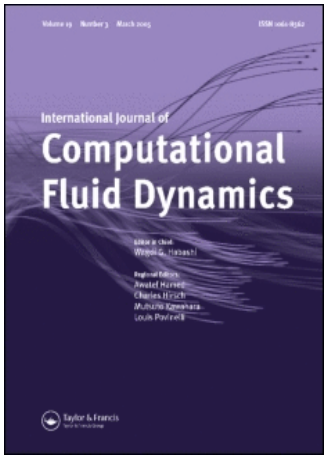

International Journal of Computational Fluid Dynamics
Publication details, including instructions for authors and subscription information:
http://www.informaworld.com/smpp/title~content=t713455064

Computation of viscous incompressible flow using pressure correction method on unstructured Chimera grid

To cite this Article: , 'Computation of viscous incompressible flow using pressure correction method on unstructured Chimera grid', International Journal of Computational Fluid Dynamics, 20:9, 637 - 650
xxxx:journal To link to this article: DOI: 10.1080/10618560601140094
URL: http://dx.doi.org/10.1080/10618560601140094

International Journal of Computational Fluid Dynamics, Vol. 20, No. 9, October–November 2006, 637–650

Computation of viscous incompressible flow using pressure correction method on unstructured Chimera grid

XING ZHANG*

The State Key Laboratory of Nonlinear Mechanics, Institute of Mechanics, Chinese Academy of Sciences, Beijing 100080, P. R. China

(Received 15 February 2005; revised in final form 11 June 2006)

In this paper, a pressure correction algorithm for computing incompressible flows is modified and implemented on unstructured Chimera grid. Schwarz method is used to couple the solutions of different sub-domains. A new interpolation to ensure consistency between primary variables and auxiliary variables is proposed. Other important issues such as global mass conservation and order of accuracy in the interpolations are also discussed. Two numerical simulations are successfully performed. They include one steady case, the lid-driven cavity and one unsteady case, the flow around a circular cylinder. The results demonstrate a very good performance of the proposed scheme on unstructured Chimera grids. It prevents the decoupling of pressure field in the overlapping region and requires only little modification to the existing unstructured Navier–Stokes (NS) solver. The numerical experiments show the reliability and potential of this method in applying to practical problems.

Keywords: Incompressible flow; Chimera grid; Unstructured grid; Schwarz method

1. Introduction

Flows around complex geometry and around multiple bodies with relative motions are two major challenges in computational fluid dynamics (CFD). To tackle theses problems, unstructured grid methods have attracted more attention in the past 20 years. Unstructured grids facilitate easy automatic grid generation in complex domain and allow for easy implementation of solution-adaptive algorithms. Such methods however can be memory and computational time intensive and less amenable (compare with structured mesh) in implementing a high order discretization. In problems that involve relative motion, an automatic mesh-moving technique is needed in the framework of unstructured grid. To prevent the mesh quality from deteriorating, some measures have to be taken, such as mesh smoothing, swapping or even partial re-meshing. These make it more computational intensive and sometime also very tedious to realize. A comprehensive review on the development of unstructured grid techniques is presented in Mavriplis (1997). More detailed information on mathematical formulations and implementation are given in Barth (1994). For dynamic unstructured mesh method, please refer to Slone *et al.* (2002) and Perot and Nallapati (2003).

One alternative to the unstructured grid method is the Chimera (or overset) grid method. It was originally designed for use in structured mesh with curvilinear coordinates. In this method, the computational domain is divided into sub-domains that overlap each other. Grids for each sub-domain can be generated separately. This makes the mesh generation less time-consuming and much simpler. Governing equations can also be solved independently and information is transferred across the interior boundaries to couple the solutions of each sub-domain. Compared with the unstructured grid counterpart, it has several advantages: (1) keeping structured data structure makes it easy to implement a high order scheme; (2) the treatment of body motion is straightforward and of very little computational burden. The major difficulty for Chimera grid algorithm is the specification of boundary conditions on the interior boundaries. Especially for unsteady flows, such information transferring must ensure the vortices to cross the overlapping region with minimum distortion. Abundant literatures have been published recently in the study of this method, e.g. Pan and Damodaran (2002) and Tang *et al.* (2003).

*Tel.: + 86-10-62545533. Ext. 3185. Fax: + 86-10-62579511. Email: zhangx@lnm.imech.ac.cn

International Journal of Computational Fluid Dynamics
ISSN 1061-8562 print/ISSN 1029-0257 online © 2006 Taylor & Francis
http://www.tandf.co.uk/journals
DOI: 10.1080/10618560601140094

It seems that the concept of combining unstructured grid with Chimera grid is very counter-intuitive. A first impression is that this combination at least cannot incorporate the best aspects of both methods. First, unstructured grid makes it difficult to enhance accuracy by increasing the order of discretization. Second, having one capability (either Chimera or unstructured grid) is enough for handling complex geometries; having both of them within one grid is only a waste. However, after deep investigations one may find that it is not necessarily true. An unstructured Chimera grid is not only possible but also of great potential in tackling practical problems. If relative motion is in presence and at the same time, the geometry of the computational model is very complex, the number of grids required in a traditional (structured) Chimera method will become fairly large. The use of unstructured Chimera grid reduces the number of sub-domains significantly. It can also greatly extend the capability of an existing unstructured solver to solve moving body problems with very little code-developing efforts. A successful example is presented in Nakahashi *et al.* (2000) and Togashi *et al.* (2001), in which a realistic CFD simulation is performed of an insect in flight. In their simulation, the unstructured grid allows a precise representation of an insect including antennas, legs and a sting. The Chimera grid method is used to treat the flapping wing motion including translation and rotation.

Generally speaking however, the use of Chimera grid in the framework of unstructured grid is relatively unexplored. Only few literatures are found in this area. In the work aforementioned, the authors solve *compressible* NS equation by finite volume method (FVM). In another example Maruoka (2003), incompressible NS equation is solved to compute flow around a rotating body using unstructured Chimera grid. Their computation is based on *Finite Element Method* (FEM). Similar study is also reported in Houzeaux and Codina (2004), which is also based on FEM.

In the CFD community, *finite volume* based *pressure correction* scheme has become a mature method for solving *incompressible* flow for a long time. The core of many widely used packages (both commercial and in-house) that targets at industrial applications is an *unstructured* solver utilizing this technique. However, the details in implementing the Chimera method on such a solver are rarely reported in academic papers at least to the knowledge of this author.

In this paper, a well known pressure correction scheme—SIMPLEC, is modified and implemented on an unstructured Chimera mesh. The primary purpose of this paper is not to simulate realistic flow around complex geometry with motions. On the contrary, only stationary and simple geometries are chosen for the numerical tests. One objective of this paper is to clarify some implementation issues, such as the specification of boundary conditions on the interior boundaries, global mass conservation and order of accuracy regarding the interpolation. Another purpose of this study is to demonstrate the credibility and potential of this method in conducting both steady and unsteady CFD simulations. In our future plan, this method is to be used to study more complex problems such as flapping wing, rotating machinery, etc.

This paper is organized into four sections. In section 2, a summary of the pressure correction method on unstructured grid is presented. The topics of temporal and spatial discretization and velocity–pressure coupling will be covered. It is then followed by some implementation issues on unstructured Chimera grids. Two numerical examples are presented in section 3 to demonstrate the potential of this method as a powerful tool to solve real problems. Conclusions are given in the last section.

2. Numerical methodologies

2.1 Navier–Stokes equations in an integral form

We consider a two-dimensional unsteady laminar flow in this study. The continuity and momentum equations used in the simulations can be written in an integral form as:

$$\int_S \mathbf{v}\cdot\mathbf{n}\,\mathrm{d}S = 0 \tag{1}$$

$$\frac{\partial}{\partial t}\int_V \mathbf{v}\,\mathrm{d}V + \int_S \left[\mathbf{v}(\mathbf{v}\cdot\mathbf{n}) - \frac{1}{Re}\left(\frac{\partial \mathbf{v}}{\partial n}\right)\right]\mathrm{d}S = -\int_{\partial S} p\mathbf{n}\,\mathrm{d}S \tag{2}$$

where t is time; $\mathbf{v}$ is the velocity vector and p is the pressure. dS and dV are the surface area and volume of a control volume (CV), respectively. $\mathbf{n}$ denotes an unit out-normal vector on the surface CVs.

Variables in equations (1) and (2) are non-dimensionalized by a characteristic length L, a characteristic velocity U_∞ and the fluid density ρ_∞, i.e.

$$\mathbf{x} = \mathbf{x}^*/L; \quad t = t^* U_\infty/L; \quad \mathbf{v} = \mathbf{v}^*/U_\infty; \quad p = p^*/(\rho_\infty U_\infty^2) \tag{3}$$

where the variables with an asterisk denote the original (dimensional) variables. The Reynolds number in equation (2) is defined as,

$$Re = \frac{U_\infty L}{\nu} \tag{4}$$

where ν is the kinetic viscosity of the fluid.

