO KANG[1,†]

[1]DEPARTMENT OF MATHEMATICAL SCIENCES, SEOUL NATIONAL UNIVERSITY, SOUTH KOREA
E-mail address: {hjli, mkang}@snu.ac.kr

ABSTRACT. We are interested in an adaptive grid method for hyperbolic equations. A multiresolution analysis, based on a biorthogonal family of interpolating scaling functions and lifted interpolating wavelets, is used to dynamically adapt grid points according to the physical field profile in each time step. Traditional finite-difference schemes with fixed stencils produce high oscillations around sharp discontinuities. In this paper, we hybridize high-resolution schemes, which are suitable for capturing singularities, and apply a finite-difference approach to the scaling functions at non-singular points. We use a total variation diminishing Runge–Kutta method for the time integration. The computational cost is proportional to the number of points present after compression. We provide several numerical examples to verify our approach.

1. INTRODUCTION

In this paper, we investigate an adaptive grid method for the simulation of hyperbolic conservation laws, which can be written as

$$\mathbf{U}_t + \mathbf{F}(\mathbf{U})_x = 0, \tag{1.1}$$

with some initial conditions and boundary conditions, where $\mathbf{U}$ is a d-dimensional vector field ($d \geq 1$) and $\mathbf{F}(\mathbf{U})$ is a vector-valued function of $\mathbf{U}$. Equation (1.1) is said to be *hyperbolic* if the Jacobian matrix $\mathbf{A}(\mathbf{U}) = \dfrac{\partial \mathbf{F}(\mathbf{U})}{\partial \mathbf{U}}$ has real eigenvalues and is diagonalizable.

It is well known that non-physical oscillations behind shocks cannot be avoided in traditional finite-difference schemes that use the interpolation of polynomials or some other type of basic functions with fixed stencils. One of the main issues in the hyperbolic conservation laws is the effective capture of sharp singularities, and special treatment must be given to computations around such points. Many studies concerning *high-resolution schemes* (HRSs) have been conducted with the aim of capturing these shocks. For example, such schemes as MINMOD, SMART, ENO, and WENO have been developed (see [13, 35, 11, 34, 18]). These HRSs use the piecewise combination of several finite-difference schemes, either with different orders or with different stencil positions. We need a certain number of points with high-resolution in order to effectively resolve singularities along the wave. This causes the excessive assignment of grid

points, even around regions where the waves are smooth. In evolution equations, the position of a shock often moves with time. It is even possible for new shocks to be created in regions where the wave was previously continuous. Therefore, the dynamic assignment of grid points according to wave profiles will improve computational efficiency for high data compression rates.

In order to determine smooth regions and sharp discontinuities in a function, we apply the concept of multiresolution analysis (MRA) [30, 31]. If we perform a wavelet decomposition of a given function, we obtain the information needed to reconstruct the original field. Not all details are *important*, as long as the field itself is highly localized or has sharp discontinuities in only a small part of the whole domain. We can obtain high accuracy with low-resolution mesh points in the smooth regions of the field, or in regions with unimportant details (or wavelet coefficients).

There have been several other papers concerning adaptive multiresolution methods for hyperbolic conservation laws [1, 5, 17, 6]. Although these papers treated shocks with HRSs, the MRA they used belongs to a biorthogonal family of interpolating scaling functions and interpolating wavelets without lifting. Theoretically, the interpolating wavelets proposed by Donoho [7] have zero vanishing moments and are not a suitable basis for MRA. The order of the vanishing moments was increased by the lifting scheme developed by Sweldens [36]. To our knowledge, the biorthogonal family of interpolating scaling functions and lifted interpolating wavelets applied in MRA was first used by Vasilyev et al. [41] for adaptive grid simulations of time evolution problems. In [41], the authors used only the finite-difference of scaling functions to approximate the spatial derivatives. An adaptive grid approach for shock computations was presented by Regele and Vasilyev [43]. They used the wavelet transform and an assistant function to locate shock positions, and modified the original equation by adding an artificial viscosity term. In this paper, we are interested in solving the original equation instead of the modified equation with artificial viscosity.

We will briefly describe HRSs for shock computations in the following section. In Section 3, we discuss the adaptive wavelet collocation method. Section 4 presents the hybrid AWCM and HRS algorithm for solving evolution PDEs. We give several numerical examples in Section 5, before ending the paper with discussions and conclusions in Section 6.

2. High resolution schemes

There are a number of HRSs for efficiently capturing shocks in conservation laws. Most of them start with a finite volume frame and fix their target by constructing a suitable flux term across the cell faces. Such techniques include *slope limiter* and *flux limiter* approaches, which are essentially equivalent (see for example [28]). These use different finite-difference schemes according to the local slope condition. Other popular approaches include *essentially non-oscillatory* (ENO) schemes [14, 34] and *weighted essentially non-oscillatory* (WENO) schemes [29]. These choose different stencils for polynomial interpolation according to a comparison of the differencing.

We now discuss the flux limiter approach. First-order upwind schemes retain the monotonicity of the solution, although the accuracy can drop due to dissipation, even for smooth waves. Second-order schemes, such as Lax–Wendroff [19] or Beam-Warming [44], give solutions with better accuracy in the smooth part of the wave. However, they perform very poorly in shock-capturing without oscillations. The idea of the flux limiter approach is to unite the best properties of two or more methods in order to keep the discontinuous part of the solution non-oscillatory and obtain the smooth part of the solution with higher accuracy [28].

It is convenient to consider the flux terms in a normalized variable formulation (NVF) or normalized variable space formulation (NVSF) [26, 27, 3]. The convective flux F and the coordinate x are normalized as

$$\tilde{F} = \frac{F - F_u}{F_d - F_u}, \qquad \tilde{x} = \frac{x - x_u}{x_d - x_u}, \tag{2.1}$$

where the subindexes d and u are the downstream and upstream nodes, respectively, of the central node c (see Figure 1). The direction of the stream is determined by the sign of the local velocity (e.g., $\partial F / \partial U$ for the scalar case). If we consider a cell centered at c (uniform mesh), the normalized flux $\tilde{F}$ at the cell face f is a function of $\tilde{F}_c$,

$$\tilde{F}_f = Q\left(\tilde{F}_c\right). \tag{2.2}$$

FIGURE 1. Description of face flux nodes according to stream direction: the main purpose of an HRS is to construct a suitable face flux using neighboring flux values.

