Development of a Flow Analysis Code Using an Unstructured Grid with the Cell-Centered Method

Hyon Kook Myong*
School of Mechanical and Automotive Engineering, Kookmin University, Seoul 136-702, Korea
Jongtae Kim
Department of Thermal Hydraulics and Safety Research, Korea Atomic Energy Research Institute, Daejon 305-353, Korea

A conservative finite-volume numerical method for unstructured grids with the cell-centered method has been developed for computing flow and heat transfer by combining the attractive features of the existing pressure-based procedures with the advances made in unstructured grid techniques. This method uses an integral form of governing equations for arbitrary convex polyhedra. Care is taken in the discretization and solution procedure to avoid formulations that are cell-shape-specific. A collocated variable arrangement formulation is developed, i.e. all dependent variables such as pressure and velocity are stored at cell centers. For both convective and diffusive fluxes the forms superior to both accuracy and stability are particularly adopted and formulated through a systematic study on the existing approximation ones. Gradients required for the evaluation of diffusion fluxes and for second-order-accurate convective operators are computed by using a linear reconstruction based on the divergence theorem. Momentum interpolation is used to prevent the pressure checkerboarding and a segregated solution strategy is adopted to minimize the storage requirements with the pressure-velocity coupling by the SIMPLE algorithm. An algebraic solver using iterative preconditioned conjugate gradient method is used for the solution of linearized equations. The flow analysis code (PowerCFD) developed by the present method is evaluated for its application to several 2-D structured-mesh benchmark problems using a variety of unstructured quadrilateral and triangular meshes. The present flow analysis code by using unstructured grids with the cell-centered method clearly demonstrate the same accuracy and robustness as that for a typical structured mesh.

Key Words : Flow Analysis Code, Unstructured Grid System, Cell-Centered Method, Finite Volume Method, Benchmark Solution, Performance Evaluation

1. Introduction

Unstructured grid methods have become increasingly popular in the computational fluid dynamics (CFD) research field over the last two decades, since these methods have two main advantages : the ease of grid generation for complex, realistic geometries, and the ability to dynamically adapt the grid to local features of interest. CFD codes for unstructured grids are thus more flexible. They need not be changed when the grid is locally refined, or when control volumes (CVs) of different shapes are used.

In addition to geometric flexibilities, a stable

* Corresponding Author,
E-mail : myong@kookmin.ac.kr
TEL : +82-2-910-4714; **FAX** : +82-2-910-4839
School of Mechanical and Automotive Engineering, Kookmin University, Seoul 136-702, Korea. (Manuscript **Received** February 15, 2006; **Revised** August 30, 2006)

implementation of physical models is equally important when developing a CFD code. In this aspect, pressure-based finite volume methods with a primitive variable approach have been the most popular ones for incompressible flows over the last three decades. The combined use of a finite volume discretization with a segregated solution strategy and pressure-velocity coupling by means of a pressure correction equation make algorithms such as SIMPLE (Patankar, 1980) and its extensions among the most efficient approaches for incompressible flows. However, an extension of this well-known methodology to unstructured grids has been hampered by two main difficulties. Staggered storage of the velocity and pressure components, necessary to avoid a checkerboarding, is not straightforward. Also, gradient determination becomes complicated by the absence of a line structure. This leads to difficulties in discretization of the diffusion fluxes and the development of higher-order schemes.

Recently, several unstructured cell-centered finite volume approaches have begun to appear (Jiang and Przekwas., 1994 ; Demirdzic and Muzaferija, 1995 ; Mathur and Murthy, 1997 ; Davison, 1996 ; Lien, 2000), which are more closely related to traditional methods for structured body-fitted meshes (Peric, 1985). Here, conservation is enforced on the basic cell itself, and all the transport variables are stored at cell centers. This arrangement is preferred over a node-based storage for several reasons. With the cell-based storage, conservation can be ensured for arbitrary control volumes with nonconforming interfaces without special interpolation techniques, and singularity points do not exist since the boundaries of calculation domain are composed of surfaces (or line) of control volume. In addition, since the gradient determination does not employ element-specific shape functions, these methods have the potential for a use with arbitrary polyhedral meshes.

In the present paper, we develop a pressure-based finite volume scheme for unstructured meshes similar in philosophy to the previous ones (Jiang and Przekwas., 1994 ; Demirdzic and Muzaferija, 1995 ; Mathur and Murthy, 1997 ; Davison, 1996 ; Lien, 2000). This method admits arbitrary convex polyhedra. Cell-centered, collocated storage is preferred without the use of an element-specific shape function. For both convective and diffusive fluxes the forms superior to both accuracy and stability are adopted and formulated through a systematic study on the existing approximation forms. Higher-order convective fluxes as well as secondary diffusion terms are computed by using a linear reconstruction. To minimize the storage requirements, a segregated solution strategy is adopted with the pressure-velocity coupling by the SIMPLE algorithm. An algebraic solver using iterative preconditioned conjugate gradient method is used for the solution of linearized equations. The present method thus combines the attractive features of the pressure-based procedures with the advances made in other unstructured grid techniques.

The next two sections present the governing equations and the details of the numerical method, respectively. For the sake of clarity, only the steady, laminar, and incompressible flows of a constant property fluid are considered, although the method is easily extended to unsteady, compressible, and turbulent regimes.

2. Numerical Methods

2.1 Discretization of a scalar transport equation

Consider the differential equation for the transport of a scalar quantity ϕ. All the governing equations can be cast into the following integral form with the appropriate choice of ϕ, Γ and S_ϕ:

$$\int_S [\rho\phi\underline{v} - \Gamma_\phi \nabla\phi] \cdot \underline{dS} = \int_V Q_\phi dV \qquad (1)$$

In order to obtain discrete counterparts of Eq. (1), the solution domain is discretized into a finite number of contiguous arbitrary unstructured convex polyhedra called control volume (CV) or cell. The computational nodes (points) are placed at the center of each CV, while boundary nodes, needed for the specification of boundary conditions, reside at the center of the boundary cell-faces. The control volume is defined by the coordinates of its vertices and can be of an arbi-

