INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
*Int. J. Numer. Meth. Fluids* **29**: 97–119 (1999)

# AN INVESTIGATION OF RANDOM CHOICE METHOD FOR THREE-DIMENSIONAL STEADY SUPERSONIC FLOWS

CHING-YUEN LOH[a,*], MENG-SING LIOU[a] AND WAI HOW HUI[b]

[a] *Mail Stop 5-11, Internal Fluid Mechanics Division, NASA Lewis Research Center, Cleveland, OH 44135, USA*
[b] *Department of Mathematics, Hong Kong University of Science and Technology, Clear Water Bay, Hong Kong*

## SUMMARY

In this paper, an unsplit random choice method (RCM) is developed and applied to numerically solve three-dimensional supersonic steady flow problems. In order to keep the contacts (slip surfaces) crisply resolved, a new Lagrangian formulation is employed. Due to the lack of exact solutions to 3D Riemann problems, approximate Riemann solutions in the weak sense are adopted. The RCM is thus as efficient as the deterministic TVD schemes, and yields almost identical results in the model problems. Copyright © 1999 John Wiley & Sons, Ltd.

KEY WORDS: random choice; Lagrangian; 3D supersonic flow

## 1. INTRODUCTION

The numerical simulation of gas dynamic flows has made enormous progress in terms of solution techniques and ever-improving accuracy in the past few decades. Various finite volume-based central difference schemes with artificial damping, Runge–Kutta time marchings, are developed. The 1980s has witnessed an exhaustive exploration and booming of upwind, monotonic schemes, approximate Riemann solvers and upwind flux splittings [1–5].

In addition to the deterministic approaches at the other end of the spectrum of numerical methods, Glimm [6] initiated the random choice method (RCM) in 1965 for constructive proof of the existence of solutions to non-linear hyperbolic systems of conservation laws. In the 1970s, Chorin [7] developed the RCM as a practical computational method for solutions to 1D Euler equations of gas dynamics. Chorin's work was followed by numerous researchers, with applications and further improvements such as Colella [8], Sod [9] and Concus and Proskurowski [10], etc. In essence, the solution is advanced in time by a sequence of operations, which includes the exact or approximate solution of Riemann problems and a random space sampling procedure. Recently, the idea of random choice by Glimm has been extended and used in a broad way, e.g. in the vortex method [11] and the front tracking method [12].

For one-dimensional flow, the RCM yields absolutely sharp shock and contact discontinuity resolution, which is indeed a repetition in the large of the local exact Riemann solutions. For

* Correspondence to: Mail Stop 5-11, Internal Fluid Mechanics Division, NASA Lewis Research Center, Cleveland, OH 44135, USA.

CCC 0271–2091/99/010097–23$17.50

*Received August 1997*
*Revised January 1998*

2D steady supersonic flow, Marshall and Plohr [13] treated one of the space variables as a time-like variable and applied the RCM in a way similar to that for the 1D unsteady flows. Based on their earlier new Lagrangian formulation, Loh and Hui [14] have successfully applied the RCM to 2D steady supersonic ideal gas flow computations. Recently [15], they have considered the real gas version of the Lagrangian RCM of Reference [14]. In either Eulerian or the new Lagrangian formulations, the basic building block—the Riemann solution for 2D steady supersonic flow—can be obtained analytically, leading to the success of the RCM for 2D steady supersonic flow computation.

For unsteady flows in genuine two (or higher) space dimensions, however, the very desirable property of RCM—no diffusive errors from spatial averaging—do not seem to persist [8]. Oliver and Grönig [16] reported an application of the RCM to a 2D shock focusing and diffraction problem when the Mach number was low. They used statistical smoothing to handle the spurious wiggling noise generated by the RCM. Colella [8] suggested using a Strang-type operator-splitting technique in the RCM to handle unsteady 2D flow computations. Since the Strang splitting requires the functions to be continuously differentiable, the technique fails across a discontinuity, and unpleasant wiggles are generated. He then suggests using a hybrid Godunov–RCM method, in which the Godunov scheme was employed for strong shocks to replace the RCM; better results were reported.

In the present paper, as a step towards a multi-dimensional RCM, the possibility of using the RCM for 3D steady supersonic flow computation is investigated. The key issue is whether an exact 3D Riemann solution can be obtained in a convenient way. The general multi-dimensional Riemann problem itself is still a current research topic [17,18]. Based on the analysis of Zhang and Zheng [19], and the numerical results of Schultz-rinne *et al.* [20] for 2D unsteady gas flow, which may be regarded as an analog of the 3D steady flow, the answer is not encouraging. First, due to the high non-linearity and complexity of the Riemann problem, very little is analytically known about its solution. Secondly, even if numerical results could be obtained, they are CPU time-consuming and inappropriate for a basic building block of the RCM. The only choice left seems to be adopting some forms of approximate Riemann solutions. In this connection, Harten and Lax [21] were the first to use a Godunov-type approximate Riemann solver in a 1D RCM. In this paper, it will be shown how to construct Godunov-type approximate Riemann solutions in the multi-dimensional sampling region and use them for the RCM.

In this investigation, a new Lagrangian formulation [22] for the 3D steady supersonic flow is chosen. The new formulation possesses the following attractive advantages:

1. Stream surfaces are used as a co-ordinate surface, so flow physics is closely followed. It is simpler to choose the sampling states than in an Eulerian formulation [14,15].
2. Contact discontinuity (slip surface) is crisply resolved [22,31] and the RCM can concentrate on achieving sharper shock resolution.

The contents are organized as follows. In Section 2, the new Lagrangian formulation for 3D steady supersonic flow is briefly introduced. A Godunov-type scheme is sketched for later use as a weak approximate Riemann solver. In Section 3, the 3D RCM with a Godunov-type approximate Riemann solver is illustrated and discussed. Numerical results are presented in Section 4 and compared with those by a deterministic approach. Finally, concluding remarks are drawn in Section 5.

 

## 2. THE NEW LAGRANGIAN FORMULATION AND THE GODUNOV-TYPE CONSERVATIVE AVERAGE

For an RCM with higher space dimensions, a hybrid Godunov–RCM operator splitting method was suggested by Colella [8]. As observed from an example in Section 4 (Example 5), however, the hybrid operator-splitting method does not seem to yield satisfactory results for the 3D supersonic flow computation. In this example, numerical noise is high and the computation is terminated at half-way due to non-linear instability. As such, hereafter in this paper, concentration will be on an unsplit RCM, using the approximate Godunov-type Riemann solver to obtain subcell conservative averages for the sampling, since an exact Riemann solver for higher space dimensions is not available.

This section serves to provide some necessary knowledge of the new 3D Lagrangian formulation and the corresponding Godunov-type scheme for use in the RCM. More details can be found in [22].

The derivation of the new Lagrangian conservation form starts from the Euler equations in conservation form the 3D steady inviscid flows:

$$\frac{\partial \mathbf{E}_1}{\partial x}+\frac{\partial \mathbf{F}_1}{\partial y}+\frac{\partial \mathbf{G}_1}{\partial z}=0, \tag{1}$$

where

$$\mathbf{E}_1=\begin{pmatrix}\rho u\\ \rho u^2+p\\ \rho uv\\ \rho uw\\ \rho uH\end{pmatrix}, \qquad \mathbf{F}_1=\begin{pmatrix}\rho v\\ \rho vu\\ \rho v^2+p\\ \rho uw\\ \rho vH\end{pmatrix}, \qquad \mathbf{G}_1=\begin{pmatrix}\rho w\\ \rho wu\\ \rho wv\\ \rho w^2+p\\ \rho wH\end{pmatrix}. \tag{2}$$

As usual, $u$, $v$, $w$, $\rho$ and $p$ are respectively, the Cartesian components of flow velocity, density and pressure of the fluid obeying the $\gamma$-law; the total enthalpy $H$ is

$$H=\frac{1}{2}(u^2+v^2+w^2)+\frac{\gamma}{(\gamma-1)}\frac{p}{\rho}, \quad \gamma=1.4. \tag{3}$$

### *2.1. The new Lagrangian conservation form for 3D steady flow*

The new 3D Lagrangian conservation form is based on the concept of Lagrangian time $\tau$ [23,24], or Lagrangian distance $\lambda$ [25,26]. The Lagrangian distance being defined as the distance traveled by a particle along its streamline:

$$\lambda=\int_0^{\tau} q\,\mathrm{d}\tau, \quad \text{with} \quad q=(u^2+v^2+w^2)^{1/2}. \tag{4}$$

and is chosen as the marching variable in this paper.

In a 3D steady flow, there exists two independent streamfunctions, say, $\xi$ and $\eta$. Each fixed $\xi$ or $\eta$ represents a stream surface. A fixed pair of $\xi$ and $\eta$ denotes a streamline in 3D space. Following the streamline, the Lagrangian distance $\lambda$ uniquely determines the location of the fluid particle. In other words, $(\lambda, \xi, \eta)$ may be considered as a new set of independent variables, which are now functions of the Cartesian co-ordinates $\mathbf{r}=(x, y, z)^T$. T'he fluid velocity $\mathbf{V}$ is:

 

$$\mathbf{V} = (u, v, w)^T = \frac{\partial \mathbf{r}}{\partial \tau} = q \frac{\partial \mathbf{r}}{\partial \lambda}. \tag{5}$$

Furthermore, the flow variable vector is defined as:

$$\mathbf{Q} = (u, v, w, p, \rho)^T \tag{6}$$

and the following Lagrangian geometrical quantities:

$$\mathbf{T} = (U, V, W)^T = \frac{\partial \mathbf{r}}{\partial \xi}, \tag{7}$$

$$\mathbf{S} = (X, Y, Z)^T = \frac{\partial \mathbf{r}}{\partial \eta}, \tag{8}$$

which represent the lateral rate of displacement of a fluid cell in the physical domain. Under the transformation from $(x, y, z)$ to $(\lambda, \xi, \eta)$, the Euler equations (Equations (1) and (2)) become (for details, see [22]):

$$\frac{\partial \mathbf{E}}{\partial \lambda} + \frac{\partial \mathbf{F}}{\partial \xi} + \frac{\partial \mathbf{G}}{\partial \eta} = 0, \tag{9}$$

where

$$\mathbf{E} = \begin{pmatrix} e_1 \\ e_2 \\ e_3 \\ e_4 \\ e_5 \\ e_6 \\ e_7 \\ e_8 \\ e_9 \\ e_{10} \\ e_{11} \end{pmatrix} \equiv \begin{pmatrix} K \\ H \\ Ku + pJ_{11} \\ Kv + pJ_{12} \\ Kw + pJ_{13} \\ U \\ V \\ W \\ X \\ Y \\ Z \end{pmatrix}, \quad \mathbf{F} = \begin{pmatrix} 0 \\ 0 \\ pJ_{21} \\ pJ_{22} \\ pJ_{23} \\ -u/q \\ -v/q \\ -w/q \\ 0 \\ 0 \\ 0 \end{pmatrix}, \quad \mathbf{G} = \begin{pmatrix} 0 \\ 0 \\ pJ_{31} \\ pJ_{32} \\ pJ_{33} \\ 0 \\ 0 \\ 0 \\ -u/q \\ -v/q \\ -w/q \end{pmatrix} \tag{10}$$

with the mass flux

$$K = \rho q \left| \frac{\partial(x, y, z)}{\partial(\lambda, \xi, \eta)} \right| = \rho \begin{vmatrix} u & v & w \\ U & V & W \\ X & Y & Z \end{vmatrix} \tag{11}$$

and co-factors

$$\begin{aligned} J_{11} &= \begin{vmatrix} V & W \\ Y & Z \end{vmatrix}, \qquad J_{12} = -\begin{vmatrix} U & W \\ X & Z \end{vmatrix}, \qquad J_{13} = \begin{vmatrix} U & V \\ X & Y \end{vmatrix}, \qquad J_{21} = -\frac{1}{q}\begin{vmatrix} v & w \\ Y & Z \end{vmatrix}, \\ J_{22} &= \frac{1}{q}\begin{vmatrix} u & w \\ X & Z \end{vmatrix}, \qquad J_{23} = -\frac{1}{q}\begin{vmatrix} u & v \\ X & Y \end{vmatrix}, \qquad J_{31} = \frac{1}{q}\begin{vmatrix} v & w \\ V & W \end{vmatrix}, \qquad J_{32} = -\frac{1}{q}\begin{vmatrix} u & w \\ U & W \end{vmatrix}, \\ J_{33} &= \frac{1}{q}\begin{vmatrix} u & v \\ U & V \end{vmatrix}. \end{aligned} \tag{12}$$

