Journal of Computational Physics **162,** 411–428 (2000)
doi:10.1006/jcph.2000.6546, available online at http://www.idealibrary.com on IDEAL®

A Second-Order Time-Accurate Finite Volume Method for Unsteady Incompressible Flow on Hybrid Unstructured Grids

Dongjoo Kim and Haecheon Choi

School of Mechanical and Aerospace Engineering, Seoul National University, Seoul 151-742, Korea; and National CRI Center for Turbulence and Flow Control Research, Institute of Advanced Machinery and Design, Seoul National University, Seoul 151-742, Korea
E-mail: choi@socrates.snu.ac.kr

Received July 29, 1999; revised March 6, 2000

A new second-order time-accurate fractional-step method for solving unsteady incompressible Navier–Stokes equations on hybrid unstructured grids is presented. The nonstaggered grid method, originally developed by Rhie and Chow (1983, *AIAA J.* **21**, 1525) for steady flow and further extended by Zang *et al.* (1994, *J. Comput. Phys.* **114**, 18) to unsteady flow on structured grids, is employed in the present study to enforce mass conservation on hybrid unstructured grids. The pressure and Cartesian velocity components are defined at the center of each cell, while the face-normal velocities are defined at the mid-points of the corresponding cell faces. A second-order fully implicit time-advancement scheme is used for time integration and the resulting nonlinear equations are linearized without losing the overall time accuracy. Both the momentum and Poisson equations are integrated with the finite volume method and the flow variables at the cell face are obtained using an interpolation scheme independent of cell shape. The present numerical method is applied to four different benchmark problems and proves to be accurate and efficient.

Key Words: finite volume method; unsteady incompressible flow; nonstaggered unstructured grid.

1. INTRODUCTION

In this paper we present a second-order time-accurate numerical method for solving unsteady incompressible Navier–Stokes equations on hybrid unstructured grids. In the case of the structured grid, one of the most popular methods to obtain a time-accurate solution for unsteady incompressible flow is a fractional-step method on the staggered grid system, where a pseudo-pressure is used to correct the velocity field such that the continuity equation

0021-9991/00 $35.00

is satisfied at each computational time step [1]. However, in the case of the unstructured grid, the staggered grid system makes the code more complicated than the nonstaggered grid system, because for the staggered grid the control volumes for the pressure and velocity are very different from each other. Therefore, in general the nonstaggered grid system is preferred to the staggered grid system in the case of the unstructured grid.

The nonstaggered grid system had been known to produce unrealistic pressure oscillations in a converged solution until Rhie and Chow [2] presented a numerical method (momentum interpolation method) for steady flow which satisfies the mass conservation on nonstaggered structured grids without artificial pressure oscillations. Later, Zang *et al.* [3] applied the concept of the momentum interpolation method to a fractional-step method for unsteady flow on nonstaggered structured grids. This momentum interpolation method has been also applied to steady flow on unstructured grids by a few researchers [4–6]. However, to the best of our knowledge, there has been no effort to apply the momentum interpolation method to unsteady flow on nonstaggered unstructured grids.

Unstructured grids have been mostly used with the finite element method or its variant, but the finite volume method with unstructured grids has become popular recently. The advantage of using the finite volume method is that it is conservative as long as surface integrals (convective and diffusive fluxes) are the same for the control volumes sharing the surface. Finite volume methods on unstructured grids have been mostly used for steady flow [4–11] but a few have been used for unsteady flow [12–16]. Miller and Wang [12] used the stream-function–vorticity formulation for two-dimensional flow, but this method cannot be directly applied to three-dimensional flow. Pan *et al.* [13] and Weiss and Smith [14] employed the artificial compressibility method, which needs a subiterative procedure at each computational time step and thus requires more computational efforts. Schulz and Kallinderis [15] and Chen and Kallinderis [16] used a first-order (forward Euler) pressure-correction method on "traditional" nonstaggered grids (i.e., without using the momentum interpolation method) and introduced artificial dissipation to suppress oscillatory solutions. Therefore, an accurate and efficient numerical method for unsteady flow on nonstaggered unstructured grids, which is also easily applied to three-dimensional flow, should be developed.

The objective of this work is to develop a new second-order time-accurate numerical method for calculating unsteady incompressible flow on hybrid unstructured grids. The momentum interpolation method is used together with the nonstaggered grid system, where the pressure and Cartesian velocity components are defined at the center of each cell and the face-normal velocities are defined at the mid-points of the corresponding cell faces, and thus artificial pressure oscillations are eliminated. The time integration method is based on a fully implicit fractional-step method and the resulting nonlinear momentum equations are linearized without losing the overall time accuracy. A finite volume method is used for the spatial derivative terms and the flow variables at the cell face are obtained using an interpolation scheme which is independent of the cell shape and thus can be applied to hybrid unstructured grids. The accuracy of the present method is verified by analyzing four benchmark problems (decaying vortices, lid-driven cavity flow, backward-facing step flow, and flow over a circular cylinder).

The numerical procedures used in the present method are described in Section 2. Section 3 shows numerical results for four different flow geometries, followed by a summary in Section 4.

2. NUMERICAL METHOD

2.1. *Governing Equations and Grid System*

The governing equations for unsteady incompressible viscous flow are

$$\frac{\partial u_i}{\partial t} + \frac{\partial}{\partial x_j} u_i u_j = -\frac{\partial p}{\partial x_i} + \frac{1}{\mathrm{Re}} \frac{\partial}{\partial x_j} \frac{\partial}{\partial x_j} u_i, \tag{1}$$

$$\frac{\partial u_i}{\partial x_i} = 0, \tag{2}$$

where x_i's are the Cartesian coordinates and u_i's are the corresponding velocity components. All variables are nondimensionalized by characteristic velocity and length scales, and Re is the Reynolds number.

Figure 1 shows four different unstructured grid systems. In a traditional nonstaggered grid shown in Fig. 1a, physically unrealistic pressure oscillations may occur due to decoupling between the pressure and the velocity fields. In contrast, staggered grids shown in Figs. 1b and 1c eliminate pressure oscillations, but the control volumes for the pressure and velocity components are very different from each other which makes the code more complicated especially in three-dimensional unstructured grids. The difference between the grid systems shown in Figs. 1b and 1c is that the former [9, 10] defines and updates the Cartesian velocity components on cell face, but the latter [17–19] updates only the face-normal velocity on cell face and then obtains the tangential velocity by interpolating the updated face-normal velocities. The staggered grid system using the face-normal velocity

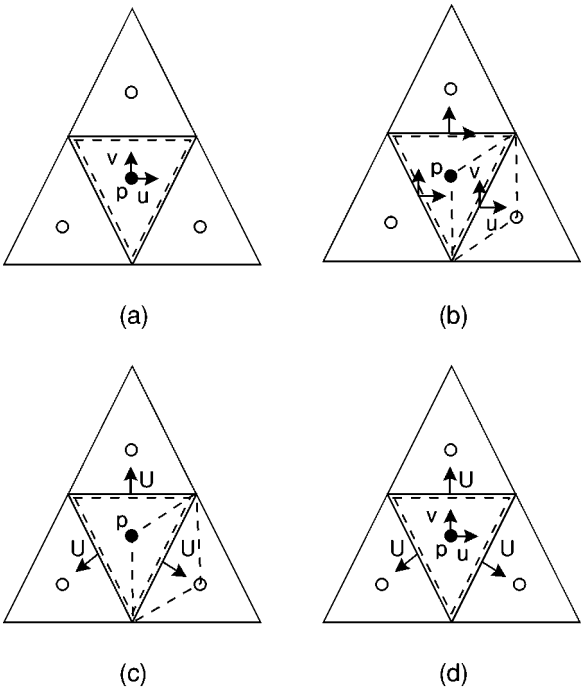

FIG. 1. Various unstructured grid systems: (a) traditional nonstaggered grid; (b) staggered grid with Cartesian velocity components; (c) staggered grid with face-normal velocity component; (d) present nonstaggered grid. Dashed lines denote control volumes for dependent variables.

only has been used more often than that using the Cartesian velocity components because of the simplicity and memory–CPU savings. However, the staggered grid system using the face-normal velocity has been proved to be first-order accurate on general triangular grids and second-order accurate on uniform triangular grids [17, 19, 20].

