World Academy of Science, Engineering and Technology
International Journal of Civil and Environmental Engineering
Vol:2, No:10, 2008

Study on a Nested Cartesian Grid Method

Yih-Ferng Peng

Open Science Index, Civil and Environmental Engineering Vol:2, No:10, 2008 publications.waset.org/3785/pdf

***Abstract*—**In this paper, the local grid refinement is focused by using a nested grid technique. The Cartesian grid numerical method is developed for simulating unsteady, viscous, incompressible flows with complex immersed boundaries. A finite volume method is used in conjunction with a two-step fractional-step procedure. The key aspects that need to be considered in developing such a nested grid solver are imposition of interface conditions on the inter-block and accurate discretization of the governing equation in cells that are with the inter-block as a control surface. A new interpolation procedure is presented which allows systematic development of a spatial discretization scheme that preserves the spatial accuracy of the underlying solver. The present nested grid method has been tested by two numerical examples to examine its performance in the two dimensional problems. The numerical examples include flow past a circular cylinder symmetrically installed in a Channel and flow past two circular cylinders with different diameters. From the numerical experiments, the ability of the solver to simulate flows with complicated immersed boundaries is demonstrated and the nested grid approach can efficiently speed up the numerical solutions.

***Keywords*—**local grid refinement, Cartesian grid, nested grid, fractional-step method.

I. Introduction

In the numerical simulation of complex physical phenomena, the crucial requirement is predictability, *i.e.*, that the simulation results remain faithful to the actual physical processes. Errors resulting from a lack of spatial resolution are particularly deleterious. However, over-resolving is computationally expensive. As a result, how to efficiently and effectively solve the partial differential equations which represent the mathematical model of physical problems concerned becomes a subject of active research in numerical analysis [1], [2].

In general, there are two approaches to obtain accurate solution of PDEs. One approach is to employ high-order numerical method [3], and the other is to improve the resolution through the computational grid. Mesh refinement is desirable to improve spatial resolution by using uniform or non-uniform grids. The uniform mesh refinement is that the resulting grid evolution equation is generally less nonlinear and less stiff which becomes very efficient in conjunction with the line successive-overrelaxation (SOR) solver. However, the uniform mesh refinement is not perfect for the applications, of which the solution may need different resolutions for different regions.

In this paper, we develop a structured, nested Cartesian grid method for simulating unsteady, viscous, incompressible flows with complex immersed boundaries. One drawback of adopting nested grid method as the local refinement technique is that the resulting nested grids are uniform and the corresponding evolution equations is less stiff and very efficient in conjunction with line-SOR. For the numerical simulations of flow past bluff body problems in the present study, we used an Immersed Boundary (IB) method where the solid object is represented by a distributed body force in the Navier–Stokes equations [4].

The current paper will focus on describing these and other salient features of the numerical methodology, validating the accuracy and fidelity of the approach and demonstrating the capabilities of the solver in some complex configurations.

Y. F. Peng is with the Department of Civil Engineering, National Chi-Nan University, Puli, Taiwan, 545, R. O. C. (Tel: 886-49-2910960 ext. 4960; e-mail: yfpeng@ncnu.edu.tw).

II. Numerical Methodology

A. Fractional-Step Method

We first describe the underlying solver, the fractional-step method, for a Cartesian mesh [5]. The governing equation is the unsteady, viscous, incompressible Navier–Stokes equation written in terms of the primitive variables. This equation is discretized on a Cartesian mesh using a cell-centered colocated (non-staggered) arrangement of the primitive variables ($\vec{u}, p$). The integral form of dimensionless governing equations is given by mass conservation

$$\int_{CS} \vec{u} \cdot \vec{n} dS = 0 , \tag{1}$$

and momentum conservation

$$\frac{\partial}{\partial t}\int_{CV} \vec{u} dV + \int_{CS} \vec{u}(\vec{u} \cdot \vec{n}) dS = -\int_{CS} p\vec{n} dS + \frac{1}{\mathrm{Re}}\int_{CS} (\nabla \vec{u}) \cdot \vec{n} dS \tag{2}$$

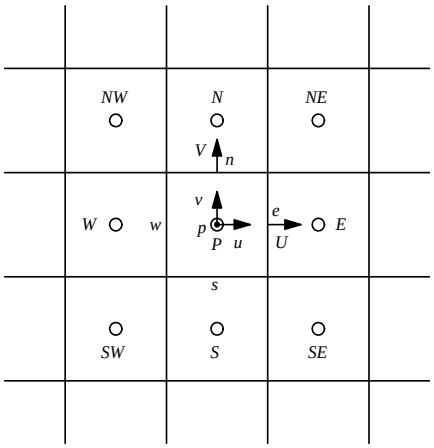
This is used as the starting point for deriving a second-order accurate finite-volume method. In the above equations CV and CS denote the control-volume and control-surface, respectively, and $\vec{n}$ is a unit vector normal to the control-surface. A second-order accurate, two-step fractional step method is used for advancing the solution in time. In this time-stepping scheme, the solution is advanced from time level "n" to "n+1" through an intermediate advection-diffusion step where the momentum equations without the pressure gradient terms are first advanced in time. A second-order Adams–Bashforth scheme is employed for the convective terms and the diffusion terms are discretized using an implicit Crank–Nicolson scheme. This eliminates the viscous stability constraint which can be quite severe in simulation of viscous flows. At this stage, in addition to the cell-center velocities which are denoted by $\vec{u}$, we also introduce face-center velocities $\vec{U}$. In a manner similar to a fully staggered

Open Science Index, Civil and Environmental Engineering Vol:2, No:10, 2008 publications.waset.org/3785/pdf

arrangement, only the component normal to the cell-face is computed and stored (see Fig.1). The face-center velocity is used for computing the volume flux from each cell in our finite-volume discretization scheme. The semi-discrete form of the advection-diffusion equation for each cell can therefore be written as

$$\frac{\partial}{\partial t}\int_{CV}\frac{\vec{u}^{*}-\vec{u}^{n}}{\Delta t}dV=-\frac{1}{2}\int_{CS}[3\vec{u}^{n}(\vec{U}^{n}\cdot\vec{n})-\vec{u}^{n-1}(\vec{U}^{n-1}\cdot\vec{n})]dS$$
$$+\frac{1}{2\,\mathrm{Re}}\int_{CS}(\nabla\vec{u}^{*}+\nabla\vec{u}^{n})\cdot\vec{n}dS\,, \qquad (3)$$

where $\vec{u}$ is the intermediate cell-center velocity and CV and CS denote the volume and surface of a cell, respectively. Following the advection-diffusion step, the intermediate face-center velocity $\vec{U}$ is computed by interpolating the intermediate cell-center velocity.



o : *5-point stencil for regular cell*

Fig. 1 Schematic plots of the regular Cartesian cells and arrangement of cell-center and face-center velocities.

