

Second-Order Accurate Godunov Scheme for Multicomponent Flows on Moving Triangular Meshes

Guoxian Chen · Huazhong Tang · Pingwen Zhang

Received: 2 November 2006 / Revised: 3 April 2007 / Accepted: 19 September 2007 /

Published online: 24 October 2007

© Springer Science+Business Media, LLC 2007

Abstract This paper presents a second-order accurate adaptive Godunov method for two-dimensional (2D) compressible multicomponent flows, which is an extension of the previous adaptive moving mesh method of Tang et al. (SIAM J. Numer. Anal. 41:487–515, 2003) to unstructured triangular meshes in place of the structured quadrangular meshes. The current algorithm solves the governing equations of 2D multicomponent flows and the finite-volume approximations of the mesh equations by a fully conservative, second-order accurate Godunov scheme and a relaxed Jacobi-type iteration, respectively. The geometry-based conservative interpolation is employed to remap the solutions from the old mesh to the newly resulting mesh, and a simple slope limiter and a new monitor function are chosen to obtain oscillation-free solutions, and track and resolve both small, local, and large solution gradients automatically. Several numerical experiments are conducted to demonstrate robustness and efficiency of the proposed method. They are a quasi-2D Riemann problem, the double-Mach reflection problem, the forward facing step problem, and two shock wave and bubble interaction problems.

Keywords Adaptive moving mesh method · Finite volume method · Godunov scheme · Multi-component flows · Unstructured mesh

1 Introduction

The hydrodynamics of the mixture of different fluids is of great interest in a wide range of physical flows. Among them some fundamental issues are the dynamics and stability of bubbles and interfaces, mixing processes, bubbly flows, and liquid suspensions, etc. Such fluid

G. Chen · H. Tang (✉) · P. Zhang
LMAM and CCSE, School of Mathematical Sciences, Peking University,
Beijing 100871, People's Republic of China
e-mail: hztang@math.pku.edu.cn

G. Chen
e-mail: gxchen@math.pku.edu.cn

P. Zhang
e-mail: pzhang@math.pku.edu.cn

flows give rise to challenging problems in both theory and numerical simulation. Recent two decades have seen a growing interest in developing numerical methods for compressible multicomponent flows and the investigation of the physical phenomena in complex fluid flows, see e.g. [1, 3, 21, 23–25, 29, 33, 35]. It is well-known that conservative computations of such flows run into unexpected difficulties commonly due to oscillations generated at material interfaces. To overcome those difficulties, many authors studied various models (the γ -model, the mass fraction model, and the level-set model etc.) and proposed some (locally) nonconservative schemes, see [2] and references therein. Recently, Abgrall and Karni in [2] reviewed some of the recent models and numerical algorithms that had been proposed and pointed key ideas that they had in common. Although it has been proved that the nonconservative schemes are very successful in simulating compressible multi-component flows, they will take a risk of producing incorrect results due to non-conservation, see [16]. Up to now, there also exists some work on conservative schemes for compressible multicomponent flows, see e.g. [22, 28, 46].

The main objective of this paper is to extend the adaptive moving mesh method developed in [39] to two-dimensional compressible multimaterial flows. The governing equations will be solved by a fully conservative second-order Godunov scheme with exact Riemann solver on unstructured triangular meshes. Although higher-order accurate Godunov-type schemes on triangular meshes have been studied in many literatures, see e.g. [4, 17, 20, 43], it would be more interesting to see them in simulating multicomponent flows on adaptive moving meshes. Locally clustering mesh points in the regions of the material interface will effectively reduce possible errors (or oscillations) produced by a fully conservative Godunov-type scheme at the material interface. Adaptive moving mesh methods have important applications for a variety of scientific and engineering areas such as solid and fluid dynamics etc., where singular or nearly singular solutions are developed dynamically in fairly localized regions of shock waves, boundary layers, and detonation waves etc. Numerically investigating these phenomena requires extremely fine meshes over a small portion of the physical domain to resolve the large solution variations. Successful implementation of an adaptive strategy can increase accuracy of the numerical approximations and decrease the computational cost. Up to now, there have been many important progresses in adaptive moving mesh methods for partial differential equations, including grid redistribution approach based on the variational principle of Winslow [44] and Brackbill [6], and Ren and Wang [32]; moving finite element methods of Millers [30], and Davis and Flaherty [10]; moving mesh PDEs methods of Russell et al. [5, 7, 19]; and moving mesh methods based on the harmonic mapping of Dvinsky [12], Li et al. [11, 26, 27, 38, 47], and Cenicerros and Hou [8]. Computational costs of moving mesh methods can be possibly saved with locally varying time steps [36], but at the cost of increasing the algorithm complexity. Recently, Chertock and Kurganov in [9] proposed a conservative locally moving mesh method for one-dimensional multifluid flows. It will be more challenging to conduct research in adaptive moving mesh methods for two- and three-dimensional multicomponent flows.

The paper is organized as follows. The governing equations for multicomponent fluid flows are introduced in Sect. 2, and approximated in Sect. 3 by a second-order accurate Godunov scheme on a fixed and unstructured triangular mesh, where a simple slope-limiter is used to avoid numerical oscillations. Section 4 discusses the iterative mesh redistribution. The conservative variables are remapped onto the newly resulting meshes by using a high-resolution geometry-based conservative interpolation. Full solution procedure will be outlined in Sect. 5. Section 6 gives numerical experiments to validate the robustness and efficiency of the proposed adaptive algorithm. Finally, we conclude this work in Sect. 7.

2 Governing Equations

Multicomponent flows we consider are a subset of multiphase flows where the different fluid components, characterized by their respective (constants) ratio of specific heats, are immiscible. Moreover, we neglect diffusive effects, surface tension and cavitation, and assume that the fluid consists of two components. The governing equations for such multicomponent flows may be written by using a single velocity and a single pressure function as follows

$$\begin{aligned}
 \frac{\partial \rho}{\partial t} + \frac{\partial(\rho u)}{\partial x} + \frac{\partial(\rho v)}{\partial y} &= 0, \\
 \frac{\partial(\rho u)}{\partial t} + \frac{\partial(\rho u^2 + p)}{\partial x} + \frac{\partial(\rho uv)}{\partial y} &= 0, \\
 \frac{\partial(\rho v)}{\partial t} + \frac{\partial(\rho uv)}{\partial x} + \frac{\partial(\rho v^2 + p)}{\partial y} &= 0, \\
 \frac{\partial E}{\partial t} + \frac{\partial(u(E + p))}{\partial x} + \frac{\partial(v(E + p))}{\partial y} &= 0,
 \end{aligned} \tag{2.1}$$

where ρ , $\mathbf{u} = (u, v)$, p , and $E = \rho e + \frac{1}{2}\rho(u^2 + v^2)$ are the density, the velocity vector, the pressure, and the total energy, respectively, e denotes the internal energy. Four equations in (2.1) express conservation of mass, momentum, and energy of the fluid mixture. Besides specifying the equation of state (EOS) for the effective thermodynamics

$$p = p(\rho, e),$$

the multicomponent flow description is completed by providing an additional equation that describes the dynamics of the fluid composition.

For the rest of the paper, the variable ϕ is used to describe the fluid composition. Various choices of ϕ have been considered in the literatures, depending on the model assumptions. For example, it is taken to be the ratio of specific heats, the mass fraction, or the level-set function. For all these models, the governing equation for the variables ϕ may be written in a conservative form of

$$\frac{\partial(\rho\phi)}{\partial t} + \frac{\partial(\rho\phi u)}{\partial x} + \frac{\partial(\rho\phi v)}{\partial y} = 0. \tag{2.2}$$

This work is restricted to the perfect gases and the γ -model. Thus the thermodynamic properties of the fluid mixture is described by the ideal EOS

$$p = (\gamma - 1)\rho e,$$

and the interface between two fluids is represented by $\phi = \frac{1}{\gamma-1}$, where γ denotes the effective ratio of specific heats of the fluid mixture and depends on the fluid composition for two fluid flows. It turns out that this particular choice of ϕ offers clear advantages.

We will use the Godunov scheme [14] with the initial data reconstruction [42] to solve (2.1) and (2.2), i.e.

$$U_t + F(U)_x + G(U)_y = 0, \tag{2.3}$$

with

$$U = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ E \\ \rho \phi \end{pmatrix}, \quad F(U) = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ u(E + p) \\ \rho u \phi \end{pmatrix}, \quad G(U) = \begin{pmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ v(E + p) \\ \rho v \phi \end{pmatrix}.$$

The detailed procedure for the exact Riemann solver of (2.3) is completely similar to that for the one-component flows given in the literature, see e.g. [41].

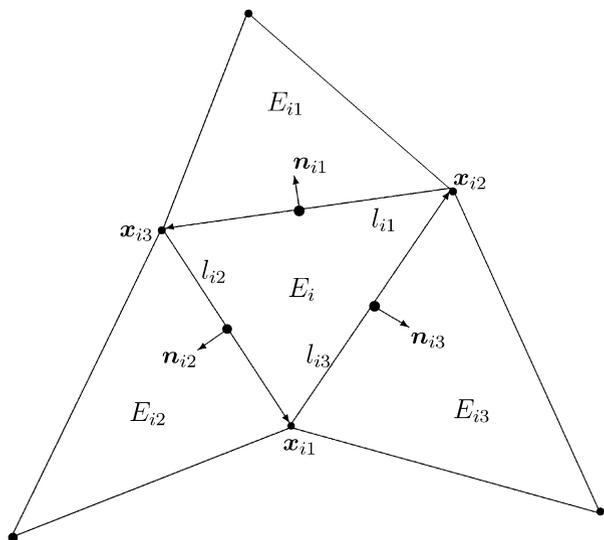
3 Second-Order Accurate Godunov Scheme on Triangular Meshes

Our adaptive multicomponent flow calculation is formed by two independent parts: the evolution of the governing equation and the iterative mesh redistribution. The first part is discussed in this section, while the second part will be introduced in Sect. 4.