The drawbacks of the traditional nonstaggered and staggered grids can be overcome by employing a nonstaggered grid shown in Fig. 1d, where the pressure and Cartesian velocity components are defined at each cell center, and the face-normal velocity is defined at the mid-point of each cell face. The face-normal velocity U is defined as

$$U = (u_i)_{\text{face}} n_i, \tag{3}$$

where $(u_i)_{\text{face}}$ and n_i are the Cartesian velocities and outward-normal unit vector on the cell face, respectively. The benefit of this grid system is that it is free from the pressure oscillation problem and simple as compared to the staggered grid especially in three-dimensional flow. The pressure oscillations are prevented by using the momentum interpolation method [2]. The nonstaggered grid system shown in Fig. 1d is used in the present study.

2.2. *Time Integration*

The time-integration method used to solve Eqs. (1) and (2) is based on a fractional-step method [1, 21, 22], where a pseudo-pressure (or pressure) is used to correct the velocity field such that the continuity equation is satisfied at each computational time step. In this study, we use a second-order fully implicit time-advancement scheme (Crank–Nicolson method) for both the convective and the viscous terms in Eq. (1),

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} + \frac{1}{2}\frac{\partial}{\partial x_j}\left(u_i^{n+1}u_j^{n+1} + u_i^n u_j^n\right) = -\frac{\partial p^{n+1}}{\partial x_i} + \frac{1}{2}\frac{1}{\text{Re}}\frac{\partial}{\partial x_j}\frac{\partial}{\partial x_j}\left(u_i^{n+1} + u_i^n\right), \tag{4}$$

where Δt is the computational time step and the superscript n is the time step. Implicit treatment of the convective and viscous terms eliminates the numerical stability restriction.

We linearize the nonlinear terms in Eq. (4) without losing the overall time accuracy [23]:

$$u_i^{n+1}u_j^{n+1} = u_i^{n+1}u_j^n + u_i^n u_j^{n+1} - u_i^n u_j^n + O(\Delta t^2). \tag{5}$$

Then, the present fully implicit time-advancement scheme for Eqs. (1) and (2) can be written as follows:

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} + \frac{1}{2}\frac{\partial}{\partial x_j}\left(u_i^{n+1}u_j^n + u_i^n u_j^{n+1}\right) = -\frac{\partial p^{n+1}}{\partial x_i} + \frac{1}{2}\frac{1}{\text{Re}}\frac{\partial}{\partial x_j}\frac{\partial}{\partial x_j}\left(u_i^{n+1} + u_i^n\right), \tag{6}$$

$$\frac{\partial u_i^{n+1}}{\partial x_i} = 0. \tag{7}$$

Next, we apply a fractional-step method [22] to Eqs. (6) and (7):

$$\frac{\hat{u}_i - u_i^n}{\Delta t} + \frac{1}{2}\frac{\partial}{\partial x_j}\left(\hat{u}_i u_j^n + u_i^n \hat{u}_j\right) = -\frac{\partial p^n}{\partial x_i} + \frac{1}{2}\frac{1}{\text{Re}}\frac{\partial}{\partial x_j}\frac{\partial}{\partial x_j}\left(\hat{u}_i + u_i^n\right), \tag{8}$$

$$\frac{u_i^* - \hat{u}_i}{\Delta t} = \frac{\partial p^n}{\partial x_i}, \tag{9}$$

$$\frac{\partial}{\partial x_i}\frac{\partial p^{n+1}}{\partial x_i} = \frac{1}{\Delta t}\frac{\partial u_i^*}{\partial x_i}, \tag{10}$$

$$\frac{u_i^{n+1} - u_i^*}{\Delta t} = -\frac{\partial p^{n+1}}{\partial x_i}, \tag{11}$$

where $\hat{u}_i$ and u_i^* are the intermediate velocities. After integrating Eqs. (8) and (10) over each cell area A and applying the divergence theorem, Eqs. (8)–(11) become

$$\frac{\delta\hat{u}_i}{\Delta t} + \frac{1}{A}\oint_l \frac{1}{2}\left(U^n\delta\hat{u}_i + u_i^n n_j \delta\hat{u}_j + 2u_i^n U^n\right) dl$$
$$= -\frac{1}{A}\int_A \frac{\partial p^n}{\partial x_i}\, dA + \frac{1}{A}\oint_l \frac{1}{2}\frac{1}{\mathrm{Re}}\frac{\partial}{\partial n}\left(\delta\hat{u}_i + 2u_i^n\right) dl, \tag{12}$$

$$u_i^* - \hat{u}_i = \Delta t\frac{\partial p^n}{\partial x_i}, \tag{13}$$

$$\frac{1}{A}\oint_l \frac{\partial p^{n+1}}{\partial n}\, dl = \frac{1}{A\Delta t}\oint_l U^*\, dl, \tag{14}$$

$$u_i^{n+1} - u_i^* = -\Delta t\frac{\partial p^{n+1}}{\partial x_i}, \tag{15}$$

$$U^{n+1} - U^* = -\Delta t\frac{\partial p^{n+1}}{\partial n}, \tag{16}$$

where $\delta\hat{u}_i = \hat{u}_i - u_i^n$, U and n_i are defined in Eq. (3), $\partial/\partial n$ is the outward-normal derivative at the cell face, A is the area of each cell, and l is the boundary surrounding A. The intermediate face-normal velocity U^* in Eq. (14) is defined on each cell face and is obtained by interpolation from adjacent cell-center velocities u_i^*, i.e., $U^* = \tilde{u}_i^* n_i$, where the superscript $\tilde{}$ denotes interpolation. The interpolation method and numerical procedure to solve Eqs. (12) and (14) are described in Sections 2.3 and 2.4. In this paper Eqs. (12)–(16) are derived for two-dimensional flow, but the extension to three-dimensional flow is straightforward.

The cell-face velocity interpolated from u_i^{n+1} does not satisfy the divergence-free condition, i.e.,

$$\oint_l \tilde{u}_i^{n+1} n_i\, dl \neq 0, \tag{17}$$

but the face-normal velocity obtained from Eq. (16) satisfies the condition

$$\oint_l U^{n+1}\, dl = 0. \tag{18}$$

It is important to note that U^{n+1} is obtained from Eq. (16) rather than interpolation from adjacent cell-center velocities u_i^{n+1} and the divergence-free velocity field U^n is used for the calculation of the convective fluxes in Eq. (12). This procedure ensures strong coupling between the pressure and the velocity. A similar procedure of using the momentum interpolation method can be found in Zang *et al.* [3] for structured grids in curvilinear coordinates.

