INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2002; **39**:441–463 (DOI: 10.1002/fld.339)

Parallel computation of viscous incompressible flows using Godunov-projection method on overlapping grids

H. Pan[‡] and M. Damodaran[*,†,§]

Division Of Thermal and Fluids Engineering, School of Mechanical and Production Engineering, Nanyang Technological University, Nanyang Avenue, Singapore 639798

SUMMARY

The Godunov-projection method is implemented on a system of overlapping structured grids for solving the time-dependent incompressible Navier–Stokes equations. This projection method uses a second-order fractional step scheme in which the momentum equation is solved to obtain the intermediate velocity field which is then projected on to the space of divergence-free vector fields. The Godunov procedure is applied to estimate the non-linear convective term in order to provide a robust discretization of this terms at high Reynolds number. In order to obtain the pressure field, a separate procedure is applied in this modified Godunov-projection method, where the pressure Poisson equation is solved. Overlapping grids are used to discretize the flow domain, as they offer the flexibility of simplifying the grid generation around complex geometrical domains. This combination of projection method and overlapping grid is also parallelized and reasonable parallel efficiency is achieved. Numerical results are presented to demonstrate the performance of this combination of the Godunov-projection method and the overlapping grid. Copyright © 2002 John Wiley & Sons, Ltd.

KEY WORDS: numerical simulation; viscous flow; unsteady flow; projection method; overlapping grid; parallel computation

1. INTRODUCTION

The primary motivation for this study stems from the need to develop suitable numerical methods for solving the unsteady incompressible Navier–Stokes (INS) equations to model the hydrodynamic environment and to estimate the hydrodynamic forces for the control and navigation of underwater robotic vehicles (URVs), which are routinely being used for underwater applications. A number of researchers such as Yuh [1], Kalske [2] and Fossen [3] have contributed towards the estimation of forces acting on the URVs by basing their predictions on potential flow theory which is a simplified linear model for hydrodynamic flows. The hydrodynamic forces acting on the vehicle were divided artificially into several parts that can

* Correspondence to: M. Damodaran, Division of Thermal and Fluids Engineering, School of Mechanical and Production Engineering, Nanyang Technological University, Nanyang Avenue, Singapore 639798, Singapore
† E-mail: mdamodaran@ntu.edu.sg
‡ Currently Research Engineer, Institute of High Performance Computing, Singapore.
§ Associate Professor and Singapore-MIT Alliance Faculty Fellow.

Received April 2001
Revised November 2001

be modelled individually by using empirical formulae based on a combination of experimental data and linear model results. In this work, it is proposed that numerical methods for solving INS be used to simulate the flow field around the underwater rigid bodies and hydrodynamic forces acting on the body are estimated by integrating the pressure and viscous forces along the body's surface.

It is reasonable to assume that the fluid around the underwater vehicle is incompressible and the vehicle is a rigid body. The main computational difficulty for simulating the incompressible flow by using the numerical method arises from the fact that the continuity equation contains only velocity components and there is no obvious link with the pressure as in the case of compressible flow, where the density carries on the link. One method, which is the focus of the current investigation for solving these equations, is the second-order Godunov-projection method introduced by Bell *et al.* [4, 5] and developed by Bell *et al.* [6] and Almgren *et al.* [7]. In this projection method, a structured grid is used to discretize the flow domain in the physical space, which can be transformed to the computational space, which is a unit square in two-dimensional space or a unit cube in three-dimensional space through a mapping Φ given by $\Phi = \partial(x, y, z)/\partial(\xi, \eta, \zeta)$, where (x, y, z) stands for the co-ordinates of the physical space and (ξ, η, ζ) denotes the co-ordinates of the computational space. The unsteady incompressible Navier–Stokes equations in the generalized curvilinear co-ordinate system are defined as follows:

$$\nabla_\xi \cdot \bar{\mathbf{U}} = 0 \tag{1}$$

$$\mathbf{U}_t + \frac{1}{J}(\bar{\mathbf{U}} \cdot \nabla_\xi)\mathbf{U} = \frac{\varepsilon}{J}\nabla_\xi \cdot \left[\frac{1}{J} T T^t \nabla_\xi \cdot \mathbf{U}\right] - \frac{1}{J} T^t \frac{\nabla_\xi \phi}{\rho} \tag{2}$$

where $J = \det|\Phi|$, $T = J\Phi^{-1}$ and $\bar{\mathbf{U}} = T\mathbf{U}$, ε is the kinematic viscosity, $\mathbf{U}$ represents the velocity field and ϕ is the pressure.

This projection method contains fractional time steps and is carried out in several steps. First, momentum equation (2) is solved with a lagged pressure term to determine the intermediate velocity field, which does not satisfy continuity equation (1). Then, the intermediate velocity field is decomposed into divergence-free and curl-free components which are the new velocity field and the update for the pressure, respectively. The Godunov procedure is incorporated to differentiate the convective term in order to provide a robust discretization scheme so that the restriction of cell Reynolds number can be overcomed.

In order to simplify the grid generation task for complex flow domains that are common for the underwater applications, a system of overlapping grids is used to discretize the computational domain. The overlapping grid system consists of several component grids that overlap each other and the union of the component grids covers the whole region over which the computation is carried out. Each component grid can be generated separately and has its own mapping function. The benefits of using an overlapping system of grids are many in that it facilitates a smooth transformation for each component grid, simplifies the task of grid generation in complex geometric domains and reduces the computational overhead for solving moving boundary problems, which will be addressed in another paper. The methodology for generating the overlapping grids for the computations reported in this work follows closely the method outlined in Reference [8].

In this paper, the Godunov-projection method is implemented on a system of overlapping grids to simulate the unsteady incompressible flows. In order to reduce the computational

costs for solving the three-dimensional problems, parallel computation is employed so that the computational power of modern high-performance computer can be used efficiently to overcome the computational overheads associated with the simulation.

In the subsequent sections of this paper, details of the Godunov-projection method are presented to cover the procedures for temporal discretization, spatial discretization and projection. This is then followed by a brief outline of the implementation of the algorithm on a system of overlapping structured grids and the parallelization strategies adopted are introduced. Selected numerical simulations of a number of two- and three-dimensional problems are then presented and discussed.

2. GODUNOV-PROJECTION METHOD

The Godunov-projection method of Bell *et al.* [4] applies high-order upwind schemes to provide a robust differencing scheme for the convective terms in the INS equations. The Godunov procedure, which was introduced for gas dynamics by Colella [9], is incorporated in this method.

The implementation of this algorithm is carried out in three steps. In the first step, the second-order Godunov method is used to approximate the conservative differences of the non-linear convective terms $1/J(\bar{\mathbf{U}} \cdot \nabla_\xi)\mathbf{U}$. In the second step, the intermediate velocity field is obtained by solving momentum equation (2) alone and by omitting the solenoidal nature of the velocity field. Finally, an approximate projection is performed to restore the divergence-free velocity field, which satisfies Equation (1) approximately, and subsequently the hydrodynamic pressure is also updated. These steps are outlined briefly below.

2.1. Temporal discretization

The second-order fractional step formulation described in Reference [5] is used for temporal discretization. A vector field $\mathbf{V}$ can be uniquely decomposed into a divergence-free component and a gradient of a scalar field as follows:

$$\mathbf{V} = \mathbf{V}^{\mathrm{d}} + \nabla\psi \tag{3}$$

where $\mathbf{V}^{\mathrm{d}}$ is a divergence-free vector field and ψ is a scalar. Hence, a projection operator $\mathbf{P}$ can be defined as $\mathbf{PV} = \mathbf{V}^{\mathrm{d}}$ and $\nabla\psi = (I - \mathbf{P})\mathbf{V}$. By using this projection, Navier–Stokes equations (1) and (2) can be written as follows:

$$U_t = \mathbf{P}\left(\frac{\varepsilon}{J}\nabla_\xi \cdot \left[\frac{1}{J} TT^t \nabla_\xi \cdot U\right] - \frac{1}{J}(\bar{U} \cdot \nabla_\xi)U\right) \tag{4}$$

By applying the Crank–Nicholson scheme for temporal discretization, Equation (4) takes the following discretized form:

$$\frac{\mathbf{U}^{n+1} - \mathbf{U}^n}{\Delta t} = \mathbf{P}\left(\frac{\varepsilon}{2J}\nabla_\xi \cdot \left[\frac{1}{J} TT^t \nabla_\xi \cdot (\mathbf{U}^{n+1} + \mathbf{U}^n)\right] - \frac{1}{J}(\bar{\mathbf{U}} \cdot \nabla_\xi)\mathbf{U}^{n+1/2}\right) \tag{5}$$

In view of the non-local behaviour of the projection, the linear algebra problem associated with solving this equation would be very costly. As an economic alternative, a fractional step method can be applied where lagged pressure field is used for the computation. An intermediate velocity field $\mathbf{U}^*$ is computed by solving the momentum equation along with lagged pressure and then the projection is applied on the intermediate velocity field for obtaining the divergence-free velocity field $\mathbf{U}^{n+1}$ and update of the pressure field. A number of forms for projection exist and according to the analysis carried out by Rider [10], the pressure form of projection is the most robust as there is no accumulation of the error associated with the no divergence assumption. In view of this, the intermediate velocity field is computed by solving the following equation and subsequently corrected by the gradient of pressure:

$$\Delta t^{-1}(\hat{U}^{*,n+1} - U^n) = \frac{1}{2}J^{-1}\varepsilon\nabla_\xi \cdot [J^{-1}TT^t\nabla_\xi \cdot (\hat{U}^{*,n+1} + U^n)] - J^{-1}[(\bar{U} \cdot \nabla_\xi)U]^{n+1/2} - J^{-1}T^t\left(\frac{\nabla_\xi \phi}{\rho}\right)^{n-1/2} \tag{6}$$

then

$$U^{*,n+1} = \hat{U}^{*,n+1} + \Delta t J^{-1}T^t\left(\frac{\nabla_\xi \phi}{\rho}\right)^{n-1/2} \tag{7}$$

Next a projection is applied for decomposing the vector field into a divergence-free component and a curl-free component so as to obtain a new velocity field $\mathbf{U}^{n+1}$ and to update the pressure gradient $J^{-1}T^t(\nabla_\xi \phi/\rho)^{n+1/2}$.