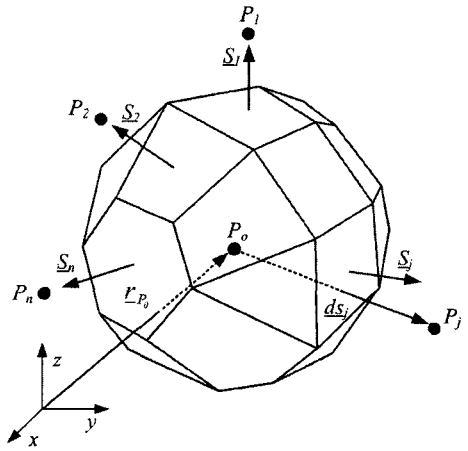


Fig. 1 Control volume of an arbitrary polyhedral shape

trary polyhedral shape, i.e. it can have an arbitrary number of cell-faces. Thus, a discretization about the CV P_0 having volume V_{P_0} and divided faces S_j $(j=1,N_f)$ as shown in Fig. 1 yields

$$\sum_j [J_j\phi_j^* - D_j] = \sum_j [C_j - D_j] = (Q_{\phi V})_{P_0} V_{P_0} \quad (2)$$

where J_j is the mass flow rate (defined to be positive if flow is leaving cell P_0), C_j and D_j are the transports due to convection and diffusion through the face j, respectively, and the summations are over the faces of the control volume.

2.2 Convection term

The mass flow rate J_j is in general assumed to be known from the solution of the momentum and continuity equations (see the following section). The task of an evaluation of the convective flux thus reduces to determining the face value, ϕ_j^*. A simple and stable method is the following first-order approximation using the value at the upwind cell.

$$\phi_j^* = \phi_j^{FO} = \phi_{upwind} \quad (3)$$

Although a higher-order accuracy is achieved by employing a linear interpolation or a bigger stencil along the appropriate grid line in structured grid methods, this is not possible in the case of unstructured grids. Attempts have been made which mimic the bigger stencil by creating local line structures and interpolating the values from neighboring cells (Jiang and Przekwas., 1994). These can be quite expensive and are also cell-shape-specific. However, if a shape-independent formulation for the cell gradients can be devised, a second-order value at the face can be obtained. At present, several forms with the second-order accuracy are proposed for the face value, ϕ_j^* (Demirdzic and Muzaferija, 1995; Mathur and Murthy, 1997; Muzaferija, 1994; Ferziger and Peric, 2002). Recently, one of the authors (Myong, 2006a) has studied systematically on the existing approximation forms for the second-order accurate face value used in unstructured cell-centered finite volume method, and found that the form proposed by Demirdzic and Muzaferija (1995) has more accurate prediction performance than the other ones. Thus, in the present study we adopts this form for ϕ_j^*, which is expressed as follows:

$$\phi_j^* = \phi_j^{SO} = \frac{1}{2}[(\phi_{P_0} + \nabla\phi_{P_0} \cdot \underline{dr}_{P_0}) + (\phi_{P_j} + \nabla\phi_{P_j} \cdot \underline{dr}_{P_j})] \quad (4)$$

where $\nabla\phi_{P_0}$ and $\nabla\phi_{P_j}$ are the gradients at cells P_0 and P_j, respectively (see the section below). Here, cell P_j is the neighboring cell of cell P_0 with which it shares a common face j, and $\underline{dr}_{P_0}$ and $\underline{dr}_{P_j}$ are $\underline{r}_j - \underline{r}_{P_0}$ and $\underline{r}_j - \underline{r}_{P_j}$, respectively, which are the vectors directed from the centroid of either cell holding the face j in common to the cell-face center.

In order to assure a stable procedure, the face value, ϕ_j^* is also usually calculated by blending the second-order accurate formula (4) with some small amount of the first-order value (3)

$$\phi_j^* = \phi_j^{FO} + \gamma_\phi(\phi_j^{SO} - \phi_j^{FO}) \quad (5)$$

where γ_ϕ is the blending factor with a value between zero and one, thus combining the accuracy of a second-order scheme and the stability of a first-order one. It is important to mention that SO scheme is implemented according to the 'deferred-correction' practice, i.e. the first term in Eq. (5) is treated implicitly and the second term is included explicitly.

2.3 Gradient computation

In the unstructured grid, the distribution of a dependent variable ϕ has to be assumed in order to evaluate the values of its derivatives at

both the cell center and the cell face, which cannot usually be written in terms of cell values by using the one-dimensional Taylor series expansions. Here, the following linear spatial di-tribution is employed.

$$\phi(\underline{r}) = \phi_{P_0} + \nabla\phi_{P_0} \cdot (\underline{r} - \underline{r}_{P_0}) \tag{6}$$

Interpolation methods such as least squares (Demirdzic and Muzaferija, 1995) can be used to calculate the gradient of ϕ at point P_0. However, these techniques can be quite expensive and are cell-shape-specific. In the present study, we use the divergence (or Gauss) theorem providing a technique that is independent of the cell shape : the gradient for cell P_0 is estimated as

$$\begin{aligned}(\nabla\phi)_{P_0} &= \left[\int_{V_{P_0}} (\nabla\phi)_{P_0} dV\right] / V_{P_0} \\ &= \sum_j (\bar{\phi} \cdot \underline{S})_j / V_{P_0}\end{aligned} \tag{7}$$

where the summation is over all the faces of the cell. The face value of ϕ, $\bar{\phi}$ is obtained by a weighted averaging of the values at the neighboring cells (Demirdzic and Muzaferija, 1995). Note that Mathur and Murthy (1997) used an arithmetic averaging for it.

