INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2000; **33**: 355–374

A pressure-based unstructured grid method for all-speed flows

Fue-Sang Lien*,[1]

Department of Mechanical Engineering, University of Waterloo, Waterloo, Ontario, Canada N2L 3G1

SUMMARY

An all-speed algorithm based on the SIMPLE pressure-correction scheme and the 'retarded-density' approach has been formulated and implemented within an unstructured grid, finite volume (FV) scheme for both incompressible and compressible flows, the latter involving interaction of shock waves. The collocated storage arrangement for all variables is adopted, and the checkerboard oscillations are eliminated by using a pressure-weighted interpolation method, similar to that of Rhie and Chow [Numerical study of the turbulent flow past an airfoil with trailing edge separation. *AIAA Journal* 1983; **21**: 1525]. The solution accuracy is greatly enhanced when a higher-order convection scheme combined with adaptive mesh refinement (AMR) are used. Copyright © 2000 John Wiley & Sons, Ltd.

KEY WORDS: adoptive mesh refinement; higher-order convection scheme; interpolation scheme; unstructured grid methods

1. INTRODUCTION

Unstructured grid methods have become increasingly popular in the development of computational fluid dynamics (CFD) technology to solve complex flow problems of industrial relevance. In the past, the finite volume method (FVM) has been combined with the curvilinear co-ordinate transformation to deal with gently curved geometries. For highly complex configurations, such as a multi-element airfoil, the common practice is to introduce the multi-block algorithm within a structured grid finite volume (FV) code [1]. Assuming that each block contains only one control volume, then this approach is equivalent to the unstructured grid FV scheme. So the key issue is to adopt a data structure which does not rely on the (I, J) indices. This is because a pair of (I, J) indices is generally related to (x, y) or (ξ, η) co-ordinate directions within the structured grid environment. Within the unstructured grid method, the definition of a co-ordinate system is not necessary. Instead, a different data structure has to be

* Correspondence to: Department of Mechanical Engineering, University of Waterloo, Waterloo, Ontario, Canada N2L 3G1.
[1] www:http://mecheng1.uwaterloo.ca:80/ ~ fslien

Received 15 June 1998
Revised 28 June 1999

adopted. One might choose either a volume-based or an edge-based data structure; both require information of adjacent neighbours and points forming the volume or the edge.

In addition to geometric flexibilities, a stable implementation of physical models, such as turbulence models at the Reynolds stress closure level, is equally important when developing a CFD code. Most of the turbulent flow calculations, particularly those using a higher-order turbulence closure, were performed with pressure-based FV methods. This is because sufficient experience of implementing consistent boundary conditions and preventing numerical instabilities has been accumulated for pressure-based methods since the early 1970s. Within the pressure-based methods, the staggered grid approach is favoured initially to prevent checkerboard oscillations [2]. However, applying this approach to a curvilinear co-ordinate system, particularly in conjunction with a multi-grid method, is cumbersome. The staggered grid approach is then superseded by the collocated grid approach, after the paper of Rhie and Chow in 1983 [3], which was originally designed for structured grid methods. Extending this idea to unstructured grid methods requires careful thought. A simple and yet efficient interpolation scheme is proposed here, and details will be addressed in Section 2.

Modelling shock-induced separation within turbulent boundary layers can be very challenging from both numerical and physical view points. In the inviscid regions, the flow is hyperbolic and involves the interaction of shock waves. Within the boundary layer and recirculation regions, the flow is low-speed and elliptic in nature, which can be 'stiff' for density-based methods as a result of sound speed approaching infinity. Although this problem can be alleviated by using the artificial compressibility method [4], pressure-based methods are generally more efficient. However, in order to capture shock waves, the pressure (or the pressure-correction) equation needs to be modified when the flow becomes supersonic. The simplest approach is to introduce artificial dissipation via the 'density retardation' method [5]. Application of this method in conjunction with the SIMPLE algorithm to turbulent shock–boundary layer interaction problems has been reported by Lien and Leschziner [6] using the structured grid approach. Extending the retarded-density approach to an unstructured grid method will be considered here, and its performance in contrast to the structured grid approach will be discussed in Section 3.

The ultimate goal is to develop an all-speed numerical procedure based on the cell-centred unstructured grid approach with application to transonic turbulent flow problems using an advanced turbulence closure, such as the non-linear eddy–viscosity model and second-moment closure. The numerical accuracy and efficiency will be enhanced by use of the adaptive mesh refinement (AMR) technique. The present paper is the first attempt to achieve the above goal and the emphasis here is on physical modelling. The remainder of the paper is divided into two principal sections before drawing conclusions. First, the numerical details, including a higher-order convection scheme, the shock-capturing SIMPLE algorithm and the AMR technique, will be addressed. This is followed by several validation tests, ranging from a scalar transport problem, a laminar recirculating flow to a supersonic inviscid flow.

2. UNSTRUCTURED GRID METHOD

2.1. Governing equations

The numerical approximation starts with the following two-dimensional Navier–Stokes (NS) equations, which govern the mass as well as momentum conservation:

continuity equation

$$\frac{\partial}{\partial t}(\rho)+\frac{\partial}{\partial x}(\rho u)+\frac{\partial}{\partial y}(\rho v)=0 \tag{1}$$

x momentum equation

$$\frac{\partial}{\partial t}(\rho u)+\frac{\partial}{\partial x}(\rho uu)+\frac{\partial}{\partial y}(\rho vu)=-\frac{\partial p}{\partial x}+\frac{\partial}{\partial x}\left(\mu\frac{\partial u}{\partial x}\right)+\frac{\partial}{\partial y}\left(\mu\frac{\partial u}{\partial y}\right) \tag{2}$$

y momentum equation

$$\frac{\partial}{\partial t}(\rho v)+\frac{\partial}{\partial x}(\rho uv)+\frac{\partial}{\partial y}(\rho vv)=-\frac{\partial p}{\partial y}+\frac{\partial}{\partial x}\left(\mu\frac{\partial v}{\partial x}\right)+\frac{\partial}{\partial y}\left(\mu\frac{\partial v}{\partial y}\right) \tag{3}$$

In order to apply the above equations to a solution domain of irregular shapes, Equations (1)–(3) need to be transformed using either the chain rule, i.e.

$$\frac{\partial\phi}{\partial x}=\phi_\xi\xi_x+\phi_\eta\eta_x, \qquad \frac{\partial\phi}{\partial y}=\phi_\xi\xi_y+\phi_\eta\eta_y \tag{4}$$

or the Green Theorem, i.e.

$$\frac{\partial\phi}{\partial x}=\frac{1}{\Omega}\oint\phi\ \mathrm{d}y, \qquad \frac{\partial\phi}{\partial y}=-\frac{1}{\Omega}\oint\phi\ \mathrm{d}x \tag{5}$$

where ϕ stands for the momentum components or the pressure. The former is widely employed within the structured grid environment, in which a quadrilateral-shaped finite volume in two-dimensional form is often used with one of the curvilinear co-ordinates penetrating the midpoint of each face. The latter, commonly adopted for the unstructured grid methods, does not require curvilinear co-ordinates as the evaluations of first-order derivatives with respect to x and y are performed on a Cartesian domain. As a result, even a polygon with an arbitrary number of edges can be used as a finite volume, which enhances significantly its geometrical flexibility in tackling complex configurations.