The boundary condition for the intermediate velocity $\hat{u}_i$ is simply $\hat{u}_i = u_i^{n+1}$ to second order in the time step. This can be shown from Eqs. (9) and (11),

$$\hat{u}_i = u_i^{n+1} + \Delta t \frac{\partial (p^{n+1} - p^n)}{\partial x_i} = u_i^{n+1} + O(\Delta t^2). \tag{19}$$

Replacing the boundary value of $\partial p^{n+1}/\partial n$ by $(U^* - U^{n+1})/\Delta t$ from Eq. (16), the Poisson equation for the cell adjacent to the boundary is modified as

$$\frac{1}{A} \sum_{\substack{j=1 \\ j \neq \text{bdry}}}^{N_f} \left(\frac{\partial p^{n+1}}{\partial n} \right)_j \Delta l_j = \frac{1}{A \Delta t} \left(\sum_{\substack{j=1 \\ j \neq \text{bdry}}}^{N_f} U_j^* \Delta l_j + \sum_{j=\text{bdry}} U_j^{n+1} \Delta l_j \right), \tag{20}$$

where N_f is the number of cell faces, Δl_j is the length of the jth cell face, and the subscript bdry denotes the boundary. Note that since the physical velocity flux $U_j^{n+1} \Delta l_j$ is used on the boundary rather than the intermediate velocity flux $U_j^* \Delta l_j$, global mass conservation is automatically satisfied.

In the present study, the Courant–Friedrichs–Lewy (CFL) number is defined as

$$\text{CFL} = \frac{1}{2} \frac{1}{A} \sum_{j=1}^{N_f} |U_j \Delta l_j| \Delta t. \tag{21}$$

For the time-advancement scheme, one may consider a semi-implicit scheme such as a second-order Adams–Bashforth method [1] or a third-order Runge–Kutta method [21] for the convective term and the Crank–Nicolson method for the viscous term. Those schemes have been successfully applied to a variety of incompressible flow problems. An implicit method is preferred when the time step limit imposed by an explicit or semi-implicit stability bound is significantly less than that imposed by the accuracy requirement. If the flow geometry contains sharp corners which exist in many flow geometries, rapid variation of flow variables in their vicinity would require dense grid clustering which restricts the computational time step. A fully implicit method overcomes this restriction with a trade-off of possible higher operation counts per time step. For example, Choi *et al.* [24] used a fully implicit method in a curvilinear coordinate system to simulate flow over riblets and reduced the required CPU time by a factor of five as compared to that of a conventional semi-implicit method.

2.3. *Interpolation*

To perform integrations in Eqs. (12) and (14), Φ and $\frac{\partial \Phi}{\partial n}$ at the mid-point on each cell face have to be evaluated by using adjacent values, where Φ is an arbitrary flow variable. In the present study, both Φ and $\frac{\partial \Phi}{\partial n}$ are obtained by using the values (Φ_1, Φ_2) at the neighboring cell center points (P_1, P_2) and the values (Φ_a, Φ_b) at the nodal points (P_a, P_b) (see Fig. 2).

First, let us consider $\frac{\partial \Phi}{\partial n}$ at P_f, where P_f is the mid-point on the cell face (Fig. 2). We define a generalized coordinate system with covariant bases $(\mathbf{e_1}, \mathbf{e_2})$ locally on each cell face, where $\mathbf{e_1}$ and $\mathbf{e_2}$ are unit vectors from P_c (the cross-sectional point of $\overline{P_1 P_2}$ and $\overline{P_a P_b}$) to P_2 and from P_c to P_b, respectively. Note that P_c is not the mid-point on the cell face. The

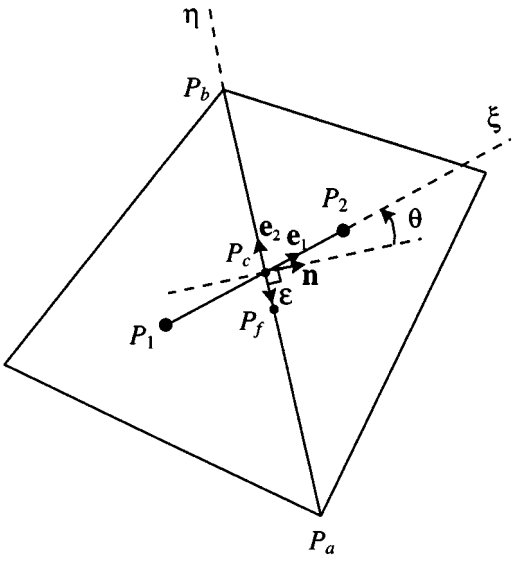

FIG. 2. Interpolation of flow variables at the mid-point on the cell face.

gradient of Φ at P_c can be expressed as

$$\nabla\Phi = \frac{\partial\Phi}{\partial\xi}\mathbf{e^1} + \frac{\partial\Phi}{\partial\eta}\mathbf{e^2}, \tag{22}$$

where ξ and η represent the directions along $\mathbf{e_1}$ and $\mathbf{e_2}$, respectively, and $\mathbf{e^1}$ and $\mathbf{e^2}$ are the corresponding contravariant bases [25]. Then, the normal component of $\nabla\Phi$ at P_f can be written as

$$\left.\frac{\partial\Phi}{\partial n}\right|_{P_f} = \nabla\Phi\cdot\mathbf{n} = \frac{\Phi_2-\Phi_1}{\delta_1+\delta_2} - \frac{\Phi_b-\Phi_a}{\Delta\eta}\tan\theta, \tag{23}$$

where δ_1 and δ_2 are the normal distances to the cell face, respectively, from P_1 and P_2, θ is the angle between $\mathbf{n}$ and $\mathbf{e_1}$, and $\Delta\eta$ is the distance from P_a to P_b. It is interesting to note that Eq. (23) is composed of two terms: the first one corresponds to the principal diffusion and the second one corresponds to the cross diffusion as in the curvilinear coordinate system. When $\mathbf{n}$ is parallel to $\overline{P_1P_2}$, the second term vanishes.

The values at the nodal points, Φ_a and Φ_b, are obtained as

$$\Phi_\alpha = \frac{\sum_{i=1}^{N_{cs}}(\Phi_i/A_i)}{\sum_{i=1}^{N_{cs}}(1/A_i)}, \tag{24}$$

where $\alpha = a$ or b, N_{cs} is the number of the cells sharing the node, Φ_i is the ith cell value, and A_i is the ith cell area. This interpolation method is second-order accurate on "regular" triangular grids and is equivalent to the bilinear interpolation on rectangular grids, where regular triangles correspond to equilateral triangles or triangles made by adding diagonals in a uniform rectangular grid. As an alternative, one may use a linear interpolation by properly choosing three cell center values among all the neighboring cell center values, but the choice of three cells is very arbitrary.

Next, let us consider Φ at P_f. The value Φ_c at P_c is easily obtained with a second-order accuracy (linear interpolation):

$$\Phi_c = \frac{\delta_1 \Phi_2 + \delta_2 \Phi_1}{\delta_1 + \delta_2}. \tag{25}$$

Then, we evaluate Φ_f by adding a correction term to Φ_c,

$$\begin{aligned} \Phi_f &= \frac{\delta_1 \Phi_2 + \delta_2 \Phi_1}{\delta_1 + \delta_2} + \nabla \Phi \cdot \boldsymbol{\varepsilon} \\ &= \frac{\delta_1 \Phi_2 + \delta_2 \Phi_1}{\delta_1 + \delta_2} + \frac{\Phi_a - \Phi_b}{\Delta \eta} |\boldsymbol{\varepsilon}|, \end{aligned} \tag{26}$$

where $\boldsymbol{\varepsilon}$ is the vector from P_c to P_f and $|\boldsymbol{\varepsilon}|$ is the magnitude of $\boldsymbol{\varepsilon}$. This interpolation method is second-order accurate and equivalent to a central difference scheme when it is used for evaluating convective fluxes. Similar interpolation methods were also used by previous researchers [4, 6], but Davidson [4] did not consider the second term in Eq. (26). However, we will show in Section 3.1 that the second term is necessary in order to maintain a second-order accuracy.

