Available online at www.sciencedirect.com

ScienceDirect

Journal of Computational Physics 225 (2007) 2098–2117

JOURNAL OF COMPUTATIONAL PHYSICS

www.elsevier.com/locate/jcp

An immersed boundary method for compressible flows using local grid refinement

M.D. de Tullio [a], P. De Palma [a], G. Iaccarino [b], G. Pascazio [a,*], M. Napolitano [a]

[a] Dipartimento di Ingegneria Meccanica e Gestionale & Centro di Eccellenza in Meccanica Computazionale,
Politecnico di Bari, via Re David 200, 70125 Bari, Italy
[b] CTR, Mechanical Engineering Department, Stanford University, Stanford, CA 94305, USA

Received 29 June 2006; received in revised form 19 January 2007; accepted 12 March 2007
Available online 24 March 2007

Abstract

This paper combines a state-of-the-art method for solving the three-dimensional preconditioned Navier–Stokes equations for compressible flows with an immersed boundary approach, to provide a Cartesian-grid method for computing complex flows over a wide range of the Mach number. Moreover, a flexible local grid refinement technique is employed to achieve high resolution near the immersed body and in other high-flow-gradient regions at a fraction of the cost required by a uniformly fine grid. The method is validated versus well documented steady and unsteady test problems, for a wide range of both Reynolds and Mach numbers. Finally, and most importantly, for the case of the laminar compressible steady flow past an NACA-0012 airfoil, a thorough mesh-refinement study shows that the method is second-order accurate.

Keywords: Preconditioning; RANS equations; Semi-structured grids; Finite volume method; Second-order accuracy

1. Introduction

The immersed boundary (IB) method simplifies the grid generation process for the simulation of flows with complex and/or moving solid boundaries, by avoiding the need for a body-fitted mesh. The IB technique was originally developed for incompressible flows [1–5] using uniform and stretched Cartesian grids in order to take advantage of simple numerical algorithms. To date, the only documented IB method for compressible flows is due to some of the authors [6]. Such an approach, which uses the two-dimensional preconditioned Navier–Stokes equations to obtain accurate and efficient solutions for a wide range of the Mach number, employs structured grids, which allow a low flexibility in distributing the grid points. In fact, since mesh lines run through the entire computational domain, by clustering grid points near solid bodies a high density of grid points is obtained also in regions away from them, where flow-gradients are usually small. A flexible local grid

* Corresponding author. Tel.: +39 080 5963221; fax: +39 080 5963411.
E-mail addresses: detullio@imedado.poliba.it (M.D. de Tullio), depalma@poliba.it (P. De Palma), jops@ctr.stanford.edu (G. Iaccarino), pascazio@poliba.it (G. Pascazio), napolita@poliba.it (M. Napolitano).

0021-9991/$ - see front matter © 2007 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2007.03.008

refinement (LGR) technique is thus warranted, using a fine grid in the high-flow-gradient regions and coarsening it where the flow is smooth.

This is the very purpose of this work, which provides a second-order-accurate IB method for compressible flows in two and three space dimensions using an LGR approach to solve flows with high gradients, such as boundary layers, wakes and shocks.

The details of the numerical method are described at first, namely: the discretization scheme, the interpolation procedure at the *interface* cells, and the semi-structured approach to handle locally refined grids. The method is then validated by computing several well documented test cases. The steady incompressible flows past a circular cylinder and a sphere at moderate Reynolds numbers are considered to test both the LGR-IB technique and the preconditioning strategy, in two and three dimensions, respectively. The low-speed unsteady flow past a heated circular cylinder is computed to test the unsteady dual-time-stepping approach as well as the implementation of the energy equation and of its boundary conditions. The laminar compressible flow past an NACA-0012 airfoil, the supersonic turbulent flow past a circular cylinder, and the transonic turbulent flow past an RAE2822 airfoil are then considered to validate the LGR-IB technique in the presence of a critical separation region, very high flow-gradients and turbulence. Finally, and most importantly, a thorough mesh-refinement study is performed for the steady laminar flow past an NACA-0012 airfoil, which shows that the proposed LGR-IB method is indeed second-order accurate at all interior and boundary cells.