Gaskell and Lau [11] introduced the convective boundedness criterion (CBC) for choosing the face flux $\tilde{F}_f$ so as to avoid oscillations. In the normalized formulation, we assume values of 0 at the upstream node u and 1 at the downstream node d. In order to avoid oscillations, Q must satisfy the following conditions:

- Q must be continuous;
- $\tilde{F}_c \leq Q\left(\tilde{F}_c\right) \leq 1$, for $\tilde{F}_c \in [0, 1]$;
- $\tilde{F}_f = \tilde{F}_c$, for $\tilde{F}_c \notin [0, 1]$.

There are many choices for the construction of the face flux $\tilde{F}_f$ [28]. In most of our numerical examples, we apply two popular schemes, MINMOD [13] and SMART [11], although in the Eulerian gas dynamics problem, we used Roe's method [33]. The fluxes given by MINMOD and SMART are as follows:

- MINMOD: $\tilde{F}_f = \max\left(\tilde{F}_c, \min\left(\frac{3}{2}\tilde{F}_c, \frac{1}{2}\tilde{F}_c + \frac{1}{2}\right)\right)$,

- SMART: $\tilde{F}_f = \max\left(\tilde{F}_c, \min\left(3\tilde{F}_c, \frac{3}{4}\tilde{F}_c + \frac{3}{8}, 1\right)\right)$.

MINMOD can also be understood from the perspective of a slope limiter [20]–[24], as it chooses the smaller slope magnitude from the Lax–Wendroff and Beam-Warming methods. If the two slopes have different signs, the slope reverts to a first-order upwind scheme, whose limiter is zero. This is why it is called the MINMOD scheme. SMART exhibits better behavior than MINMOD, because it uses a higher-order scheme, called quadratic upstream interpolation for convective kinematics (QUICK), for smooth portions of the solution.

Before we calculate the normalized flux $\tilde{F}_f$, we must determine the direction of the stream from the sign of the local velocity at the face. If the velocity is not known at a particular point, we interpolate the local velocity using neighboring points on the face. We can then revert to the original flux F_f at face position f using (2.1),

$$F_f = F_u + \tilde{F}_f(F_d - F_u).\tag{2.3}$$

By finite-differencing two successive face fluxes, we approximate the spatial derivative F_x at the center position, i.e.,

$$F_x \approx \frac{F_{i+1/2} - F_{i-1/2}}{\Delta x}.\tag{2.4}$$

3. Adaptive Wavelet Collocation Method

Generally speaking, there are two adaptive wavelet methods for numerical solutions to PDEs: adaptive wavelet Galerkin methods and adaptive wavelet collocation methods [16]. Wavelet Galerkin methods solve problems in wavelet space, and involve a complicated treatment of non-linear terms and boundary values. We introduce an adaptive wavelet collocation method (AWCM) that solves problems in physical space and easily handles non-linear terms and boundary conditions. AWCM uses lifted interpolating wavelet transforms, and was first proposed by Vasilyev et al. to solve evolution equations, mainly in the area of fluid mechanics [41], [42].

We begin with the concept of wavelet decompositions [32, 7]. For a given L^2 function f, its wavelet decomposition is defined as

$$f = \sum_{k \in \mathbb{Z}} \alpha_{j_0,k}\phi_{j_0,k} + \sum_{j=j_0}^{+\infty} \sum_{m \in \mathbb{Z}} \beta_{j,m}\psi_{j,m},\tag{3.1}$$

where ϕ and ψ are a scaling function and a wavelet function, respectively. We denote the dilating and shifting of a function g by the abbreviation $g_{j,k}(\cdot) = 2^{j/2}g(2^j \cdot -k)$. In equation (3.1), the first single sum is the *rough information* contained in f, and the double sum represents *details* of f at various resolution levels. We call the detail coefficient $\beta_{j,m}$ a *wavelet coefficient*. It is clear that a denser set of grid points is needed around the region with larger $|\beta_{j,m}|$. If a field or function is highly localized, we do not need to store huge amounts of information at

regions that are far from the position where the fields are concentrated. We only need more detailed information around the high peak, and some rough information in other regions.

There are many types of scaling functions and wavelets. In this paper, we choose the lifted interpolating wavelet transform ([37, 41, 42]), because its interpolating property enables us to treat non-linear terms and boundary conditions very conveniently, and the lifting property of the scheme provides a better approximation of functions. As we are solving equation (1.1) in finite intervals, basis functions that are close to the boundary should be changed according to the contribution of basis functions outside the original interval. We take this idea from Donoho [7], as did Vasilyev and his co-authors. Near the boundary, the shape of the basis functions changes, and therefore the notation $\phi_{j,k}$ is not valid. This is not a problem if we use the concept of second-generation wavelets [37, 41, 42], which discards the properties of translating and shifting.

3.1. **Adaptive grid with wavelet compression.** Assume f to be a L^2 function with resolution level $j_{max}(\in \mathbb{N})$ defined on a finite interval, i.e.,

$$f = \sum_{k \in \mathbb{Z}} \alpha_{j_{max},k} \phi_{j_{max},k}. \tag{3.2}$$

We consider its wavelet decomposition to be

$$f = \sum_{k \in \mathbb{Z}} \alpha_{j_{min},k} \phi_{j_{min},k} + \sum_{j=j_{min}}^{j_{max}-1} \sum_{m \in \mathbb{Z}} \beta_{j,m} \psi_{j,m}. \tag{3.3}$$

The scaling and wavelet coefficients are calculated by the following normalized forward wavelet transform (FWT),

$$d_{j,k} = \frac{1}{2} \left(c_{j+1,2k+1} - \sum_{l} \tilde{s}_{-l} c_{j+1,2k+2l} \right), \tag{3.4a}$$

$$c_{j,k} = c_{j+1,2k} + \sum_{l} s_{-l} d_{j,k+l}, \tag{3.4b}$$

where $c_{j,k} = 2^{j/2} \alpha_{j,k}$, $d_{j,k} = 2^{j/2} \beta_{j,k}$, and the coefficients s_{-l} and $\tilde{s}_{-l}$ are Lagrangian interpolating weights. We involve Lagrangian weights following [37]. Note that the difference between this wavelet transform and other adaptive multiresolution methods [1, 5, 6, 17] lies in the lifting term in (3.4b). The non-lifted interpolating wavelets have zero vanishing moments. This may be dangerous for numerical approximations. Furthermore, the wavelet coefficients cannot provide information at certain frequency levels while only showing low-pass filter information [41, 36]. With these facts in mind, we prefer to use the lifted interpolating wavelet transform for grid adaptation.