Finally, the pressure gradient term $\partial p / \partial x_i$ at the cell center (Eqs. (13) and (15)) is evaluated as

$$\frac{\partial p}{\partial x_i} = \frac{1}{A} \int_A \frac{\partial p}{\partial x_i} \, dA = \frac{1}{A} \oint_l p n_i \, dl, \tag{27}$$

where the pressure at the cell face is evaluated as in Eq. (26).

Unlike the staggered grid system, the pressure on the boundary is needed when solving for $\hat{u}_i$ (Eq. (12)) and therefore should be updated at each time step. In the present study, the pressure on the boundary is updated by extrapolation from interior pressure values (Fig. 3): p_A (pressure at the mid-point on the boundary cell face) and p_B (pressure at the boundary nodal point) are obtained by linear interpolation of p_1 and p_2 (pressures at cell centers) by assuming $\partial p / \partial n = 0$, i.e., $p_A = p_a$ and $p_B = p_b$.

The interpolation method used in this study is derived without any assumption about the cell shape and therefore can be applied to hybrid unstructured grids.

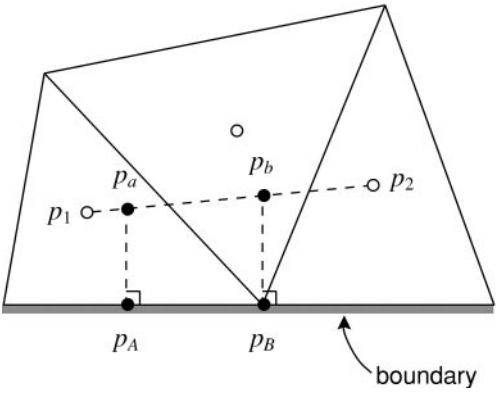

FIG. 3. Interpolation of the pressure on the boundary.

2.4. *Matrix Solver*

After the interpolation procedure described in Section 2.3, Eqs. (12) and (14) can be represented as

$$A_{ij}\phi_j = b_i, \tag{28}$$

where A_{ij} is a sparse matrix of the present linear system, ϕ_j is a vector containing the unknown flow variables at cell center points, and b_i is a vector containing the values at the nth time step.

It should be mentioned here that in Eq. (12) $\delta\hat{u}_j$ is coupled with $\delta\hat{u}_i$. We may consider two different approaches to resolve this problem. One is to include the terms containing $\delta\hat{u}_j$ $(j \neq i)$ in b_i and do global iterations until $\delta\hat{u}_i$ converges. In this case, the size of the matrix is $N_c \times N_c$, where N_c is the number of cells. The other is to solve $\delta\hat{u}_i$ and $\delta\hat{u}_j$ $(j \neq i)$ at the same time by making a larger matrix A_{ij} ($2N_c \times 2N_c$ or $3N_c \times 3N_c$, respectively, for two- or three-dimensional flow). In this case, no global iteration is needed unlike the former method, but the matrix A_{ij} becomes large and thus stiffer than that of the former method. In the present study, where two-dimensional benchmark flow problems are tested, we use the latter method. However, for three-dimensional flow, the matrix A_{ij} becomes very stiff, and thus the former method may be preferred to the latter one.

A direct solver such as the LU decomposition may be used to solve Eq. (28), but it is impractical because it demands too much memory. Therefore, an iterative method is used to solve Eq. (28) and only nonzero terms of the sparse matrix A_{ij} are stored by using the ITPACK storage format [26] which is useful for implementation of iterative methods on parallel and vector processors. In this study, a biconjugate gradient stabilized (BI-CGSTAB) method [26] is used for both the momentum equation (12) and Poisson equation (14), and the Jacobi preconditioner (Sections 3.1, 3.2, and 3.3) or an incomplete LU preconditioner (Section 3.4) is used to enhance the convergence rate. In the case of using the incomplete LU preconditioner, we reorder the grid numbers with a multicoloring technique for vectorization.

The convergence criteria for Eqs. (12) and (14) are

$$\max[|R_i(\delta\hat{u}_i)|] < \epsilon, \tag{29}$$

$$\max[\Delta t|R_i(p)|] < \epsilon, \tag{30}$$

where $R_i(\delta\hat{u}_i)$ and $R_i(p)$ are the residuals of Eqs. (12) and (14), respectively ($R_i = b_i - A_{ij}\phi_j$). Here, $\max[\Delta t|R_i(p)|]$ corresponds to the maximum value of $|\nabla \cdot \mathbf{u}|$, i.e., $|\frac{1}{A}\oint_l U^{n+1}\,dl|$. In Section 3, ϵ is prescribed to be 10^{-8}, 10^{-3}, and 10^{-4}, respectively, for decaying vortices (Section 3.1), steady flows (Sections 3.2 and 3.3), and flow over a circular cylinder (Section 3.4).

3. NUMERICAL EXAMPLES

3.1. *Decaying Vortices*

The temporal and spatial accuracy of the present numerical method is verified by simulating the following two-dimensional unsteady flow, which has been investigated by previous

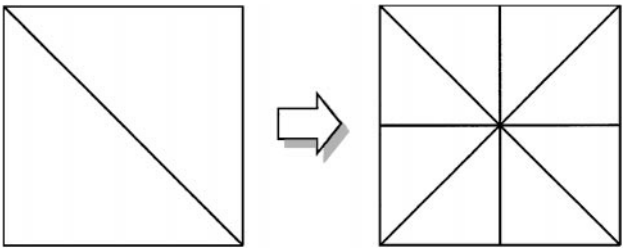

FIG. 4. Mesh refinement.

researchers [1, 3, 21]:

$$
\begin{aligned}
u(x, y, t) &= -\cos \pi x \sin \pi y \, e^{-2\pi^2 t/\mathrm{Re}}, \\
v(x, y, t) &= \sin \pi x \cos \pi y \, e^{-2\pi^2 t/\mathrm{Re}}, \\
p(x, y, t) &= -\frac{1}{4}(\cos 2\pi x + \cos 2\pi y) \, e^{-4\pi^2 t/\mathrm{Re}}.
\end{aligned} \tag{31}
$$

The computational domain is $-1/2 \le x, y \le 1/2$ and computations are carried out at $\mathrm{Re} = 10$, where $\mathrm{Re} = U_1 L/\nu$ and U_1 is the initial maximum velocity and L is the size of a vortex. The initial velocity condition at $t = 0$ and the velocities at the boundaries in time are provided from the exact solution. Four different sizes of uniformly-distributed right-angled triangles are used to determine the overall accuracy of the present numerical method: numbers of cells used are 200, 800, 3200, and 12800, respectively. In this study, with mesh refinement, each triangle is split into four right-angled triangles as shown in Fig. 4.

