Available online at www.sciencedirect.com

SCIENCE DIRECT®

Journal of Computational Physics 191 (2003) 660–669

JOURNAL OF COMPUTATIONAL PHYSICS

www.elsevier.com/locate/jcp

Short Note

A general reconstruction algorithm for simulating flows with complex 3D immersed boundaries on Cartesian grids

A. Gilmanov [a], F. Sotiropoulos [a,*], E. Balaras [b]

[a] School of Civil and Environmental Engineering, Georgia Institute of Technology, Atlanta, GA 30332-0355, USA
[b] Department of Mechanical Engineering, University of Maryland, College Park, MD 20742, USA

Received 24 December 2002; received in revised form 11 June 2003; accepted 11 June 2003

Abstract

In the present note a general reconstruction algorithm for simulating incompressible flows with complex immersed boundaries on Cartesian grids is presented. In the proposed method an arbitrary three-dimensional solid surface immersed in the fluid is discretized using an unstructured, triangular mesh, and all the Cartesian grid nodes near the interface are identified. Then, the solution at these nodes is reconstructed via linear interpolation along the local normal to the body, in a way that the desired boundary conditions for both pressure and velocity fields are enforced. The overall accuracy of the resulting solver is second-order, as it is demonstrated in two test cases involving laminar flow past a sphere.

Keywords: Cartesian grids; Immersed boundaries; Direct forcing; Finite-difference method

1. Introduction

Numerical methods for solving the Navier–Stokes equations on non-boundary-conforming grids are becoming increasingly popular due to their versatility in simulations of flows with complex immersed boundaries [1–6]. Depending on the way boundary conditions are enforced on solid surfaces, commonly adopted methodologies are the immersed boundary methods [1], Cartesian or cut-cell methods [2,3] and hybrid Cartesian/immersed boundary formulations [4–6]. The latter class of methods is particularly attractive because boundary conditions can be accurately applied on a sharp solid interface – thus, alleviating an important limitation of the immersed boundary approach – without requiring tedious cell-merging procedures typically employed in cut-cell formulations [2,3]. In particular, the application of boundary conditions in hybrid formulations involves the reconstruction of the solution at the nodes nearest to the immersed interface, via appropriate interpolation procedures [4–6] using the known boundary values on the solid surface and information from the interior of the flow. This solution reconstruction scheme is a critical

[*] Corresponding author. Tel.: +1-404-894-4432; fax: +1-404-385-1131.
E-mail address: fs30@ce.gatech.edu (F. Sotiropoulos).

0021-9991/$ - see front matter © 2003 Elsevier B.V. All rights reserved.
doi:10.1016/S0021-9991(03)00321-8

issue for the successful implementation of hybrid approaches in three-dimensional flows as it determines both the global spatial accuracy of such methods and their adaptability to flow computations with arbitrarily complex immersed boundaries.

Fadlun et al. [4] proposed a simple reconstruction algorithm based on one-dimensional interpolations along the grid line that passes from a given near-boundary node and intersects the solid body. The method is straightforward, second-order accurate, and works well for bodies that are largely aligned with the grid lines. For geometrically complex immersed boundaries, however, the choice of the reconstruction direction may not be unique as often more than one grid line passing through a near-boundary node may intersect the boundary. A multi-dimensional scheme aimed at removing this arbitrary element in the implementation of hybrid formulations was proposed in [5]. This scheme uses a bilinear reconstruction procedure that is reduced to the one-dimensional linear one when there are no available points in the vicinity of the boundary to support the two-dimensional stencil. More recently, Balaras [6] introduced a reconstruction scheme, which performs the reconstruction along the well defined line normal to the body. The accuracy of this method was demonstrated by applying it to simulate laminar flow past a circular cylinder and to carry out a large-eddy simulation of flow through an infinitely wide channel with a wavy wall [6]. The algorithm proposed in [6] eliminates the previously discussed ambiguities associated with interpolation along grid lines but its applicability is restricted to flows with immersed boundaries that are aligned with one coordinate direction (e.g., two-dimensional or axisymmetric shapes). In such cases, the solution reconstruction is greatly simplified as it needs to be performed in two-dimensional planes.

In this note we develop a new reconstruction algorithm, which is based on the ideas of [6] but is applicable to arbitrarily complex, three-dimensional immersed boundaries. The proposed methodology maintains a sharp fluid/body interface by discretizing the body surface using an unstructured, triangular mesh (see [7] for the origin of this idea in multi-phase flow simulations) and reconstructing the solution at near-boundary nodes via linear interpolation along the local normal to the body. In what follows, we describe briefly the base flow solver, present the new reconstruction scheme and demonstrate its second-order accuracy by applying it to simulate two test cases involving laminar flow past a sphere.