In the following, we begin to introduce second-order accurate Godunov scheme of (2.3) on a fixed, unstructured triangular mesh.

Give a triangulation of the physical domain Ω_p , denoted by $\mathcal{T} = \{E_0, E_1, \dots, E_{ne}\}$, where E_i is the i th triangle of the triangulation. For the triangle E_i , we denote $x_{ij} = (x_{ij}, y_{ij})$ its j th vertex, E_{ij} its j th neighboring element, l_{ij} its j th edge, $j = 1, 2, 3$, and n_{ij} the outward unit normal vector on l_{ij} , see Fig. 1 for a detailed schematic diagram. We also assume that a partition of the time interval $[0, T]$ is given as $\{t_n = t_{n-1} + \Delta t_n | \Delta t_n > 0, n \in \mathbb{N}\}$, where the time step size Δt_n should be determined by the stability condition in practice.

Fig. 1 Schematic diagram of the triangle element E_i



Integrating (2.3) over the triangle $E_i \in \mathcal{T}$, we have

$$|E_i| \frac{d}{dt} \bar{U}_{E_i}(t) = - \oint_{\partial E_i} \mathbf{F}_{n_i}(\mathbf{U}) \, ds = - \sum_{j=1}^3 \int_{l_{ij}} \mathbf{F}_{n_{ij}}(\mathbf{U}) \, ds_j, \tag{3.1}$$

where $|E_i|$ is the area of the element E_i , ds and ds_j stand for the surface element measure, and $\mathbf{F}_{n_{ij}} = \mathbf{F}(\mathbf{U})n_{ij}^x + \mathbf{G}(\mathbf{U})n_{ij}^y$ is the flux function in the $\mathbf{n}_{ij} = (n_{ij}^x, n_{ij}^y)$ direction, \mathbf{n}_i is the outward unit normal vector of the triangle boundary $\partial E_i := l_{i1} \cup l_{i2} \cup l_{i3}$. Here $\bar{U}_{E_i}(t)$ denotes the cell average of the conservative variable \mathbf{U} on the triangle E_i , defined by

$$\bar{U}_{E_i}(t) = \frac{1}{|E_i|} \int_{E_i} \mathbf{U}(\mathbf{x}, t) \, d\mathbf{x}. \tag{3.2}$$

We use the midpoint integration formula to approximate the integration in (3.1), and replace the exact solution \mathbf{U} at the middle point of the edge l_{ij} by the approximate solution (i.e. the piecewise polynomial reconstructed by using the cell averages in the finite volume method) and the flux $\mathbf{F}_{n_{ij}}(\mathbf{U})$ by any two-point Lipschitz numerical flux $\hat{\mathbf{F}}_{n_{ij}}(\mathbf{U}_{l_{ij}}^L, \mathbf{U}_{l_{ij}}^R)$, respectively. Based on those, (2.3) can be approximated as

$$\frac{d}{dt} \bar{U}_{E_i}(t) = - \frac{1}{|E_i|} \sum_{j=1}^3 \hat{\mathbf{F}}_{n_{ij}}(\mathbf{U}_{l_{ij}}^L, \mathbf{U}_{l_{ij}}^R) |l_{ij}|, \tag{3.3}$$

where $|l_{ij}|$ denotes the length of edge l_{ij} .

This work employs the Godunov flux with the exact Riemann solver [14] in (3.3):

$$\hat{\mathbf{F}}_{n_{ij}}(\mathbf{U}_{l_{ij}}^L, \mathbf{U}_{l_{ij}}^R) = \mathbf{F}_{n_{ij}}(\omega^*(0; \mathbf{U}_{l_{ij}}^L, \mathbf{U}_{l_{ij}}^R)), \tag{3.4}$$

where $\omega^*(x/t; \mathbf{U}_{l_{ij}}^L, \mathbf{U}_{l_{ij}}^R)$ denotes the exact solution of the one-dimensional Riemann problem

$$\begin{cases} \frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}_{n_{ij}}(\mathbf{U})}{\partial x} = 0, \\ \mathbf{U}(x, 0) = \begin{cases} \mathbf{U}_{l_{ij}}^L, & x < 0, \\ \mathbf{U}_{l_{ij}}^R, & x > 0. \end{cases} \end{cases} \tag{3.5}$$

We refer the readers to the literatures, e.g. [41], for its detailed derivation.

If set

$$\mathbf{U}_{l_{ij}}^L := \bar{U}_{E_i}, \quad \mathbf{U}_{l_{ij}}^R := \bar{U}_{E_j},$$

then (3.3) is only a first-order accurate semi-discrete scheme of (2.3). To get a second-order accurate spatial discretization, the initial reconstruction technique [42] is used to reset $\mathbf{U}_{l_{ij}}^L$ and $\mathbf{U}_{l_{ij}}^R$ on the edge l_{ij} , see Fig. 2, where E_i and E_j are neighboring each other and have a common edge l_{ij} . To do those, we first compute the cell vertex approximate values of the solution by using its cell averages as follows

$$\begin{aligned} \mathbf{U}_{i3} &:= \frac{1}{ne_{i3}} \sum \bar{U}_{\text{triangle surrounding } x_{i3}}, \\ \mathbf{U}_{j2} &:= \frac{1}{ne_{j2}} \sum \bar{U}_{\text{triangle surrounding } x_{j2}}, \end{aligned}$$

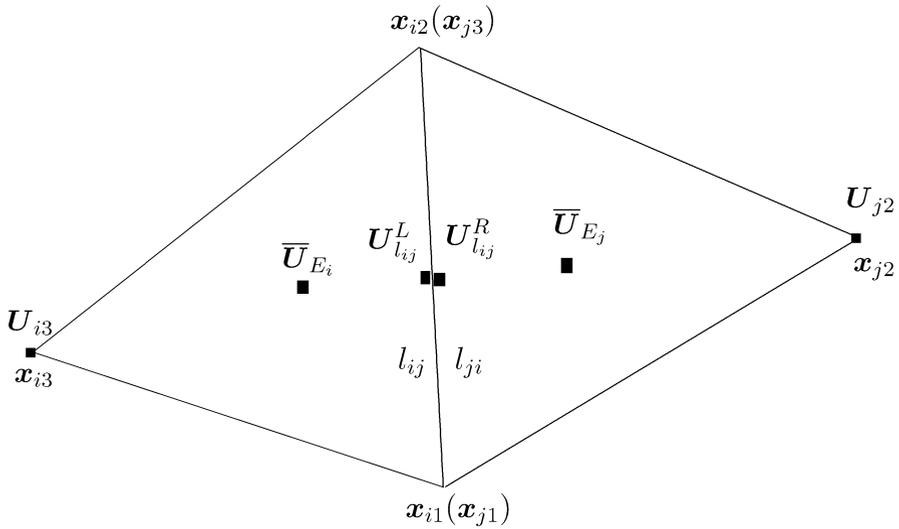


Fig. 2 Schematic diagram of the initial data reconstruction

where ne_{i3} (or ne_{j2}) denotes the number of triangles surrounding x_{i3} (or x_{j2}). Next, we define $U_{l_{ij}}^L$ and $U_{l_{ij}}^R$ on the j th edge of the triangle E_i by

$$U_{l_{ij}}^L = \bar{U}_{E_i} + \frac{1}{2} \Psi(\bar{U}_{E_i} - U_{i3}, \bar{U}_{E_j} - \bar{U}_{E_i}), \tag{3.6}$$

$$U_{l_{ij}}^R = \bar{U}_{E_j} - \frac{1}{2} \Psi(\bar{U}_{E_j} - \bar{U}_{E_i}, U_{j2} - \bar{U}_{E_j}), \tag{3.7}$$

where $\Psi(\cdot, \cdot)$ is a nonlinear limiter function which is used to suppress the possible pseudo-oscillation. If $\Psi(\cdot, \cdot) \equiv 0$, then the edge values $U_{l_{ij}}^L$ and $U_{l_{ij}}^R$ become corresponding cell averages and the semi-discrete scheme (3.3) degenerates to first order accurate spatial approximation. In our computations, we use van Leer’s slope limiter [42]

$$\psi_i = \psi_i(a, b) = (\text{sign}(a) + \text{sign}(b)) \frac{|ab|}{|a| + |b| + \varepsilon},$$

where ψ_i is the i th component of $\Psi = (\psi_1, \dots, \psi_5)^T$, ε is a small positive positive number, $0 < \varepsilon \ll 1$, which is used to avoid that the denominator becomes zero.

System (3.3) may be approximated by any stable time discretization. For example, we use an explicit second-order accurate TVD Runge-Kutta method [34] to evolve solutions of the governing equations (2.3) from t_n to t_{n+1} :

$$\bar{U}^* = \bar{U}^n + \Delta t L(\bar{U}^n), \tag{3.8}$$

$$\bar{U}^{n+1} = \frac{1}{2} \bar{U}^n + \frac{1}{2} (\bar{U}^* + \Delta t L(\bar{U}^*)), \tag{3.9}$$

where $L(\bar{U})$ denotes the term at the right hand side of (3.3). Here we have used $\bar{U}_{E_i}^n$ to stand for an approximation of $\bar{U}_{E_i}(t_n)$ at time t_n .

