COMPUTATIONAL FLUID DYNAMICS WITH MESH ADAPTIVITY *

TAO TANG † AND HUAZHONG TANG ‡

Abstract. In this work we demonstrate some recent progress on moving mesh methods with application to computational fluid dynamics. The emphasis will be on the application to the gas dynamics governed by the hyperbolic conservation laws. Several test problems in one- and two-dimensions are computed by using the moving mesh methods. The computations demonstrate that the moving mesh methods are efficient for solving problems with shock discontinuities, obtaining the same resolution with much less number of grid points than the uniform mesh approach.

1. Introduction. Adaptive mesh methods have important applications for a variety of physical and engineering areas such as solid and fluid dynamics, combustion, heat transfer, material science etc. The physical phenomena in these areas develop dynamically singular or nearly singular solutions in fairly localized regions, such as shock waves, boundary layers, detonation waves etc. The numerical investigation of these physical problems may require extremely fine meshes over a small portion of the physical domain to resolve the large solution variations. In multi-dimensions, developing effective and robust adaptive grid methods for these problems becomes necessary. Successful implementation of the adaptive strategy can increase the accuracy of the numerical approximations and also decrease the computational cost. In the past two decades, there have seen many important progress in moving mesh methods for partial differential equations, including the variational approach of Winslow [22], Brackbill et al. [2, 3], and Ren and Wang [16]; finite element methods of Millers [15]; moving mesh PDEs of Russell et al. [4, 20], Li and Petzold [11], and Ceniceros and Hou [5]; and moving mesh methods based on harmonic mapping of Dvinsky [8] and Li et al. [12, 13].

In this work we demonstrate some recent progress on the moving mesh methods with application to computational fluid dynamics. The emphasis will be on the application to the gas dynamics governed by the hyperbolic conservation laws. Harten and Hyman [9] began the earliest study in this direction, by moving the grid at an adaptive speed in each time step to improve the resolution of shocks and contact discontinuities. After their work, many other moving mesh methods for hyperbolic problems have been proposed in the literature, including Azarenok and Ivanenko [1], Liu et al. [14], Saleri and Steinberg [17], and Stockie et al. [20]. However, it is noticed that many existing moving mesh methods for hyperbolic problems are designed for one space dimension. In 1D, it is generally possible to compute on a very fine grid and so the need for moving mesh methods may not be clear. Multidimensional moving mesh methods are often difficult to use in fluid dynamics problems since the grid will typically suffer large distortions and possible tangling. It is therefore useful to design a simple and robust moving mesh algorithm for hyperbolic problems in multi-dimensions.

Following Li et al. [12] we propose a moving mesh method containing two separate parts: PDE time-evolution and mesh-redistribution. The first part can be any suitable high-resolution

methods such as the wave-propagation algorithm, central schemes, and ENO methods. Once numerical solutions are obtained at the given time level, the meshes will be redistributed based on an iteration procedure. At each iteration, the grids are moved based on a variational principle and the underlying numerical solutions at the new grids will be updated by using some simple methods (such as the conventional interpolation). It is noted that the direct use of the conventional interpolation is unsatisfactory for hyperbolic problems, since many physical properties such as mass-conservation and TVD (in 1D) may be destroyed. In order to preserve these physical properties, we propose to use conservative-interpolations in the solution-updating step. This approach also preserves the total mass of the numerical solutions, and by the well-known Lax-Wendroff theory the numerical solutions converge to the weak solution of the underlying hyperbolic system [21].

2. Mesh generation based on the variational approach. Let $\vec{x} = (x_1, x_2, \cdots, x_d)$ and $\vec{\xi} = (\xi_1, \xi_2, \cdots, \xi_d)$ denote the physical and computational coordinates, respectively. Here $d \geq 1$ denotes the number of spatial dimension. A one–to–one coordinate transformation from the computational (or logical) domain Ω_c to the physical domain Ω_p is denoted by

$$(2.1) \qquad \vec{x} = \vec{x}(\vec{\xi}), \quad \vec{\xi} \in \Omega_c.$$

Its inversion is denoted by

$$(2.2) \qquad \vec{\xi} = \vec{\xi}(\vec{x}), \quad \vec{x} \in \Omega_p.$$

In the variational approach, the mesh map is provided by the minimizer of a functional of the following form:

$$(2.3) \qquad E(\vec{\xi}) = \frac{1}{2} \sum_k \int_{\Omega_p} \nabla \xi_k^T G_k^{-1} \nabla \xi_k d\vec{x} \ ,$$

where $\nabla := (\partial_{x_1}, \partial_{x_2}, \cdots, \partial_{x_d})^T$, G_k are given symmetric positive definite matrices called *monitor functions*. In general, the monitor functions depend on the underlying solution to be adapted. More terms can be added to the functional (2.3) to control other aspects of the mesh such as orthogonality and mesh alignment with a given vector field [2, 3].

The variational mesh is determined by the Euler-Lagrange equation of the above functional:

$$(2.4) \qquad \nabla \cdot \left(G_k^{-1} \nabla \xi_k \right) = 0, \qquad 1 \leq k \leq d.$$

One of the simplest choices of the monitor functions is $G_k = \omega I$, $1 \leq k \leq d$, where I is the identity matrix and ω is a positive weight function, e.g., $\omega = \sqrt{1 + |\nabla u|^2}$. Here u is the solution of the underlying PDE. In this case, we obtain Winslow's variable diffusion method [22]:

$$(2.5) \qquad \nabla \cdot \left(\frac{1}{\omega} \nabla \xi_k \right) = 0, \qquad 1 \leq k \leq d.$$

By using the above equations, a map between the physical domain Ω_p and the logical domain Ω_c can be computed. Typically, the map transforms a uniform mesh in the logical domain to cluster grid points at the regions of the physical domain where the solution has the largest gradients.