 

In the following context, Equation (9) is applied to all the subcell regions to construct the piecewisely constant local Godunov-type conservative averages. When all these subcell conservative averages in a computational cell are pieced together, they form a *weak Riemann solution* in the cell. Then, the RCM can be applied to randomly choose one of these conservative averages as the representative of the cell.

### *2.2. The Godunov-type cell and subcell conservative averages*

This subsection begins with the evaluation of the Godunov-type cell conservative average, which is equivalent to the space marching solution by a Godunov-type scheme [22]. Then, the evaluation of the Godunov-type *subcell* conservative average follows in the same way.

*2.2.1. The Godunov-type cell conservative averages*. The system (20) is hyperbolic-like if the flow Mach number in the marching direction, i.e. $\lambda$, is supersonic [32]. In the deterministic approach of numerical computation, a Godunov-type scheme, such as van Leer's MUSCL scheme, may be used for numerically solving the hyperbolic system. In the RCM approach, if the 3D Riemann solution at the cell interface, and hence the subcell flow structure, were known, it would also be possible to solve the problem in a straightforward way, just as in the 2D cases [14,15], and Glimm's theorem [6] will guarantee its convergence. Unfortunately, in reality little is known about the Riemann solution of higher space dimensions and the subsequent subcell flow structure. On the other hand, by dimension-splitting, approximate cell interface flux **F** and **G**, are available via local Riemann solvers of lower dimensions [22]. Could this Godunov-type average be used to represent approximately the flow in a subcell region and to piece these averages together to represent the flow structure in a cell? If so, the RCM can be easily applied by choosing one of the average constants as a representative of the cell. For this purpose, how to obtain a Godunov-type conservative average over a cell or a subcell region must be discussed. More details about the Godunov-type scheme can be found in [22]. It should be emphasized that the goal in this subsection is to apply the technique to piecewisely rebuild the subcell flow structure in a cell.

For the deterministic or random approach, the physical domain and computational domain in the $\lambda$–$\xi$–$\eta$ space are divided into cells respectively, as illustrated in Figure 1(a) and (b). A cuboid mesh in the computational domain is used and the computation marches in the Lagrangian distance $\lambda$. The superscript $k$ refers to the marching step number and the subscripts $i$ and $j$ refer to the cell number on the distance plane (the plane with $\lambda =$ constant). The marching step $\Delta\lambda^k = \lambda^{k+1} - \lambda^k$ is uniform for all $i$ and $j$. The mesh divides the computational domain into cuboid control volumes or cells, which, in the $\xi$- and $\eta$-directions, are centered at $(\lambda^k, \xi_i, \eta_j)$ and have heights $\Delta\xi_i = \xi_{i+1/2} - \xi_{i-1/2}$ and $\Delta\eta_j = \eta_{j+1/2} - \eta_{j-1/2}$ (for all $k$).

In the physical space, a cuboid cell marching in the $(\lambda, \xi, \eta)$ space corresponds to a fluid marching along its stream tube, with step $\Delta\lambda$. The $\xi$–$\eta$ plane in computational space corresponds to the initial surface in the physical space. Any curvilinear co-ordinate mesh on the initial surface may be used as the $\xi$–$\eta$ co-ordinate mesh, and the initial **T** and **S** can be determined as part of the initial condition [22]. A solid wall is always a stream surface, hence a co-ordinate surface.

In the Godunov-type preprocessing MUSCL scheme, at the cell interface flow variables $u$, $v$, $w$, $p$ and $\rho$ are upgraded by non-linear interpolation before they are applied as the input for the local Riemann solvers. The upgrading is performed in a dimension-by-dimension way. For example, in the $\xi$-direction, let $f$ be any of the above physical variables, then, instead of assuming a uniform state in the cells $(i, j)$ and $(i+1, j)$, assume linearly distributed states and use the linear relation to determine the cell interface flow variables, then use them as the input to the local Riemann solver:

 

$$f_t = f_{i+1,j} - 0.5(f_{i+2,j} - f_{i+1,j})\phi(r^+), \quad r^+ = \frac{f_{i+1,j} - f_{i,j}}{f_{i+2,j} - f_{i+1,j}};$$

$$f_b = f_{i,j} + 0.5(f_{i,j} - f_{i-1,j})\phi(r^-), \quad r^- = \frac{f_{i+1,j} - f_{i,j}}{f_{i,j} - f_{i-1,j}};$$

where

$$\phi(r) = \max(0, \min(1, r))$$

is the minmod flux limiter, and subscripts t and b correspond respectively, to the top and bottom states of the cell interface. These new $f_t$ and $f_b$ are then fed to the local Riemann solvers.

The updating of Godunov-type cell averages for Equation (9) is derived by applying the divergence theorem to the cuboid cell $(i, j, k)$. Assuming $\mathbf{E}^k_{i,j}$ is known, the result is

$$\mathbf{E}^{k+1}_{i,j} = \mathbf{E}^k_{i,j} - \frac{\Delta\lambda^k}{\Delta\xi_i}(\mathbf{F}^{k+1/2}_{i+1/2,j} - \mathbf{F}^{k+1/2}_{i-1/2,j}) - \frac{\Delta\lambda^k}{\Delta\eta_j}(\mathbf{G}^{k+1/2}_{i,j+1/2} - \mathbf{G}^{k+1/2}_{i,j-1/2},$$

$$i = 1, 2, \ldots, m; \quad j = 1, 2, \ldots, n, \tag{13}$$

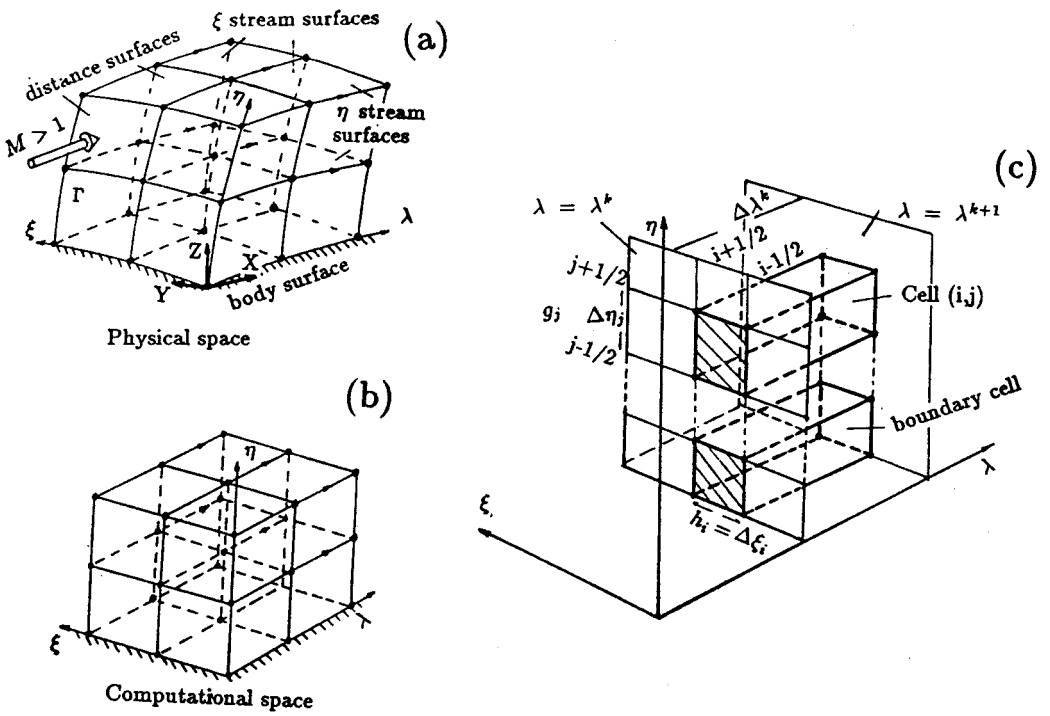


Figure 1. 3D computational space and mesh.

 

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

$$f_{i,j}^{k}=\frac{1}{\Delta\xi_i\,\Delta\eta_j}\int_{\xi_{i-1/2}}^{\xi_{i+1/2}}\int_{\eta_{j-1/2}}^{\eta_{j+1/2}} f(\lambda^k,\xi,\eta)\,\mathrm{d}\xi\,\mathrm{d}\eta \tag{14}$$

and the notations for $\lambda$ averages of $f$ are:

$$f_{i+1/2}^{k+1/2}=\frac{1}{\Delta\lambda^k}\int_{\lambda^k}^{\lambda^{k+1}} f(\lambda,\xi_{i+1/2},\eta_j)\,\mathrm{d}\lambda, \tag{15}$$

$$f_{i,j+1/2}^{k+1/2}=\frac{1}{\Delta\lambda^k}\int_{\lambda^k}^{\lambda^{k+1}} f(\lambda,\xi_i,\eta_{j+1/2})\,\mathrm{d}\lambda. \tag{16}$$

In Equation (13), 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 $(i,j)$ are obtained by applying the dimension-splitting technique [27,28] to the Godunov scheme for (9). One needs only to solve 2D Riemann problems formed by two adjacent interface states, say, $\mathbf{Q}_{i,j}$ and $\mathbf{Q}_{i+1,j}$ instead of genuine 3D Riemann problems. The interaction between two 3D supersonic flows of constant states separated by a plane is much more complicated than in the 2D case, the process is detailed in [22].

