Journal of Computational Physics **178,** 323–341 (2002)
doi:10.1006/jcph.2002.7024, available online at http://www.idealibrary.com on IDEAL®

A Smoothness Indicator for Adaptive Algorithms for Hyperbolic Systems

Smadar Karni,* Alexander Kurganov,*,† and Guergana Petrova*,‡

*Department of Mathematics, University of Michigan, Ann Arbor, Michigan 48109-1109; †Mathematics Department, Tulane University, New Orleans, Louisiana 70118; and ‡Department of Mathematics, Texas A&M University, College Station, Texas 77843-3368

E-mail: karni@math.lsa.umich.edu, kurganov@math.lsa.umich.edu, and petrova@math.lsa.umich.edu

Received April 25, 2001; revised July 20, 2001

The formation of shock waves in solutions of hyperbolic conservation laws calls for locally adaptive numerical solution algorithms and requires a practical tool for identifying where adaption is needed. In this paper, a new smoothness indicator (SI) is used to identify "rough" solution regions and is implemented in locally adaptive algorithms. The SI is based on the weak local truncation error of the approximate solution. It was recently reported in S. Karni and A. Kurganov, Local error analysis for approximate solutions of hyperbolic conservation laws, where error analysis and convergence properties were established. The present paper is concerned with its implementation in scheme adaption and mesh adaption algorithms. The SI provides a general framework for adaption and is not restricted to a particular discretization scheme. The implementation in this paper uses the central-upwind scheme of A. Kurganov, S. Noelle, and G. Petrova, *SIAM J. Sci. Comput.* **23**, 707 (2001). The extension of the SI to two space dimensions is given. Numerical results in one and two space dimensions demonstrate the robustness of the proposed SI and its potential in reducing computational costs and improving the resolution of the solution. © 2002 Elsevier Science (USA)

Key Words: hyperbolic conservation laws; local truncation error; smoothness indicator; nonoscillatory central schemes.

1. INTRODUCTION

Solutions of initial value problems for hyperbolic systems of conservation laws,

$$u_t + \nabla_x f(u) = 0, \quad u(x, t = 0) = u_0(x), \quad u \in \mathbb{R}^N, \quad x \in \mathbb{R}^d, \tag{1}$$

develop discontinuities, known as shock waves, even for infinitely smooth initial data. This loss of smoothness presents computational challenges both in terms of solution accuracy and

0021-9991/02 $35.00
© 2002 Elsevier Science (USA)
All rights reserved.

mesh requirements. Linear high-order discretizations of (1) inevitably produce oscillations near discontinuities. To prevent such oscillations, nonlinear mechanisms, known as limiters, are required. This adds substantially to the complexity of the schemes and thus reduces their efficiency. Furthermore, such limiters are not needed, or not even desired, in smooth solution regions where simple inexpensive high-order discretizations may end up doing a better job. Mesh requirements is another challenge. Shock-capturing schemes tend to "smear" shock fronts and produce $\mathcal{O}(1)$ solution errors in the shock vicinity. While shock regions may be made as small as desired by choosing sufficiently fine computational meshes, such fine meshes may not be required and are often not affordable in smooth solution regions.

Adapting the solution algorithm to the local nature of the solution is a suitable and appealing strategy and is certainly not new. In a way, nonlinear limiters provide an automatic mechanism for doing precisely that. They locally adapt the scheme from nonoscillatory low-order near shocks to high order in smooth regions. A more general scheme adaption may combine a sophisticated scheme near shocks with an inexpensive, low-complexity, low-dissipation scheme in smooth regions. Other strategies include conservative/nonconservative solution algorithms (see, for example, [6, 25]). Another powerful computational tool is local mesh refinement near shock fronts and other solution singularities in order to insure high resolution that would otherwise be beyond reach [3, 4, 16, 20]. The multiresolution algorithms in [1, 2] have a mixed flavor of scheme adaption and mesh adaption.

All adaptive strategies require a smoothness indicator, capable to detect "rough" solution regions. In [3, 4], such regions are detected using Richardson-type estimates of the local truncation error of the solution. This error estimation procedure assumes smoothness of the solution and ceases to be valid near discontinuities. A more heuristic approach is advocated in [20], where the local wave strengths of the upwind scheme are used as a measure of solution smoothness. In [1, 2], the multiresolution coefficients of the wavelets expansion are used.

More recently, a new smoothness indicator (SI) has been presented in [7]. It measures the local smoothness of the solution by computing the weak local truncation errors (LTE) in the *weak Lip′*-norm (consult Section 2.1 for a brief description). This norm is the "correct" norm to measure errors in the sense that the weak LTE can be translated into pointwise convergence estimates away from discontinuities [7, 23, 24]. It also appears to be highly correlated with the actual errors measured in numerical experiments not only in the scalar case, where the *Lip′* convergence theory is strictly valid, but also in systems [7].

A very useful property of the weak LTE is that it is of significantly different orders of magnitude in smooth and nonsmooth regions. For a scheme of order $r \leq 3$, it is $\mathcal{O}(\Delta)$ near shocks and $\mathcal{O}(\Delta^{r+2})$ away from shocks (see [7] for more detail).

This large difference in orders of magnitude provides a powerful tool for detecting rough parts of the computed solution. In this paper, we employ the recently proposed SI in locally adaptive solution algorithms. We consider (i) scheme adaption and (ii) mesh adaption algorithms. An extension of the SI to two space dimensions is also given. The SI itself provides a general framework for the design of adaptive algorithms and is not tied up to a particular discretization scheme. In this paper we implement it using the Godunov-type central-upwind scheme in [9]. The attractiveness of central schemes is that they do not employ Riemann solvers and characteristic field decomposition and thus are used as black-box solvers. Their most computationally involved ingredient is the piecewise polynomial reconstruction, which is based on nonlinear limiters. As such, they are "ideal" candidates for scheme adaption, since removing the limiters away from discontinuities is likely to result in improved solutions and significant cost reduction. The saving, resulting from local mesh

adaption, can be crudely estimated by the percentage of refined solution regions and is of course highly problem dependent. Having said that, we stress that the main purpose of the present paper is to establish the SI as a reliable general working tool, and not to associate it with any particular numerical method.

The paper is organized as follows. In Section 2, we briefly describe the SI that was derived in [7] for the one-dimensional (1D) case and give its extension to two space dimensions. In Section 3, we describe the implementation of the SI in adaption algorithms, with the central-upwind scheme of [9] being selected as an underlying numerical method. Finally, in Section 4, we apply the proposed adaptive central-upwind scheme and its mesh adaption version to the gas dynamics equations. The results demonstrate the robustness of the SI as an indicator for locally adaptive algorithms—it provides substantial savings in computational cost and improved solution quality.

2. THE LOCAL SMOOTHNESS INDICATOR AND ADAPTION STRATEGIES

2.1. One Space Dimension

Weak solutions u of (1) satisfy

$$E(u,\phi) := -\int_{t=0}^{\infty}\int_{X}\{u(x,t)\phi_t(x,t) + f(u(x,t))\phi_x(x,t)\}\,dx\,dt + \int_{X} u(x,0)\phi(x,0)\,dx = 0 \tag{2}$$

for all test-functions $\phi(x,t) \in C_0^\infty(X \times [0,\infty])$. Since solutions of hyperbolic conservation laws are, in general, discontinuous, a natural way of measuring the quality of the computed solution u^Δ is to measure by how much u^Δ fails to satisfy (2). More precisely, we would like to quantify $E(u^\Delta,\phi)$. We refer to $E(u^\Delta,\phi)$ as the weak truncation error for u^Δ with respect to ϕ.

