Journal of Computational Physics **153,** 596–637 (1999)
Article ID jcph.1999.6295, available online at http://www.idealibrary.com on IDEAL®

A Unified Coordinate System for Solving the Two-Dimensional Euler Equations[1]

W. H. Hui, P. Y. Li, and Z. W. Li[2]

Department of Mathematics and Center for Scientific Computation, Hong Kong University of Science & Technology, Clear Water Bay, Hong Kong
E-mail: whhui@ust.hk

Received January 5, 1999; revised April 29, 1999

It is well known that the use of Eulerian coordinates for shock capturing methods results in badly smeared slip lines, and that Lagrangian coordinates, while capable of producing sharp slip line resolution, may result in severe grid deformation, causing inaccuracy and even breakdown of computation. A unified coordinate system is introduced in which the flow variables are considered to be functions of time and of some permanent identification of *pseudo-particles* which move with velocity $h\mathbf{q}$, $\mathbf{q}$ being the velocity of fluid particles. It includes the Eulerian coordinates as a special case when $h = 0$, and the Lagrangian when $h = 1$. For two-dimensional inviscid flow, the free function h is chosen so as to preserve the grid angles. This results in a coordinate system which avoids excessive numerical diffusion across slip lines in the Eulerian coordinates and avoids severe grid deformation in the Lagrangian coordinates, yet it retains sharp resolution of slip lines, especially for steady flow. Furthermore, the two-dimensional unsteady Euler equations of gasdynamics in the unified coordinates are found to be hyperbolic for all values of h, except when $h = 1$ (i.e., Lagrangian). In the latter case the Euler equations are only weakly hyperbolic, lacking one eigenvector, although all eigenvalues are real. The consequences of this deficiency of the Lagrangian coordinates are pointed out in connection with numerical computation.

Key Words: unified description; Eulerian description; Lagrangian description; inviscid compressible flow; slip lines; hyperbolicity of Euler equations.

[1] Presented at the "Symposium on Modern Developments in Computational Fluid Dynamics and Computational Plasma Physics" to celebrate the 70th birthday of C. K. Chu, Columbia University, New York, May 14–15, 1998.

[2] Permanent address: China Aerodynamics Research and Development Center, Mianyang, Sichuan, China.

0021-9991/99 $30.00

1. INTRODUCTION

For over two hundred years, two different coordinate systems for describing fluid motion have existed: the Eulerian system describes fluid motion at fixed locations, whereas the Lagrangian system does so following fluid particles. Accordingly, the Eulerian description considers velocities and other properties of fluid particles to be functions of time and of fixed space coordinates. By contrast, the Lagrangian description considers the positions of fluid particles and their other properties to be functions of time and of their permanent identifications, such as their initial positions or any set of material functions of fluid particles. Analytically, both coordinate systems are capable of producing exact solutions of fluid flow, including discontinuous flow. They are regarded as equivalent to each other (for one-dimensional flow, the equivalency was proved rigorously by Wagner [1]), except that the Lagrangian system gives more information: it tells each fluid particle's history. They are *not equivalent* from a numerical computation point of view.

Computationally, in using the Eulerian coordinates the computational cells are fixed in space, while fluid particles move across cell interfaces in any direction. It is this convective flux that causes excessive numerical diffusion in the numerical solutions. Indeed, slip lines are smeared badly and shocks are also smeared, albeit somewhat better than slip lines. Moreover, the smearing of slip lines ever increases with time and distance unless special treatments, such as artificial compression or sub-cell resolution, are employed [2–4] which are, however, not always reliable. The primary efforts of the CFD algorithm researchers since the sixties have concentrated on developing better (more robust, accurate, and efficient) ways of dealing with this convective flux. Although great progress has been made and "perhaps to the point of near perfection and little return could be gained" [5], numerical diffusion still exists, causing inaccuracy, and is even more difficult to handle in multi-dimensional flow problems. Another disadvantage of the Eulerian coordinates is that a grid generation, which can be time-consuming, is needed prior to flow computation in order to satisfy boundary conditions on solid boundaries.

Computational cells in the Lagrangian coordinates, on the other hand, are literally fluid particles. Consequently, there is no convective flux across cell interfaces and numerical diffusion is thus minimized. However, the very fact that computational cells exactly follow fluid particles can result in severe grid deformation, causing inaccuracy and even breakdown of the computation. To prevent this from happening, the most famous Lagrangian method in use at the present time—the Arbitrary Lagrangian–Eulerian Technique (ALE) [6–8]—uses continuous re-zoning and re-mapping to the Eulerian grid. Unfortunately, this process requires interpolations of geometry and flow variables which result in loss of accuracy, manifested as numerical diffusion which ALE wants to avoid in the first place. Indeed, it was demonstrated in [9] that re-zoning results in diffusive errors of the type encountered in Eulerian solutions and continuously re-zoned Lagrangian computation is equivalent to an Eulerian computation. Another disadvantage of the Lagrangian coordinates is that, except in the simple case of one-dimensional unsteady flow, the governing equations for inviscid flow are not easily written in conservation form, making it difficult to capture shocks correctly.

After a series of studies [10–17] on steady supersonic flow, it was found that the advantages of Lagrangian coordinates arise from computational cells moving in the direction of the fluid particles but not with their speeds. It was also found that literally following fluid particles, as does Lagrangian, not only causes computational cells to deform with the fluid but also renders the governing equations for inviscid supersonic flow not fully hyperbolic,

as there is no complete set of eigenvectors, although all eigenvalues are still real. With this discovery, the generalized Lagrangian method [18] was introduced for steady supersonic flow and was shown to be superior to the Eulerian and the classical Lagrangian method, especially in resolving slip lines and shocks.

In this paper we extend the above idea to unsteady flow by introducing a new description of fluid motion in which the flow variables (velocities, pressure, density, etc) are considered to be functions of time and of some permanent identifications of *pseudo-particles* which move with velocity $h\mathbf{q}$, $\mathbf{q}$ being the velocity of fluid particles and h arbitrary. This turns out to be a unified description, ranging from Eulerian when $h = 0$ to Lagrangian when $h = 1$, and the freedom in choosing h makes it possible to avoid the disadvantages of excessive diffusion across slip lines in the Eulerian description and of severe grid deformation in the Lagrangian description. For these purposes, the choice of h to preserve grid angles in two-dimensional flow has been shown in this paper to be most successful.

The extension from steady supersonic flow to unsteady flow is not trivial, but it then allows us not only to compute unsteady flow but also to compute steady subsonic, supersonic, and transonic flow as the asymptotic state of unsteady flow for large time.

This paper is organized as follow: in Section 2 we introduce the unified coordinates, whereas Sections 3 and 4 discuss the mathematical properties of the 2-D unsteady Euler equations of gasdynamics written in the unified coordinates. Section 5 outlines the numerical solution strategy and Section 6 gives details of the Riemann solution needed in the numerical procedures described in Section 7. Section 8 gives results of the numerical computations on four test problems and compares them with corresponding results based on Eulerian or Lagrangian coordinates, showing the advantages of the unified coordinates. Finally, conclusions are given in Section 9.

2. THE UNIFIED COORDINATES

Starting from Cartesian coordinates (x, y, z) and time t in the Eulerian description, we make a transformation to coordinates $(\lambda, \xi, \eta, \zeta)$,

$$\begin{cases} dt = d\lambda & \text{(1a)} \\ dx = hud\lambda + Ad\xi + Ld\eta + Pd\zeta & \text{(1b)} \\ dy = hvd\lambda + Bd\xi + Md\eta + Qd\zeta & \text{(1c)} \\ dz = hwd\lambda + Cd\xi + Nd\eta + Rd\zeta, & \text{(1d)} \end{cases}$$

where u, v, and w are the x, y, and z components of fluid velocity $\mathbf{q}$, respectively. Let

$$\frac{D_h}{Dt} \equiv \frac{\partial}{\partial t} + hu\frac{\partial}{\partial x} + hv\frac{\partial}{\partial y} + hw\frac{\partial}{\partial z} \tag{2}$$

denote the material derivative following the *pseudo-particle*, whose velocity is $h\mathbf{q}$. Then, it is easy to show that

$$\frac{D_h\xi}{Dt} = 0, \qquad \frac{D_h\eta}{Dt} = 0, \qquad \frac{D_h\zeta}{Dt} = 0; \tag{3}$$

that is, the coordinates (ξ, η, ζ) are material functions of the pseudo-particles, and hence are their permanent identifications. Accordingly, *computational cells move and deform with pseudo-particles*, rather than with fluid particles as in Lagrangian coordinates.

Remarks. (a) Unlike transformations used in grid generation, which are flow-independent, the unique feature of transformation (1) is it depends on the fluid velocity.

(b) In (1), h is an arbitrary function of coordinates $(\lambda, \xi, \eta, \zeta)$. On the other hand, $(A, L, P, B, M, Q, C, N, R)$ are determined by the compatibility conditions. For example, for dx to be a total differential,

$$\begin{cases} \frac{\partial A}{\partial \lambda} = \frac{\partial (hu)}{\partial \xi} & (4a) \\ \frac{\partial L}{\partial \lambda} = \frac{\partial (hu)}{\partial \eta} & (4b) \\ \frac{\partial P}{\partial \lambda} = \frac{\partial (hu)}{\partial \zeta}. & (4c) \end{cases}$$

When (4) are satisfied the other compatibility conditions, namely

$$\begin{cases} \frac{\partial A}{\partial \eta} = \frac{\partial L}{\partial \xi} & (5a) \\ \frac{\partial L}{\partial \zeta} = \frac{\partial P}{\partial \eta} & (5b) \\ \frac{\partial P}{\partial \xi} = \frac{\partial A}{\partial \zeta}, & (5c) \end{cases}$$

are also satisfied, provided that they are at $\lambda = 0$ which can always be ensured in numerical computation. Similar equations and discussions hold for (B, M, Q) and (C, N, R).

(c) In the special case when $h = 0$, $(A, L, \ldots, R)$ are independent of λ. Then the coordinates (ξ, η, ζ) are independent of time λ and are hence fixed in space. This coordinate system is thus Eulerian. Transformation (1) is then flow-independent and is just like any other transformation from Cartesian coordinates (x, y, z) to curvilinear coordinates (ξ, η, ζ) used in grid generation. In particular, if $A = M = R = 1$ and $L = P = B = Q = C = N = 0$, (ξ, η, ζ) are identical with Cartesian coordinates (x, y, z).

(d) In the special case when $h = 1$, on the other hand, the pseudo-particles coincide with fluid particles and (ξ, η, ζ) are the material functions of fluid particles, and hence are Lagrangian coordinates. The conventional choice of the Lagrangian coordinates, i.e., $(\xi, \eta, \zeta) = (x, y, z)|_{t=0}$, is just a special choice of material functions, corresponding to choosing $A = M = R = 1$ and $L = P = B = Q = C = N = 0$. It does not offer any particular advantage in numerical computation; rather (ξ, η, ζ) should better be left to be suitably chosen to initialize numerical computation. In particular, the computational domain in (ξ, η, ζ) space can always be easily made regular, e.g., rectangular, even if it is irregular in the physical space. This cannot be done with the conventional choice of the Lagrangian coordinates.

(e) In the general case, h is arbitrary. It thus provides a new degree of freedom which may be used to advantage: to avoid excessive numerical diffusion in Eulerian coordinates, or to avoid severe grid deformation in Lagrangian coordinates. It will be shown in the next section (see Section 3.2) that for 2-D flow h may be chosen to render the coordinates orthogonal; this would give an optimal grid.

3. EULER EQUATIONS IN THE UNIFIED COORDINATES

The Euler equations in Cartesian coordinates for inviscid flow of an ideal gas obeying the γ-law are

$$\frac{\partial}{\partial t}\begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ \rho e \end{pmatrix} + \frac{\partial}{\partial x}\begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho uw \\ \rho u\left(e + \frac{p}{\rho}\right) \end{pmatrix} + \frac{\partial}{\partial y}\begin{pmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ \rho vw \\ \rho v\left(e + \frac{p}{\rho}\right) \end{pmatrix} + \frac{\partial}{\partial z}\begin{pmatrix} \rho w \\ \rho uw \\ \rho vw \\ \rho w^2 + p \\ \rho w\left(e + \frac{p}{\rho}\right) \end{pmatrix} = 0, \quad (6)$$

where ρ, p, and e are the density, pressure, and specific total energy of the gas, with

$$e = \frac{1}{2}(u^2 + v^2 + w^2) + \frac{1}{\gamma - 1}\frac{p}{\rho}. \tag{7}$$

Under transformation (1), the Euler equations (6) become

$$\frac{\partial \mathbf{E}}{\partial \lambda} + \frac{\partial \mathbf{F}}{\partial \xi} + \frac{\partial \mathbf{G}}{\partial \eta} + \frac{\partial \mathbf{H}}{\partial \zeta} = 0, \tag{8}$$

where

$$\mathbf{E} = \begin{bmatrix} \rho\Delta \\ \rho\Delta u \\ \rho\Delta v \\ \rho\Delta w \\ \rho\Delta e \\ A \\ B \\ C \\ L \\ M \\ N \\ P \\ Q \\ R \end{bmatrix}, \qquad \mathbf{F} = \begin{bmatrix} \rho I \\ \rho I u + p\xi_x\Delta \\ \rho I v + p\xi_y\Delta \\ \rho I w + p\xi_z\Delta \\ \rho I\left(e + \frac{p}{\rho}\right) - p\xi_t\Delta \\ -hu \\ -hv \\ -hw \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$\mathbf{G} = \begin{bmatrix} \rho J \\ \rho J u + p\eta_x\Delta \\ \rho J v + p\eta_y\Delta \\ \rho J w + p\eta_z\Delta \\ \rho J\left(e + \frac{p}{\rho}\right) - p\eta_t\Delta \\ 0 \\ 0 \\ 0 \\ -hu \\ -hv \\ -hw \\ 0 \\ 0 \\ 0 \end{bmatrix}, \qquad \mathbf{H} = \begin{bmatrix} \rho K \\ \rho K u + p\zeta_x\Delta \\ \rho K v + p\zeta_y\Delta \\ \rho K w + p\zeta_z\Delta \\ \rho K\left(e + \frac{p}{\rho}\right) - p\zeta_t\Delta \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ -hu \\ -hv \\ -hw \end{bmatrix} \tag{9}$$

with

$$\Delta = \det\begin{pmatrix} A & L & P \\ B & M & Q \\ C & N & R \end{pmatrix}$$

$$I = \Delta\frac{D\xi}{Dt}, \qquad J = \Delta\frac{D\eta}{Dt}, \qquad K = \Delta\frac{D\zeta}{Dt},$$

and

$$\frac{\partial(\lambda,\xi,\eta,\zeta)}{\partial(t,x,y,z)} = \left(\frac{\partial(t,x,y,z)}{\partial(\lambda,\xi,\eta,\zeta)}\right)^{-1}.$$

We note that:

(a) the system of equations (8) is in conservation form;

(b) the last 9 equations of (8) arise from the compatibility requirements of transformation (1). They are called *geometric conservation laws*, in contrast to the physical conservation laws in the first 5 equations; and

(c) system (8) is larger than system (6) in Eulerian coordinates, as there are now 14 equations for 14 unknowns $\rho, p, u, v, w, A, B, \ldots, R$. However, the additional computing costs for solving (8) are quite small, because the bulk of computing time is spent on solving Riemann problems of the physical conservation laws (see Sections 6–8 below), which are the same for (8) as for (6). In fact, in some cases, such as 2-D steady supersonic flow, it takes less time to solve (8) than to solve (6). (see [15])

As remarked earlier the unified coordinate system is Lagrangian when $h = 1$. In this case system (8) is the equations of motion in Lagrangian coordinates which are now written in conservation form. In this regard, it should be pointed out that it is difficult to write the conventional Lagrangian equations in conservation form except, of course, in the special case of 1-D unsteady flow [19]. In the Appendix we re-write the 2-D conventional Lagrangian equations of motion for inviscid flow into conservation form and show that they are a special case of our system (8) (or its 2-D version (12)) when $h = 1$, as they should.

In the remainder of this paper we shall restrict our discussions to the two-dimensional flow.

3.1. Hyperbolicity of the 2-D Unsteady Euler Equations in the Unified Coordinates

It is well known that the system of unsteady inviscid flow equations (6) in Cartesian coordinates is hyperbolic, meaning that all its eigenvalues are real and there exists a compete set of linearly independent eigenvectors. Because the transformation from (t, x, y, z) to the unified coordinates $(\lambda, \xi, \eta, \zeta)$ involves the dependent variables (u, v, w), there is no guarantee that the resulting system (8) will necessarily be hyperbolic. We now study the hyperbolicity of system (8) in the two-dimensional case.

For two-dimensional unsteady flow, the Euler equations are

$$\frac{\partial}{\partial t}\begin{pmatrix}\rho\\ \rho u\\ \rho v\\ \rho e\end{pmatrix} + \frac{\partial}{\partial x}\begin{pmatrix}\rho u\\ \rho u^2 + p\\ \rho uv\\ \rho u\left(e + \frac{p}{\rho}\right)\end{pmatrix} + \frac{\partial}{\partial y}\begin{pmatrix}\rho v\\ \rho uv\\ \rho v^2 + p\\ \rho v\left(e + \frac{p}{\rho}\right)\end{pmatrix} = 0, \tag{10}$$

where

$$e = \frac{1}{2}(u^2 + v^2) + \frac{1}{\gamma - 1}\frac{p}{\rho}.$$

Under the transformation

$$\begin{cases} dt = d\lambda & \text{(11a)} \\ dx = hud\lambda + Ad\xi + Ld\eta & \text{(11b)} \\ dy = hvd\lambda + Bd\xi + Md\eta & \text{(11c)} \end{cases}$$

we get

$$\frac{\partial \mathbf{E}}{\partial \lambda} + \frac{\partial \mathbf{F}}{\partial \xi} + \frac{\partial \mathbf{G}}{\partial \eta} = 0, \tag{12a}$$

where

$$\mathbf{E} = \begin{pmatrix} \rho\Delta \\ \rho\Delta u \\ \rho\Delta v \\ \rho\Delta e \\ A \\ B \\ L \\ M \end{pmatrix}, \quad \mathbf{F} = \begin{pmatrix} \rho(1-h)I \\ \rho(1-h)Iu + pM \\ \rho(1-h)Iv - pL \\ \rho(1-h)Ie + pI \\ -hu \\ -hv \\ 0 \\ 0 \end{pmatrix}, \quad \mathbf{G} = \begin{pmatrix} \rho(1-h)J \\ \rho(1-h)Ju - pB \\ \rho(1-h)Jv + pA \\ \rho(1-h)Je + pJ \\ 0 \\ 0 \\ -hu \\ -hv \end{pmatrix}, \tag{12b}$$

with

$$\Delta = AM - BL, \qquad I = uM - vL, \qquad J = Av - Bu. \tag{13}$$

We note that the Euler equations (12) written in the unified coordinates are in conservation form.