Remark 3.1 When we evolve the solution within E_j , see Fig. 2, the edge values $U_{l_{ji}}^L$ and $U_{l_{ji}}^R$ on the i th edge of the triangle E_j are set as

$$U_{l_{ji}}^R = \bar{U}_{E_i} - \frac{1}{2}\Psi(U_{i3} - \bar{U}_{E_i}, \bar{U}_{E_i} - \bar{U}_{E_j}),$$

$$U_{l_{ji}}^L = \bar{U}_{E_j} + \frac{1}{2}\Psi(\bar{U}_{E_i} - \bar{U}_{E_j}, \bar{U}_{E_j} - U_{j2}).$$

If the limiter function Ψ satisfies the property of that $\Psi(-V, -W) = -\Psi(V, W)$, then we have

$$U_{l_{ij}}^L = U_{l_{ji}}^R, \quad U_{l_{ij}}^R = U_{l_{ji}}^L.$$

4 Adaptive Mesh Redistribution

This section extends the adaptive mesh redistribution of Tang et al. [39, 40] to unstructured triangular meshes. It is an iterative procedure: redistribute or move triangular mesh points by iteratively solving Euler-Lagrange equations in the logical domain Ω_l and at the same time remap the physical variables onto the resulting new mesh.

4.1 Mesh-Redistribution Based on Variational Methods

Let Ω_l be the logical domain with the orthogonal coordinates $\xi = (\xi, \eta)$ and a Delaunay triangulation, denoted by \mathcal{T}_l , whose data structure is same as that of the triangulation \mathcal{T} of the physical domain Ω_p . Let V_i denote the dual cell associated with the vertex ξ_i in Ω_l , which is delimited in joining the barycenter of all the triangles surrounding ξ_i , see the left plot in Fig. 3. Since \mathcal{T}_l is assumed to be a Delaunay triangulation of Ω_l , the Voronoi diagram or Dirichlet tessellation is a natural choice of the dual partitioning of Ω_l .

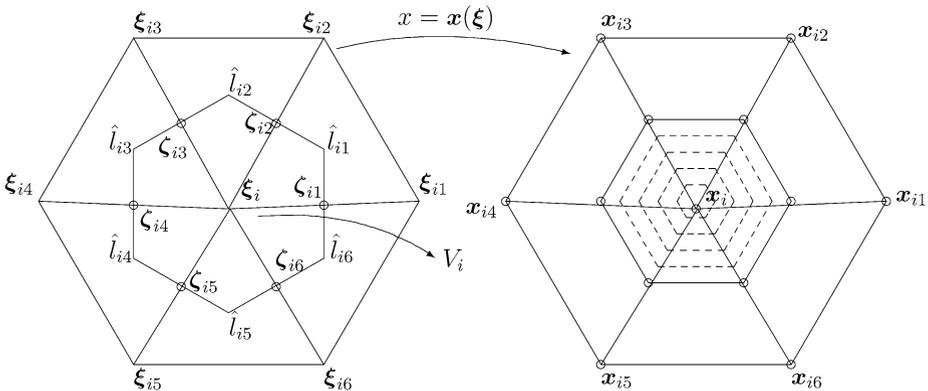


Fig. 3 Schematic diagram of the dual cell V_i and the triangle elements of Ω_l (left) and Ω_p (right)

We denote $\aleph(i)$ the number of triangles surrounding ξ_i , for example, $\aleph(i) = 6$ in the schematic in Fig. 3, and ζ_{ij} the midpoint of the edge l_{ij} connecting nodes ξ_i and ξ_{ij} . Let \hat{l}_{ij} be a linear segment connecting centroids of two neighboring triangles surrounding ξ_i and ξ_{ij} .

A one-to-one coordinate transformation from the logical or computational domain Ω_l to the physical domain Ω_p is denoted by

$$\mathbf{x} = \mathbf{x}(\boldsymbol{\xi}), \quad \boldsymbol{\xi} \in \Omega_l. \tag{4.1}$$

We limit our attention to the case of that the physical domain Ω_p is convex and the map (4.1) is to find the minimizer of the following functional [8, 39, 40]

$$\tilde{E}(\mathbf{x}) = \frac{1}{2} \sum_{i=1}^2 \int_{\Omega_l} (\tilde{\nabla} \mathbf{x}_i)^T G_i \tilde{\nabla} \mathbf{x}_i \, d\boldsymbol{\xi}, \tag{4.2}$$

where $\tilde{\nabla} = (\partial_{\xi}, \partial_{\eta})^T$, and G_i ($i = 1, 2$) are given symmetric positive definite matrices called monitor functions. In general, the monitor functions depend on the solution or its derivatives of the underlying governing equations. The simplest choice of the monitor functions is $G_i = \omega I$, $i = 1, 2$, see [44], where I denotes the identity matrix and ω is a positive weight function. More terms can be added to the above functional to control other aspects of the mesh such as orthogonality and alignment with a given vector field, see e.g. [6, 18].

Using the choice of Winslow, we deduce the Euler-Lagrange equations of the functional (4.2) to

$$\tilde{\nabla} \cdot (\omega \tilde{\nabla} \mathbf{x}) = 0. \tag{4.3}$$

In this study, $\omega = \omega(\tilde{\nabla}_{\xi} \mathbf{U}) =: \omega(\mathbf{U})$, which will be defined in Sect. 6.

We begin to give a finite volume approximation of (4.3) subject to boundary conditions

$$\mathbf{x} \in \partial\Omega_p, \quad \text{if } \boldsymbol{\xi} \in \partial\Omega_l.$$

Integrating (4.3) over the dual cell V_i as shown in Fig. 3 and using the divergence theorem gives

$$0 = \oint_{\partial V_i} w \frac{\partial \mathbf{x}}{\partial n} \, ds = \sum_{j=1}^{\aleph(i)} \int_{\hat{l}_{ij}} w(\mathbf{U}) \frac{\partial \mathbf{x}}{\partial n} \, ds, \tag{4.4}$$

where $\mathbf{n} = (n_{\xi}, n_{\eta})$ is the unit outward normal vector of ∂V_i . Using the numerical integration formula and approximating the term $\frac{\partial \mathbf{x}}{\partial n}(\zeta_{ij})$ by

$$\frac{\partial \mathbf{x}}{\partial n}(\zeta_{ij}) \approx \frac{\mathbf{x}_{ij} - \mathbf{x}_i}{|\boldsymbol{\xi}_i \boldsymbol{\xi}_{ij}|},$$

gives the following discrete mesh equation

$$\sum_{j=1}^{\aleph(i)} w_{ij} |\hat{l}_{ij}| \frac{\mathbf{x}_{ij} - \mathbf{x}_i}{|\boldsymbol{\xi}_i \boldsymbol{\xi}_{ij}|} = 0, \tag{4.5}$$

where $w_{ij} = w(\mathbf{U}(\zeta_{ij}))$. Here we have used the assumption that \hat{l}_{ij} lies on the perpendicular bisector of the edge l_{ij} connecting ξ_i and ξ_{ij} . In practical computation, this assumption may be relaxed.

Generally, (4.5) is a nonlinear algebraic system, due to dependence of w_{ij} on the solution. To avoid this difficulty, we linearize (4.5) and then use a relaxed iteration method to solve it as follows:

$$\hat{\mathbf{x}}_i = \sum_{j=1}^{\mathfrak{N}(i)} W_{ij} \mathbf{x}_{ij}^{[v]} / \sum_{j=1}^{\mathfrak{N}(i)} W_{ij}, \tag{4.6}$$

$$\mathbf{x}_i^{[v+1]} = \mu_i \hat{\mathbf{x}}_i + (1 - \mu_i) \mathbf{x}_i^{[v]}, \tag{4.7}$$

for $v = 0, 1, \dots$, where

$$W_{ij} = \frac{\Delta\tau}{|V_i|} w_{ij} |\hat{l}_{ij}| / |\xi_{ij} \xi_i|, \quad \text{and} \quad \mu_i = \max \left\{ \sum_{j=1}^{\mathfrak{N}(i)} W_{ij}, \sigma \right\}.$$

Here $\Delta\tau$ and σ are two artificial parameters that control the quality of the mesh movement. Obviously, if $0 \leq \max_i \{ \sum_{j=1}^{\mathfrak{N}(i)} W_{ij} \}, \sigma \leq 1$, then the iteration (4.6, 4.7) is positive-preserving. But if μ_i is too big, e.g. near 1, then it is easy that the mesh may be badly distorted; conversely, if μ_i is near 0, the mesh points move too slow. In this work, the mesh iteration is continued until $\|\mathbf{x}^{[v]} - \mathbf{x}^{[v+1]}\| < 10^{-6}$ or $v < 5$, and we take $\max_i \{ \sum_{j=1}^{\mathfrak{N}(i)} W_{ij} \} = \frac{1}{2}$ and $\sigma = 0.3$, which ensure basically that $\mathbf{x}_i^{[v+1]}$ will be within the convex hull of the midpoints of the edge $\mathbf{x}_i \mathbf{x}_{ij}, j = 1, 2, \dots, \mathfrak{N}(i)$, see the shaded region in the right plot of Fig. 3.