2.2 Temporal and spatial discretization

To solve the Navier–stokes (NS) equation numerically, the method described in Demirdzic and Muzaferija (1995) and Demirdzic *et al.* (1997) is followed with some modifications. This method was originally implemented on a block-structured mesh and later generalized to

unstructured mesh. The discretization procedure will only be summarized briefly here; for the details, please refer to those papers. In the current implementation, a SIMPLEC algorithm is used to couple the pressure with the velocity. A second-order upwind scheme is used for the discretization of convective term and the Crank–Nicholson scheme (second order in time) is used for the temporal advancement.

In order to obtain a discrete solution of equations (1) and (2), the computational domain is discretized into finite number of contiguous CVs or cells. Cells with arbitrary shapes are permitted in this method. Triangular mesh or hybrid mesh that consists of triangular and quadrilateral cells will be used in this study. All dependent variables are stored at the centroid of cells, i.e. a collocated arrangement is adopted.

To explain the discretization procedure of the momentum equation (2) term by term, we first reformulate it into a transport equation for an arbitrary variable ϕ.

$$\underbrace{\frac{\partial}{\partial t}\int_V \phi \,\mathrm{d}V}_{\text{Transient term}} + \underbrace{\int_S [\phi \mathbf{v}}_{\text{Convective term}} \underbrace{- \Gamma_\phi \nabla \phi]\cdot\mathbf{n}\,\mathrm{d}S}_{\text{Diffusive term}} = \underbrace{\int_S Q_\mathrm{f}\,\mathrm{d}S}_{\text{Surface source}} \tag{5}$$

where ϕ represents the velocity components $\mathbf{v}_i$ ($i = 1, 2$) and Γ_ϕ is the coefficient of diffusivity ($1/Re$).

This time-dependent equation is discretized by the Crank–Nicholson scheme (second order in time).

$$\frac{\partial}{\partial t}\int_V \phi \,\mathrm{d}V \approx \frac{1}{\Delta t}[(\phi V)^{n+1} - (\phi V)^n] = h(\phi^{n+(1/2)})$$
$$\approx \frac{1}{2}[h(\phi^{n+1}) + h(\phi^n)] \tag{6}$$

where "n" and "$n-1$" are the time step counters; h denotes the summation of all terms in equation (5) excluding the transient term.

The convective term is discretized as

$$\int_S \phi \mathbf{v}\cdot\mathbf{n}\,\mathrm{d}S \approx \sum_j \dot{m}_j \tilde{\phi}_j \tag{7}$$

where m_j is the volume flux across face j. It is computed by

$$\dot{m}_j = A_j \tilde{\mathbf{v}}_j \cdot \mathbf{n}_j \tag{8}$$

where A_j is the area of face j; $\tilde{\mathbf{v}}_j$ is the face velocity; $\tilde{\phi}_j$ is the variable interpolated to face j by using a blended scheme,

$$\tilde{\phi}_j = \phi_j^{(1)} + \gamma_\phi\left(\phi_j^{(2)} - \phi_j^{(1)}\right) \tag{9}$$

where the superscript "(1)" and "(2)" denote first order and second order interpolation, respectively. The first order interpolation is just a simple "upwind" scheme. In the second order scheme, we first use the gradient of variable ϕ and Taylor expansion to evaluate the value of ϕ on the face centers from either side, then the one from the "upwind" direction is chosen as the value for that particular face. The gradient of ϕ is constructed using a linear least square approach. γ_ϕ in equation (9) is a blending factor which is set to 1.0 in this paper, i.e. a second order upwind scheme is implemented through a "deferred correction" in all numerical examples of this paper.

The diffusive term is discretized as

$$\int_{S_j} -\Gamma_\phi \nabla\phi\cdot\mathbf{n}\,\mathrm{d}S \approx \sum_j -\Gamma_{\phi j}\frac{A_j}{L_j}\big((\phi_{P_j} - \phi_{P_0})$$
$$+[(\nabla\phi)_{P_j}\cdot\boldsymbol{\tau}_1 - (\nabla\phi)_{P_0}\cdot\boldsymbol{\tau}_2]\Big) \tag{10}$$

where L_j is the distance from the center of cell P_0 to that of cell P_j projected to the normal direction of face j. $\boldsymbol{\tau}_1$ and $\boldsymbol{\tau}_2$ are two vectors in the tangential direction of face j (figure 1). The first term on the RHS of equation (10) is the "normal diffusion" and the second term is the "cross diffusion" which is a correction for non-orthogonal meshes.

The pressure gradient term is treated as a surface source.

$$\int_S Q_\mathrm{f}\,\mathrm{d}S \approx \sum_j -p_\mathrm{f}A_\mathrm{f} \tag{11}$$

A linear system is obtained as a result of the discretization of the momentum equation.

$$a_{P0}^c\phi_{p0} = \sum_j^{\text{nb\_cells}} a_{P0}^j\phi_{P0}^j + b_{p0} \tag{12}$$

where superscript "C" denotes the diagonal elements of the coefficient matrix (related to cell $P0$) and "j" denotes the non-diagonal elements of the matrix (related to the neighboring cells of $P0$). "nb_cells" denotes that the summation is done on all neighboring cells of $P0$. The contributions to the coefficient matrix are: the mass matrix, "upwind" part of the convective term and the

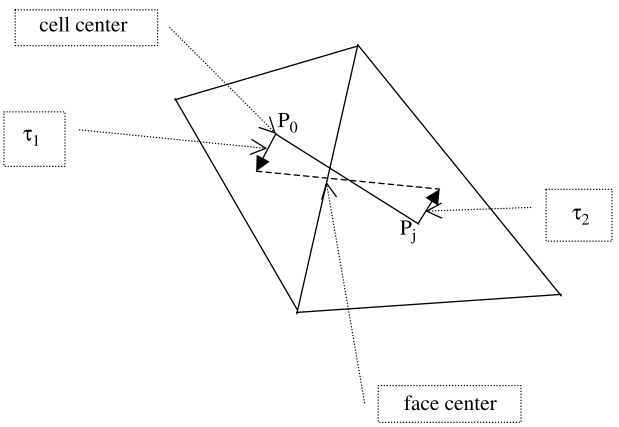

Figure 1. Stencil used in the discretization. The dashed line is perpendicular to the cell face. It is not necessarily parallel to the line connecting cell $P0$ and cell $P1$.

"normal diffusion". The source term b_{p0} in equation (12) has four contributions, the pressure gradient, the "cross diffusion" and the departures of the convective flux from the upwind differencing (deferred correction).

In the SIMPLEC algorithm, the pressure correction equation is derived from the continuity equation (1).

$$\sum_{j}^{\text{cell\_faces}} \overline{\left(\frac{1}{a'^{C}_{P0}}\right)}_{j} (\nabla p')_j \cdot A_j \mathbf{n}_j = \sum_{j}^{\text{cell\_faces}} \dot{m}_j \qquad (13)$$

where p' is the pressure correction; "—" stands for the arithmetic averaging from cell to face. "cell_faces" indicates that the summation is done on all faces that belong to cell $P0$. The coefficient on the left-hand side of equation (13) is defined as

$$a'^{C}_{P0} = a^{C}_{P0} - \sum_{j}^{\text{nb\_cells}} a^{j}_{P0} \qquad (14)$$

where the summation is done on all neighboring cells. A fully discretized form of equation of equation (13) can be written in a generic form very similar to equation (12). The Laplacian operator on the left-hand side of equation (13) is also treated similarly as that in the diffusive term of the momentum equation. Correction for the mesh non-orthogonality is also considered.

After the pressure correction p' obtained from equation (13), the pressure and velocity are corrected by

$$\begin{aligned} p^m &= p^{m-1} + p'^m \\ \mathbf{v}^m &= \mathbf{v}^{m-1} - \frac{1}{a'^{C}_{P0}} \sum_{j}^{\text{cell\_faces}} p'_j A_j \mathbf{n}_j \end{aligned} \qquad (15)$$

where m is the pressure correction loop counter at time step n. After this correction, the coefficient matrix and source term in equation (12) are computed using the updated p and $\mathbf{v}$. A new velocity is then obtained by solving equation (12) again. This velocity is substituted into equation (13) to compute a new pressure correction. The pressure correction procedure is repeated until the convergence criterion is satisfied. For the momentum equation (12) which can be expressed in a matrix form as $A\mathbf{x} = b$, the convergence criterion is,

$$\frac{\|b - A\mathbf{x}\|}{\max(\|\mathbf{v}_1\|, \|\mathbf{v}_2\|)} \leq 10^{-8} \qquad (16)$$

For the pressure correction equation, the convergence criterion is,

$$\left(\frac{1}{V_{\text{cell}}} \left| \sum_{j}^{\text{cell\_faces}} \dot{m}_j \right| \right)_{\max} \leq 10^{-4} \qquad (17)$$

The selection of criteria equations (16) and (17) is based on experiences and parameters found in published literatures. Actually, they are tight enough to ensure a converged solution.