2. Base numerical method

We solve the three-dimensional, incompressible Navier–Stokes equations in strong conservation form on non-staggered grids using a second-order accurate, finite-difference approach. The governing equations are discretized using three-point central differencing for the pressure gradient and viscous terms in the momentum equations and the velocity derivate terms in the continuity equations along with the second-order accurate QUICK upwind scheme for the convective terms. Odd–even decoupling of the pressure nodes is eliminated using the method proposed in [8]. The discrete equations are integrated in a time-accurate manner using the second-order accurate, dual time-stepping, artificial compressibility approach described in [9]. This method employs a four-stage, explicit Runge–Kutta scheme enhanced with implicit residual smoothing and local (dual) time stepping [9].

3. Treatment of complex immersed boundaries

We discretize immersed boundaries using an unstructured triangular mesh (see Figs. 1 and 2(a)). This choice is dictated by: (1) the need to develop a method for handling arbitrarily complex immersed boundaries; (2) the fact that the reconstruction algorithm we propose relies on interpolation along the normal direction, which can be readily computed for a surface discretized with triangular elements. Given the triangular discretization of the immersed boundary, the computational nodes of the Cartesian mesh can

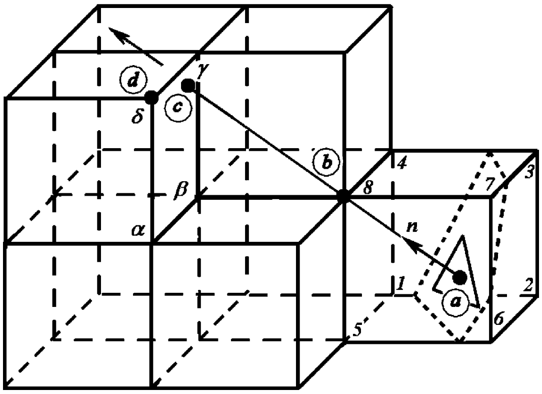

Fig. 1. Schematic depicting the reconstruction of the solution at an immersed boundary node b by interpolating along the local normal to the surface of the body. The dashed line plane is the intersection of the body surface with the Cartesian grid. The triangle is a typical element of the unstructured mesh used to discretize the surface of the immersed boundary.

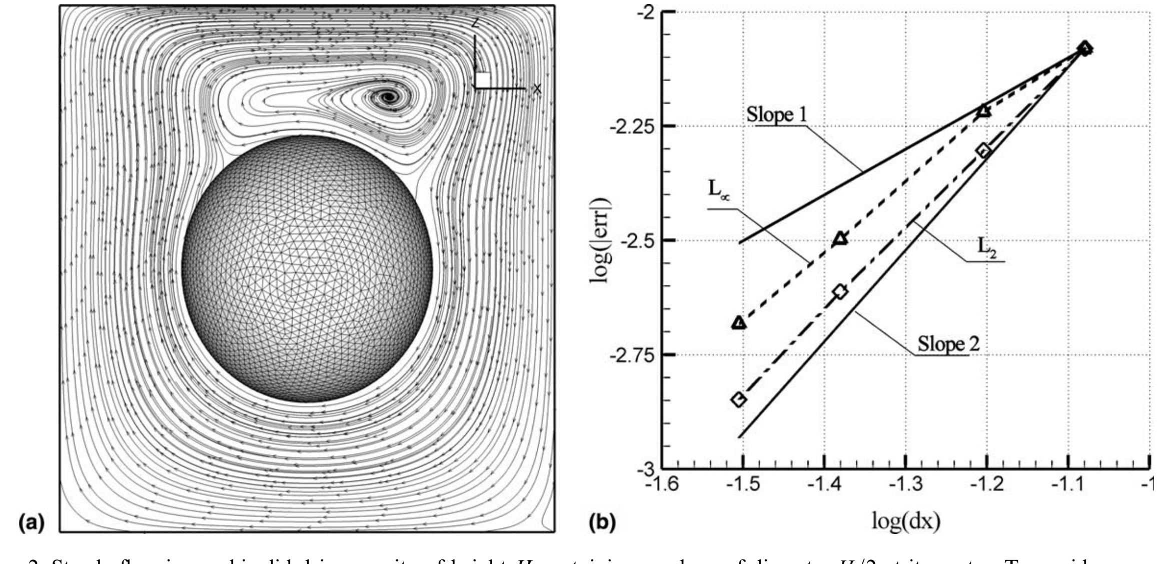

Fig. 2. Steady flow in a cubic, lid-driven cavity of height H containing a sphere of diameter $H/2$ at its center. To avoid corner singularities the lid velocity is prescribed as follows: $U_{\mathrm{Lid}}(x) = 0.5[1 + \sin(4\pi x/H - \pi/2)]$ for $x/H = 0.25$ and $x/H \geqslant 0.75$; $U_{\mathrm{Lid}}(x) = 1.0$ for $0.25 < x/H < 0.75$. Calculations were carried out on four grids: 25^3, 33^3, 49^3, and 65^3. (a) Unstructured grid on the surface of the sphere and calculated steady-state streamlines at the vertical plane of symmetry ($y = H/2$); (b) convergence of the L_∞ and L_2 norms for the error in the velocity magnitude relative to a benchmark numerical solution obtained on a 129^3 grid. The average slopes for the L_∞ and L_2 norms are 1.48 and 1.74, respectively.