In the double sum in (3.3), only a small number of wavelet coefficients are significant if f is highly localized. We want to compress the information below a certain threshold tolerance,

ε (> 0). Thus, we obtain the following approximated information f_ε,

$$f_\varepsilon = \sum_{k\in\mathbb{Z}} \alpha_{j_{min},k}\phi_{j_{min},k} + \sum_{j=j_{min}}^{j_{max}-1} \sum_{\substack{m\in\mathbb{Z} \\ |\beta_{j,m}|\geq\varepsilon_j}} \beta_{j,m}\psi_{j,m}. \qquad (3.5)$$

where $\varepsilon_j = 2^{-j/2}\varepsilon$. Because ψ is bounded and compactly supported, it is easy to estimate the compression error,

$$\|f - f_\varepsilon\|_\infty \leq C\varepsilon, \qquad (3.6)$$

where C is a constant.

3.2. **Calculation of the spatial derivatives on the adaptive grid.** When we calculate the spatial derivatives at a point on the adaptive grid, we need to determine the corresponding resolution level of that point and interpolate those neighboring points that are missing because of the compression. For an adaptive grid $\mathcal{G}$, the level of the point P is defined as

$$Lev(P) = j_{max} - log_2(dist(P, P')/\Delta x),$$

where Δx is the finest mesh size. We can then approximate f locally in the neighborhood of P by

$$f = \sum_{k\in\mathbb{Z}} \alpha_{Lev(P),k}\phi_{Lev(P),k}, \qquad \text{in some neighborhood of } P. \qquad (3.7)$$

We then differentiate the scaling function representation (3.7) of f at point P to approximate the spatial derivative f_x at P. This works well for problems with smooth functions. However, it is not sufficient to use only scaling function derivatives for those functions that contain discontinuities. In this paper, we use an HRS at those points with the highest resolution level, j_{max}. At other points, we simply differentiate the scaling function. Of course, the range of points at which the HRS is used can be adjusted. It is clear that the computational burden will become heavier if the HRS is used at more points. Our experience shows that it is sufficient to use HRS only at points with the maximum level j_{max}.

4. ADAPTIVE GRID METHOD FOR CONSERVATION LAWS

In this section, we will discuss the proposed algorithm. We begin with an FWT of the initial wave, and compress away unimportant information using a given threshold value $\varepsilon(> 0)$. After compression, we obtain a much lighter grid. Note that some points in the adaptive grid with low resolution levels may become important with time if the propagating singularities approach. Therefore, in each time step, we should extend the *adjacent zone* [41] of every surviving grid point correspondingly. For spatial differentiation, we also need the values at those points neighboring an adaptive grid point, if they are missing due to the compression. We only perform FWT on a fully dense grid in the first time step. After this point, FWT is applied on the adaptive grid. In order to perform FWT, some neighboring point values may need to be interpolated if they are missing. We extend all these missing points into the adaptive grid, and finally perform an inverse wavelet transform (IWT) to reconstruct the physical values.

This completes the preparations for the spatial calculation on the adaptive grid, as discussed in the previous section.

When the approximate spatial derivatives are obtained, we perform a time integration of the wave using a total variation diminishing (TVD) Runge–Kutta method [34]. We now present the second- and third-order TVD Runge–Kutta methods used in the numerical tests. Assume that we want to integrate an ODE $u_t = L(u)$, where L is a differential operator.

- Second-order TVD-RK:

$$u^{(1)} = u^n + \Delta t L(u^n),$$
$$u^{n+1} = \frac{1}{2} u^n + \frac{1}{2} u^{(1)} + \frac{1}{2} \Delta t L(u^{(1)}),$$

- Third-order TVD-RK:

$$u^{(1)} = u^n + \Delta t L(u^n),$$
$$u^{(2)} = \frac{3}{4} u^n + \frac{1}{4} u^{(1)} + \frac{1}{4} \Delta t L(u^{(1)}),$$
$$u^{n+1} = \frac{1}{3} u^n + \frac{2}{3} u^{(2)} + \frac{2}{3} \Delta t L(u^{(2)}).$$

We summarize the proposed algorithm as follows.

The Algorithm: given u^n at time step n and the corresponding adaptive grid mask $\mathcal{G}^n$,

1. perform an FWT of u^n at $\mathcal{G}^n$,
2. extend $\mathcal{G}^n$ for computation of the next time step, $n + 1$,
3. perform IWT to reconstruct physical values,
4. calculate spatial derivatives (using HRS at points with level j_{max}),
5. use TVD Runge–Kutta time integration to update u^n to u^{n+1},
6. if $n < N_T$: $n = n + 1$, go to step 1; else: stop.

5. NUMERICAL EXAMPLES

In this section, we present the results of several numerical experiments using the linear advection equation, Burgers' equation, and Euler gas dynamics problems. For the first two equations, we applied both continuous and discontinuous initial conditions. For the Euler systems, we tested our algorithm with a well-known initial condition given by Sod [35]. In order to obtain *well-compressed but highly accurate* results, we must be careful in our choice of threshold value ε. A large threshold value makes the computation faster at the expense of accuracy, and vice versa. Our choice of threshold ε in the experiments varied from 10^{-4} to 10^{-6}. We used cp to denote the compression percentage of the adaptive grid, i.e.,

$$cp = \frac{\text{cardinality of adaptive grid points}}{\text{cardinality of full grid points}} \times 100\%.$$

5.1. **Linear advection equation.** Our first numerical experiment considers the simplest case, the linear advection equation

$$u_t + u_x = 0, \qquad \text{in } \Omega. \tag{5.1}$$

We solved (5.1) with two types of initial condition. One is a highly localized Gaussian peak, which is continuous, and the other is a discontinuous wave. The boundary conditions in both examples were the zero Neumann conditions, $\partial u/\partial n = 0$ at $\partial\Omega$. We are aware that this is an approximation in the case of the Gaussian peak example, but we ignore this influence from the boundary to the propagating peak.

5.1.1. *Highly localized Gaussian peak as the initial condition.* In this example, the computational domain Ω was $(-0.5, 0.5)$, and the initial condition was a Gaussian pulse $u(x, 0) = \exp\left(-3200(x + 0.4)^2\right)$ in Ω. In this problem, as the Gaussian pulse is continuous even though it is "steep," we did not use an HRS to compute spatial derivatives anywhere. We used the scaling function-based finite-difference scheme, as in AWCM [41]. The adaptive parameters in this test were $j_{max} = 12$, $j_{min} = 3$, and $\varepsilon = 10^{-6}$. Figure 2 shows the evolution of the numerical solution and the numerical grid. We can observe that the numerical grid points of finer levels follow the peak region as the peak propagates to the right.