Denote by $\{u_j^n\}$ the numerical solution on a mesh given by $x_j = j\Delta x$ and $t^n = n\Delta t$. In general, in order to quantify the weak local truncation error (LTE), one needs to extend $\{u_j^n\}$ to a function $u^\Delta(x,t)$ by some reconstruction procedure, and then to integrate against a locally supported test function, which we denote by $\phi_j^n(x,t)$. This involves several considerations. The locally supported test functions need to have continuous derivatives. They also need to approximate the space of test functions to a high order of accuracy, so that the leading order terms in $E(u^\Delta,\phi)$ are determined by the quality of the numerical solution and not by how well $\phi(x,t)$ is approximated by the chosen basis $\phi_j^n(x,t)$. Finally, it is desirable that $\phi_j^n(x,t)$ consist of one type of basis functions, so that only one computation (per cell) of $E(u^\Delta,\phi_j^n)$ is required (this excludes, for example, basic Hermite polynomials). In [7], the following choices were made:

1. u^Δ is reconstructed to be the piecewise constant function

$$u^\Delta(x,t) := u_j^n, \quad \text{for } (x,t) \in \left[x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}}\right] \times \left[t^{n-\frac{1}{2}}, t^{n+\frac{1}{2}}\right]. \tag{3}$$

2. $\phi(x,t)$ are chosen to be $\phi_j^n(x,t) = B_j(x)B^n(t)$, where B_j and B^n are the quadratic B-splines, centered at $x = x_j$ and $t = t^n$, with supports of size $3\Delta x$ and $3\Delta t$, respectively (see, e.g., [5]),

$$B_j(x) := \begin{cases} \frac{1}{2}\left(\frac{x-x_{j-\frac{3}{2}}}{\Delta x}\right)^2, & \text{if } x_{j-\frac{3}{2}} \le x \le x_{j-\frac{1}{2}}, \\ \frac{3}{4} - \left(\frac{x-x_j}{\Delta x}\right)^2, & \text{if } x_{j-\frac{1}{2}} \le x \le x_{j+\frac{1}{2}}, \\ \frac{1}{2}\left(\frac{x-x_{j+\frac{3}{2}}}{\Delta x}\right)^2, & \text{if } x_{j+\frac{1}{2}} \le x \le x_{j+\frac{3}{2}}, \\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

The splines $B^n(t)$ are defined similarly. The quadratic B-splines are a suitable choice since they are C_0^1 locally supported functions and approximate the global test-function space with a third-order accuracy. The latter enables us to convert the weak LTE bounds into local error bounds [7].

A straightforward calculation of $E_j^n := E(u^\Delta, \phi_j^n)$ gives the *weak local truncation error*,

$$E_j^n = \frac{1}{12}\Big\{\big[u_{j+1}^{n+1} - u_{j+1}^{n-1} + 4\big(u_j^{n+1} - u_j^{n-1}\big) + u_{j-1}^{n+1} - u_{j-1}^{n-1}\big]\Delta x + \big[f\big(u_{j+1}^{n+1}\big) - f\big(u_{j-1}^{n+1}\big) + 4\big(f\big(u_{j+1}^{n}\big) - f\big(u_{j-1}^{n}\big)\big) + f\big(u_{j+1}^{n-1}\big) - f\big(u_{j-1}^{n-1}\big)\big]\Delta t\Big\}. \tag{5}$$

For a scheme of formal order of accuracy r, it was shown in [7] that $\{E_j^n\}$ is of order $\mathcal{O}(\Delta^p)$, $p = \min\{5, r+2\}$, away from discontinuities, and of order $\mathcal{O}(\Delta)$ near shocks (Δ is the grid size). The bound $p = 5$ is related to the choice of quadratic B-splines and is suitable for numerical methods of order $r \le 3$. Higher order B-splines lead to a higher-order bound (and result in a wider stencil for E_j^n).

By using the *Lip′* convergence theory, developed by Tadmor and coworkers [18, 19, 23], the weak LTE bounds can be converted into *Lip′* error bounds, and then into L_{loc}^∞ error bounds away from discontinuities (see [7]). While these *Lip′* convergence results are strictly valid for the 1D scalar case, a similar behavior of the weak LTE was also observed experimentally for the 1D system of gas dynamics [7]. This suggests that the weak LTE may provide a reliable smoothness indicator for other time-dependent PDEs which admit discontinuous solutions. Whether or not the weak LTE translates into actual error estimates depends on the availability of convergence theory.

The difference of several orders of magnitude between $\{E_j^n\}$ in smooth and nonsmooth regions makes it a very effective tool for detecting discontinuities. For example, for a third-order scheme $E_j^n = \mathcal{O}(\Delta^5)$ in smooth parts and is $\mathcal{O}(\Delta)$ near shocks. Therefore, a plausible criterion for detecting regions that require adaption might be

$$\begin{array}{ll} \text{if } E_j^n/\Delta^3 > K, & \text{rough region,} \\ \text{otherwise,} & \text{smooth region,} \end{array}$$

where K determines the tolerance for adaption. Once the rough regions are identified, different solution procedures may be used in the different regions, and local mesh refinement may be employed. In this paper, the SI has been used in adaptive methods, based on the central-upwind scheme recently developed in [9] (see Section 3).

2.2. Two Space Dimensions

As in the 1D case, we take u^Δ to be the piecewise constant function

$$u^\Delta(x, y, t) := u_{j,k}^n, \quad \text{for } (x, y, t) \in \left[x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}}\right] \times \left[y_{k-\frac{1}{2}}, y_{k+\frac{1}{2}}\right] \times \left[t^{n-\frac{1}{2}}, t^{n+\frac{1}{2}}\right]. \tag{6}$$

The 2D weak local truncation errors, $\{E^n_{j,k} := E(u^\Delta, \phi^n_{j,k})\}$, are directly computed from (2) for $\phi(x, y, t) = B_j(x)B_k(y)B^n(t)$. Here B_j, B_k, and B^n are the quadratic B-splines, centered at $x = x_j$, $y = y_k$, and $t = t^n$, respectively (see (4)). This results in

$$E^n_{j,k} = \frac{1}{72}\left[\Delta x \Delta y \mathcal{U}^n_{j,k} + \Delta y \Delta t \mathcal{F}^n_{j,k} + \Delta x \Delta t \mathcal{G}^n_{j,k}\right], \tag{7}$$

where

$$\begin{aligned}
\mathcal{U}^n_{j,k} = {} & \left[u^{n+1}_{j+1,k+1} - u^{n-1}_{j+1,k+1} + u^{n+1}_{j+1,k-1} - u^{n-1}_{j+1,k-1} + u^{n+1}_{j-1,k+1} - u^{n-1}_{j-1,k+1}\right. \\
& \left. + u^{n+1}_{j-1,k-1} - u^{n-1}_{j-1,k-1}\right] + 4\left[u^{n+1}_{j+1,k} - u^{n-1}_{j+1,k} + u^{n+1}_{j-1,k} - u^{n-1}_{j-1,k} + u^{n+1}_{j,k+1}\right. \\
& \left. - u^{n-1}_{j,k+1} + u^{n+1}_{j,k-1} - u^{n-1}_{j,k-1}\right] + 16\left[u^{n+1}_{j,k} - u^{n-1}_{j,k}\right], \\
\mathcal{F}^n_{j,k} = {} & \left[f\left(u^{n+1}_{j+1,k+1}\right) - f\left(u^{n+1}_{j-1,k+1}\right) + f\left(u^{n+1}_{j+1,k-1}\right) - f\left(u^{n+1}_{j-1,k-1}\right) + f\left(u^{n-1}_{j+1,k+1}\right)\right. \\
& \left. - f\left(u^{n-1}_{j-1,k+1}\right) + f\left(u^{n-1}_{j+1,k-1}\right) - f\left(u^{n-1}_{j-1,k-1}\right)\right] + 4\left[f\left(u^{n+1}_{j+1,k}\right) - f\left(u^{n+1}_{j-1,k}\right)\right. \\
& \left. + f\left(u^{n}_{j+1,k-1}\right) - f\left(u^{n}_{j-1,k-1}\right) + f\left(u^{n-1}_{j+1,k}\right) - f\left(u^{n-1}_{j-1,k}\right) + f\left(u^{n}_{j+1,k+1}\right)\right. \\
& \left. - f\left(u^{n}_{j-1,k+1}\right)\right] + 16\left[f\left(u^{n}_{j+1,k}\right) - f\left(u^{n}_{j-1,k}\right)\right],
\end{aligned}$$

and

$$\begin{aligned}
\mathcal{G}^n_{j,k} = {} & \left[g\left(u^{n+1}_{j+1,k+1}\right) - g\left(u^{n+1}_{j+1,k-1}\right) + g\left(u^{n+1}_{j-1,k+1}\right) - g\left(u^{n+1}_{j-1,k-1}\right) + g\left(u^{n-1}_{j+1,k+1}\right)\right. \\
& \left. - g\left(u^{n-1}_{j+1,k-1}\right) + g\left(u^{n-1}_{j-1,k+1}\right) - g\left(u^{n-1}_{j-1,k-1}\right)\right] + 4\left[g\left(u^{n+1}_{j,k+1}\right) - g\left(u^{n+1}_{j,k-1}\right)\right. \\
& \left. + g\left(u^{n}_{j-1,k+1}\right) - g\left(u^{n}_{j-1,k-1}\right) + g\left(u^{n-1}_{j,k+1}\right) - g\left(u^{n-1}_{j,k-1}\right) + g\left(u^{n}_{j+1,k+1}\right)\right. \\
& \left. - g\left(u^{n}_{j+1,k-1}\right)\right] + 16\left[g\left(u^{n}_{j,k+1}\right) - g\left(u^{n}_{j,k-1}\right)\right].
\end{aligned}$$

In analogy to the 1D case, for a third-order scheme, $\{E^n_{j,k}\}$ has a magnitude of $\mathcal{O}(\Delta^6)$ in smooth regions, and of $\mathcal{O}(\Delta^2)$ in shock neighborhoods. In this case, a criterion for adaption may be

$$\begin{array}{ll}
\text{if } E^n_j/\Delta^4 > K, & \text{rough region,} \\
\text{otherwise,} & \text{smooth region,}
\end{array}$$

where again K determines the tolerance for adaption.