First, computations are performed with varying the mesh size but keeping the maximum CFL number constant as in [1, 3, 21]. Figure 5a shows the variation of the maximum error in u with mesh refinement and the effect of the second term in Eq. (26) on the overall accuracy for three different maximum CFL numbers. It is clear that the present numerical method including the second term in Eq. (26) is second-order accurate for all the CFL numbers investigated. However, when the term is neglected, the error becomes larger and the accuracy is not second order. Note that the accuracy without the second term in Eq. (26) becomes nearly a second order with $\mathrm{CFL} = 3$ because the spatial-interpolation error contributes less to total error as the CFL (time step) increases. The same result is also obtained for v.

Second, the spatial and temporal accuracies are investigated, respectively. Figure 5b shows the variation of the maximum error in u by varying mesh size but keeping the computational time step constant and therefore the slope shown in Fig. 5b denotes the spatial accuracy. Note that the computational time step in this case is determined to be a small value ($\Delta t = 0.001$) such that temporal error has negligible effect on total error. It is clear that inclusion of the second term in Eq. (26) makes the spatial accuracy second-order. Similarly, the temporal accuracy is investigated by varying the time step but keeping the mesh size constant. Here, the finest grid of 12800 cells is used to minimize the spatial error. Figure 5c shows that the present time integration scheme is indeed second-order accurate.

3.2. *Lid-Driven Cavity Flow*

The geometry, boundary conditions, and grid for the flow in a lid-driven square cavity are shown in Fig. 6. The Reynolds number is defined as $\mathrm{Re} = U_1 L/\nu$, where U_1 is the velocity

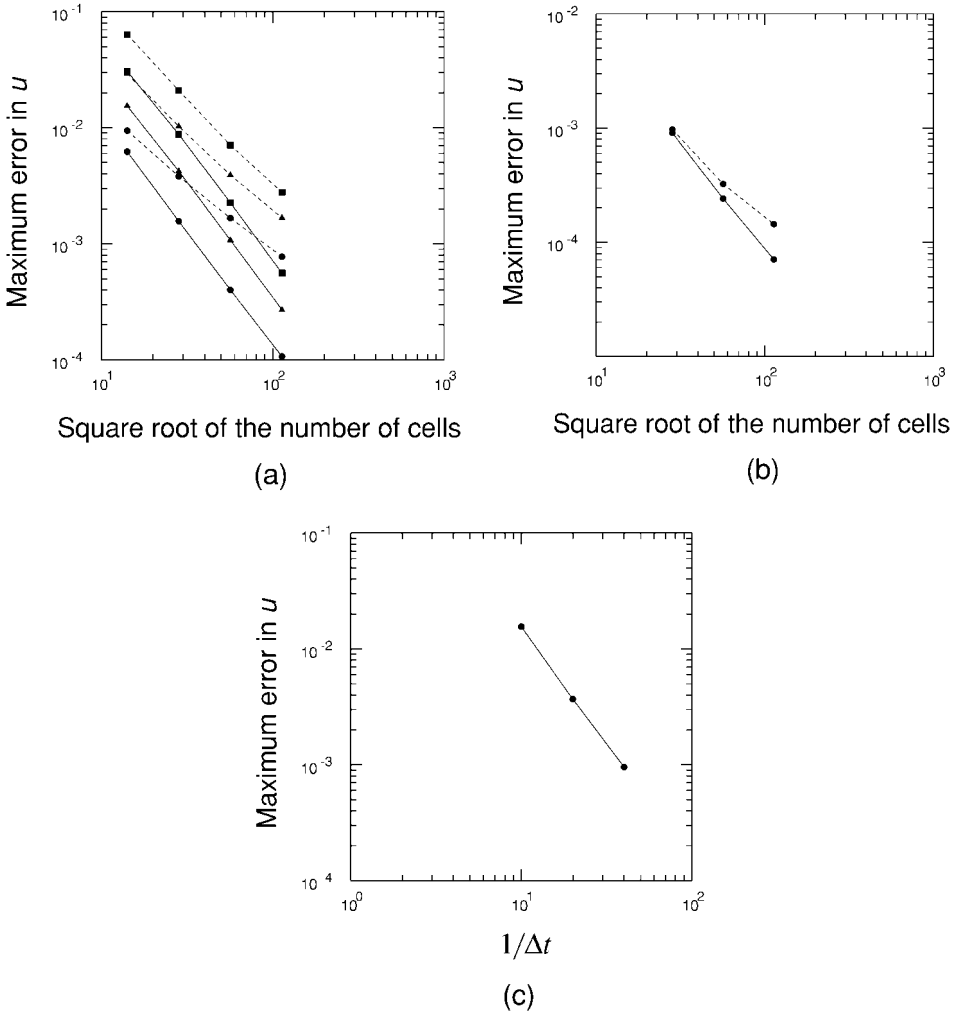

FIG. 5. Maximum error in u at $t = 0.3$: —, the second (correction) term in Eq. (26) included; ----, the term neglected. (a) Maximum CFL constant: ●, CFL = 1; ▲, CFL = 2; ■, CFL = 3; (b) spatial accuracy ($\Delta t = 0.001$); (c) temporal accuracy (12800 cells).

of the top lid and L is the length of the bottom wall. A calculation is performed at $\mathrm{Re} = 1000$ with a hybrid unstructured grid shown in Fig. 6, where 2496 rectangular cells and 3200 triangular cells are distributed near the walls and in the core region, respectively. In this calculation, the maximum CFL number is fixed to be 10. Figure 7 shows the centerline velocities $u(y)$ and $v(x)$ along the vertical and horizontal centerlines, respectively. The present result is in good agreement with that of Ghia *et al.* [27] who used 128×128 uniformly distributed rectangular cells.

3.3. *Backward-Facing Step Flow*

Another widely used benchmark problem to examine the accuracy of numerical methods is flow behind a backward-facing step in a channel. The flow geometry and boundary conditions are described in Fig. 8. The expansion ratio is 1 : 2 and the length of the computational

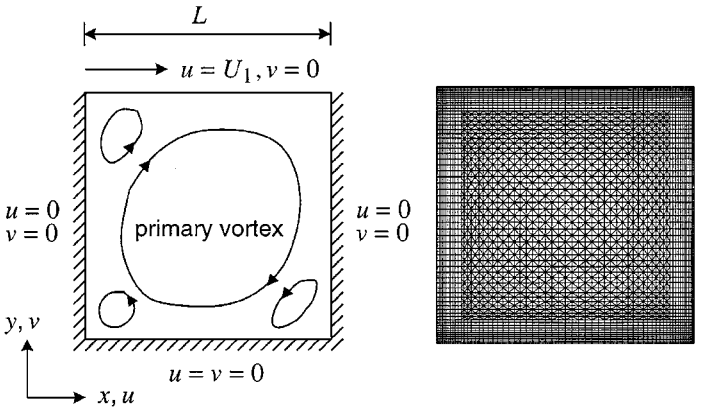

FIG. 6. Geometry, boundary conditions, and grid for the lid-driven cavity flow.