2. Governing equations and numerical method

The Reynolds averaged Navier–Stokes (RANS) equations, written in terms of Favre mass-averaged quantities and using the *k*–*ω* turbulence model, can be written as follows:

$$\frac{\partial \rho}{\partial t}+\frac{\partial}{\partial x_j}(\rho u_j)=0, \tag{1}$$

$$\frac{\partial(\rho u_i)}{\partial t}+\frac{\partial}{\partial x_j}(\rho u_j u_i)=-\frac{\partial p_{\mathrm{t}}}{\partial x_{\mathrm{i}}}+\frac{\partial \hat{\tau}_{ji}}{\partial x_j}, \tag{2}$$

$$\frac{\partial(\rho \tilde{H}-p_{\mathrm{t}})}{\partial t}+\frac{\partial}{\partial x_j}(\rho u_j \tilde{H})=\frac{\partial}{\partial x_j}\left[u_i\hat{\tau}_{ij}+(\mu+\sigma^*\mu_{\mathrm{t}})\frac{\partial k}{\partial x_j}-q_j\right], \tag{3}$$

$$\frac{\partial(\rho k)}{\partial t}+\frac{\partial}{\partial x_j}(\rho u_j k)=\tau_{ij}\frac{\partial u_i}{\partial x_j}-\beta^*\rho\omega k+\frac{\partial}{\partial x_j}\left[(\mu+\sigma^*\mu_{\mathrm{t}})\frac{\partial k}{\partial x_j}\right], \tag{4}$$

$$\frac{\partial(\rho \omega)}{\partial t}+\frac{\partial}{\partial x_j}(\rho u_j \omega)=\frac{\gamma\omega}{k}\tau_{ij}\frac{\partial u_i}{\partial x_j}-\beta\rho\omega^2+\frac{\partial}{\partial x_j}\left[(\mu+\sigma\mu_{\mathrm{t}})\frac{\partial \omega}{\partial x_j}\right]. \tag{5}$$

In the equations above: $\tilde{H}$ and p_{t} are the total enthalpy and the pressure, comprehensive of the turbulent kinetic energy, k,

$$\tilde{H}=h+\frac{1}{2}|\mathbf{u}|^2+\frac{5}{3}k, \quad p_{\mathrm{t}}=p+\frac{2}{3}\rho k. \tag{6}$$

The eddy viscosity, μ_{t}, is defined in terms of k and of the specific dissipation rate, ω, according to the *k*–*ω* turbulence model of Wilcox [7], namely:

$$\mu_{\mathrm{t}}=\gamma^*\frac{\rho k}{\omega}. \tag{7}$$

$\hat{\tau}_{ij}$ indicates the sum of the molecular and Reynolds (τ_{ij}) stress-tensor components. According to the Boussinesq approximation, one has:

$$\hat{\tau}_{ij}=(\mu+\mu_{\mathrm{t}})\left[\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_{\mathrm{i}}}-\frac{2}{3}\frac{\partial u_k}{\partial x_k}\delta_{ij}\right]-\frac{2}{3}\rho k\delta_{ij}. \tag{8}$$

The heat flux vector components, q_j, are given as:

$$q_j=-\left(\frac{\mu}{Pr}+\frac{\mu_{\mathrm{t}}}{Pr_{\mathrm{t}}}\right)\frac{\partial h}{\partial x_j}, \tag{9}$$

where Pr and Pr_{t} are the laminar and turbulent Prandtl numbers, equal to 0.71 and 1, respectively.

Finally, the molecular viscosity coefficient is computed by Sutherland's law and the coefficients of the low-Reynolds-number k–ω turbulence model of [7] are used.