Remark 4.1 (Boundary mesh redistribution) The boundary mesh points should be redistributed simultaneously along with the inner mesh movement because the discontinuities may interact with the boundary of the physical domain Ω_p at some finite time. For convenience, we assume that $l_x := \{y = 0, x_a \leq x \leq x_b\}$ is part of $\partial\Omega_p$, and mapped to part of $\partial\Omega_l$, denoted by $l_\xi := \{\eta = 0, \xi_a \leq \xi \leq \xi_b\}$. The mesh points on l_x are redistributed by solving a one-dimensional mesh equation

$$\frac{\partial}{\partial \xi} \left(w_{l_\xi} \frac{\partial x}{\partial \xi} \right) = 0, \quad \xi \in l_\xi, \tag{4.8}$$

subject to the Dirichlet boundary conditions

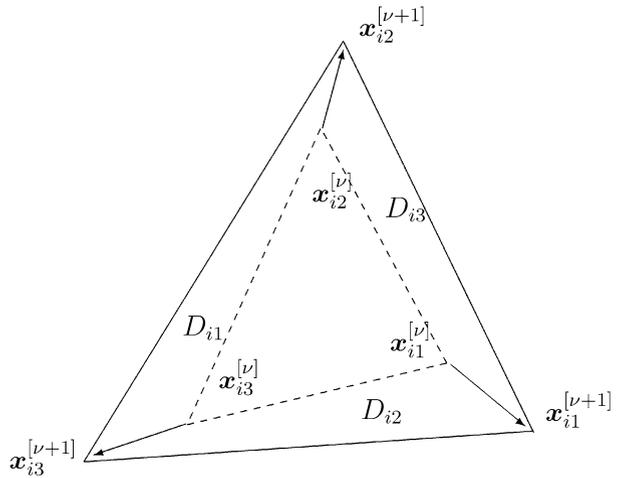
$$x(\xi_a) = x_a \quad \text{and} \quad x(\xi_b) = x_b. \tag{4.9}$$

Here, the monitor w_{l_ξ} is specified as value of the monitor over its adjacent inner element.

4.2 Conservative Interpolation on New Meshes

After each iterative step of (4.6) and (4.7), we need to remap the approximate solutions onto the newly resulting mesh $\{\mathbf{x}_i^{[v+1]}\}$ or $\{E_i^{[v+1]}\}$ from the old mesh $\{\mathbf{x}_i^{[v]}\}$ or $\{E_i^{[v]}\}$. Tang and Tang [39] proposed a conservative and upwind interpolation formula on the structured quadrangular mesh. Recently, Han and Tang [15] gave a simplified geometrical interpolation approach on the same mesh which also preserves conservation property of the conservative

Fig. 4 Movement of the control volume $E_i^{[v]}$ to $E_i^{[v+1]}$



variables U in the sense of that

$$\sum_i |E_i^{[v]}| U_{E_i}^{[v]} = \sum_i |E_i^{[v+1]}| U_{\tilde{E}_i}^{[v+1]},$$

where we have redefined $\tilde{E}_i := E_i^{[v+1]}$ and $E_i := E_i^{[v]}$ in the subscript.

In this work, we extend the geometrical approach of Han and Tang [15] to the unstructured triangular mesh. Let D_{ij} denote the region scanned by the edge $l_{ij}^{[v]}$ after one iterative step of (4.6) and (4.7), $j = 1, 2, 3$, see Fig. 4. We remap the conservative variables as

$$|\tilde{E}_i| U_{\tilde{E}_i}^{[v+1]} = |E_i| U_{E_i}^{[v]} + \sum_{j=1}^3 S_{ij}, \tag{4.10}$$

where S_{ij} is the integral of the approximate solution U over the domain D_{ij} . Following the idea in [15], we may simplify the calculation of S_{ij} . Take D_{i1} shown in Fig. 4 as an example, we first compute $|D_{i1}|$ by

$$|D_{i1}| := \frac{1}{2} ((x_{i3}^{[v+1]} - x_{i2}^{[v]})(y_{i3}^{[v]} - y_{i2}^{[v+1]}) - (y_{i3}^{[v+1]} - y_{i2}^{[v]})(x_{i3}^{[v]} - x_{i2}^{[v+1]})).$$

It is obvious that $|D_{i1}|$ is the signed area function which means that $|D_{i1}|$ is the area of D_{i1} if $x_{i2}^{[v]}, x_{i2}^{[v+1]}, x_{i3}^{[v+1]}$, and $x_{i3}^{[v]}$ are located by counter-clockwise order, and is the inverse of area of D_{i1} if the above four points are located by clockwise order. Then S_{i1} can be approximately calculated as

$$S_{i1} = \max\{|D_{i1}|, 0\} U_{l_{i1}}^R + \min\{|D_{i1}|, 0\} U_{l_{i1}}^L, \tag{4.11}$$

where $U_{l_{i1}}^L$ and $U_{l_{i1}}^R$ are the reconstructed left and right states on the edge l_{i1} by $\{U_{E_i}^{[v]}\}$, see (3.6) and (3.7).

5 Solution Procedure

Our solution procedure is formed by two independent parts: evolution of the governing equations and an iterative mesh redistribution. The first part is a second-order accurate Godunov method on fixed unstructured triangular meshes, see Sect. 3. In each iteration of the second part, the triangular mesh points are first redistributed by the relaxed iteration method (4.6) and (4.7) in Sect. 4.1, and then the conservative variables U are updated on the resulting new meshes by the conservative-interpolation formula (4.10) and (4.11) as well as (3.6) and (3.7), see Sect. 4.2. The solution procedure can be illustrated by the following flowchart:

Algorithm 1

Step 1 Give initial quasi-uniform triangulations of the physical domain Ω_p and the logical domain Ω_l , denoted by $\{x_i^0\}$ and $\{\xi_i\}$ respectively. Compute the cell average of the conservative variables U denoted by $U_{E_i}^{[0]}$.

Step 2 For $n = 0, 1, \dots$, set $x_i^{[0]} := x_i^n, U_{E_i}^{[0]} := U_{E_i}^n$, and do the following steps.

Step 3 For $v = 0, 1, 2, \dots, \mu - 1$, do the following:

- (1) Move mesh points $x^{[v]}$ to $x^{[v+1]}$ by solving (4.6) and (4.7).
- (2) Update the conservative variables $U_{E_i}^{[v+1]}$ on the new mesh $\{x_i^{[v+1]}\}$ according to (4.10) and (4.11).

Step 4 Set $x_i^n := x_i^{[\mu]}, U_{E_i}^n := U_{E_i}^{[\mu]}$, and evolve the governing equations (2.3) on the adaptive mesh $\{x_i^n\}$ by using the second-order accurate finite volume Godunov method, given in Sect. 3, to obtain the numerical approximation $U_{E_i}^{n+1}$ at the time level $t = t_{n+1}$.

Step 5 If $t_{n+1} < T$, then go to Step 2; otherwise output the computed results and stop run.

6 Numerical Experiments

In this section, we apply the proposed adaptive mesh algorithm to several two-dimensional problems to validate its efficiency and performance. Throughout our computations, the CFL number is taken 0.25 unless stated otherwise, the initial mesh is generated by the free software EASYMESH [31], and the monitor function is taken as

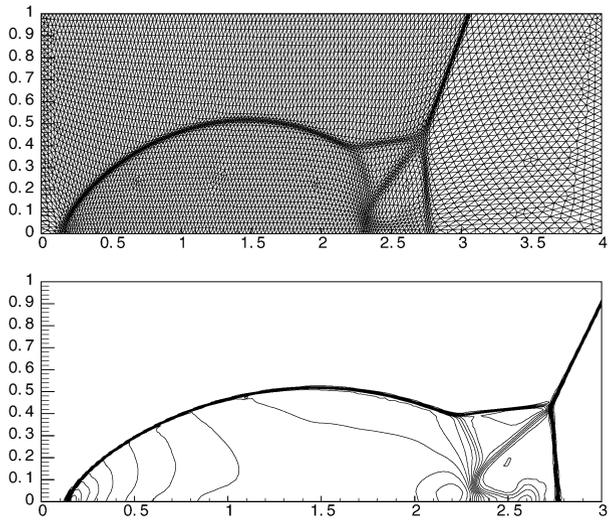
$$w_{E_i} = \sqrt{1 + \alpha_\rho \tilde{w}_{E_i}^2(\beta_\rho, \rho) + \alpha_s \tilde{w}_{E_i}^2(\beta_s, s) + \alpha_\gamma \tilde{w}_{E_i}^2(\beta_\gamma, \gamma)^2}, \tag{6.1}$$

where $s = p/\rho^\gamma, \alpha_q$ and $\beta_q \in (0, 1]$ ($q = \rho, \text{ or } s, \text{ or } \gamma$) are some problem-dependent positive parameters, and will be ascertained in each example. Here \tilde{w}_{E_i} is defined by

$$\begin{aligned} \tilde{w}_{E_i}(\beta_q, q) &=: \min\{1, |\nabla_\xi q|_{E_i} / \Theta\}, \\ \Theta &= \beta_q \max_{E_i}\{|\nabla_\xi q|_{E_i}\}. \end{aligned}$$

Example 6.1 (The double-mach reflection problem) This problem was studied extensively by Woodward and Colella in [45] and later by many others, e.g. [39]. We use exactly the same setup as in [39, 45], i.e., the same initial and boundary conditions and same solution domain $\Omega_p = [0, 4] \times [0, 1]$. The CFL number is 0.4. Initially a right-moving Mach 10 shock in air is positioned at $x = \frac{1}{6}, y = 0$ and makes a 60° angle with a horizontal wall from $x = \frac{1}{6}$