Attention should be paid to the face velocity $\tilde{\mathbf{v}}_j$ that is used to calculate the volume flux. This velocity is not approximated by an arithmetic average of the values in the neighboring cells. Instead, a Rhie–Chow interpolation, which introduces some dependency on the pressure, is used. More details of this interpolation can be found in Demirdzic and Muzaferija (1995).

Iterative methods are implemented to solve the algebraic equation results from the discretization. For the momentum equation, a generalized conjugate residue (GCR) solver is used whereas for the pressure correction equation, a conjugate gradient (CG) solver is used. A Jacobi pre-conditioner is applied in both solvers to speed up the convergence rate.

The audience has to be very careful about the aforementioned order of accuracy in the discretization. The so called "second order" in space (as in the convective term) is only valid on a uniformed mesh (equilateral triangles), which is not realistic in practical problems. Furthermore, non-uniform and non-orthogonal mesh will also reduce the order of accuracy in diffusive term (which is also second order on equilateral triangles). The resulting property of the numerical scheme is more complicated in terms of order of accuracy because the split error in the pressure correction method will also come into the picture. This author believes that (but with no rigorous proof) for the velocity field, the spatial accuracy is between first and second order and temporal accuracy is of second order. As to the conservation property, FVM is built to conserve mass and momentum. However, in reality, mass and momentum are only conserved *approximately* (and *not* to machine precision). The reason behind this is that for the pressure correction method such as SIMPLEC, it is very hard to satisfy the divergence free condition exactly (to machine precision).

2.3 Boundary conditions

To solve the NS equation, boundary and initial conditions have to be provided. Initial conditions are easy to specify. For incompressible fluid, divergence free condition is the only constraint. In this paper, a const velocity and pressure field is used as the initial condition. Boundary conditions are more complicated and sometimes can be the key factor to an accurate numerical simulation. Some commonly used boundary conditions are summarized as follows. These conditions only include the ones specified on a physical or computationally truncated boundary, whereas those on the interior boundary that are created by the use of Chimera grid will be discussed in the next section.

a. Boundary where velocity is prescribed

On this type of boundary, the normal velocity component is specified. One could either specify the tangential component of velocity, or its gradient (figure 2). The normal gradient of pressure correction

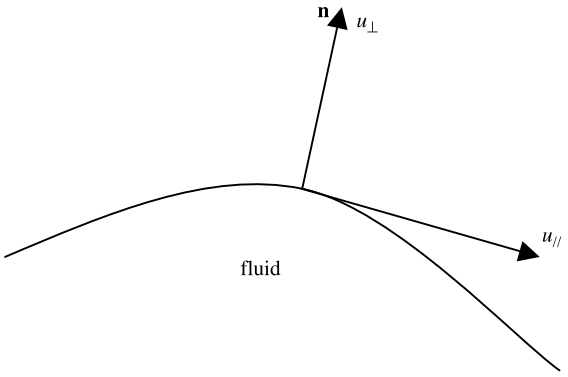

Figure 2. The boundary where velocity is prescribed.

is set to zero. This condition is often used on inlets, solid walls or symmetric centerlines. It can be expressed in a formula as,

$$\mathbf{v}_n = f(x,y) \quad \mathbf{v}_{//} = g(x,y) \quad \text{or}$$
$$\frac{\partial \mathbf{v}_{//}}{\partial n} = h(x,y) \quad \frac{\partial p'}{\partial n} = 0 \tag{18}$$

b. Boundary where pressure is prescribed

On this type of boundary, pressure is set to a constant (usually zero). In order to satisfy the continuity constraint, the velocity is corrected using the gradient of pressure correction p' in a manner similar to equation (15). The implementation of this boundary condition is addressed in detail in Ferziger and Peric (1996). It is appropriate to apply such condition on an outlet that is sufficiently far from the inlet. Since the pressure is never corrected on this boundary, the following condition is specified for pressure correction.

$$p' = 0 \tag{19}$$

c. Convective boundary

This type of condition is exclusively designed for outlets. The purpose of this boundary condition is to minimize the distortion of vortices when passing through such boundaries. Usually, velocity components are obtained by a simplified one-dimensional NS equation. Pressure gradient is also set to zero on this type of boundary, same as on the boundary of the first category. Since global mass conservation is not necessarily guaranteed by applying such condition, usually the velocities are scaled to satisfy the conservation law globally. The convective boundary condition can be expressed in a mathematical formula as,

$$\frac{\partial \phi}{\partial t} + U_{\text{conv}} \frac{\partial \phi}{\partial n} = 0 \quad \frac{\partial p'}{\partial n} = 0 \tag{20}$$

where ϕ represents the velocity components $\mathbf{v}_i$ ($i = 1, 2$); U_{conv} is the convective velocity at the outlet. The details in the implementation of such boundary condition including the choice of convective velocity can be found in Ferziger and Peric (1996).

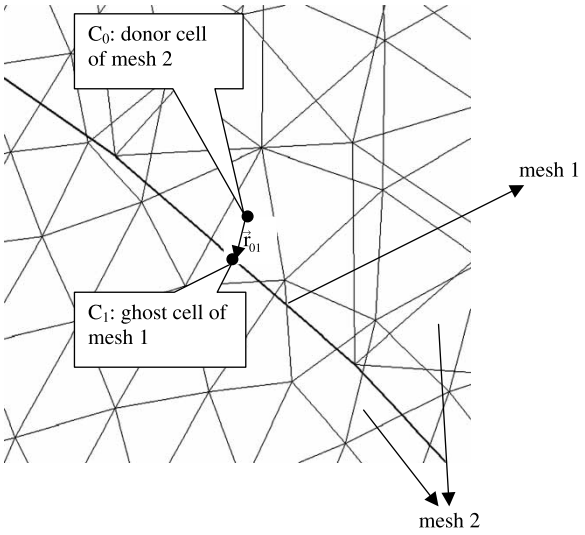

Figure 3. The interior boundary where data is transferred. *C*0: donor cell of mesh 2; *C*1: ghost cell of mesh 1.

2.4 Implementation on unstructured Chimera grid

2.4.1 Building inter-grid links. In order to implement the pressure correction scheme on a Chimera grid, information transfer is needed between sub-domains. To establish the communications between them, inter-grid links has to be built after the grids are generated separately. The "link building" is to identify the donor cell for each ghost cell on the interior boundaries. The donor cells and the ghost cells are from different component grid (figure 3). Since no mesh motion is involved in this paper, a repetitive "hole cutting" procedure is not needed. In this paper, a "neighbor to neighbor" searching scheme is used to build the inter-grid links. This algorithm is very efficient compared with a complete search. For the details of this method, please refer to Nakahashi *et al.* (2000).

2.4.2 Data transfer on the interior boundaries. The key step in implementing the pressure correction scheme on a Chimera grid is the treatment of interior boundaries. These boundaries are neither physical ones nor artificially truncated for computational purpose, thus the boundary condition on them is quite different from that described in section 2.3. A Schwarz alternating procedure is followed to couple the solutions from individual component grids. The Schwarz method states that the information between sub-domains that overlapping each other are exchanged in the form of Dirichlet condition on the interior boundaries. The data are obtained through an interpolation from the solution of the sub-domain that contains these boundaries.

For one PDE, the implementation of Schwarz procedure is straightforward. However, the incompressible NS is a system of PDEs that consists of momentum and continuity equation. Moreover, the pressure correction algorithm

described in section 2.2 has introduced more auxiliary variables besides the primary ones (velocity and pressure), such as the pressure correction p' and mass flux $\dot{m}_f$, etc. This makes the situation more complicated. In order to do it in a consistent manner, one has to be very careful while interpolating auxiliary variables. Auxiliary variables are interpolated and exchanged between sub-domains when necessary, but these exchanges of information could lead to the decoupling of primary variables (velocity and pressure).

Since the data transfer between sub-domains are done iteratively within one time step. This iterative procedure can be united with the SIMPLEC iteration where the velocity–pressure coupling is done. The major steps in this algorithm are summarized as follows.

(1) Read mesh connectivity, compute geometry and inter-grid links for later use.
(2) Start the computing from an initial guess of pressure and velocity.
(3) Solve the momentum equation based on the guessed pressure field.

The velocity values on the interior boundary (ghost cells) are obtained by interpolating from their donor cells. This interpolation can be expressed in a mathematical formula as (figure 3),

$$\phi_i^*|_{C_1} = \phi_i|_{C_0} + (\nabla\phi_i)|_{C_0}\cdot\vec{r}_{01} \tag{21}$$

where $i = 1$, 2 and ϕ_i denotes the two velocity components. The variable with an asterisk denotes the one that is updated for use in the next iteration.

(4) Solve the pressure correction equation.

The pressure correction values on the interior boundary (ghost cells) are determined by the following two-steps.

a. A temporary pressure values on these cells are obtained by interpolating from their donor cells.
b. Subtract the values of current pressure from the temporary pressure and the result is treated as a Dirichlet condition prescribed on the ghost cells for the pressure correction equation.