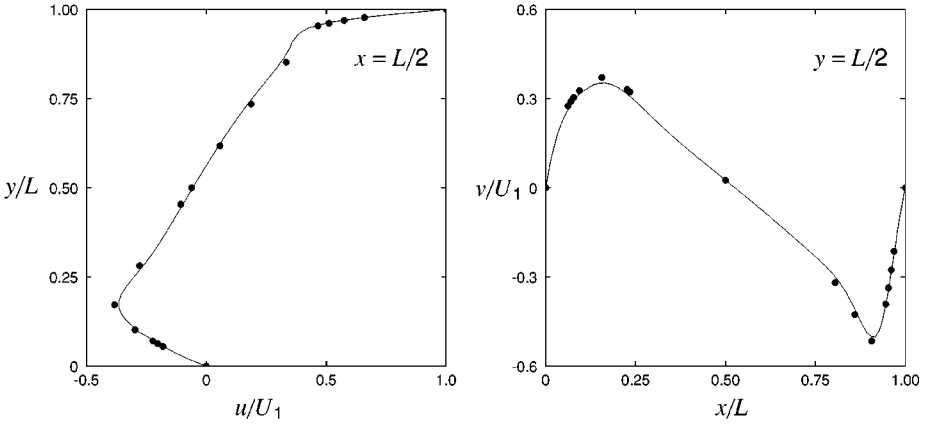

FIG. 7. Centerline velocities u and v (Re = 1000): —, present study; ●, Ghia *et al.* [27].

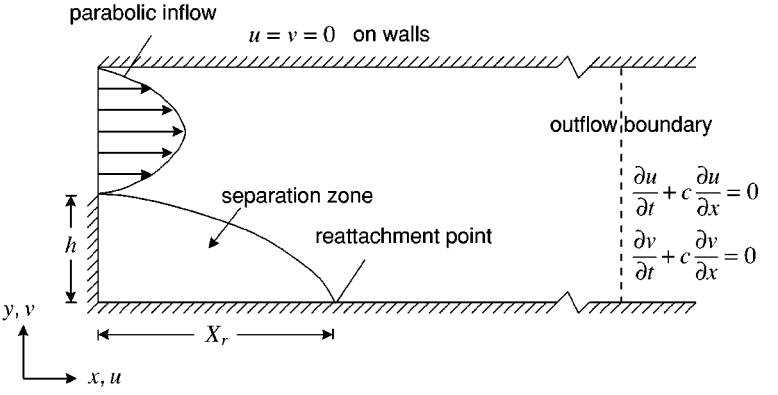

FIG. 8. Geometry and boundary conditions for the backward-facing step flow.

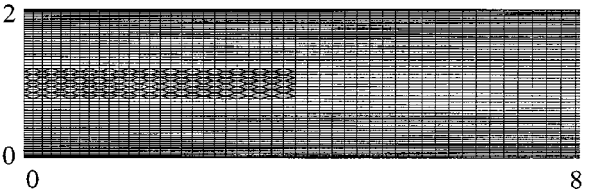

FIG. 9. Grid for the backward-facing step flow.

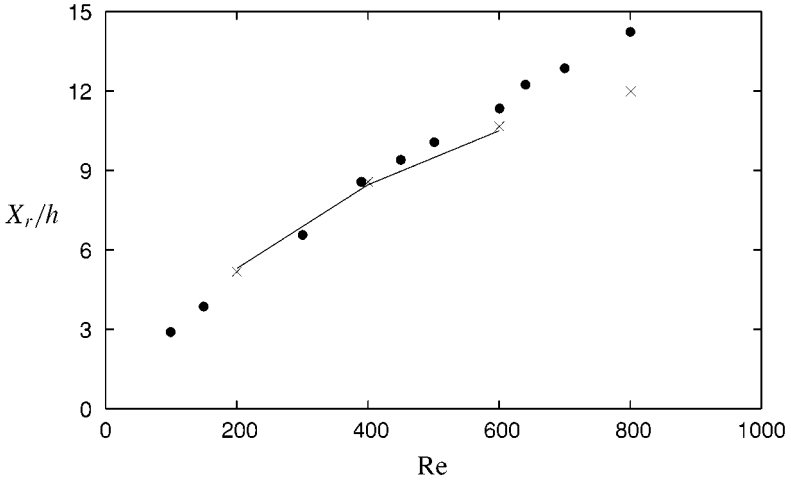

FIG. 10. Reattachment length as a function of the Reynolds number: —, present study; ●, Armaly *et al.* [29]; ×, Kim and Moin [1].

domain is $30h$, where h is the step height. A fully developed parabolic velocity profile is prescribed at the inflow boundary and a convective boundary condition, $\partial u_i/\partial t + c\partial u_i/\partial x = 0$, is used at the outflow boundary, where c is the space-averaged streamwise velocity at the exit [28]. A hybrid unstructured grid of 7932 rectangles and 520 triangles is used (Fig. 9), where the triangles are locally distributed near the step ($0 \le x/h \le 4$, $-0.8 \le y/h \le 0.8$). Calculations are performed at $\mathrm{Re} = 200$, 400, and 600, where $\mathrm{Re} = U_{\mathrm{avg}}(2h)/\nu$ and U_{avg} is the bulk velocity. The computational time step is fixed to be $\Delta t = 0.15h/U_{\mathrm{avg}}$ (maximum CFL $\simeq$ 2.2∼3.2).

Figure 10 shows the calculated reattachment length as a function of the Reynolds number, in comparison with the previous results of Armaly *et al.* [29] and Kim and Moin [1]. The present result is in good agreement with the computational result of Kim and Moin [1] for all the Reynolds numbers calculated. However, at $\mathrm{Re} > 400$, a difference exists between the computational (present study and [1]) and experimental [29] results. This difference may result from the three-dimensionality of the flow as Armaly *et al.* [29] pointed out. At $\mathrm{Re} = 600$, a secondary separation bubble exists on the upper wall (Fig. 11). The bubble length is $7.5h$, which agrees well with the result $7.8h$ of Kim and Moin [1].

3.4. *Flow over a Circular Cylinder*

Flow over a circular cylinder at $\mathrm{Re} > 47$ is a typical example of unsteady flow because vortex shedding takes place at that Reynolds number range [30, 31]. Figure 12 shows the geometry and boundary conditions for the flow. The computational domain used is $-20d \le x \le 20d$ and $-50d \le y \le 50d$, where d is the diameter of the cylinder and $(x = 0, y = 0)$ corresponds to the center of the cylinder. Dirichlet boundary conditions are used at farfield boundaries as

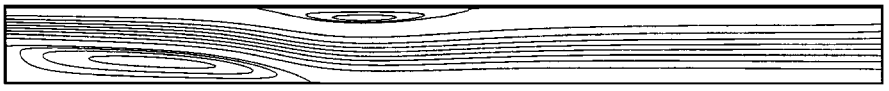

FIG. 11. Streamlines at $\mathrm{Re} = 600$.

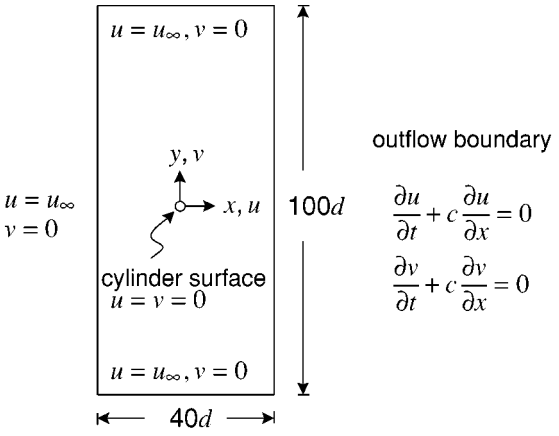

FIG. 12. Geometry and boundary conditions for flow over a circular cylinder.

well as at the cylinder surface, and a convective boundary condition is used for the outflow boundary. The convective boundary condition allows vortices to smoothly pass away from the computational domain.

Figure 13 shows a hybrid unstructured grid used in this study, which is generated by heaping up six layers of quadrilaterals and then filling the rest with 10384 triangles by the advancing front method [32]. Sixty four grid points are distributed on the cylinder surface.