Our solution procedure is based on two independent parts: a mesh-redistribution algorithm and a solution algorithm. The first part will be based on an iteration procedure to redistribute the mesh points and to interpolate the solution on the underlying new grid points. The second part will be independent of the first one, which can be any of the standard codes for solving the given PDEs, e.g. finite volume schemes. The solution procedure can be illustrated by the following flowchart:

Algorithm 0.

Step 1: Given a uniform (fixed) partition of the logical domain Ω_c, and use Eq. (2.5) to generate an initial partition $x_j^{[0]} := x_j$ of the physical domain Ω_p. Then compute the grid values $u_{j+\frac{1}{2}}^{[0]}$ based on the cell average for the initial data $u(x,0)$.

$\Downarrow$

Step 2: Move grid $\{x_j^{[\nu]}\}$ to $\{x_j^{[\nu+1]}\}$ by solving (2.5) with a couple of Gauss-Seidel iterations. Then compute $\{u_{j+\frac{1}{2}}^{[\nu+1]}\}$ on the new grid based on a conservative interpolation scheme. Repeat this updating procedure for a fixed number of iterations or until $\|x^{[\nu+1]} - x^{[\nu]}\|$ is sufficiently small.

$\Downarrow$

Step 3: Evolve the underlying PDEs by using high-resolution finite volume methods on the mesh $\{x_j^{[\nu+1]}\}$ to obtain the numerical approximations $u_{j+\frac{1}{2}}^{n+1}$ at the time level t_{n+1}.

$\Downarrow$

Step 4: If $t_{n+1} \leq T$, then let $u_{j+\frac{1}{2}}^{[0]} := u_{j+\frac{1}{2}}^{n+1}$ and $x_j^{[0]} := x_j^{[\nu+1]}$ and go to **Step 2**.

The detail descriptions on Steps 2 and 3, in particular the conservative interpolation formulas can be found in Tang and Tang [21].

3. Numerical experiments. EXAMPLE 3.1. Sod's problem. In this example, we test our adaptive mesh algorithm to the one-dimensional Euler equations of gas dynamics

$$(3.1) \qquad \begin{bmatrix} \rho \\ \rho u \\ E \end{bmatrix}_t + \begin{bmatrix} \rho u \\ \rho u^2 + p \\ u(E + p) \end{bmatrix}_x = 0,$$

where ρ, u, p, and E are density, velocity, pressure, and the total energy, respectively. The above system is closed by the equation of state, $p = (\gamma - 1)(E - \rho u^2/2)$. The initial data is chosen as

$$(\rho, \rho u, E) = \begin{cases} (1, 0, 2.5), & \text{if } x < 0.5, \\ (0.125, 0, 0.25), & \text{if } x > 0.5. \end{cases}$$

This is a well-known test problem proposed by Sod [19]. The monitor function in (2.5) used is

$$(3.2) \qquad \omega = \sqrt{1 + \alpha_1 \left(\frac{u_\xi}{\max_\xi |u_\xi|}\right)^2 + \alpha_2 \left(\frac{s_\xi}{\max_\xi |s_\xi|}\right)^2}$$

where $s = p/\rho^\gamma$, and the parameters α_i ($i = 1, 2$) are some non-negative constants. The above monitor function was suggested by Stockie et al. [20] who also discussed several other choices for the monitor function. The numerical results are obtained with $J = 100$, $\alpha_1 = 20, \alpha_2 = 100$ and are presented in Figure 3.1. It is found that the contact and shock discontinuities are well resolved, although quite a number of grid points are also moved to the rarefaction wave region.

EXAMPLE 3.2. A 2D Riemann problem. Two-dimensional Euler equations of gas dynamics can be written as

$$(3.3) \qquad \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ E \end{bmatrix}_t + \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho u v \\ u(E + p) \end{bmatrix}_x + \begin{bmatrix} \rho v \\ \rho u v \\ \rho v^2 + p \\ v(E + p) \end{bmatrix}_y = 0,$$
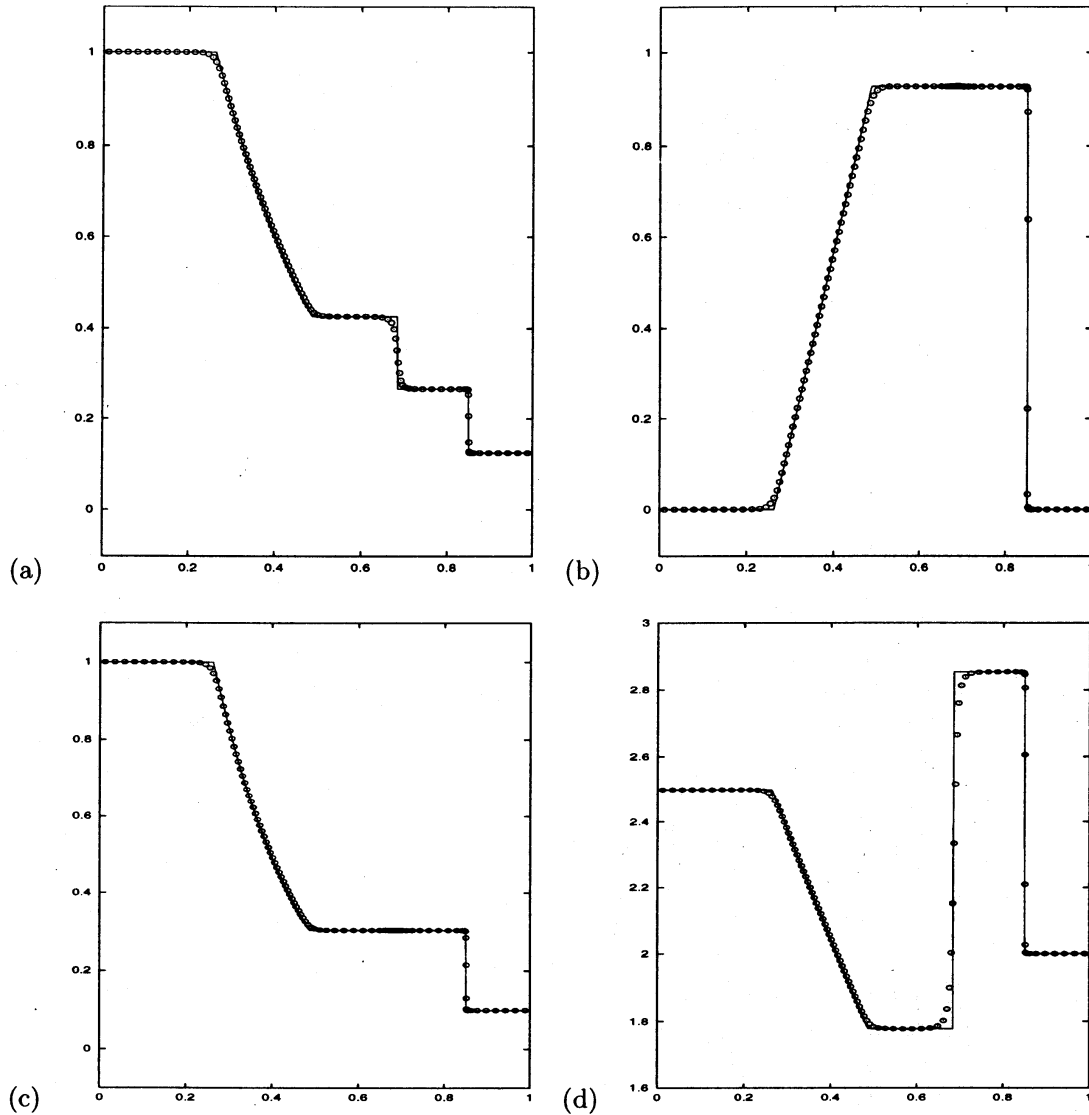
FIG. 3.1. Adaptive mesh solutions at $t = 0.2$ for Sod's shock tube problem. (a): density; (b): velocity; (c): pressure and (d): internal energy. "o" and solid lines denote numerical and exact solutions, respectively.