To study the hyperbolicity of (12), we re-write it as

$$\mathbf{A}\frac{\partial \mathbf{U}}{\partial \lambda} + \mathbf{B}\frac{\partial \mathbf{U}}{\partial \xi} + \mathbf{C}\frac{\partial \mathbf{U}}{\partial \eta} = \mathbf{S}, \tag{14}$$

where

$$\mathbf{U} = (\rho, p, u, v, A, B, L, M)^{\mathrm{T}}$$
$$\mathbf{A} = \frac{\partial \mathbf{E}}{\partial \mathbf{U}}, \qquad \mathbf{B} = \frac{\partial \mathbf{F}}{\partial \mathbf{U}}, \qquad \mathbf{C} = \frac{\partial \mathbf{G}}{\partial \mathbf{U}}$$
$$\mathbf{S} = (0, 0, 0, 0, uh_\xi, vh_\xi, uh_\eta, vh_\eta)^{\mathrm{T}}.$$

System (14) is said to be hyperbolic (also called strongly hyperbolic, or fully hyperbolic) in λ if [20]

(i) all the eigenvalues σ of

$$\det(\sigma\mathbf{A} - \alpha\mathbf{B} - \beta\mathbf{C}) = 0$$

are real for every pair $(\alpha, \beta) \in R^2 : \alpha^2 + \beta^2 = 1$; and

(ii) associated with the eigenvalues there exists a complete set of eight linearly independent right eigenvectors in the state space.

System (14) is said to be weakly hyperbolic in λ if (i) is satisfied but there does not exist a complete set of linearly independent right eigenvectors.

The eigenvalues of (14) can be found using a method similar to [17], and the results are as follows:

Case (a): $h \neq 1$. In this case, we get

$$\begin{aligned} \sigma_1 &= 0 && \text{(multiplicity of 4)} \\ \sigma_2 &= (1-h)(\alpha' u + \beta' v) && \text{(multiplicity of 2)} \\ \sigma_\pm &= \sigma_2 \pm a\sqrt{\alpha'^2 + \beta'^2}, \end{aligned} \tag{15}$$

where a is the speed of sound, and

$$\alpha' = (\alpha M - \beta B)/\Delta, \qquad \beta' = -(\alpha L - \beta A)/\Delta.$$

The corresponding right eigenvectors are

$$\begin{aligned} \mathbf{r}_1 &= (0,0,0,0,1,0,0,0)^{\mathrm{T}} \\ \mathbf{r}_2 &= (0,0,0,0,0,1,0,0)^{\mathrm{T}} \\ \mathbf{r}_3 &= (0,0,0,0,0,0,1,0)^{\mathrm{T}} \\ \mathbf{r}_4 &= (0,0,0,0,0,0,0,1)^{\mathrm{T}} \end{aligned} \tag{16}$$

for σ_1,

$$\begin{aligned} \mathbf{r}_5 &= (0,1,0,0,0,0,0,0)^{\mathrm{T}} \\ \mathbf{r}_6 &= (0,0,b\sigma_2,\sigma_2,-\alpha bh,-\alpha h,-\beta bh,-\beta h)^{\mathrm{T}} \end{aligned} \tag{17}$$

for σ_2, and

$$\mathbf{r}_{7,8} = \left(1, \frac{1}{a^2}, \pm c, \pm d, \mp\frac{\alpha ch}{\sigma_\pm}, \mp\frac{\alpha dh}{\sigma_\pm}, \mp\frac{\beta ch}{\sigma_\pm}, \mp\frac{\beta dh}{\sigma_\pm}\right)^{\mathrm{T}} \tag{18}$$

for $\sigma_\pm$, where

$$b = -\beta'/\alpha', \qquad m = a\sqrt{\alpha'^2 + \beta'^2}, \qquad c = \frac{\alpha'}{\rho m}, \qquad d = \frac{\beta'}{\rho m}.$$

The eigenvectors $\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_8$ are linearly independent, forming a complete basis in the state space; system (14) is therefore hyperbolic for $h \neq 1$. This includes the Eulerian case as special case when $h = 0$.

Case (b): $h = 1$ (Lagrangian Case). In this case the eigenvalues are

$$\begin{aligned} \sigma_1 &= 0 \qquad \text{(multiplicity of 6)} \\ \sigma_\pm &= \pm a\sqrt{\alpha'^2 + \beta'^2}. \end{aligned} \tag{19}$$

The eigenvectors associated with $\sigma_\pm$ are

$$\mathbf{r}_\pm = \left(1, \frac{1}{a^2}, \frac{\alpha'}{\rho\sigma_\pm}, \frac{\beta'}{\rho\sigma_\pm}, \frac{-\alpha\alpha'}{\rho(\sigma_\pm)^2}, \frac{-\alpha\beta'}{\rho(\sigma_\pm)^2}, \frac{-\alpha'\beta}{\rho(\sigma_\pm)^2}, \frac{-\beta\beta'}{\rho(\sigma_\pm)^2}\right)^{\mathrm{T}}. \tag{20}$$

Associated with $\sigma_1 = 0$ (multiplicity of 6),

$$\operatorname{rank}(\sigma\mathbf{A} - \alpha\mathbf{B} - \beta\mathbf{C})\big|_{\sigma=\sigma_1} = 3;$$

hence there exist five, and only five, linearly independent eigenvectors:

$$\begin{aligned} \mathbf{r}_1 &= (0, 0, 0, 0, 1, 0, 0, 0)^{\mathrm{T}} \\ \mathbf{r}_2 &= (0, 0, 0, 0, 0, 1, 0, 0)^{\mathrm{T}} \\ \mathbf{r}_3 &= (0, 0, 0, 0, 0, 0, 1, 0)^{\mathrm{T}} \\ \mathbf{r}_4 &= (0, 0, 0, 0, 0, 0, 0, 1)^{\mathrm{T}} \\ \mathbf{r}_5 &= (0, 1, 0, 0, 0, 0, 0, 0)^{\mathrm{T}}. \end{aligned} \tag{21}$$

We therefore arrive at the conclusion that the system of unsteady 2-D Euler equations of inviscid flow in Lagrangian coordinates *is weakly hyperbolic*, lacking one eigenvector although all eigenvalues are real. This is rather surprising in view of the facts that the system of unsteady Euler equations in Eulerian (Cartesian) coordinates is long known to be hyperbolic and that it has hitherto been taken for granted that the system in Lagrangian coordinates is also hyperbolic. This turns out to be true only in the simple case of one-dimensional unsteady flow [21], but is not true for two-dimensional flow. This degeneracy from hyperbolic to weakly hyperbolic may be traced back to the fact that transformation (11) involves not only the independent variables but also the dependent variables, u and v. In this regard we note that in the case $h \neq 1$, e.g., $h = 1/2$, the transformation (11) also involves u and v but does not lead to degeneracy, as shown in Case (a). So there is something peculiar about $h = 1$, i.e., Lagrangian coordinates. That the 2-D Euler equations in conventional Lagrangian coordinates are weakly hyperbolic is shown, and related to the present formulation, in the Appendix.

The lesson to be learned is that it is insufficient to literally follow fluid particles to describe their motion, as the Lagrangian coordinate system does, because the system of inviscid unsteady flow equations is only weakly hyperbolic. Being only weakly hyperbolic, it does not possess the many desirable properties of a strongly hyperbolic system. For instance, (a) the system cannot be written in characteristic form, rendering the powerful method of characteristics inapplicable; (b) its solution may grow unbounded; (c) the local Riemann problem may have no solution; and (d) the Cauchy problem may be not well posed. Despite these possible defects, some of our computations with $h = 1$ encounter no difficulty and produce results almost identical to that for $h = 0.99$. But this is not guaranteed, and we shall not present computational results for the case $h = 1$. We also note with interest that

some promising work on shock capturing methods for weakly hyperbolic systems has just appeared [22].

In summary, use of Lagrangian coordinates in CFD for two-dimensional unsteady flow not only can cause severe cell deformation but also renders the Euler equations weakly hyperbolic, with all its possible consequences on numerical computation. In this regard, the unified coordinate system with $h \neq 1$ (no matter how close h is to 1), being strongly hyperbolic, is superior to the Lagrangian coordinate system.

Although the hyperbolicity of the system of Euler equations is discussed in this paper only for the case of 2-D unsteady flow, we mention here the corresponding results in other cases:

(a) For 1-D unsteady flow, the system of equations in the unified coordinates is strongly hyperbolic for all values of h [21].

(b) For 3-D unsteady flow, it is strongly hyperbolic for all values of h except for $h = 1$; in the latter case it is only weakly hyperbolic.

(c) For 2-D steady supersonic flow the system of Euler equations resulting from the transformation

$$\begin{cases} dx = hud\lambda + Ad\xi \\ dy = hvd\lambda + Bd\xi \end{cases} \tag{22}$$

is strongly hyperbolic for any $h(\lambda, \xi)$ except when $h = 1$, or $h =$ constant; in the latter cases, it is only weakly hyperbolic.

(d) For 3-D steady supersonic flow, the system of Euler equations resulting from the transformation

$$\begin{cases} dx = hud\lambda + Ad\xi + Ld\eta \\ dy = hvd\lambda + Bd\xi + Md\eta \\ dz = hwd\lambda + Cd\xi + Nd\eta \end{cases} \tag{23}$$

is quite similar to (12), but it is only weakly hyperbolic for any h, although the sub-system representing the physical conservation laws and the sub-system representing the geometric conservation laws are each strongly hyperbolic [17].

3.2. Determination of h

As mentioned earlier, the chief advantage of the unified coordinates is the new degree of freedom in choosing h. Many choices are possible and the simplest one would be to choose a constant value for it. Numerical experiments for constant h will be presented in Section 8 to show its effects on grid deformation and on resolution of flow discontinuities. In general, it is necessary to restrict h to within the range $0 \leq h \leq 1$. For $h > 1$, the eigenvalue σ_2 in (15) has sign opposite to that for $h < 1$, indicating that signals propagate in the wrong direction. Our computations for $h > 1$ break down immediately. On the other hand, for $h < 0$, which means that the pseudo-particles are moving in a direction opposite to that of the fluid particles, computation can be carried out initially but after some finite time it breaks down also. No difficulty has been encountered in all our computations if h is restricted to $0 \leq h < 1$. Our computer code actually also works in many cases for $h = 1$ (recall that the Euler equations are only weakly hyperbolic), producing results which are indistinguishable from results using $h = 0.99$ (for which the Euler equations are strongly hyperbolic).

A good choice for h is to preserve the grid angles in the solution process which marches in λ, i.e.,

$$\frac{\partial}{\partial\lambda}\left[\frac{\nabla\xi}{|\nabla\xi|}\cdot\frac{\nabla\eta}{|\nabla\eta|}\right]=0. \tag{24}$$

Since

$$\begin{aligned}\nabla\xi &= (M,-L)/\Delta\\ \nabla\eta &= (-B,A)/\Delta,\end{aligned} \tag{25}$$

condition (24) becomes

$$\frac{\partial}{\partial\lambda}\left[\frac{AL+BM}{\sqrt{A^2+B^2}\sqrt{L^2+M^2}}\right]=0. \tag{26}$$

By making use of the last four equations of (12), it is easy to show that (26) is equivalent to

$$S^2J\frac{\partial h}{\partial\xi}+T^2I\frac{\partial h}{\partial\eta}=\left[S^2\left(B\frac{\partial u}{\partial\xi}-A\frac{\partial v}{\partial\xi}\right)-T^2\left(M\frac{\partial u}{\partial\eta}-L\frac{\partial v}{\partial\eta}\right)\right]h, \tag{27}$$

where

$$S^2=L^2+M^2,\qquad T^2=A^2+B^2. \tag{28}$$

A consequence of determining h from (27) is that if the grid is orthogonal at $\lambda=0$ it will remain so for subsequent λ. An orthogonal grid is known to possess many desirable properties over non-orthogonal grids, e.g., attaining higher accuracy than non-orthogonal grids.

Computationally, Eq. (27) is to be solved at every time step after the flow variables $\mathbf{Q}=(\rho,p,u,v)^{\mathrm{T}}$ and the geometric variable $\mathbf{K}=(A,B,L,M)^{\mathrm{T}}$ are found. It is thus a first-order linear partial differential equation for $h(\xi,\eta;\lambda)$ with λ appearing as a parameter. To find solution h in the range

$$0\le h\le 1 \tag{29}$$

we note that (27) is linear and homogeneous, and therefore it possesses two properties: (a) positive solution $h>0$ always exists, and (b) if h is a solution to (27) so is h/C, C being any constant. Making use of property (a), we let $g=\ln(hq)$ to get

$$\begin{aligned}&S^2(A\cos\theta-B\sin\theta)\frac{\partial g}{\partial\xi}+T^2(M\cos\theta-L\sin\theta)\frac{\partial g}{\partial\eta}\\ &\quad=S^2\left(B\frac{\partial\cos\theta}{\partial\xi}-A\frac{\partial\sin\theta}{\partial\xi}\right)-T^2\left(M\frac{\partial\cos\theta}{\partial\eta}-L\frac{\partial\sin\theta}{\partial\eta}\right),\end{aligned} \tag{30}$$

where $q=\sqrt{u^2+v^2}$ and θ is the flow angle: $u=q\cos\theta$, $v=q\sin\theta$. Now, if g_1 is any solution to (30) then $h=e^{g_1}/qC$ is a solution to (27) satisfying condition (29), provided that we choose C equal to the maximum of e^{g_1}/q over the whole flow field being computed. The reason to work with $\ln(hq)$ instead of $\ln h$ is that from our experience with steady

flow [15], hq is continuous across slip lines, and hence working with hq can minimize the numerical errors. This is confirmed in our unsteady computation on the four test problems in Section 8.

Numerically, Eq. (30) is solved easily by the method of characteristics if their slopes do not change sign as in example 1; otherwise it is solved by iteration.

We note in passing that an extended Lagrangian method in which streamlines are used as coordinate lines was given in [5] for 2-D unsteady flow, and excellent resolution of slip lines was obtained with asymptotic approach to steady flow over time. However, any streamline coordinate system will encounter difficulties if the initial flow is at rest (see, e.g., example 4 in Section 8) or if there is an interior stagnation point, because the transformation would be singular there and the unknown functions become multi-valued. Indeed, the general transformation from Cartesian coordinates (x, y) to streamline coordinates (ξ, η) is

$$\begin{cases} dt = d\lambda \\ dx = Ld\lambda + hud\xi + Ad\eta \\ dy = Md\lambda + hvd\xi + Bd\eta, \end{cases} \tag{31}$$

where h is arbitrary. Clearly $\eta = \text{const.}$ corresponds to an instantaneous streamline. The Jacobian of this transformation is equal to $h(uB - vA)$, which vanishes at stagnation points, rendering the functions multi-valued. In our system, the coordinate lines are pathlines of the pseudo-particles which avoid these difficulties, and yield excellent resolution of slip lines as seen in Section 8. Indeed, the Jacobian of our transformation is $\Delta = AM - BL$, which is the area of the computational cell in the physical plane and is never zero in our computations.

4. BOUNDARY CONDITIONS AND RESOLUTION OF DISCONTINUITIES

In this section we point out some advantages of the unified coordinate system over those of the Eulerian in the following three aspects:

4.1. Boundary Conditions on Solid Boundaries

Consider a time-independent solid boundary (this includes steady flow as a special case)

$$S: \qquad B(x, y, z) = 0. \tag{32}$$

The boundary condition on it is

$$\mathbf{q} \cdot \nabla B = 0 \qquad \text{on } S; \tag{33}$$

hence

$$h\mathbf{q} \cdot \nabla B = 0 \qquad \text{on } S. \tag{34}$$

Equation (33) implies that fluid particles move on S, whereas (34) implies that pseudo-particles also move on S. Therefore, S is a material function of the pseudo-particles. Consequently, $B(x, y, z)$ can be taken to correspond to one of the coordinates, ξ_0 say. In other words, a coordinate surface in the unified coordinate system can be taken to represent a

time-independent solid surface and there is no need for a grid generation prior to flow computation, as is needed if Eulerian coordinates are used.

4.2. Slip Line Resolution

In steady flow, pathlines are identical with streamlines. Hence a slip line coincides with the streamline of a fluid particle and, therefore, also with the streamline of a pseudo-particle. Consequently, it can be taken to correspond to one of the coordinates, ξ^* say, thus avoiding the Godunov averaging across it. Hence, in the unified coordinate system a slip line can be sharply resolved. This is in direct contrast to the Eulerian coordinates where a slip line does not coincide with a coordinate line and, as a result, the Godunov averaging across a slip line in a computational cell will forever smear it.

For unsteady flow, pathlines are in general distinct from streamlines. While a slip line still coincides with the pathline of a fluid particle, it does not always coincide with a streamline. Hence, a slip line does not always coincide with a coordinate line in the unified coordinate system. In this regard, numerical experiments (Section 8) clearly indicate the trend that slip line resolution increases with increasing h from $h = 0$ (Eulerian) to $h = 1$ (Lagrangian) and the unified coordinates using grid-angle preserving h, Eq. (27), yield better slip line resolution than the Eulerian coordinates. Furthermore, if a steady flow is computed as an asymptotic state of unsteady flow for large time, sharp resolution of slip lines is achieved when h is determined by (27), which at the same time avoids severe grid deformation.