2.2. Spatial discretization

The spatial discretization of momentum equation (2) is based on a cell-centred approximation since this arrangement provides the most natural way for implementing the Godunov method. The diffusion term can be discretized by using the standard second-order central difference in the computational space and the non-linear convective term is discretized by the second-order Godunov method in this method in order to provide the robust approximation.

Figure 1 shows a schematic of a three-dimensional grid with the co-ordinate indices defining cell centres and cell faces. In order to compute the flux on the faces of a cell, the velocity on the faces is extrapolated from the values computed at the cell centres. Since this method is also second-order accurate in time direction, the flow variables are extrapolated both in space and time, where the Taylor series expansion is used. After combining the derivative in the normal direction, the following equations are used to do the extrapolation of the velocity $\mathbf{U}$ on the faces $(i + \frac{1}{2}, j, k)$

$$\mathbf{U}_{i+1/2}^{n+1/2,L} = \mathbf{U} + \left[\frac{\Delta\xi}{2} - \frac{\Delta t}{2J}\bar{u}\right]\mathbf{U}_\xi - \frac{\Delta t}{2J}\left(\bar{v}\mathbf{U}_\eta + \bar{w}\mathbf{U}_\zeta\right) + \varepsilon\left[\frac{\Delta t}{2J}\nabla_\xi\left(\frac{1}{J}TT^t\nabla_\xi \cdot \mathbf{U}\right)\right] - \frac{\Delta t}{2J}T^t\frac{\nabla_\xi \phi}{\rho} \tag{8}$$

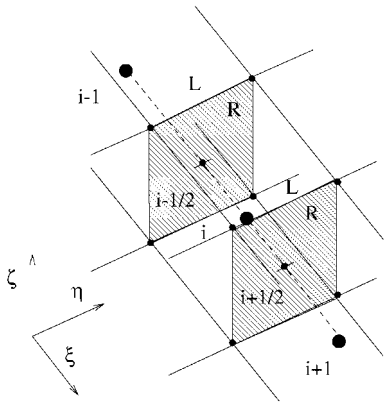


Figure 1. Location of cell- and face-centred variables.

$$\mathbf{U}_{i+1/2}^{n+1/2,L} = \mathbf{U}_{i+1} - \left[\frac{\Delta\xi}{2} + \frac{\Delta t}{2J}\bar{u}_{i+1}\right](\mathbf{U}_\xi)_{i+1} - \frac{\Delta t}{2J}(\bar{v}\mathbf{U}_\eta + \bar{w}\mathbf{U}_\zeta)_{i+1} + \varepsilon\left[\frac{\Delta t}{2J}\nabla_\xi\left(\frac{1}{J}TT^t\nabla_\xi\cdot\mathbf{U}_{i+1}\right)\right] - \frac{\Delta t}{2J}T^t\left(\frac{\nabla_\xi\phi}{\rho}\right)_{i+1} \tag{9}$$

where $\mathbf{U} = \mathbf{U}_{i,j,k}^n$ and $\mathbf{U}_{i+1/2}^{n+1/2} = \mathbf{U}_{i+1/2,j,k}^{n+1/2}$, etc. are implied by the subscript indices. A slope limiter, such as superbee and minmod limiter, can be used to estimate the gradient of the flow variables in the cell in each co-ordinate direction. The variable has two values, one of which is extrapolated from left cell and the other from the right cell. The extrapolation is handled by using a Riemann solver, where an upwind averaging is applied. It is also possible to employ an approximate Riemann solver presented in Reference [11]. It should be noted that the pressure terms appearing in Equations (8) and (9) are not used in this extrapolation in view of the weak instability which had been observed by Lai *et al.* [12] if lagged pressure terms and if the CFL number is greater than 0.5. Hence the marker-and-cell (MAC) projection method is used so that the resulting velocity field on the staggered grid is divergence-free. The convective terms are discretized as follows:

$$\left(\frac{1}{J}[(\bar{\mathbf{U}}\cdot\nabla_\xi)\mathbf{U}]\right)^{n+1/2} \approx \frac{(\bar{u}_{i+1/2} + \bar{u}_{i-1/2})}{2J}\frac{(\mathbf{U}_{i+1/2} - \mathbf{U}_{i-1/2})}{\Delta\xi} + \frac{(\bar{v}_{j+1/2} + \bar{v}_{j-1/2})}{2J}\frac{(\mathbf{U}_{j+1/2} - \mathbf{U}_{j-1/2})}{\Delta\eta} + \frac{(\bar{w}_{k+1/2} + \bar{w}_{k-1/2})}{2J}\frac{(\mathbf{U}_{k+1/2} - \mathbf{U}_{k-1/2})}{\Delta\zeta} \tag{10}$$

where the superscript $n + 1/2$ has been omitted from the terms in the RHS of the equation for convenience.

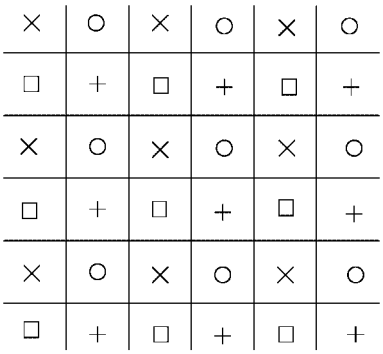


Figure 2. The four decoupled grids for a two-dimensional grid.

2.3. Projection

The final step of this projection algorithm decomposes the vector field into a divergence-free component and a gradient of scalar quantity. The projection is defined by the divergence operator D and gradient operator G as in References [13, 5], which satisfy the property:

$$(D\mathbf{V}, \psi)_{\mathrm{s}} = -(\mathbf{V}, G\psi)_{\mathrm{v}} \tag{11}$$

where the terms $(\cdot,\cdot)_{\mathrm{s}}$ and $(\cdot,\cdot)_{\mathrm{v}}$ represent the appropriate inner products on the discrete spaces of scalars and vectors, respectively. This condition guarantees that the numerical projection is orthogonal. Exact discrete projection utilizes the central difference for both D and G operator and the discrete Laplacian operator derived decouples the grid [14] as shown in Figure 2, where a two-dimensional grid is decoupled into four distinct subgrids. Almgren *et al.* [7] introduced an approximate projection, which kept the same discrete form of the Poisson equation so that it could exploit the advantage of the fast solvers. However, the resulting divergence-free component is not really divergence-free even though its divergence is of the order $O(h^2)$ as reported in Almgren *et al.* [7].

Since the pressure form of projection is used in this work, $\mathbf{U}^{*,n+1}$ is projected to extract the following linear system:

$$L_{st}\psi = D\left(\frac{\mathbf{U}^{*,n+1}}{\Delta t}\right) \tag{12}$$

where L_{st} represents the standard difference Laplacian operator and the equations used to compute the new velocity field and the gradient of pressure are as follows:

$$\frac{\mathbf{U}^{n+1}}{\Delta t} = \frac{\mathbf{U}^{*,n+1}}{\Delta t} - G\psi \tag{13}$$

$$\frac{1}{J}\,T^t\frac{\nabla_\xi \phi^{n+1/2}}{\rho} = G\psi \tag{14}$$

2.4. Estimation of pressure field

It is important to estimate the pressure field from the velocity field so that forces induced by fluid flow can be estimated. It is noted that as the pressure at half time step is computed in the numerical scheme outlined in the previous sections, the pressure at each time level is not available. Hence, a separate procedure is incorporated into the algorithm to obtain the pressure field after computing the velocity field. The pressure Poisson equation which is obtained by taking the divergence on both sides of momentum equation (2) and also by considering the satisfaction of the divergence-free property of the velocity field as shown below

$$\frac{\nabla \cdot (\nabla \phi)}{\rho} = -\nabla(\mathbf{U} \cdot \nabla)\mathbf{U} + \nabla \cdot \mathbf{F} \tag{15}$$

is solved in this procedure to obtain the pressure field.

3. OVERLAPPING GRID

A system of overlapping grids is used to simplify the grid generation task for geometrically complicated flow domains. Even though the geometry of the test cases considered in this work are relatively simple, the aim of this work is to demonstrate the implementation of the algorithm on overlapping grids and to gain confidence before applying this method to more complex geometries. The overlapping grid consists of several component grids, which overlap each other. Since it is not required to match on the interior boundaries between the component grids, the grid generation can be more flexible so that the cost for constructing the structured grid for complex flow simulation problems can be greatly reduced. However, the overlap region between the component grids must be large enough so that the flow information can be transferred between component grids correctly. Chesshire and Henshaw [8] described a method for generating such overlapping grids and a procedure for determining interpolation coefficients to enable flow variable transfer and interpolation. The grid generator, Ogen [15], is used to generate the overlapping grids in this work. The internal boundary of component grid consists of the interpolation cells whose flow variable values should be extracted from other component grids with which it overlaps.