5.1.2. *Discontinuous initial condition.* In this experiment, the computational domain $\Omega = (-0.5, 0.5)$, and the initial condition was a discontinuous step function given by:

$$u(x, 0) = \begin{cases} 1.0, & \text{if } x < 0, \\ -0.5, & \text{otherwise} \end{cases} \qquad \text{in } \Omega.$$

For this problem, we used a hybrid differentiation for the spatial derivatives. The adaptive parameters were $j_{max} = 12$, $j_{min} = 3$, and $\varepsilon = 10^{-6}$. Figure 3 shows the evolution of the wave and the adaptive numerical grid.

5.2. **Burgers' equation.** Burgers' equation can be written as:

$$u_t + uu_x = 0, \qquad \text{in } \Omega = (-0.5, 0.5). \tag{5.2}$$

Unlike the linear wave equation, the shape of the solution to Burgers' equation changes during its evolution, because the local velocities are not constant. This creates a new singularity, even if the initial wave is continuous. Therefore, for Burgers' equation, we always use a hybrid differentiation consisting of an HRS and scaling function-based finite-difference scheme. We also test two cases for Burgers' equation. One is a continuous initial condition, and the other is a discontinuous initial condition.

5.2.1. *Sine functions as the initial condition.* We applied a continuous initial condition composed of sine functions,

$$u(x, 0) = \sin(2\pi(x + 0.5)) + \sin(\pi(x + 0.5))/3, \qquad \text{in } \Omega.$$

The boundary condition in this test problem was the zero Dirichlet condition. We set $j_{max} = 10$, $j_{min} = 3$, and $\varepsilon = 10^{-4}$.
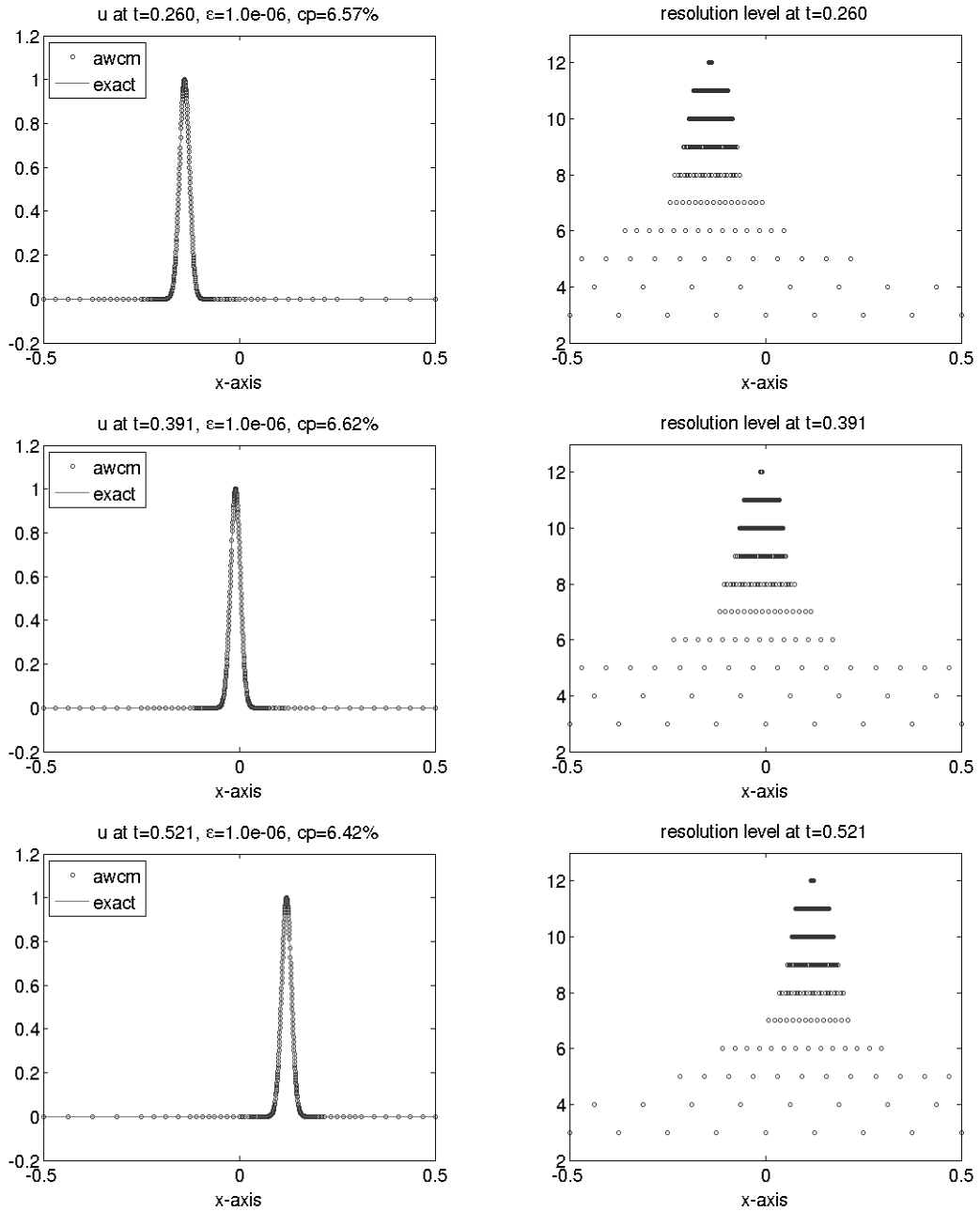
FIGURE 2. Evolution of linear advection equation: a Gaussian peak.