It is convenient to rewrite the RANS equations (1)–(5) in compact form, as:

$$\frac{\partial Q}{\partial t}+\frac{\partial E}{\partial x}+\frac{\partial F}{\partial y}+\frac{\partial G}{\partial z}-\frac{\partial E_{\mathrm{v}}}{\partial x}-\frac{\partial F_{\mathrm{v}}}{\partial y}-\frac{\partial G_{\mathrm{v}}}{\partial z}=D, \tag{10}$$

where Q is the conservative variable vector, E, F, G and E_{v}, F_{v}, G_{v} indicate the inviscid and viscous fluxes, respectively, and D is the vector of the source terms.

A pseudo-time derivative is added to the left-hand side of Eq. (10) in order to use a time marching approach for both steady and unsteady flows; the preconditioning matrix, $\mathbf{\Gamma}$, proposed in [8–10], is used to premultiply the pseudo-time derivative in order to maintain a good accuracy and efficiency of the method at all flow speeds. The final system reads:

$$\mathbf{\Gamma}\frac{\partial Q_{\mathrm{v}}}{\partial \tau}+\frac{\partial Q}{\partial t}+\frac{\partial E}{\partial x}+\frac{\partial F}{\partial y}+\frac{\partial G}{\partial z}-\frac{\partial E_{\mathrm{v}}}{\partial x}-\frac{\partial F_{\mathrm{v}}}{\partial y}-\frac{\partial G_{\mathrm{v}}}{\partial z}=D, \tag{11}$$

where $Q_{\mathrm{v}}=(p_{\mathrm{t}},u,v,w,T,k,\omega)^{\mathrm{T}}$ is the primitive variable vector, which is related to Q by the Jacobian matrix $\mathbf{P}=\partial Q/\partial Q_{\mathrm{v}}$. Discretizing equation (11) by an Euler implicit scheme in the pseudo-time and approximating the physical-time derivative by a second-order-accurate three-point backward difference, the following semi-discrete equation is obtained in delta form:

$$\begin{aligned}&\left[\mathbf{\Gamma}+\frac{3}{2}\frac{\Delta\tau}{\Delta t}\mathbf{P}+\Delta\tau\frac{\partial}{\partial x}\left(\mathbf{A}_{\mathrm{v}}-\mathbf{R}_{xx}\frac{\partial}{\partial x}-\mathbf{R}_{xy}\frac{\partial}{\partial y}-\mathbf{R}_{xz}\frac{\partial}{\partial z}\right)+\Delta\tau\frac{\partial}{\partial y}\left(\mathbf{B}_{\mathrm{v}}-\mathbf{R}_{yx}\frac{\partial}{\partial x}-\mathbf{R}_{yy}\frac{\partial}{\partial y}-\mathbf{R}_{yz}\frac{\partial}{\partial z}\right)\right.\\&\left.\quad+\Delta\tau\frac{\partial}{\partial z}\left(\mathbf{C}_{\mathrm{v}}-\mathbf{R}_{zx}\frac{\partial}{\partial x}-\mathbf{R}_{zy}\frac{\partial}{\partial y}-\mathbf{R}_{zz}\frac{\partial}{\partial z}\right)\right]\Delta Q_{\mathrm{v}}=-\Delta\tau\left[\frac{3Q^{\mathrm{r}}-4Q^{n}+Q^{n-1}}{2\Delta t}+\mathscr{R}^{\mathrm{r}}\right],\end{aligned} \tag{12}$$

where r and $\Delta\tau$ indicate the pseudo-time level and step, n and Δt indicate the physical-time level and step, $\mathbf{A}_{\mathrm{v}}=\partial E/\partial Q_{\mathrm{v}}$, $\mathbf{B}_{\mathrm{v}}=\partial F/\partial Q_{\mathrm{v}}$, $\mathbf{C}_{\mathrm{v}}=\partial G/\partial Q_{\mathrm{v}}$, and $\mathbf{R}_{ij}$ are the viscous coefficient matrices [11]. The steady residual is given as:

$$\mathscr{R}^{\mathrm{r}}=\frac{\partial(E^{\mathrm{r}}-E^{\mathrm{r}}_{\mathrm{v}})}{\partial x}+\frac{\partial(F^{\mathrm{r}}-F^{\mathrm{r}}_{\mathrm{v}})}{\partial y}+\frac{\partial(G^{\mathrm{r}}-G^{\mathrm{r}}_{\mathrm{v}})}{\partial z}-D^{\mathrm{r}}, \tag{13}$$

and the delta unknowns to be annihilated iteratively (at each physical time step, for the case of unsteady flows) read:

$$\Delta Q_{\mathrm{v}}=Q_{\mathrm{v}}^{r+1}-Q_{\mathrm{v}}^{\mathrm{r}}. \tag{14}$$

The left-hand side (LHS) of Eq. (12) is modified to improve the efficiency of the method, without affecting the residual, that is, the solution. Firstly, the non-orthogonal viscous coefficient matrices, $\mathbf{R}_{xy}$, $\mathbf{R}_{xz}$, $\mathbf{R}_{yx}$, $\mathbf{R}_{yz}$, $\mathbf{R}_{zx}$, and $\mathbf{R}_{zy}$, are neglected, and the remaining ones are approximated by the corresponding spectral radii multiplied times the identity matrix, $\mathbf{R}_{xx}=R_x\mathbf{I}$, $\mathbf{R}_{yy}=R_y\mathbf{I}$, and $\mathbf{R}_{zz}=R_z\mathbf{I}$; then, as proposed in [11], the pseudo- and physical-time terms are grouped together into a new term $\mathbf{S}$,

$$\mathbf{S}=\mathbf{\Gamma}+\frac{3}{2}\frac{\Delta\tau}{\Delta t}\mathbf{P}, \tag{15}$$

which is factored out of the LHS in Eq. (12), yielding:

$$\begin{aligned}&\mathbf{S}\left[\mathbf{I}+\Delta\tau\mathbf{S}^{-1}\frac{\partial}{\partial x}\left(\mathbf{A}_{\mathrm{v}}-R_x\mathbf{I}\frac{\partial}{\partial x}\right)+\Delta\tau\mathbf{S}^{-1}\frac{\partial}{\partial y}\left(\mathbf{B}_{\mathrm{v}}-R_y\mathbf{I}\frac{\partial}{\partial y}\right)+\Delta\tau\mathbf{S}^{-1}\frac{\partial}{\partial z}\left(\mathbf{C}_{\mathrm{v}}-R_z\mathbf{I}\frac{\partial}{\partial z}\right)\right]\Delta Q_{\mathrm{v}}\\&\quad=-\Delta\tau\left[\frac{3Q^{\mathrm{r}}-4Q^{n}+Q^{n-1}}{2\Delta t}+\mathscr{R}^{\mathrm{r}}\right].\end{aligned} \tag{16}$$

In order to solve the resulting linear system, the diagonalization procedure of Pulliam and Chaussee [12] is applied. The matrices $\mathbf{S}^{-1}\mathbf{A}_{\mathrm{v}}$, $\mathbf{S}^{-1}\mathbf{B}_{\mathrm{v}}$ and $\mathbf{S}^{-1}\mathbf{C}_{\mathrm{v}}$ are written as:

$$\mathbf{S}^{-1}\mathbf{A}_{\mathrm{v}} = \mathbf{M}_x\mathbf{\Lambda}_x\mathbf{M}_x^{-1}, \tag{17}$$

$$\mathbf{S}^{-1}\mathbf{B}_{\mathrm{v}} = \mathbf{M}_y\mathbf{\Lambda}_y\mathbf{M}_y^{-1}, \tag{18}$$

$$\mathbf{S}^{-1}\mathbf{C}_{\mathrm{v}} = \mathbf{M}_z\mathbf{\Lambda}_z\mathbf{M}_z^{-1}, \tag{19}$$