For overlapping grids, each component grid has its own mapping function Φ. Hence, Equations (1) and (2) can be set up on each component grid and the Godunov-projection algorithm outlined here can be applied on each component grid provided the values on the interior boundary are known. Hence, the flow fields on the component grids are coupled to the flow variables on the interior boundary cells. In this work, a high-order interpolation is used to compute the flow variables on the interior boundary cells. According to Chesshire and Henshaw [8], a third-order interpolation is required in order to retain the overall second-order accuracy of the Godunov-projection method and the incompressible Navier–Stokes equations.

Figure 3 shows the method for interpolating on a set of two two-dimensional overlapping grids, where the interpolation is performed in the computational space. The interpolation cell of grid B, which is labelled by the plus symbol, is identified during the phase of grid generation. Its co-ordinates in the physical space, $(x, y)_{i,j}$, are mapped into the computational space of grid A as $(\xi, \eta)_{i,j}$. Hence, the location of its interpolation stencil can be easily determined since the grid in the unit computational space is uniformly distributed. The interpolation stencil is

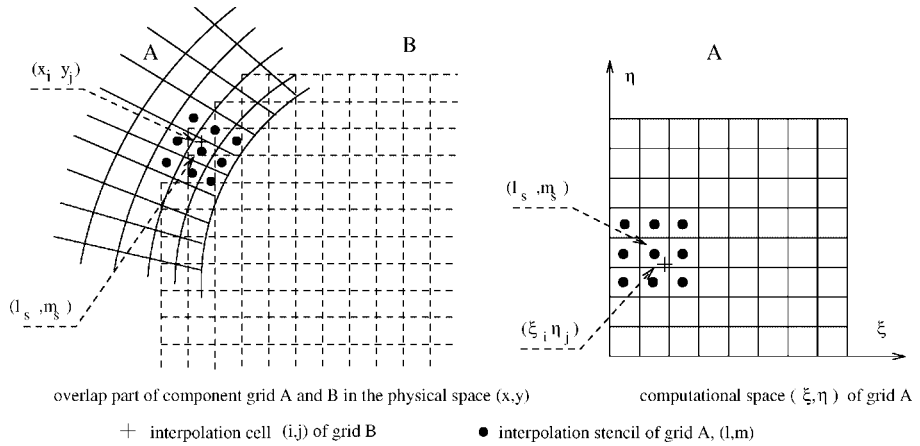


Figure 3. Interpolation stencil.

denoted by the indices of the central cell of the stencil (l_s, m_s). Then, the interpolation of variables can be estimated using the following equation:

$$v_{B(i,j)} = \sum_{l=-1}^{1} \sum_{m=-1}^{1} \alpha_{l,m,i,j} v_{A(l_s+l, m_s+m)} \tag{16}$$

where $v_{B(i,j)}$ is the variable at the interpolation cell and the $v_{A(l_s+l, m_s+m)}$ are the values at the cells belonging to the interpolation stencil as shown in Figure 3. The interpolation weight, $\alpha_{l,m,i,j}$, is computed from the co-ordinates $(\xi, \eta)_{i,j}$ by using the bi-quadratic interpolation. Similar interpolation method is employed for the computations on the three-dimensional overlapping grids.

The Godunov-projection method is applied on the overlapping grid by employing the original difference method on every component grid and using the interpolation equation on the interior boundary cells. The resulting system of linear equations from all the cells of the overlapping grid must be solved together in the construction of the evolving numerical flow field.

4. PARALLEL COMPUTATION

In order to overcome the computational overhead of the three-dimensional numerical flow simulations, a combination of the Godunov-projection method and the overlapping grid is parallelized. Generally, the parallelization is accomplished by partitioning the computational domain and distributing the partitioned sub-domains among the processors. In this work, the partition is achieved by partitioning the overlapping grids ensuring good load balancing. Each component grid is partitioned individually into a number of parts and distributed among the processors. Therefore, each processor has a collection of grid partitions that belong to different component grids. Each component grid is partitioned in a scalable manner where the grid is divided as equally as possible and the interface between grid partitions is minimized so that

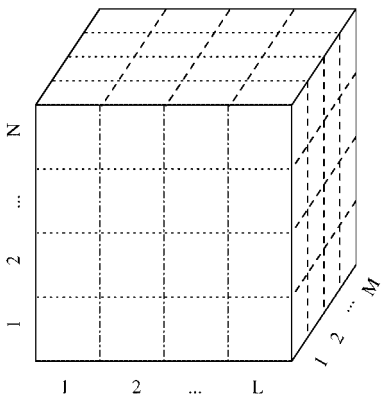


Figure 4. Grid partitioning of a component grid for scalability and load balancing. Total number of partitions is $L \times M \times N$, which should equal the number of processors.

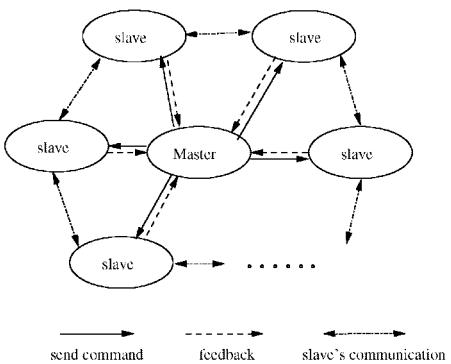


Figure 5. The structure of master/worker program model.

the amount of data exchange during the computation is minimized as well and the efficiency of the parallel algorithm can be enhanced. In this work, a boxwise partition is employed to partition the component grid as shown in Figure 4, where the component grid can be divided in all three co-ordinate directions and the number of the grid partition is equal to the number of the employed processors.

In order to make the current algorithm portable, a message passing standard, MPI [16], has been used to implement the numerical simulation algorithm on a 64 processor SGI Origin 2000 High Performance Computing (HPC) System to perform the message passing among the different processors which process the partitioned computational domain. The master/slave model shown in Figure 5 has been used as the model for implementing the algorithm in parallel by exploiting the scalable parallel computer architecture of the SGI Origin 2000 HPC Platform, where one processor (master) controls the execution of the overall computation including I/O and other processors (slaves) execute the flow computation on their partitioned computational

domain simultaneously. Basically, the slaves run in a loop awaiting the commands from the master. Upon completion of the job, the slaves signal the master so that subsequent actions can be initiated by the master. The slave is also allowed to communicate with other slaves directly under the supervision of the master so that unnecessary message passing is eliminated. The synchronization of processes is fulfilled by using blocked message sending and receiving of MPI. The performance of this parallelized program executed in this mode on the SGI Origin 2000 HPC system will be demonstrated in the subsequent sections for a sample three-dimensional problem.

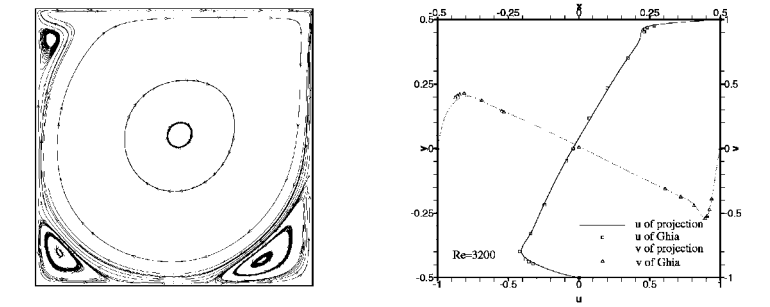
5. NUMERICAL RESULTS

In order to demonstrate the performance of the Godunov-projection method combined with the overlapping grids, several test problems are numerically solved. The test problems considered are the lid-driven square cavity flow, uniform flow past a cylinder and flow past a sphere. The motivation for this is to ensure that the basic algorithm works correctly in single as well as overlapping grids.