4.3. Shock Resolution

In using the unified coordinate system for flow computation, once the grid is set initially it is subsequently generated by the motion of the pseudo-particles. In this regard, it is interesting to note that the pseudo-particles, which move parallel to the fluid particles, tend to crowd together when compressed, resulting in automatic refinement of the grid in the compression region. Consequently, shock resolution is improved in the unified coordinates over the Eulerian. Moreover, the improvements increase with increasing shock strength.

5. SOLUTION STRATEGIES

From the above discussion, we see that the system of Euler equations (12) is potentially superior to its counterpart (10) using Eulerian coordinates in slip line resolution, especially for steady flow. Furthermore, with h determined by the grid-angle preserving condition (27), it can avoid the severe grid deformation encountered in the Lagrangian coordinates.

As the system of Euler equations (12) is in conservation form, any well-established shock capturing method can be used to solve it. We shall use the Godunov method with the MUSCL update to higher resolution to solve system (12). The computation will be done entirely in the λ–ξ–η space. A physical cell in the x–y plane marching along the pseudo-particle's pathline corresponds to a rectangular cell in the ξ–η plane marching in the λ direction in the computational space λ–ξ–η. The superscript k refers to the marching time step number and the subscripts i and j refer to the cell index number on a time plane $\lambda = \text{const}$. The time step $\Delta\lambda^k = \lambda^{k+1} - \lambda^k$ is uniform for all i and j, but is always chosen to satisfy the CFL stability condition. The grid divides the computational domain into cuboid control volumes, or cells, which in the ξ and η direction are centered at $(\lambda^k, \xi_i, \eta_j)$ and

have widths $\Delta\xi_i = \xi_{i+1/2} - \xi_{i-1/2}$ and $\Delta\eta_j = \eta_{j+1/2} - \eta_{j-1/2}$ (for all k). Unless otherwise stated we shall use uniform cell width $\Delta\xi_i$ for all i and $\Delta\eta_j$ for all j.

In the physical space (t, x, y) a cuboid cell marching in (λ, ξ, η) space corresponds to a pseudo-particle marching along its path tube with step Δt ($\Delta t = \Delta\lambda$). The pseudo-particle is bounded by four path surfaces $\xi = \xi_{i\pm1/2}$ and $\eta = \eta_{j\pm1/2}$ around it. Initially, any curvilinear coordinate grid on the x–y plane may be used as the ξ–η coordinate grid and the initial geometric variables $\mathbf{K} = (A, B, L, M)^{\mathrm{T}}$ can be determined from (11) as part of the initial conditions. A stationary solid wall is always a path surface of the fluids and hence also of the pseudo-fluids; it is therefore a coordinate surface.

We shall apply the Godunov scheme [23] with MUSCL update [24] to solve (12). Applying the divergence theorem to (12) over the cuboid cell (i, j, k) results in

$$\mathbf{E}_{i,j}^{k+1} = \mathbf{E}_{i,j}^{k} - \frac{\Delta\lambda^k}{\Delta\xi_i}\left(\mathbf{F}_{i+1/2,j}^{k+1/2} - \mathbf{F}_{i-1/2,j}^{k+1/2}\right) - \frac{\Delta\lambda^k}{\Delta\eta_j}\left(\mathbf{G}_{i,j+1/2}^{k+1/2} - \mathbf{G}_{i,j-1/2}^{k+1/2}\right), \qquad i = 1, 2, \ldots, m;\ j = 1, 2, \ldots, n, \tag{35}$$

where the notation for the cell average of any quantity f is

$$f_{i,j}^{k} = \frac{1}{\Delta\xi_i\,\Delta\eta_j}\int_{\xi_{i-1/2}}^{\xi_{i+1/2}}\int_{\eta_{j-1/2}}^{\eta_{j+1/2}} f(\lambda^k, \xi, \eta)\,d\xi\,d\eta, \tag{36}$$

and the notation for time λ average of f is

$$f_{i+1/2,j}^{k+1/2} = \frac{1}{\Delta\lambda^k}\int_{\lambda^k}^{\lambda^{k+1}} f(\lambda, \xi_{i+1/2}, \eta_j)\,d\lambda, \tag{37}$$

$$f_{i,j+1/2}^{k+1/2} = \frac{1}{\Delta\lambda^k}\int_{\lambda^k}^{\lambda^{k+1}} f(\lambda, \xi_i, \eta_{j+1/2})\,d\lambda. \tag{38}$$

According to Godunov's idea, the cell interface fluxes $\mathbf{F}_{i+1/2,j}^{k+1/2}$ and $\mathbf{G}_{i,j+1/2}^{k+1/2}$ for the cell (i, j) should be obtained from the self-similar solution of a local two-dimensional Riemann problem formed by the averaged constant state $\mathbf{Q}_{i,j} = (\rho, p, u, v)_{i,j}^{T}$ of the cell (i, j) and those of its adjacent cells. Unfortunately, such a solution to (12) is unavailable at the present time. Indeed, even a 2-D Riemann solution to the simpler system (10), which is a special case of (12) when $h = 0$, is not yet available. On the other hand, it is known that a monotone difference scheme to a general conservation form converges to the physically relevant entropy-satisfying solution. In particular, Crandall and Majda [25] establish the rigorous convergence for dimensional splitting algorithms when each step is approximated by a monotone difference scheme (such as the Godunov scheme) for a single conservation law of multi-dimension.

In view of the above, we shall numerically solve (12) using a Godunov-type scheme based on the following strategies: a time step-wise Eulerian approximation to decouple the geometric conservation laws from the physical conservation laws, and a dimensional splitting approximation to reduce the two-dimensional flow problem to two one-dimensional flow problems. These are explained as follows.

5.1. The Time Step-Wise Eulerian (TSE) Approximation

The essence of TSE is that while solving the physical conservation laws (the first four equations of (12)) for the flow variables $\mathbf{Q} = (\rho, p, u, v)^{\mathrm{T}}$ in the time step of λ from λ^k to λ^{k+1}, the geometric variables $\mathbf{K} = (A, B, L, M)^{\mathrm{T}}$ and h are kept unchanged with λ but are in general functions of ξ and η; hence the effects of cell shapes (grid) on the flow are accounted for in a time-frozen manner. More precisely, in solving the physical conservation laws in $\Omega^k(\lambda) : \lambda^k < \lambda \le \lambda^{k+1}$, we use $\mathbf{K} = \mathbf{K}(\lambda^k, \xi, \eta)$ and $h = h(\lambda^k, \xi, \eta)$. After obtaining the solution $\mathbf{Q}(\lambda, \xi, \eta)$, $\lambda \in \Omega^k(\lambda)$, we update the geometric conservation laws (the last four equations of (12)) to get $\mathbf{K}(\lambda^{k+1}, \xi, \eta)$ (this is a rather trivial step) and then solve (27) to get $h(\lambda^{k+1}, \xi, \eta)$ as explained in Section 4. In this way the effects of the flow on the cell shapes are taken into account. This completes the advancing of solution for one time step from $\lambda = \lambda^k$ to $\lambda = \lambda^{k+1}$ and the process can be repeated to advance the solution for the next time step.

Physically, the TSE idea is equivalent to temporarily freezing the shape of the fluid particles over $\Omega^k(\lambda)$ while the flow field evolves. Mathematically, the problem of solving the physical conservation laws over $\Omega^k(\lambda)$ keeping $\mathbf{K}$ and h frozen is equivalent to that of solving the Euler equations in fixed curvilinear coordinates (ξ, η) with coefficients in the governing equations varying in ξ and η. The Riemann problem in the curvilinear coordinates is more difficult than that in Cartesian coordinates but is solvable as will be explained in Section 6.

At this point it is necessary and possible to comment on the equivalence of the weak solution of the extended system (12) to that of the Eulerian system (10). As shown in Section 3.1 the extended system has an additional eigenvalue $\sigma_1 = 0$ (multiplicity of 4) corresponding to the geometric conservation laws. Since this eigenfield is linearly degenerated, it might be anticipated that the solution of the extended system possesses a new slip line, in addition to the slip line corresponding to σ_2 of Eq. (15). We note that this is *not* the case for one-dimensional flow [21] but, unfortunately, no theoretical result is available at present for the two-dimensional case under consideration. On the other hand, in using the TSE approximation to solve the extended system, this additional eigenfield does not come into play (because the geometric variables $\mathbf{K}$ and h are treated as given) and the extended system is truncated and reduced to the Eulerian system in curvilinear coordinates. Therefore, the weak solution of the extended system as obtained by the time step-wise Eulerian method is equivalent to the weak solution of the Eulerian system. Any differences between the two solutions must arise from the different grids used, and it is the purpose of this paper to show that such differences are indeed very significant; see Section 8.

5.2. Dimensional Splitting Approximation

The dimensional splitting technique for finding an approximate solution to the Riemann problem in multi-dimensional flow is now well established and used widely. This technique renders the solution of a multi-dimensional problem to a sequential solution of several one-dimensional problems. The Godunov splitting and the Strang splitting [26] are frequently used in practical applications. Theoretically, if the time accuracy of the one-dimensional solution is of the first order, both of these two splitting techniques are also first-order time accurate. But our numerical test on the two-dimensional Riemann problem (the first test example) shows that the Strang splitting gives more accurate results. Thus we shall use

the Strang splitting in this paper. Let $\mathcal{L}^{\xi}_{\Delta\lambda}$ represent the exact solution operator for the 1-D equation in the λ–ξ plane and $\mathcal{L}^{\eta}_{\Delta\lambda}$ similarly defined; then

$$\mathbf{Q}^{k+1} = \mathcal{L}^{\xi}_{\frac{\Delta\lambda}{2}} \mathcal{L}^{\eta}_{\Delta\lambda} \mathcal{L}^{\xi}_{\frac{\Delta\lambda}{2}} \mathbf{Q}^{k}, \tag{39}$$

where $\Delta\lambda = \lambda^{k+1} - \lambda^{k}$.

The solution operator $\mathcal{L}^{\xi}_{\Delta\lambda}$ for the Riemann problem with variable coefficients in the governing equations in the λ–ξ plane will now be given in detail.

6. THE RIEMANN SOLUTION IN THE λ–ξ PLANE

Based on the solution strategies explained in the last section, the key step is the solution to the 1-D Riemann problem over the time step $\Omega^k(\lambda) : \lambda^k < \lambda \le \lambda^{k+1}$ resulting from dimensional splitting and the time step-wise Eulerian approximation.

In this section, we explain how to derive the 1-D Riemann solution in the λ–ξ plane, in particular the flow variable $\mathbf{Q}$ at the interface $\xi = 0$ for $\lambda \in \Omega^k(\lambda)$. The 1-D Riemann problem in the λ–η plane can be obtained similarly.

From (12), at time step λ^k (to be taken as 0 for simplicity) the 1-D physical conservation law equations in the λ–ξ plane resulting from dimensional splitting are

$$\frac{\partial \mathbf{E}_p}{\partial \lambda} + \frac{\partial \mathbf{F}_p}{\partial \xi} = 0, \qquad \lambda \in \Omega(\lambda) : 0 < \lambda \le \Delta\lambda, \tag{40a}$$

where

$$\mathbf{E}_p = \begin{pmatrix} \rho\Delta \\ \rho\Delta u \\ \rho\Delta v \\ \rho\Delta e \end{pmatrix}, \qquad \mathbf{F}_p = \begin{pmatrix} \rho(1-h)I \\ \rho(1-h)Iu + pM \\ \rho(1-h)Iv - pL \\ \rho(1-h)Ie + pI \end{pmatrix} \tag{40b}$$

with

$$\Delta = AM - BL, \qquad I = uM - vL, \qquad e = \frac{1}{2}(u^2 + v^2) + \frac{1}{\gamma - 1}\frac{p}{\rho}. \tag{41}$$

In (40), the physical variables $\mathbf{Q} = (\rho, p, u, v)^{\mathrm{T}}$ are regarded as (unknown) functions of λ and ξ while the geometric variables $\mathbf{K} = (A, B, L, M)^{\mathrm{T}}$ and h, which appear in the equations' coefficients, are independent of λ, i.e.,

$$\mathbf{K} = \mathbf{K}(0, \xi), \qquad h = h(0, \xi). \tag{42}$$

η in (40) is treated as a parameter. In applying the Godunov scheme to advance the solution from $\lambda = 0$ to $\lambda = \Delta\lambda$, the initial data for the adjacent cells (i, j) and $(i+1, j)$ are the following Riemann (constant) data (for simplicity we take the cell interface between these two cells to be located at $\xi = 0$):

$$\mathbf{Q}|^{\lambda=0} = \begin{cases} \mathbf{Q}_\ell \left(= \mathbf{Q}^{\lambda=0}_{i,j}\right), & \xi < 0 \\ \mathbf{Q}_r \left(= \mathbf{Q}^{\lambda=0}_{i+1,j}\right), & \xi > 0. \end{cases} \tag{43}$$

At the same time, based on the time step-wise Eulerian approximation, the coefficients in Eqs. (40) are

$$(\mathbf{K}, h)|^{\lambda\in\Omega(\lambda)} = \begin{cases} (\mathbf{K}, h)_\ell^{\lambda=0} \big(=(\mathbf{K}, h)_{i,j}^{\lambda=0}\big), & \xi < 0 \\ (\mathbf{K}, h)_r^{\lambda=0} \big(=(\mathbf{K}, h)_{i+1,j}^{\lambda=0}\big), & \xi > 0. \end{cases} \tag{44}$$

We note that these coefficients are constants separately for $\xi < 0$ and $\xi > 0$, but are in general not equal to each other.

To put the Riemann problem in the λ–ξ plane more explicitly in one-dimensional form, we note that the normal direction of the plane $\xi =$ constant is

$$\mathbf{n} = \frac{\nabla\xi}{|\nabla\xi|} = (M, -L)/S \tag{45}$$

and project the flow velocity $\mathbf{q}$ into the normal direction $\mathbf{n}$ and the tangential direction $\mathbf{t}$ to get

$$\begin{cases} \omega = \mathbf{q}\cdot\mathbf{n} = (uM - vL)/S \\ \tau = \mathbf{q}\cdot\mathbf{t} = (uL + vM)/S. \end{cases} \tag{46}$$

We also replace (L, M) by S and ψ as

$$\begin{cases} S = \sqrt{L^2 + M^2} \\ \tan\psi = M/L. \end{cases} \tag{47}$$

We shall now transform (40) for $\xi < 0$ and for $\xi > 0$, separately. For $\xi < 0$, $(\mathbf{K}, h) = (\mathbf{K}, h)_\ell$ are constant. Hence $\Delta = \Delta_\ell$, $S = S_\ell$, and $\psi = \psi_\ell$ are also constant, and Eqs. (40) become

$$\frac{\partial \mathbf{E}'_\ell}{\partial\lambda} + \frac{\partial \mathbf{F}'_\ell}{\partial\xi} = 0, \qquad \lambda \in \Omega(\lambda), \xi < 0, \tag{48a}$$

where

$$\mathbf{E}'_\ell = \Delta_\ell \begin{pmatrix} \rho \\ \rho\omega \\ \rho\tau \\ \rho e \end{pmatrix}, \qquad \mathbf{F}'_\ell = S_\ell \begin{pmatrix} \rho(1-h_\ell)\omega \\ \rho(1-h_\ell)\omega^2 + p \\ \rho(1-h_\ell)\omega\tau \\ \rho(1-h_\ell)\omega e + \omega p \end{pmatrix}. \tag{48b}$$

Similarly, for $\xi > 0$ Eqs. (40) become

$$\frac{\partial \mathbf{E}'_r}{\partial\lambda} + \frac{\partial \mathbf{F}'_r}{\partial\xi} = 0, \qquad \lambda \in \Omega(\lambda), \ \xi > 0 \tag{49a}$$

with

$$\mathbf{E}'_r = \Delta_r \begin{pmatrix} \rho \\ \rho\omega \\ \rho\tau \\ \rho e \end{pmatrix}, \qquad \mathbf{F}'_r = S_r \begin{pmatrix} \rho(1-h_r)\omega \\ \rho(1-h_r)\omega^2 + p \\ \rho(1-h_r)\omega\tau \\ \rho(1-h_r)\omega e + \omega p \end{pmatrix}. \tag{49b}$$

These equations (48) and (49) are in the same form as the system

$$\frac{\partial}{\partial t}\begin{pmatrix}\rho\\ \rho u\\ \rho v\\ \rho e\end{pmatrix}+\frac{\partial}{\partial x}\begin{pmatrix}\rho u\\ \rho u^2+p\\ \rho uv\\ \rho ue+up\end{pmatrix}=0 \tag{50}$$

obtained from the Euler equations in Cartesian coordinates after dimensional splitting if we equate ω with u and τ with v; hence they can be solved by a similar method. We note that in each of the systems (48), (49), or (50) the coefficients are constant over $\Omega(\lambda)$ and the variable v (or τ) can be decoupled. We also note that the Riemann problem consisting of (48), (49), and initial condition (43) has a new feature—and hence new difficulty—in that the coefficients, though constant, in general are different for $\xi<0$ as for $\xi>0$.

6.1. Special Case: $(\mathbf{K}, h)_\ell = (\mathbf{K}, h)_r$

We consider first the special case when the constants are equal, i.e.,

$$(\mathbf{K}, h)_\ell^{\lambda=0} = (\mathbf{K}, h)_r^{\lambda=0} \equiv (\mathbf{K}, h)^{\lambda=0}. \tag{51}$$

In this case, (48) and (49) are identical and become

$$\frac{\partial \mathbf{E}'}{\partial \lambda}+\frac{\partial \mathbf{F}'}{\partial \xi}=0, \qquad \lambda\in\Omega(\lambda), \tag{52a}$$

where

$$\mathbf{E}'=\Delta\begin{pmatrix}\rho\\ \rho\omega\\ \rho\tau\\ \rho e\end{pmatrix}, \qquad \mathbf{F}'=S\begin{pmatrix}\rho(1-h)\omega\\ \rho(1-h)\omega^2+p\\ \rho(1-h)\omega\tau\\ \rho(1-h)\omega e+\omega p\end{pmatrix} \tag{52b}$$

and Δ, S, and h are constant.