2.4 Diffusion term

The diffusion term at the face is

$$D_j = \Gamma_{\phi_j} (\nabla\phi)_j^* \cdot \underline{S}_j \tag{8}$$

Here, we would like to express D_j in terms of the values of ϕ at two neighboring cells P_0 and P_j. The usual method by a transformation from physical coordinates to computational coordinates, which is typical in the case of a structured grid system, requires the use of face tangents and nodes. In a structured grid formulation the face node values are typically found by averaging the values at the cell centers and the face tangent gradient component is treated explicitly. Similar treatment is easily possible in two dimensions for unstructured meshes. However, extending this methodology to three dimensional polyhedra of an arbitrary shape is not straightforward, since there are no unique face tangential directions and nodes that can be employed to write an equivalent form in three dimensions. Another method to avoid the use of face tangents and nodes is to use $d\phi$ $(=\phi_{P_j} - \phi_{P_0})$ in the direction of $\underline{ds}_j$ $(=\underline{r}_{P_j} - \underline{r}_{P_0})$ as a primary diffusion term and to use the difference between the total diffusion and primary component as secondary (or cross) diffusion term (Demirdzic and Muzaferija, 1995; Mathur and Murthy, 1997; Muzaferija, 1994; Ferziger and Peric, 2002), which is generally expressed as follows :

$$\begin{aligned}D_j = \Gamma_{\phi_j} [& f_j (\phi_{P_j} - \phi_{P_0}) \\ & + (\overline{(\nabla\phi)}_j \cdot \underline{S}_j - f_j \overline{(\nabla\phi)}_j \cdot \underline{ds}_j)\,]\end{aligned} \tag{9}$$

where the first term in the brackets represents the primary diffusion and the second term within ( ) is the cross-diffusion. Terms f_j are geometrical factors. Several forms are proposed for both f_j and the gradient of ϕ at the face, $\overline{(\nabla\phi)}_j$ (Demirdzic and Muzaferija, 1995; Mathur and Murthy, 1997; Davison, 1996; Lien, 2000). Recently, one of the authors (Myong, 2006b) has clarified the existing approximation forms of diffusion term to have indefinite expressions in several respects through a systematic study, and has proposed a new diffusion term with the second-order accuracy even on irregular grid, similar in form to Eq. (9), which is expressed as follows :

$$\begin{aligned}D_j = {} & (\phi_{P_j} - \phi_{P_0}) \frac{\Gamma_{\phi_j} S_j}{\underline{ds}_j \cdot \hat{n}} + \Gamma_{\phi_j} [(1-\omega_j)(\nabla\phi)_{P_0} + \omega_j (\nabla\phi)_{P_j}] \cdot \underline{S}_j \\ & - \Gamma_{\phi_j} [(\nabla\phi)_{P_0} \cdot (\underline{r}_j - \underline{r}_0^*) + (\nabla\phi)_{P_j} \cdot (\underline{r}_{P_j} - \underline{r}_j)] \frac{S_j}{\underline{ds}_j \cdot \hat{n}}\end{aligned} \tag{10}$$

where $\hat{n}$ is the outward unit normal vector to the cell face and the interpolation factor is expressed as follows :

$$\omega_j = |(\underline{r}_j - \underline{r}_{P_j})| / |(\underline{r}_{P_j} - \underline{r}_{P_0})| \tag{11}$$

where $\underline{r}_j$ is the position vector at the intersection of the cell face and the straight line connecting cell points P_0 and P_j. Note here that this line may not pass through the cell face center. As discussed in the above section, Gauss theorem is used to determine the cell derivative of ϕ, $\nabla\phi$ in a manner independent of the cell shape. The formulation above is applicable without any modification to the three-dimensional configurations. It is noted that, for stability the primary term in Eq. (10) is included implicitly and the remaining terms explicitly.

Here, it is valuable to briefly comment on a new approximation form for diffusion term, Eq. (10). First, it points out that the proper form for the geometrical factors, f_j in Eq. (9) is as follows:

$$f_j=\frac{S_j}{\underline{ds_j}\cdot\hat{n}} \tag{12}$$

which is superior to the numerical stability aspect when the straight line connecting cell points P_0 and P_j is non-orthogonal to the cell face. Second, when the straight line connecting cell points P_0 and P_j passes through the cell face center, its form coincides that of Eq. (9), but the gradient of ϕ at the face, $\overline{(\nabla\phi)}_j$, which is usually taken to be the interpolated value of the derivatives at the two adjacent cells, should be calculated as follows:

$$\overline{(\nabla\phi)}_j=(1-\omega_j)(\nabla\phi)_{P_0}+\omega_j(\nabla\phi)_{P_j} \tag{13}$$

which is a new form different from the existing ones. Third, Eq. (10) is directly applicable to the calculation of the boundary diffusion flux (see the following section), while most of the existing ones have logically problems. The more detailed analysis between Eq. (10) and existing approximation forms of Eq. (9) is referred to Myong (2006b).

2.5 Boundary conditions

In addition to cell centers, ϕ is also stored at the boundary face centers. The boundary diffusion flux can then be linearized in the same manner as an interior face, i.e., using Eq. (10). For the boundary face, this yields

$$\begin{aligned}D_b=&(\phi_{P_b}-\phi_{P_0})\frac{\Gamma_{\phi_b}\underline{S_b}}{\underline{ds_b}}\cdot\hat{n}\\&+\Gamma_{\phi_b}[(\nabla\phi)_{P_0}\cdot\underline{S_b}+((\nabla\phi)_{P_0}\cdot\underline{ds_b})]\frac{\underline{S_b}}{\underline{ds_b}}\cdot\hat{n}\end{aligned} \tag{14}$$

where ϕ_{P_b} is the value at the boundary and $\underline{ds_b}$ is the vector from the cell centroid to the boundary face centroid.

In the case of the Dirichlet boundary conditions, the primary term is included implicitly and the secondary term explicitly. For the Neumann boundary conditions the specified flux can be added directly to the control volume balance; Eq. (14) is employed to compute the boundary value, ϕ_{P_b}, for a postprocessing. Other boundary specifications, for example, convection and/or radiation conditions at the wall for the energy equation, are easily accommodated within this framework.

2.6 Discretized equations

The above discretization procedure yields the following linear system of equations which links the value of the dependent variable ϕ at the CV center with those values at the centers of the neighboring cells:

$$A_{P_0}\phi_{P_0}+\sum_j^{nnb}A_{P_j}\phi_{P_j}=b_\phi \tag{15}$$

Here the summation is over all the neighbors nnb of cell P_0. The source term b_ϕ contains any of the volumetric sources of ϕ, the second-order contributions for the convective flux, as well as the secondary diffusion fluxes. Flux contributions at boundaries are also included in b_ϕ. Equation (15) is underrelaxed in the manner described in the reference (Patankar, 1980).

2.7 Linear solver

The resulting set of coupled nonlinear algebraic equations is solved by employing a segregated approach, leading to a decoupled set of linear algebraic equations for each dependent variable, with a sparse diagonally dominant coefficient matrix. These equations are solved by an iterative preconditioned conjugate gradient solver such as the CGSTAB (CG STABilized) method with an incomplete Cholesky preconditioning which retains the sparsity of the coefficient matrix, thus achieving a very efficient use of computer resources.