In the beginning, before the shock was created, the numerical grids were coarse. Therefore, the spatial derivatives were obtained by finite-difference scheme. However, the steep part of the domain became steeper and converted into a shock. After this, HRS was used to approximate
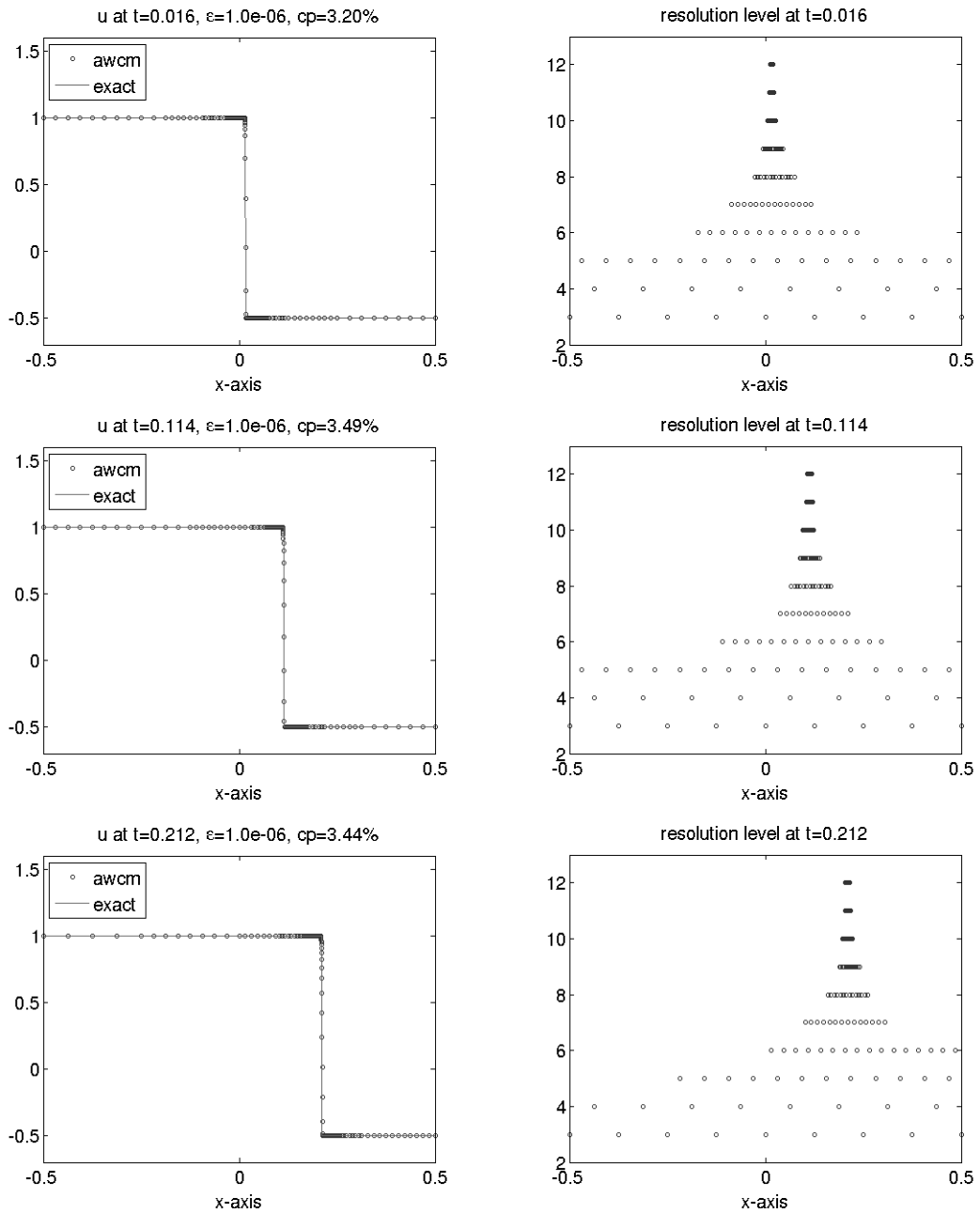
FIGURE 3. Evolution of linear advection equation: a discontinuous wave.

the spatial derivatives at the points around the shock region. We can see that the levels of grid points around the shock region reach the maximum (see Figure 4). If we tested this equation

with an initial condition composed only of a sine function with a single frequency, we would observe that the position of the newly created shock is fixed at its origin.
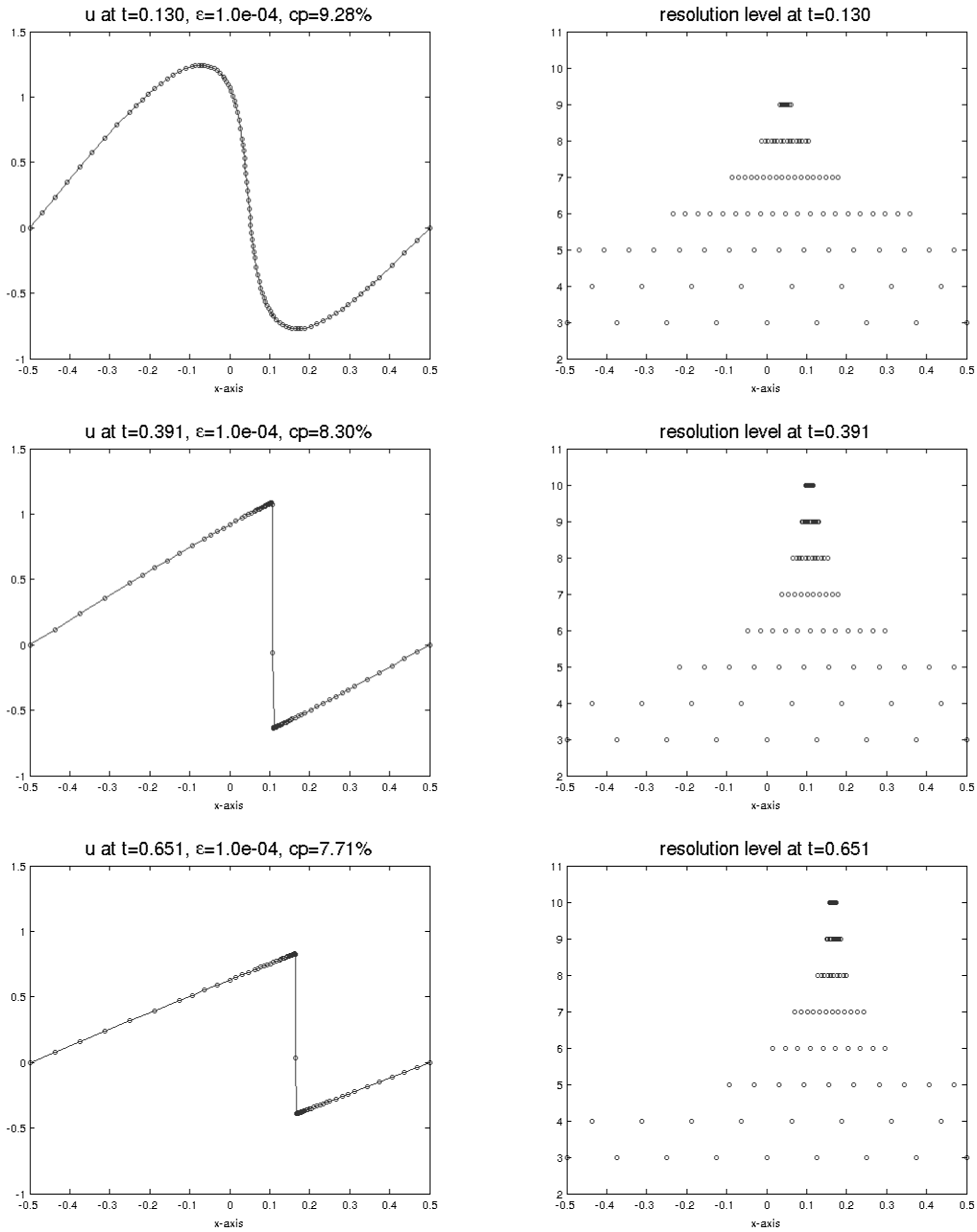