Fig. 5 Example 6.1: Adaptive mesh and contour plots of the density at $t = 0.2$



to 4. The gas density ahead of the shock is 1.4, and the pressure is 1; the density behind the shock is 8. More precisely, the initial data are

$$U = \begin{cases} (8, 57.1597, -33.0012, 563.544)^T, & \text{for } y \leq h(x, 0), \\ (1.4, 0, 0, 2.5)^T, & \text{otherwise,} \end{cases}$$

where

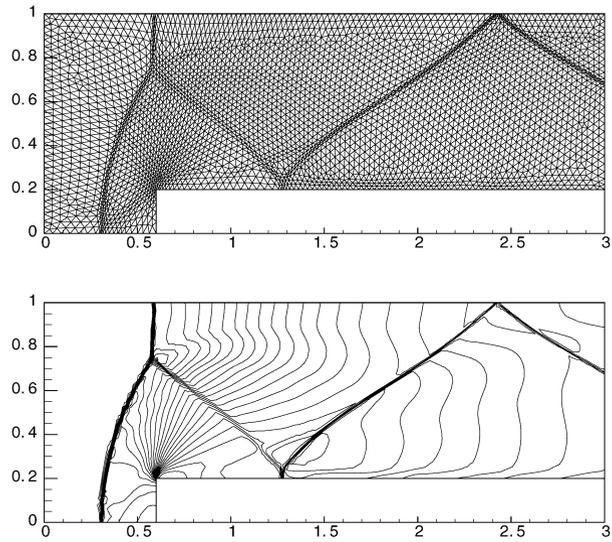
$$h(x, t) = \sqrt{3} \left(x - \frac{1}{6} \right) - 20t$$

is the position of the right-moving shock, the output time $t = 0.2$. The reflective boundary condition is specified on the wall. The fluid variables are specified as the left state of the initial shock on the rest of the bottom boundary, while the left and right states of the right-moving shock at $(\frac{1}{6} + (1 + 20t)/\sqrt{3}, 1)$ on the whole top boundary, respectively. The inflow and outflow boundary conditions are used on the left and right boundaries, respectively.

In this example, we take $\alpha_\rho = 50$, $\beta_\rho = 0.1$, $\alpha_s = \alpha_\gamma = \beta_s = \beta_\gamma = 0$, and an initial quasi-uniform triangulation of the physical domain Ω_p as well as the logical domain Ω_l is generated with the horizontal (and vertical) boundary partition of 134 (and 34) segments. Such quasi-uniform triangulation is formed by 5417 nodes, 15912 edges, and 10495 elements, whose edge length and element area are approximately equal to 3×10^{-2} and 3.811×10^{-4} , respectively.

Figure 5 shows the adaptive mesh and the density contours at $t = 0.2$, which are obtained by using the proposed method. Note that the density contours are only shown with 30 equally spaced contour lines in a part of the physical domain: $0 < x < 3$. At the output time $t = 0.2$, the smallest and largest element areas are respectively 4.074×10^{-5} and 1.8742×10^{-3} , and the smallest and largest edge lengths are 4.488×10^{-3} and 1.395×10^{-1} , respectively. Comparing them with the initial quasi-uniform mesh, we see obviously that the mesh points are well-redistributed, and efficiently clustered in fairly localized regions of shock waves and a small portion of the physical domain containing the large solution variations. The dense jet along the wall has been resolved which is generally sensitive to the numerical dissipation and one of the critical components in this problem.

Fig. 6 Example 6.2: Adaptive mesh and contour plots of the density at $t = 4$



Example 6.2 (The forward facing step problem) This problem was first studied by Emery in [13], and then considered extensively by many other researchers, e.g. [37, 45]. The problem begins with uniform Mach 3 flow in a wind tunnel containing a step. The wind tunnel is 1 length unit wide and 3 length units long. The step is 0.2 length units high and is located 0.6 length units from the left-hand end of the tunnel. Initially the wind tunnel is filled with a gamma-law gas, with $\gamma = 1.4$, which everywhere has density 1.4, pressure 1, and velocity 3. Along the walls of the tunnel reflecting boundary conditions are applied. The in-flow and out-flow boundary conditions are specified at the left- and right-hand ends of the tunnel.

Figure 6 shows the adaptive mesh and the density contours at $t = 4$. The smallest and largest element areas and edge lengths of the final mesh are 2.9946×10^{-5} , 8.7830×10^{-4} , 5.9591×10^{-3} and 5.7834×10^{-2} , respectively. This example takes $\alpha_\rho = 50$, $\beta_\rho = 0.1$, $\alpha_s = \alpha_\gamma = \beta_s = \beta_\gamma = 0$, and an initial quasi-uniform triangulation of the physical domain Ω_p as well as the logical domain Ω_l is generated by 3424 nodes, 10001 edges, and 6578 elements, whose edge length and element area are approximately equal to 2×10^{-2} and 3.6485×10^{-4} , respectively. The sonic glitch phenomenon is almost invisible. Those results are comparable results in [37] obtained on the uniform structure mesh with $\Delta x = \Delta y = 1/200$.

Example 6.3 (The quasi-2D Riemann problem) It is a genuine multimaterial flow calculation, but a simple two-dimensional extension of the Riemann problem of Abgrall [1]. We take the computational domain Ω_p as $[0, 1] \times [0, 0.0335]$ and specify the initial fluid variables as

$$U = \begin{cases} (14.54903, 0, 0, 2.9 \times 10^7, 1.67)^T, & \text{for } x \leq 0.5, \\ (1.16355, 0, 0, 2.5 \times 10^5, 1.4)^T, & \text{otherwise,} \end{cases}$$

because of the difficulty in preventing the numerical oscillation of the pressure and velocity, even in one dimensional case, this Riemann problem is very interesting and are attracting many researchers' attention. In this example, we take $\alpha_\rho = \alpha_\gamma = 60$, $\alpha_s = 0$, $\beta_s = \beta_\gamma = 2\beta_\rho = 1$, and an initial quasi-uniform triangulation of the physical domain Ω_p as well as the logical domain Ω_l is generated by the horizontal (and vertical) boundary partition of 150 (and 6) segments. Such quasi-uniform triangulation is formed by 1078 nodes, 2915

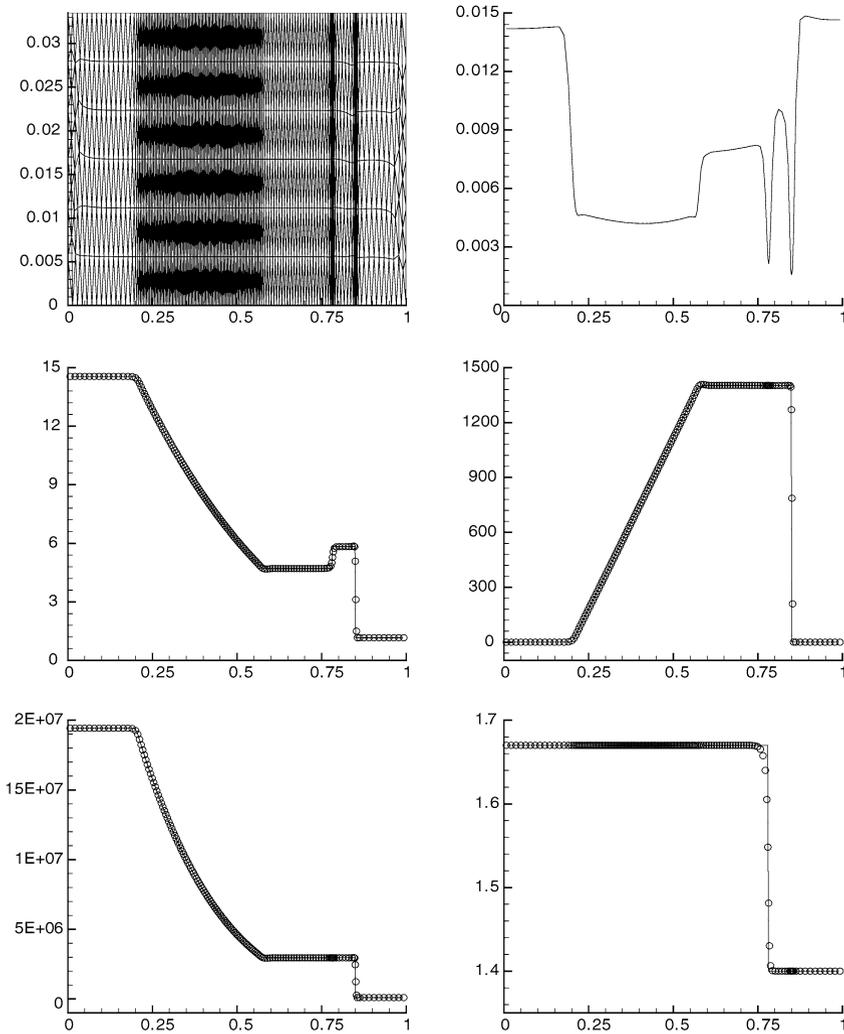
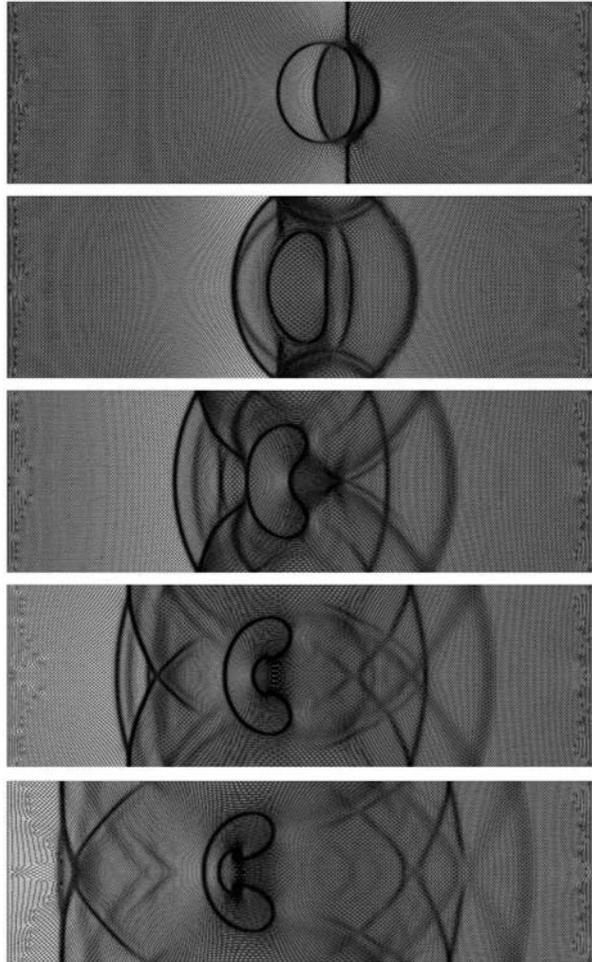