The remainder of this section is divided into five subsections. A cell-based data structure adopted in the present study is explained in Section 2.2. Integration of convective and diffusive fluxes over a finite volume and the resulting discretized equations, corresponding to Equations (2) and (3), are addressed in Section 2.3. The continuity equation (1), solved *indirectly* using a pressure-correction scheme, is described in Section 2.4, and its extension to supersonic conditions is given in Section 2.5. Section 2.6 contains AMR issues, which are based on the algorithms of Rivara [7] and Chen *et al.* [8].

2.2. Storage arrangement and data structure

A collocated storage arrangement for momentum components, pressure and density within a triangular finite volume is adopted here. The connectivity matrices consist of *forming points* and *neighbours*, which are denoted by 'no_fp(n,no_edges)' and 'no_nb(n,no_edges)' respectively in a FORTRAN program to follow. The basic idea of this data structure, in contrast to its structured grid counterpart, can be conveyed by reference to Figure 1 and Table I. As can be seen, for a structured grid the variable ϕ is stored in the form of a two-dimensional array $\phi(I^c, J^c)$. Its four neighbours can be easily identified as $\phi(I^c \pm 1, J^c \pm 1)$, with I and J pointing to x and y (or ξ and η in a curvilinear co-ordinate system) respectively. The same data structure also applies to x and y co-ordinates of the forming points, which are stored at the vertices of a finite volume. For the unstructured grid approach, no (I, J) indices are used. Instead, all neighbours and forming points are numbered in the counterclockwise direction, the same as that adopted in the Green Theorem.

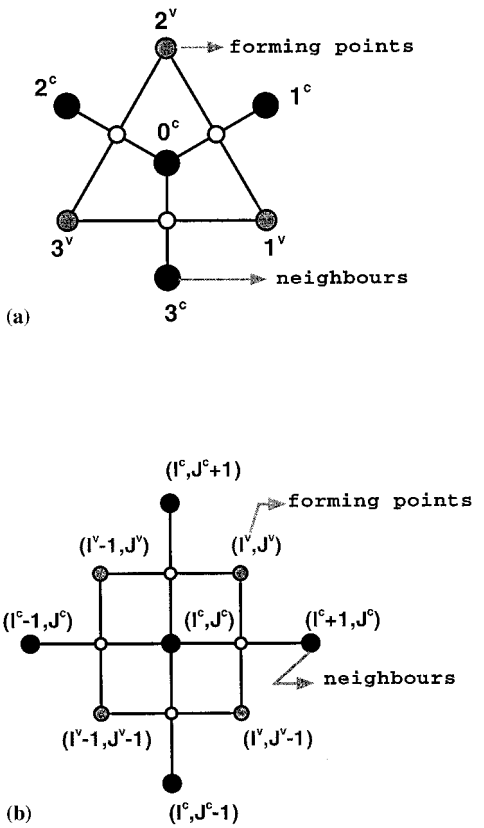

Figure 1. Data structure: (a) for an unstructured grid; (b) for a structured grid.

Table I. Data structures for structured grid and unstructured grid approaches.

Structured grid approach		
Variables	Neighbours	Forming points
$\phi(I^c, J^c)$	(I^c+1, J^c), (I^c-1, J^c), (I^c, J^c+1), (I^c, J^c-1)	(I^v, J^v), (I^v-1, J^v) (I^v-1, J^v-1), (I^v, J^v-1)
Unstructured grid approach		
Variables	Neighbours	Forming points
$\phi(n)$	$(1^c, 2^c, 3^c)$	$(1^v, 2^v, 3^v)$

2.3. Finite volume discretization

2.3.1. Convection approximation. Integration of the convective terms in Equations (2) and (3) over a finite volume shown in Figure 2(a), and application of the Green Theorem gives

$$\iint \left(\frac{\partial \rho u \phi}{\partial x} + \frac{\partial \rho v \phi}{\partial y} \right) \mathrm{d}x \, \mathrm{d}y = \sum C_{0-m} \phi_{0-m} \tag{6}$$

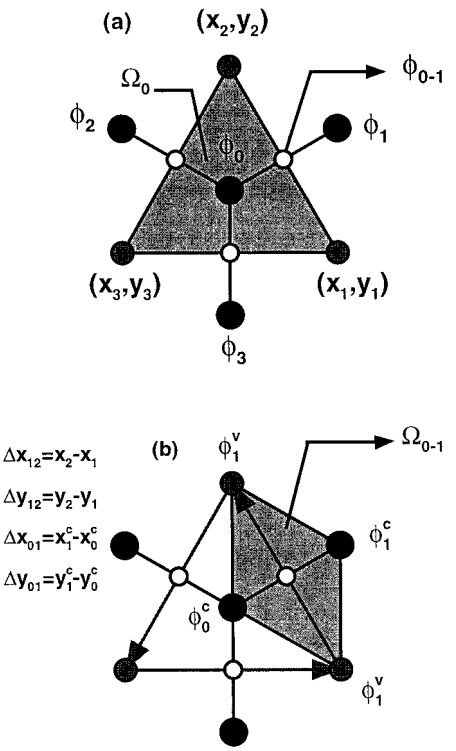

Figure 2. Integration areas: (a) for convective fluxes; (b) for diffusive fluxes.

where $\phi = u, v$, the mass fluxes

$$C_{0-1} = \rho(u_{0-1}\Delta y_{12} - v_{0-1}\Delta x_{12})$$
$$C_{0-2} = \rho(u_{0-2}\Delta y_{23} - v_{0-2}\Delta x_{23}) \tag{7}$$
$$C_{0-3} = \rho(u_{0-3}\Delta y_{31} - v_{0-3}\Delta x_{31})$$

and u_{0-m}, v_{0-m} $(m = 1, 2, 3)$ are evaluated by the use of a pressure-weighted interpolation method similar to that proposed by Rhie and Chow for the structured grid method, which will be explained in Section 2.4. The face value of ϕ, i.e $\phi_{0-\mathrm{m}}$ in Equation (6) is approximated by the following second-order upwind scheme:

$$\begin{aligned} \phi_{0-m} &= \underline{\phi_0} + (\nabla\phi)_0 \cdot \frac{\Delta\vec{r}_{0-m}}{2}, \qquad \text{if } C_{0-m} > 0 \\ \phi_{0-m} &= \underline{\phi_m} - (\nabla\phi)_m \cdot \frac{\Delta\vec{r}_{0-m}}{2}, \qquad \text{if } C_{0-m} < 0 \end{aligned} \tag{8}$$

where $\nabla\phi = (\partial\phi/\partial x, \partial\phi/\partial y)$ and $\Delta\vec{r} = (\Delta x, \Delta y)$. The underlined terms, corresponding to a first-order upwind scheme, form part of the matrix coefficients and are treated implicitly. The rest are lumped into an additional source term S_ϕ^{DC}—the 'deferred-correction' approach—which is given as follows:

$$S_\phi^{\mathrm{DC}} = \frac{1}{2}\sum_{m=1,2,3}[C_{0-m}^{+}(\nabla\phi)_0 \cdot \Delta\vec{r}_{0-m} - C_{0-m}^{-}(\nabla\phi)_m \cdot \Delta\vec{r}_{0-m}] \tag{9}$$

where $C_{0-m}^{\pm} = \frac{1}{2}(C_{0-m} \pm |C_{0-m}|)$.