The advection-diffusion step is followed by the pressure-correction step

$$\int_{CV}\frac{\vec{u}^{n+1}-\vec{u}^{*}}{\Delta t}dV=\int_{CV}\nabla p^{n+1}dV, \qquad (4)$$

where we require that the final velocity field satisfy the integral mass conservation equation given by

$$\int_{CS}\vec{U}^{n+1}\cdot\vec{n}\;\;dS=0\,. \qquad (5)$$

This results in the following equation for pressure

$$\int_{CS}(\nabla p^{n+1})\cdot\vec{n}dS=\frac{1}{\Delta t}\int_{CS}(\vec{U}^{*}\cdot\vec{n})dS, \qquad (6)$$

which is the integral version of the pressure Poisson equation. Note that the pressure correction step is represented by Eq. 4 and is well posed only if the velocity component normal to the boundary is specified. The velocity boundary condition consistent with Eq. 4 is $\vec{u}^{n+1}\cdot\vec{N}=\vec{v}^{n+1}\cdot\vec{N}$, where $\vec{v}$ is the prescribed boundary velocity and $\vec{N}$ is the unit normal to the boundary of the flow domain. It can be easily shown that this implies that $(\nabla p^{n+1})\cdot\vec{N}=0$ be used as the boundary condition for Eq. 6. Once the pressure is obtained by solving this equation, both the cell-center (cc) and face-center (fc) velocities are updated separately as

$$\vec{u}^{n+1}=\vec{u}^{*}-\Delta t(\nabla p^{n+1})_{cc},\ \vec{U}^{n+1}=\vec{U}^{*}-\Delta t(\nabla p^{n+1})_{fc}\,. \qquad (7)$$

It should be pointed out that the pressure gradient computed at the face-center is not simply an interpolated version of the pressure gradient at the cell-center. For instance with reference to Fig. 1 the x-direction pressure gradient at the cell center is computed as

$$(\partial p/\partial x)_{p}=(p_{E}-p_{W})/2\Delta x, \qquad (8)$$

whereas the same gradient on the east face is given by

$$(\partial p/\partial x)_{e}=(p_{E}-p_{P})/\Delta x. \qquad (9)$$

It follows that $\vec{U}^{n+1}$ is not simply an interpolated version of the face-center velocities $\vec{u}^{n+1}$. In fact the pressure equation (Eq. 6) is discretized in terms of the pressure gradients on the cell faces and with the separate update of the face-center velocity as shown in Eq. 7, exact satisfaction of Eq. 5 is guaranteed.

B. Nested Grid Methodology

In this study, the local grid refinement technique is completed by a nested grid method. Sketch of the flow domain and the inside coarse and fine grid area are plotted and shown as in Fig.2. For simplicity, we illustrate our nested grid technique in a two-dimensional (2D) flow with a 2-block domain. It is worth noting that the present nested grid technique can be further applied to a 3D flow and higher multi-block domain without mystery. As shown in Fig.2, the coarse and fine grid domains are denoted by D_1 by D_2, respectively. Again, for simplicity, the fine grid length is defined as half as that of coarse grid. Our approach is to solve the flow in both D_1 and D_2 domains simultaneously according to the procedure of fractional step method.

The advection-diffusion step

The approach for a nested grid method is extended from the uniform grid method. We get start from the advection-diffusion step, the scalar form of u^{*} discretization equation (Eq. 3) can be read as follows (v^{*} equation is similar and not shown in this paper),

$$\frac{u^{*}-u^{n}}{\Delta t}dxdy=-\frac{1}{2}\{3[(u_{e}^{n}U_{e}^{n}-u_{w}^{n}U_{w}^{n})dy+(u_{n}^{n}V_{n}^{n}-(u_{s}^{n}V_{s}^{n})dx]$$
$$-[(u_{e}^{n-1}U_{e}^{n-1}-u_{w}^{n-1}U_{w}^{n-1})dy+(u_{n}^{n-1}V_{n}^{n-1}-u_{s}^{n-1}V_{s}^{n-1})dx]\}$$
$$+\frac{1}{2Re}\{[(\frac{\partial u^{*}}{\partial x})_{e}-(\frac{\partial u^{*}}{\partial x})_{w}]dy+[(\frac{\partial u^{*}}{\partial y})_{n}-(\frac{\partial u^{*}}{\partial y})_{s}]dx$$
$$+[(\frac{\partial u^{n}}{\partial x})_{e}-(\frac{\partial u^{n}}{\partial x})_{w}]dy+[(\frac{\partial u^{n}}{\partial y})_{n}-(\frac{\partial u^{n}}{\partial y})_{s}]dx\} \qquad (10)$$

Where u and v are the x- and y-directions velocities, respectively, at the cell-center, U_{e}, U_{w}, V_{n}, and V_{s} are the introduced face center velocities.

Eq. 10 is the discretization form of advection-diffusion equation for a general cell. We will describe how these

Open Science Index, Civil and Environmental Engineering Vol:2, No:10, 2008 publications.waset.org/3785/pdf

equations are implemented in cells without domain interface, in D_1 cells with domain interface, and in D_2 cells with domain interface in sequence.