where ρ, (u, v), p, and E are density, velocity, pressure, and total energy, respectively. For an ideal gas, the equation of state, $p = (\gamma - 1)(E - \rho(u^2 + v^2)/2)$, is provided. The initial data is chosen as

$$(\rho, u, v, p) = \begin{cases} (1.1, 0.0, 0.0, 1.1) & \text{if } x > 0.5, \quad y > 0.5, \\ (0.5065, 0.8939, 0.0, 0.35) & \text{if } x < 0.5, \quad y > 0.5, \\ (1.1, 0.8939, 0.8939, 1.1) & \text{if } x < 0.5, \quad y < 0.5, \\ (0.5065, 0.0, 0.8939, 0.35) & \text{if } x > 0.5, \quad y < 0.5, \end{cases}$$

which corresponds to the case of left forward shock, right backward shock, upper backward shock, and lower forward shock. We refer the readers to [10, 18] for details.

In [10], Lax and Liu computed 2D Riemann problems with various initial data using the positive schemes. The problem considered here corresponds to the Configuration 4 discussed
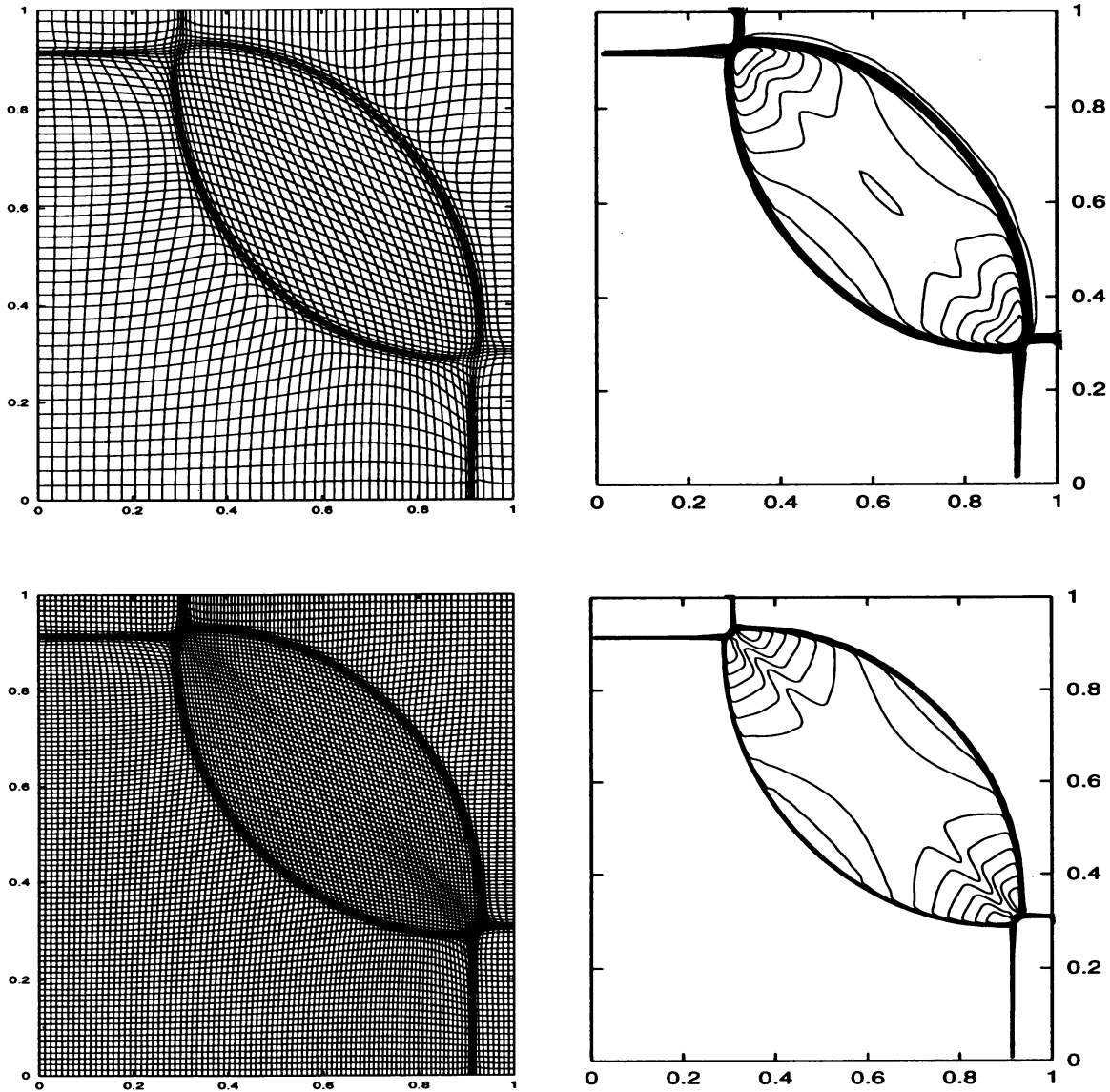
FIG. 3.2. Example 3.2: The contours of mesh (left) and the density (right). Top: $J_x = J_y = 50$ and bottom: $J_x = J_y = 100$. 30 equally spaced contour lines are used for the density.