Now the numerical procedure for computing conservation *cell* averages is summarized:

(i) With all $\mathbf{E}_{i,j}^{k}$ and $\mathbf{Q}_{i,j}^{k}$ known at the previous step $k$ ($k=0,1,2,\ldots$), solve the local Riemann problems (or local boundary Riemann problem) at the cell interfaces and obtain the cell interface flow variables: $\mathbf{V}_{i\pm1/2,j}^{k+1/2}$ or $\mathbf{V}_{i,j\pm1/2}^{k+1/2}$ and $p_{i\pm1/2,j}^{k+1/2}$ or $p_{i,j\pm1/2}^{k+1/2}$ as described in the above context. The fluxes $\mathbf{F}$ and $\mathbf{G}$ are calculated from the $\mathbf{F}$ and $\mathbf{G}$ expressions in Equation (10). In order to do so, $\mathbf{T}_{i,j}^{k}$ and $\mathbf{S}_{i,j}^{k}$ must first be updated to $\mathbf{T}_{i,j}^{k+1}$ and $\mathbf{S}_{i,j}^{k+1}$:

$$\begin{pmatrix}U_{i,j}^{k+1}\\ V_{i,j}^{k+1}\\ W_{i,j}^{k+1}\end{pmatrix}=\mathbf{T}_{i,j}^{k+1}=\mathbf{T}_{i,j}^{k}-\frac{\Delta\lambda^k}{\Delta\xi_i}\left(\frac{\mathbf{V}_{i+1/2,j}^{k+1/2}}{q_{i+1/2,j}^{k+1/2}}-\frac{\mathbf{V}_{i-1/2,j}^{k+1/2}}{q_{i-1/2,j}^{k+1/2}}\right), \tag{17}$$

$$\begin{pmatrix}X_{i,j}^{k+1}\\ Y_{i,j}^{k+1}\\ Z_{i,j}^{k+1}\end{pmatrix}=\mathbf{S}_{i,j}^{k+1}=\mathbf{S}_{i,j}^{k}-\frac{\Delta\lambda^k}{\Delta\eta_j}\left(\frac{\mathbf{V}_{i,j+1/2}^{k+1/2}}{q_{i,j+1/2}^{k+1/2}}-\frac{\mathbf{V}_{i,j-1/2}^{k+1/2}}{q_{i,j-1/2}^{k+1/2}}\right), \tag{18}$$

$$i=1,2,\ldots,m;\quad j=1,2,\ldots,n;\quad k=0,1,2,\ldots$$

With $(U,V,W)^T$ and $(X,Y,Z)^T$ from Equations (17) and (18), and the computed cell interface $\mathbf{V}_{i\pm1/2,j}^{k+1/2}$ or $\mathbf{V}_{i,j\pm1/2}^{k+1/2}$ from the local Riemann solvers, the cofactors $J_{rs}$ ($r=2,3$; $s=1,2,3$) at the cell interfaces can be calculated. Furthermore, by utilizing the cell interface pressure $p_{i\pm1/2,j}^{k+1/2}$ or $p_{i,j\pm1/2}^{k+1/2}$ and $J_{rs}$, the third, fourth and fifth components of the interface fluxes $\mathbf{F}$ and $\mathbf{G}$ are calculated according to their expressions in Equation (10). For example, for the cell $(i,j)$, the third component of $\mathbf{F}$ at the interface between cells $(i+1,j)$ and $(i,j)$ is evaluated as:

$$-\frac{p_{i+1/2,j}^{k+1/2}}{q_{i+1/2,j}^{k+1/2}}\begin{vmatrix}v_{i+1/2,j}^{k+1/2} & w_{i+1/2,j}^{k+1/2}\\ Y_{i,j}^{k+1} & Z_{i,j}^{k+1}\end{vmatrix}.$$

(ii) Use Equation (13) to update $e_3$, $e_4$ and $e_5$ since there is no need to update the constants $e_1$ and $e_2$; while $e_6$, $e_7$, ..., $e_{11}$, being identical to $U$, $V$, $W$, $X$, $Y$, and $Z$, have been computed in the previous step.

 

(iii) Decode $\mathbf{E}_{i,j}^{k+1}$ to get $\mathbf{Q}_{i,j}^{k+1}$. For simplicity, all the $i$, $j$, $k$ superscripts and subscripts are dropped off. Define

$$A = \frac{(1+\gamma)}{(1-\gamma)}(J_{11}^2 + J_{12}^2 + J_{13}^2), \qquad B = \frac{2}{(\gamma-1)}(J_{11}e_3 + J_{12}e_4 + J_{13}e_5),$$

$$C = e_3^2 + e_4^2 + e_5^2 - 2K^2H.$$

Then the pressure $p$ satisfies the quadratic equation

$$Ap^2 + Bp + C = 0.$$

It can be shown that $\Delta = B^2 - 4AC \geq 0$ and the physically relevant solution for $p$ is:

$$p = \frac{-B + \sqrt{\Delta}}{2A}, \tag{19a}$$

and the other flow variables follow directly:

$$u = \frac{e_3 - pJ_{11}}{K}, \qquad v = \frac{e_4 - pJ_{12}}{K}, \qquad w = \frac{e_5 - pJ_{13}}{K}, \tag{19b}$$

$$\rho = \frac{K}{(J_{11}u + J_{12}v + J_{13}w)}. \tag{19c}$$

We thus complete the evaluation of cell conservative averages of a *computational cell* $(i, j)$. This procedure is identical to the numerical solution procedure of a Godunov-type TVD scheme. It is used here to indicate how the *subcell region conservative averaging* can be performed.

*2.2.2. The Godunov-type subcell conservative averages*. The *conservative averages* for a subcell region can be obtained in a similar way by the divergence theorem as described above, provided the fluxes around the subcell region are known. To be more specific, the cell partition shown in Figure 2(b) is used as an example, where the subcell region boundaries are straight lines parallel to the cell boundaries. For subcell region 5 in Figure 2(b), the flow variables $\mathbf{Q}_{i,j}^5$, geometrical quantities $\mathbf{T}_{i,j}^5$ and $\mathbf{S}_{i,j}^5$, where the superscript 5 denotes the fifth subcell region, are identical to the original $\mathbf{Q}$, $\mathbf{T}$ and $\mathbf{S}$ from the previous marching step at the cell $(i, j)$, since it is regarded that $\Delta\lambda$ is small enough so that the center part of the cell is not yet disturbed.

For subcell region 2 of Figure 2(b), the procedure is briefly summarized as follows:

(i) At the lower boundary, evaluate the cell interface fluxes $\mathbf{G}$ resulting from the cell interfaces between cells $(i, j)$ and $(i, j-1)$ in exactly the same manner as described in step (i) in Section 2.2.1. At the top boundary, the fluxes $\mathbf{G}$ can be evaluated directly from the free-stream state, i.e. $\mathbf{Q}$, $\mathbf{T}$ and $\mathbf{S}$ of the previous marching step at this cell $(i, j)$. At the left and right boundaries, the fluxes $\mathbf{F}$ are regarded as identical and will cancel with each other in the updating of Equation (13), since the flow in subcell region 2 is considered as approximately a 2D flow.

(ii) Follow steps (ii) and (iii) in Section 2.2.1 to find the geometrical and flow variables $\mathbf{S}^2$, $\mathbf{T}^2$ and $\mathbf{Q}^2$ in the subcell region 2. Note that the $\Delta\xi$ and $\Delta\eta$ in (13)–(18) must be correspondingly reduced to the appropriate subcell sizes.

For a corner subcell region such as 1, the fluxes of the lower and left boundaries are those resulting from cell interfaces between cells $(i, j)$ and $(i, j-1)$, $(i, j)$ and $(i-1, j)$ respectively. The fluxes of the top and right boundaries can be directly calculated from the subcell region

 

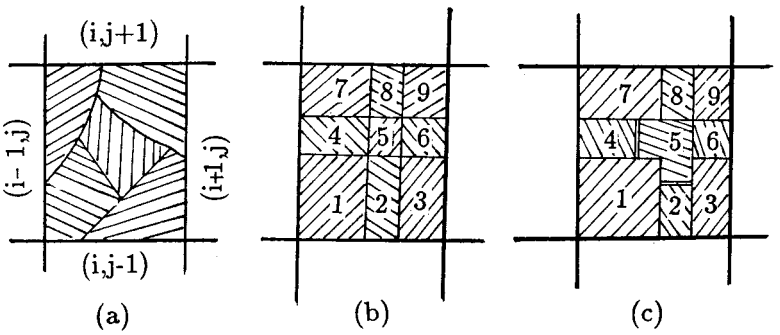


Figure 2. Partition of cell $(i, j)$ at $\lambda =$ constant plane; (a) arbitrary partition, (b) partition for Approach A, (c) partition for Approach B.

average states ($\mathbf{Q}^4$, $\mathbf{S}^4$, $\mathbf{T}^4$) and ($\mathbf{Q}^2$, $\mathbf{S}^2$, $\mathbf{T}^2$) respectively, or even directly from ($\mathbf{Q}^5$, $\mathbf{S}^5$, $\mathbf{T}^5$). The rest of the procedure is the same as in step (ii) above.

Thus, the subcell flow structure may be approximated piecewisely by several *constant conservative averages*. As these subcell averages are obtained through application of a Godunov-type scheme to subcell regions, they are termed *Godunov-type conservative averages*, representing an approximate local Riemann solution in a computational cell in the weak sense. The RCM is then applied to choose one of them as the solution of the *entire* cell. In the next two sections, the above idea of a weak Riemann solution for 3D supersonic flows will be described and tested.

## 3. LAGRANGIAN RCM FOR 3D SUPERSONIC STEADY FLOW

In the new Lagrangian formulation, a fluid particle in the physical space corresponds to a rectangular computational cell in the $\xi$–$\eta$ plane, at a given $\lambda$ (Figure 1). If the 3D Riemann solution, and hence the subcell flow structure, namely shocks, slip surfaces and expansions and their locations, were exactly known, it would have been possible to determine the cell representative state by using the RCM sampling procedure. In the 2D case, the Lagrangian RCM yields satisfactory results [14,15] since the 2D Riemann problem is exactly solvable in a convenient way. For the 3D case unfortunately, the situation is different. Due to its highly non-linear nature, a 3D Riemann problem cannot be solved by simply superimposing 2D Riemann solutions [8]. More frustratingly, based on the theoretical analysis of Zhang and Zheng [19], and the numerical results by Schultzrinne *et al.* [20] for 2D unsteady gas flow, the authors are skeptical that there is any direct and convenient way to solve a general 3D Riemann problem exactly and use it as a building block in the RCM.

However, if one does not insist on using an *exact Riemann solution*, and adopts the weak Riemann solution such as described in Section 2.2.2 as an approximation, the RCM may still be applied in an efficient way. The price paid, in general, is the *smearing of discontinuities* (*shocks* and *slip surfaces*) due to the nature of weak solutions. In the Lagrangian RCM, one enjoys the benefit from the formulation, a slip surface will remain crisp all the time [22] and only shocks may be smeared. For a weak Riemann solution, the solution region (a computa-

 

tional cell) is divided into subcell regions (Figure 2(a)); in each subregion a Godunov-type conservative average replaces the unavailable exact Riemann solution, and the complete solution is obtained by piecing together the constant state solutions in all the subregions. The most attractive advantage of the weak Riemann solution is that one does not need any detail inside the subregion, numerical fluxes around the boundarv surfaces of the subregion alone will suffice to generate the conservative average. The Lax–Wendroff theorem [29] guarantees its convergence to a weak solution, as the size of the subregion reduces to zero. Therefore, the conservative average is indeed an approximate Riemann solution.