This interpolation can be expressed in a mathematical formula as,

$$\begin{aligned} p^{\text{temp}}|_{C_1} &= p|_{C_0} + (\nabla p)_{C_0}\cdot\vec{r}_{01} \\ p'^*|_{C_1} &= p^{\text{temp}}|_{C_1} - p|_{C_1} \end{aligned} \tag{22}$$

where the variable with an asterisk denotes the one which has been updated for use in the next iteration.

(5) Correct velocity and pressure, respectively.
(6) Check if the convergence criteria are met. If not, replace the initial value of pressure and velocity with the current one and go to (3); if yes, then exit from this iteration and advance to the next time step.
(7) Repeat (1)–(6) until the terminating time is reached.

In step (4), the transfer of velocity $\mathbf{v}$ between sub-domains is done by using a direct interpolation. However, the situation is more complicated as to the transfer of pressure correction p'. Through numerical test, it is found that such interpolation only works well for steady flows (such as the lid-driven cavity). It will lead to a decoupling of pressure in the overlapping region when unsteady flows (such as the vortex shedding) are computed. A detailed discussion concerning this phenomenon will be presented in the next section.

A remedy to this is proposed in step (4) and is verified in the numerical examples. The philosophy behind this method is to "hook" the pressures of different sub-domains to each other more tightly. A pressure correction value is assigned to the interior boundary (on ghost cells) such that after this correction, the pressure on that boundary will reach the value that is obtained by interpolation (using the current pressure and pressure gradient value) from its donor cell.

For Chimera grid method, global conservation is an important issue that attracts some attentions. Especially for incompressible flows, global mass conservation is a necessary condition for the existence and uniqueness of a smooth pressure field. Thus it is crucial for the design of any numerical schemes in solving incompressible NS equations. Since the treatment of the interior boundary condition by using interpolations is non-conservative in nature, it seems that some corrections are needed. However, after further investigation, it is found that such correction is not necessary in the modified pressure correction algorithm proposed in this paper. Our treatment of the interior boundary is very similar to the boundaries where pressure is prescribed (category b in section 2.3). The only difference is that pressure correction is not zero on them; instead pressure correction is set to some value in order for the pressure to "hook" to the solution of the sub-main in which this boundary is located. On these boundaries, the mass flux $\dot{m}_f$ is corrected iteratively until the convergence criterion is met. It is seen that in this algorithm mass conservation is satisfied approximately both locally and globally on these boundaries (if the convergence criterion is stringent enough). The only remaining issue is that whether the velocity computed by interpolating is compatible with this mass flux. From our experiences in the numerical test conducted, the answer is yes. Nonphysical "boundary layers" are not observed near these interior boundaries.

The order of accuracy in the interpolation is another important issue for any numerical methods on Chimera grid. For the traditional Chimera methods (such as those in Pan and Damodaran 2002, Tang *et al.* 2003), it is very common in these literatures where the order of accuracy in the interpolation is chosen to be of one order higher than that in spatial discretization of the flow solver. In this paper, however, these two are of the same order (second order).

There are several reasons for our choice. First, the cost of quadratic construction is much higher in terms of CPU time and memory. In the framework of FVM, it requires the connectivity information of "second-level" neighbors. The matrix which is inverted on each cell has a larger size (3 by 3 for linear and 5 by 5 for quadratic construction in two-dimensional problems). Second, for unstructured mesh, it is very easy to refine the mesh locally near the overlapping region with only a minor increase in cell number. Third, through numerical tests, it is shown that no significant improvement on the quality of solution in the overlapping region is observed by simply replacing the linear reconstruction with a quadratic reconstruction in the interpolation of velocity and pressure correction (equations (21) and (22)).

3. Numerical experiments

3.1 Two-dimensional lid-driven cavity

The first numerical example is a two-dimensional lid-driven cavity. The Reynolds number for this flow is based on the lid width and lid driven velocity. Simulation is performed at $Re = 100$. A square domain of dimension 1×1 is decomposed into two overlapping rectangular sub-domains (Ω_1 and Ω_2) each with the dimensions of 1×0.625 (figure 4). These two domains are meshed with triangular cells independently with cell number of 2244 and 2250, respectively. The longer side of the rectangular domain is partitioned into 40 cells. The two overlapping meshes are placed together and presented in figure 5(a). It is seen from the figure that there are 10 grid points across the overlapping region.

On the topside of the domain Ω_1, velocity of $\mathbf{v} = (1,0)$ is prescribed, whereas on other boundaries (excluding the two interior boundaries Γ_{21} and Γ_{12}) of domain Ω_1 and Ω_2, no-slip condition with velocity $\mathbf{v} = (0,0)$ is prescribed. As that mentioned in section 2.3, a zero gradient condition is applied to pressure correction p' on all boundaries (excluding Γ_{21} and Γ_{12}).

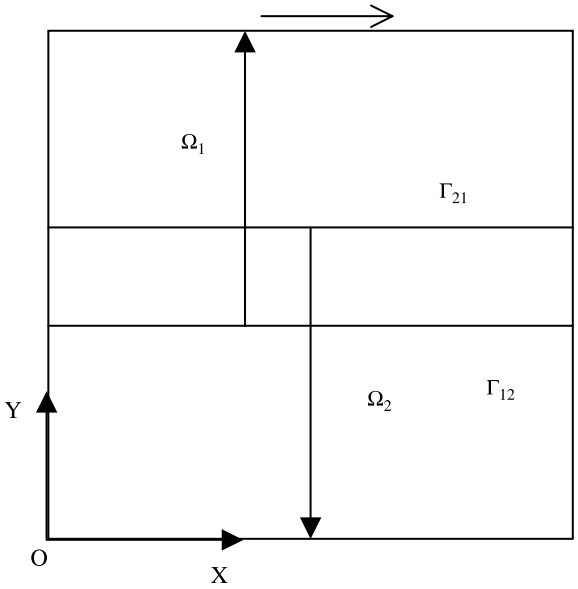

Figure 4. Domain decomposition for the simulation of lid-driven cavity.

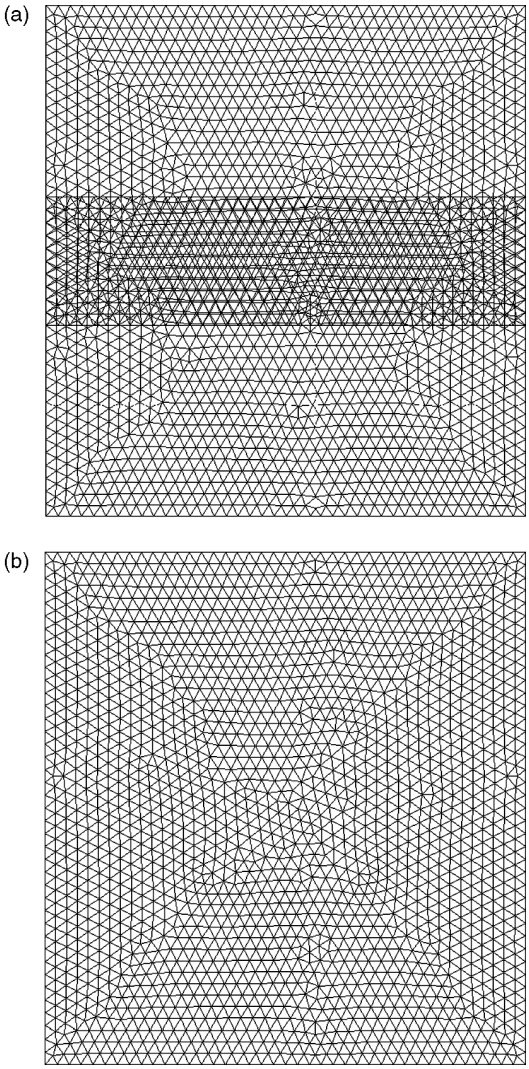

Figure 5. Grids used in the lid-driven cavity problem. (a) Chimera grid; (b) single grid.

A steady solution is achieved by a time advancing procedure. When the location of the prime vortex ceases to move any more, the flow field is considered to have reached the steady state. For this mesh resolution, approximately 10 dimensionless time units are needed to obtain a steady solution if the computation is started from scratch (i.e. $\mathbf{v} = (1, 0)$ and $p = 0$ everywhere).

For the purpose of comparison, a single triangular mesh consists of 3602 cells is generated. The partitions on each side of the cavity is also 40, thus the resolution of this mesh is very close to the Chimera mesh (figure 5(b)). The same driven cavity problem is also computed on this mesh.