Equations (52) become, after decoupling the tangential velocity component τ, the conventional Riemann problem for a 1-D unsteady flow,

$$\begin{cases}\dfrac{\partial \mathbf{E}''}{\partial \lambda}+\dfrac{\partial \mathbf{F}''}{\partial \xi}=0, & \lambda\in\Omega(\lambda) \qquad (53a)\\ (\rho, p, \omega)|^{\lambda=0}=\begin{cases}(\rho, p, \omega)_\ell, & \xi<0\\ (\rho, p, \omega)_r, & \xi>0,\end{cases} & (53b)\end{cases}$$

where

$$\mathbf{E}''=\Delta\begin{pmatrix}\rho\\ \rho\omega\\ \rho e_\perp\end{pmatrix}, \qquad \mathbf{F}''=S\begin{pmatrix}\rho(1-h)\omega\\ \rho(1-h)\omega^2+p\\ \rho(1-h)\omega e_\perp+\omega p\end{pmatrix} \tag{53c}$$

with

$$e_\perp=\frac{1}{2}\omega^2+\frac{1}{\gamma-1}\frac{p}{\rho}. \tag{53d}$$

The tangential velocity component τ can be found , after solving (53), from

$$\frac{\partial(\rho\Delta\tau)}{\partial\lambda} + \frac{\partial(S(1-h)\rho\omega\tau)}{\partial\xi} = 0, \tag{54}$$

which can be simplified to, after using the first equation of (53a),

$$\begin{cases} \dfrac{\partial\tau}{\partial\lambda} + \dfrac{(1-h)S}{\Delta}\omega\dfrac{\partial\tau}{\partial\xi} = 0 & \text{(55a)} \\ \tau = \begin{cases} \tau_\ell, & \xi < 0 \\ \tau_r, & \xi > 0. \end{cases} & \text{(55b)} \end{cases}$$

Now since Δ, S, and h are constant in (53), the Riemann problem (53) can be solved in exactly the same way as solving the 1-D unsteady flow equations of gasdynamics. As usual, the physical entropy condition that the entropy of a fluid particle shall not decrease on crossing a shock is imposed to select the physically correct solution. The solution consists of four uniform flow regions separated by three non-linear singular waves: a shock, a slip line, and an expansion wave, with the slip line situated in between the shock and the expansion wave (Fig. 1).

The solution to the Riemann problem for $0 \le h < 1$ is now given in detail:

(a) *The Eigenfields*

The eigenvalues of Eq. (52) are $(0 \le h < 1)$

$$\begin{aligned} \sigma_2 &= \frac{(1-h)S}{\Delta}\omega \qquad \text{(multiplicity of 2)} \\ \sigma_\pm &= \frac{S}{\Delta}[(1-h)\omega \pm a]. \end{aligned} \tag{56}$$

Their corresponding right eigenvectors are

$$\begin{aligned} \mathbf{r}_{1,1} &= (0, 1, 0, 0)^{\mathrm{T}} \\ \mathbf{r}_{1,2} &= (0, 0, 0, 1)^{\mathrm{T}} \end{aligned} \tag{57}$$

for σ_2, and

$$\mathbf{r}_\pm = \left(\frac{1}{a^2}, 1, \pm\frac{1}{a\rho}, 0\right)^{\mathrm{T}} \tag{58}$$

for $\sigma_\pm$. It is easy to see that the eigenfield σ_2 is linearly degenerated, whereas the eigenfields $\sigma_\pm$ are genuinely non-linear.

(b) *Smooth Solutions*

The smooth solutions for the eigenfields $\sigma_\pm$ are determined from

$$\begin{cases} \frac{d\rho}{dp} = \frac{1}{a^2} \\ \frac{d\omega}{dp} = \pm\frac{1}{a\rho} \\ \frac{d\tau}{dp} = 0. \end{cases} \tag{59}$$

The solution for ρ, ω, and τ relates the flow state $\mathbf{Q} = (\rho, p, \omega, \tau)^{\mathrm{T}}$ in the expansion fan

to the initial state $\mathbf{Q}_0 = (\rho_0, p_0, \omega_0, \tau_0)^{\mathrm{T}}$ upstream of the fan. This solution can be easily found and is most conveniently given in terms of the pressure ratio $\alpha = p/p_0$ as

$$\begin{cases} \rho = \rho_0 \alpha^{\frac{1}{\gamma}} \\ \omega = \omega_0 \pm \frac{2a_0}{\gamma - 1}\left(\alpha^{\frac{\gamma-1}{2\gamma}} - 1\right) \\ \tau = \tau_0, \end{cases} \tag{60}$$

where $a_0 = \sqrt{\gamma p_0/\rho_0}$. Note that τ does not change across an expansion fan and that Eqs. (60) are identical to those of the purely 1-D unsteady flow; in particular, they are independent of $\mathbf{K}$ and h.

Let (λ, ξ) be a general point inside the expansion fan. The slope of the characteristic is given by

$$\frac{d\xi}{d\lambda} = \frac{\xi}{\lambda} = \sigma_{\pm}. \tag{61}$$

The solution for flow inside the fan is

$$\begin{cases} p = p_0 \left\{ \frac{2(1-h)}{\gamma - 2h + 1} \pm \frac{\gamma - 1}{(\gamma - 2h + 1)a_0} \left((1-h)\omega_0 - \frac{\Delta}{S}\frac{\xi}{\lambda} \right) \right\}^{\frac{2\gamma}{\gamma-1}} \\ \rho = \rho_0 \alpha^{\frac{1}{\gamma}} \\ \omega = \omega_0 \pm \frac{2a_0}{\gamma - 1}\left(\alpha^{\frac{\gamma-1}{2\gamma}} - 1\right) \\ \tau = \tau_0. \end{cases} \tag{62}$$

If we put $h = 0$ in (62), we recover the solution as obtained in the Eulerian coordinates, as it should.

For discontinuous solutions, we start from the Rankine–Hugoniot conditions for (52),

$$\begin{cases} c\Delta[\rho] = (1-h)S[\rho\omega] \\ c\Delta[\rho\omega] = S[(1-h)\rho\omega^2 + p] \\ c\Delta[\rho\tau] = S(1-h)[\rho\omega\tau] \\ c\Delta[\rho e] = S[(1-h)\rho\omega e + \omega p], \end{cases} \tag{63}$$

where $[\cdot\,]$ denotes the jump across the discontinuity whose speed is denoted by $c = d\xi/d\lambda$.

(c) *Shock Waves*

We denote the pre-shock (upstream) flow state by $\mathbf{Q}_0 = (\rho_0, p_0, \omega_0, \tau_0)^{\mathrm{T}}$ and the post-shock (downstream) flow state by $\mathbf{Q} = (\rho, p, \omega, \tau)^{\mathrm{T}}$, respectively. Then the shock jump relations can be expressed in terms of $\alpha = p/p_0$ as follows:

$$\begin{cases} \rho = \rho_0 \frac{\alpha(\gamma+1) + \gamma - 1}{\alpha(\gamma-1) + \gamma + 1} \\ \omega = \omega_0 \pm \frac{a_0(\alpha - 1)}{\sqrt{\frac{1}{2}\gamma(\alpha(\gamma+1) + \gamma - 1)}} \\ \tau = \tau_0. \end{cases} \tag{64}$$

Again, we see that τ does not jump across a shock and that Eqs. (64) are identical to those of the purely 1-D unsteady flow; in particular they are independent of $\mathbf{K}$ and h.

(d) *Slip Lines*

In this case, we get

$$\begin{cases} p = p_0 \\ \omega = \omega_0 \end{cases} \tag{65}$$

but the density jump and tangential velocity jump are arbitrary. Once again, we note that (65) are identical to the purely 1-D flow; and in particular they are independent of **K** and h.

In summary, we note that for the flow variables $\mathbf{Q} = (\rho, p, \omega, \tau)^{\mathrm{T}}$ their relations across a shock (64), across a slip line (65), and across an expansion wave (60) hold separately in their regions $\xi < 0$ or $\xi > 0$ and are all independent of the values of the geometric variables $\mathbf{K} = (A, B, L, M)^{\mathrm{T}}$ and h, provided that the expansion wave lies entirely in the region $\xi < 0$ or entirely in $\xi > 0$. On the other hand, shock speed, slip line speed, and the structure of the flow inside the expansion fan (62), e.g., fan width and location, are dependent on the values of **K** and h. Such dependence would be needed to construct the complete Riemann solution for $\lambda \in \Omega(\lambda)$ and for all ξ values. But, in using the Godunov scheme to advance the solution from $\lambda = 0$ to $\lambda = \Delta\lambda$, *we need only the flow variables* **Q** *at the cell interface* $\xi = 0$ (to compute the flux $\mathbf{F}^{k+\frac{1}{2}}_{i+\frac{1}{2},j}$) which are entirely independent of the values of **K** and h and are continuous across the interface, provided that the expansion wave lies entirely in $\xi < 0$ or $\xi > 0$.

For instance, to find $\mathbf{Q}|_{\xi=0}$ in $\Omega(\lambda)$, we consider the generic case shown in Fig. 1. We start by assuming a value p^* for pressure at region 3, i.e., $p_3 = p^*$; then on the one hand $p_2 = p^*$ and from $\mathbf{Q}_\ell$ and p_2 we can determine $\mathbf{Q}_2$ in region 2 (we use (64) if $p_\ell < p^*$, and (60) if $p_\ell \geq p^*$). On the other hand, from $\mathbf{Q}_r$ and p_3 we can determine $\mathbf{Q}_3$ in region 3 (we use (64) if $p_r < p^*$, and (60) if $p_r > p^*$); we then compare ω_2 with ω_3: if $\omega_2 = \omega_3$, the initial guess p^* is the correct value for pressure in region 3 and **Q** at the interface is completely determined. If $\omega_2 \neq \omega_3$, we go back to adjust p^* until $\omega_2 = \omega_3$ is reached.

This process is formally done using the Newton method of iteration to find the roots of the non-linear equation.

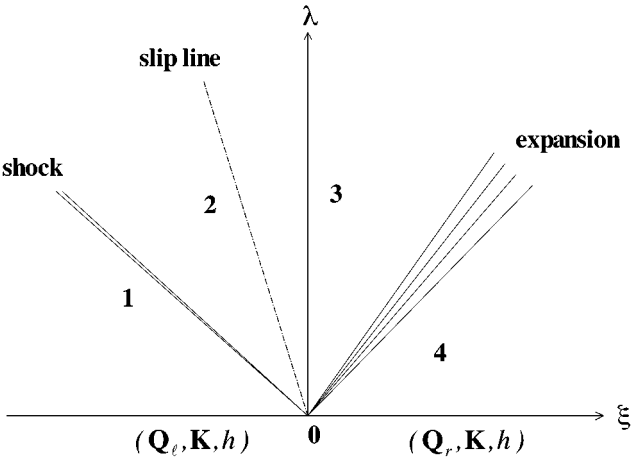


FIG. 1. Generic structure of Riemann solution: special case.

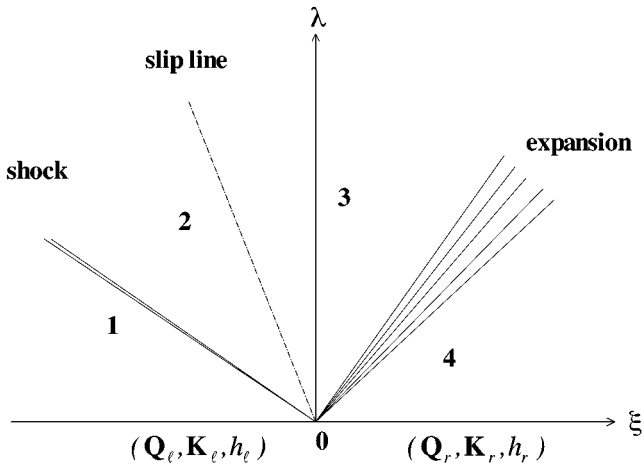


FIG. 2. Generic structure of Riemann solution: general case.

6.2. General Case: $(\mathbf{K}, h)_\ell \neq (\mathbf{K}, h)_r$

This case is sketched in Fig. 2. It is the Riemann problem that would arise in a purely Eulerian computation if the space coordinates were not Cartesian but were curvilinear.

In this case, we first solve the problem with the data

$$(\mathbf{Q}, \mathbf{K}, h)^{\lambda=0} = \begin{cases} (\mathbf{Q}_\ell, \mathbf{K}_\ell, h_\ell), & \xi < 0 \\ (\mathbf{Q}_r, \mathbf{K}_\ell, h_\ell), & \xi > 0. \end{cases} \tag{66}$$

The problem is solved as explained in Section 6.1 above. In particular, the flow variables $\mathbf{Q}$ at the interface $\xi = 0$ are independent of $\mathbf{K}_\ell$ and h_ℓ, but are completely determined by $\mathbf{Q}_\ell$ and $\mathbf{Q}_r$. Now, when we change $(\mathbf{K}_\ell, h_\ell)$ for $\xi > 0$ to $(\mathbf{K}_r, h_r)$, the flow variables $\mathbf{Q}$ at $\xi = 0$ are not changed since they are independent of the geometric variables $\mathbf{K}$ and h (as noted in Section 6.1), provided that the expansion wave lies entirely in the region $\xi < 0$ or in the region $\xi > 0$. In the rare case when the expansion fan covers the interface $\xi = 0$, we see from (62) that the pressure there depends on h and we make an additional approximation that $h = \frac{1}{2}(h_\ell + h_r)$ is used for calculating p in (62). In this way we obtain the Riemann solution for $\mathbf{Q}$ at the interface.

To summarize, in computing the interface flow variables $\mathbf{Q}$ at $\xi = 0$ for the Godunov flux the geometric variables $\mathbf{K}$ and h on the two sides, $\xi < 0$ and $\xi > 0$, are never used and only a conventional 1-D unsteady flow Riemann problem is solved, whose solution is completely determined by the initial data of the flow field, i.e., $\mathbf{Q}_\ell$ and $\mathbf{Q}_r$ (at $\lambda = 0$)

7. NUMERICAL PROCEDURE

The numerical procedure of the Godunov/MUSCL scheme can now be summarized as follows:

Step 1: Initialization. Assume that the initial conditions of a flow problem are given at $t = 0$ ($\lambda = 0$) in the x–y plane. Then an appropriate ξ–η coordinate grid is laid on the

x–y plane (for instance, we take ξ and η equal to the arclength of their corresponding coordinate line on the x–y plane), with $\xi = \xi_0, \xi_1, \xi_2, \ldots, \xi_m$, $\eta = \eta_0, \eta_1, \eta_2, \ldots, \eta_n$, and the curve $\xi = \xi_0$ (or $\eta = \eta_0$) coinciding with the solid surface if there is one. Hence $\mathbf{K}^0_{i,j}$ as well as the flow variable $\mathbf{Q}^0_{i,j} = (\rho^0, p^0, u^0, v^0)^{\mathrm{T}}_{i,j}$ is obtained by averaging the given flow over the computational cell (i, j). They are used together with $h^0_{i,j} = 0$ as initial conditions. Subsequently, $\mathbf{E}^0_{i,j}$, $i = 1, 2, \ldots, m$, $j = 1, 2, \ldots, n$, are available. For example, if we choose ξ and η to be the respective arclengths of the x and y coordinate lines then, from (7), $\mathbf{K}^0_{i,j} = (1, 0, 0, 1)^{\mathrm{T}}$ and $\mathbf{E}^0_{i,j}$ follow from its expressions in (12b).

Step 2: The operation $\mathcal{L}^{\xi}_{\Delta\lambda}$ for marching from λ^k to $\lambda^{k+1} = \lambda^k + \Delta\lambda$, $k = 0, 1, 2, \ldots$. We first take

$$(\mathbf{K}_{i,j}(\lambda), h_{i,j}(\lambda)) = \left(\mathbf{K}_{i,j}(\lambda^k), h_{i,j}(\lambda^k)\right) \tag{67}$$

to be constant over the inverval $\lambda^k < \lambda < \lambda^{k+1}$. Then for every pair of adjacent cells (i, j) and $(i + 1, j)$,

(1) Do a MUSCL type data reconstruction in a component by component manner. For example, in the ξ direction, let f be any of the above physical variables ρ, p, u, and v; then, instead of assuming a uniform state in the cells (i, j) and $(i + 1, j)$, we assume linearly distributed states and use linear extrapolation to determine cell interface flow variables: $f_r = f_{i+1,j} - 0.5(f_{i+2,j} - f_{i+1,j})\phi(r^+)$ with $r^+ = (f_{i+1,j} - f_{i,j})/(f_{i+2,j} - f_{i+1,j})$ and $f_\ell = f_{i,j} + 0.5(f_{i,j} - f_{i-1,j})\phi(r^-)$ with $r^- = (f_{i+1,j} - f_{i,j})/(f_{i,j} - f_{i-1,j})$, where $\phi(r) = \max(0, \min(1, r))$ is the minmod flux limiter and subscripts r and ℓ of f correspond to right and left states, respectively.

(2) Define the normal direction of the cell interface $\xi_{i+\frac{1}{2},j}$ between the two adjacent cells (i, j) and $(i + 1, j)$ as

$$\mathbf{n} = \frac{(\nabla\xi)_{i,j} + (\nabla\xi)_{i+1,j}}{|(\nabla\xi)_{i,j} + (\nabla\xi)_{i+1,j}|}, \tag{68}$$

i.e., the average of $(\nabla\xi)_{i,j}$ and $(\nabla\xi)_{i+1,j}$. Project the velocity vector $\mathbf{q} = (u, v)$ into the normal and the tangential components (ω and τ) using Eq. (46).

(3) Solve the Riemann problem of (53) as explained in Sections 6.1 and 6.2 to get the interfacial flow variables $(\rho, p, \omega, \tau)^{\mathrm{T}}$ and hence $(\rho, p, u, v)^{\mathrm{T}}$ at $\xi = \xi_{i+\frac{1}{2},j}$. These are constants and will be denoted by $(\cdot)_{i+\frac{1}{2},j}$.