There is no need to solve Eq. (15) to a tight tolerance since the coefficients and sources are only approximated (based on the values of dependent variables from the previous iteration) and a reduction of the sum of absolute residuals for one order of magnitude normally suffices.

2.8 Discretization of the momentum equation

In the momentum equation, the diffusive flux contains a few more terms than does the corre-

sponding term in the generic conservation equation, e.g. for u_i:

$$D_j^i=\int_{S_j}\mu\nabla u_i\cdot\underline{dS_j}+\underline{\int_{S_j}\mu\frac{\partial u_j}{\partial x_i}\hat{n}_j\cdot dS_j} \quad (16)$$

The underlined term is absent in the generic conservation equation. If ρ and μ are constant, the sum of the underlined terms over all the CV faces is zero by virtue of the continuity equation. If ρ and μ are not constant, they vary smoothly except near shocks and the integral of the underlined terms over the whole CV surface is smaller than the integral of the principal term. For this reason, the underlined term is usually treated explicitly. As shown above, the derivatives are also easily calculated at the cell face by using the derivatives at the CV center. By the way, when the first term is calculated by Eq. (9) or Eq. (10), the underlined term is an additional term treated explicitly. In the present paper, thus, we calculate the diffusive flux directly from the stress tensor as follows.

$$D_j^i=\int_{S_j}\underline{\tau_j^i}\cdot\underline{dS_j}\approx\underline{\tau_j^i}\cdot\underline{S_j} \quad (17)$$

The final form of the diffusive flux in the momentum equation is as follows:

$$D_j^i=(u_{P_j}^i-u_{P_0}^i)\frac{\mu_j S_j}{\underline{ds_j}\cdot\hat{n}}+\left[\underline{\tau_j^i}\cdot\underline{S_j}-\mu_j(\overline{(\nabla u_i)_j}\cdot\underline{ds_j})\frac{S_j}{\underline{ds_j}\cdot\hat{n}}\right] \quad (18)$$

And the source term in the momentum equation contains the discretized pressure gradient term. Irrespective of how this term is approximated, one can write:

$$Q_{u,V}=Q_{u,V}^*+Q_{u,V}^p=Q_{u,V}^*-(\nabla p)_{P_0}V_{P_0} \quad (19)$$

2.9 Discretization of the continuity equation

The discrete continuity equation is written as

$$\sum_j J_j=\sum_j\rho\underline{v_j^*}\cdot\underline{S_j}=0 \quad (20)$$

Since the pressure and velocity components are stored at the cell centers, computing the face mass flow rate by averaging the cell velocity is prone to a checkerboarding. To avoid this, a scheme similar to that of Rhie and Chow (1983) is used. For the face j, the fluid velocity is written as

$$\underline{v_j^*}=\underline{v_j}+\delta\underline{v_j} \quad (21)$$

where the first term is the spatially interpolated velocity defined as follows:

$$\underline{v_j}=(1-w_j)\,\underline{v_{P_0}}+w_j\underline{v_{Pj}} \quad (22)$$

while the second term is a third-order pressure diffusion term, defined as

$$\delta\underline{v_j}=D_p\hat{n}=-\overline{\left(\frac{V_0}{A_{v0}}\right)_j}\left[\frac{p_{P_j}-p_{P_0}}{\underline{ds_j}\cdot\hat{n}}-\frac{\overline{\nabla p}_j\cdot\underline{ds_j}}{\underline{ds_j}\cdot\hat{n}}\right]\frac{\underline{S_j}}{|\underline{S_j}|} \quad (23)$$

where A_{v0} is the coefficient A_{P_0} in the momentum equation and $\overline{\nabla p}_j$ is the interpolated pressure gradient at cell face j. This term smooths out oscillatory the pressure velocity profile, and at the same time introduces a pressure into the continuity equation in a manner that a pressure-correction equation can be easily constructed (only the explicit pressure difference across the face in Eq. (23) is treated implicitly), and the predictor-corrector procedure defined by the SIMPLE algorithm can be established.

2.10 Pressure-correction equation

The SIMPLE algorithm (Patankar, 1980) is used for a pressure-velocity coupling. Accordingly, we require that

$$\sum_j J_j=\sum_j\rho_j(\underline{v_j^*}+\underline{v_j'})\cdot\underline{S_j}=\sum_j J_j^*+J_j'=0 \quad (24)$$

where J_j^* is the flow rate computed from the velocities satisfying the discrete momentum equations. We define the velocity correction $\underline{v_j'}$ as

$$\underline{v_j'}\approx-\overline{\left(\frac{V_0}{A_{v0}}\right)_j}\frac{p'_{P_j}-p'_{P_0}}{|\underline{ds_j}|}\cdot\frac{\underline{S_j}}{|\underline{S_j}|} \quad (25)$$

Thus,

$$J_j'\approx-\rho_j\overline{\left(\frac{V_0}{A_{v0}}\right)_j}\frac{|\underline{S_j}|}{|\underline{ds_j}|}(p'_{P_j}-p'_{P_0}) \quad (26)$$

Substituting Eq. (26) into Eq. (24) yields the pressure correction equation:

$$A_{P_0}p'_{P_0}+\sum_j^{nnb}A_{P_j}p'_{P_j}=b_{P_0} \quad (27)$$

where

$$A_{P_0}=-\sum_j^{nnb}A_{P_j} \quad (28)$$

The term b_{P_0} is the net mass inflow into the cell,

and the summation is over all the cells neighboring cell P_0.

Once the corrections p' are available, the cell pressure, cell velocity and face mass flow rate are corrected by using,

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

$$\underline{v}=\underline{v}^*+\underline{v}'=\underline{v}^*-\frac{V_{P_0}}{A_{P_0}}(\nabla p')_{P_0} \tag{30}$$

$$J_j=J_j^*+J_j'=J_j^*+A_{P_j}p'_{P_j}-A_{P_j}p'_{P_0} \tag{31}$$

Here, p^* and $\underline{v}^*$ are the values prevailing after the solution of the momentum equations, α_p is the underrelaxation factor for the pressure. The face pressure correction p'_j is computed by a weighted averaging of the p' values of the cells neighboring face j.