3.1.1 Pressure field. The pressure contours are plotted in figure 6. Figure 6(a) is the pressure computed using the

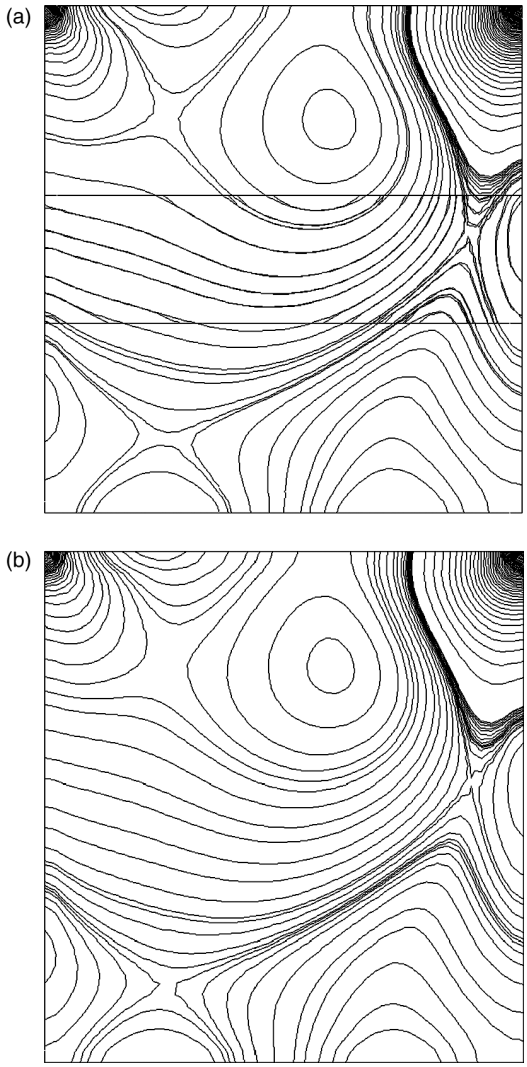

Figure 6. Pressure contours in the lid-driven cavity at $Re = 100$. (a) Chimera grid; (b) single grid.

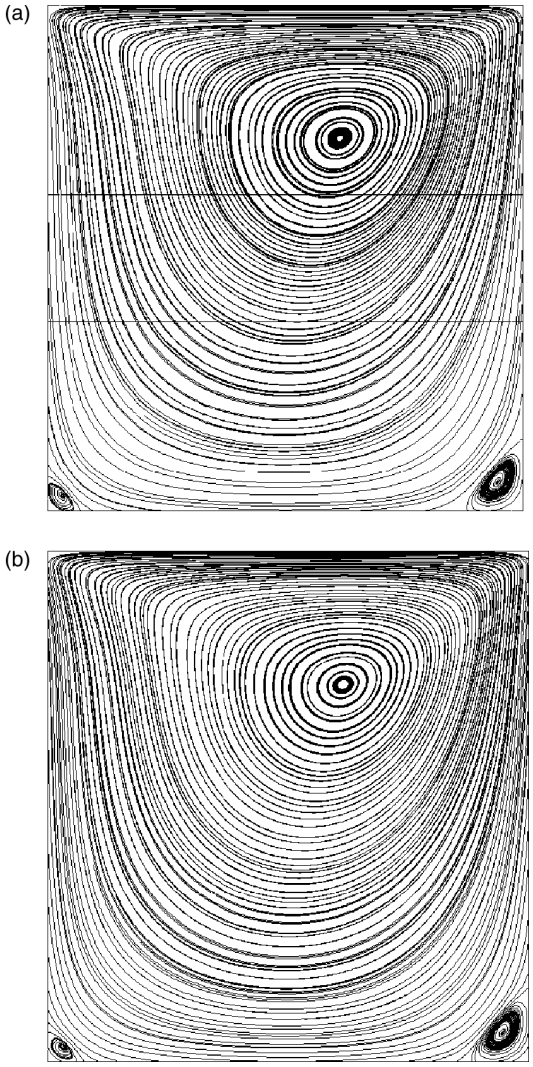

Figure 7. Streamlines in the lid-driven cavity at $Re = 100$. (a) Chimera grid; (b) single grid.

Chimera grid and figure 6(b) is the one computed using the single grid. It is seen that on the one using the Chimera grid, no spurious oscillation has occurred in the overlapping region and there is a good collapse of contours from Ω_1 and Ω_2. The nearly smooth pressure field is also in good agreement with the one computed using a single grid.

Our modified interpolation scheme (22) is used in the simulation. However, further experiments show that for a steady flow like the lid-driven cavity, identical and correct answer can also be achieved even a direct interpolation of pressure correction is used.

3.1.2 Streamlines, velocity and vorticity. The streamlines are plotted in figure 7. Figure 7(a) is the result using the Chimera grid and figure 7(b) is the result using the single grid. It is seen that they are almost identical. Three vortices are visible in the figures; one primary vortex in the middle and two smaller ones at the bottom corners. The location of the primary vortex is at (0.62, 0.74) in our computation. It is in good agreement with the prediction of (0.62, 0.73) by Ghia *et al.* (1982).

The contours of horizontal and vertical velocity component and vorticity are presented in figures 8–10, respectively. It is seen that almost identical results have been obtained by using the Chimera grid and the single grid. For the results of the Chimera grid, all contours in the overlapping region are quite smooth.

3.1.3 Velocity profiles. Two velocity profiles are presented in figure 11. Figure 11(a) is the x-component of velocity on the vertical centerline and figure 11(b) is the y-component of velocity on the horizontal centerline. For comparison, the data from Ghia *et al.* (1982) is also plotted together with the result using the single grid. It is seen that the prediction of the Chimera grid agrees well with that of the single grid and both results (Chimera and single grid) match well with that from Ghia *et al.* (1982).

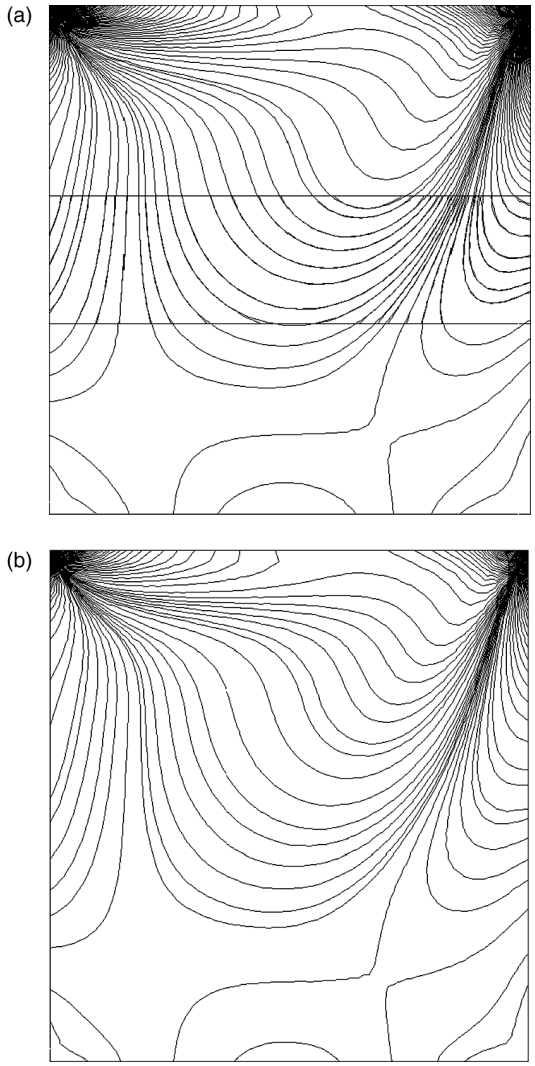

Figure 8. Vorticity contours in the lid-driven cavity at $Re = 100$. (a) Chimera grid; (b) single grid.

3.1.4 Global mass conservation on the interior boundaries. The total mass fluxes on the two interior boundaries are computed to monitor the global mass conservation in our numerical scheme. The total mass fluxes are defines as,

$$M_{12} = \sum_{i \in \Gamma_{12}} (\dot{m}_f)_i \quad M_{21} = \sum_{i \in \Gamma_{21}} (\dot{m}_f)_i \tag{23}$$

It is found that when the steady state is reached, the recorded M_{12} and M_{21} are sufficiently small ($|M_{12}| \leq 2.0 \times 10^{-6}$ and $|M_{21}| \leq 2.0 \times 10^{-6}$). They are very close to machine zero in a computer code of single precision.