Calculations are performed at three different Reynolds numbers: $\mathrm{Re} = 80$, 100, and 120, where $\mathrm{Re} = u_\infty d/\nu$ and u_∞ is the uniform inlet velocity. The computational time step is fixed to be $\Delta t = 0.06 d/u_\infty$, which corresponds to the maximum CFL number of about 3.3. Initially, random disturbances are imposed on a uniform velocity to quickly generate vortex shedding.

Figure 14 shows the Strouhal number as a function of the Reynolds number, where $\mathrm{St} = f d/u_\infty$ and f is the shedding frequency. The calculated Strouhal number is compared with the experimental correlation of Williamson's [33] and the computational result of Park

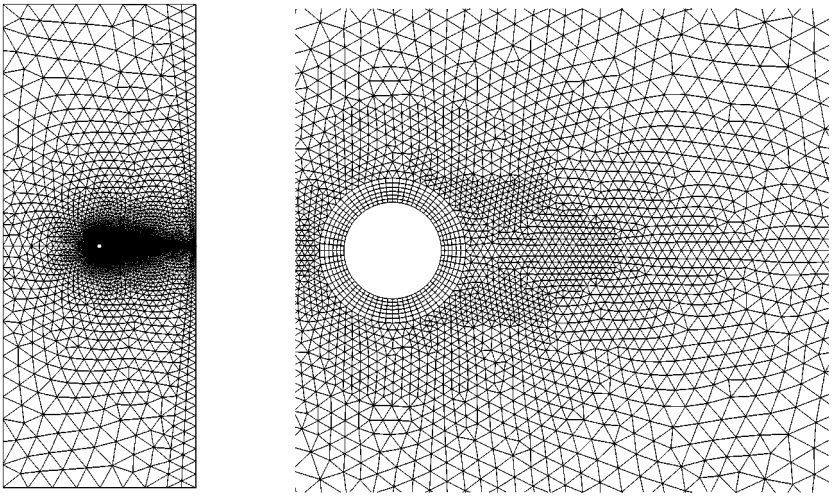

FIG. 13. Hybrid unstructured grid for flow over a circular cylinder.

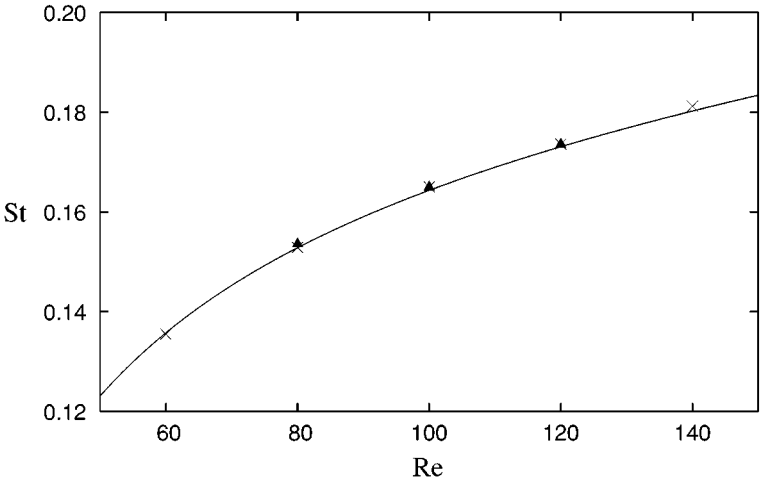

FIG. 14. Strouhal number as a function of the Reynolds number: ▲, present study; —, $St = -3.3265/Re + 0.1816 + 1.6 \times 10^{-4} Re$ (Williamson [33]); ×, Park *et al.* [30].

et al. [30] obtained by using a very fine structured grid with about 150,000 quadrilaterals. An excellent agreement is found among the three results.

Figure 15 shows the time-averaged separation bubble size, which is the distance from the base of the cylinder to the point where the time-averaged streamwise velocity is zero. Also shown is the computational result of Park *et al.* [30]. The present result shows an excellent agreement with that of Park *et al.* [30].

Figure 16 shows contours of instantaneous pressure and vorticity near the cylinder at $Re = 100$. As is shown, instantaneous pressure contours are very smooth, indicating that the pressure oscillations, which may occur with the traditional nonstaggered grid system, are eliminated by the present numerical method. Contours of the vorticity are also smooth,

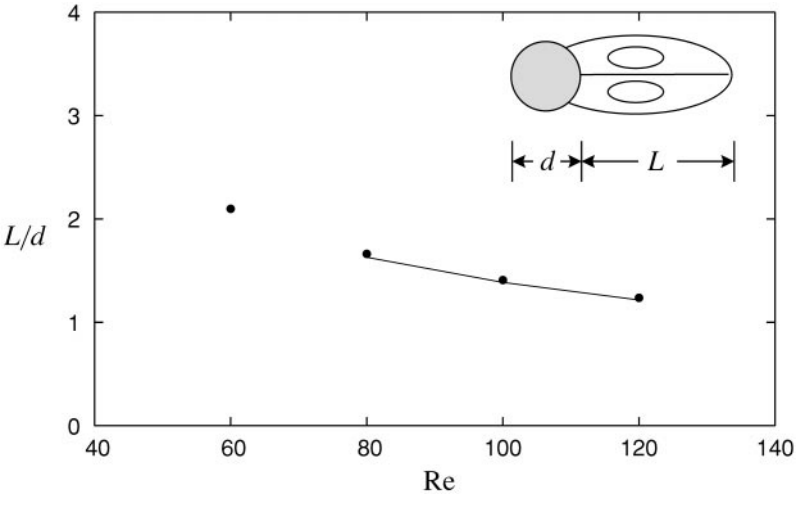

FIG. 15. Size of the time-averaged separation bubble vs Reynolds number: —, present study; ●, Park *et al.* [30].

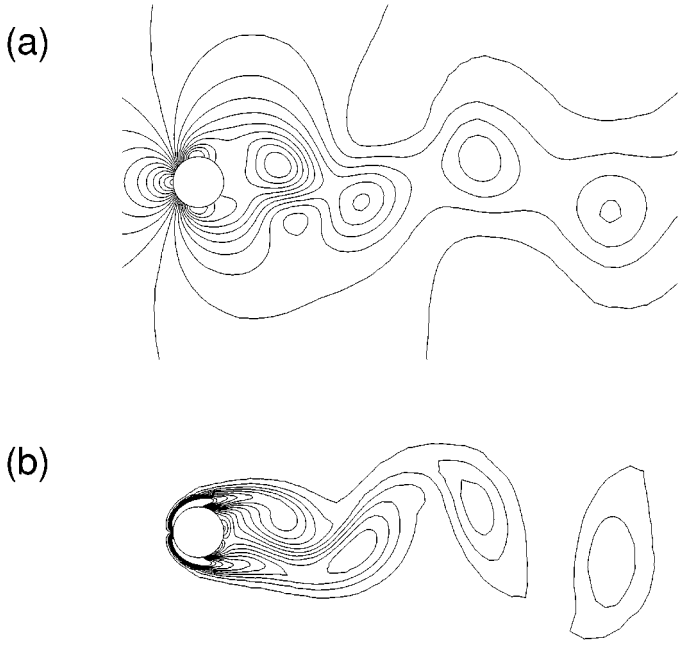

FIG. 16. Contours of instantaneous pressure and vorticity near the cylinder at $\mathrm{Re} = 100$: (a) pressure; (b) vorticity.

showing that the present interpolation method is successfully applied to the hybrid grid used in this study.