2.3.2. Diffusion approximation. The values of $\partial\phi/\partial x$ and $\partial\phi/\partial y$ at the edge 0–1 can be evaluated using Equation (5) over the quadrilateral shaded area shown in Figure 2(b). This gives

$$\left(\frac{\partial\phi}{\partial x}\right)_{0-1} = \frac{1}{2\Omega_{0-1}}[(\phi_1^{\mathrm{c}} - \phi_0^{\mathrm{c}})\Delta y_{12} - (\phi_2^{\mathrm{v}} - \phi_1^{\mathrm{v}})\Delta y_{01}] \tag{10}$$

$$\left(\frac{\partial\phi}{\partial y}\right)_{0-1} = \frac{-1}{2\Omega_{0-1}}[(\phi_1^{\mathrm{c}} - \phi_0^{\mathrm{c}})\Delta x_{12} - (\phi_2^{\mathrm{v}} - \phi_1^{\mathrm{v}})\Delta x_{01}] \tag{11}$$

The values of ϕ^{v} are obtained by averaging over all surroundings cell-centred nodal values for any vertex being considered, which generally requires local searching. This problem can be

circumvented by performing three 'do' loops exemplified below, where 'n_cell' and 'n_grid' are the total number of cells and grids respectively.

- **Initialization:**

  do $n^v = 1$,n_grid
  $\phi^v(n^v) = 0$
  n_total(n^v) = 0
  end do

- **Equal distribution of ϕ^c to ϕ^v:**

  do $n^c = 1$,n_cell
  ϕ^v(no_fp(n^c,1)) = ϕ^v(no_fp(n^c,1)) + $\phi^c(n^c)$
  ϕ^v(no_fp(n^c,2)) = ϕ^v(no_fp(n^c,2)) + $\phi^c(n^c)$
  ϕ^v(no_fp(n^c,3)) = ϕ^v(no_fp(n^c,3)) + $\phi^c(n^c)$
  n_total(no_fp(n^c,1)) = n_total(no_fp(n^c,1) + 1
  n_total(no_fp(n^c,2)) = n_total(no_fp(n^c,2) + 1
  n_total(no_fp(n^c,3)) = n_total(no_fp(n^c,3) + 1
  end do

- **Averaging ϕ^v by the total number of surrounding cells:**

  do $n^v = 1$,n_grid
  $\phi^v(n^v) = \phi^v(n^v)$/float(n_total($n^v$))
  end do

Similar expressions apply to the edges 0–2 and 0–3. The end result of integrating the diffusion terms pertaining to Equations (2) and (3) is

$$\begin{aligned}
&\iint\left[\frac{\partial}{\partial x}(\Gamma\phi_x)+\frac{\partial}{\partial y}(\Gamma\phi y)\right]\mathrm{d}x\,\mathrm{d}y\\
&=\frac{\Gamma_{0-1}}{2\Omega_{0-1}}\left[\underline{(\phi_1^c-\phi_0^c)(\Delta x_{12}^2+\Delta y_{12}^2)}-(\phi_2^v-\phi_1^v)(\Delta x_{01}\Delta x_{12}+\Delta y_{01}\Delta y_{12})\right]\\
&\quad+\frac{\Gamma_{0-2}}{2\Omega_{0-2}}\left[\underline{(\phi_2^c-\phi_0^c)(\Delta x_{23}^2+\Delta y_{23}^2)}-(\phi_3^v-\phi_2^v)(\Delta x_{02}\Delta x_{23}+\Delta y_{02}\Delta y_{23})\right]\\
&\quad+\frac{\Gamma_{0-3}}{2\Omega_{0-3}}\left[\underline{(\phi_3^c-\phi_0^c)(\Delta x_{31}^2+\Delta y_{31}^2)}-(\phi_1^v-\phi_3^v)(\Delta x_{03}\Delta x_{31}+\Delta y_{03}\Delta y_{31})\right]
\end{aligned}\tag{12}$$

Here $\Gamma = \mu$—the laminar viscosity. Note that the underlined terms pertain to the *normal diffusion*, which contribute to the implicit part of the solution matrix and the rest relate to the *cross-diffusion*, which are lumped into the source term S_ϕ and treated explicitly. It is worthwhile to point out that under certain circumstance, the above discretization is equivalent to its structured grid counterpart, which is based on the curvilinear co-ordinate transformation. Consider a square finite volume shown in Figure 3. The first right-hand side term in Equation (12), corresponding to Figure 3(a), is the same as the following expression, which relates to Figure 3(b) in the (ξ, η) co-ordinate system:

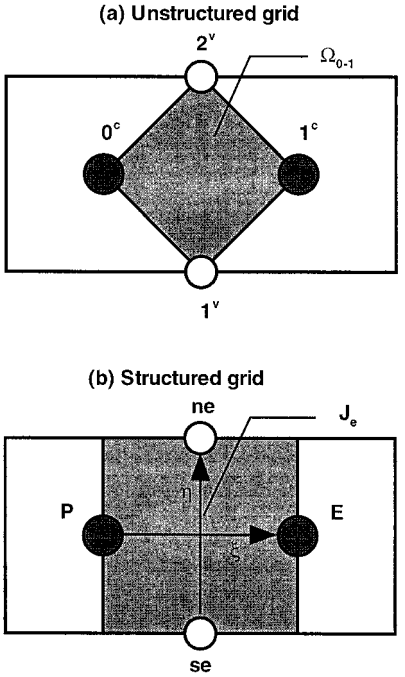

Figure 3. Integration areas.

$$\frac{\Gamma_e}{J_e}\left[(\phi_E-\phi_P)(x_\eta^2+y_\eta^2)_e-(\phi_{ne}-\phi_{se})(x_\xi x_\eta+y_\xi y_\eta)_e\right] \tag{13}$$

Proof:
Since

$$\phi_E\leftrightarrow\phi_1^{\mathrm{c}},\qquad \phi_P\leftrightarrow\phi_0^{\mathrm{c}},\qquad \phi_{ne}\leftrightarrow\phi_2^{\mathrm{v}},\qquad \phi_{se}\leftrightarrow\phi_1^{\mathrm{v}}$$

$$\frac{(x_\eta^2+y_\eta^2)_e}{J_e}\leftrightarrow\frac{(\Delta x_{12}^2+\Delta y_{12}^2)}{2\Omega_{0-1}}$$

$$\frac{(x_\xi x_\eta+y_\xi y_\eta)_e}{J_e}\leftrightarrow\frac{(\Delta x_{01}\Delta x_{12}+\Delta y_{01}\Delta y_{12})}{2\Omega_{0-1}}$$

and $\Gamma_e\leftrightarrow\Gamma_{0-1}$, both expressions are identical.