FIGURE 4. Numerical solution of Burgers' equation: sine functions as initial condition.

5.2.2. *Riemann problem.* For the discontinuous case, we tested a Riemann problem:

$$u(x,0) = \begin{cases} 1, & x \le 0, \\ -0.5, & x > 0, \end{cases} \quad \text{in } \Omega$$

with the boundary condition $\partial u / \partial x = 0$. In this example, we considered the function before the discontinuity reached the right boundary. Our adaptive parameters in this problem were $j_{max} = 12$, $j_{min} = 3$, and $\varepsilon = 10^{-5}$. The evolution of the wave and its adaptive grid is plotted in Figure 5.

5.3. **Euler conservation laws.** We now consider the system:

$$\frac{\partial}{\partial t} \begin{pmatrix} \rho \\ \rho u \\ E \end{pmatrix} + \frac{\partial}{\partial x} \begin{pmatrix} \rho u \\ \rho u^2 + p \\ u(E + p) \end{pmatrix} = 0,$$

where

$$\rho : \text{density}$$
$$u : \text{velocity}$$
$$E : \text{total energy}$$
$$p : \text{pressure}$$

and $p = (\gamma - 1)(E - \rho u^2 / 2)$ (in air, $\gamma = 1.4$).

We solve a Riemann problem of this system:

$$(\rho \; \rho u \; E)|_{t=0} = \begin{cases} (1 \; 0 \; 2.5), & x < 0, \\ (0.125 \; 0 \; 0.25), & x > 0. \end{cases}$$

In this example, we applied Roe's scheme [33] for shock computations. We adapted the grid according to the wavelet decomposition of the density ρ at each time step. The adaptive parameters were $j_{max} = 12$, $j_{min} = 3$, and $\varepsilon = 10^{-4}$. Figure 6 shows the evolution of density with the corresponding adaptive grid.

6. Conclusions

In this paper, we tested an adaptive grid method for shock computations in one-dimensional hyperbolic conservation laws by hybridizing HRS and AWCM. The effectiveness of the proposed method was demonstrated by several numerical experiments that exhibited high compression rates for the grid points. We used a wavelet representation for both grid adaptation and discretization of the spatial derivatives at smooth points. The computation time of the algorithm is proportional to the number of surviving grid points as time evolves. The higher the compression rate of adaptive grid points, the faster the method. This method is suitable for simulating highly localized problems.
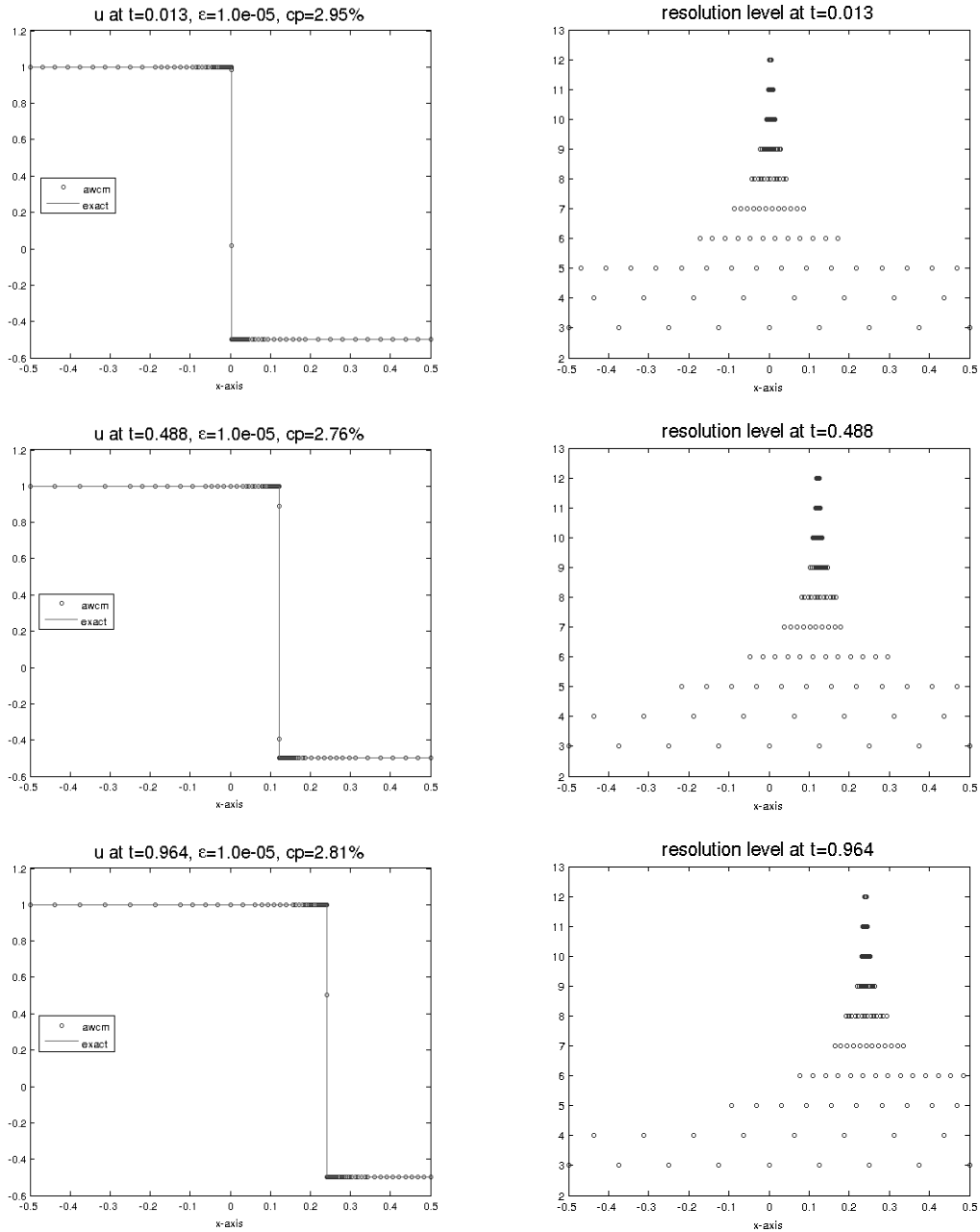
FIGURE 5. Numerical solution of Burgers' equation: discontinuous initial condition.