5.1. Lid-driven flow in a square cavity

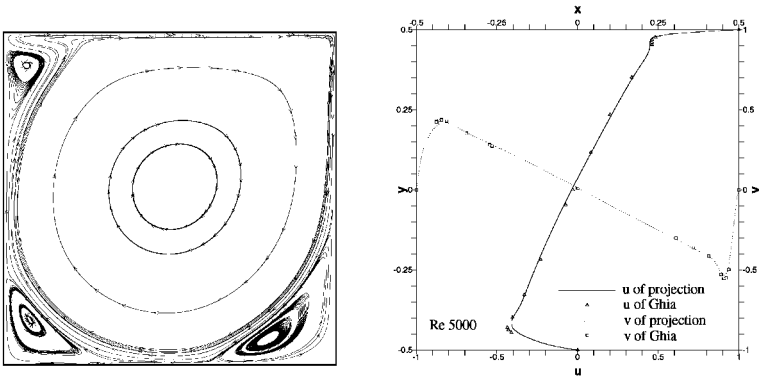
Lid-driven flow in a unit square cavity is a widely used benchmark test to verify the calculation of incompressible flow. In this work, cavity flows corresponding to Reynolds number $Re = 3200,\ 5000$ and $10\,000$ are computed. Specific details on the performance of the algorithm, convergence, etc. and results for other Reynolds numbers, $Re = 100,\ 400$ and 1000, can be found in Reference [17]. A non-uniformly distributed grid of 64×64 points has been used for these calculations where the grid points are clustered along the wall boundaries. The initial field is set to a stationary flow field everywhere except on the lid boundary where the velocity corresponds to the velocity of the lid motion. The flow fields at low Reynolds numbers are used as the initial conditions for simulating high Reynolds numbers flows to reduce computational costs. The computed results are shown in Figure 6, where the computed streamlines are shown in figures (a), (c) and (e) and the velocity profiles along the central line of cavity are displayed in figures (b), (d) and (f). From the streamline plots, it is clear that four vortices, i.e. one primary central vortex, two secondary vortices one each in the lower left, and right corners and another secondary vortex in the upper left corner have manifested in the flow field at Reynolds number $Re = 3200$ and 5000, and these are almost steady in nature. At Reynolds number $Re = 10\,000$, the flow becomes more complex and drifts further from the steady state than previous two cases. A tertiary vortex can be seen in the lower right corner in Figure 6. These computed flow phenomena are similar to the results reported Reference [18]¶. Further examination is done by comparing the u component of velocity along the central vertical line and v along the central horizontal line. This comparison is given in the right column of Figure 6. It is noted that minor discrepancies exist at high Reynolds number. This is caused by the coarse grids and can be remedied by refining the grid. (Ghia used 257×257 grid points for these cases.) The locations of the vortices are presented in Table I and compared with

¶Ghia *et al.* [18] calculated the flow in a square cavity at the same Reynolds numbers with a fine grids, which have 129×129 and 257×257 grid points.



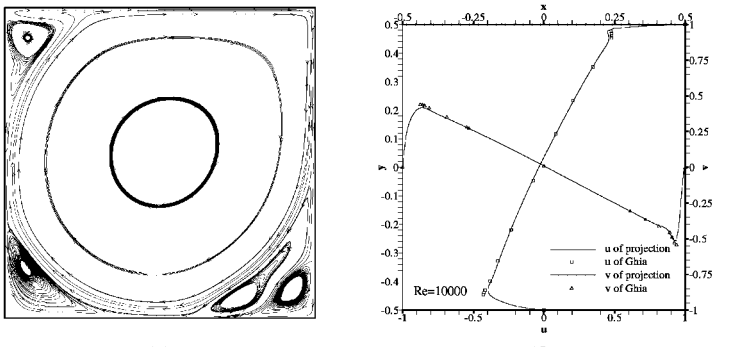
(a) (b)

$Re = 3,200$



(c) (d)

$Re = 5,000$



(e) (f)

$Re = 10,000$

Figure 6. Lid-driven square cavity flows for Reynolds numbers $Re = 3200$, 5000 and 10 000 on a 64×64 grid. Figure in the left column is streamline plot and velocity profile along the central, vertical and horizontal lines in the right column.

Table I. The location of vortices of lid-driven square cavity flows at different Reynolds numbers.

Vortex	$Re = 3200$	$Re = 5000$	$Re = 10\,000$
Present method			
PV	0.5197, 0.5444	0.5142, 0.5377	0.5132, 0.5376
BLV	0.0819, 0.1160	0.0748, 0.1321	0.0614, 0.1604
BRV	0.8182, 0.0869	0.7968, 0.0738	0.7606, 0.0603
TV	0.0523, 0.8937	0.0647, 0.9048	0.0701, 0.9064
SBR	—	—	0.9301, 0.0820
Ghia et al. [18]			
PV	0.5165, 0.5469	0.5117, 0.5352	0.5117, 0.5333
BLV	0.0859, 0.1094	0.0703, 0.1367	0.0586, 0.1641
BRV	0.8125, 0.0859	0.8086, 0.0742	0.7656, 0.0586
TV	0.0547, 0.8984	0.0625, 0.9102	0.0703, 0.9141
SBR	—	—	0.9336, 0.0625

the values obtained by Ghia. In this table, PV represents the primary vortex, BLV and BRV represent the lower left and right vortex, respectively, TV is the upper vortex and SBR refers to the second lower right vortex. Except for the second lower right vortex, the positions of vortices calculated by using the projection method at different Reynolds numbers are in close agreement with that obtained by Ghia. In reality, the flows in the lid-driven square cavity at high Reynolds number are unsteady. The differences that occur for the second lower right vortex are caused by the coarse grid that is used and the computed quasi-steady flow field. It can be concluded that the results obtained by using current projection method is accurate and are in close agreement with Ghia's results [18] even though only a coarse grid was used for the simulation.

5.2. *Flow past a circular cylinder*

Unsteady flow past a circular cylinder is computed as a test case for this time-accurate projection method. According to Panton [19], the flow past a circular cylinder has attached vortices when $4 < Re < 40$ (where Re, the Reynolds number, is based on the diameter of the cylinder). The von Kármán vortex street would arise when $40 < Re < 60 - 100$ and alternate vortex shedding in the wake of the cylinder occur when the Reynolds number is increased to about 200. Different modes of vortex shedding would appear at different Reynolds numbers, such as oblique and parallel shedding mode demonstrated in experiments as shown in Reference [20]. The cylinder is located at the origin and has a unit diameter. The inflow boundary is located 8 diameters upstream of the cylinder and the outflow boundary is extended to 16 diameters aft of the cylinder. An overlapping grid shown in Figure 7, which contains two component grids, a cylindrical grid with 50×20 grid points and a background rectangular grid with 110×56 grid points, is generated to discretize the domain over which the computation is done. Computational results of the flow field corresponding to a Reynolds number of 200 for which experimental and numerical results are available [21, 22] are obtained using the numerical scheme. The initial condition was set by assuming the velocity everywhere to correspond to the uniform velocity, which was also applied at the inflow, top and bottom boundaries. The

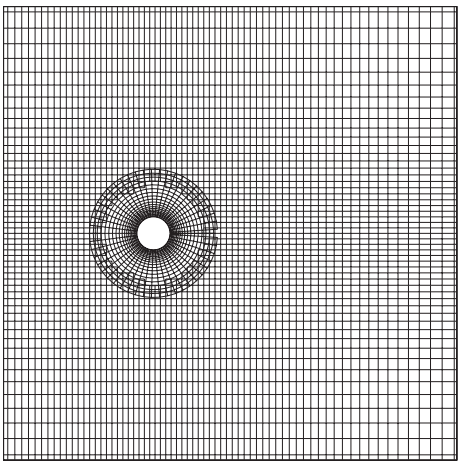


Figure 7. Overlapping grid for flow past a circular cylinder.

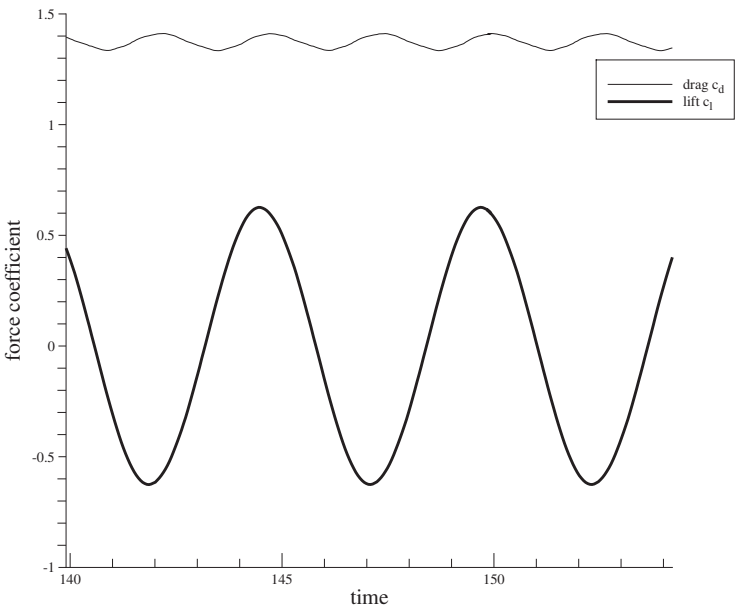


Figure 8. Time history plot of drag and lift coefficients of flow past a circular cylinder at $Re = 200$.

outflow boundary condition is specified as $\partial \mathbf{U}/\partial n = 0$. Various aspects of the application of this boundary condition at the outflow boundary are outlined in References [23–25].

The computation shows periodic vortex shedding in the wake of cylinder. The drag and lift forces acting on the cylinder varied according to the formation and separation of the

Table II. Drag and lift coefficients and Strouhal numbers for the flow past circular cylinder at $Re = 200$.