2.3.3. Coefficient assembly. Insertion of the above convection and diffusive fluxes into the volume-weighted equation gives

$$A_0\phi_0 = \sum_{m=1,2,3} A_m\phi_m + S_\phi \tag{14}$$

where

$$\begin{aligned} A_1 &= \mathrm{MAX}(-C_{0-1}, 0) + \frac{\Gamma_{0-1}(\Delta x_{12}^2 + \Delta y_{12}^2)}{2\Omega_{0-1}} \\ A_2 &= \mathrm{MAX}(-C_{0-2}, 0) + \frac{\Gamma_{0-2}(\Delta x_{23}^2 + \Delta y_{23}^2)}{2\Omega_{0-2}} \\ A_3 &= \mathrm{MAX}(-C_{0-3}, 0) + \frac{\Gamma_{0-3}(\Delta x_{31}^2 + \Delta y_{31}^2)}{2\Omega_{0-3}} \\ A_0 &= A_1 + A_2 + A_3 \end{aligned} \tag{15}$$

The source term S_ϕ contains the cross-diffusion term

$$\begin{aligned} &-\frac{\Gamma_{0-1}}{2\Omega_{0-1}}[(\phi_2^{\mathrm{v}} - \phi_1^{\mathrm{v}})(\Delta x_{01}\Delta x_{12} + \Delta y_{01}\Delta y_{12})] - \frac{\Gamma_{0-2}}{2\Omega_{0-2}}[(\phi_3^{\mathrm{v}} - \phi_2^{\mathrm{v}})(\Delta x_{02}\Delta x_{23} + \Delta y_{02}\Delta y_{23})] \\ &-\frac{\Gamma_{0-3}}{2\Omega_{0-3}}[(\phi_1^{\mathrm{v}} - \phi_3^{\mathrm{v}})(\Delta x_{03}\Delta x_{31} + \Delta y_{03}\Delta y_{31})] \end{aligned} \tag{16}$$

the 'deferred-correction' term defined in Equation (9), when a higher-order convection scheme is adopted, and the pressure gradient terms

$$+(p_{0-1}\Delta y_{12} + p_{0-2}\Delta y_{23} + p_{0-3}\Delta y_{31}), \quad \text{for } \phi = u \tag{17}$$

and

$$-(p_{0-1}\Delta x_{12} + p_{0-2}\Delta x_{23} + p_{0-3}\Delta x_{31}), \quad \text{for } \phi = v \tag{18}$$

At present, only the Gauss–Seidel method is used to solve Equation (14).

2.4. Pressure-correction scheme

The tendency of provoking checkerboard oscillations arising from the pressure–velocity decoupling is due to the use of the collocated storage arrangement, regardless of whether a structured grid or an unstructured grid method being used. This problem can be avoided by introducing the 'Rhie–Chow' interpolation for the evaluation of face velocities, which consists of an arithmetic average of two adjacent velocities and a third-order pressure-smoothing term.

A similar approach is adopted here to evaluate the face velocities on a triangular mesh. For example,

$$u_{0-1} = \frac{1}{2}\left[u_0 + u_1 + \left(\frac{\Omega(\partial p/\partial x)}{A_0}\right)_0 + \left(\frac{\Omega(\partial p/\partial x)}{A_0}\right)_1 \right] - \frac{2(\Omega(\partial p/\partial x))_{0-1}}{(A_0)_0 + (A_0)_1} \tag{19}$$

$$v_{0-1} = \frac{1}{2}\left[v_0 + v_1 + \left(\frac{\Omega(\partial p/\partial y)}{A_0}\right)_0 + \left(\frac{\Omega(\partial p/\partial y)}{A_0}\right)_1 \right] - \frac{2(\Omega(\partial p/\partial y))_{0-1}}{(A_0)_0 + (A_0)_1} \tag{20}$$

The cell-centred pressure gradients $[(\partial p/\partial x)_0, (\partial p/\partial x)_1, (\partial p/\partial y)_0, (\partial p/\partial y)_1]$ are discretized using the Green Theorem in Equation (5). The pressure gradients at edges, however, are approximated by the following expressions involving only two adjacent pressure nodes:

$$\frac{2(\Omega(\partial p/\partial x))_{0-1}}{(A_0)_0 + (A_0)_1} \approx \frac{2\Delta y_{12}(p_1 - p_0)}{(A_0)_0 + (A_0)_1} \tag{21}$$

$$\frac{2(\Omega(\partial p/\partial y))_{0-1}}{(A_0)_0 + (A_0)_1} \approx \frac{-2\Delta x_{12}(p_1 - p_0)}{(A_0)_0 + (A_0)_1} \tag{22}$$

Similar expressions relate to other edges. Perturb the face velocities via

$$u = u^* + u', \qquad v = v^* + v' \tag{23}$$

where, for example,

$$u'_{0-1} \sim -\frac{2\Delta y_{12}(p'_1 - p'_0)}{(A_0)_0 + (A_0)_1}, \qquad v'_{0-1} \sim \frac{2\Delta x_{12}(p'_1 - p'_0)}{(A_0)_0 + (A_0)_1} \tag{24}$$

and substitute the above into the discretized continuity equation, giving rise to the following pressure-correction equation:

$$a_0 p'_0 = \sum_{m=1,2,3} a_m p'_m - R_m \tag{25}$$

The mass imbalance, R_m, is defined by

$$R_m = C^*_{0-1} + C^*_{0-2} + C^*_{0-3} \tag{26}$$

and the coefficients a_m $(m = 0, 1, 2, 3)$ are

$$a_1 = 2\frac{\Delta x_{12}^2 + \Delta y_{12}^2}{(A_0)_0 + (A_0)_1}, \qquad a_2 = 2\frac{\Delta x_{23}^2 + \Delta y_{23}^2}{(A_0)_0 + (A_0)_2}, \qquad a_3 = 2\frac{\Delta x_{31}^2 + \Delta y_{31}^2}{(A_0)_0 + (A_0)_3},$$

$$a_0 = a_1 + a_2 + a_3 \tag{27}$$

Then the new mass fluxes and pressure field are obtained via

$$C_{0-m} = C^*_{0-m} + a_m(p'_0 - p'_m), \quad m = 1, 2, 3 \tag{28}$$

$$p = p^* + \alpha_p p' \tag{29}$$

where $\alpha_p \approx 0.4$–0.6 is the underrelaxation factor for p'. In summary, the pressure-correction scheme consists of six major steps

1. Guess the pressure field p^*.
2. Solve the momentum equations by using Equations (14)–(18) to obtain u^* and v^*.
3. Solve Equation (25) for p'.
4. Calculate p from Equation (29) by adding p' to p^*.
5. Calculate u and v and mass fluxes C_{0-m} from their starred values using the velocity-correction formulae (23), (24) and (28).
6. Treat the corrected pressure p as a new guessed pressure p^*, return to step (2), and repeat the whole procedure until a converged solution is obtained.