be classified in three categories based on their location relative to the body: (1) nodes in the fluid phase (fluid nodes); (2) nodes in the solid body (solid nodes); and (3) near-boundary nodes. The latter are grid nodes in the fluid that are closest to the solid surface and in our algorithm are treated as boundary points. We shall refer to these nodes as the *immersed boundary* (IB) nodes. For a convex body (i.e., a body that contains *all* the line segments connecting any pair of its points) the following algorithm can be used to classify the Cartesian grid nodes in the three categories mentioned above. The Cartesian grid node (i, j, k)

will be internal to the body (solid node) if the following condition is satisfied (no summation over repeated indices):

$$\boldsymbol{n}_m \cdot (\boldsymbol{r}_{i,j,k} - \boldsymbol{r}_m) < 0 \qquad \forall m = 1, M, \tag{1}$$

where M is the number of triangular elements used to discretize the body, $\boldsymbol{r}_{i,j,k}$ is the position vector of the (i, j, k) Cartesian grid node, $\boldsymbol{r}_m$ is the position vector of the centroid of the mth triangular element ($m = 1$–M) on the interface, and $\boldsymbol{n}_m$ is the normal unit vector at the centroid of the mth triangular element. Nodes that do not satisfy Eq. (1) for all m are in the fluid phase and will be either fluid or IB nodes. To determine the IB nodes, we examine the six grid nodes $(i \pm 1, j, k)$, $(i, j \pm 1, k)$, $(i, j, k \pm 1)$ surrounding each solid node (i, j, k). Each such node that is not a solid node is classified as an IB node. At the end of this procedure all the remaining nodes in the fluid phase are the fluid nodes. For example, with reference to Fig. 1 nodes 2, 3, and 6 are tagged as solid, nodes α, β, γ and δ as fluid, and nodes 1, 4, 5, 7, and 8, as IB. It is important to emphasize that the criterion given by Eq. (1) is strictly valid only for a Cartesian domain containing a single, convex, immersed boundary. For domains containing multiple immersed boundaries and/or boundaries of more general shape, further constraints need to be imposed on Eq. (1), which, however, is left as a topic for future research. Future extension of the method should also address situations involving bodies with regions of very high curvature where the projection from an *IB* node to the body may not be always uniquely defined or even may not exist.

The governing equations are solved at all fluid nodes with boundary conditions specified at the IB nodes. All solid nodes are blanked out of the computation. Assuming that the solution at all fluid nodes at the ℓ-iteration of the iterative solution procedure for solving the governing equations is known, advancing the velocity and pressure fields to the $\ell + 1$ iteration requires specification of boundary conditions at all IB nodes. Since the immersed boundary is tracked as a sharp interface, boundary conditions are known at the M nodes of the unstructured surface mesh. These conditions typically consist of Dirichlet conditions for the velocity field (no-slip conditions, i.e., $u = v = w = 0$ on the surface for a fixed rigid body) and a Neumann condition for the pressure field (derived from the normal momentum equation to the body), which can also be viewed as a Dirichlet condition for $\partial p/\partial n$.

Let f_m denote any one of these four quantities on the interface – in other words, f stands for u, v, w, or $\partial p/\partial n$ and is known at all times for all $m = 1, M$. With reference to Fig. 1, the boundary conditions for $f_b^{\ell+1}$ at the IB node b are determined as follows. According to the notation of this figure, IB node b is associated with the triangular element with normal $\boldsymbol{n}$ on the interface mesh. Since $\boldsymbol{n}$ is known, the line that passes through node b and is parallel to $\boldsymbol{n}$ can be readily constructed. This line intersects the surface element at point a (the projection of node b on the immersed boundary) and the Cartesian grid element defined by nodes α–β–γ–δ at point c. With f assumed to be known at all vertices of the surface element, f_a can be computed by linear interpolation among the vertices of the triangular surface element as follows:

$$f_a = \left(\sum_{m=1,3} f_m/s_m \right) \Bigg/ \left(\sum_{m=1,3} 1/s_m \right), \tag{2}$$