Cells without domain interface

For those control volumes without domain interface, Eq. 10 is exactly the same as Eq. 3 of a single block domain which has been outlined in the previous section. By using second order accurate center difference method for spatial terms, the cell-center velocities and gradient of these velocities at the cell-face are computed from those adjacent cell-center velocities, *i.e.*, (equation of v_s (Eq. 12) is not shown)

$$u_e = (u_E + u_P)/2,\ u_w = (u_W + u_P)/2,$$
$$u_n = (u_N + u_P)/2, \qquad u_s = (u_S + u_P)/2, \tag{11}$$

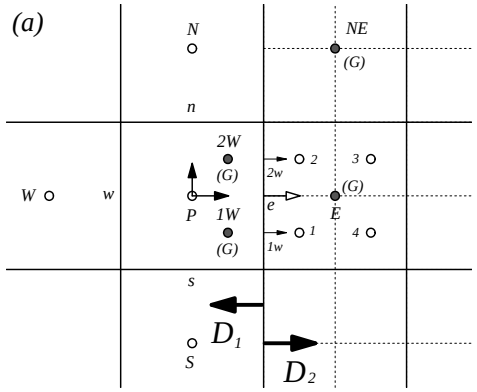
And $$(\frac{\partial u}{\partial x})_e - (\frac{\partial u}{\partial x})_w = \frac{(u_E - 2u_P + u_W)}{dx},$$
$$(\frac{\partial u}{\partial y})_n - (\frac{\partial u}{\partial y})_s = \frac{(u_N - 2u_P + u_S)}{dy}. \tag{13}$$

D_1 cells with domain interface

For those D_1 cells with domain interface as shown in Fig.2 (a), since u_E and v_E are not defined in the computational domain, the cell-center velocities at the east surface can not be computed by using Eqs. 11~12 directly, we need to computed velocities at the ghost cell by introducing appropriated interpolations first. In this study, a simple way devoted to calculate these ghost velocities is using averaged velocity of the occupied D_2 cells instead, *i.e.*,

$$u_E = u_G = (u_1 + u_2 + u_3 + u_4)/4, \tag{14}$$

where subscript G denotes the D_1 ghost cell (for instant E=G, E denotes the eastern cell), as shown in Fig.2 (a), which control volume is occupied by those four 1, 2, 3, and 4 D_2 cells. With the use of Eq. 14, the general form of the cell-surface velocities, Eq. 12 can be preserved. Most importantly, the second-order accurate of the present numerical method are also prevented.



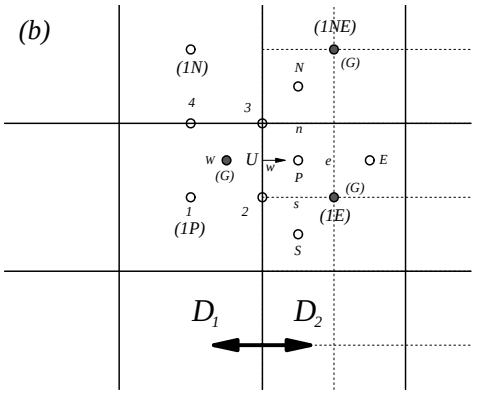


Fig. 2 Schematic plots of refinement cells. (a) Coarse cell. (b) Fine cell.

That is the domain-interface processing which plays the essential part of the present nested grid method. In the general advection-diffusion equation Eq. 10, except u_e and v_e which has been processed in the previous paragraph, there are some other terms such as U_e, $(\partial u/\partial x)_e$ and $(\partial v/\partial x)_e$ located at the block interface. Where U_e and $(\partial u/\partial x)_e$ and $(\partial v/\partial x)_e$ are used for computing the volume flux and momentum flux, respectively, from D_1 to D_2 cells. Although there are varied interpolations available, we use the volume and momentum flux from D_2 to D_1 cells instead, i.e.,

$$U_e \times dy_1 = U_{1w} \times dy_2 + U_{2w} \times dy_2, \tag{15}$$

$$(\frac{\partial u}{\partial x})_e \times dy_1 = (\frac{\partial u}{\partial x})_{1w} \times dy_2 + (\frac{\partial u}{\partial x})_{2w} \times dy_2 =$$
$$\frac{u_1 - u_{1W}}{dx_2} \times dy_2 + \frac{u_2 - u_{2W}}{dx_2} \times dy_2 \tag{16}$$

where dy_1 and dy_2 denote the vertical lengths of D_1 and D_2 cells, respectively. It has to noted that the use of Eqs. 15~16 can make sure the mass and the momentum diffusion fluxes, respectively, through the D_1 and D_2 domain interface and vice versa are consistent.

D_2 cells with domain interface

For the D_2 cells with domain interface as shown in Fig.2 (b), since u_W , and v_W are not defined in the computational domain, the general form of the cell-face velocities (Eq. 12) can not be applied directly, we need to compute these terms by introducing second order accurate interpolations. Similar to the processing of ghost D_1 cells, these ghost velocities are obtained by using averaged value of those adjacent velocities, i.e.,

$$u_W = u_G = (u_1 + u_2 + u_3 + u_4)/4, \tag{17}$$

where $u_1 = u_{1P}$, $u_2 = (u_{1P} + u_{1E})/2$, $u_3 = (u_{1P} + u_{1E} + u_{1NE} + u_{1N})/4$, and $u_4 = (u_{1P} + u_{1N})/2$, in which the subscript 1P, 1E, 1N, and 1NE denote those corresponding D_1 cell, D_1 eastern, northern, and north-east cells, respectively, as shown in Fig.2 (b). Here, $u_{1E}, v_{1E}, u_{1NE}, v_{1NE}$ are not defined in our computational domains, these velocities are again obtained from Eq. 14.

Computing U^* and V^*

Open Science Index, Civil and Environmental Engineering Vol:2, No:10, 2008 publications.waset.org/3785/pdf

Following the advection-diffusion step, the intermediate face-center velocity U^* and V^* are computed by interpolating the intermediate cell-center velocity.

(1) For fine-fine or coarse-coarse non-inter-domain cell-face

$U^*_{i,j} = (u^*_{i,j} + u^*_{i+1,j})/2$, $V^*_{i,j} = (v^*_{i,j} + v^*_{i+1,j})/2$. (18)

(2) For fine-coarse domain interface

At the fine-coarse domain interface shown in Fig. 2, U_e is obtained from Eq. 15,

$U^*_e \times dy_1 = U^*_{1w} \times dy_2 + U^*_{2w} \times dy_2$.

While U_{1w} and U_{2w} are computed by interpolating the intermediate cell-center velocity, i.e.,

$U_{1w} = (u_{1W} + u_1)/2$ and $U_{2w} = (u_{2W} + u_2)/2$, (19)

where subscripts 1W and 2W denote those ghost D_2 cells, consequently, u_{1W} and u_{2W} are again obtained from ghost cell equation (Eq. 17).