3. Results

In this section, we apply the flow analysis code (PowerCFD) using the method developed above to a number of benchmark problems in the literature. These tests seek to establish the accuracy and convergence characteristics of the method, and to demonstrate that a variety of mesh topology can be used. In particular, care is taken to use truly unstructured meshes when making comparisons with the structured-mesh benchmarks. This is critical because unstructured meshes created, for example, by triangulating an underlying structured mesh are not good indicators of a performance for realistic industrial applications.

The solution is in general considered converged when all scaled residuals fall below 10^{-3}. For the differential equation for the transport of a scalar quantity ϕ, the scaled residual is defined as

$$R=\frac{\sum_{cells}\left|A_{P_0}\phi_{P_0}-\sum_{j}^{nnb}A_{P_j}\phi_{P_j}-b_\phi\right|}{M_\phi} \tag{32}$$

where M_ϕ is a normalization factor and the inflow rate of ϕ is usually used as the default value. In case of no inflow/outflow boundaries like the below benchmark problems, however, the residual for each variable is normalized by its maximum value in the first 10 iterations and the solution is considered converged when all scaled residuals fall below 10^{-7}.

3.1 Flow in a skewed lid-driven cavity

Demirdzic et al.(1992) have published benchmark solutions for a lid-driven flow in a skewed cavity, as shown in Fig. 2. We consider here the case $\theta=30°$, at a Reynolds number $\mathrm{Re}(=UL/\nu)=1{,}000$. This case is challenging for structured quadrilateral meshes because of the extreme skewness of the cavity. We solve the problem by using both unstructured quadrilateral and triangular meshes. The objective of the calculation is (1) to compare the computed results with those of Demirdzic et al.(1992) and to establish an accuracy and convergence to a mesh-independent solution, and (2) to compare the performance of unstructured quadrilateral and unstructured triangular cells.

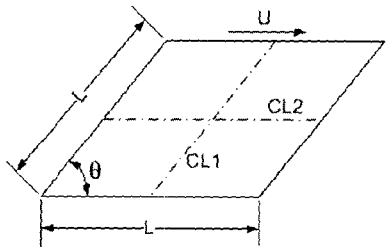
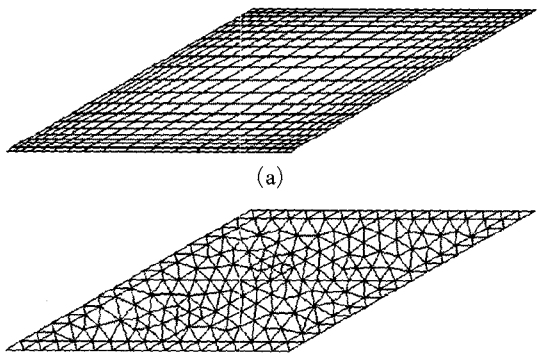


Fig. 2 Schematic diagram of the skewed cavity



(a)

(b)

Fig. 3 Segments of the grid used for the flow in a skewed lid-driven cavity ($\theta=30°$). (a) Segments of the grid used for the flow in a skewed lid-driven cavity ($\theta=30°$). (a) quadrilateral mesh, (b) triangular meshQuadrilateral mesh, (b) Triangular mesh

To establish a convergence of the scheme to a mesh-independent solution, flow in the cavity is computed using an unstructured quadrilateral mesh of 20×20, 40×40, 80×80 and 160×160 cells. The 20×20 mesh is shown in Fig. 3(a). The finer meshes are created by subdividing each cell into four. The velocity on the centerline CL1 is shown in Fig. 4, along with the tabulated benchmark values from Demirdzic et al.(1992) obtained by using 320×320 quadrilateral cells. The y velocity on the centerline CL2 is shown in Fig. 5, along with the benchmark values. We find that the solutions for mesh densities greater than 80×80 cells are indistinguishable from the benchmark and from each other. This rate of convergence to a mesh-independent solution is similar to that of the second-order structured mesh scheme used in Demirdzic et al.(1992).

A similar test of the mesh independence is done by using unstructured triangular meshes as well. We start with a coarse mesh of 396 triangles, shown in Fig. 3(b). This mesh is approximately

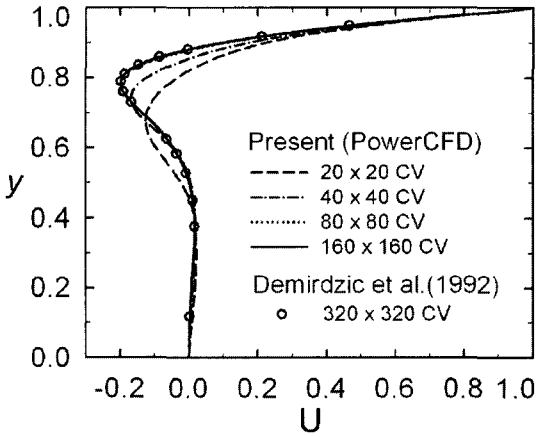


Fig. 4 Variation of the centerline U-velocity profiles on CL1 for the skewed lid-driven cavity flow at Re=1000 as a function of the grid fineness (quadrilateral mesh, θ=30°)

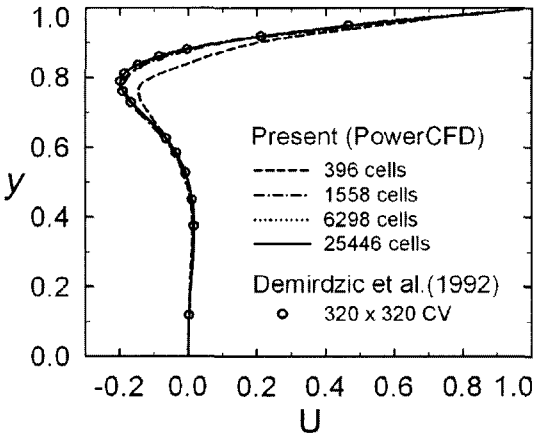


Fig. 6 Variation of the centerline U-velocity profiles on CL1 in a skewed lid-driven cavity flow at Re=1000 as a function of grid fineness (triangular mesh, θ=30°)