where $m = 1, 3$ are the three vertices of the triangular element within which point a lies, and s_m is the distance between a and m-vertex. For a fixed, rigid body this step is not required for the velocity components since all three of them are set equal to zero everywhere on the surface ($f_a = 0$) to satisfy the no-slip condition. A similar interpolation procedure is employed to obtain $f_c^{\ell+1}$, by applying Eq. (2) to interpolate among the internal Cartesian grid nodes α–β–γ–δ where the solution is already known at the previous iteration. This interpolation procedure is straightforward for the three Cartesian velocity components. To calculate the normal pressure gradient $(\partial p/\partial n)_c^{\ell+1}$, we first compute the pressure derivatives $(\partial p/\partial x_i)_c^{\ell}$ in all three Cartesian coordinate directions ($i = 1, 2, 3$) at c by interpolating between their known values at nodes α–β–γ–δ. The normal pressure gradient at c is then calculated as follows:

$$\left(\frac{\partial p}{\partial n}\right)_c^{\ell+1} = \boldsymbol{n}_a \cdot (\nabla p)_c^{\ell}. \tag{3}$$

With all quantities known at points a and c, boundary conditions for the velocity components and the pressure at IB node b are determined as follows. The velocity components $(u, v, w)_b^{\ell+1}$ are calculated using linear interpolation along a–c between the known values at points a and c. Linear interpolation between the known values at points a and c is also used to determine the normal pressure gradient $(\partial p/\partial n)_{bc}$ at the mid-point of the segment b–c and the value of the pressure at IB node b is determined via the following central, second-order accurate approximation:

$$p_b^{\ell+1} = p_c^{\ell} - \Delta s_{bc}\left(\frac{\partial p}{\partial n}\right)_{bc}^{\ell+1}, \tag{4}$$

where Δs_{bc} is the distance between points b and c.

In the above approach the IB nodes are treated as boundary nodes in computations with body-fitted meshes and, thus, discrete operators with stencil greater than three in the governing equations need to be modified. For example, at a node immediately adjacent to an IB node (such as node β in Fig. 1) the QUICK scheme cannot be applied if the sign of the local convective velocity is such that a blanked, external node is required in the unmodified stencil. In such cases, the QUICK scheme is replaced with the first order upwind scheme. Such an approximation does not affect the second order accuracy of the method (see Figs. 2 and 6) since the truncation error of the first-order upwind scheme involves second-order spatial derivatives, which vanish for a linearly varying velocity field. Moreover, the linear variation of the velocity field near the boundary also allows us to use homogeneous Neumann boundary condition for the pressure as for this case and assuming that the boundary is stationary the normal momentum equation reduces to $\partial p/\partial n = 0$. The algorithm presented above, however, is generally applicable to flows where a non-zero pressure gradient is imposed on the surface – say due to motion of the immersed boundary.

4. Results

We test the spatial accuracy of our algorithm by applying it to calculate steady flow in a cubic lid-driven cavity containing a stationary, rigid sphere at its center. This problem is the three-dimensional equivalent of the 2D problem used in the recent study by Kirpatrick et al. [3] to test the accuracy of their cut-cell formulation. The cavity height is H, the diameter of the sphere is $H/2$, and the lid velocity is made to vary smoothly in space to avoid the corner singularities (see caption of Fig. 2). We carry out simulations for $Re = 20$ (based on the sphere diameter and the lid maximum velocity) on five, uniform, Cartesian grids: 25^3, 33^3, 49^3, 65^3, and 129^3. For all grids, the sphere surface is discretized with 7770 triangular elements. The solution obtained on the finest grid 129^3 is considered as the benchmark ("exact") solution and the errors of the solutions obtained on the four coarser grids are quantified in terms of the L_∞ and L_2 norms. The error norms are calculated at all grid nodes exterior to the sphere, including the *IB* nodes where the reconstruction algorithm is employed. As seen in Fig. 2, the convergence of both norms indicates near second-order accuracy. The reasons for the small deviations from second-order convergence are not entirely clear but could possibly be attributed to the relative coarseness of the benchmark grid. Note that the benchmark grid used in the 2D convergence study in [3], who reported convergence somewhat closer to second-order than that shown in Fig. 2, was almost three times finer along each direction (they used a 384^2 grid). Such fine grid resolution is prohibitively expensive in our 3D study. The local convergence results shown in Fig. 2, however, along with the subsequently presented global convergence study for the sphere problem (Fig. 6) establish the second-order accuracy of our reconstruction approach.