In principle, the partition of a cell may be conducted in an arbitrary way (Figure 2(a)), as long as one can always find the numerical fluxes around each subcell region. In practice, however, a cell partition is chosen such that the numerical fluxes around all the subcell regions can be found in an easy way. Figure 2(b) illustrates a convenient partition. The rectangle cell $(i, j)$ in the $\xi$–$\eta$ plane (plane of $\lambda = \text{constant}$) is subdivided into nine rectangular subcell regions by straight lines parallel to the cell boundaries. Then the interface fluxes between different subcell regions are calculated. If a subregion boundary coincides with a shock, the shock will be resolved absolutely sharp. The following is devoted to describing two approaches, A and B, for 3D Lagrangian RCM, depending on whether a subcell region boundary coincides with the exact shock location in the computational cell.

### *3.1. Approach A*

(i) The computational domain $\lambda$–$\xi$–$\eta$ is discretized as described in Section 2.2 and the TVD (MUSCL) numerical cell interface fluxes between cell $(i, j)$ and its surrounding cells $(i, j+1)$, $(i+1, j)$, $(i, j-1)$ and $(i-1, j)$ are calculated; $i = 1, 2, \ldots, m$; $j = 1, 2, \ldots, n$.

(ii) In the $\xi$–$\eta$ plane, use straight lines parallel to the cell boundaries to divide cell $(i, j)$ into nine subcell regions (Figure 2(b)). In subcell region 5, the flow state remains unchanged and identical to the original state $\mathbf{Q}^k_{i,j}$, $\mathbf{S}^k_{i,j}$ and $\mathbf{T}^k_{i,j}$. For subcell regions 2, 4, 6 and 8, which locate in the middle of each edge of cell $(i, j)$, they are regarded as approximately 2D flow states. For example, for region 2, the numerical fluxes resulting from cell interface $(i, j)$ and $(i, j-1)$ and the fluxes resulting from the original state in region 5 are used to compute the conservative average, as described in Section 2.2.2. Fluxes at the other two regional interfaces are considered identical, based on the 2D flow assumption, and cancel each other during conservative averaging.

In subcell corner regions 1, 3, 7 and 9, genuine 3D flow occurs and they generally have different fluxes around their four boundaries. For example, when computing the conservative average in region 1, the numerical fluxes resulting from cell interfaces between $(i, j)$ and $(i, j-1)$, and $(i, j)$ and $(i-1, j)$, and the fluxes resulting from the conservative averages of regions 2 and 4 (or region 5). Details have been given in Section 2.2.2.

Once the subcell regions and their conservative average states, i.e. **Q**, **T** and **S**, are available, the unsplit RCM sampling procedure is performed. In order to do so, random numbers that distribute uniformly over a unit square $[0,1] \times [0,1]$ are needed. This step is furnished by direct multiplication of two Van der Corput pseudo-random number generators [8,30], one being with the mode (2,1) and the other with (3,2). Then the unit square is linearly mapped onto the cell $(i, j)$ (in the $\xi$–$\eta$ plane) and a one-to-one correspondence is established. For any random number pair generated in the unit square, the state of the subcell region in which the corresponding point in the cell falls will be sampled as the representative state of the cell $(i, j)$. Only one pair of random numbers is needed at a marching step for all $i$ and $j$.

 

(iii) Generate grid points (co-ordinates of cell centers) along streamlines and complete the procedure of marching forward by one step:

$$x_{i,j}^{k+1} = x_{i,j}^{k} + \frac{1}{2}\Delta\lambda^{k}\left(\frac{u_{i,j}^{k}}{q_{i,j}^{k}} + \frac{u_{i,j}^{k+1}}{q_{i,j}^{k+1}}\right),$$

$$y_{i,j}^{k+1} = y_{i,j}^{k} + \frac{1}{2}\Delta\lambda^{k}\left(\frac{v_{i,j}^{k}}{q_{i,j}^{k}} + \frac{v_{i,j}^{k+1}}{q_{i,j}^{k+1}}\right),$$

$$z_{i,j}^{k+1} = z_{i,j}^{k} + \frac{1}{2}\Delta\lambda^{k}\left(\frac{w_{i,j}^{k}}{q_{i,j}^{k}} + \frac{w_{i,j}^{k+1}}{q_{i,j}^{k+1}}\right),$$

$$i = 1, 2, \ldots, m; \qquad j = 1, 2, \ldots, n.$$

### *3.2. Approach B*

Approach B is designed in attempt to sharply capture a shock and is only a slight variation of Approach A. The difference lies in the shape of some of the subcell regions where a 2D shock is present. For example, if a 2D wedge shock is present in region 2, the upper subcell region boundary is adjusted to the exact shock location. By this way, the region conservative average state is identical to that of the exact (strong) solution. The same procedure may be carried out for other subcell regions, including the corner ones, if local 2D shocks are present there. In the preliminary tests for 2D cases, Approach B yields absolutely sharp shock resolution, while the Lagrangian formulation provides crisp resolution for slip lines (contact discontinuity). It is hoped that by such an approach, the shock resolution can be improved. Note that the subcell region boundaries are only adjusted according to 2D shock locations, since in general 3D shock locations are not directly available.

Before concluding this section, it should be emphasized that the subcell partition plays an important role in the RCM. A general guideline is that the partition should be conducted in such a way that computation accuracy and efficiency are optimized. The above approaches, A and B, are but two convenient examples.

## 4. TEST PROBLEMS FOR 3D LAGRANGIAN RCM APPROACHES A AND B

The above approaches, A and B, are tested in several model problems. When dividing a cell into subcell regions, it is divided into nine equal regions for convenience, as depicted in Figure 2(b). Only when Approach B is employed, some of the subcell regional boundaries are adjusted to the 2D shock locations (Figure 2(c)).

### *4.1. Example 1 (Approach B)*

The first problem is a trivial one. A supersonic flow of Mach number $M = 4$ past a converging channel formed by two wedges, with wedge dihedral angles as in Figure 3(a). The problem is equivalent to a 2D one, a $40 \times 40$ grid is used, in the $\xi$–$\eta$ plane. Approach B yields an excellent result. Figure 3(b) and (c) are respectively, isobars and isomachs on a typical stream surface and the pressure and Mach number distribution along a typical $\lambda$ line $AA'$ on that stream surface. The shocks are absolutely sharp, while the slip surface is crisp, enjoying the benefit from the Lagrangian formulation. Note that the pressure and Mach number distributions in Figure 3(b) and (c) agree well with the exact (2D) solutions.

 

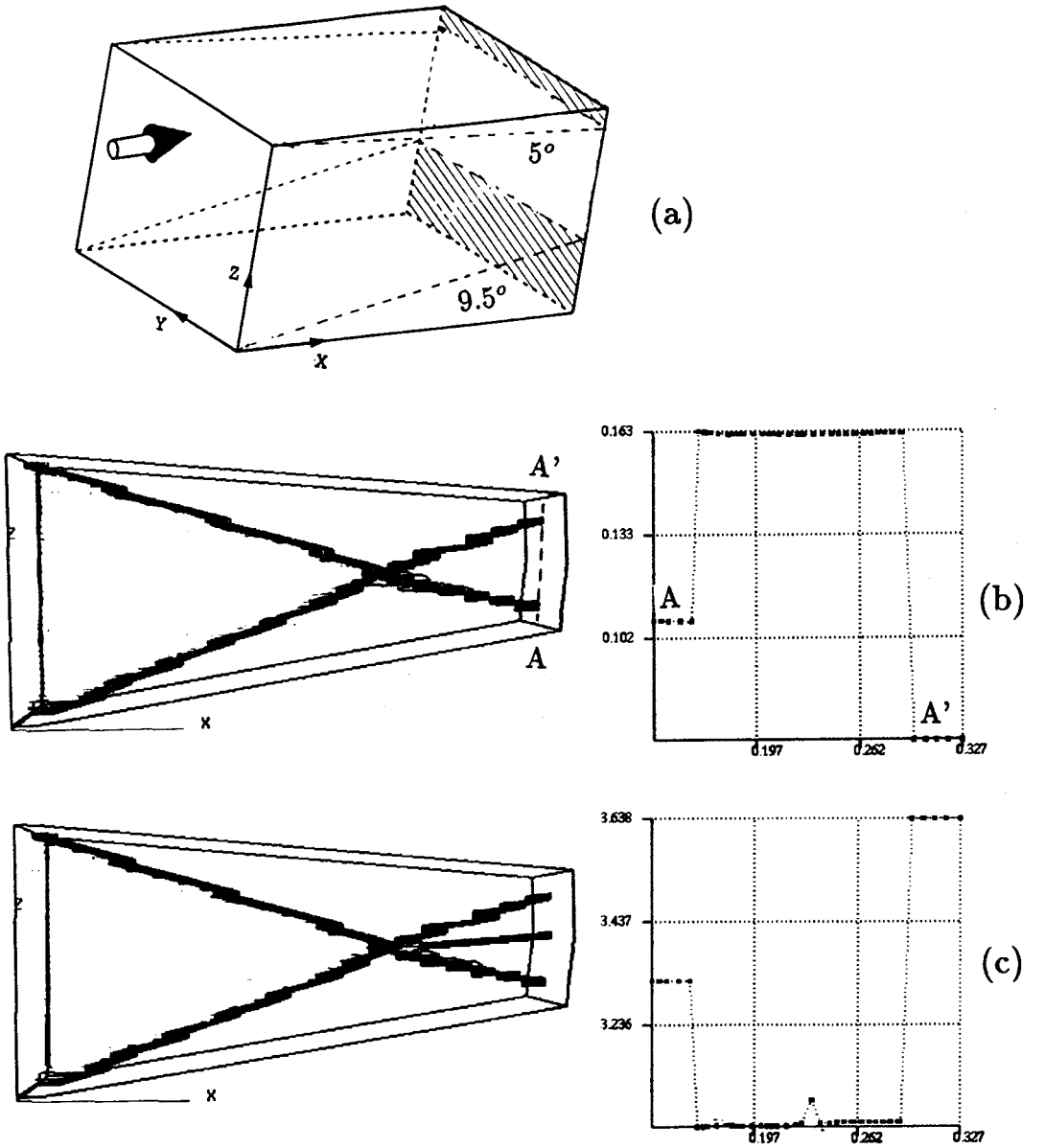


Figure 3. Supersonic flow of $M = 4$ past a converging channel, $40 \times 40$ grid; (a) sketch, (b) isobars and pressure distribution on a typical stream surface, (c) isomachs and Mach number distribution on a typical stream surface.

### *4.2. Example 2 (Approach B)*

The second model problem is a genuine 3D one: the symmetrical 3D Riemann problem. The problem is sketched in Figure 4(a). The problem was solved numerically in Reference [22] and identical states in the first and third, second and fourth quadrants were chosen, i.e.