Fig. 7 Example 6.3: The 2D adaptive mesh at $t = 2 \times 10^{-4}$, and the edge length and the computed solutions (ρ, u, p, γ) at the same time along the line $y = 0$

edges and 1838 elements, whose edge length and element area are approximately equal to 6.6×10^{-3} and 1.8226×10^{-5} , respectively.

Figure 7 shows the two-dimensional adaptive mesh at $t = 2 \times 10^{-4}$, and the edge length and the solutions at the same time along the line $y = 0$, where the solid lines in the solution plots denote the exact solutions. We see that the left-moving rarefaction wave, the material interface, and the right-moving shock wave are well resolved; the mesh points are distributed in fairly localized regions of three waves; and the pressure and velocity are oscillatory-free and constant around the material interface. At the output time $t = 2 \times 10^{-4}$, the smallest and largest element area are respectively 4.348×10^{-6} and 4.7733×10^{-5} , the smallest and largest side length are 1.55×10^{-3} and 1.6021×10^{-2} , respectively.

Fig. 8 Example 6.4: The adaptive meshes at $t = 25, 50, 75, 100, 125$

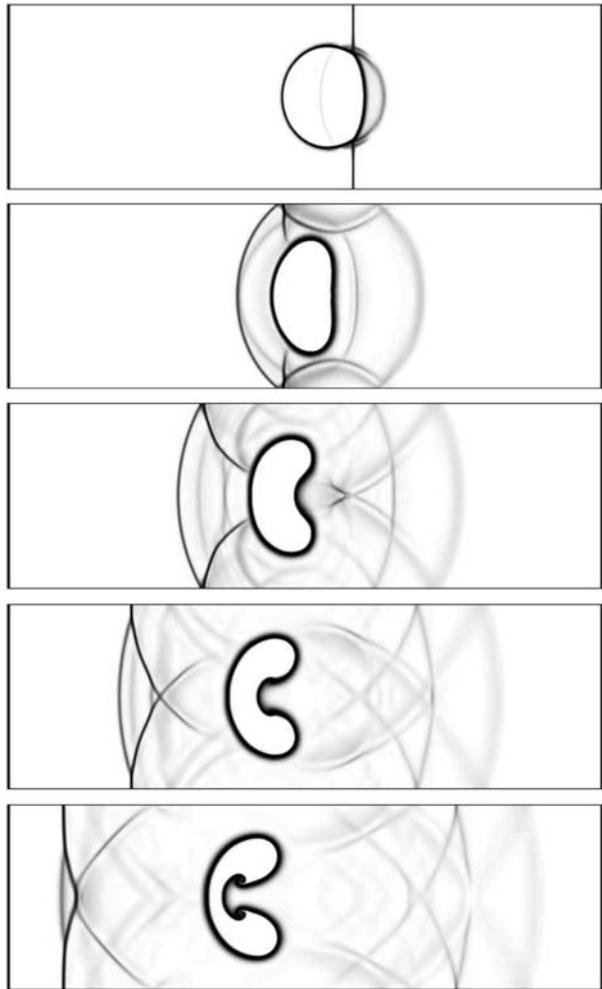


Example 6.4 (The shock wave and a Helium cylindrical bubble interaction) This problem has been extensively studied by many authors, see e.g. [28]. We examine the interaction of a $M_s = 1.22$ planar shock wave, moving in the air, with a Helium cylindrical bubble in the physical domain $\Omega_p = [0, 325] \times [-45, 45]$ with the top and bottom reflective boundaries, the left inflow and right outflow boundaries. Here M_s denotes the shock Mach number. The initial flow is determined from the shock condition with the given shock Mach number. The bubble is assumed to be in both thermal and mechanical equilibrium with the surrounding air. More precisely, the initial dimensionless data are

$$\mathbf{W} = \begin{cases} (1, 0, 0, 1, 1.4), & \text{if } 0 \leq x \leq 225, -44.5 \leq y \leq 44.5, \\ (1.3764, -0.394, 0, 1.5698, 1.4), & \text{if } 225 < x \leq 325, -44.5 \leq y \leq 44.5, \\ (0.1358, 0, 0, 1, 1.67), & \text{if } \sqrt{(x-175)^2 + y^2} \leq 25, \end{cases}$$

where $\mathbf{W} = (\rho, u, v, p, \gamma)$.

Fig. 9 Same as Fig. 8 except for the schlieren images of the density ρ



In this example, we take $\alpha_\rho = 30, \alpha_s = 0, \alpha_\gamma = 20, \beta_\rho = 0.02, \beta_s = 0, \beta_\gamma = 0.5$, and an initial quasi-uniform triangulation of the physical domain Ω_ρ as well as the logical domain Ω_l is generated by the horizontal (and vertical) boundary partition of 328 (and 90) segments. Such quasi-uniform triangulation is formed by 34023 nodes, 101238 edges and 67216 elements, whose edge length and element area are approximately equal to 1 and 0.4352, respectively.

Figures 8 and 9 show the adaptive meshes and the schlieren images of the density ρ at $t = 25, 50, 75, 100, 125$, respectively, where we have chosen a scalar function ψ as

$$\psi = \exp\left(-k \cdot \frac{|\nabla\rho|}{|\nabla\rho|_{\max}}\right), \tag{6.2}$$

with $k = 10$ in the bubble and 60 otherwise, here

$$|\nabla\rho| = \sqrt{\left(\frac{\partial\rho}{\partial x}\right)^2 + \left(\frac{\partial\rho}{\partial y}\right)^2}.$$

Fig. 10 Same as Fig. 8 except for densities along $y = 0$

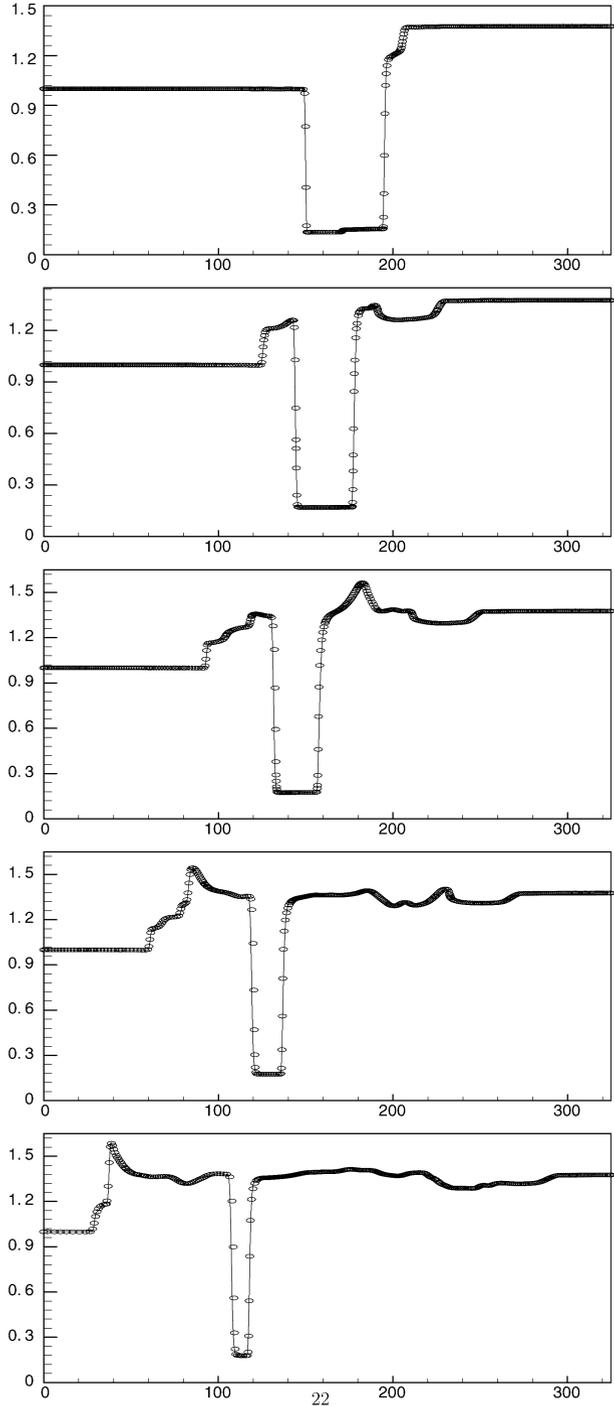
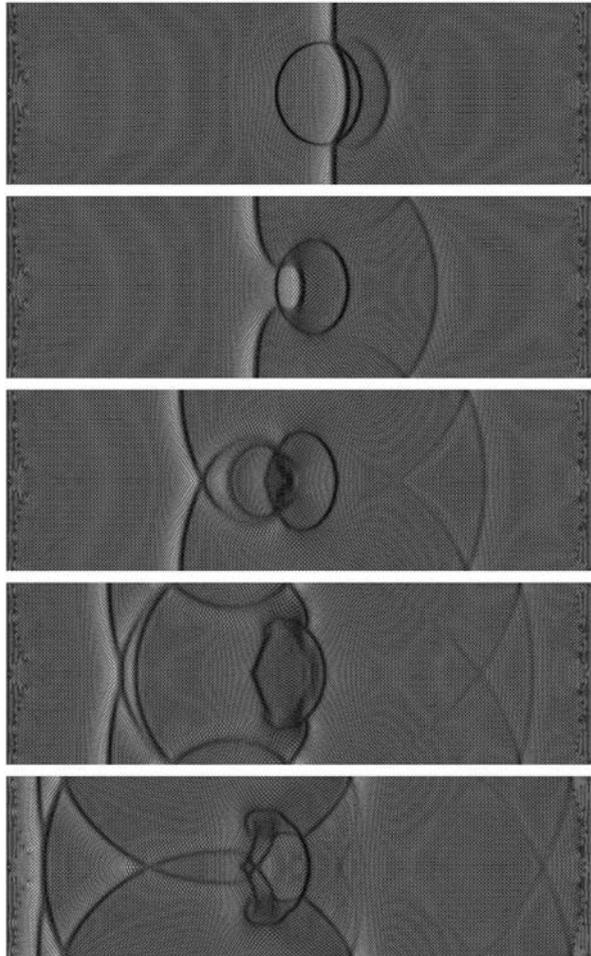