This operates in precisely the same manner as the SIMPLE algorithm detailed by Reference [2] for the staggered grid and by Reference [9] for the collocated grid arrangements.

2.5. Extension to supersonic condition

The extension to supersonic flows, including the shock-capturing capability, consists of three basic elements. First, the unsteady form of the Navier–Stokes equation is adopted, in which the transient term is approximated by the first-order one-sided differencing scheme. Second, the total enthalpy H_0 is assumed constant so that solving explicitly the energy equation is not required. Third, the flux variables $(\rho u, \rho v)$ rather than the primitive variables (u, v) are chosen as the dependent variables, the perturbations of which, i.e.

$$\rho u = (\rho u)^* + (\rho u)', \qquad \rho v = (\rho v)^* + (\rho v)' \tag{30}$$

are very similar to those of Equation (23). As a result, no mass-flux linearization, as used by Karki and Patankar [10], is required. Fourth, a second-order artificial dissipation based on the 'density-retardation concept' [5] is introduced, the basic concept of which can be explained by use of the potential equation in conservative form as

$$(\tilde{\rho}\Phi_x)_x + (\tilde{\rho}\Phi_y)_y = 0 \tag{31}$$

with $\tilde{\rho}$—the retarded density—given by

$$\tilde{\rho} = \rho - \bar{\mu}\rho_s \Delta s = \rho - \bar{\mu}(\cos(\beta)\rho_x \Delta x + \sin(\beta)\rho_y \Delta y) \tag{32}$$

where $\cos(\beta) = u/\sqrt{u^2 + v^2}$ and $\sin(\beta) = v/\sqrt{u^2 + v^2}$. The *monitor function* $\bar{\mu}$ in Equation (32) is defined by

$$\bar{\mu} = \mathrm{MAX}\left(0, 1 - \frac{1}{M^2}\right) \tag{33}$$

where M is the local Mach number. A physical interpretation of the above *density-biasing* concept is conveyed in Figure 4.

The same idea has been extended to Navier–Stokes solvers based on the pressure-correction SIMPLE scheme on a structured grid [6]. In one-dimensional conditions, the gradient of the convective flux of any flux variable (ρu or ρv here) may be expressed, implicitly, as follows:

$$\frac{\partial}{\partial x}\left(\frac{\overset{\sim \text{mass flux}}{\tilde{\rho}\tilde{u}} \quad \phi}{\tilde{\rho}}\right) = \frac{\partial}{\partial x}\left(\frac{\rho u \phi}{\rho - \bar{\mu}\dfrac{\partial \rho}{\partial x}\Delta x}\right) = \frac{\partial}{\partial x}(u\phi) + \underline{\frac{\partial\left[\bar{\mu}\left(\dfrac{u\phi}{\rho}\right)\dfrac{\partial \rho}{\partial x}\Delta x\right]}{\partial x}} + \mathrm{HOT} \tag{34}$$

where the right-hand side is an expanded (i.e. explicit) form of the left-hand side, with HOT denoting higher-order terms. The underlined term in Equation (34) represents a dissipative mechanism equivalent to upwind-biasing. In two-dimensional conditions on an unstructured grid, the mass flux C in Equation (7) is modified as

$$C \leftarrow \frac{C}{\tilde{\rho}} \tag{35}$$

or, on face 1^{v}–2^{v} in Figure 1, as

$$C_{0-1} = \frac{C^{+}_{0-1}}{\bar{\mu}_{0-1}\rho_0 + (1 - \bar{\mu}_{0-1})\rho_1} + \frac{C^{-}_{0-1}}{\bar{\mu}_{0-1}\rho_1 + (1 - \bar{\mu}_{0-1})\rho_0} \tag{36}$$

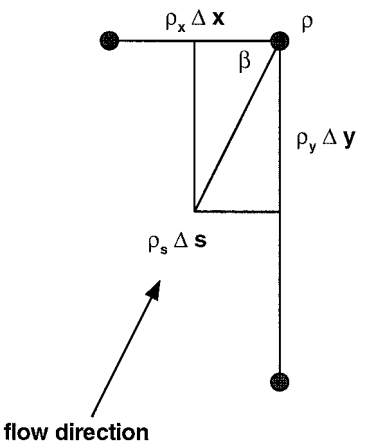

Figure 4. Schematic of density-biasing concept.

where

$$C_{0-1}^{\pm} = \frac{C_{0-1} \pm |C_{0-1}|}{2} \tag{37}$$

$$C_{0-1} = (\rho u)_{0-1}\Delta y_{12} - (\rho v)_{0-1}\Delta x_{12} \tag{38}$$

The monitoring function $\bar{\mu}_{0-1}$ defined in Equation (33), used to control the level of artificial dissipation at supersonic condition, is modified to include two free parameters κ and M_{ref}

$$\bar{\mu}_{0-1} = \text{MAX}\left[0, \kappa\left(1 - \left(\frac{M_{\text{ref}}}{M_{0-1}}\right)^2\right)\right] \tag{39}$$

Similar expressions relate to C_{0-2} and C_{0-3}. Typical values for M_{ref} and κ are approximately 0.7 and 0.55 respectively. In summary, the sequence of the present shock-capturing pressure-correction scheme consists of six major steps

1. Calculate $(\rho u)^*$ and $(\rho v)^*$ by use of p^*.
2. Introduce density retardation by modifying the convective fluxes C_{0-m} $(m = 1, 2, 3)$ according to Equations (36)–(39).
3. Evaluate R_m for all cells.
4. Solve p'-equation and then correct the velocity and pressure fields.
5. Obtain the density using constant total enthalpy assumption, i.e.

$$\rho = \frac{\gamma p}{(\gamma - 1)[H_0 - \frac{1}{2}(u^2 + v^2)]} \tag{40}$$

   where $\gamma = 1.4$ is the specific heat ratio for air.
6. Go to the next time step and repeat steps (1)–(5) until steady state is reached.