(4) Update $\mathbf{K}^k_{i,j}$ to $\mathbf{K}^{k+1}_{i,j}$ as follows:

$$\begin{aligned}
\begin{pmatrix} A^{k+1}_{i,j} \\ B^{k+1}_{i,j} \end{pmatrix} &= \begin{pmatrix} A^k_{i,j} \\ B^k_{i,j} \end{pmatrix} + \frac{\Delta\lambda^k}{\Delta\xi_i} h^k_{i,j} \begin{pmatrix} u_{i+\frac{1}{2},j} - u_{i-\frac{1}{2},j} \\ v_{i+\frac{1}{2},j} - v_{i-\frac{1}{2},j} \end{pmatrix} \\
\begin{pmatrix} L^{k+1}_{i,j} \\ M^{k+1}_{i,j} \end{pmatrix} &= \begin{pmatrix} L^k_{i,j} \\ M^k_{i,j} \end{pmatrix}.
\end{aligned} \tag{69}$$

(5) Calculate the first four components of the cell interface flux. For instance, the secondcomponent of the interface flux $\mathbf{F}^{k+\frac{1}{2}}_{i+\frac{1}{2},j}$ is evaluated as

$$\rho_{i+\frac{1}{2},j}\left(1 - h^k_{i,j}\right)\left(u_{i+\frac{1}{2},j} M^{k+1}_{i,j} - v_{i+\frac{1}{2},j} L^{k+1}_{i,j}\right) + p_{i+\frac{1}{2},j} M^{k+1}_{i,j}. \tag{70}$$

(6) Update the conserved variables $\mathbf{E}_p$ in the physical conservation laws (40) using

$$\mathbf{E}_{p_{i,j}}^{k+1} = \mathbf{E}_{p_{i,j}}^{k} - \frac{\Delta\lambda^k}{\Delta\xi_i}\left(\mathbf{F}_{i+\frac{1}{2},j}^{k+\frac{1}{2}} - \mathbf{F}_{i-\frac{1}{2},j}^{k+\frac{1}{2}}\right). \tag{71}$$

(7) Decode $\mathbf{E}_{p_{i,j}}^{k+1}$ to get $\mathbf{Q}_{i,j}^{k+1}$, using $\Delta = A_{i,j}^{k+1}M_{i,j}^{k+1} - B_{i,j}^{k+1}L_{i,j}^{k+1}$.

(8) Apply Strang splitting, Eq. (39), to advance $\mathbf{Q}_{i,j}^{k}$ to $\mathbf{Q}_{i,j}^{k+1}$.

(9) Update $h_{i,j}^{k}$ to $h_{i,j}^{k+1}$ by solving Eq. (27), using the updated values $\mathbf{Q}_{i,j}^{k+1}$ and $\mathbf{K}_{i,j}^{k+1}$ in its coefficients. (Note: This step (9) is, of course, to be bypassed if $h = \text{const}$ is assumed in the computation.)

(10) Calculate the grid in the x–y plane at λ^{k+1}:

$$\begin{cases} x_{i,j}^{k+1} = x_{i,j}^{k} + \frac{1}{2}\left(h_{i,j}^{k}u_{i,j}^{k} + h_{i,j}^{k+1}u_{i,j}^{k+1}\right)\Delta\lambda \\ y_{i,j}^{k+1} = y_{i,j}^{k} + \frac{1}{2}\left(h_{i,j}^{k}v_{i,j}^{k} + h_{i,j}^{k+1}v_{i,j}^{k+1}\right)\Delta\lambda. \end{cases} \tag{72}$$

By a grid we mean the lines joining the cell centers, not the cell interface lines.

We remark that the grid in the physical plane is not used in the subsequent computation (only the values of $\mathbf{K}$ are used) as the whole computation is carried out in the transformed plane (the ξ–η plane). So, this step (10) is optional. However, the grid information is useful in computing steady flow as an asymptotic state of unsteady flow for large λ. In this case to determine if a steady state is reached, which means the flow at every fixed location in the x–y plane does not change with increasing time, we should compare the flow variables $\mathbf{Q}$ at the same fixed point (x, y) in the physical plane and not at the same points (ξ, η) in the transformed plane; the latter are simply the pseudo-particles whose positions in the x–y plane in general move with λ and never reach an asymptotic state.

After this, we repeat Step 2 to advance the solution further to λ^{k+2}, and so on.

8. TEST EXAMPLES

In this section, the unified coordinates approach is tested numerically on four examples. Two of them are unsteady flows and the other two are steady flows which are computed as asymptotic states of unsteady flow for large time. $\gamma = 1.4$ is used in all the cases. The numerical results are then compared with the exact solutions, experimental results, or other Euler solver's solutions wherever available. In addition, example 3 is chosen for the grid convergence test. In all the cases, the effects of h on the computational robustness and accuracy are discussed.

The first example is a two-dimensional steady Riemann problem generated by two uniform parallel flows as

$$(p, \rho, M, \theta) = \begin{cases} (0.25, 0.5, 7, 0), & y > 0 \\ (1, 1, 2.4, 0), & y < 0, \end{cases}$$

where M is the Mach number and θ the flow angle, $\theta = \tan^{-1}(v/u)$. The flow contains a shock wave, a slip line, and an expansion wave (Fig. 3). The slip line is sensitive to the dissipative property of the numerical methods. Since the analytical solution for the

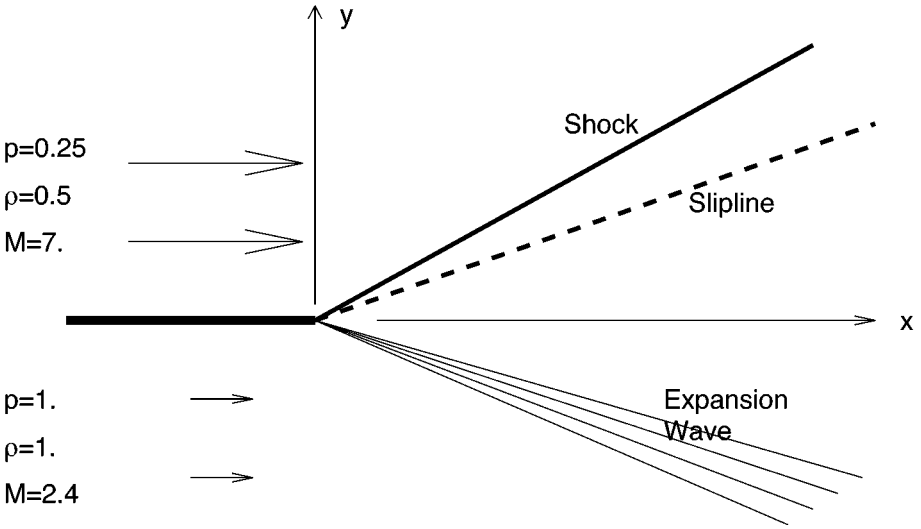


FIG. 3. Sketch of a steady Riemann problem.

problem is available, it is an excellent benchmark problem for the verification of numerical methods. In the computation, the steady flow is achieved with time marching until the flow structure and the variables do not change with time. A grid of 60×100 with $\Delta\xi = \Delta\eta = 0.01$ is employed in the computation. Initially, a grid with $\Delta x = \Delta y = 0.01$ in the physical plane is laid over a domain of $\{0 \le x \le 0.6, -0.5 \le y \le 0.5\}$. The initial data are given at each cell according to its position in $y > 0$ or $y < 0$, representing cell-average values. The physical domain will change with time according to the pseudo-particle's velocity $h\mathbf{q}$ if h is not zero. If we follow the computational cells (pseudo-particles), they will move out of the initial physical domain, and it would be difficult to have a steady state of flow in the original physical domain. To avoid this, a special technique called the "motionless viewing window" is applied as in the classical Lagrangian method. Accordingly, the column of cells which have moved out of the original physical domain to the right is deleted, while a new column of cells is added at the input flow boundary on the left.

For this problem, we first compute the flow by the well-known solver CLAW developed by R. J. LeVeque based on Eulerian coordinates. Figure 4 shows its density distribution compared with the exact solution. It is seen that the slip line is badly smeared and the computed density has a dip near the slip line.

In Figs. 5a to 5d we show computed density using our unified code for $h = 0$, $h = 0.25$, $h = 0.5$, and $h = 0.999$, again compared with the exact solution. We see that the result for $h = 0$ (Eulerian coordinates) is similar to those of LeVeque (Fig. 4), except the dip is now somewhat less severe. This could be attributed to the fact that we use the exact Riemann solution, whereas LeVeque uses Roe's approximate Riemann solution. However, the very poor resolution of the slip line is a common feature of any method based on Eulerian coordinates as a result of Godunov averaging across slip lines which, in general, do not coincide with (Eulerian) coordinate lines. A comparison of Figs. 5a to 5d also shows that the slip line resolution improves with increasing h from $h = 0$ to $h = 0.999$, as expected. It is worth noting that even when h is small, the slip line resolution is much better than

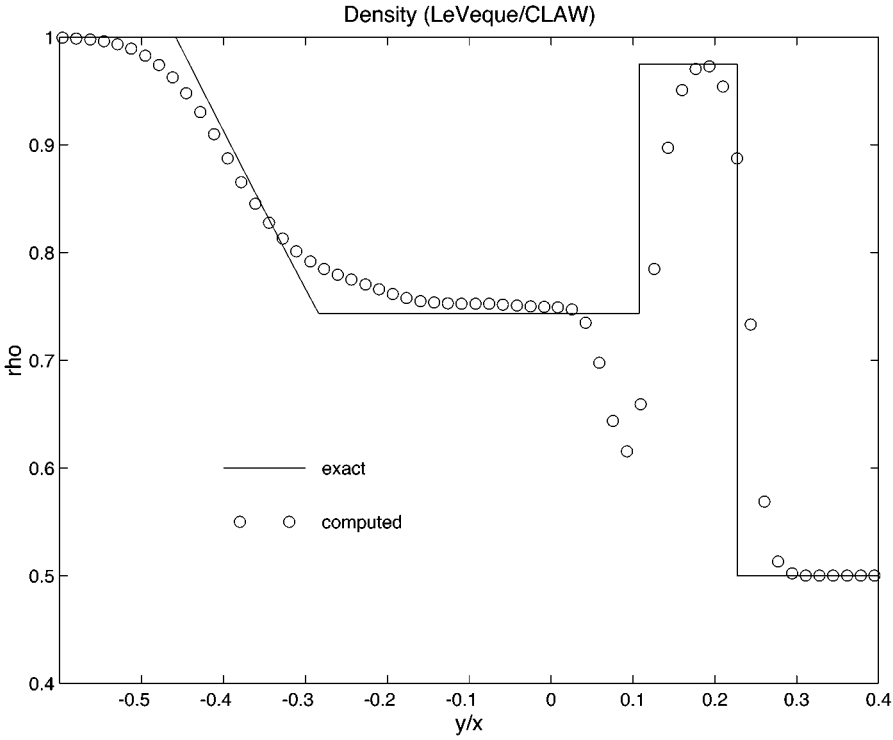


FIG. 4. Density distribution in a steady Riemann problem computed by LeVeque's CLAW code.

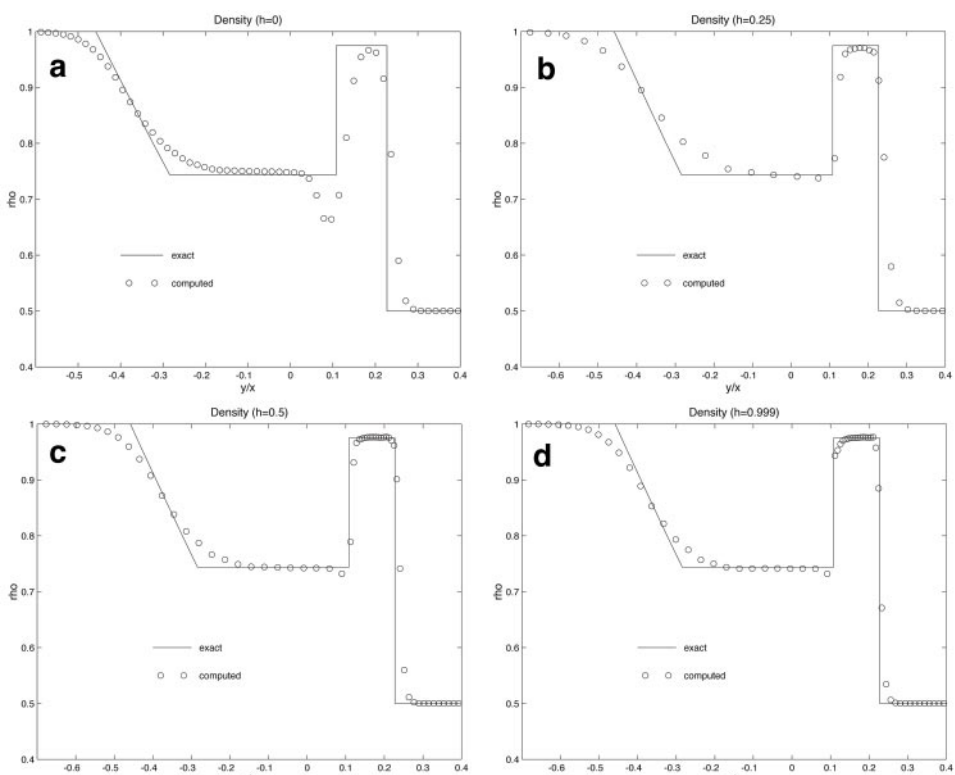


FIG. 5. Density distribution in a steady Riemann problem computed by the present unified code, (a) $h = 0$ (Eulerian), (b) $h = 0.25$, (c) $h = 0.5$, (d) $h = 0.999$.

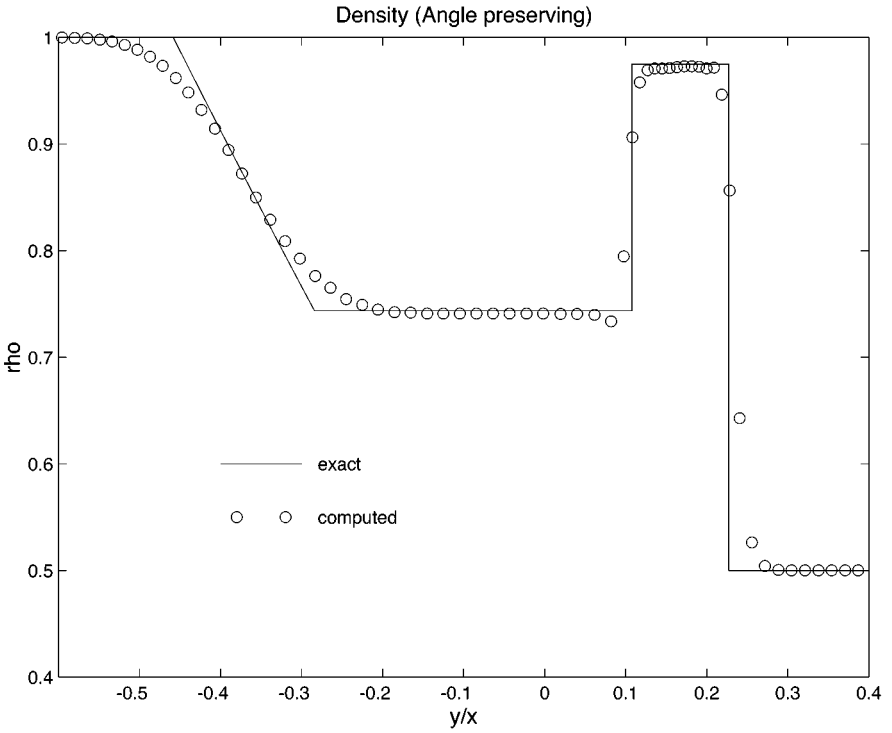


FIG. 6. Density distribution in a steady Riemann problem computed by the present unified code with h chosen to preserve grid angles, Eq. (27).

that using Eulerian coordinates. This is because the flow is steady and the slip lines coincide with the streamlines which, in turn, coincide with the gird lines, thus avoiding the Godunov averaging across slip lines, as pointed out earlier in Section 4.2. The computing times for these four cases are the same and are approximately equal to that for the CLAW code.

Figure 6 shows the computed density using the grid-angle preserving h as determined by Eq. (27), which is solved at each time step using the method of characteristics. While its slip line resolution is seen as less sharp than that for $h = 0.999$, its predicted density in the uniform flow region between the shock and the slip line is better. The computing time is about 1–2% more than that required in a Eulerian code or in the $h = \text{const}$ cases. The bulk of computing time is spent on solving the Riemann problems, and the excessive computing time is spent on solving Eq. (27) for h.

All the computations started with the Eulerian grid (Fig. 7a). The flow-generated grids, i.e., the lines joining the cell centers, at steady state are shown in Figs. 7b to 7d. We note that: (a) the grid using grid-angle preserving h is everywhere orthogonal, (b) a seemingly small change from the initial grid (Fig. 7a) to the final grid (Fig. 7d) has resulted in great improvement in computational accuracy (compare Fig. 6 with Fig. 5a), and (c) the grids for $h = 0.5$ and $h = 0.999$ are severely deformed near the slip line, and such grid deformation causes inaccuracy locally, as seen in Fig. 5d.