in their paper. We use our adaptive mesh algorithm with $(J_x, J_y) = (50, 50)$ and $(100, 100)$ to compute this Riemann problem and display the meshes and density at $t = 0.25$ in Figure 3.2. It is found that our results with $J_x = J_y = 100$ give sharper shock resolution than that of the positive schemes with $J_x = J_y = 400$ (see [10], p. 333). The monitor function used in this computation is $G = \omega I$ with $\omega = \sqrt{1 + 2(\rho_\xi^2 + \rho_\eta^2)}$.

EXAMPLE 3.3. **The double-Mach reflection problem.** This problem was studied extensively in Woodward and Colella [23] and later by many others. We use exactly the same setup as in [23], i.e. the same initial and boundary conditions and same solution domain $\Omega_p = [0, 4] \times [0, 1]$. Initially a right-moving Mach 10 shock is positioned at $x = \frac{1}{6}$, $y = 0$ and makes a 60° angle

$$J_x = 80, J_y = 20$$
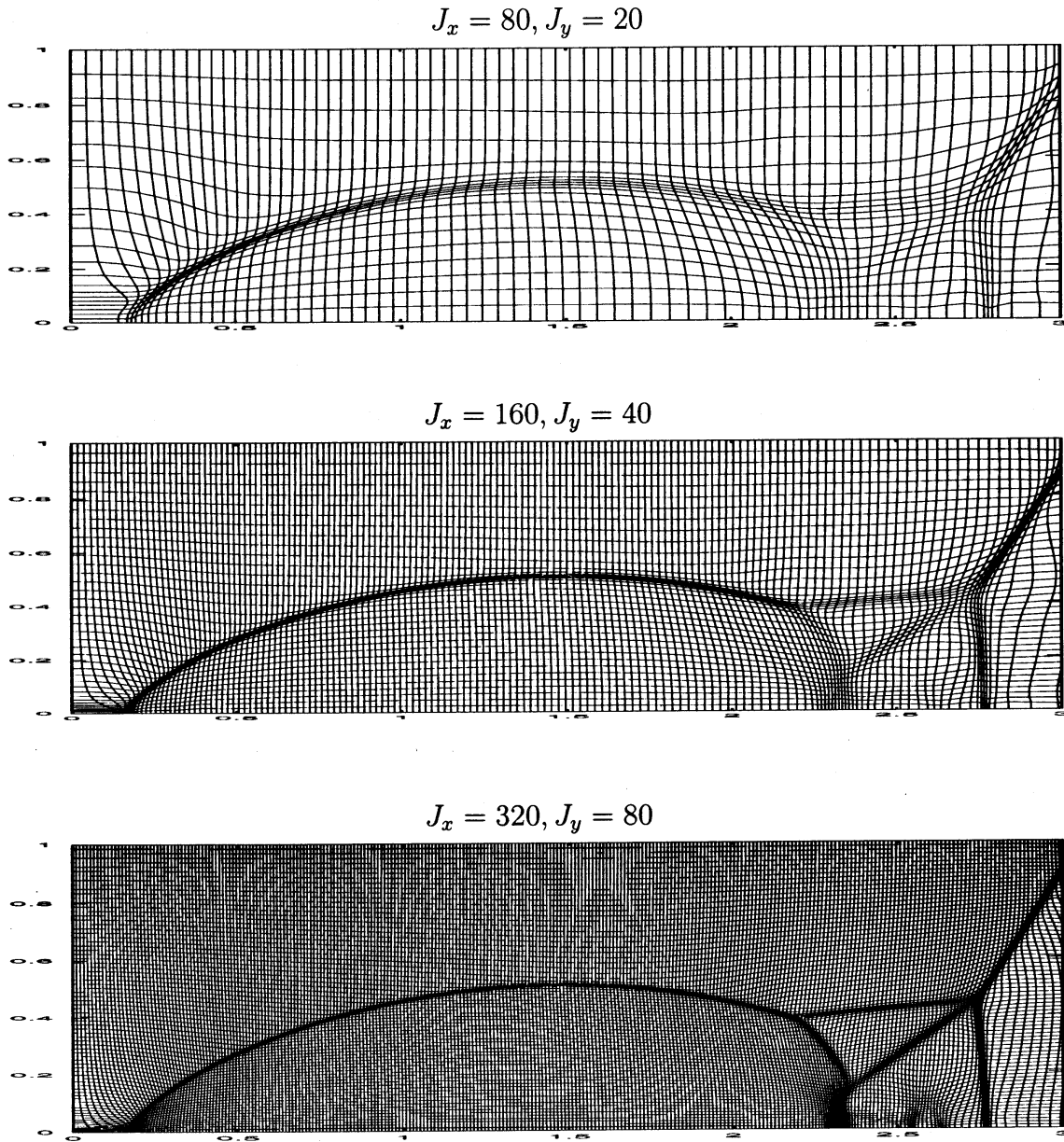
$$J_x = 160, J_y = 40$$

$$J_x = 320, J_y = 80$$

FIG. 3.3. 2D double Mach reflection at $t = 0.2$: the contours of meshes. From top to bottom: $(J_x, J_y) = (80, 20), (160, 40)$ and $(320, 80)$.