	C_D	C_L	S_t
Present	1.37 ± 0.04	± 0.63	0.192
Rogers fifth-order method [22]	1.23 ± 0.05	± 0.65	0.185
Rosenfeld *et al.* [31]	1.46 ± 0.05	± 0.69	0.211
Wille [21]	—	—	0.19

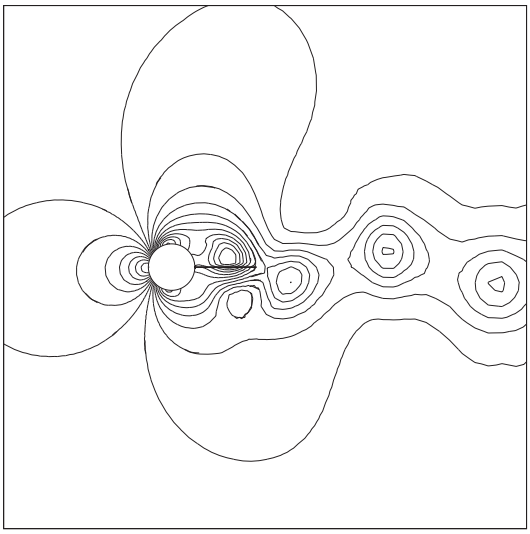
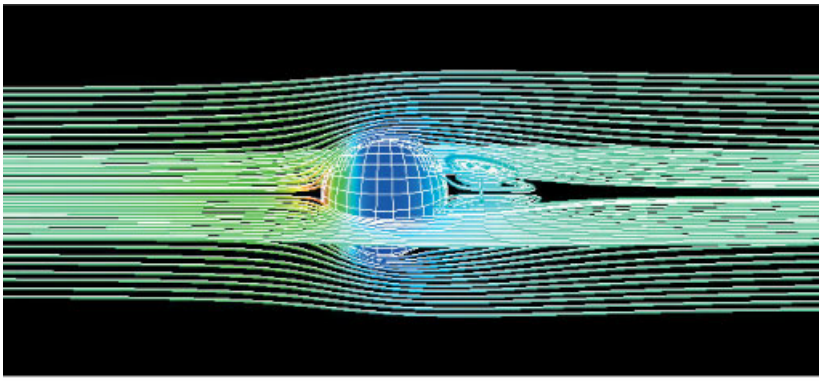


Figure 9. Pressure contour at an instant of flow past a circular cylinder at $Re = 200$.

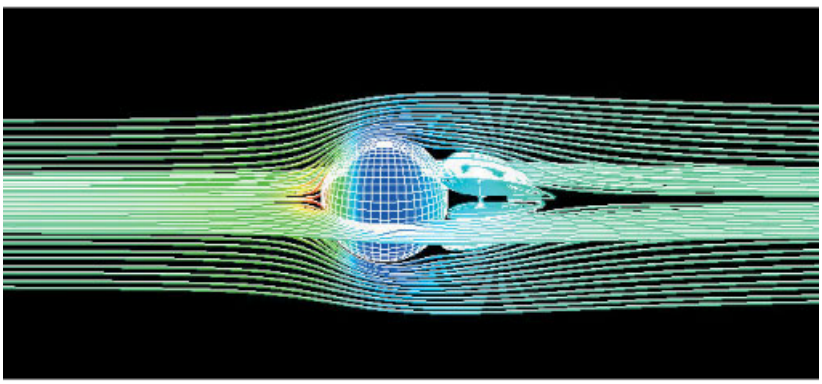
vortices. The computed variation of lift and drag coefficients vs time is shown in Figure 8. The computed Strouhal number for this flow is estimated to be 0.192. Computed values are compared with reported values in literature and is summarized in Table II. Figure 9 shows computed pressure contours at an instant of time. The variation of the computed streamlines for a cycle of vortex shedding are shown in Figure 10.

5.3. *Flow past a sphere*

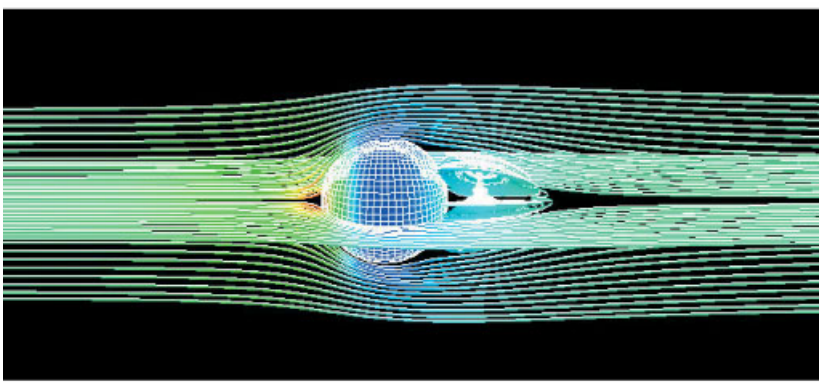
The flow past a sphere is a three-dimensional flow and is more computationally intensive than any of the previous two-dimensional steady and unsteady computation. This is simulated here to verify the performance of the parallelized three-dimensional Godunov-projection algorithm which can be used for more complex flow problems. As the wake of a sphere exhibits many interesting and rich flow structures as outlined in Reference [26], the simulation of this problem will assess the capabilities of this algorithm in capturing these rich flow structures and the associated hydrodynamic forces. The flow is axisymmetric and attached if the Reynolds number $Re < 20$. As Reynolds number increases, flow separation takes place and a cylindrical



(a)



(b)



(c)

Plate 1. Streamlines of flow past sphere at $Re = 100$ on different grids: (a) coarse grid; (b) regular grid; and (c) fine grid.

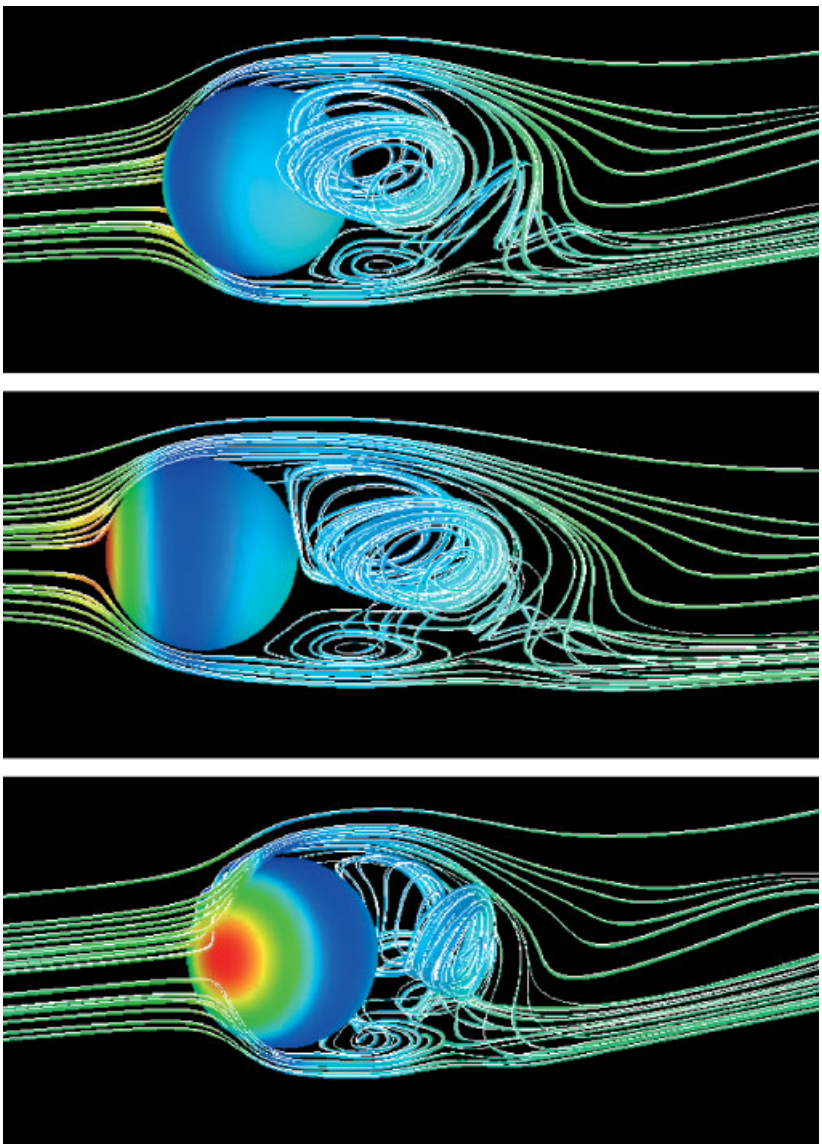


Plate 2. Streamlines at an instant of flow past sphere at $Re = 400$ at different viewpoints.