Remarks.

1. The *Lip′* theory does not extend to the multidimensional case, even for scalar equations. In this case, the proposed SI relies solely on the definition of a weak solution (2). While it may still be computed, convergence estimates do not follow. Yet our numerical examples (see Section 4) demonstrate the reliability of this SI even for multidimensional systems of conservation laws.

2. The SI can be generalized by complete analogy for three-dimensional systems of hyperbolic conservation laws.

3. ADAPTIVE ALGORITHMS

The proposed SI is universal in the sense that it provides a general framework for adaption and may be used with any underlying numerical methods for conservation laws. In this section, we describe the implementation of the SI using the Godunov-type central-upwind scheme in [9]. The scheme, used for the computation in Section 4, is briefly described below for completeness, and also for giving the flavor of how it might be adapted.

3.1. The Underlying Scheme

Godunov-type central schemes are based on averaging exact solutions over Riemann fans [17]. By using the local speeds of propagation, the width of the Riemann fan may be estimated more accurately. This results in schemes with significantly reduced numerical dissipation and which admit a semidiscrete form (see [8, 10, 11]). *One-sided* estimates of the propagation speed yields further reduced dissipation and are referred to as central-upwind schemes in [9].

Denote by $\bar{u}_j(t)$ the computed cell averages at time t,

$$\bar{u}_j(t) := \frac{1}{\Delta x}\int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} u(x,t)\,dx.$$

Using this data, we reconstruct a nonoscillatory piecewise polynomial of the form

$$\tilde{u}(x,t) := \sum_j p_j(x,t)\chi_j(x), \tag{8}$$

where χ_j is the characteristic function of the interval $[x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}}]$.

Then the semidiscrete *central-upwind* scheme can be written in the conservative form (see [9])

$$\frac{d}{dt}\bar{u}_j(t) = -\frac{H_{j+\frac{1}{2}}(t) - H_{j-\frac{1}{2}}(t)}{\Delta x}, \tag{9}$$

where the numerical fluxes $H_{j+\frac{1}{2}}$ are given by

$$H_{j+\frac{1}{2}}(t) := \frac{a^+_{j+\frac{1}{2}} f\left(u^-_{j+\frac{1}{2}}\right) - a^-_{j+\frac{1}{2}} f\left(u^+_{j+\frac{1}{2}}\right)}{a^+_{j+\frac{1}{2}} - a^-_{j+\frac{1}{2}}} + \frac{a^+_{j+\frac{1}{2}} a^-_{j+\frac{1}{2}}}{a^+_{j+\frac{1}{2}} - a^-_{j+\frac{1}{2}}}\left[u^+_{j+\frac{1}{2}} - u^-_{j+\frac{1}{2}}\right]. \tag{10}$$

Here, $u^+_{j+\frac{1}{2}} := p_{j+1}(x_{j+\frac{1}{2}}, t)$ and $u^-_{j+\frac{1}{2}} := p_j(x_{j+\frac{1}{2}}, t)$ stand for the corresponding right and left values of the piecewise polynomial interpolant $\{p_j\}$ at $x_{j+\frac{1}{2}}$. The one-sided local speeds $a^\pm_{j+\frac{1}{2}}$ are determined by

$$\begin{aligned} a^+_{j+\frac{1}{2}} &= \max\left\{\lambda_N\left(\frac{\partial f}{\partial u}\left(u^-_{j+\frac{1}{2}}\right)\right), \lambda_N\left(\frac{\partial f}{\partial u}\left(u^+_{j+\frac{1}{2}}\right)\right), 0\right\}, \\ a^-_{j+\frac{1}{2}} &= \min\left\{\lambda_1\left(\frac{\partial f}{\partial u}\left(u^-_{j+\frac{1}{2}}\right)\right), \lambda_1\left(\frac{\partial f}{\partial u}\left(u^+_{j+\frac{1}{2}}\right)\right), 0\right\}, \end{aligned} \tag{11}$$

with $\lambda_1 < \cdots < \lambda_N$ being the N eigenvalues of the Jacobian $\frac{\partial f}{\partial u}$.

Remarks.

1. The semidiscretization (9)–(11) results in a system of ODEs, which must be solved by a stable ODE solver.

2. The accuracy of the resulting fully discrete schemes is determined by the order of the piecewise polynomial reconstruction and the order of the ODE solver. In the multidimensional case, it also depends on the quadrature used in the derivation of the numerical fluxes (see [9, 10] for details).

For second-order schemes, a piecewise linear reconstruction is needed—a plethora of such (essentially) nonoscillatory reconstructions is available. To achieve third-order accuracy, a piecewise quadratic approximation is required. Building a third-order nonoscillatory reconstruction is a highly nontrivial problem. Such reconstructions, satisfying the number of extrema diminishing property in 1D, were proposed in [14, 15]. Their less dissipative modification and multidimensional extension, recently introduced in [10], have been used in our numerical examples.

The multidimensional central-upwind schemes are outlined in the Appendix.

3.2. Scheme Adaption

In this section, we describe the implementation of the SI in a scheme adaption algorithm, based on the third-order central-upwind scheme.

The SI (5) uses point values of the computed solution (rather than cell averages) and involves solution values at three consecutive time levels. We therefore assume that the reconstructed solution is available for the first two time levels.

3.2.1. *One-Dimensional Algorithm*

For the evaluation of the numerical fluxes (10), the flux functions are needed at cell interfaces. The evaluation of E_j^n, as given in (5), requires the fluxes at cell centers. To avoid additional flux function evaluations, we compute $E_{j+\frac{1}{2}}^n$ (instead of E_j^n) and use the flux functions already available at the cell interfaces, $\{f(u_{j\pm\frac{1}{2}}^{\pm})\}$. Two steps in the algorithm lend themselves to adaption. Away from rough regions, (i) the numerical flux reverts to a simple and computationally economical flux and (ii) the quadratic reconstruction reverts to a simple reconstruction with limiters switched off. We proceed as follows.

1. Compute the weak local truncation errors $\{E_{j+\frac{1}{2}}^{n-1}\}$ from (5). Note that the point values of the reconstruction are evaluated at cell interfaces $x = x_{j+\frac{1}{2}}$ (instead of $\{u_j^n\}$). Since the reconstruction is, in general, discontinuous at cell interfaces, two values $u_{j+\frac{1}{2}}^{\pm}$ are available. Either of them can be used. All indices j, for which

$$E_{j+\frac{1}{2}}^{n-1} > K(\Delta x)^3, \tag{12}$$

are marked for adaption (K is a constant to be selected).

2a. If cell interface $j+\frac{1}{2}$ is marked for adaption, then use the numerical flux

$$H_{j+\frac{1}{2}}(t) := \frac{a_{j+\frac{1}{2}}^+ f\left(u_{j+\frac{1}{2}}^-\right) - a_{j+\frac{1}{2}}^- f\left(u_{j+\frac{1}{2}}^+\right)}{a_{j+\frac{1}{2}}^+ - a_{j+\frac{1}{2}}^-} + \frac{a_{j+\frac{1}{2}}^+ a_{j+\frac{1}{2}}^-}{a_{j+\frac{1}{2}}^+ - a_{j+\frac{1}{2}}^-}\left[u_{j+\frac{1}{2}}^+ - u_{j+\frac{1}{2}}^-\right]. \tag{13a}$$

2b. Otherwise, use (see [8])

$$H_{j+\frac{1}{2}}(t) := \frac{1}{2}\Big[f\Big(u^+_{j+\frac{1}{2}}\Big) + f\Big(u^-_{j+\frac{1}{2}}\Big)\Big] - \frac{a_{j+\frac{1}{2}}}{2}\Big[u^+_{j+\frac{1}{2}} - u^-_{j+\frac{1}{2}}\Big]. \tag{13b}$$

Here, $a_{j+\frac{1}{2}}$ are the local speeds of propagation, which are the spectral radii of the corresponding Jacobians.

3. Solve the system of ODEs, (9), using a third-order ODE solver (to increase the efficiency, a multistep method is suggested).