2.6. Adaptive mesh refinement

The mesh refinement techniques proposed by Rivara [7] and Chen *et al.* [8] are adopted here. A cell is flagged when 'func(ϕ)'—the sensor function—exceeds the threshold value *Cr*. The sensor function chosen here is based on the gradient or magnitude of a related property, which will be given separately for each case to follow. The value *Cr* is

$$Cr = \text{func}(\phi)_{\min} + \text{ratio}[\text{func}(\phi)_{\max} - \text{func}(\phi)_{\min}] \tag{41}$$

where 'ratio' is a user-defined free parameter. However, an *optimal* choice of *Cr* requires experience, and the refined pattern obtained initially is often seen to be unsatisfactory. Instead of adjusting the 'ratio', to which the result can be very sensitive in certain cases, a more effective approach—*the isotropic advancing front method*—is proposed here, which in essence, extends the range of refined regions by flagging neighbours of previously flagged cells repeatedly for a fixed number of times. Its schematic representation together with a practical

example—a supersonic flow over a circular bump (see Section 3.3 for details)—is illustrated in Figure 5. Then all flagged cells are bisected by simply connecting the midpoint at the longest edge to the point opposite to that edge. However, during the above refinement sequence, non-conforming triangles, i.e. triangles with at least a 'hanging' point on one of its sides, might exist. If this does happen, the adjacent coarse cells will also be bisected, which might on its own introduce additional non-conforming triangles. It is recursive and the whole refinement procedure will repeat itself until all 'hanging' points have been completely removed.

3. APPLICATION

3.1. Overview of test cases

In order to validate the newly developed unstructured grid code applicable to flows at all speeds, three tests with very different characteristics are chosen here

- convection–diffusion transport of a scalar property by a rotational field [11];
- supersonic inviscid flow over a circular bump [12];
- laminar lid-driven cavity flow.

All results obtained will be compared with the structured grid solutions returned by the STREAM code [13], which is also based on an all-speed pressure-correction algorithm, including supersonic condition. In the present study, only a two-level AMR is used for all cases to follow.

3.2. Scalar advection by a rotational velocity field

The convective–diffusive transport of a scalar step across a Cartesian mesh by a rotational motion is examined here to test the quality of a convection scheme, particularly one formulated on the basis of one-dimensional consideration but applied to multi-dimensional transport. The steady transport of a scalar in two-dimensional space is governed by the equation

$$\frac{\partial u\phi}{\partial x}+\frac{\partial v\phi}{\partial y}=\frac{1}{P_e}\left(\frac{\partial^2\phi}{\partial x^2}+\frac{\partial^2\phi}{\partial y^2}\right) \quad \text{for} \quad -1\le x\le 1, \quad 0\le y\le 1 \tag{42}$$

subject to

$$u=2y(1-x^2), \qquad v=-2x(1-y^2) \tag{43}$$

The boundary conditions pertaining to (42) are

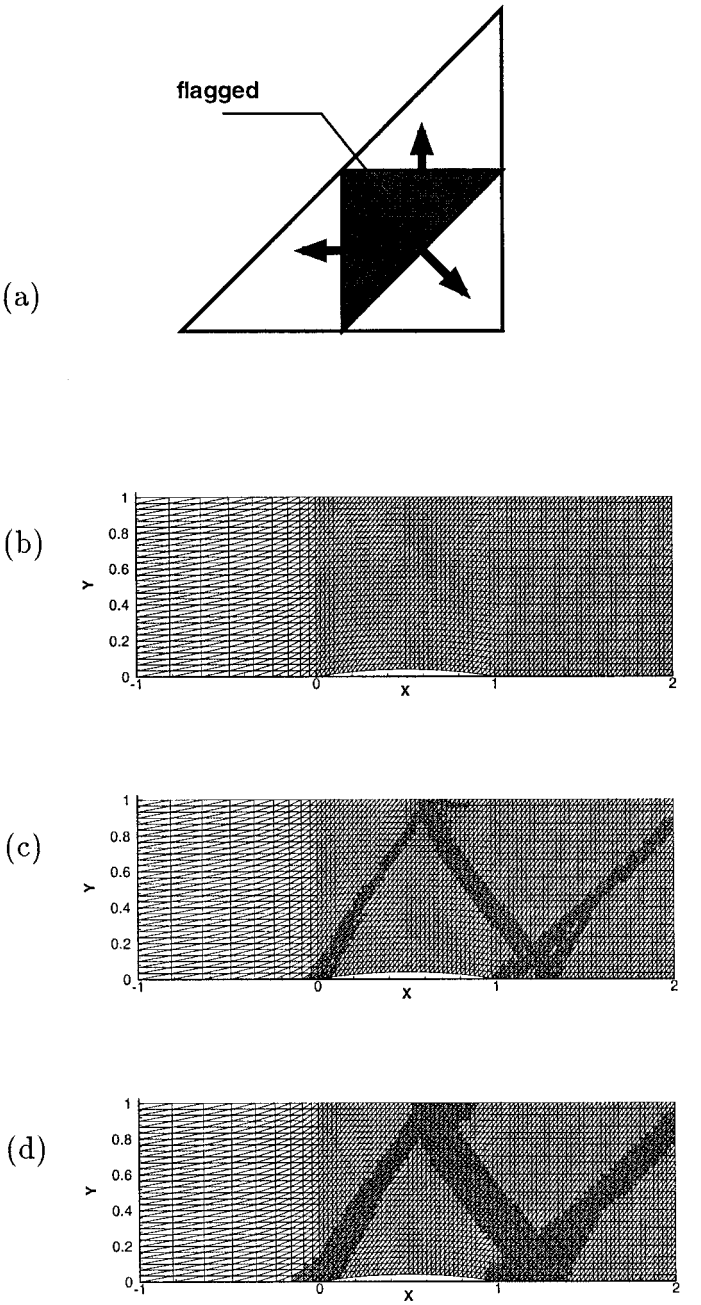

Figure 5. Isotropic advancing front method: (a) a schematic representation; (b) initial mesh; (c) ratio = 0.05; (d) ratio = 0.05 together with repeating the isotropic advancing front three times.

$$\begin{aligned}
&\phi = 1 - \tanh(10) && \text{for } x = 1, \quad 0 \le y \le 1 \\
&\phi = 1 - \tanh(10) && \text{for } x = -1, \quad 0 \le y \le 1 \\
&\phi = 1 - \tanh(10) && \text{for } y = 1, \quad -1 \le x \le 1 \\
&\phi = 1 - \tanh[10(2x+1)] && \text{for } y = 0, \quad -1 \le x \le 0 \\
&\frac{\partial \phi}{\partial y} = 0 && \text{for } y = 0, \quad 0 \le x \le 1
\end{aligned} \tag{44}$$

The contour plots of ϕ at $P_e = 10^6$ and its corresponding profiles at $y = 0$, obtained with both the first-order (UDS) and the second-order (LUDS) upwind schemes on a mesh of 4288 cells, are shown in Figure 6. As seen, the UDS scheme clearly returns unacceptable erosion of the scalar gradients due to a high level of second-order diffusion provoked by the leading truncation error of the scheme. With the same UDS scheme, the unstructured gird solution with a two-level mesh refinement is better than its structured grid counterpart without mesh refinement. However, a good agreement between structured grid and unstructured grid solutions was achieved when the LUDS scheme was both adopted.