Finally, Fig. 8 shows the computed density using the steady flow code of Hui and Chu [15]. It is clearly the best result and requires much less computing time. Its sharp resolution

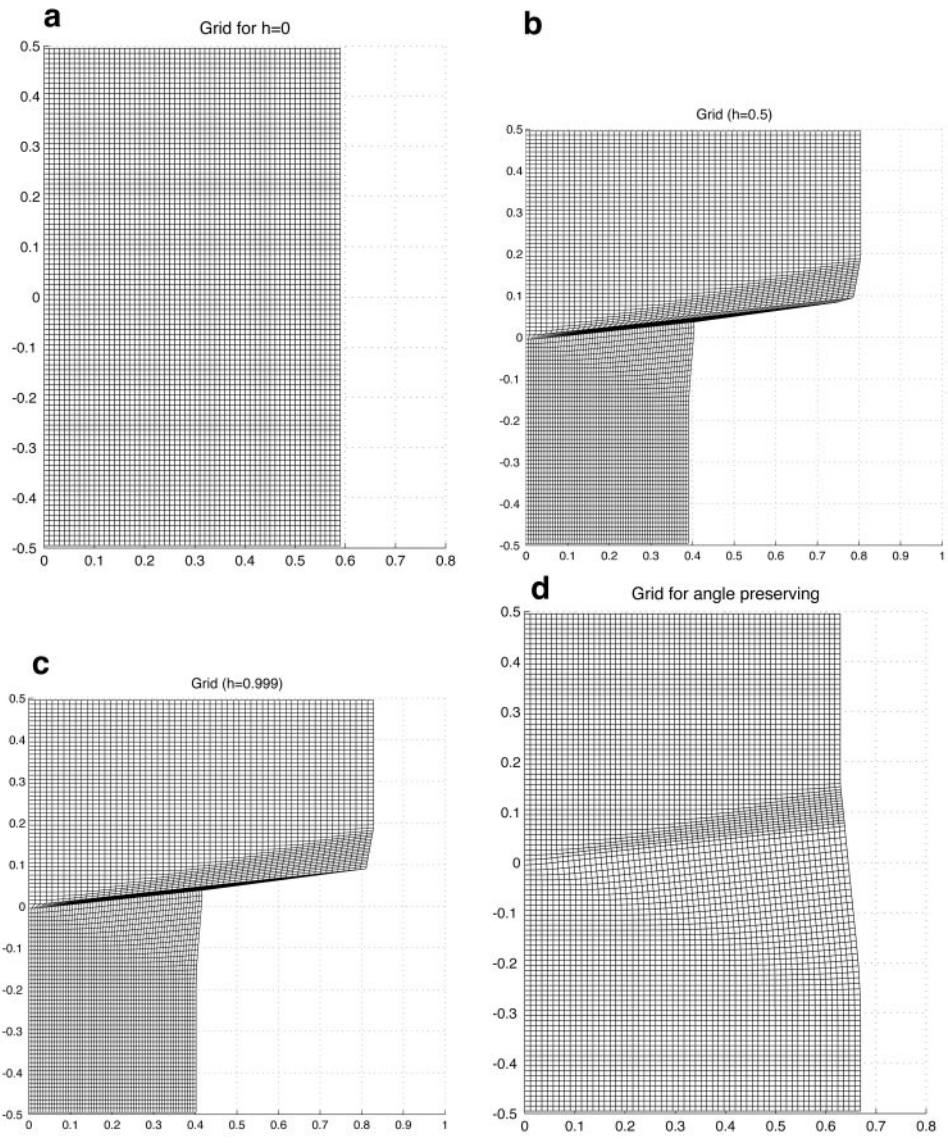


FIG. 7. (a) Grid for $h = 0$ (Eulerian), also initial grid for all cases in steady Riemann problem. (b) Flow-generated grid in a steady Riemann problem, $h = 0.5$. (c) Flow-generated grid in steady Riemann problem, $h = 0.999$. (d) Flow-generated grid in steady Riemann problem, h chosen to preserve grid angles, Eq. (27).

of slip line is a consequence of using pseudo-particle coordinates, but its sharp resolution of shock wave is the result of applying an adaptive Godunov scheme. However, the steady code [15] is applicable only to purely supersonic and steady flow.

The second example is the supersonic flow passing through a channel with a ramp segment. A ramp of 15° is located at the bottom wall between $x = 0.5$ and $x = 1$. The top wall and the other part of the bottom wall connecting the ramp are flat and parallel to each other (Fig. 9). When a flow of $M = 1.8$ passes through the channel, an oblique shock, a Mach stem, a slip line, and reflected shocks are generated. The computational grid is 180×50 with $\Delta\xi = \Delta\eta = 0.02$. Initially, let $\Delta x = \Delta\xi$, and Δy is

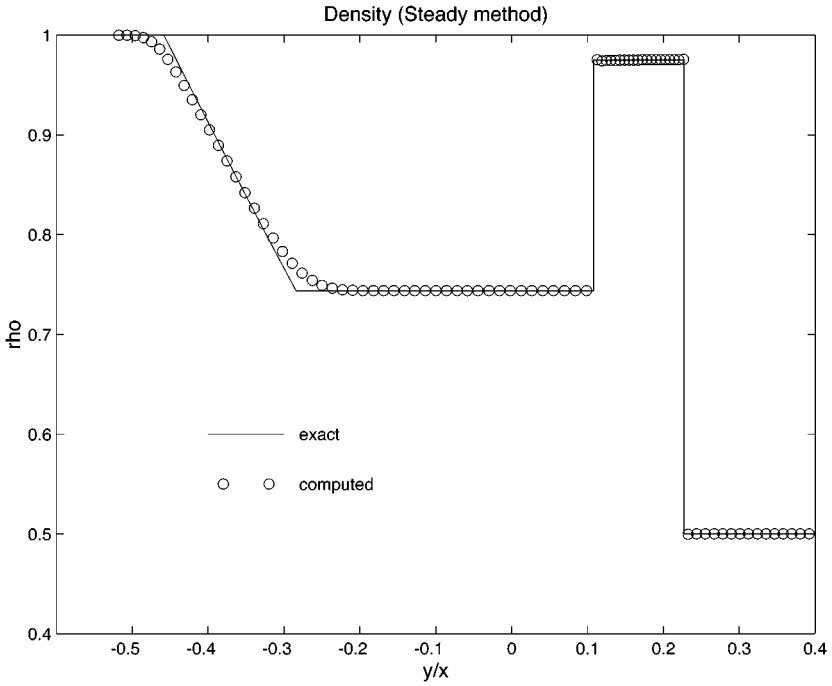


FIG. 8. Density distribution in steady Riemann problem computed by the steady code of Hui and Chu [15].

calculated according to the distance between the top and bottom wall divided by the grid number in the η direction. The initial grid system is shown in Fig. 9. A physical domain $\{0 \le x \le 3.6,\ 0 \le y \le 1.\}$ is given initially and the *motionless viewing window* technique is applied. The initial flow data $(p, \rho, M, \theta) = (1, 1, 1.8, 0)$ are given at each cell. This flow is also imposed as the boundary condition at the inflow boundary, while at the outflow boundary a zero-gradient condition is imposed. The flow approaches its steady state asymptotically with increased time. We test two situations. First, we take $h = 0.999$. Figures 10a to 10c give the pressure and Mach number contours and the flow-generated grid at steady state. Although the grid is relatively coarse, all the flow features are well captured: the Mach stem is about 20% of the inlet height; the oblique shock wave, the corner expansion waves, and the reflected shock waves between two flat walls are all well resolved. Particularly, the slip line stem from the triple point is captured clearly.

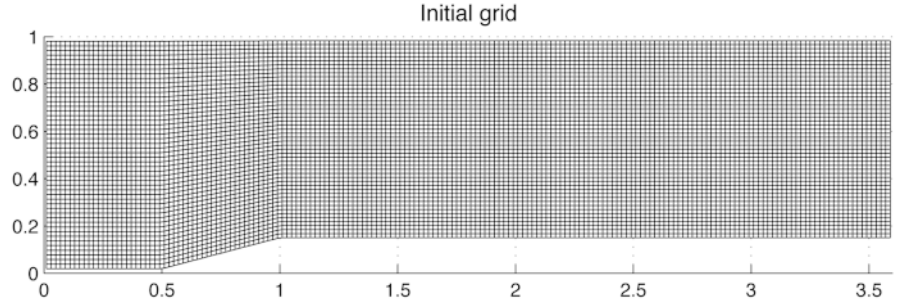


FIG. 9. Initial grid for channel flow problem.

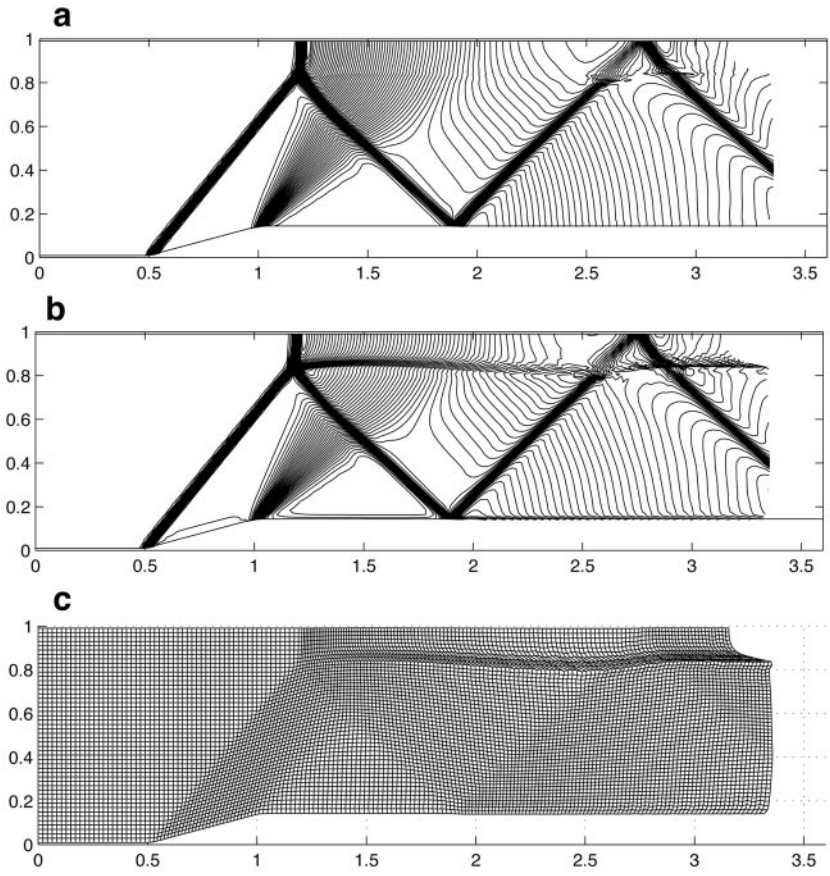


FIG. 10. Computed steady channel flow using the present unified code with $h = 0.999$. (a) Pressure contours, (b) Mach contours, and (c) flow-generated grid.

However, the pressure near the slip line is not smooth enough and this is due to the severe grid deformation there as seen in Fig. 10c. A notch at the upper right corner of Fig. 10c is due to the slowdown of the flow behind the Mach stem. Second, a grid-angle preserving method is applied to compute the channel flow (Fig. 11). The h equation (27) is solved in an iterative way. It is noticed that the flow-generated grid is orthogonal everywhere. This is because the grid at the inlet is orthogonal, and the grid angles are preserved while they move downstream with the pseudo-particles. The flow structure is as well captured as in the case $h = 0.999$. What is more, the pressure is smooth at the slip line, as it should be.

The third example is the Mach reflection of a shock wave from a wedge. It is an unsteady flow: a plane shock of $M = 1.3$ moves from left to right across a wedge of 25°, generating Mach reflection. The initial grid of 200×100 with $\Delta\xi = 0.01$ and $\Delta\eta = 0.0075$ is laid similarly as in the second example. The initial flow state is given as $(p, \rho, u, v) = (1/1.4, 1, 0, 0)$ everywhere apart from that at the input boundary, where the flow state is $(p, \rho, u, v) = (1.2893, 1.5157, 0.44231, 0)$. We take h to be 0, 0.999, and grid-angle preserving. The Mach number contours at $t = 1.25$ are shown in Figs. 12a to 12c. While the flow features are well captured in all three cases, the slip line stemming from the

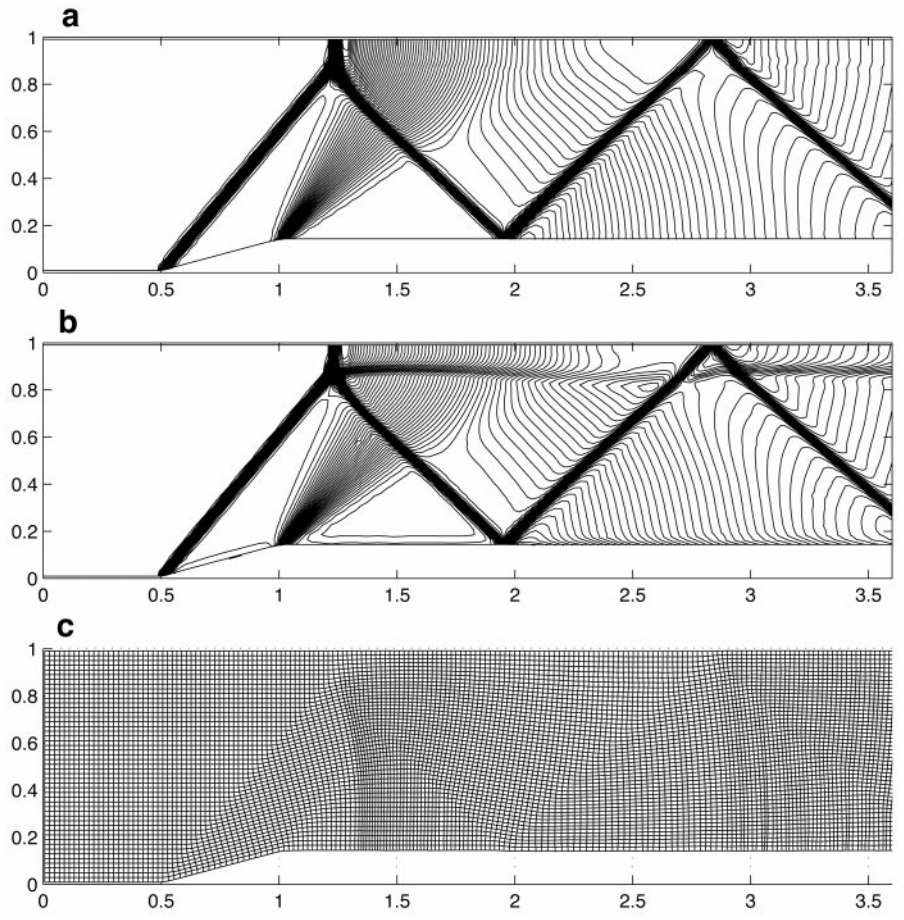


FIG. 11. Computed steady channel flow using the present unified code with h chosen to preserve grid angles, Eq (27). (a) Pressure contours, (b) Mach contours, and (c) flow-generated grid.

triple point is a little more smeared with $h = 0$ than the other two cases. With $h = 0.999$, the normal shock is sharper, because the grid automatically becomes denser near the shock.

To demonstrate the convergence of the computed results as the grid is refined, we use a finer grid (400×200) and the grid-angle preserving h and compare our results with experiment (Fig. 13a) [27]. The same code is run to $t = 1.25$. The shock wave and the slip line become sharper (Fig. 13b) with the refinement of the grid. In addition, they all agree with experimental observation.

The last example is an interesting implosion/explosion problem. It is an unsteady flow in a two-dimensional container. Inside the container, the gases at rest are separated into two regions with a square diaphragm (Fig. 14). The centers of these two squares coincide. At $t = 0$, the diaphragm is ruptured, and the inner and the outer gases begin to interact with each other. Since the flow is confined with solid walls, it will be reflected from the walls continuously and become more and more complex. In our test, we choose the initial flow state as follows: for the inner region, $(p, \rho, u, v) = (0.14, 0.125, 0, 0)$, and for

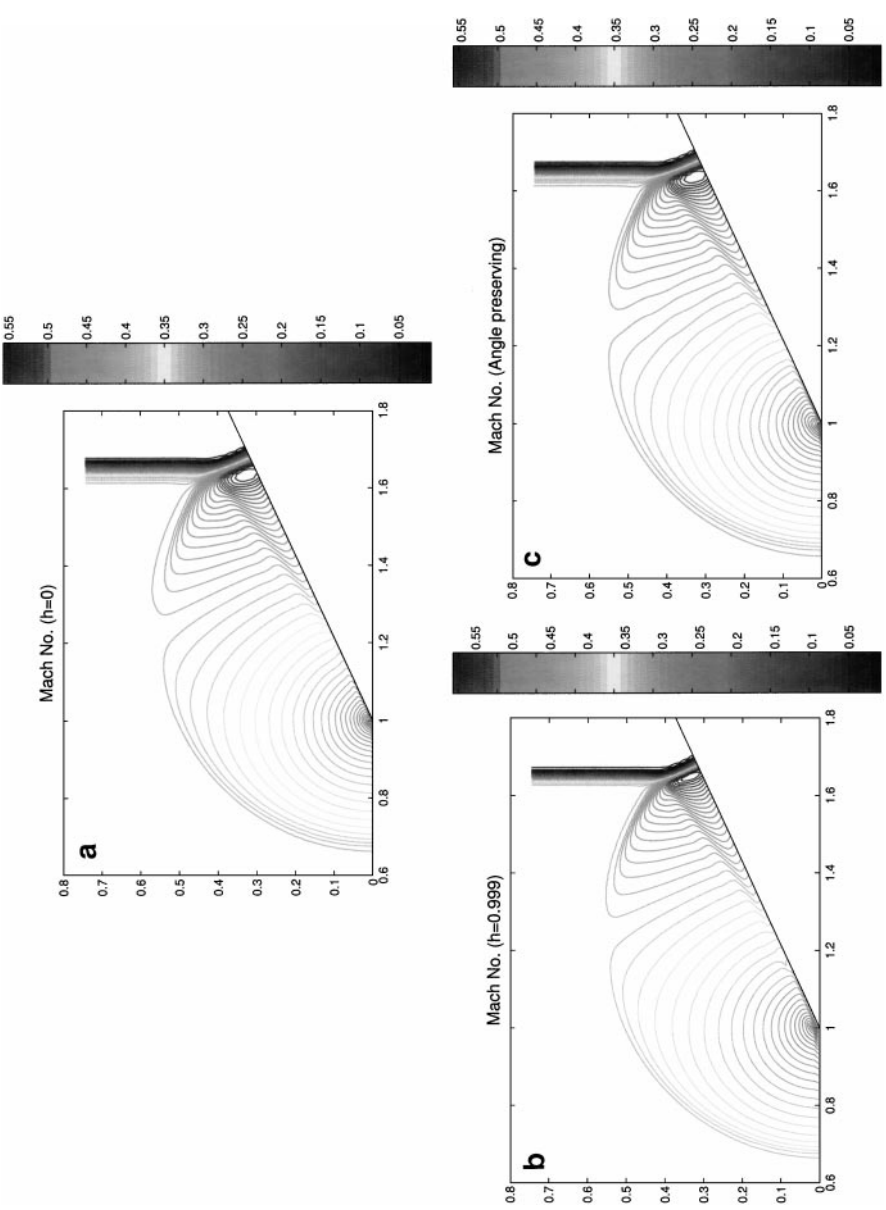
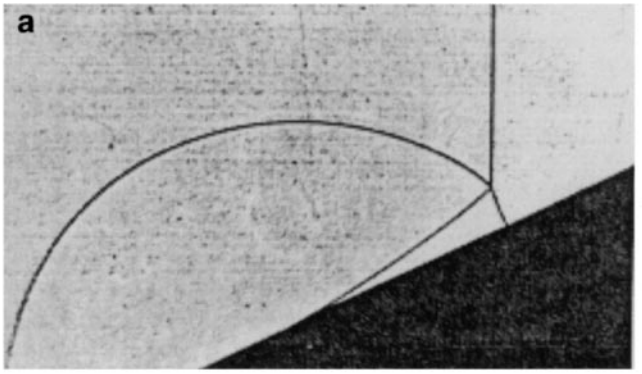


FIG. 12. Mach contours in a Mach reflection problem computed using the present unified code with (a) $h = 0$ (Eulerian), (b) $h = 0.999$, (c) h chosen to preserve grid angles, Eq. (27).