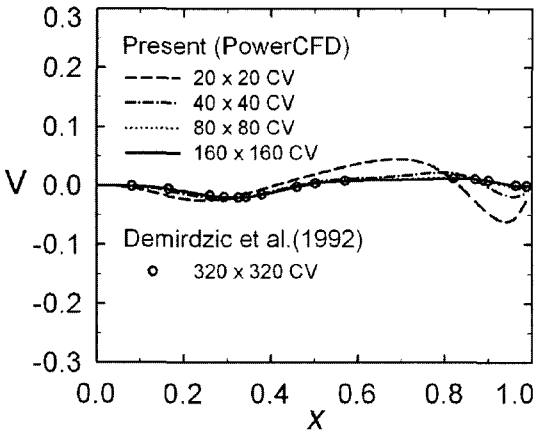


Fig. 5 Variation of the centerline V-velocity profiles on CL2 in a skewed lid-driven cavity flow at Re=1000 as a function of grid fineness (quadrilateral mesh, θ=30°)

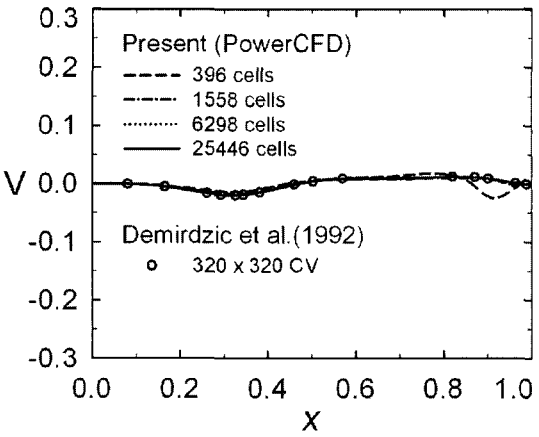


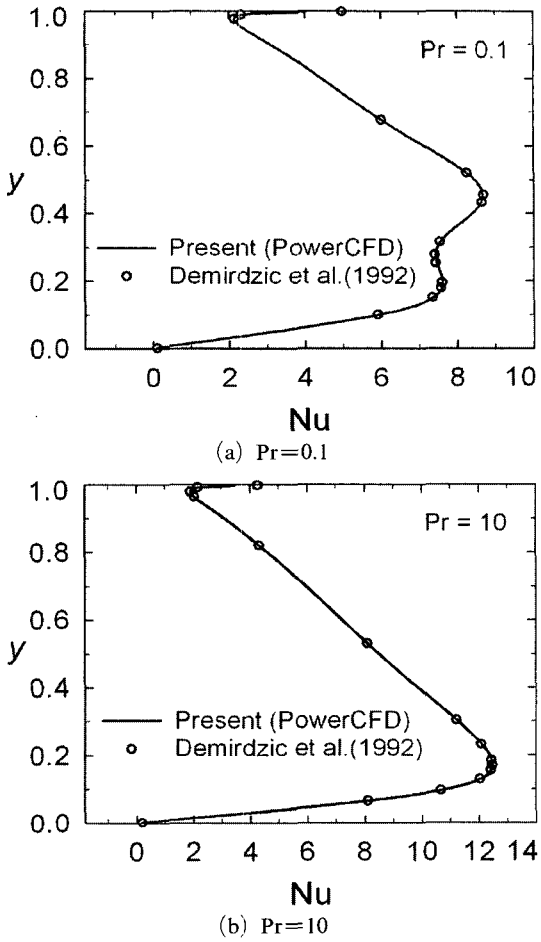
Fig. 7 Variation of the centerline V-velocity profiles on CL2 in a skewed lid-driven cavity flow at Re=1000 as a function of grid fineness (triangular mesh, θ=30°)

equivalent to the 20×20 quadrilateral mesh used above. Computations are done for 1,558, 6,298 and 25,446 cells, respectively. The x velocity on the centerline CL1 is shown in Fig. 6 and the y velocity on the centerline CL2 is shown in Fig. 7. Mesh-independent solutions are obtained for mesh densities greater than 6,298 cells, similar to the quadrilaterals. The rate of convergence to a mesh-independent solution is also similar to that observed for the quadrilaterals.

3.2 Flow and heat transfer in a skewed buoyancy-driven cavity

Demirdzic et al.(1992) have also published benchmark solutions for a buoyancy-driven flow in a skewed cavity, as shown in Fig. 2. The inclined walls are kept at constant temperatures T_H and T_C, respectively, while the horizontal walls are assumed adiabatic. We consider here the case $\theta=45°$ at a Rayleigh number $\mathrm{Ra}(=\rho^2 g\beta L^3 \triangle T \mathrm{Pr}/\mu^2)=10^6$ with Pr=0.1 and 10. Gravity acts in the vertical direction. This case is also challenging for heat transfer calculations using structured quadrilateral meshes, since the flow is driven by a buoyancy as well as the extreme skewness of the cavity, and, in addition to the Navier-Stokes equations, an equation for the temperature has to be solved. The temperature is coupled with the velocity field through the convection terms, and the velocity field is coupled with the temperature through the buoyancy term which acts as a source of momentum. We solve the problem by using an unstructured quadrilateral mesh of 224×192. The Boussinesq approximation is used to model the buoyancy. The objective of the calculation is to compare the computed results with those of Demirdzic et al.(1992). Local Nusselt number distributions on the cold wall at Pr=0.1 and 10 are shown in Figs. 8(a) and (b), respectively, along with the tabulated benchmark values from Demirdzic et al.(1992) obtained by using the 224×192 structured quadrilateral cells. We find that the solutions are indistinguishable from the benchmark for both Prandtl numbers.

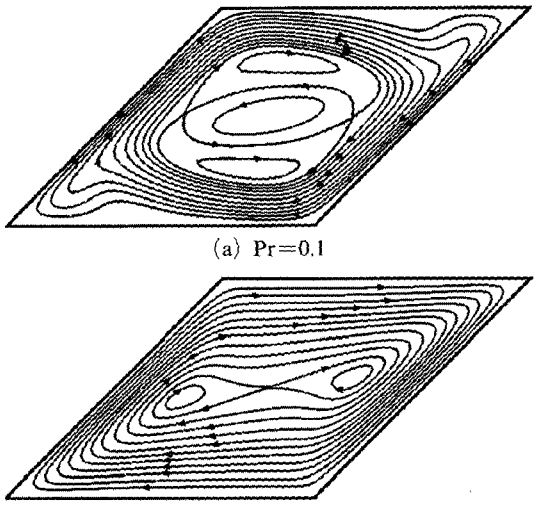
The predicted flow patterns for the two Prandtl numbers are shown in Fig. 9. It is obvious that the Prandtl number has a significant influence on the flow and heat transfer. The results show that for Pr=10, the flow is directed along the walls and it stays attached to them, forming a single roll, while in the case of Pr=0.1, the main current in the recirculating region tends to separate from the horizontal walls. And another big difference is shown in that for Pr=0.1, a large region of slowly moving fluid is formed in the central part of the cavity, including two free stagnation points and a counterrotating eddy in the center, while, in the case of Pr=10 one free stagnation point exists in the cavity center, but there are no counterrotating eddies. The results are very consistent with the figured benchmark ones from Demirdzic et al.(1992).