 

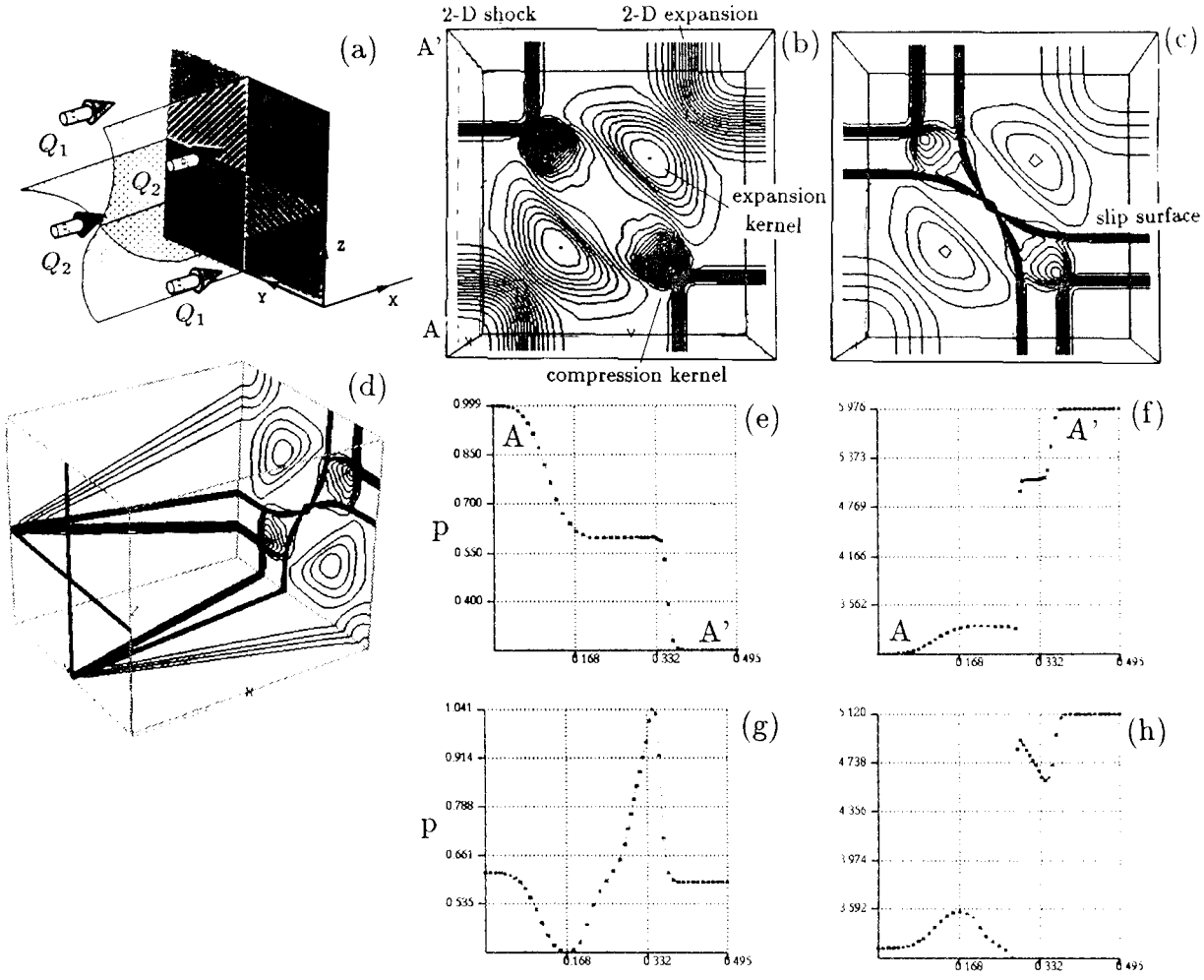


Figure 4. The symmetrical 3D Riemann problem with numerical results by the TVD scheme, $50 \times 50$ grid: (a) sketch; (b) isobars and (c) isomachs at a typical $\lambda$-plane; (d) side-view of isomachs; (e) pressure distribution and (f) Mach number distribution on an outermost stream surface; (g) pressure distribution and (h) Mach number distribution on a typical interior stream surface.

$$\mathbf{Q}_1 = (u_1, v_1, w_1, p_1, \rho_1)^T = (5, 0, 0, 0.25, 0.5)^T,$$

$$\mathbf{Q}_2 = (u_2, v_2, w_2, p_2, \rho_2)^T = (3.5, 0, 0, 1, 1)^T.$$

At $\lambda = 0$ all four states begin to interact with one another and march forward. For comparison with the RCM results, the problem is recalculated using the deterministic TVD scheme [22] with a grid of $50 \times 50$ cells. Figure 4(b) shows the isobars on a typical $\lambda$-plane, where the flow is fully developed ($\lambda = 0.45$). Due to symmetry, only part of the waves are labeled. At the outskirts around the four sides, as expected, the flow behaves like a 2D flow, with all the 2D elementary waves, namely, shocks, slip lines (surfaces) and Prandtl–Meyer expansion fans. In the interior, a genuine 3D flow occurs. A singular point (line) is produced when two 2D shocks collide with each other. This singular point forms the nucleus of a compression kernel. When two 2D expansions come across each other, they form an expansion kernel. The compression kernels and expansion kernels are connected by a continuous interior flow, except across a slip

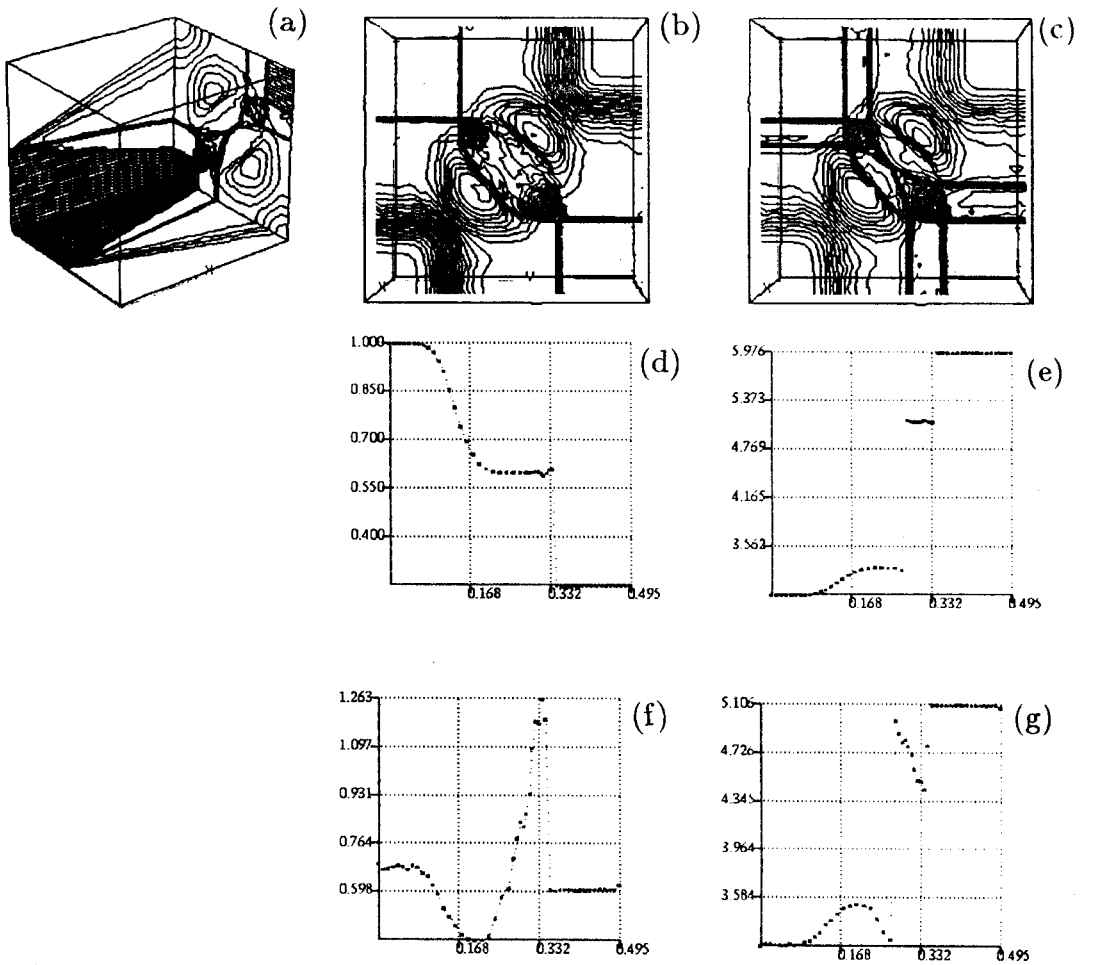


Figure 5. The RCM results for comparison with Figure 4, $50 \times 50$ grid; (a) side-view of isomachs, (b) isobars and (c) isomachs on a typical $\lambda$-plane; (d) pressure distribution and (e) Mach number distribution on an outermost stream surface; (f) pressure distribution and (g) Mach number distribution on a typical interior stream surface.

 

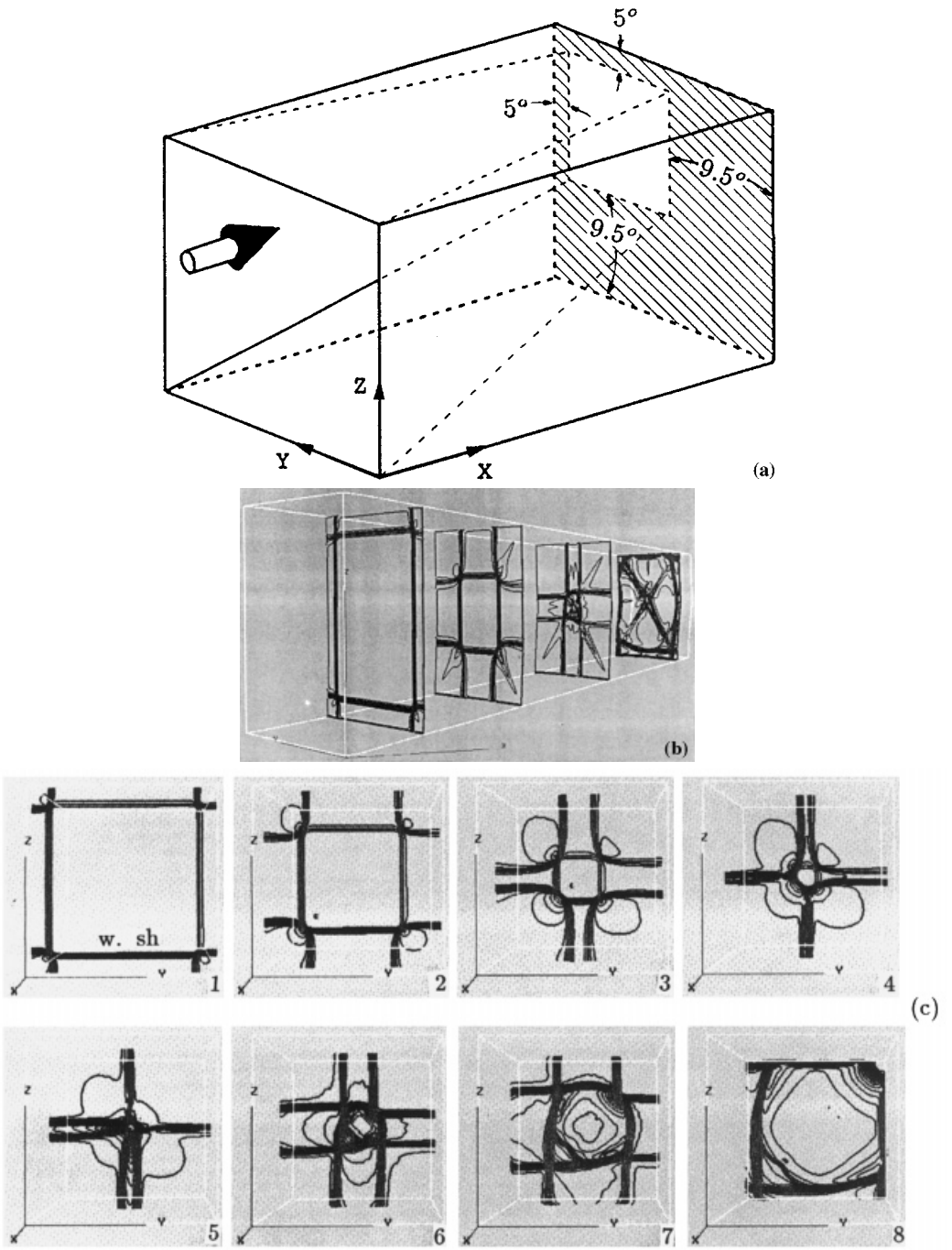


Figure 6. Supersonic flow of $M = 4$ past a converging duct, with the TVD scheme; (a) sketch, (b) marching stations, (c) isobars, (d) isopycnics with $\rho$ distributions along typical centerlines in the $y$-direction at various stations.