with the x-axis. More precisely, the initial data are

$$U = \begin{cases} U_L, & \text{for} \quad y \geq h(x, 0), \\ U_R, & \text{otherwise}, \end{cases}$$

where the state on the left, the state on the right, and the shock height are

$$U_L = (8, \ 57.1597, \ -33.0012, \ 563.544)^T,$$
$$U_R = (1.4, \ 0.0 \ 0.0, \ 2.5)^T,$$
$$h(x, t) = \sqrt{3}(x - 1/6) - 20t.$$

$$J_x = 80, J_y = 20$$
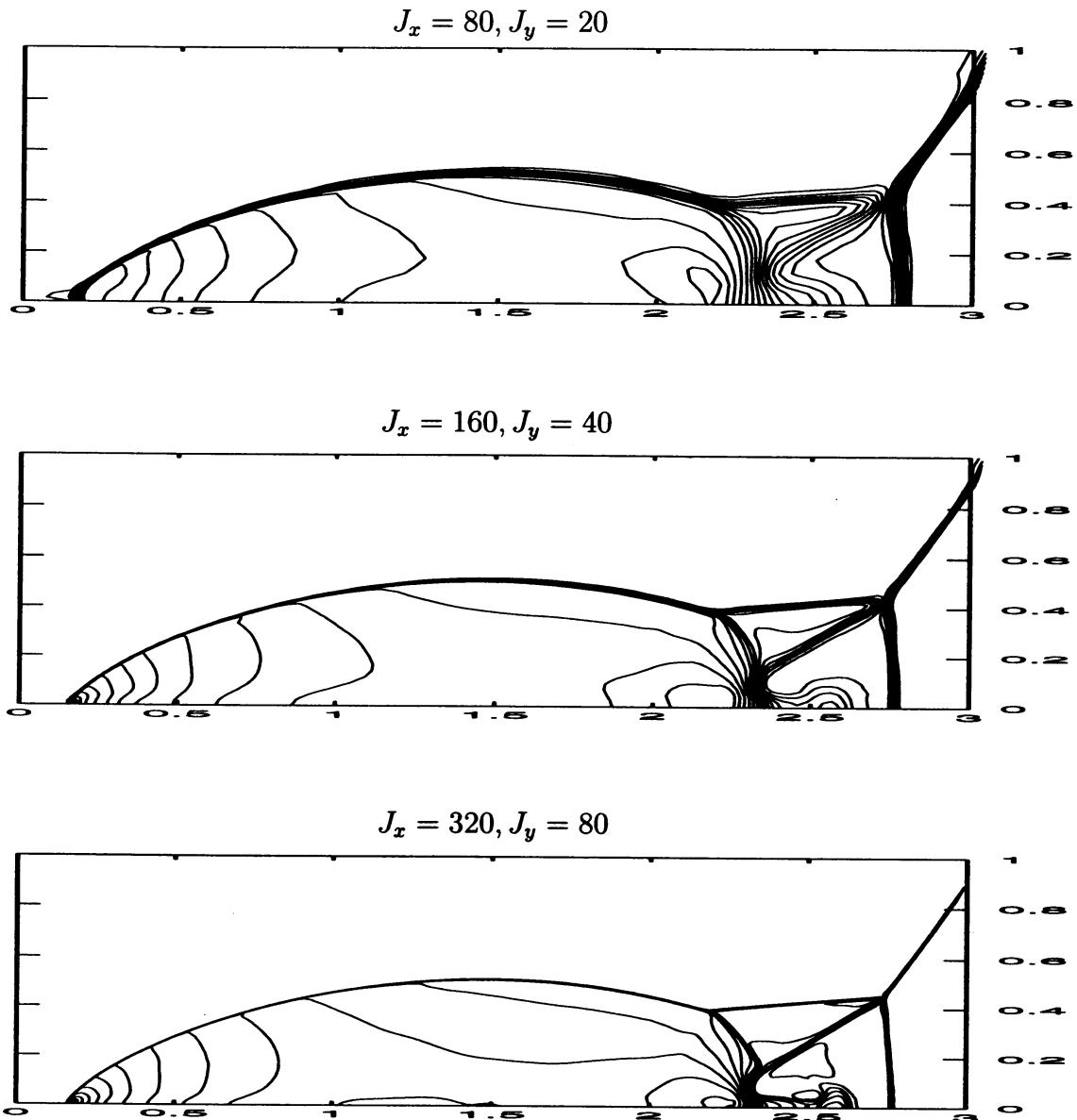

$$J_x = 160, J_y = 40$$

$$J_x = 320, J_y = 80$$

FIG. 3.4. 2D double Mach reflection at $t = 0.2$: the contours of density. From top to bottom: $(J_x, J_y) = (80, 20), (160, 40)$ and $(320, 80)$. 30 equally spaced contour lines are used.