To demonstrate the accuracy of our algorithm for a flow with more complex physics we apply it to simulate laminar flow past a sphere for $50 \leqslant Re \leqslant 300$. The Reynolds number is based on the sphere diameter D and the freestream velocity U – in our subsequent discussion we have used D and U to scale lengths and velocities, respectively. This flow has been previously computed by Johnson and Patel [10] (see also [11]), who used a body fitted grid, Fadlun et al. [4] and Kim et al. [5], who both employed variants of hybrid Cartesian/immersed boundary techniques. It is important to note that in [4,5] cylindrical, polar coordinates were used to discretize the flow domain and, thus, the solution reconstruction scheme in these studies was implemented only in two-dimensional azimuthal planes. In this work we treat the sphere as a three-dimensional surface immersed in the fluid and solve the governing equations in Cartesian coordinates in order to further test the accuracy of our three-dimensional reconstruction scheme.

The sphere is centered at the origin of the Cartesian coordinate system in a rectangular box extending between: $-10 \leqslant x \leqslant 10$, $-5 \leqslant y \leqslant 5$, and $-5 \leqslant z \leqslant 5$. Free stream boundary conditions ($u = 1, v = w = 0$) are

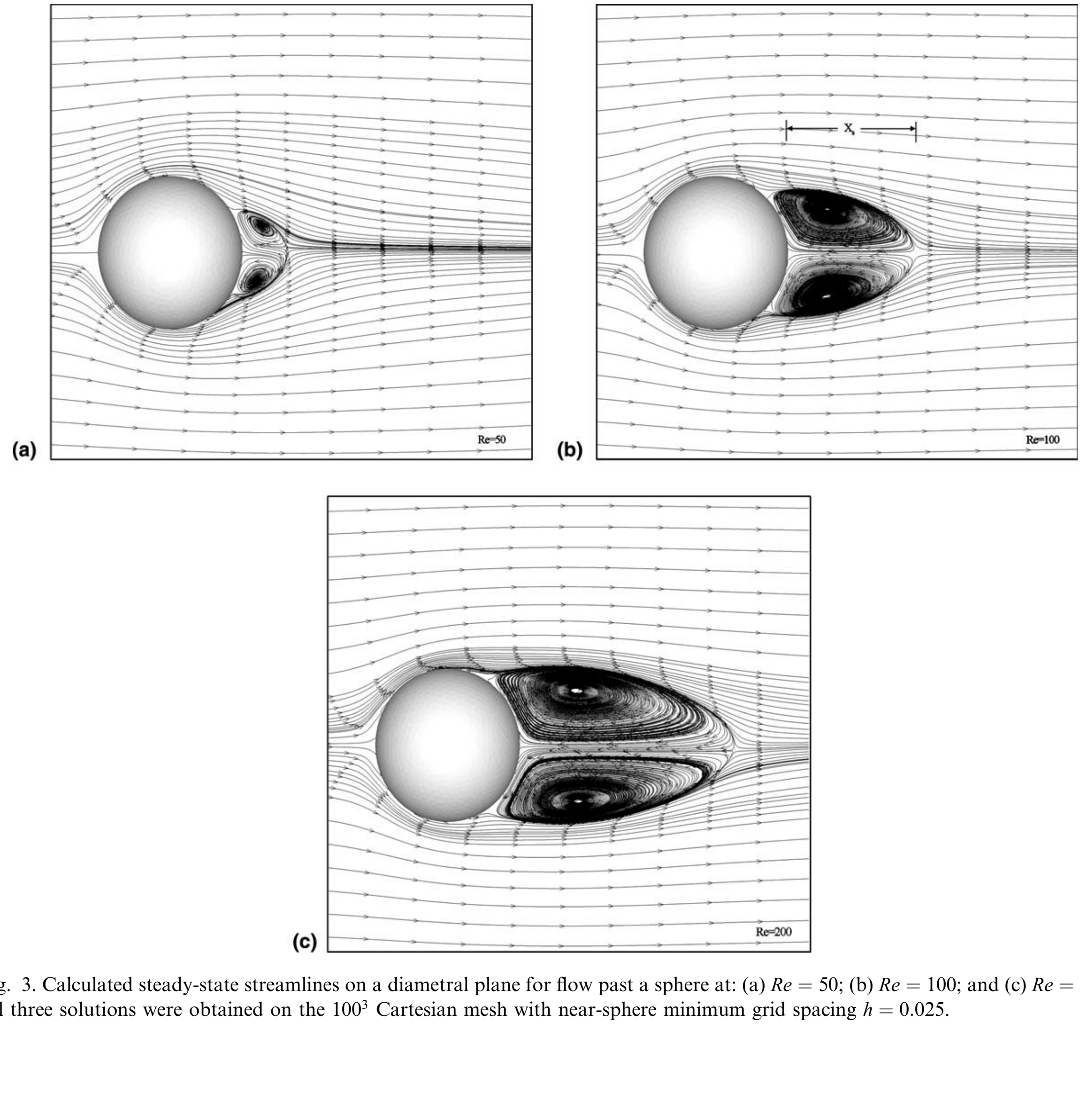

Fig. 3. Calculated steady-state streamlines on a diametral plane for flow past a sphere at: (a) $Re = 50$; (b) $Re = 100$; and (c) $Re = 200$. All three solutions were obtained on the 100^3 Cartesian mesh with near-sphere minimum grid spacing $h = 0.025$.