4a. If any of the two indices $j \pm \frac{1}{2}$ is marked for adaption, use a piecewise quadratic reconstruction from $\{\bar{u}^{n+1}_j\}$, with the help of the nonlinear limiter in [10].

4b. Otherwise, switch the limiters off, and use the basic conservative parabolas [10, 15].

5. Go back to point 1.

Remarks.

1. The input parameter K in (12) determines the tolerance for scheme adaption. The larger the K, the tighter the rough region. Our numerical experiments suggest that the proposed method is not very sensitive to the choice of K.

2. A second-order version of this adaptive central-upwind scheme can be obtained if one uses a second-order piecewise linear reconstruction, and a second-order ODE solver.

3.2.2. *Two-Dimensional Algorithm*

Again, we assume that the reconstructed solution is available for the first two time levels and compute $E^n_{j+\frac{1}{2},k+\frac{1}{2}}$ (instead of $E^n_{j,k}$) in order to avoid additional flux function evaluations at cell centers.

1. Calculate the weak local truncation errors $\{E^{n-1}_{j+\frac{1}{2},k+\frac{1}{2}}\}$ from (7). Note that the point values of the reconstruction at the cell corners $(x_{j+\frac{1}{2}}, y_{k+\frac{1}{2}})$ are used (instead of $u^n_{j,k}$). In general, four different values, $u^{NE}_{j,k}$, $u^{NW}_{j+1,k}$, $u^{SE}_{j,k+1}$, and $u^{SW}_{j+1,k+1}$, are available (see (A.4)). Any of them can be used. All indices $(j+\frac{1}{2}, k+\frac{1}{2})$, for which

$$E^{n-1}_{j+\frac{1}{2},k+\frac{1}{2}} > K(\Delta x)^4, \tag{14}$$

are marked for adaption (K is a constant to be chosen).

2a. If any one of the indices $(j+\frac{1}{2}, k \pm \frac{1}{2})$ is marked for adaption, compute the x-flux $H^x_{j+\frac{1}{2},k}$ using (A.2). Similarly, if one of the indices $(j \pm \frac{1}{2}, k+\frac{1}{2})$ is marked, compute the y-flux $H^y_{j,k+\frac{1}{2}}$ using (A.3).

2b. Otherwise, use (see [10])

$$\begin{aligned} H^x_{j+\frac{1}{2},k}(t) = {} & \frac{f\big(u^{NW}_{j+1,k}\big) + f\big(u^{NE}_{j,k}\big) + 4\big(f\big(u^{W}_{j+1,k}\big) + f\big(u^{E}_{j,k}\big)\big) + f\big(u^{SW}_{j+1,k}\big) + f\big(u^{SE}_{j,k}\big)}{12} \\ & - \frac{a_{j+\frac{1}{2},k}}{12}\big[u^{NW}_{j+1,k} - u^{NE}_{j,k} + 4\big(u^{w}_{j+1,k} - u^{E}_{j,k}\big) + u^{SW}_{j+1,k} - u^{SE}_{j,k}\big], \end{aligned} \tag{15}$$

$$\begin{aligned} H^y_{j,k+\frac{1}{2}}(t) = {} & \frac{g\big(u^{SW}_{j,k+1}\big) + g\big(u^{NW}_{j,k}\big) + 4\big(g\big(u^{S}_{j,k+1}\big) + g\big(u^{N}_{j,k}\big)\big) + g\big(u^{SE}_{j,k+1}\big) + g\big(u^{NE}_{j,k}\big)}{12} \\ & - \frac{b_{j,k+\frac{1}{2}}}{12}\big[u^{SW}_{j,k+1} - u^{NW}_{j,k} + 4\big(u^{S}_{j,k+1} - u^{N}_{j,k}\big) + u^{SE}_{j,k+1} - u^{NE}_{j,k}\big], \end{aligned} \tag{16}$$

Here, $a_{j+\frac{1}{2},k}$ and $b_{j,k+\frac{1}{2}}$ are the local speeds of propagation in the x- and y-direction, which are the spectral radii of the corresponding Jacobians.

3. Solve the system of ODEs, (A.1), using a third-order ODE solver, preferably a multistep one.

4a. If any of the four indices, $(j \pm \frac{1}{2}, k \pm \frac{1}{2})$ is marked, reconstruct a piecewise quadratic interpolant from $\{\bar{u}_{j,k}^{n+1}\}$ with the help of the nonlinear limiter in [10].

4b. Otherwise, switch off the limiters and use the basic parabolas in [10].

5. Go back to point 1.

3.3. Mesh Adaption

We have implemented the smoothness indicator in a "baby" adaptive mesh refinement (AMR) algorithm. The goal here is primarily to demonstrate the smoothness indicator as a working concept in adaptive mesh calculations, rather than to embed it in a general purpose AMR code. We employ a simple two-grid strategy with only one level of refinement. The interpolation between coarse and fine grids, as well as the (conservative) course-fine mesh interface treatment, follows the general framework in [3, 20].

- The solution is reconstructed on the coarse mesh, and $(u_{coarse})_{j,k}^{n+1}$ is computed.
- If the cell is flagged for refinement, the coarse grid reconstruction is projected onto the fine grid cells by splitting it into appropriate pieces.
- In cells adjacent to a refined mesh, Hermite time interpolation is used to generate solution values at $(u_{fine})_{j,k}^{n+\frac{l}{r}}, l = 1 \ldots r$ (here r is the refinement ratio), so that numerical flux functions at coarse–fine grid interfaces may be calculated.
- The fine grid solution $(u_{fine})_{j,k}^{n+1}$ is computed.
- The coarse grid solution $(u_{coarse})_{j,k}^{n+1}$ is corrected by the difference between (the sum of) the fine grid fluxes and the (single) coarse grid flux, to insure conservation.
- The fine grid solution is projected back onto the coarse grid by simple averaging.

We note that while we recognize that extending the existing code to a general purpose AMR code constitutes a major undertaking, we feel that a simple two-grid strategy is sufficient to demonstrate the viability of the smoothness indicator in AMR algorithms.

4. NUMERICAL RESULTS

In Sections 4.1 and 4.2, we present numerical solutions for the gas dynamics equations for ideal gases,

$$\frac{\partial}{\partial t}\begin{bmatrix}\rho\\ \rho u\\ \rho v\\ E\end{bmatrix} + \frac{\partial}{\partial x}\begin{bmatrix}\rho u\\ \rho u^2 + p\\ \rho uv\\ u(E+p)\end{bmatrix} + \frac{\partial}{\partial y}\begin{bmatrix}\rho v\\ \rho uv\\ \rho v^2 + p\\ v(E+p)\end{bmatrix} = 0, \quad p = (\gamma - 1)\left[E - \frac{\rho}{2}(u^2 + v^2)\right]. \tag{17}$$

Here ρ, u, v, p, and E are the density, the x- and y-velocities, the pressure, and the total energy, respectively. The obtained results demonstrate that the proposed smoothness indicator captures faithfully all rough solution regions, and that adaptive algorithms based on

the smoothness indicator not only increase the efficiency of the computation but may also yield improved computed solutions.

In the examples below, the semidiscrete central-upwind scheme from Section 3.1 was used, with a Runge–Kutta ODE solver for the first two time steps and an explicit Adams method afterward. The smoothness indicator was based on the density field only, since the other components of the solution, ρu, ρv, and E, are smooth wherever ρ is smooth.

4.1. One-Dimensional Euler Equations of Gas Dynamics

In one space dimension, we compute the solution of (17) with the initial data

$$(\rho, u, p)(x, 0) = \begin{cases} (3.857143, -0.920279, 10.33333), & x < 0, \\ (1 + \varepsilon \sin(5x), -3.549648, 1.00000), & 0 < x < 10, \\ (1.000000, -3.549648, 1.00000), & x \geq 10, \end{cases}$$

corresponding to a density perturbation running leftward into a stationary shock of Mach number $M_s = 3$. This calculation has the numerical "flavor" of an acoustic wave, propagating through a steady discontinuous flow field. We take $\varepsilon = 0.2$, $\Delta t/\Delta x = 0.02$, and present the solution at time $T = 2$.

We first compute the solution using the second-order central-upwind scheme, (9) and (10), with the standard piecewise linear minmod reconstruction (see, e.g., [13, 17]),

$$p_j(x) = \bar{u}_j + s_j(x - x_j), \quad s_j = \text{minmod}\left(\frac{\bar{u}_{j+1} - \bar{u}_j}{\Delta x}, \frac{\bar{u}_j - \bar{u}_{j-1}}{\Delta x}\right).$$

This limiter is rather dissipative and leads to excessive numerical damping, mostly noticeable in the postshock phase of the perturbation.