As in [23], only the results in $[0,3] \times [0,1]$ are displayed. In Figure 3.3, the adaptive meshes with $(J_x, J_y) = (80, 20)$, $(160, 40)$ and $(320, 80)$ are displayed, while the corresponding contours of density are displayed in Figure 3.4. By comparing the density plots, it is found the adaptive computation results with $(J_x, J_y) = (320, 80)$ have similar resolution to the results obtained by the second-order discontinuous Galerkin methods with $(J_x, J_y) = (960, 240)$ (p. 214, [7]) and by the second-order central scheme with $(J_x, J_y) = (960, 240)$ (p. 67, [7]). Moreover, the adaptive results with $(J_x, J_y) = (160, 40)$ have slightly better resolution than the results of fifth order weighted ENO and the fourth order ENO with 480×119 grids (p. 406, [7]). Of course, this is not too surprising since these published results are computed by use of uniform meshes.
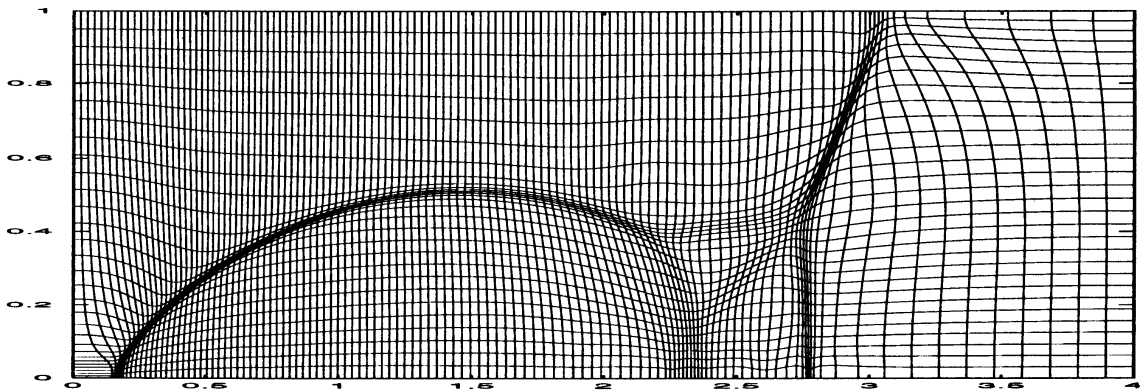
T. Tang and H.-Z. Tang

FIG. 3.5. The adaptive mesh in the whole physical domain for the double Mach reflection problem with 120 × 30 grid points.

Figure 3.5 shows the adaptive mesh in the *whole* physical domain $[0,4] \times [0,1]$, with 120×30 grid points. It is observed that less grid points are distributed for $x > 3$, which is a desired feature for this test problem. Similar observations are observed for the computations with 80×20, 160×40 and 320×80 grid points. The monitor function used for this example is taken as $G = \omega I$ with $\omega = \sqrt{1 + 0.125(\rho_\xi^2 + \rho_\eta^2)}$.

To better appreciate the effectiveness of the adaptive mesh algorithm, we show a *blow up* portion around the double Mach region in Figure 3.6. The fine details of the complicated structure in this region was previously obtained by Cockburn and Shu [6] who used high-order discontinuous Galerkin methods with 960×240 and 1920×480 grid points. In our computations, we used 640×160 and 960×240 grid points. We can see that our result with 960×240 has qualitatively the same resolution as the finer mesh results of the high-order discontinuous Galerkin computations. The corresponding mesh contours in the blow up region are shown in Figure 3.7. The smallest Δx and Δy in these runs are listed in Tables 3.1 and 3.2. It is seen that ratios between the largest and smallest mesh sizes in the adaptive grids are quite large (≥ 20), which is a desired feature of the adaptive grid methods.

REFERENCES

[1] B.N. Azarenok and S. A. Ivanenko, Application of adaptive grids in numerical analysis of time-dependent problems in gas dynamics, *Comput. Maths. Math. Phys.*, 40 (2000), pp. 1330-1349.

[2] J.U. Brackbill, An adaptive grid with directional control, *J. Comput. Phys.* 108 (1993), pp. 38-50.

[3] J.U. Brackbill and J.S. Saltzman, Adaptive zoning for singular problems in two dimensions, *J. Comput. Phys.* 46 (1982), pp. 342-368.

[4] W.M. Cao, W.Z. Huang and R.D. Russell, An r-adaptive finite element method based upon moving mesh PDEs, *J. Comput. Phys.* 149 (1999), pp. 221-244.

[5] H. D. Ceniceros and T. Y. Hou, An efficient dynamically adaptive mesh for potentially singular solutions. *J. Comput. Phys.*, 172, 609-639 (2001).

[6] B. Cockburn and C.-W. Shu, The Runge-Kutta discontinuous Galerkin method for conservation laws V: multidimensional systems, *J. Comput. Phys.* 141 (1998), pp. 199-224.

[7] B. Cockburn, C. Johnson, C.-W. Shu and E. Tadmor, *Advanced Numerical Approximation of Nonlinear Hyperbolic Equations.* Lecture Notes in Math. 1697 (A. Quarteroni ed.). Springer, 1997.