It seems that the total mass flux defined in equation (23) is a simple and good "indicator" to the convergence behavior of a Chimer grid method. We consider the Chimera method to have "converged" if the convergence criteria in each sub-domain (such as equations (16) and (17)) are satisfied *and the global mass conservation "indicator" defined in* equation (23) *is sufficiently small.*

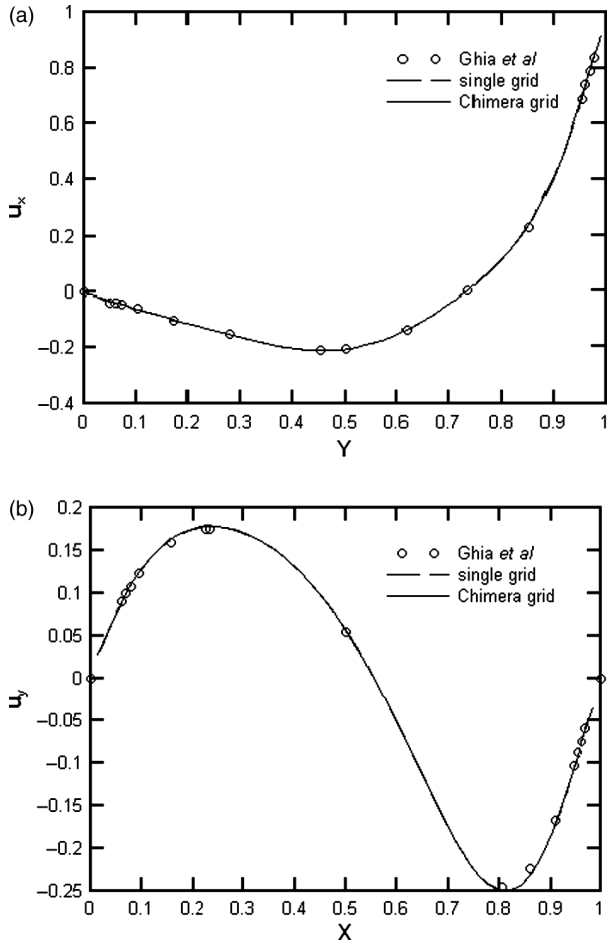

Figure 9. Velocity profiles on the vertical and horizontal centerlines of the lid-driven cavity at $Re = 100$. (a) x-component of velocity against y; (b) y-component of velocity against x.

3.2 Flow over a circular cylinder

The second example is laminar flow around a two-dimensional circular cylinder. The Reynolds number of this simulation is based on the diameter of the cylinder and the inlet velocity. This problem is inherently unsteady when the Reynolds number is larger than a critical value of approximately 40. For simulations at a *Re* number higher than the critical one, unsteady vortex shedding can be triggered by machine error alone even if the boundary conditions seems to be perfectly symmetric and no artificial perturbation is imposed on the initial condition. Simulations are performed at $Re = 100$ and 200. Vortices that shed periodically from the cylinder have to pass through the interior boundaries of the Chimera grids. This makes it a very good test to our numerical scheme.

The computational domain consists of two overlapping sub-domains are shown in figure 12. The larger domain Ω_1 is a 30×20 rectangle with a "hole". This hole is of circular shape with the radius of 2. The smaller one Ω_2 is a circular domain with the radius of 2.45 subtracting the circular cylinder with the radius of 0.5. The three circles aforementioned share a common center at (10, 0).

Sub-domain Ω_1 is meshed with 12,586 triangular cells. Sub-domain Ω_2 is meshed with 1260 rectangular cells and

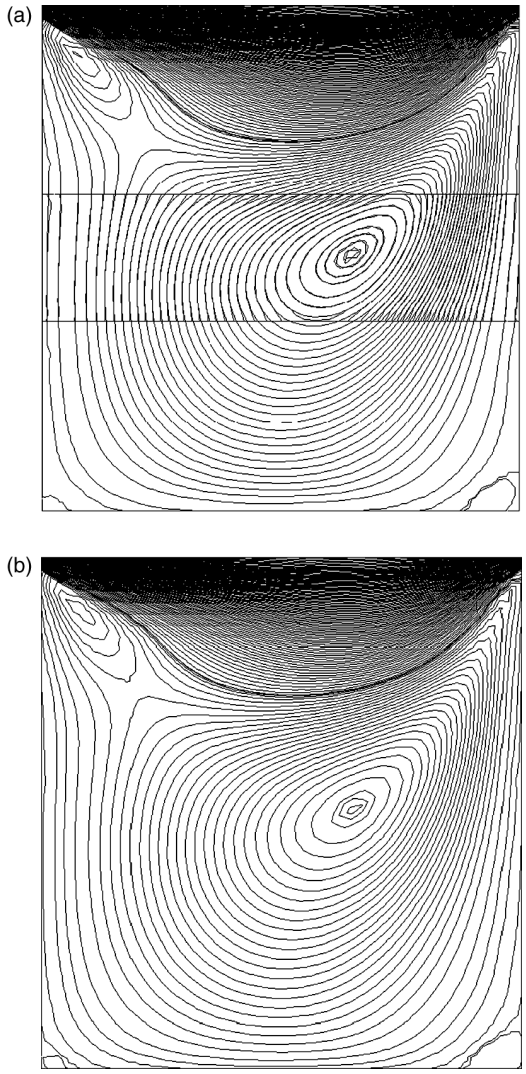

Figure 10. Contours of the x-component of velocity in the lid-driven cavity at $Re = 100$. (a) Chimera grid; (b) single grid.

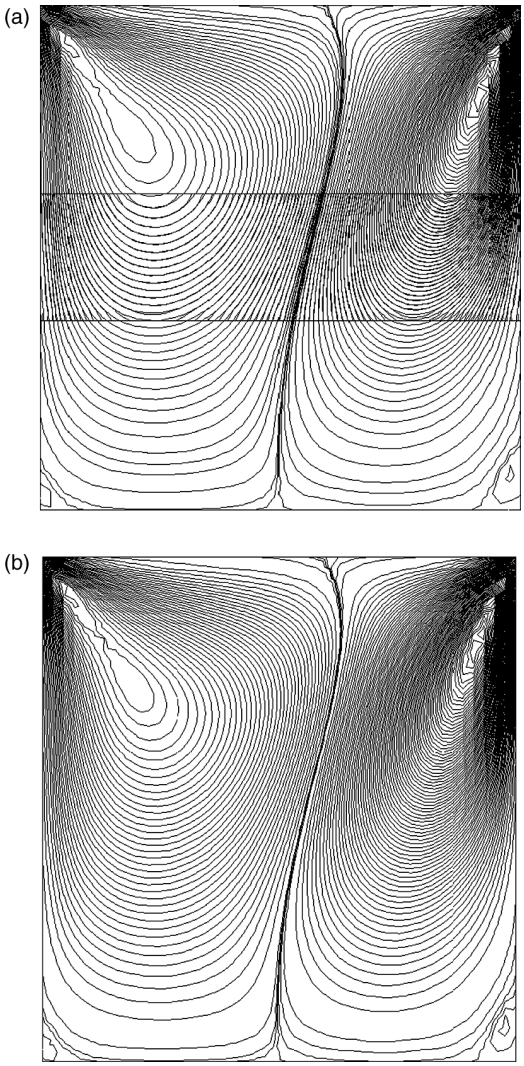

Figure 11. Contours of the y-component of velocity in the lid-driven cavity at $Re = 100$. (a) Chimera grid; (b) single grid.

4830 triangular cells. The reason for using a hybrid mesh is to resolve the boundary layer properly. The grids of the two sub-domains are presented in figure 13.

For the purpose of comparison, a single hybrid mesh (see figure 14(b)) that consists of 1260 rectangular cells and 18,722 triangular cells is generated. The same vortex shedding problem is also computed on this mesh. The resolution of this mesh is very close to the composite (Chimera) mesh (figure 14(a)).

These highly inhomogeneous and anisotropic meshes are pre-processed by a spring-analogy smoothing before they are actually used in the code. Mesh quality has been greatly improved after such treatment.

3.2.1 Pressure contours. An instantaneous pressure field (which is recorded after the period pattern has been established) is plotted in figure 15(a). It is seen that the solution in the overlapping region is of satisfactory quality. As the steady case in the previous section, this result is also obtained by using the modified interpolation scheme (22). Further investigation is conducted on the use of a direct interpolation of pressure correction. For an unsteady problem like this, such interpolation could also result in a solution eventually, but with de-coupled

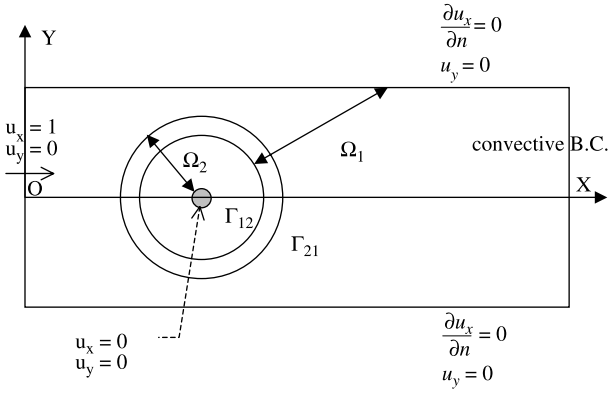

Figure 12. Domain decomposition for the simulation of the flow over a circular cylinder.

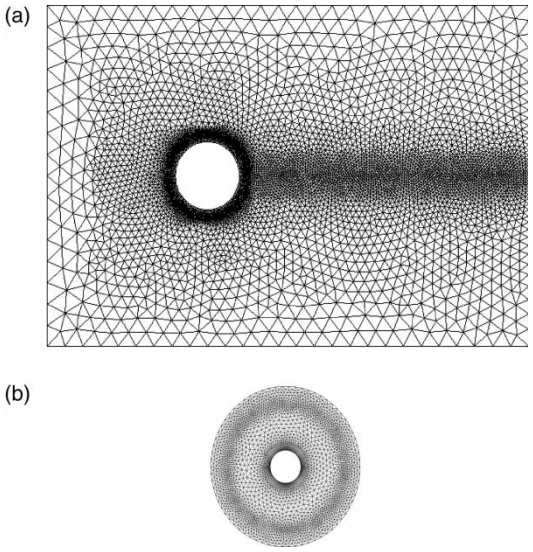

Figure 13. Sub-grids for the computation of flow over a circular cylinder. (a) background grid; (b) grid around the circular cylinder.

pressure field in the overlapping region. Such a pressure field is presented in figure 15(b) for comparison.