applied at the inflow boundary ($x = -10$). At all other boundaries of the computational domain, boundary conditions are specified by implementing in the dual time-stepping artificial compressibility algorithm, the non-reflective characteristic approach proposed in [12]. Two grid densities were considered for the Cartesian domain: a coarse grid with 50^3 grid nodes and a finer grid with 100^3. Hyperbolic tangent stretching was used to increase the spatial resolution near the surface of the sphere. More specifically, along each coordinate direction the grid nodes were clustered in the vicinity of the corresponding Cartesian planes that are tangent to the surface of the sphere, i.e., the planes x, y, and $z = 0.5$ for the grid nodes along the x, y, and z directions, respectively. The near-sphere minimum grid spacing h is constant along all three coordinate directions and is varied from 0.2 to 0.025. The surface of the sphere is discretized using 7770 triangular elements (3887 vertices). Numerical sensitivity tests showed that discretization of the sphere with finer resolution did not have any appreciable effect on the accuracy of the computed results even on the finest Cartesian grid employed in this study.

Fig. 3 shows steady-state streamlines at the x–y plane of symmetry for $Re = 50$, 100, and 200 obtained on the (100^3, $h = 0.025$) mesh. In accordance with previous experimental and computational results [11], the calculated flowfields are steady and axisymmetric with the size of the recirculating region at the downstream end of the sphere increasing continuously with Re. In Fig. 4 we compare the variation of the separation length X_s (see Fig. 3(b) for definition) with Reynolds number. For the $Re = 100$ case we also show results obtained on four grids to illustrate the importance of fine, near-body grid resolution in the accuracy of the computed results (see also Table 1). On the finest mesh resolution our computed results are

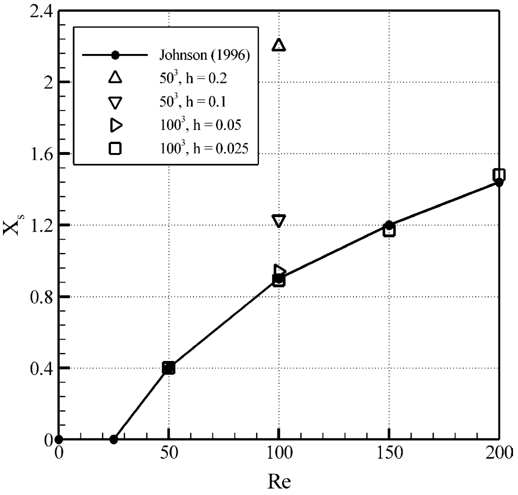

Fig. 4. Comparison of the calculated (100^3 grid with $h = 0.025$) variation of separation length X_s (see Fig. 3) with Reynolds number with the benchmark computations of [11]. For the $Re = 100$ case results obtained on four grid resolutions are also shown.

Table 1
Flow past a sphere at $Re = 100$

h	X_s	C_d	γ_{X_s}	γ_{C_d}
0.2	2.194	5.261		
0.1	1.226	2.196	1.93	1.94
0.05	0.944	1.311	2.48	2.45
0.025	0.894	1.153	2.37	2.28

Separation length, drag coefficient and corresponding convergence rates at successively finer grids. The convergence rate γ is calculated as follows: $\gamma = \log(err^{(k)}/err^{(k-1)})/\log(h^{(k)}/h^{(k-1)})$, where k is the current grid level and $k-1$ is the immediately coarser grid. The error, err, at every grid level is calculated relative to the benchmark results of [11]: $X_s = 0.882$ and $C_d = 1.112$.

in excellent agreement with the benchmark results [11]. Fig. 5 compares the variation of the calculated pressure coefficient along the θ and ϕ directions with the results of [11]. The profiles in both directions should be indistinguishable for axisymmetric flow and are shown in order to establish the degree of axisymmetry of our solutions. As seen, the calculated pressure field exhibits a high degree of axisymmetry and is in very good agreement with the benchmark data. In Fig. 6 we provide further evidence of the second-order spatial accuracy of our method by plotting the log–log variation of the error for the drag coefficient