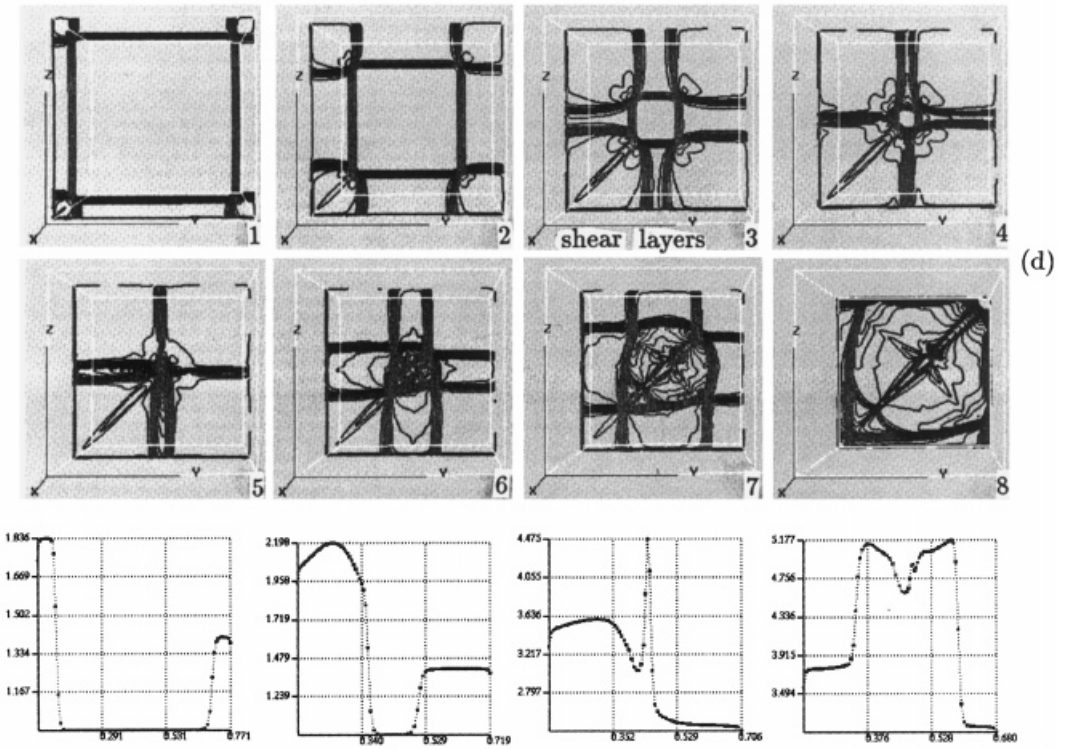


Figure 6 (*Continued*)

surface. More detailed descriptions can be found in [22]. The isomachs are presented in Figure 4(c). The slip surfaces are crisp since they are never smeared in the Lagrangian formulation. Figure 4(d) shows side-views of the isomachs at the outermost stream surfaces in $\xi$- and $\eta$-directions. On these stream surfaces, the flow is identical to a 2D Riemann problem solution. In Figure 4(e) and (f), as well as Figure 4(g) and (h), the pressure and Mach number distributions along a typical $\lambda$ line on an outermost stream surface and on a typical interior stream surface are displayed respectively. For all the contour plots in Figure 4, 40 uniform relative levels are used.

The numerical results of the Lagrangian RCM with Approach B for the same 3D Riemann problem are shown in Figure 5. The same grid is used but 100 relative contour levels are employed in the plots. Figure 5(a) corresponds to Figure 4(d) but looks sharper. Figure 5(b) and (c) correspond to Figure 4(b) and (c). Although the number of contour levels is more than doubled in Figure 5(b) and (c), these contours still look sharper than their counterparts in Figure 4, but with some noise. When comparing the pressure and Mach number distributions on the outermost stream surface (Figure 5(d) and (e) versus 4(e) and (f)), one finds that Approach B produces excellent results due to the 2D nature of the flow. However, on the interior stream surface (Figure 5(f) and (g) versus 4(g) and (h)), one finds that quite strong noise (up to 21% of the maximum pressure value) spikes appear in pressure and some in Mach number as well, although shocks are still better resolved (1 point). The reason that wiggle noise could be generated is explained as follows. Approach B is indeed a hybrid of weak and strong Riemann solutions. When it so happens that two (2D) strong shock waves of different directions, say, region 2 and 4 in Figure 4(c), are sampled in turn, this approach is equivalent

 

to applying an operator-splitting procedure, which is valid only for continuous solutions, to obtain a discontinuous 3D solution, and will certainly fail. The wiggle-like noise is then generated. Colella [8] describes such a procedure as an illegal exchange of limits. Geometrically, it is observed that for the discontinuous shock wave solution, the solution region, which is defined dynamically by the 3D shock profile, cannot be acquired by simply superimposing the 2D solution regions. For example, in the model problem of Figure 6, the embedded shocks at the corner cannot be obtained by straightforwardly superimposing the 2D straight line-shaped shocks.

Thus, Approach B turns out to be inadequate for genuine 3D flow with shocks. More convincing evidence is found in the next example. However, it still compares favorably with the hybrid operator-splitting method. In a typical example, the latter blows-up during computation (see Example 5). Thus, this paper will continue to consider Approach A, which utilizes only weak solutions in the subcell regions. It is expected that the numerical results be very similar to those of the deterministic approaches of the Godunov-type scheme, since they are all based on the Godunov-type conservative averages.

### *4.3. Example 3 (Approach B and A)*

The next problem is a supersonic flow past a duct formed by four planes. Figure 6(a) shows the sketch of a converging duct and Figure 6(b) the marching of flow. A grid of $80 \times 80$ cells is employed. In Figure 6(c) and (d), the isobars and isopycnics are presented respectively, at various typical stations, computed by the deterministic TVD finite difference scheme [22] for later comparison with the RCM results. In Figure 6(d), density distributions along the $y$-direction centerline at four typical stations are displayed. In all the contours, 100 uniform relative levels are used. At station 1 and 2, early stages of the flow, four wedge shocks (w. sh) surround the interior free-stream flow. Around the four corners of the duct body, embedded shocks (e. sh) are formed and gradually expand. The triple points and the corner shocks (c. sh) are clearly shown at station 3. Up to this stage, the embedded shocks at the four corners still expand freely. Isopycnics (density contours) at the corner areas reveals the fine structure of shear layers. At station 4, the embedded shocks surrounding the lower left corner collide with their counterparts, while the embedded shocks around the upper right corner are still developing. Station 5 records the moment when these embedded shocks collide. At station 6, due to the collision of all shocks surrounding it, the central nucleus area becomes highly compressed. In stations 7 and 8, the nucleus expands quickly. Inside the nucleus area, a new embedded shock is generated at the lower left corner and new dragon fly-like shear layers appear. The computation stops when one of the shocks around the nucleus hits the solid wall and the flow becomes subsonic after reflection.

In the RCM approach, Approach B for this problem is tested first. A grid of $40 \times 40$ cells is used. Figure 7(a)–(e) demonstrates isopycnics at five typical stations. Qualitatively comparing with Figure 6, the presence of severe noise in the numerical results is observed. It is difficult to identify the shear layers in the contour plot. From Figure 7(f) and (h), typical density distributions along the centerline in the $z$-direction, wiggles can be found and the shocks do not appear to be much sharper than that from a deterministic TVD scheme. On the other hand, despite the noise, the computation is still carried out to the stage when subsonic flow occurs, as shown in the side-view isopycnics of Figure 7(g).

Numerical results by the RCM Approach A are displayed in Figure 8. Qualitatively, the isobars and isopycnics agree well with those by the TVD scheme in Figure 6, except some minor variation and little ripples in flow patterns due to randomness. Experimentation using different numerical fluxes between subcell regions (e.g. region 1 and region 2 in Figure 2(b))

 

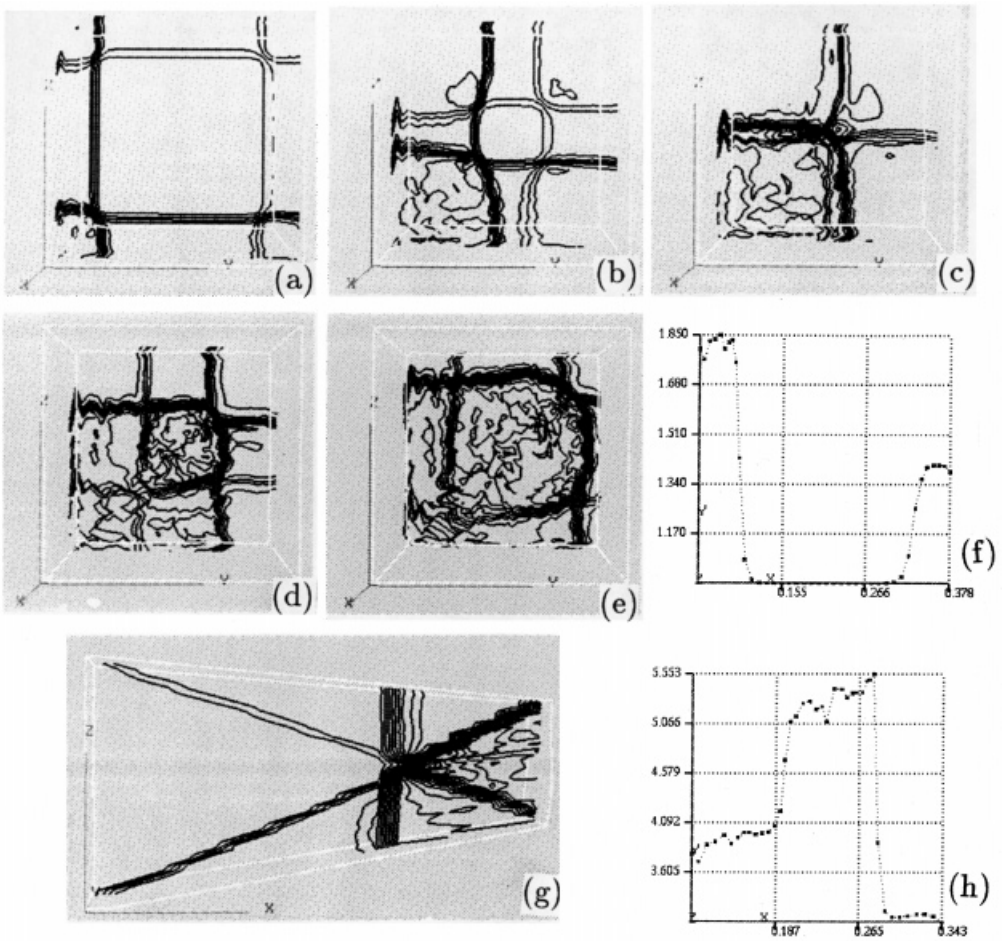


Figure 7. RCM (Approach B) results, for comparison with Figure 6, $40 \times 40$ grid; (a)–(e) isopycnics at different stations, (g) side-view on a typical stream surface, (f) and (h) density distributions along typical $\lambda$ lines at different stations.