(a) Pr=0.1

(b) Pr=10

Fig. 8 Local Nusselt numbers on the cold wall for the flow in a buoyancy-driven cavity at Ra=10^6 (224×192 CV)



(a) Pr=0.1

(b) Pr=10

Fig. 9 Predicted streamlines for the flow in a buoyancy-driven cavity at $Ra=10^6$. (224×192 CV)

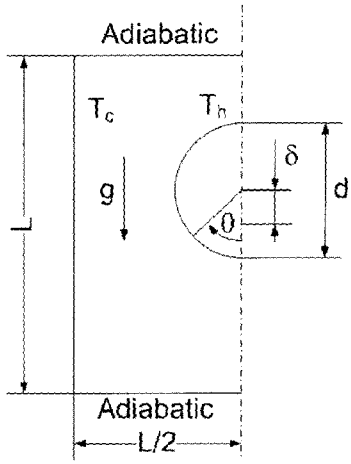


Fig. 10 Schematic diagram of the natural convection in the cavity with a cylinder

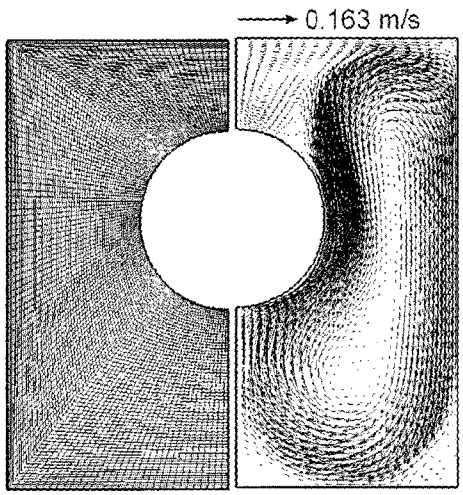


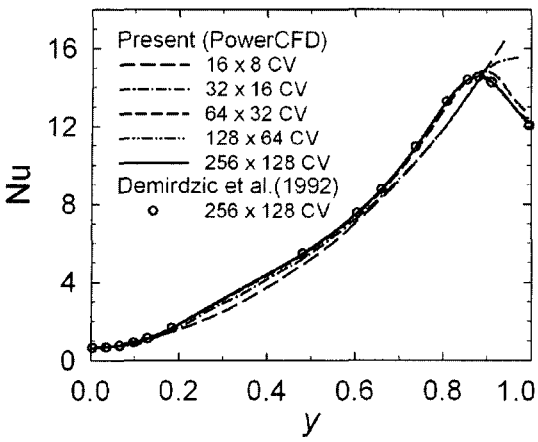
Fig. 11 Left: computational mesh used for the natural convection around a cylinder in a square duct, right: velocity vector plots at Pr=0.1, $Ra=10^6$ (64×32 CV)

3.3 Natural convection in a cavity

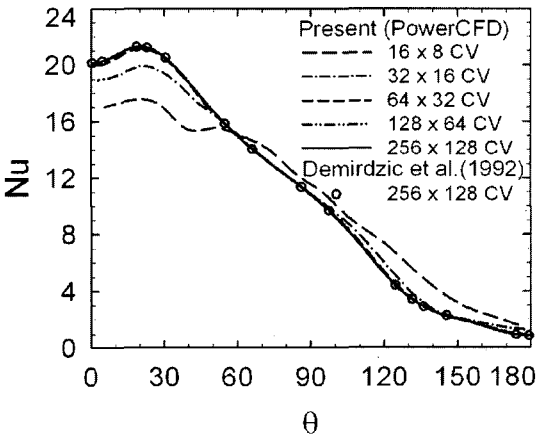
Here we compute another flow and heat transfer due to a heated cylinder in a square cavity, as shown in Fig. 10. The cylinder center is displaced from the center of the cavity by δ. The Rayleigh number based on the cylinder diameter is 10^6, and the Prandtl number is 0.1. $\delta/L=0.1$ and $d/L=0.4$ are used. The Boussinesq approximation is used to model the buoyancy. A benchmark solution for this configuration has been published by Demirdzic et al.(1992) by using a structured mesh of 256×128 cells. The objective of this computation is to compare the computed results with those of Demirdzic et al.(1992) and to establish an accuracy and convergence to a mesh-independent solution.

To establish the convergence of the scheme to a mesh-independent solution, the flow and heat transfer are computed by using an unstructured quadrilateral mesh of 16×8, 32×16, 64×32, 128×64 and 256×128 cells. The 64×32 mesh is shown in the left half of Fig. 11. The finer meshes are created by subdividing each cell into four. Plots of the local Nusselt numbers {defined as $Nu=q''L/[k(T_h-T_c)]$, where q'' is the local heat flux} along the vertical cold wall and the hot cylinder wall are shown in Fig. 12, along with the tabulated benchmark values from Demirdzic et al.(1992) obtained using the 256×128 quadrilateral cells. We find that the solutions for mesh densities greater than 128×64 cells are indistinguishable from the benchmark and from each other. This rate of convergence to a mesh-independent solution is similar to that of the second-order structured mesh scheme used in Demirdzic et al.(1992).

The right half of Fig. 11 shows the velocity vector plot. It captures the hot plume rising around the cylinder, and the downward flowing stream at



(a) along the vertical cold wall



(b) along the hot surface of the circular cylinder

Fig. 12 Comparison of the Nusselt number distributions at Pr=0.1, Ra=10^6

the cold wall, and it is very consistent with the figured benchmark one from Demirdzic et al. (1992).