[8] A.S. Dvinsky, Adaptive grid generation from harmonic maps on Riemannian manifolds, *J. Comput. Phys.* 95 (1991), pp. 450-476.
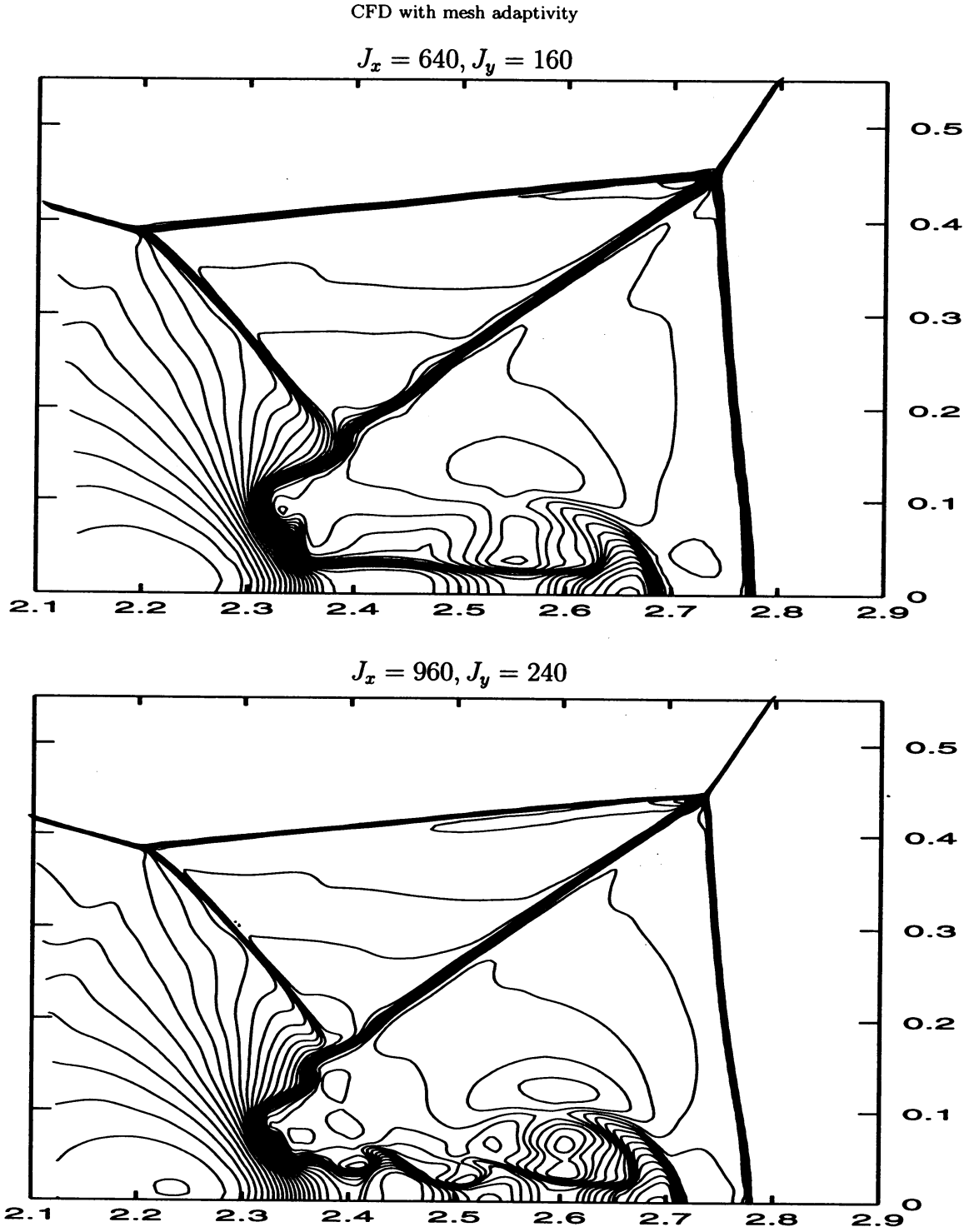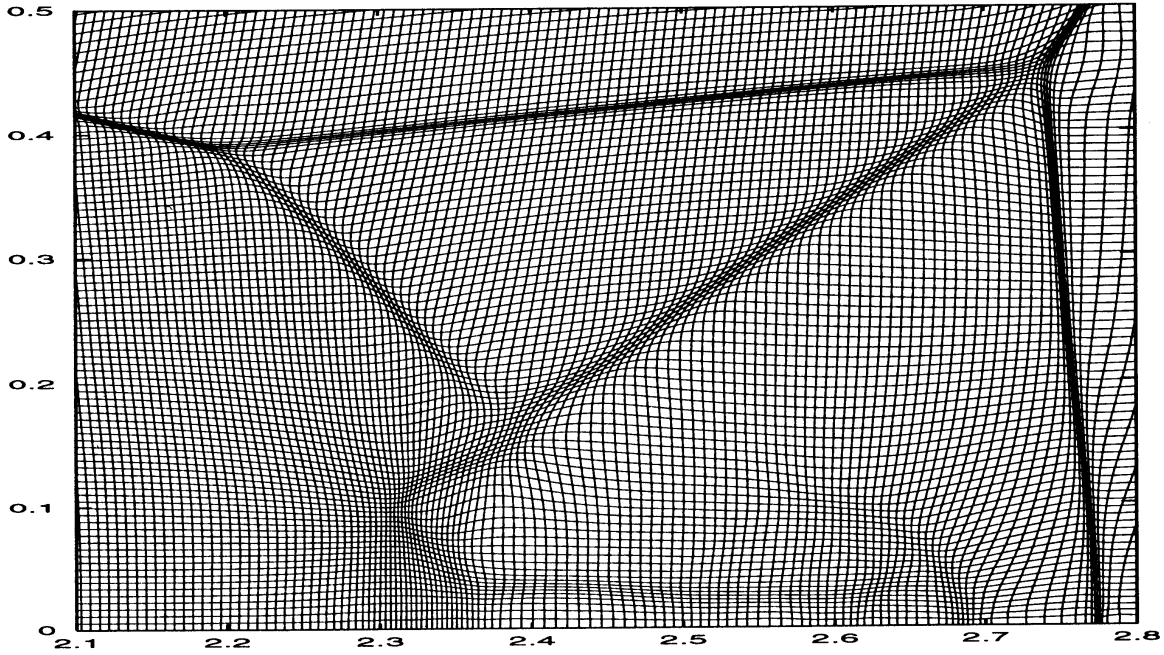
FIG. 3.6. The density contour for the double Mach reflection problem with 640 × 160 grid points (top) and 960 × 240 grid points (bottom). 45 equally spaced contour lines are used.

[9] A. Harten and J.M. Hyman, Self-adjusting grid methods for one–dimensional hyperbolic conservation laws, J. Comput. Phys. 50(1983), pp. 235–269.

[10] P.D. Lax and X.D. Liu, Solutions of two–dimensional Riemann problems of gas dynamics by positive schemes, SIAM J. Sci. Comput. 19(1998), pp.319–340.