Solving pressure

The scalar form of the pressure descritization equations (Eq. 6) can be read as

$$[(\frac{\partial p^{n+1}}{\partial x})_e - (\frac{\partial p^{n+1}}{\partial x})_w]dy + [(\frac{\partial p^{n+1}}{\partial y})_n - (\frac{\partial p^{n+1}}{\partial y})_s]dx$$

$$= \frac{1}{\Delta t}[(U^*_e - U^*_w)dy + (V^*_n - V^*_s)dx] \quad (20)$$

Cells without domain interface

By second order accurate center difference method, pressure gradients at cell surface are computed from

$(\partial p/\partial x)_e = (p_E - p_P)/dx$, $(\partial p/\partial x)_w = (p_P - p_W)/dx$,

$(\partial p/\partial y)_n = (p_N - p_P)/dy$, $(\partial p/\partial y)_s = (p_P - p_S)/dy$ (21)

D_1 cells with domain interface

For those D_1 cells with domain interface as shown in Fig. 2(a), the pressure gradient term at the interface, $(\partial p/\partial x)_e$, need to be processed rather than be computed by using Eq. 21 directly. In a manner similar to the momentum diffusions processing, the pressure gradient is obtained from the diffusion consistency equation,

$$(\frac{\partial p}{\partial x})_e \times dy_1 = (\frac{\partial p}{\partial x})_{1w} \times dy_2 + (\frac{\partial p}{\partial x})_{2w} \times dy_2$$

$$= \frac{p_1 - p_{1W}}{dx_2} \times dy_2 + \frac{p_2 - p_{2W}}{dx_2} \times dy_2 . \quad (22)$$

D_2 cells with domain interface

For those D_2 cells with domain interface as shown in Fig.2 (b), since W is ghost cell and p_W is not defined in the computational domain, the pressure gradient at the east surface can not be computed by using Eq. 21 directly, we need to computed ghost pressure p_G ($p_G = p_W$). Similar to the advection-diffusion processing, the interpolation is listed as

$p_W = p_G = (p_1 + p_2 + p_3 + p_4)/4$, (23)

where $p_1 = p_{1P}$, $p_2 = (p_{1P} + p_{1E})/2$, $p_3 = (p_{1P} + p_{1E} + p_{1NE} + p_{1N})/4$, and $p_4 = (p_{1P} + p_{1N})/2$, in which p_{1E} and p_{1NE} are not defined in our computational domains, these pressures are once again obtained from the ghost sell equation

$p_{1E} = p_G = (p_1 + p_2 + p_3 + p_4)/4$, (24)

where subscript G denotes the D_1 ghost cell (G=1E, 1E denotes the eastern cell), as shown in Fig.2 (a), which control volume is occupied by those four D_2 cells denoted by 1, 2, 3, and 4.

Velocity correction

Once the pressure is obtained by solving pressure equation, both the cell-center and face-center velocities are updated separately as

$u^{n+1} = u^* - \Delta t(\partial p^{n+1}/\partial x)_{cc}$, and $v^{n+1} = v^* - \Delta t(\partial p^{n+1}/\partial y)_{cc}$;

$U^{n+1} = U^* - \Delta t(\partial p^{n+1}/\partial x)_{fc}$, and $V^{n+1} = V^* - \Delta t(\partial p^{n+1}/\partial y)_{fc}$.

Again, we will describe how the cell-center velocity be corrected in cells without domain interface, in D_1 cells with domain interface, and in D_2 cells with domain interface, and how the face-center velocity be corrected at the non domain-interface and at the domain interface in sequence.

Cell-center velocity correction in cells without domain interface

For instance with reference to Fig. 1, velocity correction at the cell center is computed as

$u^{n+1} = u^* - \Delta t \times (\partial p/\partial x)_p = u^* - \Delta t \times (p_E - p_W)/2dx$ (25)

Cell-center velocity correction in D_1 cells with domain interface

For those D_1 cells with domain interface as shown in Fig.2 (a), since p_E is not defined in the computational domain, the cell-center velocity can not be corrected by using Eq. 25 directly, we need to computed p_E by introducing appropriated interpolation first. The ghost pressure is the averaged pressure of the occupied D_2 pressures (Eq. 24), i.e.,

$p_E = p_G = (p_1 + p_2 + p_3 + p_4)/4$,

Cell-center velocity correction in D_2 cells with domain interface

For those D_2 cells with domain interface as shown in Fig. 2(b), since p_W is not defined in the computational domain, the pressure gradient at the east surface can not be computed by using Eq. 25 directly, we need to computed p_W first. Similar to those processing in the previous paragraph, the ghost pressure is computed by Eq. 23,

$p_W = p_G = (p_1 + p_2 + p_3 + p_4)/4$,

Velocity correction at non inter-domain face-center

For fine-fine or coarse-coarse non inter-domain face-center, we have general form of velocity correction at face-center as

$U^{n+1} = U^* - \Delta t \times (\partial p^{n+1}/\partial x)_e = U^* - \Delta t \times (p_E - p_P)/dx$ (26)

Velocity correction at domain interface

At the fine-coarse cell interface shown in Fig. 2, coarse cell face-center velocity, U^{n+1}_e , is obtained from the consistent equation (Eq. 15),

$U^{n+1}_e \times dy_1 = U^{n+1}_{1w} \times dy_2 + U^{n+1}_{2w} \times dy_2$,

while $U^{n+1}_{1,w}$ and $U^{n+1}_{2,w}$ are corrected by Eq. 26 with the fine mesh domain ghost velocity formulation Eq. 17.

Open Science Index, Civil and Environmental Engineering Vol:2, No:10, 2008 publications.waset.org/3785/pdf

C. Immersed Boundary Method

In this study, a simple concept of Immersed Boundary method is adopted for numerical simulation of bluff body flows by using a distributed body force in the Navier-Stokes equations instead of the existence of solid body. The governing equations for fluid flow are solved everywhere, including cells which are occupied by the solid body. The distributed body force is determined at every time step of the iteration by requiring the value of the velocity in those cells to match the prescribed velocity of the solid body. Similar procession has been applied by Ravoux *et al.* (2003) and they referred it as Embedding method.

III. Numerical Results

In this section, two examples are presented to examine the performance of the proposed nested grid refinement method.