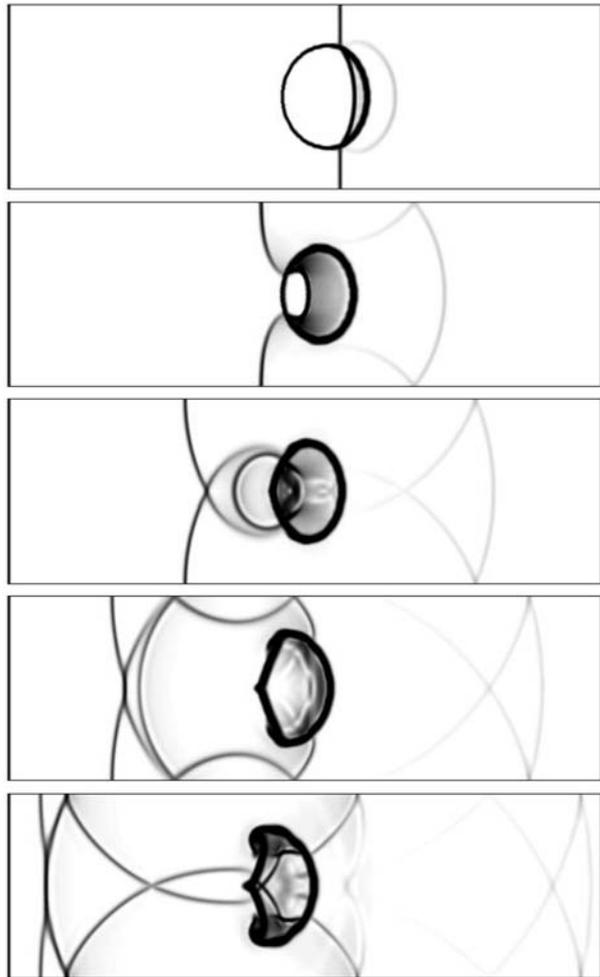
Fig. 11 Example 6.5: The adaptive meshes at $t = 30, 60, 90, 120, 150$



We see that the adaptive redistribution of the computational mesh improves the quantity of the solution successfully; the material interface is captured very well; at the same time, some small wave structures are also resolved clearly. At the final output time $t = 125$, the smallest and largest element area are 5.3992×10^{-2} and 2.1018 respectively, and the smallest and largest side length are 0.19735 and 4.6151, respectively. In Fig. 10, we give a comparison of densities at several output times projected onto a uniform mesh for $\{x|y = 0\}$ obtained by using the second order accurate Godunov scheme with adaptive moving mesh ('o') and with fixed mesh of 68672×4 triangular elements (solid line). The results show that the discontinuities are resolved well and accurately.

Example 6.5 (The shock wave and a R22 cylindrical bubble interaction) This problem is similar to the above example, but the present gas R22 in bubble is heavier than the ambient air, whereas Helium is an inert gas that is lighter than air. The differences between Helium and R22 will yield different flow patterns around the material interface after its interaction with the shock. In this example, we take the physical domain Ω_p and its initial quasi-uniform triangulation and boundary conditions same as those in the shock wave and the Helium

Fig. 12 Same as Fig. 11 except for the schlieren images of the density ρ



bubble problem. The initial data are specified as

$$\mathbf{W} = \begin{cases} (1, 0, 0, 1, 1.4)^T, & \text{if } 0 \leq x \leq 225, -44.5 \leq y \leq 44.5, \\ (1.3764, -0.394, 0, 1.5698, 1.4)^T, & \text{if } 225 < x \leq 325, -44.5 \leq y \leq 44.5, \\ (3.1538, 0, 0, 1, 1.249), & \text{if } \sqrt{(x-175)^2 + y^2} \leq 25. \end{cases}$$

Figures 11 and 12 show the adaptive meshes and the schlieren images of the density ρ at $t = 30, 60, 90, 120, 150$, respectively, where we have chosen a scalar function ψ defined in (6.2) with $k = 4$ in the bubble and 80 outside the bubble.

Here we have taken we take $\alpha_\rho = 30, \alpha_s = 0, \alpha_\gamma = 30, \beta_\rho = 0.02, \beta_s = 0, \beta_\gamma = 0.5$. The results show that the mesh points are well distributed and improve the quantity of the solution effectively; the wave patterns and the material interface are resolved very well. At the final output time $t = 150$, the smallest and largest element area are 1.807×10^{-1} and 1.1705, respectively, and the smallest and largest side length are 0.47909 and 2.3324, respectively. We also give a similar comparison of densities to Example 6.4 is given in Fig. 13. The com-

Fig. 13 Same as Fig. 11 except for densities along $y = 0$

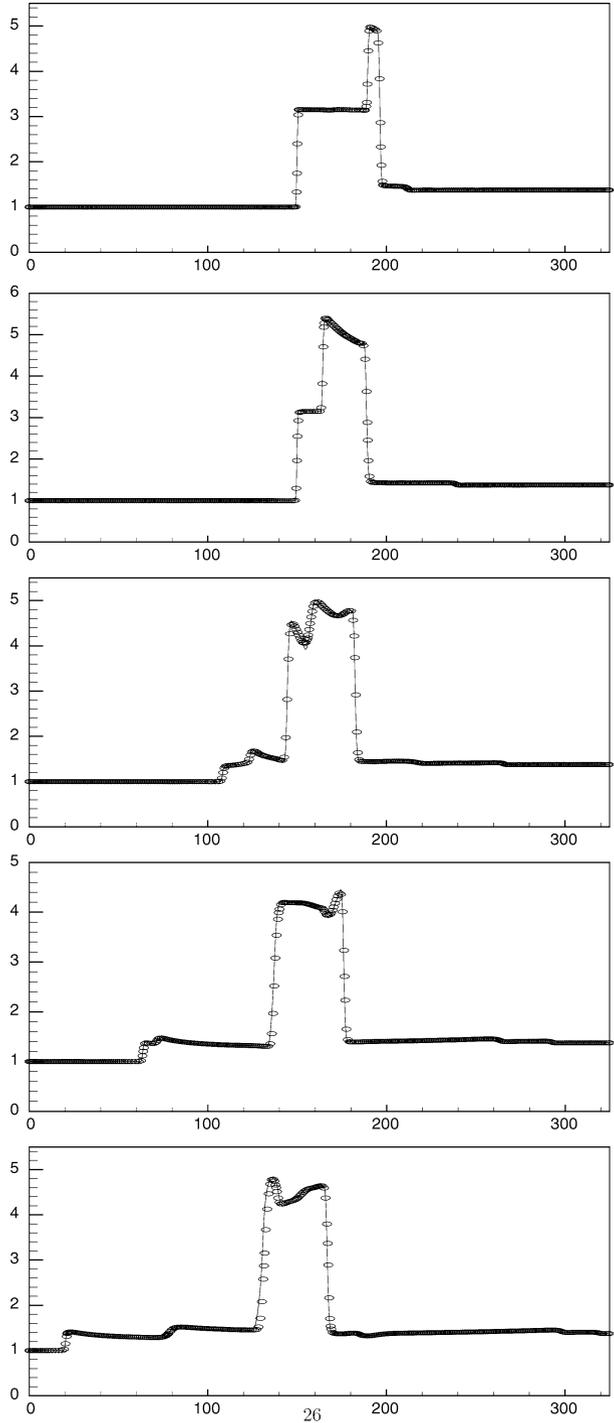


Table 1 Example 6.5: estimated CPU times (minutes) from $t = 0$ to 150

| Algorithm | Element number | CPU time (minutes) |
|-------------|----------------|--------------------|
| Moving mesh | 67216 | 37 |
| Fixed mesh | 268864 | 62 |

putational efficiency comparison between the moving and fixed meshes is shown in Table 1, where the recorded CPU times on the Lenovo PC (Pentium IV, 3 GHz) under the Windows environment.