We then apply an adaptive version of the same method, where this time the reconstruction in the smooth areas does not employ any limiters, and it is

$$p_j(x) = \bar{u}_j + \frac{\bar{u}_{j+1} - \bar{u}_{j-1}}{2\Delta x}(x - x_j).$$

Both solutions are obtained on a 1600-point mesh and compared with a fine grid reference solution using 12,800 points. The effect of removing the limiters in the reconstruction step is clearly seen in Figs. 1a and 1b, obtained with $K = 16$ and 8, respectively. The minmod

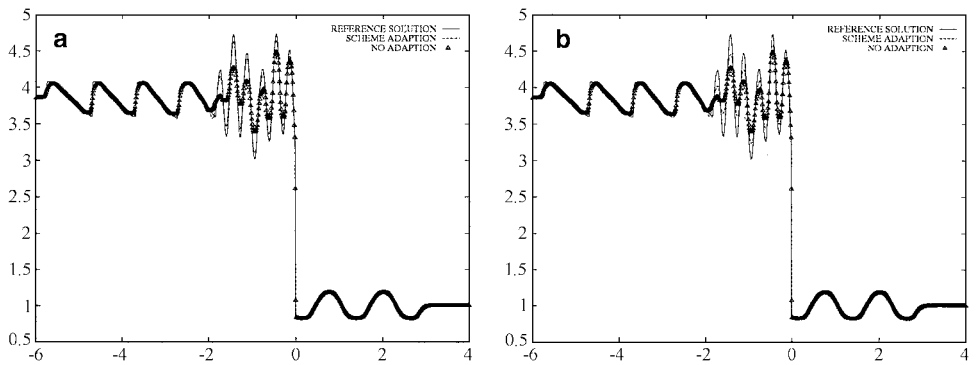

FIG. 1. Density, $K = 16$ (a) and $K = 8$ (b).

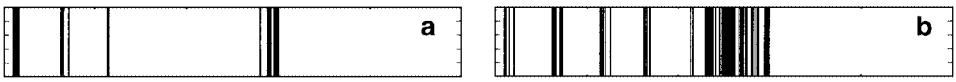

FIG. 2. Nonsmooth areas, $K = 16$ (a) and $K = 8$ (b).

limited solution exhibits excessive dissipation in the postshock region and has difficulties maintaining the amplitude of the perturbation. Removing the limiter leads to significantly improved resolution in the smooth postshock region. We observe that as the perturbation travels downstream, the wave fronts steepen to form shock waves which are detected by the smoothness indicator and are flagged for the use of limiters. Notice that the larger the value of K, the narrower the region in which limiters are being used. This results in reduced dissipation in the smooth region. Figures 2a and 2b show the areas that are flagged as rough. Only in those areas was the minmod limiter applied. The corresponding LTEs are presented in Figs. 3a and 3b, where we observe the sensitivity of the smoothness indicator in identifying the left running shock fronts as they form.

4.2. Two-Dimensional Euler Equations of Gas Dynamics

The 2D Riemann problem for (17) corresponds to initial data of the general form

$$(p, \rho, u, v)(x, y, 0) = \begin{cases} (p_1, \rho_1, u_1, v_1), & \text{if } x > 0.5 \text{ and } y > 0.5, \\ (p_2, \rho_2, u_2, v_2), & \text{if } x < 0.5 \text{ and } y > 0.5, \\ (p_3, \rho_3, u_3, v_3), & \text{if } x < 0.5 \text{ and } y < 0.5, \\ (p_4, \rho_4, u_4, v_4), & \text{if } x > 0.5 \text{ and } y < 0.5, \end{cases} \tag{18}$$

It admits 19 genuinely different configurations for polytropic gas [22], distinguished by the three types of 1D waves between each two neighboring states, namely rarefaction ($\vec{R}$), shock ($\vec{S}$), and contact waves ($\vec{J}$) (consult [21, 22] for details).

4.2.1. *Scheme Adaption Results*

Below are the results for four different wave configurations, numbered Configuration 4, 8, and 19, following [22]. The computations use the third-order central-upwind scheme,

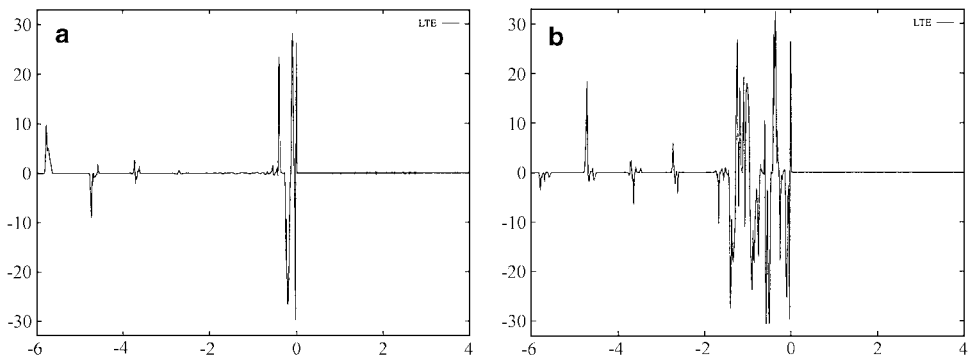

FIG. 3. LTE, $K = 16$ (a) and $K = 8$ (b).

TABLE I
CPU Times

Conf. no.	Final time	$\frac{\Delta t}{\Delta x}$	Adaptive scheme $K=1$	Adaptive scheme $K=2$	No adaption
4	0.25	0.05	3:45:50	3:46:43	6:23:44
8	0.25	0.08	2:23:54	2:22:15	4:09:02
10	0.15	0.05	2:16:17	2:17:14	3:51:52
19	0.30	0.10	2:20:05	2:15:26	3:55:31

described in the Appendix, with and without the scheme adaption algorithm from Section 3.2. The computational domain is $[0, 1] \times [0, 1]$, where we use a 400×400 grid. In the figures below, we show density contour lines as well as the regions that were flagged as rough regions for scheme adaption.

The advantage of the adaptive strategy is two fold. The resulting scheme is both cheaper (by more than 40% for the problems shown, see Table I) and the solution is better resolved, due to switching off the limiters where not needed. Also, notice that the method is not too sensitive to the choice of the smoothness indicator constant K, and that no noise is generated due to the switching between the two numerical procedures.

Configuration 4 (Figs. 4–6). $\overrightarrow{S_{32}}\ \overset{\overleftarrow{S_{21}}}{\underset{\overrightarrow{S_{34}}}{}}\ \overleftarrow{S_{41}}$: The initial data are

$$\begin{array}{llll} p_2 = 0.35 & \rho_2 = 0.5065 & p_1 = 1.1 & \rho_1 = 1.1 \\ u_2 = 0.8939 & v_2 = 0 & u_1 = 0 & v_1 = 0 \\ p_3 = 1.1 & \rho_3 = 1.1 & p_4 = 0.35 & \rho_4 = 0.5065 \\ u_3 = 0.8939 & v_3 = 0.8939 & u_4 = 0 & v_4 = 0.8939 \end{array}$$

Configuration 8 (Figs. 7–9). $J_{32}^-\ \overset{\overleftarrow{R_{21}}}{\underset{J_{34}^-}{}}\ \overleftarrow{R_{41}}$: The initial data are

$$\begin{array}{llll} p_2 = 1 & \rho_2 = 1 & p_1 = 0.4 & \rho_1 = 0.5197 \\ u_2 = -0.6259 & v_2 = 0.1 & u_1 = 0.1 & v_1 = 0.1 \\ p_3 = 1 & \rho_3 = 0.8 & p_4 = 1 & \rho_4 = 1 \\ u_3 = 0.1 & v_3 = 0.1 & u_4 = 0.1 & v_4 = -0.6259 \end{array}$$

Configuration 19 (Figs. 10 and 11). $\overleftarrow{S_{32}}\ \overset{J_{21}^+}{\underset{J_{34}^-}{}}\ \overrightarrow{R_{41}}$: The initial data are

$$\begin{array}{llll} p_2 = 1 & \rho_2 = 2 & p_1 = 1 & \rho_1 = 1 \\ u_2 = 0 & v_2 = -0.3 & u_1 = 0 & v_1 = 0.3 \\ p_3 = 0.4 & \rho_3 = 1.0625 & p_4 = 0.4 & \rho_4 = 0.5197 \\ u_3 = 0 & v_3 = 0.2145 & u_4 = 0 & v_4 = -0.4259 \end{array}$$

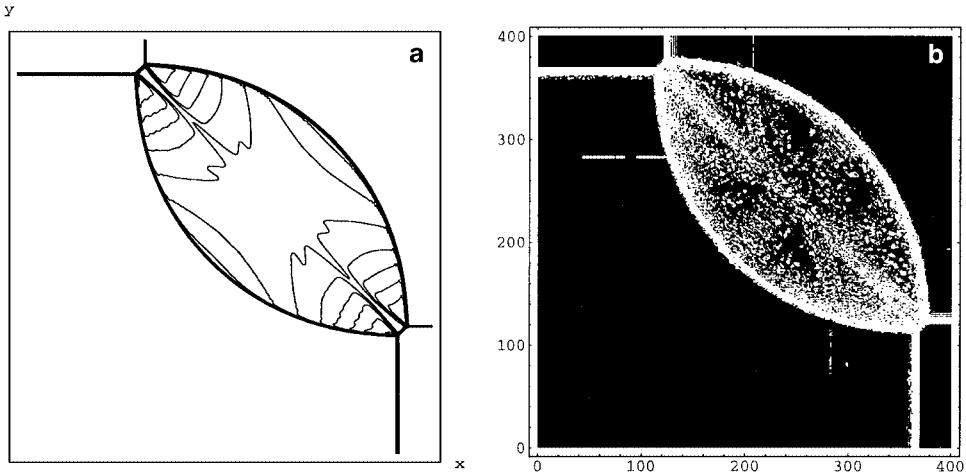

FIG. 4. $K = 1$ adaption (a) and nonsmooth areas (b).