4. SUMMARY

A new second-order time-accurate finite volume method using a fully implicit fractional-step method is presented for computing unsteady incompressible flow on hybrid unstructured grids. A nonstaggered grid system is employed rather than a staggered grid system because of the simplicity and easier extension to three dimensions. In this study, the momentum interpolation method, developed by Rhie and Chow [2] and further extended by Zang *et al.* [3], is applied to unstructured grids to resolve the pressure oscillation problem occurring in a nonstaggered grid system. A second-order fully implicit time-advancement (Crank–Nicolson) scheme is used in order to remove the time step restriction and to reduce the required CPU time for complex geometries. The nonlinear equations resulting from this fully implicit scheme are linearized without losing the overall time accuracy. A finite volume method is used for the spatial derivative terms and the flow variables at the cell face are obtained using an interpolation scheme which is independent of cell shape and thus applicable to hybrid grids. The sparse matrices are solved using a biconjugate gradient stabilized method with the Jacobi preconditioner or an incomplete LU preconditioner. The present numerical method is applied to four benchmark problems and the results show good agreement with previous experimental and numerical results. It is also shown that the present method is second-order accurate in time and space, when the values at the nodal and cell-face center points are interpolated with a second-order accuracy.

ACKNOWLEDGMENTS

This work is supported by Creative Research Initiatives of the Korean Ministry of Science and Technology. The computations were performed on CRAY YMP C90 at the ETRI Supercomputer Center.

REFERENCES

1. J. Kim and P. Moin, Application of a fractional-step method to incompressible Navier–Stokes equations, *J. Comput. Phys.* **59**, 308 (1985).
2. C. M. Rhie and W. L. Chow, Numerical study of the turbulent flow past an airfoil with trailing edge separation, *AIAA J.* **21**, 1525 (1983).
3. Y. Zang, R. L. Street, and J. R. Koseff, A non-staggered grid, fractional step method for time-dependent incompressible Navier–Stokes equations in curvilinear coordinates, *J. Comput. Phys.* **114**, 18 (1994).
4. L. Davidson, A pressure correction method for unstructured meshes with arbitrary control volumes, *Int. J. Numer. Methods Fluids* **22**, 265 (1996).
5. S. R. Mathur and J. Y. Murthy, A pressure-based method for unstructured meshes, *Numer. Heat Transfer Part B* **31**, 195 (1997).
6. Y. G. Lai, An unstructured grid method for a pressure-based flow and heat transfer solver, *Numer. Heat Transfer Part B* **32**, 267 (1997).
7. M. Thomadakis and M. Leschziner, A pressure-correction method for the solution of incompressible viscous flows on unstructured grids, *Int. J. Numer. Methods Fluids* **22**, 581 (1996).
8. N. Taniguchi and T. Kobayashi, Finite volume method on the unstructured grid system, *Comput. Fluids* **19**, 287 (1991).
9. Y.-H. Hwang, Calculations of incompressible flow on a staggered triangular grid, Part I: Mathematical formulation, *Numer. Heat Transfer Part B* **27**, 323 (1995).
10. Y.-H. Hwang, Stability and accuracy analyses for the incompressible Navier-Stokes equations on the staggered triangular grid, *Numer. Heat Transfer Part B* **32**, 321 (1997).
11. M. H. Kobayashi, J. M. C. Pereira, and J. C. F. Pereira, A conservative finite-volume second-order-accurate projection method on hybrid unstructured grids, *J. Comput. Phys.* **150**, 40 (1999).
12. J. J. H. Miller and S. Wang, An exponentially fitted finite volume method for the numerical solution of 2D unsteady incompressible flow problems, *J. Comput. Phys.* **115**, 56 (1994).
13. D. Pan, C.-H. Lu, and J.-C. Cheng, Incompressible flow solution on unstructured triangular meshes, *Numer. Heat Transfer Part B* **26**, 207 (1994).
14. J. M. Weiss and W. A. Smith, Preconditioning applied to variable and constant density flows, *AIAA J.* **33**, 2050 (1995).
15. K. W. Schulz and Y. Kallinderis, Unsteady flow structure interaction for incompressible flows using deformable hybrid grids, *J. Comput. Phys.* **143**, 569 (1998).
16. A. J. Chen and Y. Kallinderis, Adaptive hybrid (prismatic-tetrahedral) grid for incompressible flows, *Int. J. Numer. Methods Fluids* **26**, 1085 (1998).
17. C. A. Hall, J. C. Cavendish, and W. H. Frey, The dual variable method for solving fluid flow difference equations on delaunay triangulations, *Comput. Fluids* **20**, 145 (1991).
18. S. H. Chou, Analysis and convergence of a covolume method for the generalized Stokes problem, *Math. Comput.* **66**, 85 (1997).
19. B. Perot, Conservation properties of unstructured staggered mesh schemes, *J. Comput. Phys.*, to appear.
20. R. A. Nicolaides, Direct discretization of planar div-curl problems, *SIAM J. Numer. Anal.* **29**, 32 (1992).
21. H. Le and P. Moin, An improvement of fractional step methods for the incompressible Navier-Stokes equations, *J. Comput. Phys.* **92**, 369 (1991).
22. H. Choi and P. Moin, Effects of the computational time step on numerical solutions of turbulent flow, *J. Comput. Phys.* **113**, 1 (1994).
23. R. M. Beam and R. F. Warming, An implicit factored scheme for the compressible Navier-Stokes equations, *AIAA J.* **16**, 393 (1978).

24. H. Choi, P. Moin, and J. Kim, Direct numerical simulation of turbulent flow over riblets, *J. Fluid Mech.* **255**, 503 (1993).
25. J. F. Thompson, Z. U. A. Warsi, and C. W. Mastin, *Numerical Grid Generation* (Elsevier Science, New York, 1985).
26. R. Barrett, M. Berry, T. F. Chan, J. Demmel, J. M. Donato, J. Dongarra, V. Eijkhout, R. Pozo, C. Romine, and H. V. der Vorst, *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods* (SIAM, Philadelphia, 1994).
27. U. Ghia, K. N. Ghia, and C. T. Shin, High-Re solutions for incompressible flow using the Navier-Stokes equations and a multigrid method, *J. Comput. Phys.* **48**, 387 (1982).
28. L. L. Pauley, P. Moin, and W. C. Reynolds, The structure of two-dimensional separation, *J. Fluid Mech.* **220**, 397 (1990).
29. B. F. Armaly, F. Durst, J. C. F. Pereira, and B. Schönung, Experimental and theoretical investigation of backward-facing step flow, *J. Fluid Mech.* **127**, 473 (1983).
30. J. Park, K. Kwon, and H. Choi, Numerical solutions of flow past a circular cylinder at Reynolds numbers up to 160, *KSME Int. J.* **12**, 1200 (1998).
31. U. Fey, M. König, and H. Eckelmann, A new Strouhal-Reynolds-number relationship for the circular cylinder in the range $47 < \mathrm{Re} < 2 \times 10^5$, *Phys. Fluids* **10**, 1547 (1998).
32. S. Pirzadeh, Structured background grids for generation of unstructured grids by advancing-front method, *AIAA J.* **31**, 257 (1993).
33. C. H. K. Williamson, Oblique and parallel modes of vortex shedding in the wake of a circular cylinder at low Reynolds numbers, *J. Fluid Mech.* **206**, 579 (1989).