Finally, it is noted here that the more detailed evaluation of the flow analysis code (PowerCFD) developed by the present method for its application to several 2-D or 3-D benchmark problems using a variety of unstructured quadrilateral, triangular and hybrid meshes are referred to Myong et al.(2005), Myong (2005) and Myong (2005c, d).

4. Conclusions

A conservative pressure-based finite-volume numerical method for unstructured meshes has been developed for computing flow and heat transfer by combining the attractive features of the pressure-based procedures with the advances made in existing unstructured grid techniques. The method uses an integral form of governing equations for arbitrary convex polyhedra. Care is taken in the discretization and solution procedure to avoid formulations that are cell-shape-specific. A collocated variable arrangement formulation is developed, i.e. all dependent variables such as pressure and velocity are stored at cell centers. For both convective and diffusive fluxes the forms superior to both accuracy and stability are adopted and formulated through a systematic study on their existing approximation ones. Gradients required for the evaluation of diffusion fluxes and for second-order-accurate convective operators are computed by using a linear reconstruction based on the divergence theorem. Momentum interpolation is used to prevent a pressure checkerboarding and the SIMPLE algorithm is used for a pressure-velocity coupling. The resulting set of coupled nonlinear algebraic equations is solved by employing a segregated approach, leading to a decoupled set of linear algebraic equations for each dependent variable, with a sparse diagonally dominant coefficient matrix. These equations are solved by an iterative preconditioned conjugate gradient solver which retains the sparsity of the coefficient matrix, thus achieving a very efficient use of computer resources. The flow analysis code (PowerCFD) developed by the present method is evaluated for its application to several 2-D structured-mesh benchmark problems by using a variety of unstructured quadrilateral and triangular unstructured meshes. The present pressure-based cell-centered numerical method for an unstructured mesh clearly demonstrate the same accuracy and robustness as that for a typical structured mesh.

References

Davison, L., 1996, "A Pressure Correction Method for Unstructured Meshes with Arbitrary Control Volumes," *Int. J. Numer. Methods Fluids*, Vol. 22, pp. 265~281.

Demirdzic, I. and Muzaferija, S., 1995, "Numerical Method for Coupled Fluid Flow, Heat Transfer and Stress Analysis using Unstructured Moving Meshes with Cells of Arbitrary Topology," *Comput. Methods Appl. Mech. Engrg.*, Vol. 125, pp. 235~255.

Demirdzic, I., Lilek, Z. and Peric, M., 1992, "Fluid Flow and Heat Transfer Test Problems for Non-Orthogonal Grids: Bench-Mark Solutions," *Int. J. Numer. Methods Fluids*, Vol. 15, pp. 329~354.

Ferziger, J. H. and Peric, M., 2002, Computational Methods for Fluid Dynamics, 3rd ed., Springer.

Hortmann, M., Peric, M. and Scheuerer, G., 1990, "Finite Volume Multigrid Prediction of Laminar Natural Convection: Benchmark Solutions," *Int. J. Numer. Methods Fluids*, Vol. 11, pp. 189~207.A.

Jiang, Y. and Przekwas, A. J., 1994, "Implicit, Pressure-Based Incompressible Navier-Stokes Equations Solver for Unstructured Meshes," AIAA-94-0305.

Lien, F. S., 2000, "A Pressure-Based Unstructured Grid Method for All-speed Flows," *Int. J. Numer. Meth. Fluids*, Vol. 33, pp. 355~374.

Mathur, S. R. and Murthy, J. Y., 1997, "A Pressure-Based Method for Unstructured Meshes," *Numerical Heat Transfer, Part B*, Vol. 31, pp. 195~215.

Miettien, A., 1997, A Study of the Pressure Correction Approach in the Collocated Grid Arrangement, *Ph.D. thesis*, Helsinki Univ. of Technology.

Muzaferija, S., 1994, Adaptive Finite Volume Method for Flow Predictions using Unstructured Meshes and Multigrid Approach, *Ph.D. Thesis*, University of London.

Myong, H. K., Kim, J. and Kim, J. E., 2005, "Development of 3-D Flow Analysis Code using Unstructured Grid System (II)- Code's Performance Evaluation -," (in Korean) *Trans. KSME Part B*, Vol. 29, No. 9, pp. 1057~1064.

Myong, H. K., 2005, "Numerical Simulation of Lid-Driven Flow in a Square Cavity at High Reynolds Numbers," (in Korean) *KSCFE J. of Computational Fluid Engineering*, Vol. 10, No. 4, pp. 18~23.

Myong, H. K., 2006a, "Evaluation of Numerical Approximations of Convection Flux in Unstructured Cell-Centered Method," (in Korean) *KSCFE J. of Computational Fluid Engineering*, Vol. 11, No. 1, pp. 36~42.

Myong, H. K., 2006b, "A New Numerical Approximation of Diffusion Flux in Unstructured Cell-Centered Method," (in Korean) *KSCFE J. of Computational Fluid Engineering*, Vol. 11, No. 1, pp. 8~15.

Myong, H. K., 2006c, "Numerical Study on the Characteristics of Natural Convection Flows in a Cubical Cavity," (in Korean) *Trans. KSME Part B*, Vol. 30, No. 4, pp. 337~342.

Myong, H. K., 2006d, "Numerical Simulation of Developing Turbulent Flow in a Circular Pipe of 180° Bend," (in Korean) to appear in Trans. KSME Part B.

Patankar, S. V., 1980, Numerical Heat Transfer and Fluid Flow, McGraw-Hill, New York.

Peric M., 1985, A Finite Volume Method for the Prediction of Three-Dimensional Fluid Flow in Complex Ducts, *Ph.D. Thesis*, Univ. of London, UK.

Rhie, C. M. and Chow, W. L., 1983, "Numerical Study of the Turbulent Flow past an Airfoil with Trailing Edge Separation," *AIAA J.*, Vol. 21, pp. 1523~1532.

Zwart, P. J., Raithby, G. D. and Raw, M. J., 1999, "An Integrated Space-Time Finite Volume Method for Moving Boundary Problems," *J. Comp. Physics*, Vol. 154, pp. 497~519.

de Vahl Davis, G., 1983, "Natural Convection of Air in a Square Cavity: A Bench Mark Numerical Solution," *Int. J. Numer. Methods Fluids*, Vol. 3, pp. 249~264.