A. Flow past a Circular Cylinder in a Channel

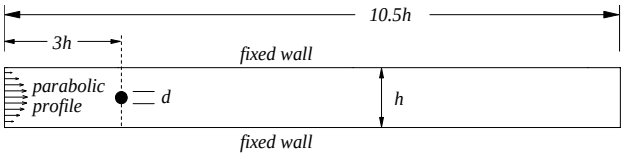


Fig. 3 Computational domain of flow past a cylinder placed symmetrically in a planar channel.

The first test case is to validate the solver in the finite Reynolds number regime by simulating unsteady flow past a circular cylinder of diameter d placed symmetrically in a planar channel of height h. Fig. 3 shows a schematic of the flow configuration that has been simulated in this study. Parabolic profile is specified at the channel inlet and the two main parameters in this flow are the blockage ratio $\beta = d/h$ and the Reynolds number defined as $Re = Q/\nu$ where Q denotes the inlet volume flux. A systematic numerical study of this flow configuration over a range of parameters has been conducted by Chen *et al.* (1995) [6] and results from this study are used to validate our simulations. This flow shares some features with the case of a cylinder immersed in a free stream. In particular, vortex shedding is observed in this flow beyond a critical Reynolds numbers. However, the critical Reynolds number is a strong function of the blockage ratio. Furthermore, the development of the vortices is also significantly affected by the boundary layers that develop on the channel walls. Thus, overall this is a more complicated flow and a good test case for our simulation methodology.

We first conducted a resolution study by solving the vortex shedding flow at *Re*=500 for a blockage ratio of 0.2. A series of grid systems including 1block, 2block, and 3block are used for simulation this flow. The resolution study included two parts, local-refined-domain inference study and grid independence study, which are separately conducted and results are collected and listed in Table 1 and Table 2, respectively. In Table 1, numerical results including dimensionless dynamics forces acting on the circular cylinder, mean drag force coefficient $\overline{C_d}$ and lift force amplitude C_l, and dimensionless vortex shedding frequency, the Strouhal number, *St*, of uniform grid systems are listed. Where B1G1 denotes a one block coarse uniform grid system with $\Delta x = \Delta y = 0.02h$, and B1G2 denotes a one block finer uniform grid system with $\Delta x = \Delta y = 0.01h$. It is noted that the numerical results are strongly dependent on the resolutions. The difference of C_l obtained by B1G1 to that from B1G2 up to 34.4%, while differences of $\overline{C_d}$ and *St* between coarse (B1G1) and finer (B1G2) grids are 6.8% and 5.0%, respectively. Although the improve numerical results of B1G2 to B1G1 are significant, the B1G2 grid number are four times of B1G1 and the CPU time can be more than 6 times of that of B1G1, which means there is necessary of a local refinement. To explore the local-refinement-domain inference on the simulated results, we performed a series of numerical simulations by the 2block method arranging the refined area around the circular cylinder. Eight grid systems denoted by B2D1 to B2D8 are conducted, the refined area of each grid system can be seen in Fig. 4 and the grid numbers, including coarse (G1) and finer (G2) grid numbers, are listed on Table 1. From the numerical results listed on Table 1, it is noted that refined grids around cylinder improve the numerical simulation results significantly. For instant with extra 6,000 refined grids (B2D5 in Table 1), about 20% of the G1 grid number, the present 2block method can predict a closely resembled results of B1G2. The differences of computed $\overline{C_d}$, C_l, and *St* values between B1G2 and B2D5 are all within 1.0%. This local-refined-domain inference study not even demonstrates efficiency of the present nested grid method, but also indicates the local-refinement necessary around the circular cylinder.

From the image of the local-refined-domain inference study, we conducted a three block grid system B3G3 for simulating this vortex shedding flow with high resolutions. In B3G3 grid system, the computational domain are divided into 3 blocks, which arranged uniformly distribution of coarse (G1), finer (G2), and finest (G3) Cartesian grids, respectively, where G1 denotes a uniform cell with $\Delta x = \Delta y = 0.02h$, G2 denotes a uniform cell with $\Delta x = \Delta y = 0.01h$, and G3 denotes a uniform cell with $\Delta x = \Delta y = 0.005h$. Computational vorticity contours of B3G3 are simulated and plotted on Fig. 5, in Fig. 5 the 3 blocks are also marked. It is seen from Fig. 5 that the vorticity contours are distributing around those block-interfaces smoothly, which shows that the present nested grid method works smoothly even across complex and reversed flow fields. Numerical results of $\overline{C_d}$, C_l, and *St*, grid numbers, and CPU time of B3G3 simulations are collected, these values are listed on Table 2 and compared to those results or values of B1G1, B1G2, and B2D5. All the computations are carried out on a personal computer with Pentium 4 (3.4G) and Fortran 90 compiler. As shown on Table 2 that the computational results are greatly improved over those of B1G1, B1G2, and B2D5. It is expected that computational results are closed to the grid independent results, although further investigations are necessary. For instance, we found that the simulated C_l differences between B1G1, B1G2, and B2D5 to that of B3G3 are up to 39%, 7.2% and 7.9%, respectively. While the present nested grid method improves the coarse single-block simulations greatly, the grid size and CPU time of nested grid

method are still preserved at the coarse-grid level. So far, it can be observed from the numerical tests that our nested grid algorithm can achieve an equivalent accuracy as it obtains on the finest uniform mesh, and at the same time greatly save the CPU time.

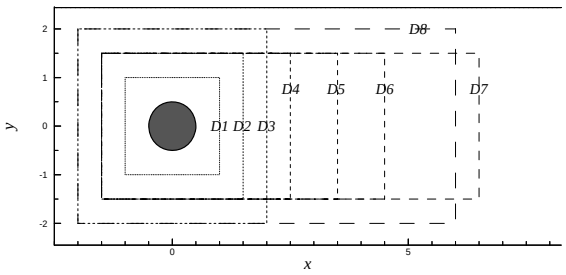


Fig. 4 Refine domain plots of B2D1 to B2D8.