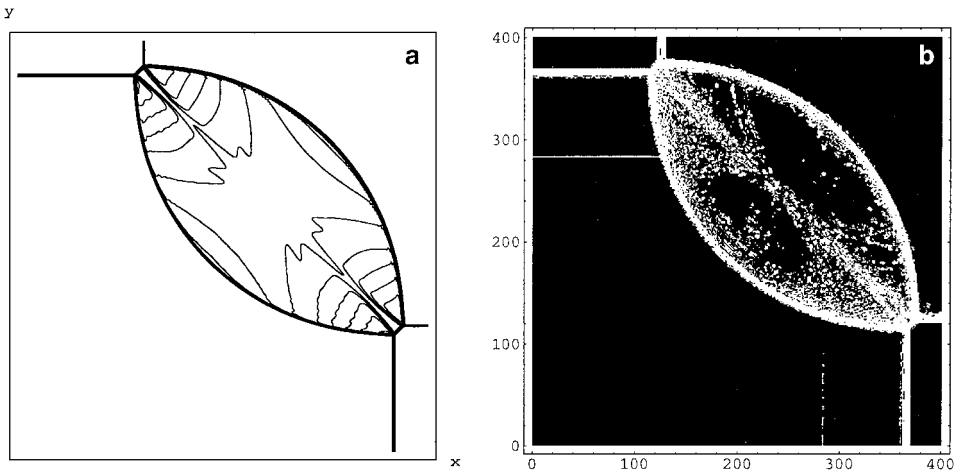

FIG. 5. $K = 2$ adaption (a) and nonsmooth areas (b).

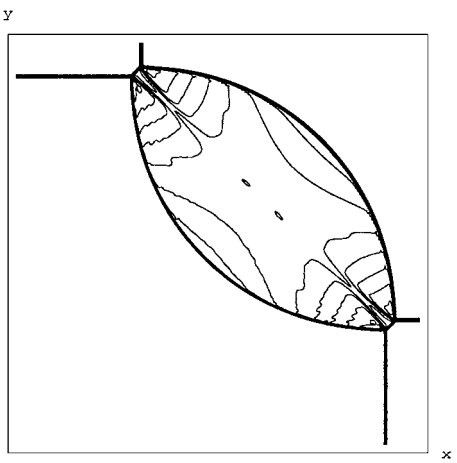

FIG. 6. No adaption.

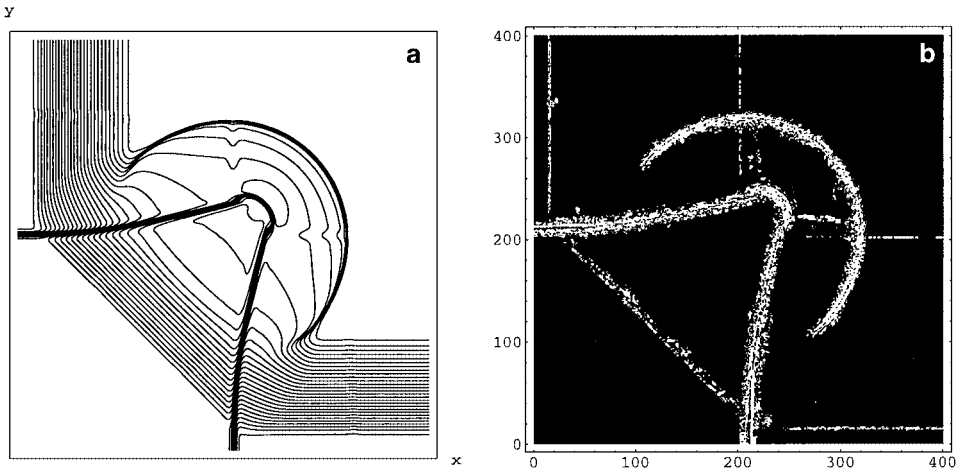

FIG. 7. $K = 1$ adaption (a) and nonsmooth areas (b).

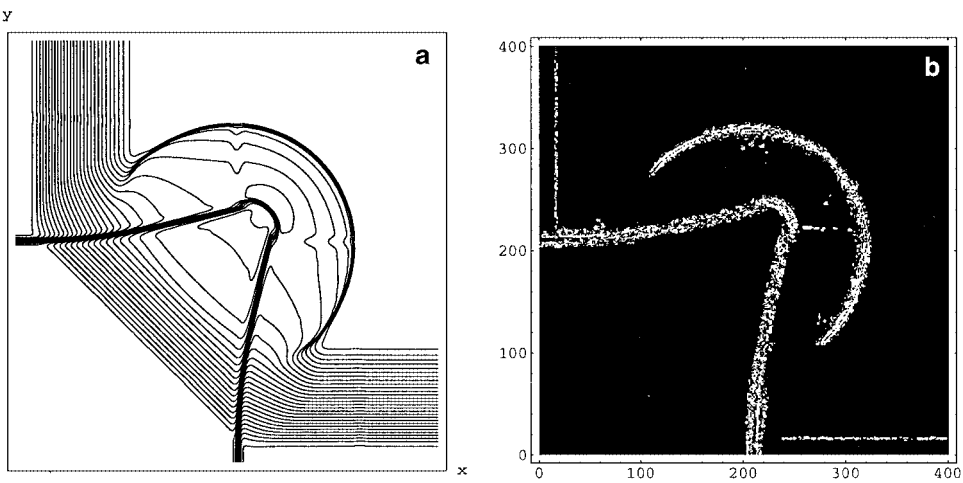

FIG. 8. $K = 2$ adaption (a) and nonsmooth areas (b).

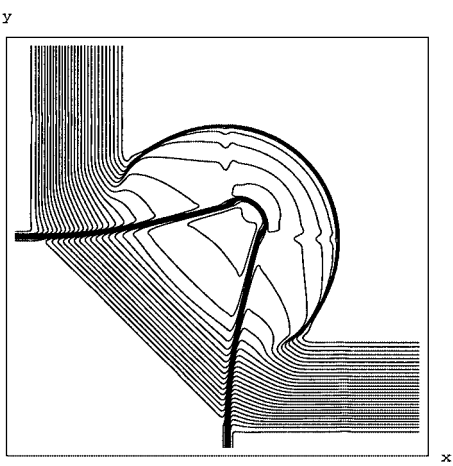

FIG. 9. No adaption.

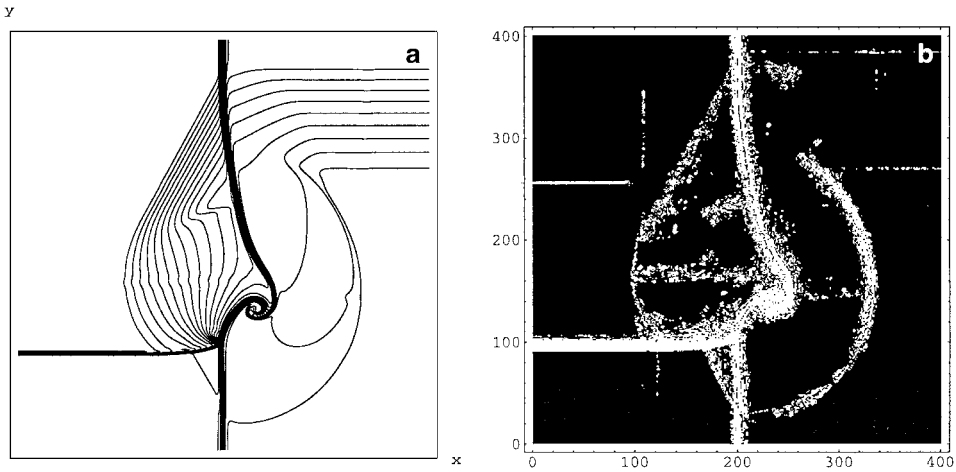

FIG. 10. $K = 1$ adaption (a) and nonsmooth areas (b).