ACKNOWLEDGMENTS
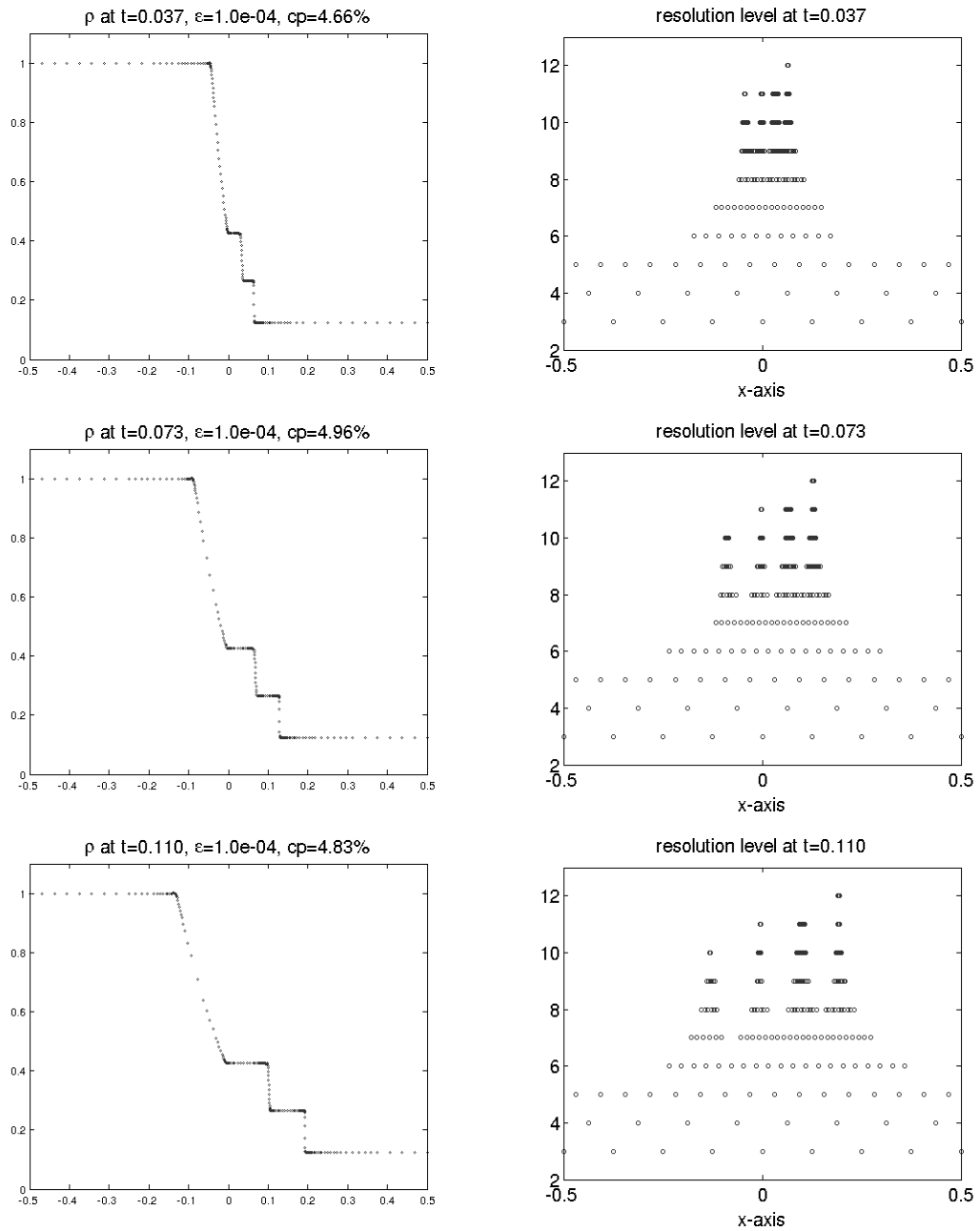
FIGURE 6. Evolution of density.

References

[1] M. A. Alves, P. Cruz, A. Mendes, F. D. Magalhaes, F. T. Pinho, P. J. Oliveira, *Adaptive multiresolution approach for solution of hyperbolic PDEs*, Comput. Methods Appl. Mech. Engrg. 191 (2002) 3909–3928.

[2] B. L. Bihari, A. Harten, *Application of generalized wavelets: An adaptive multiresolution scheme*, J. Comput. Appl. Math. 61 (1995), 275–321.

[3] M. S. Darwish, F. Moukalled, *Normalized variable and space formulation methodology for high-resolution schemes*, Numer. Heat Transfer Part B 30 (1994) 217–237.

[4] I. Daubechies, *Ten Lectures on Wavelets*, CBMS-NSF Regional Conf. Series in Appl. Math. 61, SIAM, 1992.

[5] M. O. Domingues, S. M. Gomes, O. Roussel, K. Schneider, *An adaptive multiresolution scheme with local time stepping for evolutionary PDEs*, J. Comput. Phys. 227 (2008), 3758–3780.

[6] M. O. Domingues, S. M. Gomes, O. Roussel, K. Schneider, *Space-time adaptive multiresolution methods for hyperbolic conservation laws: Applications to compressible Euler equations*, Appl. Nume. Math. 59 (2009), 2303–2321.

[7] D. L. Donoho, *Interpolating wavelet transforms*, Tech. rep., Department of Statistics, Stanford University (1992).

[8] S. Dubuc, *Interpolation through an iterative scheme*, J. Math. Anal. Appl. 114 (1986) 185–204.