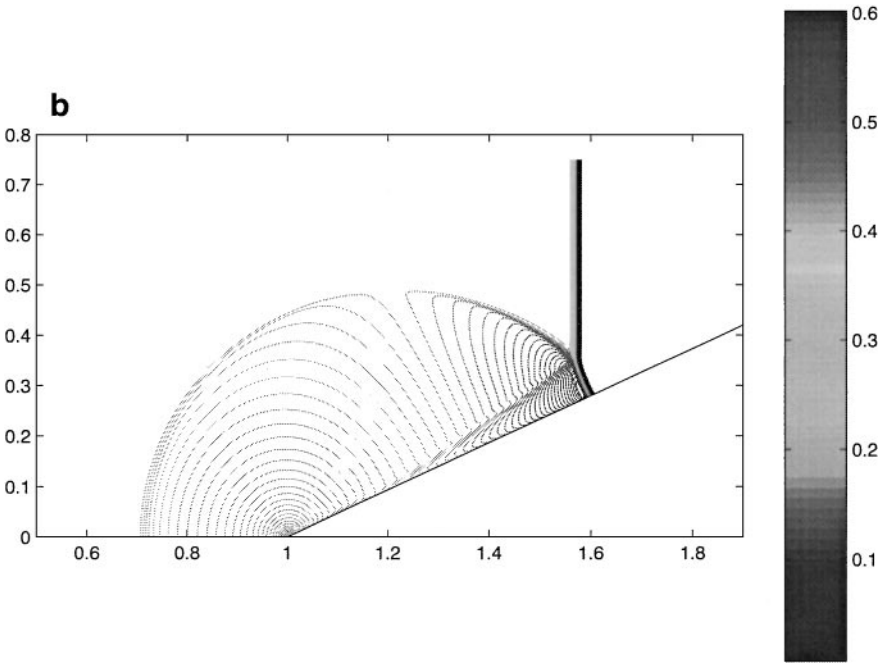


FIG. 13. Mach reflection of a shock wave. (a) Shadowgraph showing a vertical plane shock wave with $M = 1.3$ striking a 25 wedge, producing a reflected wave, a slip line, and a Mach stem normal to the wedge. (b) Computed Mach contours using unified code with h chosen to preserve grid angles, showing excellent agreement with observation (grid: 400×200).

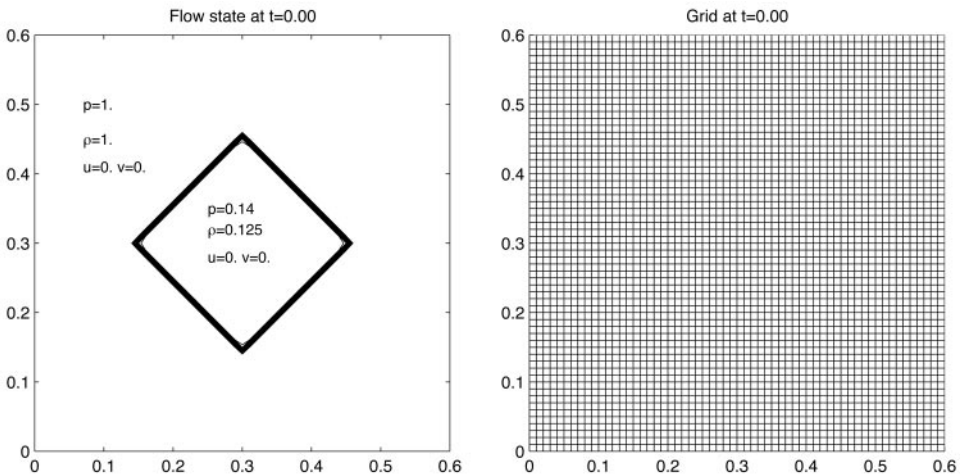


FIG. 14. An implosion/explosion problem showing the initial state and initial grid.

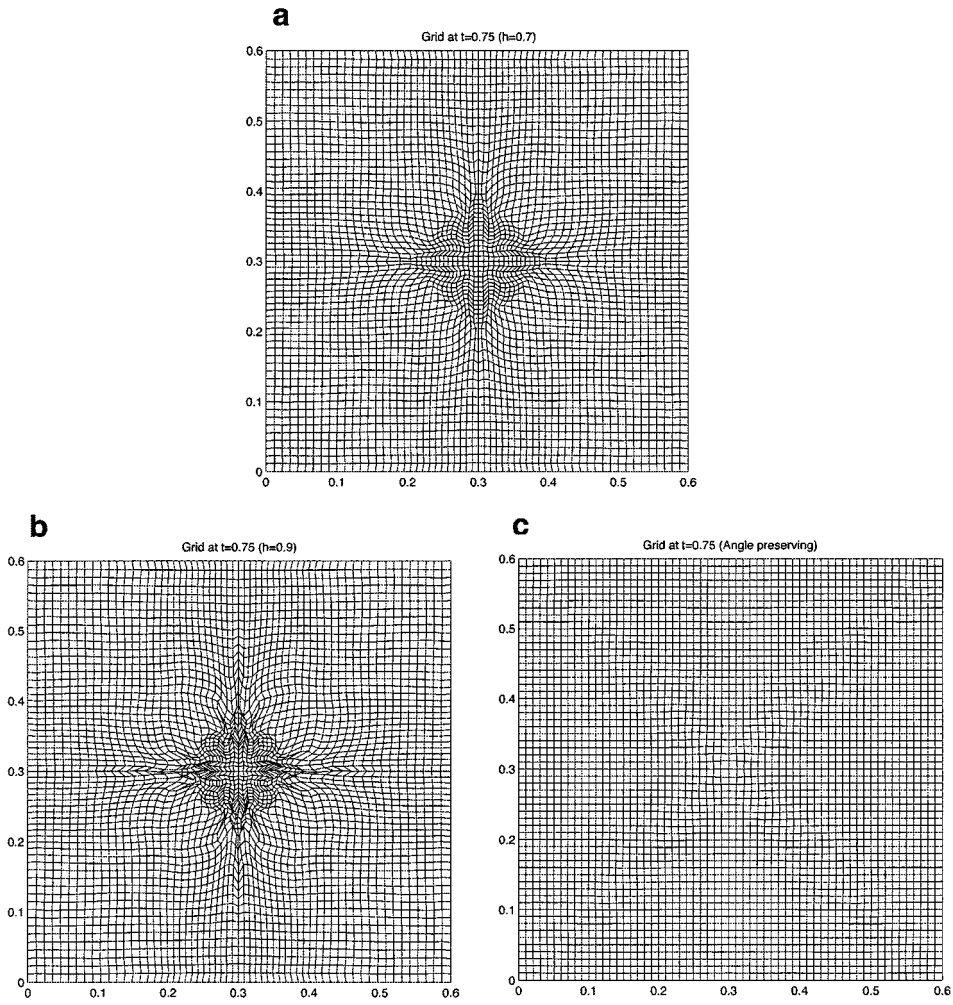
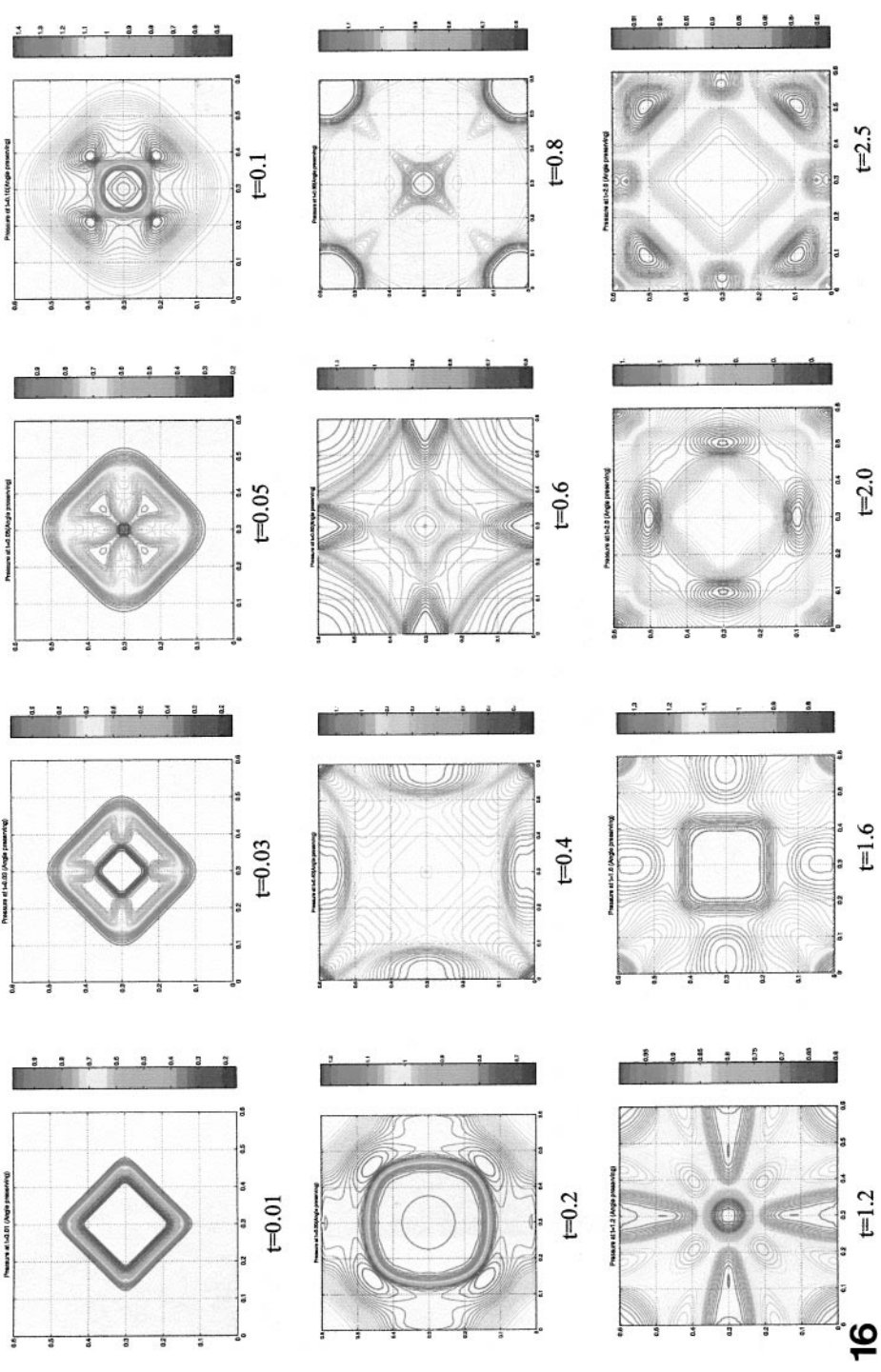
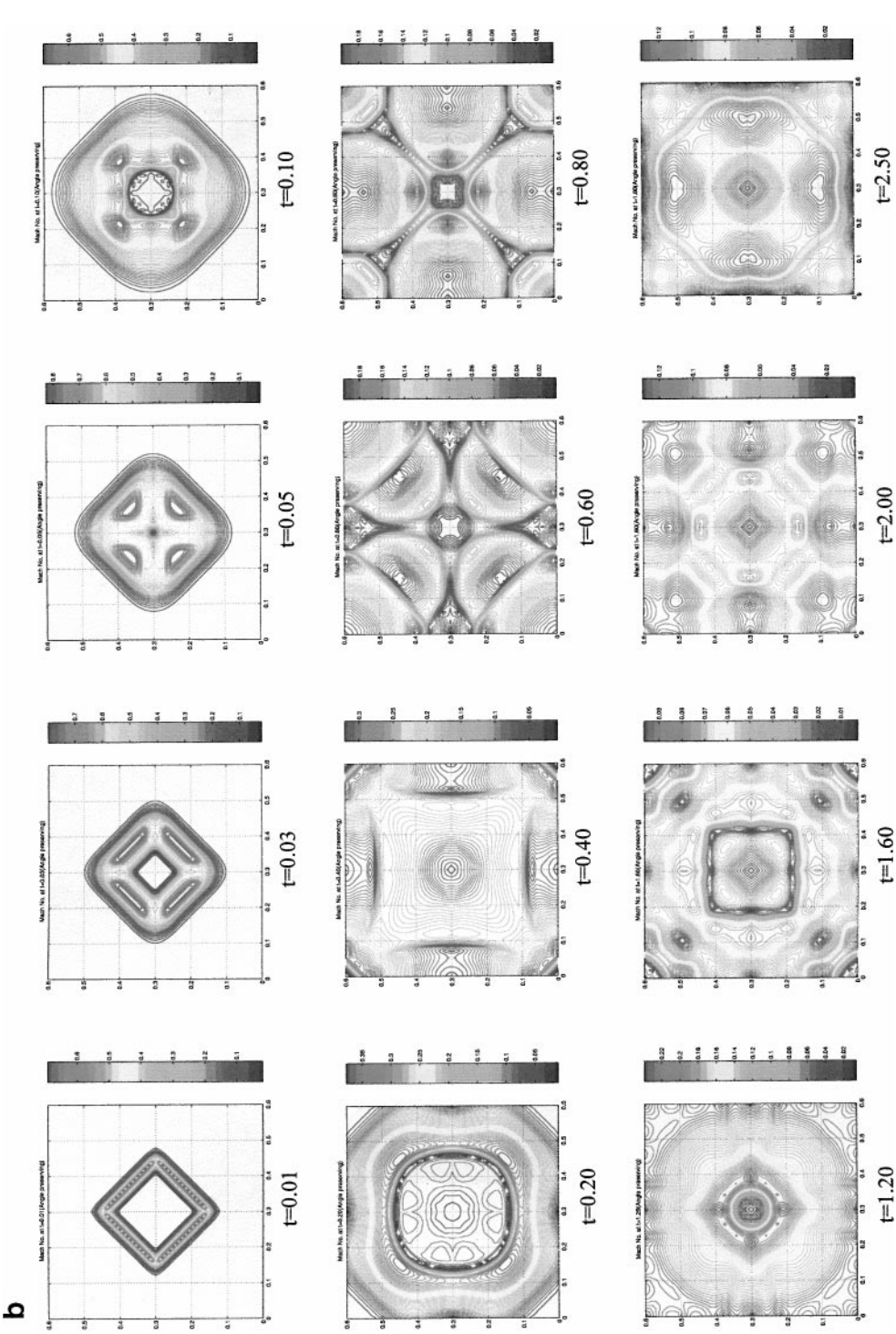


FIG. 15. Flow-generated grid at $t = 0.75$ in an implosion/explosion problem, (a) $h = 0.7$, (b) $h = 0.9$, (c) h chosen to preserve grid angles, Eq. (27).

the outside region, $(p, \rho, u, v) = (1., 1., 0., 0.)$. Initially, a uniform grid of 60×60 with $\Delta\xi = \Delta\eta = \Delta x = \Delta y = 0.01$ is given (Fig. 14). We test this example with $h = 0$, $h = 0.7$, $h = 0.9$, and the grid-angle preserving h, Eq. (27). Of course, the computer code can run non-stop when $h = 0$. But when $h = 0.9$, the code can run only until $t = 0.75$; soon afterwards it breaks down. We also note that in the Lagrangian case [28] which corresponds to $h = 1$, it breaks down at an earlier time, $t = 0.6$. This is because the computational cells move with the pseudo-particles and for large h, can become severely deformed. If we reduce h, say $h = 0.7$, the code can run longer until $t = 1.7$. This shows that smaller h can delay the severe cell deformation, but cannot remove it. With the grid-angle preserving method, which keeps

FIG. 16. (a) Evolution of pressure contours in an implosion/explosion problem. (b) Evolution of Mach contours in an implosion/explosion problem.


a
t=0.01
t=0.03
t=0.05
t=0.1
t=0.2
t=0.4
t=0.6
t=0.8
t=1.2
t=1.6
t=2.0
t=2.5


b
t=0.01
t=0.03
t=0.05
t=0.10
t=0.20
t=0.40
t=0.60
t=0.80
t=1.20
t=1.60
t=2.00
t=2.50