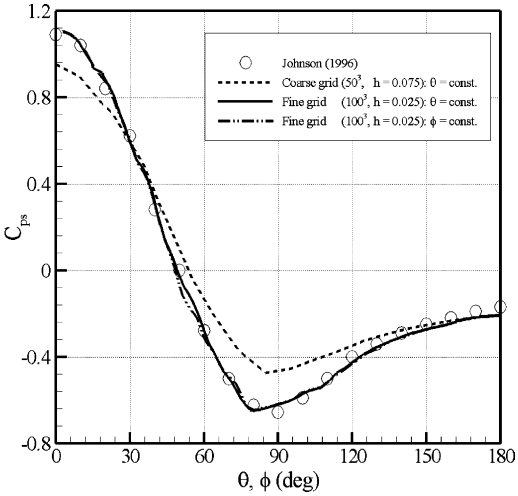

Fig. 5. Comparison of the calculated (100^3 grid with $h = 0.025$) variation of pressure coefficient $C_{ps} = 2(p_s - p_\infty)/(\rho U^2)$ on the surface of the sphere with the benchmark computations of [11] for $Re = 100$. θ is the azimuthal angle in the x–z plane and ϕ is the polar angle in the x–y plane, both measured from the negative x-axis. The computed C_{ps} on a coarse mesh is also included for comparison.

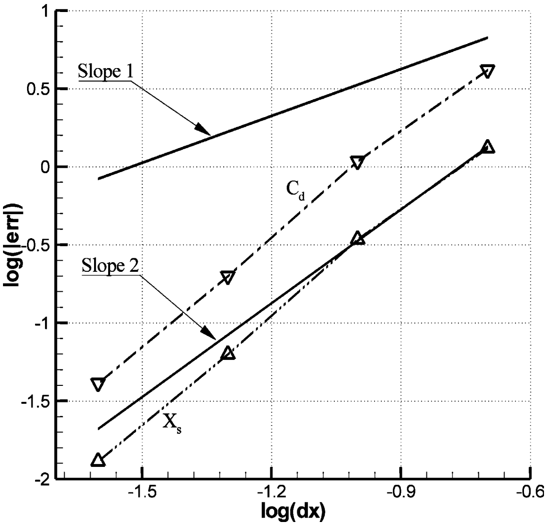

Fig. 6. Variation of the error in the predicted drag coefficient C_d (line with inverted triangles) and separation length X_s (line with triangles) with the near-sphere minimum grid spacing h for $Re = 100$. The benchmark computations of [11] are considered to be the exact solution. Lines of slope 1 and 2 have also been included for reference (see Table 1 for numerical values).

C_d and the separation length X_s as a function of the minimum, near-sphere, grid spacing h for $Re = 100$. For both quantities the results of [11] are assumed to be the "exact" solution (see Table 1 for a detailed presentation of these results).

Fig. 7(a) shows the calculated steady streamlines at two mutually perpendicular diametral planes for $Re = 250$. The solution shown in this figure was obtained by integrating the equations in a time-accurate manner and, in agreement with experimental observations and previous calculations, is steady and asymmetric. At even higher Re, unsteady flow sets in with a train of hairpin vortices shed asymmetrically in the wake of the sphere. As shown in Fig. 7(b), which visualizes coherent structures in the wake of the sphere using the λ_2-method [13], our simulations capture this feature of the flow correctly. Moreover, the