TABLE I COMPUTATIONAL RESULTS OF MEAN DRAG COEFFICIENT AND LIFT COEFFICIENT AMPLITUDE OF VARIOUS 1BLOCK AND 2BLOCK GRID SYSTEMS

Grid	$\overline{C_d}$	Diff. to B1G2	C_l	Diff. to B1G2	Grid number
B1G1	2.608	6.8%	0.510	34.4%	26,250
B1G2	2.797	0.0%	0.777	0.0%	105,000
B2D1	2.759	1.4%	0.687	11.6%	27,850
B2D2	2.785	0.4%	0.760	2.2%	29,850
B2D3	2.796	0.0%	0.779	0.3%	32,650
B2D4	2.791	0.2%	0.768	1.2%	31,050
B2D5	2.791	0.2%	0.771	0.8%	32,250
B2D6	2.791	0.2%	0.770	0.9%	33,450
B2D7	2.791	0.2%	0.770	0.9%	35,850
B2D8	2.792	0.2%	0.772	0.6%	39,050

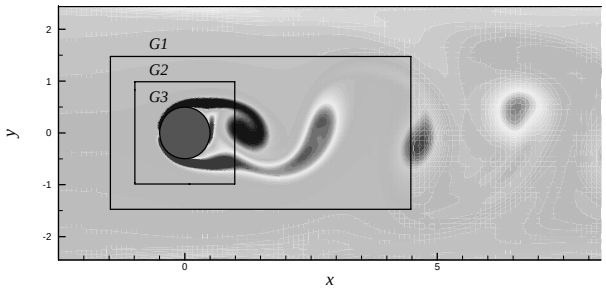


Fig. 5 Computed vorticity contours in the near wake of the flow past a circular cylinder in a channel at *Re*=500 with grid system B3G3.

We then focused on predicting the critical Reynolds number for a blockage ratio of 0.2. The bifurcation analysis of Chen *et al.* (1995) indicates that for this particular blockage ratio the critical Reynolds number is 231.

TABLE II COMPUTATIONAL RESULTS OF MEAN DRAG COEFFICIENT, LIFT COEFFICIENT AMPLITUDE, STROUHAL NUMBER, AND CPU TIME OF VARIOUS 1BLOCK, 2BLOCK, AND 3BLOCK GRID SYSTEMS

Grid system	$\overline{C_d}$ Dif. to B3G3	C_l Dif. to B3G3	St Dif. to B3G3	Grid number	CPU(min)
B1G1	2.608 6.8%	0.510 34.4%	0.277 0.0%	26,250	174
B1G2	2.797 0.0%	0.777 0.0%	0.276 0.3%	105,000	800
B2D5	2.791 0.2%	0.771 0.8%	0.277 0.0%	32,250	227
B3G3	2.828 0.0%	0.837 0.0%	0.277 0.0%	40,100	300

This result has been confirmed by Ye *et al.* (1999). Herein, we devoted to simulated results of B1G1, B1G2, and B3G3. For each grid system, we have performed numerical simulations ranging from a Reynolds number of 225 to 270 in order to pinpoint the critical Reynolds number. From the flow stability analysis of the present study (not shown), it is expected that the first bifurcation of steady flow happens between *Re* = 225 and 270. The bifurcation of flow past circular as well as square cylinders has been clarified as a supercritical pitchfork bifurcation. In order to show the characteristic of the bifurcation, C_l^2 as a function of *Re* is plotted in Fig. 6. Since the linear relation $C_l^2 \propto Re$ holds, the supercritical pitchfork bifurcation from steady flow to vortex shedding flow is confirmed (Peng, 2003). We explore the bifurcation of present vortex shedding flow by investigating C_l^2 vs. *Re* in a similar manner with all the three B1G1, B1G2, and B3G3 grid systems. The value of C_l^2 as functions of *Re* are plotted as shown in Fig. 6. It is noted that Re_{cr} is evaluated as Re_{cr}=230.7, Re_{cr}=234.2, and Re_{cr}=262.6 for B3G3, B1G2, and B1G1, respectively. Comparing with numerical result (Re_{cr}=231) of Chen *et al.* (1995) and Ye *et al.* (1999), it is found that the predicted Re_{cr} of present B3G3 is in very good agreement with previous studies, while error of simulated results of B1G2 and B1G1 are about 1.4% and 13.7%, respectively.

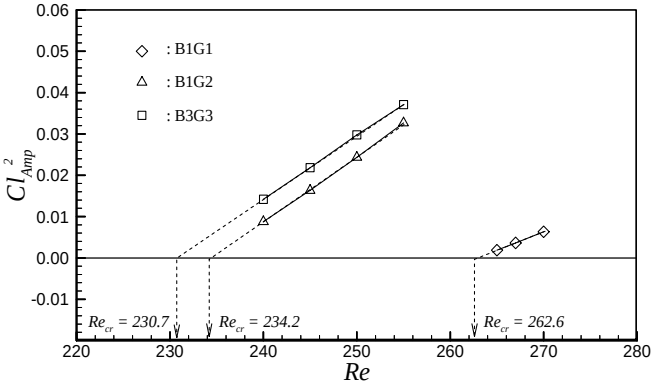


Fig. 6 $C_{l,Amp}^2$ as functions of *Re* and critical Reynolds numbers Re_{cr}.

B. Flow past two circular cylinders with different diameters

The second test case is flow past two cylinders with different diameters. In 1990, Strykowski and Sreenivasan investigated

Open Science Index, Civil and Environmental Engineering Vol:2, No:10, 2008 publications.waset.org/3785/pdf

Open Science Index, Civil and Environmental Engineering Vol:2, No:10, 2008 publications.waset.org/3785/pdf

the suppression of the vortex shedding from a circular cylinder by introducing a small circular cylinder at low Reynolds numbers [7]. Since the experiments of Strykowski and Sreenivasan [7], many experimental and computational studies have been performed in order to control wake flows at supercritical Reynolds numbers. Sakamoto *et al.* (1991) [8] and Sakamoto and Haniu (1994) [9] studied the suppression of the fluid force acting on a square cylinder and a circular cylinder, respectively, at high Reynolds number in subcritical regime. Dalton *et al.* (2001) also simulated the suppression of lift force on a circular cylinder by a small cylinder numerically [10].