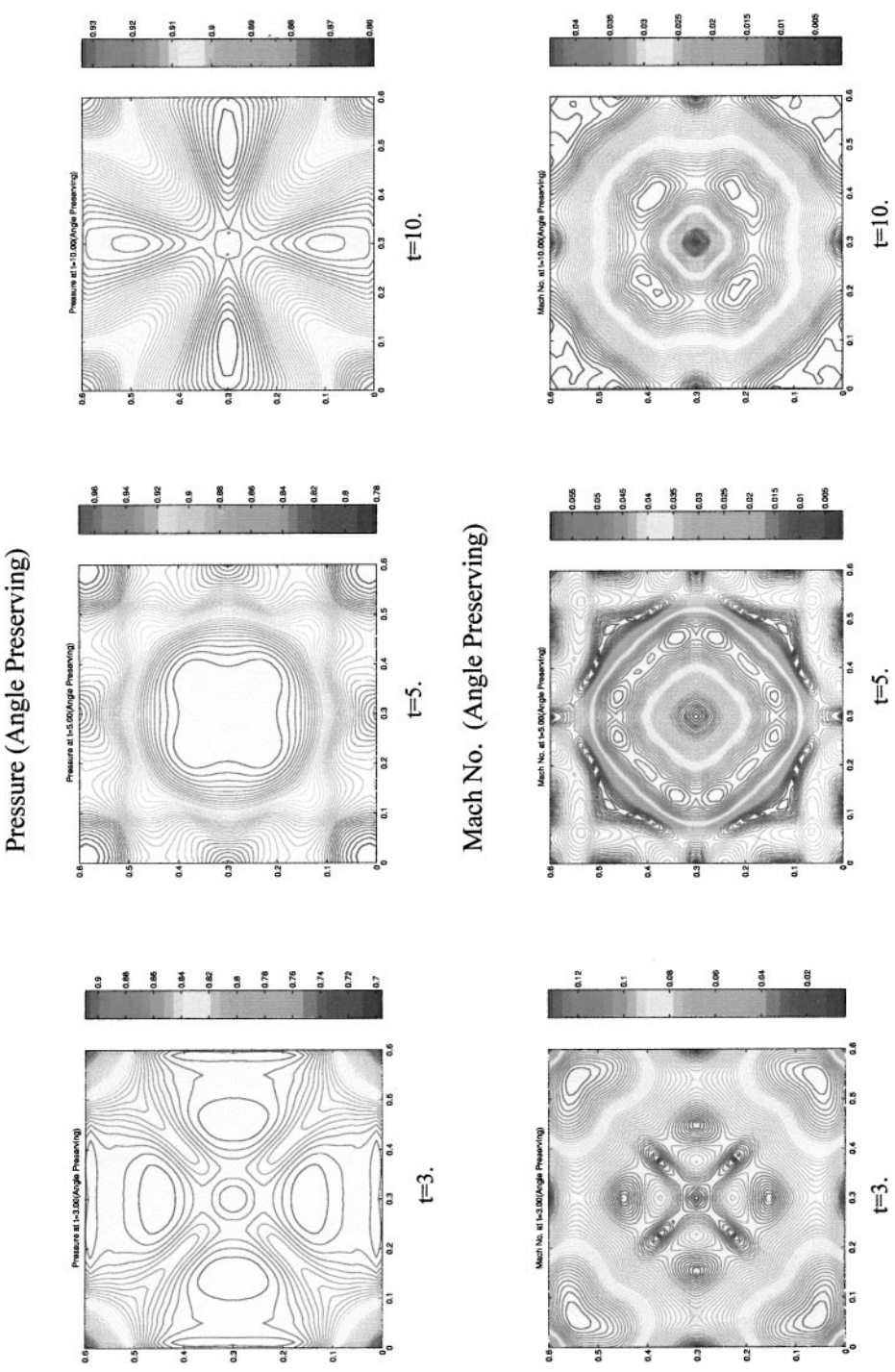


FIG. 17. Comparison of pressure and Mach contour in an implosion/explosion problem.

the grid regular, the code can run for a long time (we have computed to $t = 10$) without any indication of severe grid deformation. Figures 15a to 15c give the grids at $t = 0.75$ for different cases. We see that irregular grids prevail when h is constant and a regular grid prevails when h satisfies the grid-angle preserving property. Figures 16a and 16b give

computed pressure and Mach contours at different times up to $t = 2.5$. They display clearly the flow evolution process. Figure 17 compares pressure and Mach number contours at $t = 3$, 5, and 10. We see that as time increases pressure tends to a uniform distribution while Mach number diminishes, reflecting the process of conversion of kinetic energy to heat, as expected.

9. CONCLUSIONS

A unified coordinate system has been developed to describe fluid motion in which the flow variables are considered to be functions of time and of some permanent identification of *pseudo-particles* which move with velocity $h\mathbf{q}$, $\mathbf{q}$ being the velocity of fluid particles. It includes the Eulerian coordinates as a special case when $h = 0$ and the Lagrangian when $h = 1$.

Systematical comparisons show that with increasing h from $h = 0$ to $h = 1$, slip line resolution improves while grid deformation gets worse. It has been shown that for two-dimensional flow the choice of h to preserve grid angles results in a coordinate system which keeps the grid regular, thus avoiding the severe grid deformation in the Lagrangian coordinates, yet it retains sharp resolution of slip lines, especially for steady flow. It is, therefore, superior to both the Lagrangian and the Eulerian coordinates.

Extension to three-dimensional flow is being carried out. On the other hand, for one-dimensional flow the Lagrangian system of coordinates ($h = 1$) is shown [21] to be the best in slip line resolution. It can also be used to incorporate a shock-adaptive Godunov scheme to produce infinite shock resolution as well.

However, for two- and three-dimensional unsteady flow the system of Euler equations of gasdynamics written in Lagrangian coordinates is only weakly hyperbolic, lacking a complete set of eigenvectors, with all its possible negative consequences in numerical computation.

APPENDIX

Conventional Lagrangian Equations of Motion for an Inviscid Perfect Gas

In this Appendix we re-write the conventional Lagrangian equations of motion for an inviscid perfect gas in conservation form and show that they are just a special case of system (8) or (12) when $h = 1$. We consider the unsteady two-dimensional smooth flow of an inviscid perfect gas obeying the γ-law. The three-dimensional case can be treated similarly. The conventional Lagrangian coordinates (a, b) are the Cartesian coordinates (x, y) of fluid particles at initial time $t = 0$; i.e., $(a, b) = (x, y)|_{t=0}$. The continuity equation, the momentum equations, and the energy equation are, respectively,

$$\rho \frac{\partial(x, y)}{\partial(a, b)} = \rho_0(a, b) \tag{A1}$$

$$\frac{\partial^2 x}{\partial t^2} + \frac{1}{\rho}\frac{\partial p}{\partial x} = 0 \tag{A2}$$

$$\frac{\partial^2 y}{\partial t^2} + \frac{1}{\rho}\frac{\partial p}{\partial y} = 0 \tag{A3}$$

$$\frac{\partial \ln(p/\rho^{\gamma})}{\partial t} = 0, \tag{A4}$$

where p and ρ are pressure and density of the gas. Equation (A4) states that the entropy is constant following a fluid particle, which is true for smooth flow.

The pressure gradient terms in (A2) and (A3) in xy space can be eliminated to yield

$$\frac{\partial^2 x}{\partial t^2}\frac{\partial x}{\partial a} + \frac{\partial^2 y}{\partial t^2}\frac{\partial y}{\partial a} + \frac{1}{\rho}\frac{\partial p}{\partial a} = 0 \tag{A5}$$

$$\frac{\partial^2 x}{\partial t^2}\frac{\partial x}{\partial b} + \frac{\partial^2 y}{\partial t^2}\frac{\partial y}{\partial b} + \frac{1}{\rho}\frac{\partial p}{\partial b} = 0. \tag{A6}$$

Furthermore, in order to render this system of second-order non-linear partial differential equations a system of first-order quasillinear ones we introduce new dependent variables u, v, A, B, L, and M through the following equations:

$$\frac{\partial x}{\partial t} - u = 0 \tag{A7}$$

$$\frac{\partial y}{\partial t} - v = 0 \tag{A8}$$

$$\frac{\partial x}{\partial a} - A = 0 \tag{A9}$$

$$\frac{\partial y}{\partial a} - B = 0 \tag{A10}$$

$$\frac{\partial x}{\partial b} - L = 0 \tag{A11}$$

$$\frac{\partial y}{\partial b} - M = 0. \tag{A12}$$

Then after taking $\frac{\partial}{\partial t}$, (A1) becomes

$$\frac{\partial(\rho\Delta)}{\partial t} = 0, \tag{A13}$$

where

$$\Delta = AM - BL$$

and (A9)–(A12) become

$$\frac{\partial A}{\partial t} - \frac{\partial u}{\partial a} = 0 \tag{A14}$$

$$\frac{\partial B}{\partial t} - \frac{\partial v}{\partial a} = 0 \tag{A15}$$

$$\frac{\partial L}{\partial t} - \frac{\partial u}{\partial b} = 0 \tag{A16}$$

$$\frac{\partial M}{\partial t} - \frac{\partial v}{\partial b} = 0. \tag{A17}$$

Furthermore, (A5) and (A6) become

$$\rho\left(A\frac{\partial u}{\partial t}+B\frac{\partial v}{\partial t}\right)+\frac{\partial p}{\partial a}=0$$

$$\rho\left(L\frac{\partial u}{\partial t}+M\frac{\partial v}{\partial t}\right)+\frac{\partial p}{\partial b}=0$$

which may also be written as

$$\frac{\partial(\rho\Delta u)}{\partial t}+M\frac{\partial p}{\partial a}-B\frac{\partial p}{\partial b}=0$$

$$\frac{\partial(\rho\Delta v)}{\partial t}-L\frac{\partial p}{\partial a}+A\frac{\partial p}{\partial b}=0.$$

By use of (A9)–(A12), the last two equations are equivalent to

$$\frac{\partial(\rho\Delta u)}{\partial t}+\frac{\partial(pM)}{\partial a}-\frac{\partial(pB)}{\partial b}=0 \tag{A18}$$

$$\frac{\partial(\rho\Delta v)}{\partial t}-\frac{\partial(pL)}{\partial a}+\frac{\partial(pA)}{\partial b}=0. \tag{A19}$$

Evidently, Eqs. (A7) and (A8) are decoupled from (A13) to (A19) and (A4). The eight equations (A9) to (A15) and (A4) form a closed system and can be shown (see below) to be just a special case of Eq. (12) with $h=1$. After this system is solved for $(\rho, p, u, v, A, B, L, M)$, the functions $x(a,b,t)$ and $y(a,b,t)$ can be found from (A7) and (A8). This process is the same as the transformation (18) together with

$$\left(\frac{\partial L}{\partial a}-\frac{\partial A}{\partial b}\right)_{t=0}=0 \tag{A20}$$

$$\left(\frac{\partial M}{\partial a}-\frac{\partial B}{\partial b}\right)_{t=0}=0 \tag{A21}$$

which are easily ensured computationally.

To show that the system of equations (A13) to (A19) and (A4) is equivalent to (12) with $h=1$, we first identify λ with t, ξ with a, and η with b. Then (A13) is the first equation of (12), and (A14) to (A17) are the same as the last four equations of (12). Furthermore, (A18) and (A19) are the same as the second and third equations of (12). We now show that the fourth equation of (12), i.e., the energy equation

$$\frac{\partial(\rho\Delta e)}{\partial t}+\frac{\partial}{\partial a}[p(uM-vL)]+\frac{\partial}{\partial b}[p(Av-Bu)]=0, \tag{A22}$$

is equivalent to (A4) as follows. Thus (A22) $-u\times$ (A18) $-v\times$ (A19) yields

$$\begin{aligned}0&=\frac{\rho\Delta}{\gamma-1}\frac{\partial}{\partial t}\left(\frac{p}{\rho}\right)+pM\frac{\partial u}{\partial a}-pL\frac{\partial v}{\partial a}-pB\frac{\partial u}{\partial b}+pA\frac{\partial v}{\partial b}\\&=\frac{\rho\Delta}{\gamma-1}\frac{\partial}{\partial t}\left(\frac{p}{\rho}\right)+p\frac{\partial\Delta}{\partial t},\end{aligned} \tag{A23}$$

after making use of (A14) to (A17). Using (A13), Eq. (A23) becomes

$$\frac{\partial}{\partial t}\ln\frac{p}{\rho^\gamma}=0 \tag{A24}$$

which is the same as (A4).

In conclusion, (a) the system of conventional Lagrangian equations of motion [(A1)–(A4)], i.e., the system of Euler equations in conventional Lagrangian coordinates, is rewritten in conservation form [(A13)–(A17), (A18), (A19), (A24)]; (b) it is just a special case of Eq. (12) with $h=1$; and (c) it is thus weakly hyperbolic.

The special choice of $(a, b)=(x, y)_{t=0}$ corresponds to choosing $A=M=1$ and $B=L=0$. Such a choice of coordinates, however, offers no advantage in computation. For instance, if the fluid initially occupies a domain that is complicated geometrically, the domain in the computational ab plane is the same complicated domain. But with a suitable choice of the Lagrangian coordinates (ξ, η), one can simplify the computational domain in the $\xi\eta$ plane to a rectangular domain.

ACKNOWLEDGMENTS

This research was funded by the Research Grants Council of Hong Kong to the senior author. We thank Dr. C. Y. Loh and Mr. S. Koudriakov for many valuable discussions.

REFERENCES

1. D. H. Wagner, Equivalence of Euler and Lagrangian equations of gas dynamics for weak solutions, *J. Differential Equations* **68**, 118 (1987).
2. A. Harten, The artificial compression method for computation of shocks and contact discontinuities: III. Self-adjusting hybrid schemes, *Math. Comp.* **32**, 363 (1978).
3. A. Harten, ENO schemes with subcell resolution, *J. Comput. Phys.* **83**, 148 (1989).
4. C. W. Shu and S. Osher, Efficient implementation of essentially non-oscillatory shock-capturing schemes II, *J. Comput. Phys.* **83**, 32 (1989).
5. M. S. Liou, An extended Lagrangian method, *J. Comput. Phys.* **118**, 294 (1995).
6. C. W. Hirt, A. A. Amsden, and J. L. Cook, An arbitrary Lagrangian–Eulerian computing method for all flow speeds, *J. Comput. Phys.* **14**, 227 (1974).
7. W. Pracht, Calculating three-dimensional fluid flows at all speeds with an Eulerian–Lagrangian computing mesh, *J. Comput. Phys.* **17**, 132 (1975).
8. L. G. Margolin, Introduction to "an arbitrary Lagrangian–Eulerian computing method for all flow speeds," *J. Comput. Phys.* **135**, 198 (1997).
9. M. S. Hall, A comparison of first and second order rezoned and Lagrangian Godunov solutions, *J. Comput. Phys.* **90**, 458 (1990).
10. C. Y. Loh and W. H. Hui, A new Lagrangian method for steady supersonic flow computation. Part I. Godunov scheme, *J. Comput. Phys.* **89**, 207 (1990).
11. W. H. Hui and C. Y. Loh, A new Lagrangian method for steady supersonic flow computation. Part II. Slipline resolution, *J. Comput. Phys.* **103**, 450 (1992).
12. W. H. Hui and C. Y. Loh, A new Lagrangian method for steady supersonic flow computation. Part III. Strong shocks, *J. Comput. Phys.* **103**, 465 (1992).
13. W. H. Hui and Y. C. Zhao, A generalized Lagrangian method for solving the Euler equations, in *Nonlinear Hyperbolic Problems: Theoretical, Applied and Computational Aspects*, edited by A. Donato and F. Oliveri (Vieweg, Braunschwig, Germany, 1992), p. 336.

14. C. Y. Lepage and W. H. Hui, A shock-adaptive godunov scheme based on the generalized Lagrangian formulation, *J. Comput. Phys.* **122**, 291 (1995).
15. W. H. Hui and D. L. Chu, Optimal grid for the steady Euler equations, *Comput. Fluid Dyn. J.* **4**, 403 (1996).
16. C. Y. Loh and M. S. Liou, A new Lagrangian method for three-dimensional steady supersonic flows, *J. Comput. Phys.* **113**, 224 (1994).
17. W. H. Hui and Y. He, Hyperbolicity and optimal coordinates for the three-dimensional supersonic Euler equations, *SIAM J. Appl. Math.* **57**, 893 (1997).
18. W. H. Hui, Generalized Lagrangian formulation of computational fluid dynamics in *Computational Fluid Dynamics Review—1995*, edited by M. Hafez and K. Oshima (Wiley, New york, 1995), p. 382.
19. E. Godlewski and P.-A. Raviart, *Numerical Approximation of Hyperbolic System of Conservation Laws* (Springer, New York, 1996).
20. H. O. Kreiss and J. Lorenz, *Initial-Boundary Value Problems and the Navier–Stokes Equations* (Academic Press, San Diego, 1989).
21. W. H. Hui and S. Koudriakov, The role of coordinates in the computation of discontinuities in one-dimensional Flow, submitted for publication.
22. R. P. Fedkiew, B. Merriman, and S. Osher, Efficient characteristic projection upwind difference schemes for hyperbolic systems, *J. Comput. Phys.* **141**, 22 (1998).
23. S. K. Godunov, Difference method of nuemrical computations of discontinuous solutions in hydrodynamic equations, *Math. Sbornik* **47**, 271 (1959).
24. B. Van Leer, Towards the ultimate conservative difference scheme, V. A second order sequel to Godunov's method, *J. Comput. Phys.* **32**, 234 (1979).
25. M. G. Crandall and A. Majda, Monotone difference approximations for scalar conservation laws, *Math. Comp.* **34**, 1 (1980).
26. G. Strang, On the construction and comparison of difference schemes, *SIAM J. Numer. Anal.* **5**, 506 (1968).
27. M. Van Dyke, *An Album of Fluid Motion* (The Parabolic Press, Stanford, CA, 1982), p. 143.
28. C. Y. Loh and W. H. Hui, *A new Lagrangian method for time-dependent inviscid flow computation, SIAM J. Sci. Comput.*, in press (1999).