The reason behind this phenomenon is still not very clear. Our conjecture is that it is probably related to the method we use to solve the pressure correction equation. This Poisson-typed equation is solved by an iterative method (CG). Every time when the algebraic equation is solved, the iteration in the CG solver is stopped if it has reduced the residual by one order or the iteration number has reached 10 (depending on which comes first). It is

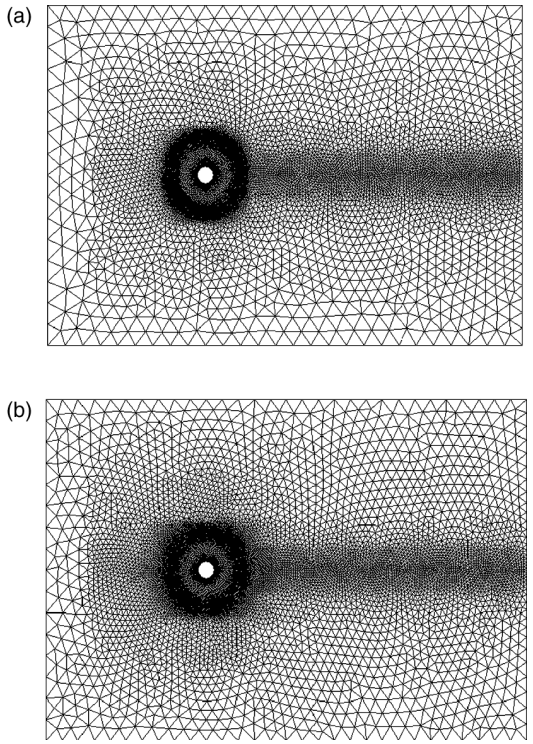

Figure 14. Grids used in the flow over a circular cylinder. (a) Composite (Chimera) grid; (b) single grid.

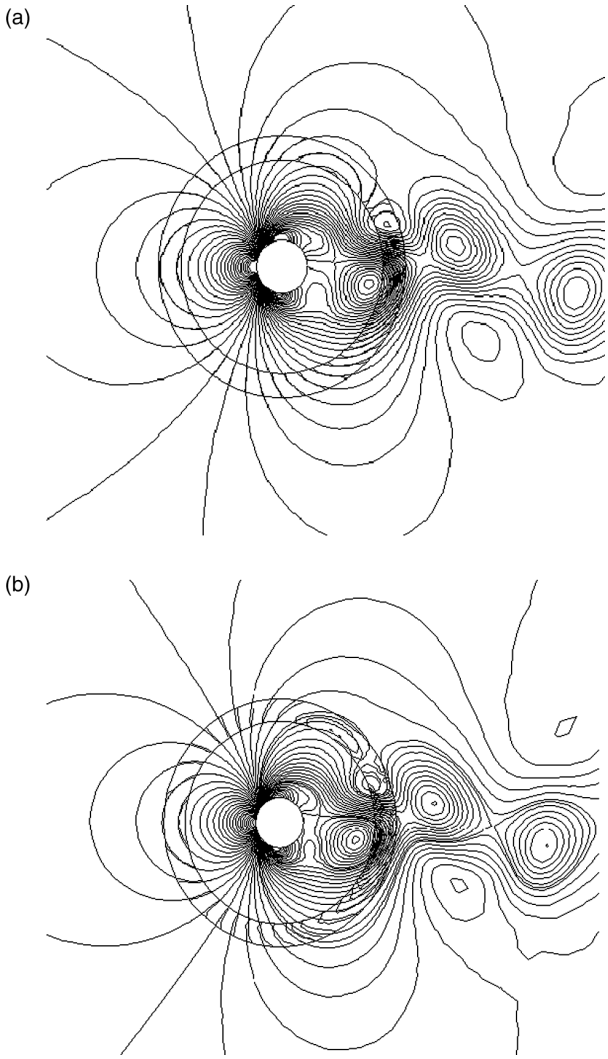

Figure 15. Instantaneous pressure field of the flow over a cylinder at $Re = 100$. (a) using the proposed interpolation scheme for pressure correction; (b) using a direction interpolation of pressure correction.

believed that under certain circumstance, this type of error can be accumulated through repeatedly interpolations on the interior boundaries.

3.2.2 Drag and lift coefficients and Strouhal number. The drag and lift coefficients are computed by the integration of the forces along the surface of the cylinder. The definitions of drag and lift coefficients are,

$$C_{\mathrm{d}} = \frac{F_x}{\frac{1}{2}\rho_0 U_\infty^2} \quad C_{\mathrm{l}} = \frac{F_y}{\frac{1}{2}\rho_0 U_\infty^2} \tag{24}$$

where F_x and F_y are the x- and y-components of the force exerted on the cylinder respectively.

The variations of C_{d} and C_{l} with time for $Re = 100$ and 200 are presented in figure 16. The frequency of vortex shedding is determined by taking the FFT of the time series of the lift coefficient. The Strouhal number based

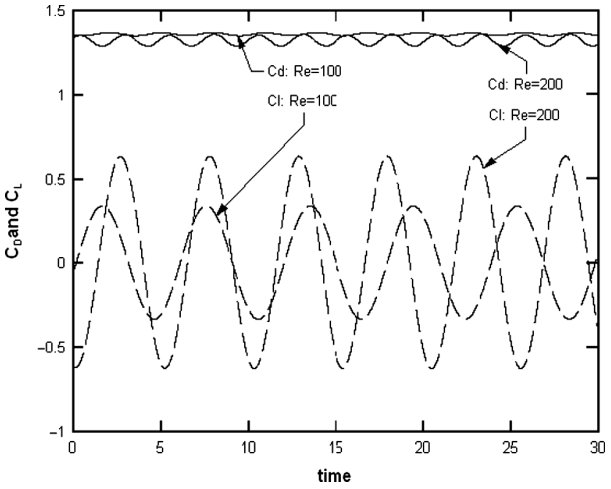

Figure 16. Variation of drag and lift coefficients with time for the flow over a cylinder at $Re = 100$ and 200.

on this frequency is defined as,

$$St = \frac{fD}{U_\infty} \tag{25}$$

where f is the shedding frequency and D is the diameter of the cylinder.

The drag and lift coefficients and Strouhal number are computed and reported in table 1. For the purpose of comparison, data from some other references are also included in the table. From this table, it is clearly seen that our result using a Chimera grid matches well with both the one using the single grid and also those from other literatures.

3.2.3 Velocity, streamline and vorticity. Instantaneous velocity-magnitude contours, streamlines and vorticity contours are plotted in figures 17–19, respectively. It is seen from these figures that the contours in the overlapping region are fairly smooth although the quality of the vorticity contours is slightly worse. Together with the evidence shown in table 1, it is confirmed that vortices pass the two interior boundaries with only minor disturbances. This example demonstrates the excellent performance of this numerical scheme in handling unsteadiness and vortex crossing.

3.2.4 Global mass conservation on the interior boundaries. The total mass fluxes on the two interior boundaries are also computed in this numerical test. $|M_{12}| \leq 1.0 \times 10^{-4}$ and $|M_{21}| \leq 5.0 \times 10^{-5}$ are recorded after the periodic pattern has been established. They are of one or two orders larger than machine zero in a single precision code. However if one considers the inlet mass flux in this test is much larger (20.0 comparing with zero in the lid-driven cavity) and the total areas (lengths) are much larger (15.39 and 12.57 comparing to 1.0 in the lid-driven cavity), they are also sufficiently small.

3.2.5 Influence of accuracy in the interpolation. In all computations of this paper, the gradient of pressure is re-constructed using a Gauss–Green method whereas the gradient of velocity is re-constructed using a linear least-square approach. To exam the influence of accuracy in the interpolation to the final solution, the gradients used in the interpolations are re-constructed using a quadratic least-square procedure (see Barth 1994 for the details). It should be noted that these high order gradients (including first and second order derivatives) are only used in the interpolations of data that are transferred between individual sub-domains and not used in the NS solver.

Numerical experiments using the high-order interpolation have results in almost identical lift and drag coefficients and shedding frequency. No significant improvement of solution quality has been observed in the overlapping region. This observation will favor a linear interpolation plus local mesh refinement over the utilization of higher order interpolation. However, since no tests on grids with different resolution are performed, it is hard to compare the efficacy of these two approaches quantitatively.