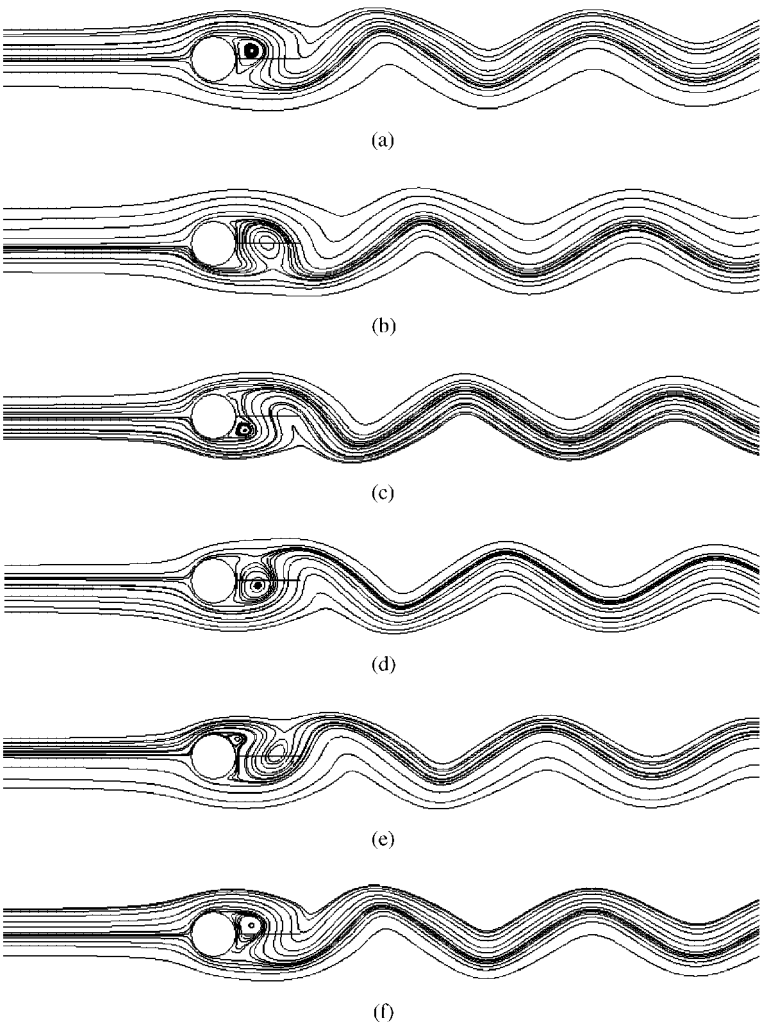


Figure 10. Streamlines for flow past a circular cylinder at $Re = 200$ at various instants during a vortex shedding cycle: (a) $t = 88.9261$; (b) $t = 89.9452$; (c) $t = 90.9643$; (d) $t = 91.9833$; (e) $t = 93.0024$; and (f) $t = 94.1186$.

vortex surface would appear in the flow field. When the Reynolds number is increased further, this axisymmetric flow would lose its stability in the range $120 < Re < 300$ [27] and the vortices will detach periodically. A description and visualization of the vortex formation associated with flow past spherical bodies can be found in Reference [28], where the Strouhal number of the vortex shedding was claimed to be in the range of 0.120–0.160. Here, the flow problem at Reynolds numbers $Re = 100$ and 400 are simulated using the Godunov-projection algorithm with the CFL number set to 0.9.

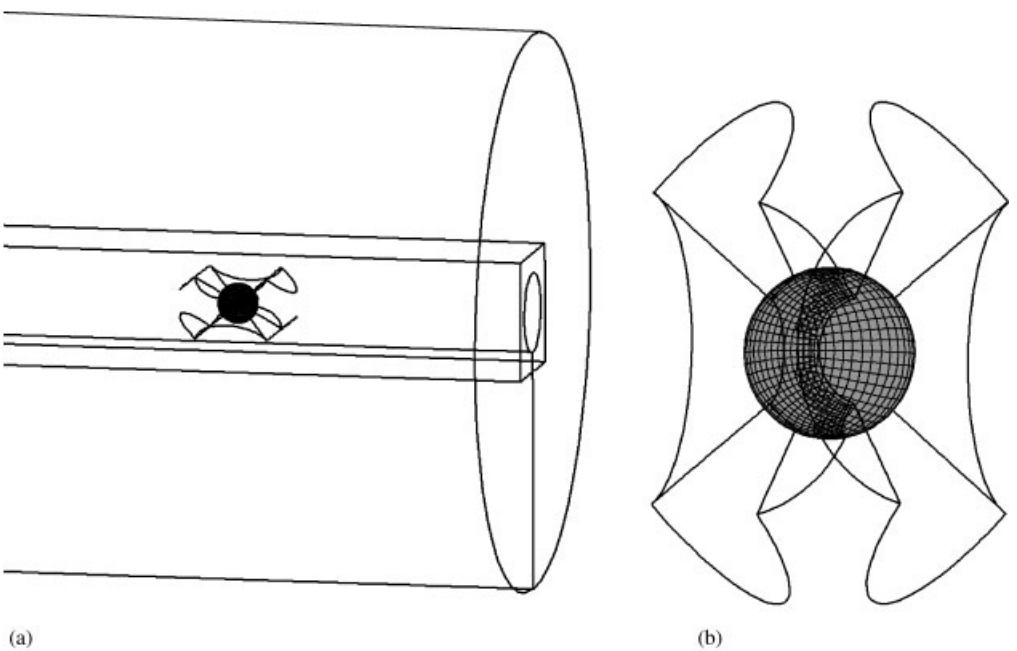


Figure 11. Overlapping grids around a sphere: (a) overlapping sub-domains around a sphere; and (b) sphere sub-domains.

The diameter of the sphere is assumed to be unity. The computational domain extends 7.5 units upstream and 15 units downstream and corresponds to the computational domain used by Kalro and Tezduyar [26]. An overlapping grid shown in Figure 11 is generated to discretize the computational domain by using Ogen [15]. This overlapping grid system consists of four component grids, namely a cylindrical domain, a rectangular domain and two patched domains grids. The grids in the cylindrical domain serves as the background grid covering the range and extent of the outer boundary and the grids in the rectangular domain is used to fill up the hole left by the cylindrical domain. The two patched spherical sub-domains embedded in the union of box and cylindrical grids are used to construct the near-field grid around the sphere. Figure 11(b) shows the boundaries of these two sub-domains, referred here as north and south patched sub-domains at the interfacing location to form the overlapped grid structure around the sphere. Figure 12 shows the exploded views of these two patched domains around the sphere and from this figure the extent of the boundaries of the computational area around the sphere can be seen. Some redundant grid points of north patched sub-grid have been removed so that the interpolation boundary from the south patched grid no longer forms the interior boundaries. The singularities normally associated with grid generation around three-dimensional bodies such as spheres, etc. have been removed by using these two patched domain grids.

The flow past the sphere should be axisymmetric at Reynolds number 100, and the numerical simulation using the current algorithm confirms this observation. The calculations were executed for a sufficiently long duration at the end of which a steady axisymmetric recirculation flow structure forms and persists in the wake of sphere. The effect of the grid density

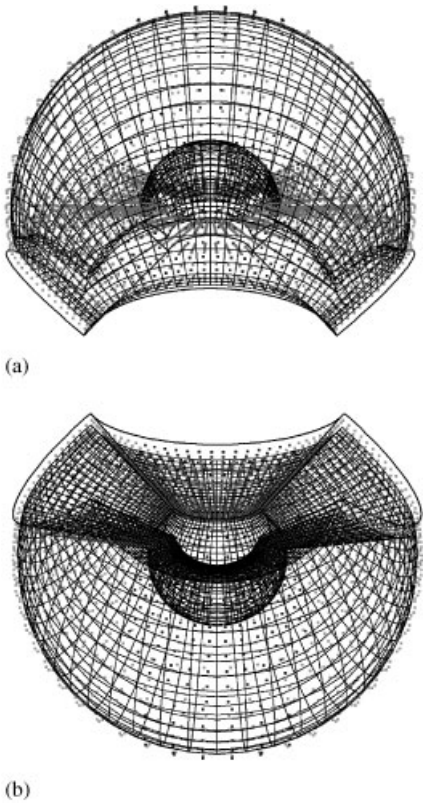


Figure 12. Exploded views of the grids in the patched subdomains and their interface of overlap: (a) interpolation cells on the north placed grid; and (b) interpolation cells on the south patched grid.

on the convergence of the current numerical algorithm is considered by simulating the flow on three sets of overlapping grids which has different number of grid cells, namely 78 400 grid cells for the coarse grid, 425 088 grid cells for the intermediate grid and 1 132 384 grid cells for the fine grid. The extent of the computational domain is kept the same for all the grids and the same overlapping structure is used for all these grids. The computed streamlines of this flow on these three grid systems are shown in Plate 1, where the axisymmetric ring configuration of the recirculation zone in the wake of the sphere can be clearly seen for all the computed flow fields. This is in close agreement with the structures which have been reported by Kim and Pearlstein [27]. In these figures, the colour used on the surface of the sphere and the streamlines is determined by scaling with the pressure field. The surface grids on the sphere are also shown in these streamline plots, where the size of spatial step length on the three grids can be clearly seen for the three grid systems. The flow field computed by using the coarse grid is shown in Plate 1(a), and Plate 1(b) displays the flow computed on the intermediate grid. The computed solution on the fine grid is shown in Plate 1(c). It can be seen that similar flow fields are produced by using these three systems of overlapping grids although the density of grid points are quite different. The effects of grid density on the

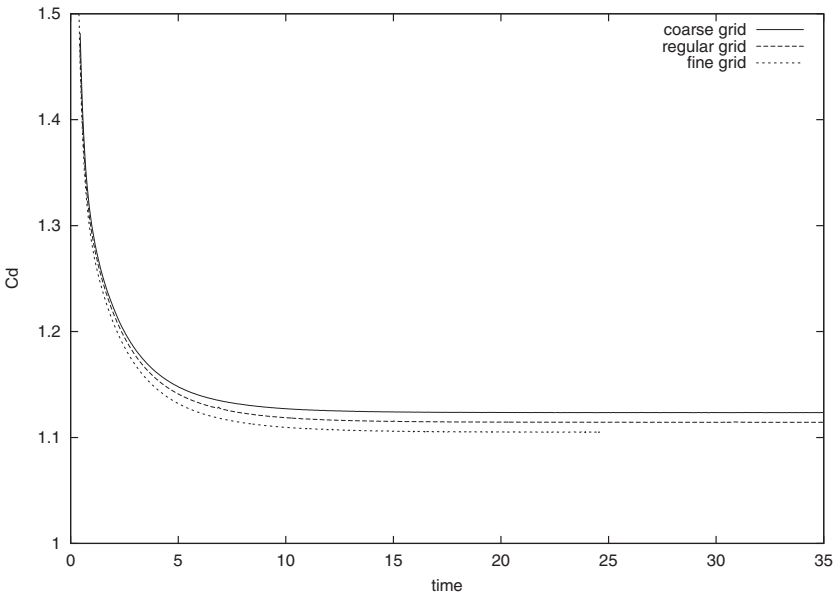


Figure 13. Time variation of drag coefficient C_D of flow past sphere at $Re = 100$ on three different grids.