Recently, there were numerical studies directed towards the flow past two cylinders of different diameter (Zhao *et al.*, 2005 [11], Delaunay and Kaiktsis, 2001 [12], and Yang *et al.*, 2001 [13]). These numerical studies were either based on the finite element method or finite volume method with body fitted grids. In this study, flow past two circular cylinders of different diameters is investigated numerically. The Navier-Stokes equations are solved using present nested grid method. The aim of this study is to investigate the ability of the present numerical method by solving the suppression of the vortex shedding flow behind the two-cylinder system which was kind of difficult to reach by a Cartesian grid method. The two cylinders considered in this study are shown in Fig. 7. The diameter ratio between the small cylinder and the large cylinder is 0.2. The Reynolds number based on the diameter of the cylinders is 100(80) for the large cylinder and 20(16) for the small cylinder. The gap between the small cylinder and the large cylinder is 2.0 times the diameter of the large cylinder. The position angle of the small cylinder relative to the flow direction ranges from 20^o to 40^o. In all computations, a rectangular computational domain is used. The large cylinder is located at 5D from the inflow boundary. The distance between the large cylinder and the outgoing boundary is 20D. The two lateral boundaries are located at 7D away from the large cylinder. The effects of the position angle of the small cylinder on drag and lift coefficients, pressure distributions around the cylinders, the vortex shedding frequencies from the two cylinders and flow characteristics are investigated.

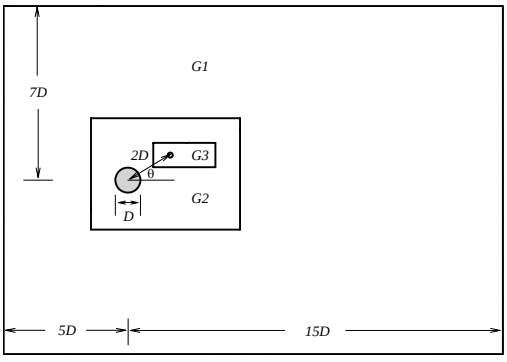


Fig. 7 Configuration of flow past two circular cylinders with different diameters.

To validate the numerical model, uniform flow past a singular cylinder for the Reynolds number ranging from 10 to 200 is first simulated. The computational domain of $25\mathrm{D}\times 14\mathrm{D}$ was divided into two blocks. In the outer block, G1 grids with $\Delta x=\Delta y=0.1D$ were used to compute the outer flow. In the inner block of $6D\times 4D$ (please see Fig. 9(a)), G2 grids with $\Delta x=\Delta y=0.05D$ were used to compute the flow around the cylinder. In Fig. 8, the computed time averaged drag coefficients for Reynolds number from 10 to 100 are compared with the finite difference results by Lei *et al.* [14] and the finite element results by Ghao *et al.* [11]. The difference between the numerical results in Fig. 8 is small.

The numerical model is then applied to study the flow past two circular cylinders as shown in Fig. 9. Before we discuss the computational results of the suppression of vortex shedding flow, a typical computational mesh for the flow is shown in Fig. 9(b), the computational domain of $25\mathrm{D}\times 14\mathrm{D}$ is herein divided into three blocks. In the outer block, G1 grids with $\Delta x=\Delta y=0.1D$ were used to compute the flow. In the middle block of $6D\times 4D$, G2 grids with $\Delta x=\Delta y=0.05D$ were used to compute the flow around the main cylinder. In the inner block of $2.5D\times 1D$, G3 grids with $\Delta x=\Delta y=0.025D$ were used to compute the flow around the control cylinder. The total cell number is 45200.

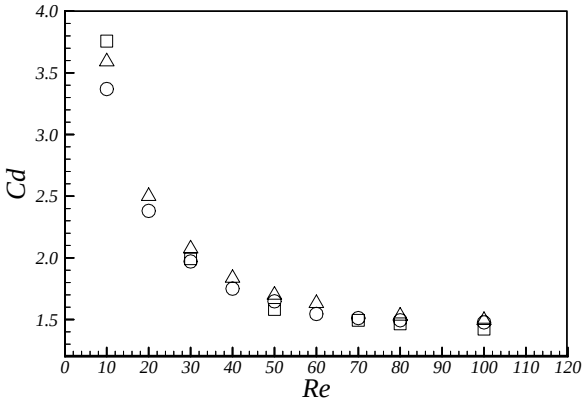


Fig. 8 Comparison of the computed mean drag coefficient, square symbols: this study, circular symbols: Lie *et al.* (2000), triangle symbols: Zhao (2005).

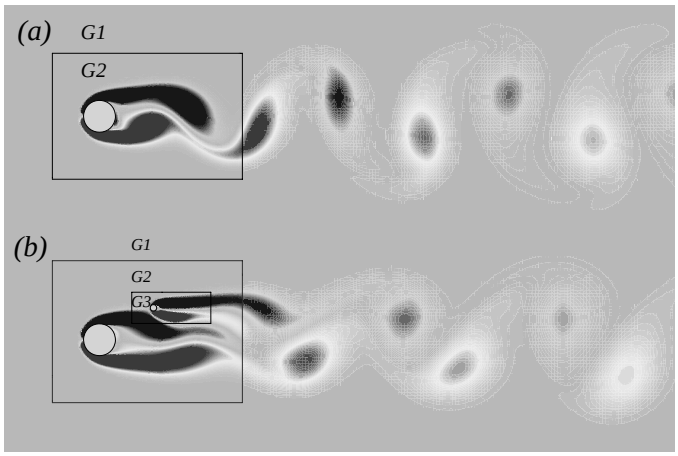


Fig. 9 Computed iso-vorticity plots and nested grid of the flow past (a) a single cylinder and (b) two cylinders with different diameters.

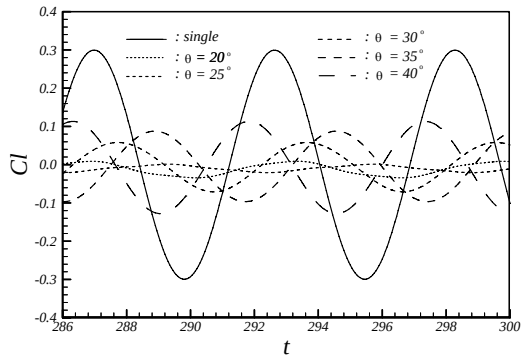


Fig. 10 Computational C_l of flow past two cylinders of different diameters at *Re*=100 with various position angles.