7 Conclusions

This paper extended successfully the previous adaptive moving mesh method developed by Tang and Tang [39] to two-dimensional (2D) compressible multicomponent flows and unstructured triangular meshes. The proposed method solved the equations governing 2D flows and the finite-volume approximations of the mesh equations by a fully conservative, second-order accurate Godunov scheme and a relaxed Jacobi-type iteration, respectively, and implemented a simple and low-dissipative slope limiter in each initial reconstruction stage in order to get oscillation-free solutions. In the mesh iterative redistribution, the geometry-based conservative interpolation was employed to remap the solutions from the old mesh to the newly resulting mesh, and a new monitor function was carefully chosen to track and resolve both small, local, and large solution gradients automatically. As a result, the current adaptive mesh method is fully conservative and non-oscillatory. Its robustness and efficiency were demonstrated by several numerical experiments. Our future work is to extend the current method to three-dimensional multicomponent flow problems and nontrivial domains.

Acknowledgements The authors would like to thank the anonymous referees for valuable comments and suggestions. Huazhong Tang was partially supported by the National Basic Research Program under the Grant 2005CB321703, the National Natural Science Foundation of China (No. 10431050, 10576001), SRF for ROCS, SEM, NCET, and Laboratory of Computational Physics. Pingwen Zhang was partially supported by the National Basic Research Program under the Grant 2005CB321704.

References

1. Abgrall, R.: How to prevent oscillations in multicomponent flow calculations: A quasi conservative approach. *J. Comput. Phys.* **125**, 150–160 (1996)
2. Abgrall, R., Karni, S.: Computations of compressible multifluids. *J. Comput. Phys.* **169**, 594–623 (2001)
3. Abgrall, R., Saurel, R.: Discrete equations for physical and numerical compressible multiphase mixtures. *J. Comput. Phys.* **186**, 361–396 (2003)
4. Barth, T.J., Jespersen, D.C.: The design and application of upwind schemes on unstructured meshes. AIAA Paper No. 89-0366 (1989)
5. Beckett, G., Mackenzie, J.A., Robertson, M.L.: An r -adaptive finite element method for the solution of the two-dimensional phase-field equations. *Commun. Comput. Phys.* **1**, 805–826 (2006)
6. Brackbill, J.U.: An adaptive grid with directional control. *J. Comput. Phys.* **108**, 38–50 (1993)
7. Cao, W.M., Huang, W.Z., Russell, R.D.: An r -adaptive finite element method based upon moving mesh PDEs. *J. Comput. Phys.* **149**, 221–244 (1999)
8. Ceniceros, H.D., Hou, T.Y.: An efficient dynamically adaptive mesh for potentially singular solutions. *J. Comput. Phys.* **172**, 609–639 (2001)

9. Chertock, A., Kurganov, A.: Conservative locally moving mesh method for multifluid flows. In: Proc. 4th Inter. Sym. on Finite Volumes for Complex Appl., Marrakech, pp. 273–284 (2005)
10. Davis, S.F., Flaherty, J.E.: An adaptive finite element method for initial-boundary value problems for partial differential equations. *SIAM J. Sci. Stat. Comput.* **3**, 6–27 (1982)
11. Di, Y.N., Li, R., Tang, T., Zhang, P.W.: Moving mesh finite element methods for the incompressible Navier-Stokes equations. *SIAM J. Sci. Comput.* **26**, 1036–1056 (2005)
12. Dvinsky, A.S.: Adaptive grid generation from harmonic maps on Riemannian manifolds. *J. Comput. Phys.* **95**, 450–476 (1991)
13. Emery, A.F.: An evaluation of several differencing methods for inviscid fluid flow problem. *J. Comput. Phys.* **2**, 306–331 (1968)
14. Godunov, S.K.: Finite difference method for numerical computation of discontinuous solution of the equations of fluid dynamics. *Mat. Sb.* **47**, 271–306 (1959)
15. Han, J.Q., Tang, H.Z.: An adaptive moving mesh method for multidimensional ideal magnetohydrodynamics. *J. Comput. Phys.* **220**, 791–812 (2007)
16. Hou, T.Y., LeFloch, P.G.: Why nonconservative schemes converge to wrong solutions: Error analysis. *Math. Comput.* **62**, 497–530 (1994)
17. Hu, C.Q., Shu, C.W.: Weighted essentially non-oscillatory schemes on triangular meshes. *J. Comput. Phys.* **150**, 97–127 (1999)
18. Huang, W.Z.: Metric tensors for anisotropic mesh generation. *J. Comput. Phys.* **204**, 633–665 (2005)
19. Huang, W.Z.: Mathematical principles of anisotropic mesh adaptation. *Commun. Comput. Phys.* **1**, 276–310 (2006)
20. Jameson, A., Mavriplis, D.: Finite volume solution of the two-dimensional Euler equations on a regular triangular mesh. AIAA Paper No. 85-0435 (1985)
21. Jia, P.Y., Jiang, S., Zhao, G.P.: Two-dimensional compressible multimaterial flow calculations in a unified coordinate system. *Comput. Fluids* **35**, 168–188 (2006)
22. Jiang, S., Ni, G.X.: A γ -model BGK scheme for compressible multifluids. *Int. J. Numer. Meth. Fluids* **46**, 163–182 (2004)
23. Karni, S.: Hybrid multifluid algorithms. *SIAM J. Sci. Comput.* **17**, 1019–1039 (1996)
24. Karni, S.: Multicomponent flow calculations by a consistent primitive algorithm. *J. Comput. Phys.* **112**, 31–43 (1994)
25. Larouturou, B.: How to preserve the mass fraction positive when computing compressible multicomponent flows. *J. Comput. Phys.* **95**, 59–84 (1991)
26. Li, R., Tang, T., Zhang, P.W.: Moving mesh methods in multiple dimensions based on harmonic maps. *J. Comput. Phys.* **170**, 562–588 (2001)
27. Li, R., Tang, T., Zhang, P.W.: A moving mesh finite element algorithm for singular problems in two and three space dimensions. *J. Comput. Phys.* **177**, 365–393 (2002)
28. Lian, Y.S., Xu, K.: A gas-kinetic schemes for multimaterial flows and its application in chemical reactions. *J. Comput. Phys.* **163**, 349–375 (2000)
29. Liu, T.G., Khoo, B.C., Yeo, K.S.: The simulation of compressible multi-medium flow. Part I: A new methodology with applications to 1D gas-gas and gas-water cases. *Comput. Fluids* **30**, 291–314 (2001)
30. Miller, K., Miller, R.N.: Moving finite element I. *SIAM J. Numer. Anal.* **18**, 1019–1032 (1981)
31. Niceno, B.: A two-dimensional quality mesh generator. <http://www-dinma.univ.trieste.it/nirftc/research/easymesh/>
32. Ren, W.Q., Wang, X.P.: An iterative grid redistribution method for singular problems in multiple dimensions. *J. Comput. Phys.* **159**, 246–273 (2000)
33. Saurel, R., Abgrall, R.: A multiphase Godunov method for compressible multifluid and multiphase flows. *J. Comput. Phys.* **150**, 425–467 (1999)
34. Shu, C.W., Osher, S.: Efficient implementation of essentially non-oscillatory shock-capturing schemes II. *J. Comput. Phys.* **83**, 32–78 (1989)
35. Shyue, K.M.: A wave-propagation based volume tracking method for compressible multicomponent flow in two space dimensions. *J. Comput. Phys.* **215**, 219–244 (2006)
36. Tan, Z.J., Zhang, Z.R., Tang, T., Huang, Y.Q.: Moving mesh methods with locally varying time steps. *J. Comput. Phys.* **200**, 347–367 (2004)
37. Tang, H.Z.: A wave-propagation based volume tracking method for compressible multicomponent flow in two space dimensions. *J. Comput. Phys.* **215**, 219–244 (2006)
38. Tang, H.Z.: A moving mesh method for the Euler flow calculations using a directional monitor function. *Commun. Comput. Phys.* **1**, 656–676 (2006)
39. Tang, H.Z., Tang, T.: Adaptive mesh methods for one- and two-dimensional hyperbolic conservation laws. *SIAM J. Numer. Anal.* **41**, 487–515 (2003)
40. Tang, H.Z., Tang, T., Zhang, P.W.: An adaptive mesh redistribution method for nonlinear Hamilton-Jacobi equations in two- and three-dimensions. *J. Comput. Phys.* **188**, 543–572 (2003)

41. Toro, E.F.: Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction, 2nd edn. Springer, Berlin (1999)
42. van Leer, B.: Towards the ultimate conservative difference schemes V. A second-order sequel to Godunov's method. *J. Comput. Phys.* **32**, 101–136 (1979)
43. Wang, J., Liu, R.X.: A comparative study of finite volume methods on unstructured meshes for simulation of 2D shallow water wave problems. *Math. Comput. Simul.* **53**, 171–184 (2000)
44. Winslow, A.: Numerical solution of the quasi-linear Poisson equation. *J. Comput. Phys.* **1**, 149–172 (1967)
45. Woodward, P.R., Colella, P.: The numerical simulation of two dimensional fluid flow with strong shocks. *J. Comput. Phys.* **54**, 115–173 (1984)
46. Xu, K.: BGK-based scheme for multicomponent flow calculations. *J. Comput. Phys.* **134**, 122–133 (1997)
47. Zhang, Z.R.: Moving mesh method with conservative interpolation based on L^2 -projection. *Commun. Comput. Phys.* **1**, 930–944 (2006)