3.3. Supersonic inviscid flow over a circular bump

The second case examined is a supersonic flow at $M_\infty = 1.4$ over a 4 per cent circular bump. The mesh in Figure 7 contains 7224 cells, which is obtained with the following sensor function:

$$\text{func}(\phi)_{\phi = p} = \sum_{m = 1,2,3} |p_m - p_0| \tag{45}$$

together with the parameter 'ratio' in Equation (33) set to 0.05. The refined pattern is subsequently enlarged by repeating the isotropic advancing front technique three times to ensure a smooth transition in the vicinity of shock waves. The predicted pressure contours in Figure 7 are compared with the structured grid solution obtained with a mesh of 128×64 nodes. As seen both results are fairly close to each other and give a credible representation of shock-reflection pattern: the formation of a strong oblique shock close to the outlet as a result of the coalescence of the second reflection and the trailing edge shock, which is far better than Karki and Patankar's solution [10].

3.4. Laminar flow in a square cavity

The last case investigated is a laminar lid-driven cavity flow at $Re = 400$. In order to capture the corner eddies, the sensor function is designed on the basis of velocity magnitude as follows:

$$\text{func}(\phi)_{\phi = \sqrt{u^2 + v^2}} = 1 - \sqrt{u^2 + v^2} \tag{46}$$

With 'ratio' set to 0.75 followed by repeating the isotropic advancing front technique twice, the resulting mesh, containing 13 592 cells, is shown in Figure 8. Comparisons of velocity profiles along the vertical and horizontal centrelines of the cavity with those obtained with a structured

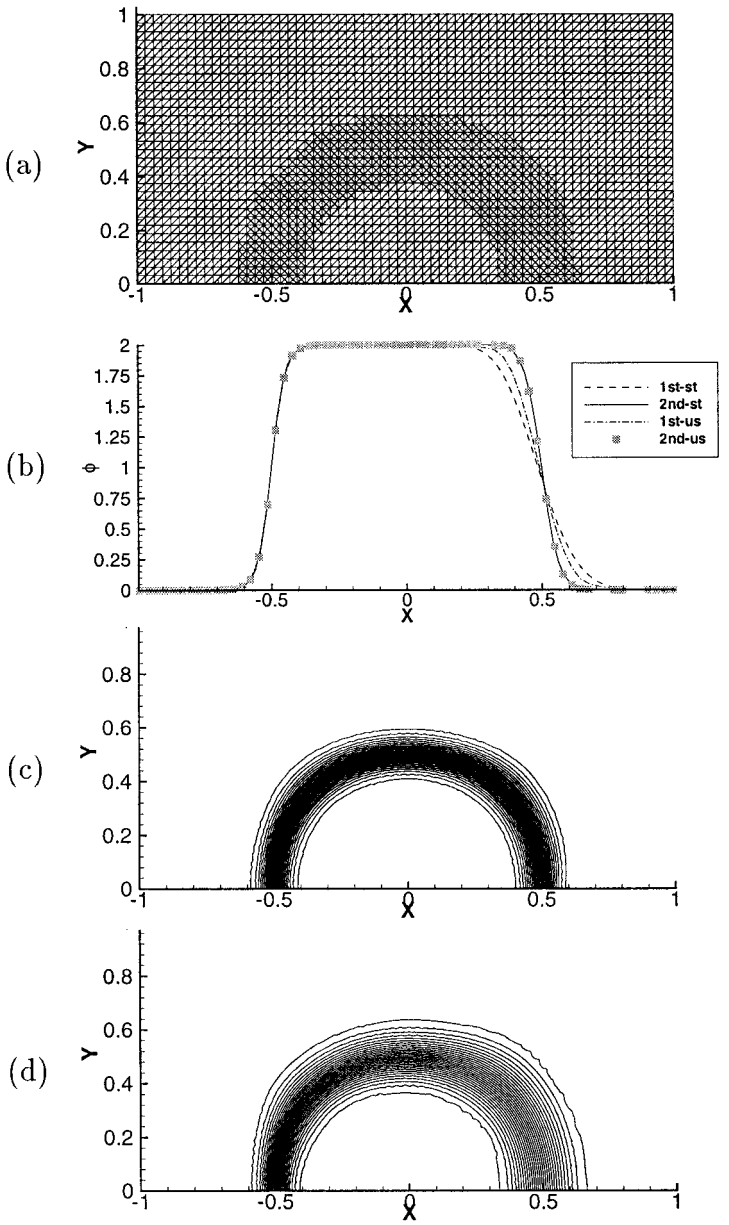

Figure 6. Scalar transport problem: (a) computational mesh; (b) profiles of scalar property at $y = 0$; (c) contour plots obtained with LUDS; (d) contour plots obtained with UDS.

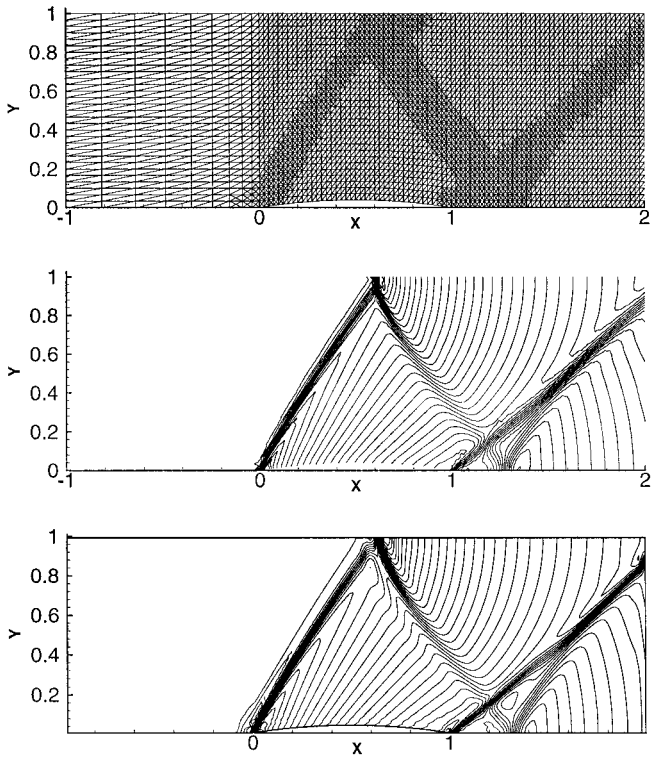

Figure 7. Supersonic bump flow: (top) computational mesh; (middle) unstructured grid solution; (bottom) structured grid solution.

grid method, the latter performed on a mesh of 128×128 nodes, are shown in Figure 8. As demonstrated in this figure, the unstructured grid solutions agree well with the structured grid ones. More importantly, the resulting smooth pressure contours in the same figure suggest that checkerboard oscillations have been eliminated by the present pressure-weighted interpolation scheme for face velocities similar to that of Rhie and Chow [5].