convergence history of the computations on these three grids are shown in Figure 13, where the variation of drag coefficient, C_d, with time or number of iterations to steady-state flow fields computed on different grids are shown. The drag coefficients computed on these three systems of overlapping grids with different grid densities converge after a short initial stage when the transients have been flushed out of the computational domain. The effect of the grid size on the converged values of the steady-state drag coefficients are shown in Figure 14, where the x co-ordinate is the inverse of the total number of grid cells N. The computed drag coefficients are very close to the experimental value presented by Schlichting [29], 1.1. As the computational grid becomes finer and finer, the drag coefficient reaches a closer agreement with the experimental data.

The flow past sphere at $Re = 100$ is steady. Unsteady flow should appear when the Reynolds number is increased further and the vortex shedding process is initiated. Compared with this case, much more complex flow structures can be expected in flow at $Re = 400$. In the flow past the sphere at $Re = 400$, it is observed that the vortices are shed periodically. The computed variation of the drag coefficient during a short duration are shown in Figure 15. The time-averaged drag coefficient is estimated to be about 0.648, which compares well with the value 0.619 presented by Kalro and Tezduyar [26]. The computed variation of the coefficients of side forces, $(c_l)_y$ and $(c_l)_z$ are shown in Figure 15. It seems that the side forces do not oscillate periodically. However, Kalro and Tezduyar observed that the vortex surface rotated in low frequency when he carried out a long-duration computation. This fact does explain the variation of the side forces during this short duration. The computed Strouhal number based on the vortex shedding frequency is about 0.14 compared to 0.131 obtained by

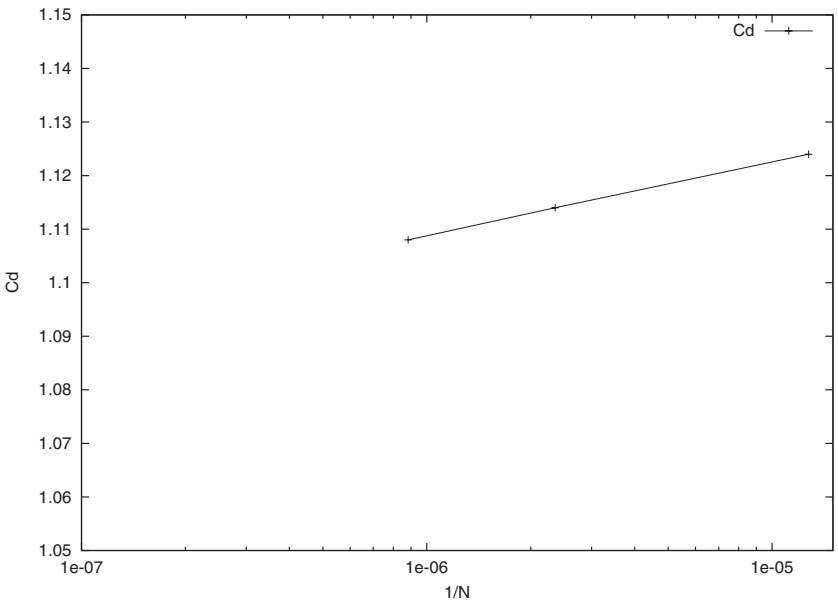


Figure 14. Drag coefficient of flow past sphere at $Re = 100$ on three different grids.

Kalro and Tezduyar. Figure 16 shows the velocity vector and pressure contour plots in the various streamwise planes, which are placed at angular intervals of 45°. These plots show that the vortex shedding is not symmetric. The back flow region extends to about 1.5 diameters downstream and this is in agreement with that computed by Karlo and Tezduyar [26]. The computed streamlines of this flow are shown in Plate 2, where the colour shows the pressure.

In order to evaluate the performance of the parallel code, the speedup and efficiency has been measured for the flow past sphere at $Re = 100$. Figure 17(a) shows the speedup vs number of processors used in the computation and the efficiency is shown in Figure 17(b). It is same as in this case that the speedup increases as the number of processors is increased. However, the variation of efficiency is not straightforward. The speedup has a large increment when 32 slave processors are used and its efficiency also increases. This phenomena is unusual and it may be due to the architecture of the SGI Origin. The SGI Origin [30] applies cache coherent non-uniform memory access (ccNUMA) architecture where a local memory (cache) is attached to each processor. Processors also have access to the huge shared remote memory through the dedicated scalable interconnect network. This arrangement is named as distributed shared memory(DSM) and the cache coherence is maintained in hardware. The number of grid cells for this computation is large. When only a limited number of processors are employed, the overlapping grid can only be divided into the same number of grid partitions as the slave processors used and each partition has a relatively large number of grid cells. Therefore, the data cannot be held in the cache entirely and the frequent swap between the cache and the main memory can be expected. When more processors are used, the partitioned grid parts become smaller in view of the increased number of grid partitions. When the grid is partitioned into 32 parts, the cache is large enough for the local grid

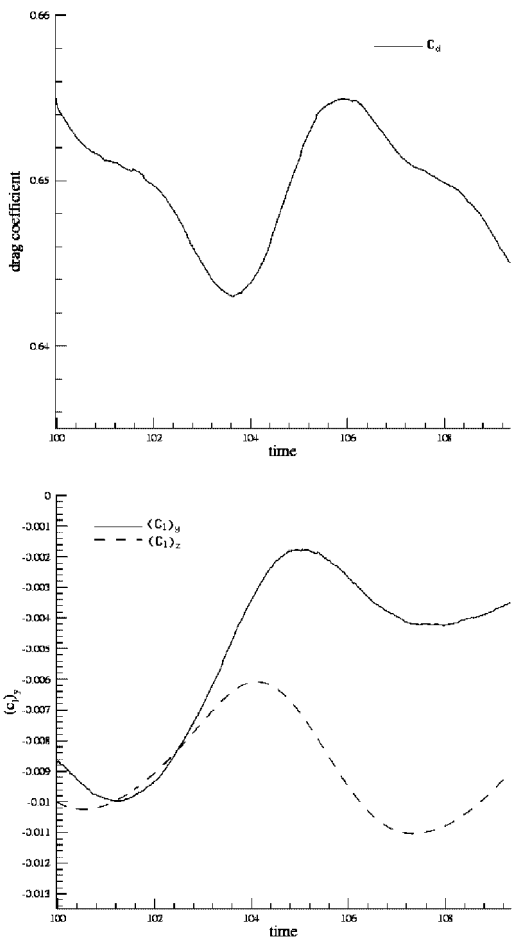


Figure 15. Time variation of drag and side forces acting on the sphere at $Re = 400$.

partitions. Therefore, the swap is no longer performed commonly in which case the efficiency is improved greatly and the speedup also shows an increment. For the case when 42 slave processors are used for the computation, the efficiency dropped even though the speedup is increased. After checking the grid partitions held by each processor, it was found that the number of grid cells of grid partitions were different. Therefore, the processors have different amount of computation. This fact confirmed the conclusion that appropriate grid partitioning can achieve better computational load balancing among all processors.

6. CONCLUSION

In this work, the Godunov-projection method for solving the unsteady incompressible Navier–Stokes equations has been implemented on the overlapping grids. This scheme is

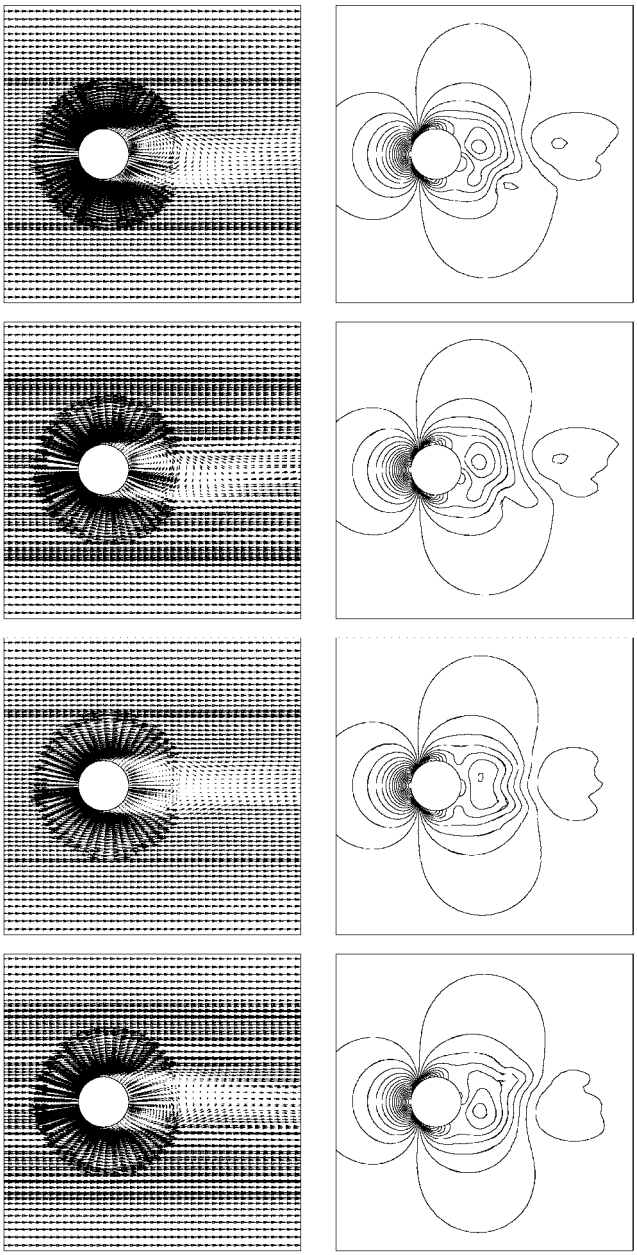


Figure 16. Vector and pressure plots at the various streamwise planes of flow past sphere at $Re = 400$.