4. Conclusions

We have modified the pressure correction algorithm to compute viscous incompressible flow on an unstructured Chimera grid. A new interpolation method is proposed to transfer pressure correction between sub-domains. Compared with the method of directly interpolating pressure correction, this algorithm prevents the unphysical

Table 1. Drag and lift coefficients and Strouhal number of the flow over a circular cylinder at Reynolds number 100 and 200.

	$Re = 100$			$Re = 200$		
	C_d	C_l	St	C_d	C_l	St
Chimera grid	1.36 ± 0.01	± 0.34	0.168	1.33 ± 0.03	± 0.63	0.196
Single grid	1.36 ± 0.01	± 0.33	0.168	1.33 ± 0.04	± 0.65	0.198
Braza *et al.* (1986)	1.36 ± 0.015	± 0.25	–	–	–	–
Calhoun (2002)	1.33 ± 0.01	± 0.30	0.175	–	–	–
Liu *et al.* (2000)	1.36 ± 0.01	± 0.34	0.164	–	–	–
Pan and Damodaran (2002)	–	–	–	1.37 ± 0.04	± 0.63	0.192
Rogers and Kwak (1990)	–	–	–	1.23 ± 0.05	± 0.65	0.185
Rosenfeld *et al.* (1991)	–	–	–	1.46 ± 0.05	± 0.69	0.211
Wille (1960)	–	–	–	–	–	0.190

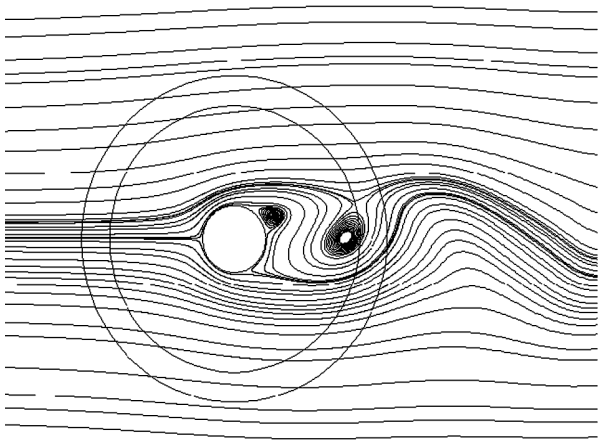

Figure 17. Instantaneous streamlines of the flow over a cylinder at $Re = 100$.

decoupling of the pressure field from occurring in the overlapping region and gives better results.

From the two numerical examples, one steady and one unsteady case, it is shown that this algorithm does not necessitate corrections for global mass conservation. It is also observed that on an unstructured Chimera gird, the improvement of solution quality in the overlapping region due to a higher order interpolation alone is quite limited. If one takes the complexity and cost of such schemes into consideration, an easy way in practice is a local mesh refinement near the overlapping boundaries. The only drawback of this method is minor increase in the total number of grid points.

Our numerical examples have demonstrated the credibility of this method in computing both steady and unsteady flows. Although only two-dimensional simulations are conducted in this paper, the extension of this method to three-dimensional geometry is straightforward. This method has great potential in the study of moving body with complex geometry. Future work will focus on moving boundary problems such as flapping wing and rotating machinery.

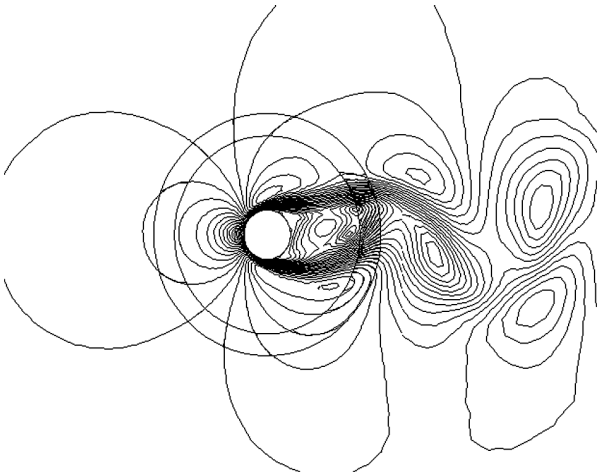

Figure 18. Instantaneous velocity magnitude contours of the flow over a cylinder at $Re = 100$.

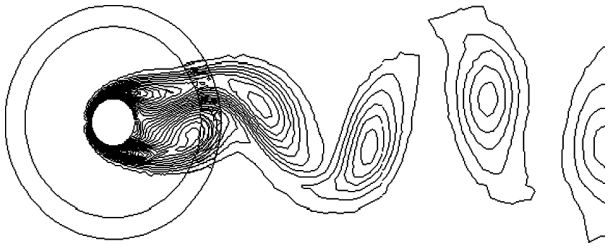

Figure 19. Instantaneous vorticity contours of the flow over a cylinder at $Re = 100$.

Acknowledgements

This work was partially sponsored by the National Natural Science Foundation of China under the project No. 10325211 and the Chinese Academy of Sciences under the innovative project "Multi-scale modeling and simulation in complex system" (KJCX-SW-L08). The author also expresses his appreciation to the support and funds from "Project-sponsored by SRF for ROCS, SEM".

References

Barth, T.J., Aspects of unstructured grids and finite-volume solvers for the Euler and Navier–Stokes equations. Lecture Notes Presented of the VKI Lecture Series 1994–05, 1994 (Von Karman Institute for Fluid Dynamics: Belgium).

Braza, M., Chassaing, P. and Minh, H.H., Numerical study and physical analysis of the pressure and velocity fields in the near wake of a circular cylinder. *J. Fluid Mech.*, 1986, **165**, 79–130.

Calhoun, D., A Cartesian grid method for solving the two-dimensional stream function–vorticity equations in irregular regions. *J. Comput. Phys.*, 2002, **196**, 231–275.

Demirdzic, I. and Muzaferija, S., Numerical method for coupled fluid flow, heat transfer and stress analysis using unstructured moving meshes with cells of arbitrary topology. *Comput. Meth. Appl. Mech. Eng.*, 1995, **125**, 235–255.

Demirdzic, I., Muzaferija, S. and Peric, M., Advances in computation of heat transfer, fluid flow, and solid body deformation using finite volume approaches. In *Advances in Numerical Heat Transfer*, edited by Minkowycz, W.J. and Sparow, E.M., Vol. 1, Chapter 2, 1997 (Taylor & Frances: New York).

Ferziger, J.H. and Peric, M., *Computational Methods for Fluid Dynamics*, pp. 196–264, 1996 (Springer-Verlag: Berlin and Heidelberg).

Ghia, U., Ghia, K.N. and Shin, C.T., High *Re* solutions for incompressible flow using Navier–Stokes equation and multi-grid methods. *J. Comput. Phys.*, 1982, **48**, 387–411.

Houzeaux, G. and Codina, R., A Dirichlet/Neumann domain decomposition method for incompressible turbulent flows on overlapping subdomains. *Comput. Fluids*, 2004, **33**, 771–782.

Liu, Z., Zheng, X. and Sung, C.H., Preconditioned multi-grid methods for unsteady incompressible flows. *J. Comput. Phys.*, 2000, **160**(1), 151–178.

Maruoka, A., Finite element analysis for flow around a rotating body using Chimera method. *Int. J. Comput. Fluid Dynamics*, 2003, **17**(4), 289–297.

Mavriplis, D.J., Unstructured grid techniques. *Annu. Rev. Fluid Mech.*, 1997, **29**, 473–514.

Nakahashi, K., Togashi, F. and Sharov, D., Inter-grid boundary definition methods for overset unstructured grid approach. *AIAA J.*, 2000, **38**(11), 2077–2084.

Pan, H. and Damodaran, M., Parallel computation of viscous incompressible flows using Godunov-projection method on overlapping grids. *Int. J. Numer. Meth. Fluids*, 2002, **39**, 441–463.

Perot, J.B. and Nallapati, R., A moving unstructured staggered mesh method for the simulation of incompressible free-surface flows. *J. Comput. Phys.*, 2003, **184**, 192–214.

Rogers, S.E. and Kwak, D., Upwinding differencing scheme for the time-accurate incompressible Navier–Stokes equations. *AIAA J.*, 1990, **28**(2), 253–262.

Rosenfeld, M., Kwak, D. and Vinokur, M., A fractional-step solution method for the unsteady and incompressible Navier–Stokes equations in generalized co-ordinate systems. *J. Comput. Phys.*, 1991, **94**(1), 102–137.

Slone, K.A., Pericleous, K., Bailey, C. and Cross, M., Dynamic fluid structure interaction using finite volume unstructured mesh procedures. *Comput. Struct.*, 2002, **80**, 371–390.

Tang, H.S. and Casey Jones, S., Fotis Sotiropoulos, An overset-grid method for 3D unsteady incompressible flows. *J. Comput. Phys.*, 2003, **191**, 567–600.

Togashi, F., Ito, Y., Murayana, M., Nakahashi, K. and Kato, T., Flow simulation of flapping wings of an insect using overset unstructured grid, AIAA 2001-261, 2001.

Wille, R., Karman vortex street. *Adv. Appl. Mech.*, 1960, **6**, 273–287.