[9] S. Goedecker, *Wavelets and their applications for the solution of partial differential equations in physics*, Vol. 4, Presses Polytechniques et Universitaires Romandes, 1998.

[10] G. Deslauriers, S. Dubuc, *Symmetric iterative interpolation processes*, Constr. Approx. 5 (1989) 49–68.

[11] P. H. Gaskell and A. K. C. Lau, *Curvature-Compensated Convective Transport: SMART, A New Boundedness-preserving Transport Algorithm*, Inter. J. Nume. Methods in Fluids 8 (1988), 617–641.

[12] S. Gottlieb, C.-W. Shu, *Total Variation Diminishing Runge-Kutta Schemes*, Math. of Computation 67(2231) (1998) 73–85.

[13] A. Harten, *High Resolution Schemes for Hyperbolic Conservation Laws*, J. Comput. Phys. 49 (1983), 357–393.

[14] A. Harten, B. Engquist, S. Osher and S. Chakravarthy, *Uniformly high order essentially non-oscillatory schemes*, III, J. Comput. Phys. 71 (1987), 231–303.

[15] J. Hesthaven, T. Warburton, *Nodal Discontinuous Galerkin Methods: Algorithms, Analysis, and Applications*, Springer Texts in Applied Mathematics, Springer Verlag, 2008.

[16] L. Jameson, *A Wavelet-optimized Very High Order Adaptive Grid And Order Numerical Method*, SIAM J. Sci. Comput. 19(6) (1998) 1980–2013.

[17] A. J. Kozakevicius, L. C. C. Santos, *ENO adaptive method for solving one-dimensional conservation laws*, Appl. Nume. Math. 59 (2009), 2337–2355.

[18] A. Kurganov, E. Tadmor, *New High-Resolution Central Schemes for Nonlinear Conservation Laws and Convection-Diffusion Equations*, J. Comput. Phys. 160 (2000), 241–282.

[19] P. D. Lax, B. Wendroff, *Systems of conservation laws*, Commun. Pure Appl. Math. 13 (1960), 217–237.

[20] B. van. Leer, *Towards the ultimate conservative difference scheme I. The quest of monotonicity*, Springer Lecture Notes Phys. 18 (1973), 163–168.

[21] B. van. Leer, *Towards the ultimate conservative difference scheme II. Monotonicity and conservation combined in a second order scheme*, J. Comput. Phys. 14 (1974), 361–370.

[22] B. van. Leer, *Towards the ultimate conservative difference scheme III. Upstream-centered finite difference schemes for ideal compressible flow*, J. Comput. Phys. 23 (1977), 263–275.

[23] B. van. Leer, *Towards the ultimate conservative difference scheme IV. A new approach to numerical convection*, J. Comput. Phys. 23 (1977), 276–299.

[24] B. van. Leer, *Towards the ultimate conservative difference scheme V. A second order sequel to Godunov's method*, J. Comput. Phys. 32 (1979), 101–136.

[25] B. P. Leonard, *A stable and accurate convective modelling procedure based on quadratic upstream interpolation*, Comput. Methods Appl. Mech. Eng, 19 (1979) 59–98.

[26] B. P. Leonard, *Locally Modified Quick Scheme for Highly Convective 2-D and 3-D Flows*, Num. Methods in Lam. and Turb. Flows, eds. Taylor C., Morgan K., Pineridge Press, Swansea, UK 5 (1987) 35–47.

[27] B. P. Leonard, *Simple High-accuracy Resolution Program For Convective Modelling Of Discontinuities*, Inter. J. Numer. Meth. Fluids 8 (1988) 1291–1318.

[28] R. J. Leveque, *Finite Volume Methods for Hyperbolic Problems*, Cambridge University Press, 2002.

[29] X.-D. Liu, S. Osher and T. Chan, *Weighted essentially non-oscillatory schemes*, J. Comput. Phys. 115 (1994) 200–212.

[30] S. Mallat, *A Theory for Multiresolution Signal Decomposition: The Wavelet Representation*, IEEE Trans. Patte. Machi. Intel. 11(7) (1989).

[31] S. Mallat, *Multiresolution approximations and wavelet orthonormal bases of $L^2(R)$*, Trans. Amer. Math. Soc. 315 (1989) 69–87.

[32] S. Mallat, *A Wavelet Tour of Signal Processing*, 2nd Edition, Academic Press, 1998.

[33] P. L. Roe, *Approximate Riemann solvers, parameter vectors and difference scheme*, J. Comput. Phys. 43 (1981) 357–372.

[34] C.-W. Shu, S. Osher, *Efficient Implementation of Essentially Non-oscillatory Shock-Capturing Schemes*, J. Comput. Phys. 77 (1988) 439–471.

[35] G. A. Sod, *A Survey of Several Finite Difference Methods for Systems of Nonlinear Hyperbolic Conservation Laws*, J. Comput. Phys. 27 (1978) 1–31.

[36] W. Sweldens, *The lifting scheme: A custom-design construction of biorthogonal wavelets*, Appl. Comput. Harmon. Anal. 3 (1996) 186–200.

[37] W. Sweldens, *The lifting scheme: A construction of second generation wavelets*, SIAM, J. Math. Anal 29 (2) (1998) 511–546.

[38] E. F. Toro, *Riemann Solvers and Numerical Methods for Fluid Dynamics, third edition*, Springer, 2009.

[39] O. V. Vasilyev, S. Paolucci, *A dynamically adaptive multilevel wavelet collocation method for solving partial differential equations in a finite domain*, J. Comput. Phys. 125 (1996) 498–512.

[40] O. V. Vasilyev, S. Paolucci, *A fast adaptive wavelet collocation algorithm for multidimensional PDEs*, J. Comput. Phys. 138 (1997) 16–56.

[41] O. V. Vasilyev, C. Bowman, *Second-generation wavelet collocation method for the solution of partial differential equations*, J. Comput. Phys. 165 (2000) 660–693.

[42] O. V. Vasilyev, *Solving multi-dimensional evolution problems with localized structures using second generation wavelets*, Int. J. Comput. Fluid Dynam. 17 (2) (2003) 151–168.

[43] J. D. Regele, O. V. Vasilyev, *An adaptive wavelet-collocation method for shock computations*, Inter. J. Comput. Fluid Dynam. 23(7) (2009) 503–518.

[44] R. F. Warming, R. M. Beam, *Upwind second-order difference schemes and applications in unsteady aerodynamic flows*, in Proc. AIAA 2nd Computational Fluid Dynamics Conf., Hartford, Conn, 1975.