second-order accurate temporally and spatially. The Godunov procedure is incorporated to provide a robust discretization of the convective terms for high Reynolds number flow. The pressure Poisson equation is solved to obtain the pressure field. The usage of overlapping grids

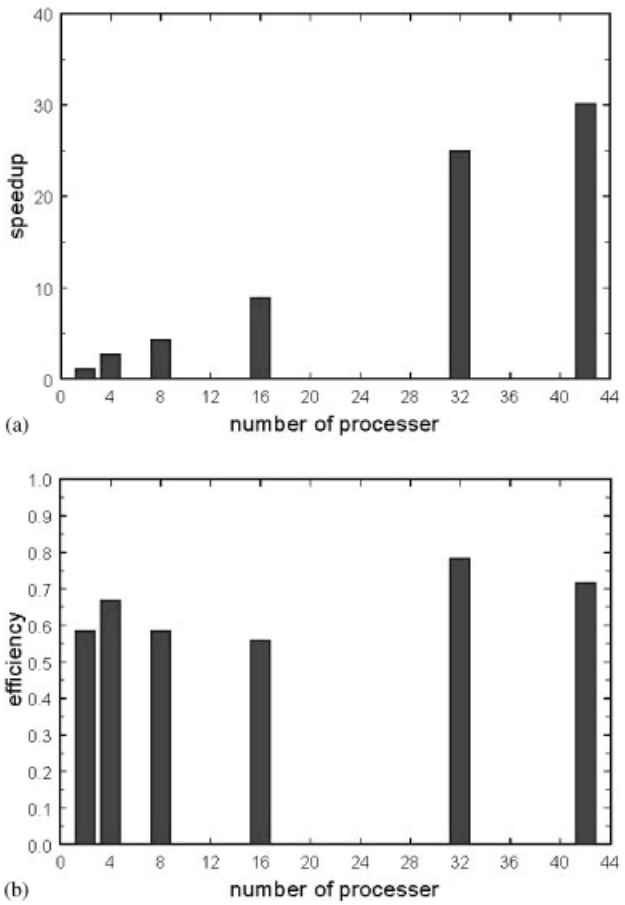


Figure 17. Speedup (a) and efficiency (b) of the parallel computation of flow past a sphere.

can greatly reduce the complexity of structured grid generation. The third-order interpolation of the flow variables is used to couple the flow field across the interface regions of the component grids. Furthermore, the parallelization of the combination of the projection method and the overlapping grid by partitioning the grids in a scalable and efficient manner offers considerable economy for three-dimensional flow simulations. The benchmark flow simulation problems that have been solved by using this algorithm and discussed in the previous section demonstrate the capability and the performance of the method. It can be concluded that this method can achieve reasonable accuracy, speedup and efficiency when used as a flow prediction method.

REFERENCES

1. Yuh J. Modeling and control of underwater robotic vehicles. *IEEE Transactions on Systems, Man and Cybernetics* 1990; **20**:1475–1483.
2. Kalske S. Motion simulation of underwater vehicles. *VTT-PUBS-97*, Ship Laboratory, Technical Research Centre of Finland, ESPOO, Finland, 1992.
3. Fossen TI. *Guidance and Control of Ocean Vehicles*. Wiley: Chichester, NY, U.S.A., 1994.

4. Bell JB, Colella P, Glaz HM. A second-order projection method for viscous incompressible flow. In *AIAA Paper 87-1776-CP*, *Proceedings of AIAA 8th Computational Fluid Dynamics Conference*, Honolulu, Hawaii, 1987; 789–794.
5. Bell JB, Colella P, Glaz HM. A second-order projection method for the incompressible Navier–Stokes equations. *Journal of Computational Physics* 1989; **85**(2):257–283.
6. Bell JB, Solomon JM, Szymczak WG. A projection method for viscous incompressible flow on quadrilateral grids. *AIAA Journal* 1994; **32**(10):1961–1969.
7. Almgren AS, Bell JB, Szymczak WG. A numerical method for the incompressible Navier–Stokes equations based on an approximate projection. *SIAM Journal of Scientific Computing* 1996; **17**(2):358–369.
8. Chesshire G, Henshaw WD. Composite overlapping meshes for the solution of partial differential equations. *Journal of Computational Physics* 1990; **90**:1–64.
9. Colella P. A direct Eulerian MUSCL scheme for gas dynamics. *SIAM Journal on Scientific and Statistical Computing* 1985; **6**:104–117.
10. Rider WJ. The robust formulation of approximate projection methods for incompressible flows. *Technical Report LA-UR*-94-3015, Los Alamos National Laboratory, 1994.
11. Roe PL. Approximate Riemann solvers, parameter vectors and difference schemes. *Journal of Computational Physics* 1981; **43**:357–372.
12. Lai MF, Bell JB, Colella P. A projection method for combustion in the zero Mach number limit. In *Proceedings of AIAA 11th Computational Fluid Dynamics Conference*, Orlando, FL, U.S.A., 1993; 776–783.
13. Chorin AJ. Numerical solution of the Navier–Stokes equations. *Mathematics of Computation* 1968; **22**: 745–762.
14. Minion ML. A projection method for locally refined grids. *Journal of Computational Physics* 1996; **127**: 158–177.
15. Henshaw WD. Ogen: an overlapping grid generator for overture. *Technical Report LA-UR*-96-3466, Lawrence Livermore National Laboratory, U.S., 1996.
16. Gropp W, Lusk W, Skjellum A. *Using MPI*: *Portable Parallel Programming with the Message Passing Interface* (2nd ed.), MIT Press: Cambridge, MA, 1999, 1994.
17. Pan H. Computational hydrodynamics of moving underwater rigid bodies using Godunov-projection method on overlapping grids. *Ph.D. Thesis*, Nanyang Technological University, Singapore, August 2001.
18. Ghia U, Ghia KN, Shin CT. High-Re solution for incompressible flow using the Navier–Stokes equations and a multigrid method. *Journal of Computational Physics* 1982; **48**:388–411.
19. Panton RL. *Incompressible Flow*. Wiley-Interscience: New York, 1996.
20. Williamson CHK. Oblique and parallel modes of vortex shedding in the wake of a circular cylinder at low Reynolds numbers. *Journal of Fluid Mechanics* 1989; **206**:579–627.
21. Wille R. Karman vortex street. *Advances in Applied Mechanics* 1960; **6**:273–287.
22. Rogers SE, Kwak D. Upwinding differencing scheme for the time-accurate incompressible Navier–Stokes equations. *AIAA Journal* 1990; **28**(2):253–262.
23. Peyret R, Taylor TD. *Computational Methods For Fluid Flow*. Springer: New York, 1983.
24. Gresho PM. On the theory of semi-implicit projection methods for viscous incompressible flow and its implementation via a finite element method that also introduces a nearly consistent mass matrix. Part 1: theory. *International Journal for Numerical Methods in Fluids* 1990; **11**:587–620.
25. Gresho PM. Some interesting issues in incompressible fluid dynamics, both in the continuum and in numerical simulation. *Advances in Applied Mechanics* 1992; **28**:45–140.
26. Kalro V, Tezduyar T. 3D computation of unsteady flow past a sphere with a parallel finite element method. *AHPCRC Preprint 97-011*, Army High Performance Computing Research Center, University of MN, U.S.A., 1997.
27. Kim I, Pearlstein J. Stability of the flow past a sphere. *Journal of Fluid Mechanics* 1990; **211**:73–93.
28. Magarvey RH, Maclatchy CS. Wakes in liquid–liquid system. *Physics of Fluids* 1961; **4**:800–805.
29. Schlichting H. *Boundary-Layer Theory*. McGraw-Hill: New York, 1979.
30. Laudon J, Lenoski D. The SGI Origin: a ccNUMA highly scalable server. http://www.sgi.com, 2000.
31. Rosenfeld M, Kwak D, Vinokur M. A fractional-step solution method for the unsteady and incompressible Navier–Stokes equations in generalized co-ordinate systems. *Journal of Computational Physics* 1991; **94**(1):102–137.