Fig. 10 shows the computed lift coefficient history on the main cylinder at Reynolds number equals to 100. For the purpose of comparison, the lift coefficient on a single cylinder is also plotted. It can be seen from Fig. 10 that the effect of the small cylinder on the mean force coefficients is significant. Specifically, the suppression of the mean lift on the large cylinder is significant for all the computed cases with $20^o \le \theta \le 40^o$ (Fig. 10). The maximum suppression ratio of $C_{l,Amp}$ on the main cylinder is found around $\theta = 25^o$. The mean lift and drag coefficients, amplitude of lift coefficients, and *St* of the main cylinder as functions of the position angle of the small cylinder were collected and listed on Table 3. From Table 3, it is seen that the maximum suppression ratio of $C_{l,Amp}$ on the main cylinder is about 4%.

To investigate the effect of the control cylinder on the suppression of the vortex shedding clearly, Fig. 11 showed vortisity distributions at *Re*=80 and $\theta = 30^o$. It is clear that when there is control cylinder exist availably, the vortex shedding pattern is suppressed apparently.

This numerical study investigated the suppression of vortex shedding of the main cylinder on the effect of a small control cylinder in the wake. The application gave reasonable and satisfactory results comparable with the available experimental and numerical results in literature and showed the feasibility of the present model.

TABLE III COMPUTATIONAL RESULTS OF FLOW PAST TWO CYLINDERS OF DIFFERENT DIAMETERS AT *RE*=80 WITH VARIOUS POSITION ANGLES

θ	$\overline{C_d}$	$\overline{C_l}$	C_l	$\frac{C_l}{C_{l,single}}$	*St*
20	1.335	-0.008	0.0127	5.4%	0.142
25	1.337	-0.008	0.0	0.0%	0.0
30	1.349	-0.005 3	0.0035	1.5%	0.152
35	1.371	-0.004 3	0.0377	16.2%	0.160
40	1.389	-0.005 8	0.0540	23.1%	0.166
single	1.465	0.0	0.2333	100%	0.166

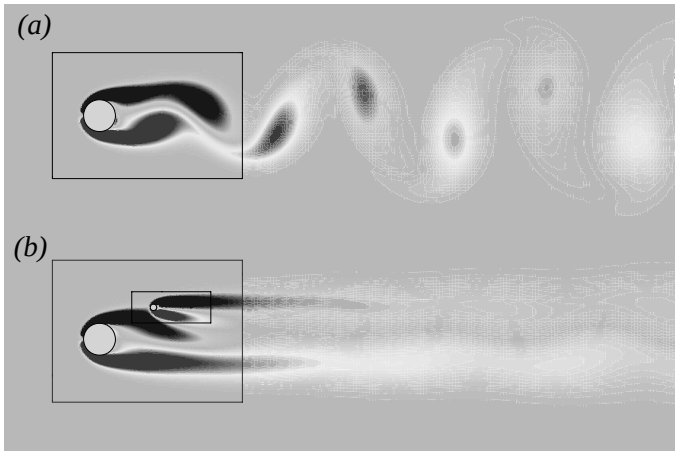


Fig. 11 Iso-vorticity plots of the flow past (a) a single cylinder and (b) two cylinders with different diameters at *Re*=80 and $\theta = 30^o$.

ACKNOWLEDGMENT

This work was supported by the National Science Council of the Republic of China (Taiwan) under contract no. NSC 97-2221-E-260-017- .

REFERENCES

[1] L. Chacon, and G. Lapenta, "A fully implicit, nonlinear adaptive grid strategy," *J. Comput. Phys.*, vol. 212, pp 703-717, 2006.
[2] H. Ding, and C. Shu, "A stencil adaptive algorithm for finite difference solution of incompressible viscous flows," *J. Comput. Phys.*, vol. 214, pp 397-420, 2006.
[3] Y. F. Peng, Y. H. Shiau, and R. R. Hwang, "Transition in a 2-D lid-driven cavity flow," *Comput. & Fluids*, vol. 32, pp 337-352, 2003.
[4] J. F. Ravoux, A. Nadim, and H. Hariri, "An Embedding Method for Bluff Body Flows: Interactions of Two Side-by-Side Cylinder Wakes," Theo. Comput. Fluid Dyn., vol. 16, pp. 433-466, 2003
[5] T. Ye, R. Mittal, H. S. Udaykumar, and W. Shyy, "An accurate Cartesian grid method for viscous incompressible flows with complex immersed boundaries," *J. Comput. Phys.*, vol. 156, pp 209-240, 1999.
[6] J. H. Chen, W. G. Pritchard, and S. J. Tavener, "Bifurcation for flow past a cylinder between parallel planes," *J. Fluid Mech.*, vol. 284, pp 23-52, 1995.
[7] B. J. Strykowski, and K. R. Sreenivasan, "On the formation and suppression of vortex 'shedding' at low Reynolds numbers," *J. Fluid Mech.*, vol. 218, pp 71-107, 1990.
[8] H. Sakamoto, K. Tan, and H. Haniu, "An optimum suppression of fluid forces by controlling a shear layer separated from a square prism," *J. Fluid Eng.*, vol. 113, pp 183-189, 1991.
[9] H. Sakamoto, and H. Haniu, "Optimum suppression of fluid forces acting on a circular cylinder," *J. Fluid Eng.*, vol. 116, pp 221-227, 1994.
[10] C. Dalton, Y. Xu, and J. C. Owen, "The Suppression of lift on a circular cylinder due to vortex shedding at moderate Reynolds numbers," *J. Fluid Struct.*, vol. 15, pp 61-128, 2001.
[11] M. Zhao, L. Cheng, B. Teng, and D. Liang, "Numerical simulation of viscous flow past two circular cylinders of different diameters," *Appl. Ocean Res.*, vol. 27, pp 39-55, 2005.
[12] Y. Delaunay, and L. Kaiktsis, "Control of circular cylinder wakes using base mass transpiration," *Phys. Fluid*, vol. 13, pp 3285-302, 2001.
[13] D. L. Young, J. L. Huang, and T. I. Eldho, "Simulation of laminar vortex shedding flow past cylinders using a coupled BEM and FEM model," *Comput. Method Appl. Mech. Eng.*, vol. 190, pp 5975-5998, 2001.
[14] C. Lei, L. Cheng, K. and Kavanagh, "A finite difference solution of the shear flow over a circular cylinder," *Ocean Eng*, vol. 27, pp 271-90, 2000.

Open Science Index, Civil and Environmental Engineering Vol:2, No:10, 2008 publications.waset.org/3785/pdf