4. CONCLUSIONS AND FUTURE WORK

This paper addresses the details of an effort in which an all-speed numerical algorithm based on the pressure-correction scheme SIMPLE has been implemented into an unstructured grid finite volume code for both incompressible and compressible flows, the latter involving shock waves. Although the study is not exhaustive, the test calculations permit the following conclusions to be drawn:

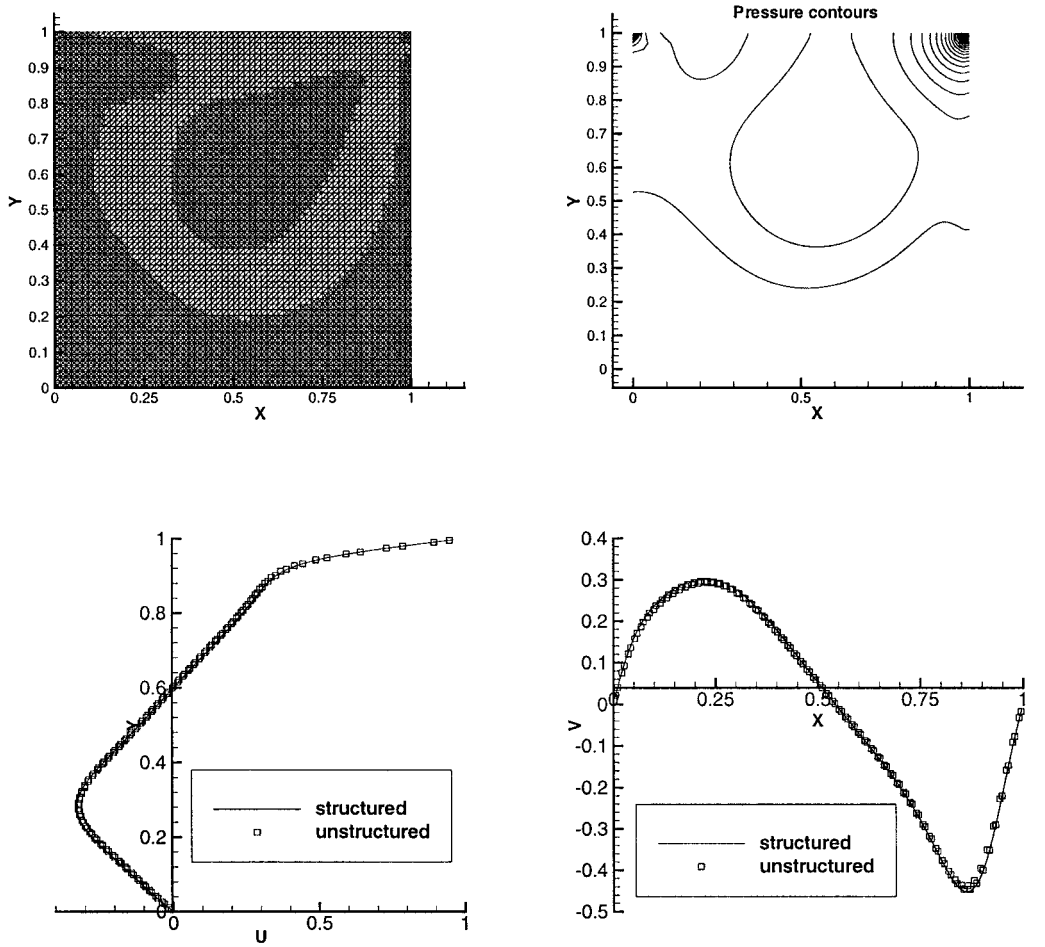

Figure 8. Lid-driven cavity flow: (top-left) computational mesh; (top-right) pressure contours; (bottom-left) u velocity profile at $x = 0.5$; (bottom-right) v velocity profile at $y = 0.5$.

- The solution accuracy can be achieved economically by combining a higher-order convection scheme with AMR strategy, the sensor function of which is based on the gradient or magnitude of flow properties.
- An all-speed pressure-correction scheme, based on the 'density retardation' approach, has been implemented successfully in the present unstructured grid method.
- The CPU overhead of using the second-order LUDS scheme relative to the first-order UDS is about 20 per cent.
- The checkerboard oscillations have been avoided by relating the face velocities to two adjacent pressure nodes, which is similar to the Rhie–Chow interpolation for the structured grid method.

- Extension of the present unstructured grid scheme to turbulent flow calculations, involving shock–boundary layer interaction, will be reported in future accounts.

REFERENCES

1. Lien FS, Chen WL, Leschziner MA. A multi-block implementation of a non-orthogonal, collocated finite volume algorithm for complex turbulent flows. *International Journal for Numerical Methods in Fluids* 1996; **23**: 567.
2. Patankar SV. *Numerical Heat Transfer and Fluid Flow*. Hemisphere: New York, 1980.
3. Rhie CM, Chow WL. Numerical study of the turbulent flow past an airfoil with trailing edge separation. *AIAA Journal* 1983; **21**: 1525.
4. Chorin AJ. A numerical method for solving incompressible viscous flow problems. *Journal of Computers in Physics* 1967; **2**: 12.
5. Hafez M, South J, Murman E. Artificial compressibility method for numerical solutions of transonic full potential equation. *AIAA Journal* 1970; **17**: 838.
6. Lien FS, Leschziner MA. A pressure velocity solution strategy for compressible flow and its application to shock/boundary layer interaction using second moment turbulence closure. *ASME Journal of Fluid Engineering* 1993; **115**: 717.
7. Rivara MC. Selective refinement/derefinement algorithms for sequences of nested triangulations. *International Journal for Numerical Methods in Engineering* 1989; **28**: 2880.
8. Chen WL, Lien FS, Leschziner MA. Local mesh refinement within a multi-block structured grid scheme for general flows. *Computer Methods in Applied Mechanics and Engineering* 1997; **144**: 327.
9. Ferziger JH, Peric M. *Computational Methods for Fluid Dynamics*. Springer: Berlin, 1997.
10. Karki KC, Patankar SV. Pressure based calculation procedure for viscous flows at all speeds in arbitrary configurations. *AIAA Journal* 1989; **27**: 1167.
11. Smith RM, Hutton AG. The numerical treatment of advection a performance comparison of current methods. *Numerical Heat Transfer* 1982; **5**: 439.
12. Ni RH. Multiple grid scheme for solving the Euler equation. *AIAA Journal* 1982; **5**: 1565.
13. Lien FS, Leschziner MA. A general non orthogonal collocated FV algorithm for turbulent flow at all speeds incorporating, second moment closure, Part 1. Computational implementation. *Computer Methods in Applied Mechanics and Engineering* 1994; **114**: 123.