shows that the choice of different numerical fluxes does not substantially influence the final flow pattern. Therefore, Approach A may be simplified in such a way that for the conservative subcell region averages in regions 1, 3, 7 and 9, the numerical fluxes at their region boundaries inside the cell, i.e. at the regional interfaces between regions 1 and 2, 2 and 3, 1 and 4, 3 and 6, . . . , are directly obtained from the original state (the state in region 5). Thus, the simplified Approach A needs only as much information as the deterministic TVD scheme, i.e. the numerical fluxes at all the cell interfaces. Figure 8 demonstrates the isobars and isopycnics at different stations corresponding to Figure 6. The flow patterns are almost identical, if some little random ripples in Figure 8 are ignored. It is also observed that the two distributions resemble each other in every respect, with a maximum relative error of $\approx 2\%$ possibly due to randomness and off-position in the RCM.

In the numerical experiments with Examples 2 and 3 on a CRAY-YMP supercomputer, both the deterministic TVD approach [22] and the RCM (Approach A or B) are efficient and consume about the same amount of CPU time (with a difference of about 1–2%).

 

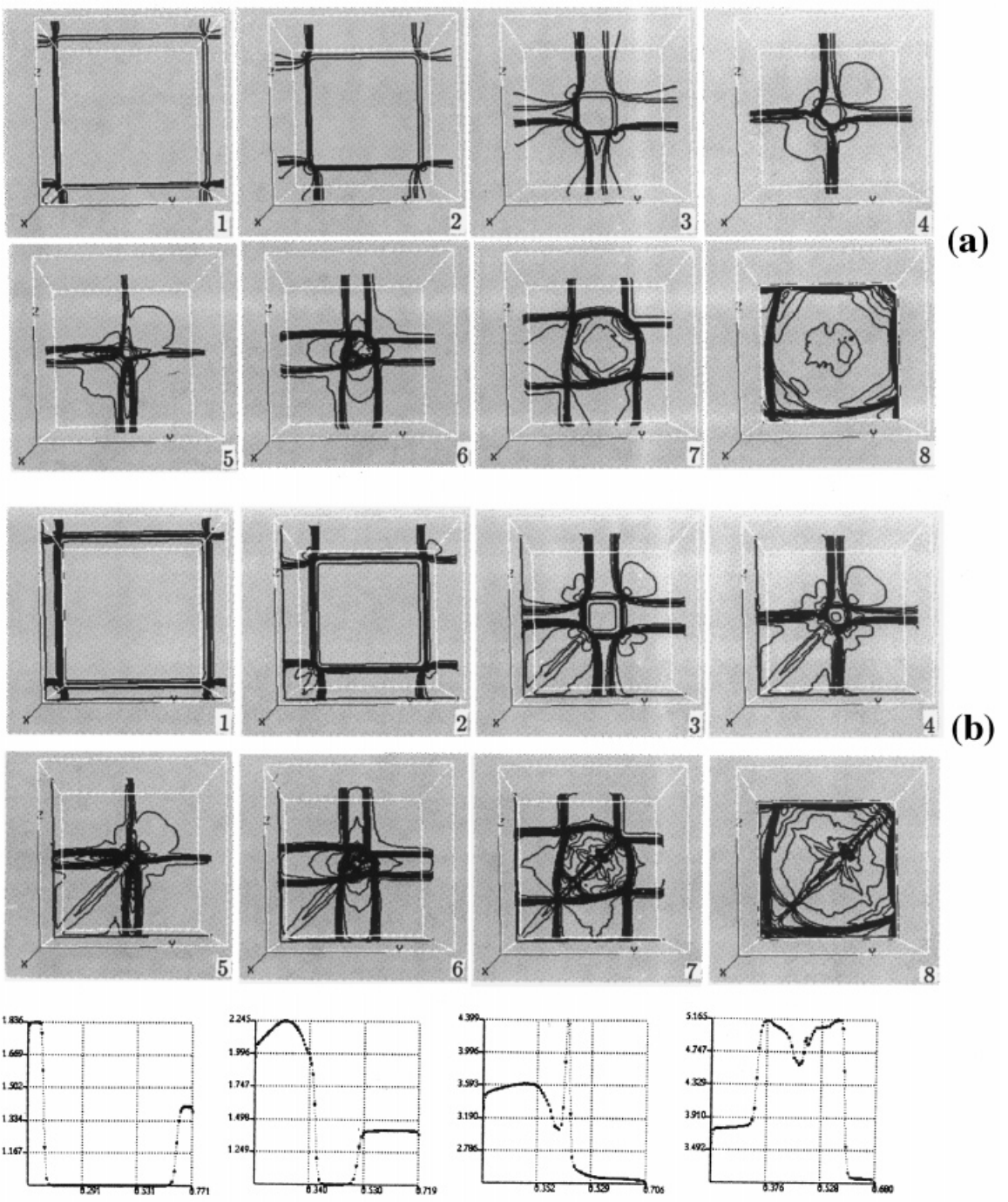


Figure 8. RCM results, for comparison with Figure 6; (a) isobars, (b) isopycnics with $\rho$ distributions along typical centerlines in the $y$-direction at the same stations as in Figure 6.

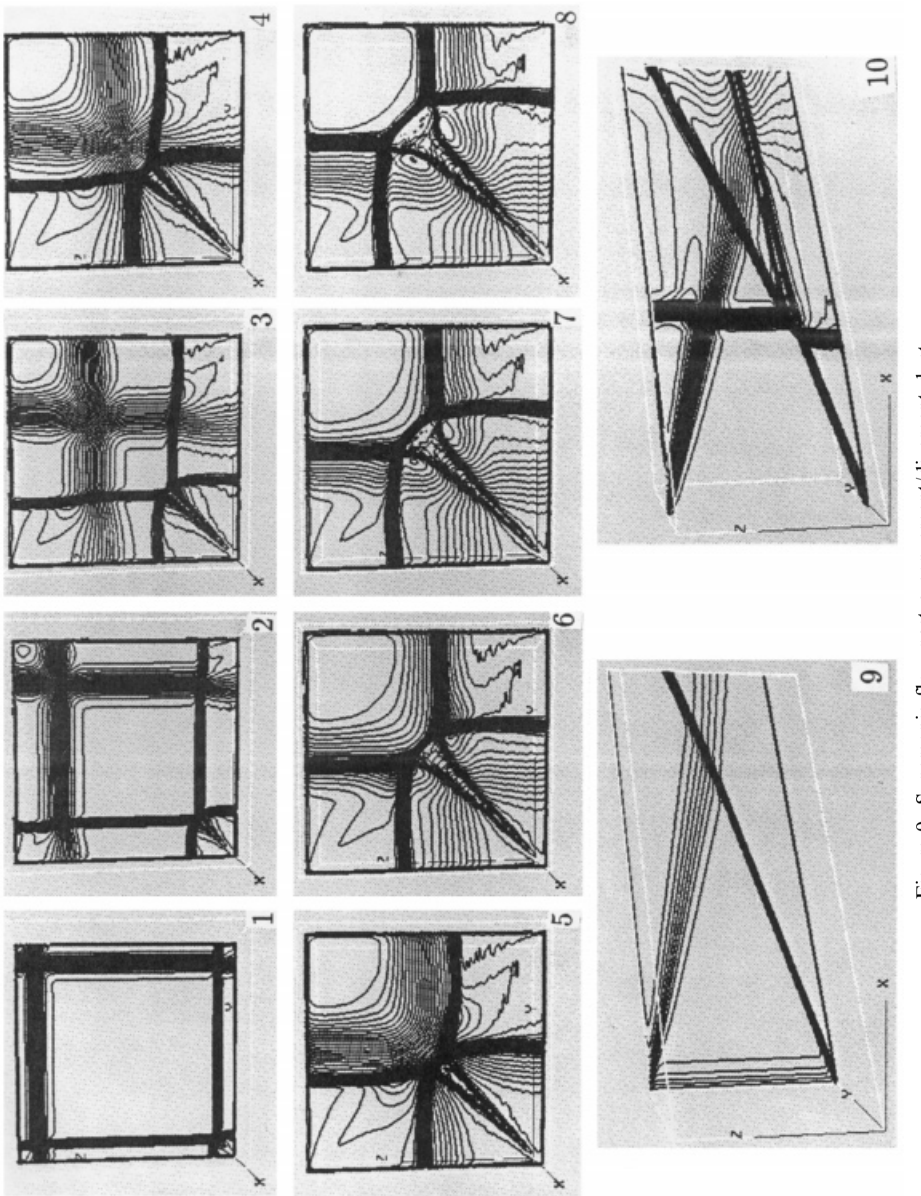


Figure 9. Supersonic flow past a convergent/divergent duct.