[11] S. Li and L. Petzold, Moving mesh methods with upwinding schemes for time-dependent PDEs, J. Comput. Phys. 131 (1997), pp. 368-377.
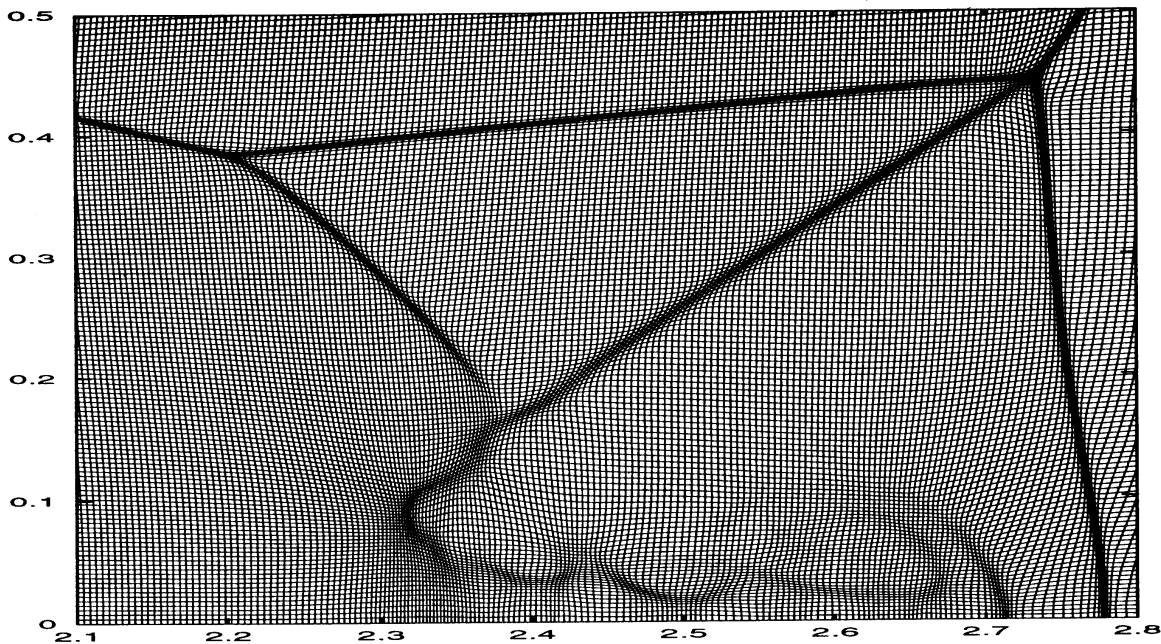
$$J_x = 640,\ J_y = 160$$

$$J_x = 960,\ J_y = 240$$

FIG. 3.7. The adaptive mesh for the double Mach reflection problem with 640 × 160 grid points (top) and 960 × 240 grid points (bottom).

[12] R. Li, T. Tang, and P.W. Zhang, Moving mesh methods in multiple dimensions based on harmonic maps, *J. Comput. Phys.* 170 (2001), pp. 562-588.

[13] R. Li, T. Tang, and P.W. Zhang, moving mesh finite element algorithm for singular problems in two and three space dimensions, To appear in *J. Comput. Phys.*.

[14] F. Liu, S. Ji and G. Liao, An adaptive grid method and its application to steady Euler flow calculations, *SIAM J. Sci. Comput.* 20 (1998), pp. 811-825.

CFD with mesh adaptivity

TABLE 3.1

The smallest mesh sizes for the double Mach reflection problem with 640 × 160 *grid points.*

	$\min\{\Delta x\}$	$\max\{\Delta x\}$	$\max\{\Delta x\}/\min\{\Delta x\}$
Δx	6.5E-04	2.0E-02	30.8
Δy	4.8E-04	1.0E-02	20.8
$\sqrt{\Delta x^2 + \Delta y^2}$	8.2E-04	2.0E-02	24.4

TABLE 3.2

The smallest mesh sizes for the double Mach reflection problem with 960 × 240 *grid points.*

	$\min\{\Delta x\}$	$\max\{\Delta x\}$	$\max\{\Delta x\}/\min\{\Delta x\}$
Δx	4.3E-04	1.1E-02	25.6
Δy	3.1E-04	5.9E-03	19.0
$\sqrt{\Delta x^2 + \Delta y^2}$	5.3E-04	1.2E-02	22.6

[15] K. Miller and R.N. Miller, Moving finite element. I, *SIAM J. Numer Anal.* 18(1981), pp. 1019–1032.

[16] W.Q. Ren and X.P. Wang, An iterative grid redistribution method for singular problems in multiple dimensions, *J. Comput. Phys.* 159(2000), pp.246–273.

[17] K. Saleri and S. Steinberg, Flux-corrected transport in a moving grid. *J. Comput. Phys.* 111 (1994), pp. 24-32.

[18] 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(1993), pp.1394–1414.

[19] G. A. Sod, A survey of finite difference methods for systems of nonlinear hyperbolic conservation laws. *J. Comput. Phys.* 27 (1978), 1-31.

[20] J.M. Stockie, J.A. Mackenzie, and R.D. Russell, A moving mesh method for one–dimensional hyperbolic conservation laws, *SIAM J. Sci. Comput.*, 22 (2001), pp. 1791-1813.

[21] H.-Z. Tang and T. Tang, Adaptive Mesh Methods for One- and Two-Dimensional Hyperbolic Conservation Laws, Preprint, 2001. Available in http://www.math.ntnu.no/conservation/2001/014.html.

[22] A. Winslow, Numerical solution of the quasi–linear Poisson equation, *J. Comput. Phys.* 1 (1967), pp.149–172.

[23] P. Woodward and P. Colella, The numerical simulation of two dimensional fluid flow with strong shocks, *J. Comput. Phys.* 54 (1984), pp. 115-173.