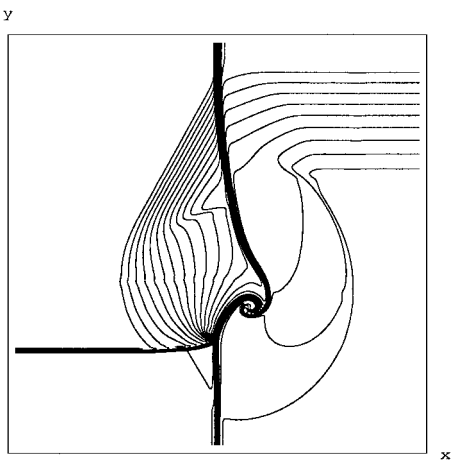

FIG. 11. No adaption.

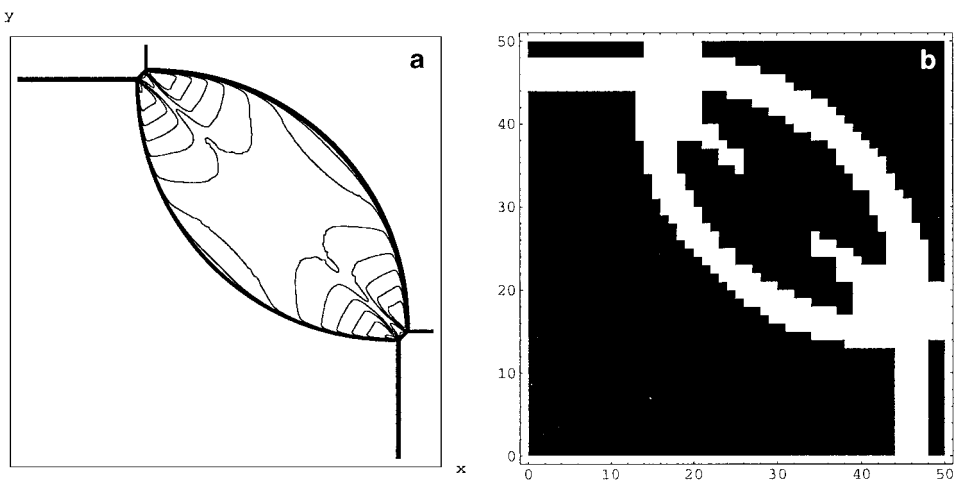

FIG. 12. (a) Grid ratio 8 : 1; (b) nonsmooth areas, $K = 1$.

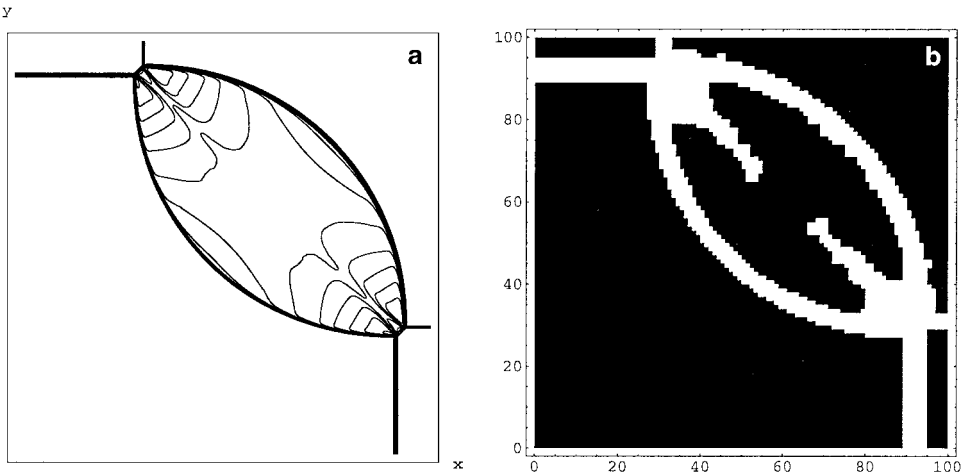

FIG. 13. (a) Grid ratio 4 : 1; (b) nonsmooth areas, $K = 1$.

4.2.2. *Mesh and Scheme Adaption Results*

The 2D gas dynamics equations are solved numerically for the Riemann problem given by Configuration 4, Section 4.2.1. In this computation, the smoothness indicator is used both for mesh adaption and for scheme adaption. The third-order central-upwind scheme is applied in the smooth regions, and its second-order version [12] (with the minmod limiter) is used in the indicated nonsmooth areas. In Figs. 12a and 13a, the solution is obtained on a 50×50 mesh with a refinement ratio of 8, and on a 100×100 mesh with a refinement ratio of 4, respectively. Note that both are effectively equivalent to a 400×400 mesh in the nonsmooth areas. The regions, flagged by the smoothness indicator, are shown in Figs. 12b and 13b. We also note the clean transition of the computed solution across the coarse–fine mesh interfaces.

As expected, the CPU times for the mesh adaption algorithm are significantly lower than that of the equivalent uniform mesh, up to a factor of 5 for the problem shown. This speedup factor should be taken as a conservative estimate. Additional speedup may result from further optimization of the code.

APPENDIX: 2D CENTRAL-UPWIND SCHEME

Here, we consider multidimensional systems of conservation laws. Without loss of generality, we restrict ourselves to the case $d = 2$,

$$u_t + f(u)_x + g(u)_y = 0.$$

Then cell averages at time t,

$$\bar{u}_{j,k}(t) := \frac{1}{\Delta x \Delta y} \int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} \int_{y_{k-\frac{1}{2}}}^{y_{k+\frac{1}{2}}} u(x, y, t)\, dx\, dy,$$

are evolved using the genuinely multidimensional semidiscrete central-upwind scheme (see [9])

$$\frac{d}{dt}\bar{u}_{j,k}(t) = -\frac{H^x_{j+\frac{1}{2},k}(t) - H^x_{j-\frac{1}{2},k}(t)}{\Delta x} - \frac{H^y_{j,k+\frac{1}{2}}(t) - H^y_{j,k-\frac{1}{2}}(t)}{\Delta y}. \tag{A.1}$$

The fourth-order 2D numerical fluxes are given by

$$\begin{aligned} H^x_{j+\frac{1}{2},k} :=\ & \frac{a^+_{j+\frac{1}{2},k}}{6\left(a^+_{j+\frac{1}{2},k} - a^-_{j+\frac{1}{2},k}\right)}\left[f\left(u^{NE}_{j,k}\right) + 4f\left(u^{E}_{j,k}\right) + f\left(u^{SE}_{j,k}\right)\right] \\ & - \frac{a^-_{j+\frac{1}{2},k}}{6\left(a^+_{j+\frac{1}{2},k} - a^-_{j+\frac{1}{2},k}\right)}\left[f\left(u^{NW}_{j+1,k}\right) + 4f\left(u^{W}_{j+1,k}\right) + f\left(u^{SW}_{j+1,k}\right)\right] \\ & + \frac{a^+_{j+\frac{1}{2},k}a^-_{j+\frac{1}{2},k}}{6\left(a^+_{j+\frac{1}{2},k} - a^-_{j+\frac{1}{2},k}\right)}\left[u^{NW}_{j+1,k} - u^{NE}_{j,k} + 4\left(u^{W}_{j+1,k} - u^{E}_{j,k}\right) + u^{SW}_{j+1,k} - u^{SE}_{j,k}\right], \end{aligned} \tag{A.2}$$

and

$$\begin{aligned} H^y_{j,k+\frac{1}{2}} :=\ & \frac{b^+_{j,k+\frac{1}{2}}}{6\left(b^+_{j,k+\frac{1}{2}} - b^-_{j,k+\frac{1}{2}}\right)}\left[g\left(u^{NW}_{j,k}\right) + 4g\left(u^{N}_{j,k}\right) + g\left(u^{NE}_{j,k}\right)\right] \\ & - \frac{b^-_{j,k+\frac{1}{2}}}{6\left(b^+_{j,k+\frac{1}{2}} - b^-_{j,k+\frac{1}{2}}\right)}\left[g\left(u^{SW}_{j,k+1}\right) + 4g\left(u^{S}_{j,k+1}\right) + g\left(u^{SE}_{j,k+1}\right)\right] \\ & + \frac{b^+_{j,k+\frac{1}{2}}b^-_{j,k+\frac{1}{2}}}{6\left(b^+_{j,k+\frac{1}{2}} - b^-_{j,k+\frac{1}{2}}\right)}\left[u^{SW}_{j,k+1} - u^{NW}_{j,k} + 4\left(u^{S}_{j,k+1} - u^{N}_{j,k}\right) + u^{SE}_{j,k+1} - u^{NE}_{j,k}\right], \end{aligned} \tag{A.3}$$