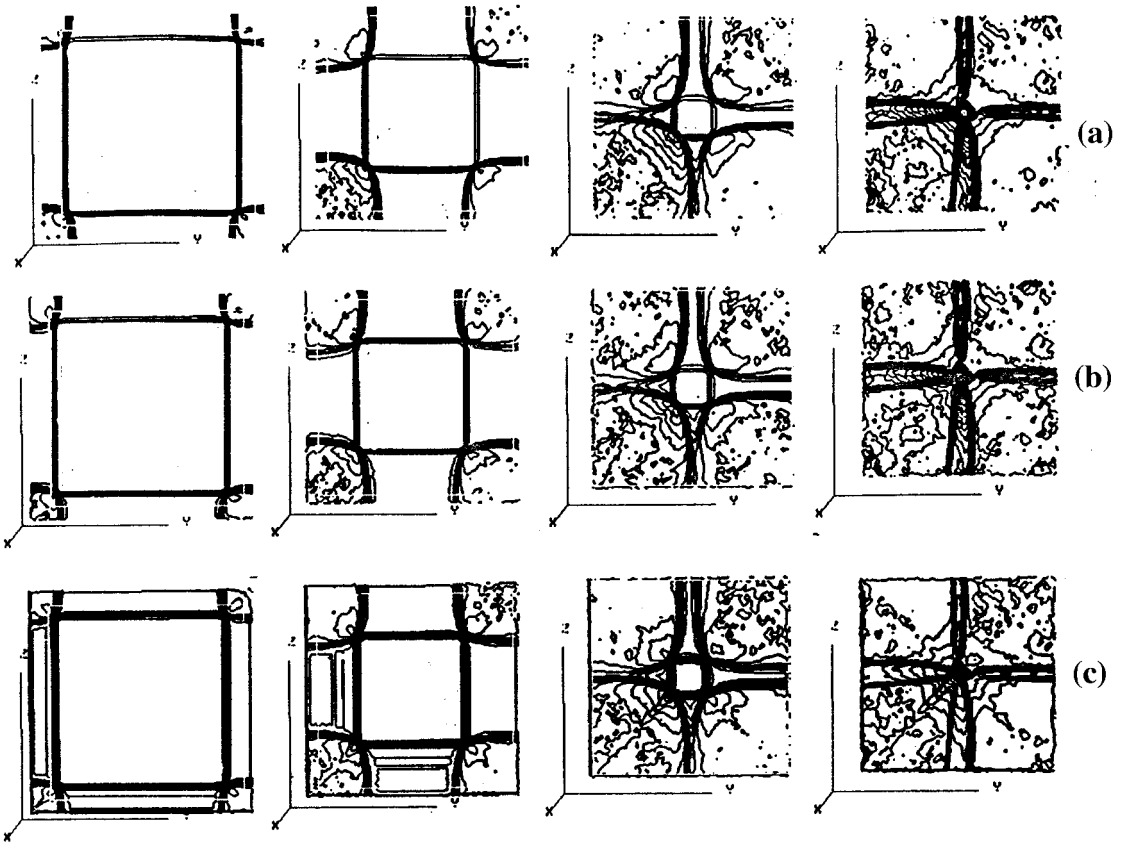


Figure 10. Numerical results by the hybrid method for the supersonic flow past a converging duct problem; (a) isobars, (b) isopycnics and (c) isomachs at four typical marching stations, with $C_0 = 0.025$. Computation is terminated half way.

### *4.4. Example 4 (Approach A)*

In Figure 9, Approach A is used to compute a supersonic convergent/divergent duct flow with the upper and left dihedral angles equal to $-5°$, and the lower and right dihedral angles equal to 5°. Again, a grid of $80 \times 80$ is used. The free stream Mach number is still $M = 4$. Figure 9(1)–(8) are the isomachs at several typical stations. They show how the 3D shocks and expansions gradually propagate into each other. Figure 9(9) and (10) display two side-views along different stream surfaces, one on the side-wall and the other in the middle. Still, in these contour plots, one finds some random, unsymmetric ripples due to the RCM.

### *4.5. Example 5 (Operator-splitting method [8])*

Finally, the performance of the operator-splitting method is demonstrated for comparison. The model problem identical to the converging duct one in Example 3 is chosen. In Figure 10, the results by the hybrid method using a grid of $80 \times 80$ with the parameter $C_0 = 0.025$ [8], are displayed. By $C_0 = 0.025$, it means the method behaves 'mostly' like a Godunov-type scheme. Still, it is found that:

 

1. the numerical results become gradually noisier;
2. the computation cannot be carried out to the same stage as the unsplit RCM does, and blows up at the moment all shocks collide together.

Obviously, the hybrid method for the 3D steady flow problems does not appear to work as well as the 2D unsteady flow reported in [8].

## 5. CONCLUDING REMARKS

In this paper, based on the idea of piecewise-constant approximations of flow structure within a computational cell and the idea of an approximate Riemann solver in the weak sense, the unsplit 3D Lagrangian RCM has been tested and validated in model problems. The RCM enjoys benefits from the Lagrangian formulation: resolution of slip surface remains crisp, and it is more convenient to choose the sampling states in a Lagrangian computational cell than in an Eulerian one. It is observed that the RCM idea of Glimm [6] works very well for high space dimension problems.

Despite the popular image of RCM (for 1D flow), the shock resolution is somewhat smeared (e.g. Approach A). The unsplit RCM itself should not be blamed for the smearing; lack of a general and efficient 3D exact Riemann solver is the culprit. Perhaps one has to be satisfied with smeared shocks in the RCM for higher space dimensions until there is a breakthrough in finding exact or better approximate Riemann solutions.

Compared with the Lagrangian deterministic TVD scheme [22], the unsplit RCM is just as efficient and yields almost identical numerical results (Approach A, if some minor randomness is ignored). In addition, the RCM is capable of exploiting information as much as possible from a Riemann solver. It has the potential for further improvement on shock resolution, if more information about the flow and the corresponding exact Riemann solver is available. For example, if in some region of the physical domain, the flow is detected to be 2D, then a hybrid of Approach A and Approach B can be used to yield crisp shock resolution in this region (e.g. Example 1 and 2). In the future, should an efficient 3D Riemann solver be available, one would be able to adjust the subcell region partition according to the 3D shock profiles, and crisp shock resolution would be achieved. This possibility has been confirmed with the preliminary test run for 2D cases.

### ACKNOWLEDGMENTS

This research is partially funded by the Research Grants Council of Hong Kong.

### REFERENCES

1. S.K. Godunov, 'Difference method of numerical computations of discontinuous solutions in hydrodynamics', *Mat. Sbornik*, **47**, 271–306 (1959).
2. S. Osher and F. Solomon, 'Upwind difference schemes for hyperbolic systems of conservation laws', *Math. Comput.*, **38**, 339–374 (1982).
3. P.L. Roe, 'Approximate Riemann solvers, parameter vectors and difference schemes', *J. Comput. Phys.*, **43**, 357–372 (1981).
4. B. van Leer, 'On the relation between the upwind differencing schemes of Godunov, Enquist–Osher and Roe', *SIAM J. Sci. Stat. Comput.*, **5**, 1–20 (1984).
5. J.L. Steger and R.F. Warming, 'Flux vector splitting of the inviscid gas dynamic equations with application to finite difference methods', *J. Comput. Phys.*, **40**, 263–293 (1981).
6. J.G. Glimm, 'Solution in the large for non-linear hyperbolic systems of equations', *Commun. Pure Appl. Math.*, **18**, 697–715 (1965).

 

7. A.J. Chorin, 'Random choice solution of hyperbolic systems', *J. Comput. Phys.*, **22**, 517–536 (1976).
8. P. Colella, 'Glimm's method for gas dynamics', *SIAM J. Sci. Stat. Comput.*, **3**, 76–110 (1982).
9. G.A. Sod, 'A numerical study of a cylindrical shock', *J. Fluid Mech.*, **83**, 785–794 (1977).
10. P. Concus and W. Proskurowski, 'Numerical solution of a non-linear hyperbolic equation by the random choice method', *J. Comput. Phys.*, **30**, 153–166 (1979).
11. A.J. Chorin, 'Vortex sheet approximation of boundary layers', *J. Comput. Phys.*, **27**, 428–442 (1978).
12. J.G. Glimm, D. Marchesin and O. McBryan, 'A numerical method for two phase flow with an unstable interface', *J. Comput. Phys.*, **39**, 179–200 (1981).
13. G. Marshall and B. Plohr, 'A random choice method for two-dimensional shock wave diffraction problems', *J. Comput. Phys.*, **56**, 410–427 (1984).
14. C.Y. Loh and W.H. Hui, 'A new Lagrangian random choice method for solving the steady Euler equations', *Comput. Fluid Dynam. J.*, **2**, 247–268 (1993).
15. C.Y. Loh and M.S. Liou, 'A Lagrangian random choice approach for supersonic real gas flows', *SIAM J. Sci. Comput.*, **15**, 1038–1058 (1994).
16. H. Oliver and H. Grönig, 'The random choice method applied to two-dimensional shock focusing and diffraction problems', *J. Comput. Phys.*, **63**, 85–106 (1986).
17. D.H. Wagner, 'The Riemann problem in two space dimensions for a single conservation law', *SIAM J. Math. Anal.*, **14**, 534–559 (1983).
18. W.B. Linquist, 'The scalar Riemann problem in two space dimensions: piecewise smoothness of solutions and its breakdown', *SIAM J. Math. Anal.*, **17**, 1178–1197 (1986).
19. T. Zhang and Y. Zheng, 'Conjecture on the structure of solutions of the Riemann problem for two dimensional gas dynamics systems', *SIAM J. Math. Anal.*, **21**, 593–630 (1990).
20. C.W. Schultz-rinne, J.P. Collins and H.M. Glaz, 'Numerical solution of the Riemann problem for two-dimensional gas dynamics', *SIAM J. Sci. Comput.*, **14**, 1394–1414 (1993).
21. A. Harten and P.D. Lax, 'A random choice finite difference scheme for hyperbolic conservation laws', *SIAM J. Numer. Anal.*, **18**, 289–315 (1981).
22. C.Y. Loh and M.S. Liou, 'A new Lagrangian method for three-dimensional steady supersonic flow computation', *J. Comput. Phys.*, **113**, 224–248 (1994).
23. C.Y. Loh and W.H. Hui, 'A new Lagrangian method for steady supersonic flow computation, Part. I: Godunov scheme', *J. Comput. Phys.*, **89**, 207–240 (1990).
24. C.Y. Loh and W.H. Hui, 'Application of Lagrangian time to steady supersonic airfoil', *AIAA Paper 89-1963-CP*, 1989.
25. W.H. Hui and Y.C. Zhao, 'A generalized Lagrangian method for solving the Euler equations', in A. Donato and F. Oliveri (Eds.) *Proc. 4th Int. Conf. on Hyperbolic Problems* (pp. 336–346), Notes on Numerical Fluid Mechanics, Vieweg, Wiesbaden, 1993.
26. W.H. Hui, 'Generalized Lagrangian formulation of computational fluid dynamics', Comput. Fluid Dynam. Rev., **1**, 382–398 (1995).
27. M.G. Crandall and A. Madja, 'Monotonic difference approximations for scalar conservation laws', *Math. Comp.*, **34**, 1–21 (1980).
28. F. Coquel and P. LeFloch, 'Convergence of finite difference schemes for conservation laws in several space dimensions: a general theory', *SIAM J. Numer. Anal.*, **30**, 675–700 (1993).
29. P. Lax and B. Wendroff, 'System of conservation laws', *Commun. Pure Appl. Math.*, **8**, 217–237 (1960).
30. J.M. Hammersley and D.C. Handscromb, *Monte Carlo Methods*, Methuen, London, 1964.
31. W.H. Hui and C.Y. Loh, 'A new Lagrangian method for steady supersonic flow computation, Part II, slip-line resolution', *J. Comput. Phys.*, **103**, 450–464 (1993).
32. W.H. Hui and Y. He, 'Hyperbolicity and optimal co-ordinates for the three-dimensional supersonic Euler equations', *SIAM J. Appl. Math.*, **57**, 893–928 (1997).

 