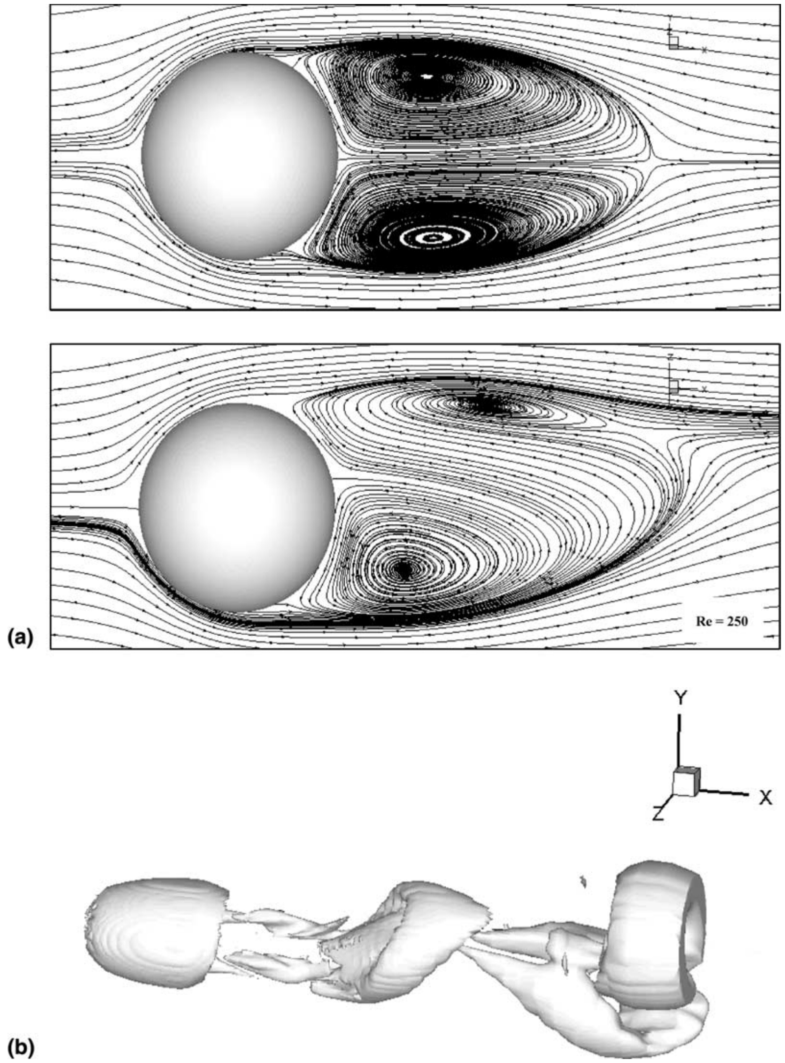

Fig. 7. (a) Calculated steady-state streamlines at two mutually perpendicular diametral planes illustrating the asymmetry of the flow for $Re = 250$. (b) Instantaneous structure of hairpin vortices in the wake of the sphere for $Re = 300$ visualized using the λ_2-method [13]. Both solutions were obtained on the 100^3 Cartesian mesh with near-sphere minimum grid spacing $h = 0.025$.

calculated Strouhal number for this case is 0.135 and this is in close agreement with the 0.137 value reported in [11].

5. Conclusions

In this note we proposed a reconstruction algorithm that is straightforward to implement, eliminates algorithmic ambiguities inherent in methods based on interpolation along grid lines and facilitates the implementation of Neumann boundary conditions for the pressure on the immersed boundary. The above results demonstrated that the method is second-order accurate and is readily applicable to flows with arbitrarily complex immersed boundaries. Our approach, therefore, should be promising for simulating complex flow/structure interaction problems on Cartesian grids.

Acknowledgements

A.G. and F.S. were supported by NSF Career Grant 9875691, a grant from Oak Ridge National Laboratory and DOE, and NIH Grant RO1-HL-07262. E.B. was supported by NIH Grant RO1-HL-07262 and a grant from the Minta Martin Foundation.

References

[1] C.S. Peskin, The immersed boundary method, Acta Numerica (2002) 1–39.

[2] T. Ye, R. Mittal, H.S. Udaykumar, W. Shyy, An accurate Cartesian grid method for viscous incompressible flows with complex immersed boundaries, J. Comput. Phys. 156 (1999) 209–240.

[3] M.P. Kirkpatrick, S.W. Armfield, J.H. Kent, A representation of curved boundaries for the solution of the Navier–Stokes equations on a staggered three-dimensional Cartesian grid, J. Comput. Phys. 184 (1) (2003) 1–36.

[4] E.A. Fadlun, R. Verzicco, P. Orlandi, J. Mohd-Yusof, Combined immersed-boundary finite-difference methods for three-dimensional complex flow simulations, J. Comput. Phys. 161 (2000) 35–60.

[5] J. Kim, D. Kim, H. Choi, An immersed-boundary finite-volume method for simulations of flow in complex geometries, J. Comput. Phys. 171 (2001) 132–150.

[6] E. Balaras, Modeling complex boundaries using an external force field on fixed Cartesian grids in large-eddy simulations, Comput. Fluids (in press).

[7] G. Tryggvason, B. Bunner, A. Esmaeeli, D. Juric, A. Al-Rawahi, W. Tauber, J. Han, S. Nas, Y.-J. Jan, A front-tracking method for the computations of multiphase flow, J. Comput. Phys. 169 (2001) 708–759.

[8] F. Sotiropoulos, S. Abdallah, The discrete continuity equation in primitive variable solutions of incompressible-flow, J. Comput. Phys. 95 (1) (1991) 212–227.

[9] F. Sotiropoulos, Y. Ventikos, Transition from bubble-type vortex breakdown to columnar vortex in a confined swirling flow, Int. J. Heat Fluid Flow 19 (1998) 446–458.

[10] T. Johnson, V.C. Patel, Flow past a sphere up to a Reynolds number of 300, J. Fluid Mech. 378 (1999) 19.

[11] T.A. Johnson, Numerical and experimental investigation of flow past a sphere up to a Reynolds number of 300, Ph.D. Dissertation, University of Iowa, 1996.

[12] J. Thomson, Time-dependent boundary conditions for hyperbolic systems II, J. Comput. Phys. 89 (1990) 439–461.

[13] J. Jeong, F. Hussein, On the identification of a vortex, J. Fluid Mech. 285 (1995) 69.