and the corresponding point values are

$$\begin{gathered} u^N_{j,k} := p_{j,k}\left(x_j, y_{k+\frac{1}{2}}, t\right), \quad u^S_{j,k} := p_{j,k}\left(x_j, y_{k-\frac{1}{2}}, t\right), \\ u^E_{j,k} := p_{j,k}\left(x_{j+\frac{1}{2}}, y_k, t\right), \quad u^W_{j,k} := p_{j,k}\left(x_{j-\frac{1}{2}}, y_k, t\right), \quad u^{NE}_{j,k} := p_{j,k}\left(x_{j+\frac{1}{2}}, y_{k+\frac{1}{2}}, t\right), \\ u^{NW}_{j,k} := p_{j,k}\left(x_{j-\frac{1}{2}}, y_{k+\frac{1}{2}}, t\right), \quad u^{SE}_{j,k} := p_{j,k}\left(x_{j+\frac{1}{2}}, y_{k-\frac{1}{2}}, t\right), \quad u^{SW}_{j,k} := p_{j,k}\left(x_{j-\frac{1}{2}}, y_{k-\frac{1}{2}}, t\right). \end{gathered} \tag{A.4}$$

As in the 1D case, these values are computed from a nonoscillatory piecewise polynomial reconstruction,

$$\tilde{u}(x, y, t) := \sum_{j,k} p_{j,k}(x, y, t)\chi_{j,k}(x, y), \tag{A.5}$$

where $\chi_{j,k}$ is the characteristic function of the cell $[x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}}] \times [y_{k-\frac{1}{2}}, y_{k+\frac{1}{2}}]$. For examples of such reconstructions, consult [10] and the references therein. The one-sided local

speeds of propagation $a^{\pm}_{j+\frac{1}{2},k}$, $b^{\pm}_{j,k+\frac{1}{2}}$ are calculated via

$$
\begin{aligned}
a^{+}_{j+\frac{1}{2},k} &:= \max\left\{\lambda_N\left(\frac{\partial f}{\partial u}\left(u^{W}_{j+1,k}\right)\right), \lambda_N\left(\frac{\partial f}{\partial u}\left(u^{E}_{j,k}\right)\right), 0\right\},\\
b^{+}_{j,k+\frac{1}{2}} &:= \max\left\{\lambda_N\left(\frac{\partial g}{\partial u}\left(u^{S}_{j,k+1}\right)\right), \lambda_N\left(\frac{\partial g}{\partial u}\left(u^{N}_{j,k}\right)\right), 0\right\},\\
a^{-}_{j+\frac{1}{2},k} &:= \min\left\{\lambda_1\left(\frac{\partial f}{\partial u}\left(u^{W}_{j+1,k}\right)\right), \lambda_1\left(\frac{\partial f}{\partial u}\left(u^{E}_{j,k}\right)\right), 0\right\},\\
b^{-}_{j,k+\frac{1}{2}} &:= \min\left\{\lambda_1\left(\frac{\partial g}{\partial u}\left(u^{S}_{j,k+1}\right)\right), \lambda_1\left(\frac{\partial g}{\partial u}\left(u^{N}_{j,k}\right)\right), 0\right\},
\end{aligned}
\tag{A.6}
$$

with $\lambda_1(\cdot) < \cdots < \lambda_N(\cdot)$ the eigenvalues of $\frac{\partial f}{\partial u}$ or $\frac{\partial g}{\partial u}$.

ACKNOWLEDGMENTS

The work of S. Karni was supported in part by NSF Grant DMS-9973291, and by ONR Award N00014-99-1-0449. A. Kurganov was supported in part by the NSF Group Infrastructure Grant, and by NSF Grant DMS-0073631. G. Petrova was supported in part by NSF Grant DMS-0104112.

REFERENCES

1. F. Aràndiga and R. Donat, Nonlinear multiscale decompositions: the approach of A. Harten, *Numer. Algorithms* **23**, 175 (2000).
2. F. Aràndiga, R. Donat, and A. Harten, Multiresolution based on weighted averages of the hat function. II. Nonlinear reconstruction techniques, *SIAM J. Sci. Comput.* **20**, 1053 (1999).
3. M. J. Berger and P. Colella, Local adaptive mesh refinement for shock hydrodynamics, *J. Comput. Phys.* **82**, 67 (1989).
4. M. J. Berger and J. Oliger, Adaptive mesh refinement for hyperbolic partial differential equations, *J. Comput. Phys.* **53**, 482 (1984).
5. C. de Boor, *A Practical Guide to Splines*, Applied Mathematical Sciences (Springer-Verlag, New York, 1978), vol. 27.
6. E. Harabetian and R. Pego, Nonconservative hybrid shock capturing schemes, *J. Comput. Phys.* **105**, 1 (1993).
7. S. Karni and A. Kurganov, Local error analysis for approximate solutions of hyperbolic conservation laws, submitted for publication.
8. A. Kurganov and D. Levy, A third-order semi-discrete central scheme for conservation laws and convection-diffusion equations, *SIAM J. Sci. Comput.* **22**, 1461 (2000).
9. A. Kurganov, S. Noelle, and G. Petrova, Semi-discrete central-upwind scheme for hyperbolic conservation laws and Hamilton-Jacobi equations, *SIAM J. Sci. Comput.* **23**, 707 (2001).
10. A. Kurganov and G. Petrova, A third-order semi-discrete genuinely multidimensional central scheme for hyperbolic conservation laws and related problems, *Numer. Math.* **88**, 683 (2001).
11. A. Kurganov and E. Tadmor, New high-resolution central schemes for nonlinear conservation laws and convection-diffusion equations, *J. Comput. Phys.* **160**, 241 (2000).
12. A. Kurganov and E. Tadmor, Solution of two-dimensional Riemann problems for gas dynamics without Riemann problem solvers, submitted for publication.
13. B. van Leer, Towards the ultimate conservative difference scheme. V. A second order sequel to Godunov's method, *J. Comput. Phys.* **32**, 101 (1979).
14. X.-D. Liu and S. Osher, Nonoscillatory high order accurate self similar maximum principle satisfying shock capturing schemes, I, *SIAM J. Numer. Anal.* **33**, 760 (1996).

15. X.-D. Liu and E. Tadmor, Third order nonoscillatory central scheme for hyperbolic conservation laws, *Numer. Math.* **79**, 397 (1998).
16. M. L. Minion, A projection method for locally refined grids, *J. Comput. Phys.* **127**, 158 (1996).
17. H. Nessyahu and E. Tadmor, Non-oscillatory central differencing for hyperbolic conservation laws, *J. Comput. Phys.* **87**, 408 (1990).
18. H. Nessyahu and E. Tadmor, The convergence rate of approximate solutions for nonlinear scalar conservation laws, *SIAM J. Numer. Anal.* **29**, 1505 (1992).
19. H. Nessyahu, E. Tadmor, and T. Tassa, The convergence rate of Godunov type schemes, *SIAM J. Numer. Anal.* **31**, 1 (1994).
20. J. J. Quirk, An Adaptive mesh refinement algorithm for computational shock hydrodynamics, Ph.D. thesis (Cranfield Institute of Technology, UK, 1991).
21. C. W. Schulz-Rinne, Classification of the Riemann problem for two-dimensional gas dynamics, *SIAM J. Math. Anal.* **24**, 76 (1993).
22. C. W. Schulz-Rinne, J. P. Collins, and H. M. Glaz, Numerical solution of the Riemann problem for two-dimensional gas dynamics, *SIAM J. Sci. Comput.* **14**, 1394 (1993).
23. E. Tadmor, Local error estimates for discontinuous solutions of nonlinear hyperbolic equations, *SIAM J. Numer. Anal.* **28**, 891 (1991).
24. E. Tadmor and T. Tang, Pointwise error estimates for scalar conservation laws with piecewise smooth solutions, *SIAM J. Numer. Anal.* **36**, 1739 (1999).
25. E. F. Toro, Primitive, conservative and adaptive schemes for hyperbolic conservation laws, in *Numerical Methods for Wave Propagation*, edited by E. F. Toro and J. F. Clarke (Kluwer Academic, Dordrecht/Norwell, MA, 1998).