where $\mathbf{M}_x$, $\mathbf{M}_y$, $\mathbf{M}_z$ are the right-eigenvector matrices, $\mathbf{M}_x^{-1}$, $\mathbf{M}_y^{-1}$, $\mathbf{M}_z^{-1}$ are the left-eigenvector matrices, and $\mathbf{\Lambda}_x$, $\mathbf{\Lambda}_y$ and $\mathbf{\Lambda}_z$ are diagonal matrices containing the eigenvalues of $\mathbf{S}^{-1}\mathbf{A}_{\mathrm{v}}$, $\mathbf{S}^{-1}\mathbf{B}_{\mathrm{v}}$ and $\mathbf{S}^{-1}\mathbf{C}_{\mathrm{v}}$, respectively. Therefore, Eq. (16) is rewritten as:

$$\begin{aligned}&\mathbf{S}\left[\mathbf{I}+\Delta\tau\mathbf{M}_x\frac{\partial}{\partial x}\left(\mathbf{\Lambda}_x - R_x\mathbf{I}\frac{\partial}{\partial x}\right)\mathbf{M}_x^{-1}+\Delta\tau\mathbf{M}_y\frac{\partial}{\partial y}\left(\mathbf{\Lambda}_y - R_y\mathbf{I}\frac{\partial}{\partial y}\right)\mathbf{M}_y^{-1}+\Delta\tau\mathbf{M}_z\frac{\partial}{\partial z}\left(\mathbf{\Lambda}_z - R_z\mathbf{I}\frac{\partial}{\partial z}\right)\mathbf{M}_z^{-1}\right]\Delta Q_{\mathrm{v}}\\ &\quad = -\Delta\tau\left[\frac{3Q^{\mathrm{r}}-4Q^n+Q^{n-1}}{2\Delta t}+\mathscr{R}^{\mathrm{r}}\right].\end{aligned} \tag{20}$$

The LHS of Eq. (20) is then factorized, to reduce the bandwidth of the matrices associated with the algebraic system to be solved at each pseudo-time level:

$$\begin{aligned}&\mathbf{S}\mathbf{M}_x\left[\mathbf{I}+\Delta\tau\frac{\partial}{\partial x}\left(\mathbf{\Lambda}_x - R_x\mathbf{I}\frac{\partial}{\partial x}\right)\right]\mathbf{M}_x^{-1}\mathbf{M}_y\left[\mathbf{I}+\Delta\tau\frac{\partial}{\partial y}\left(\mathbf{\Lambda}_y - R_y\mathbf{I}\frac{\partial}{\partial y}\right)\right]\mathbf{M}_y^{-1}\mathbf{M}_z\left[\mathbf{I}+\Delta\tau\frac{\partial}{\partial z}\left(\mathbf{\Lambda}_z - R_z\mathbf{I}\frac{\partial}{\partial z}\right)\right]\mathbf{M}_z^{-1}\Delta Q_{\mathrm{v}}\\ &\quad = -\Delta\tau\left[\frac{3Q^{\mathrm{r}}-4Q^n+Q^{n-1}}{2\Delta t}+\mathscr{R}^{\mathrm{r}}\right].\end{aligned} \tag{21}$$

Finally, the BiCGStab [13] approach is employed to solve the system above by the following six steps:

$$\left[\mathbf{I}+\Delta\tau\frac{\partial}{\partial x}\left(\mathbf{\Lambda}_x - R_x\mathbf{I}\frac{\partial}{\partial x}\right)\right]\Delta Q_{\mathrm{v}}^{(1)} = -\Delta\tau\mathbf{M}_x^{-1}\mathbf{S}^{-1}\left[\frac{3Q^{\mathrm{r}}-4Q^n+Q^{n-1}}{2\Delta t}+\mathscr{R}^{\mathrm{r}}\right], \tag{22}$$

$$\Delta Q_{\mathrm{v}}^{(2)} = \mathbf{M}_x\Delta Q_{\mathrm{v}}^{(1)}, \tag{23}$$

$$\left[\mathbf{I}+\Delta\tau\frac{\partial}{\partial y}\left(\mathbf{\Lambda}_y - R_y\mathbf{I}\frac{\partial}{\partial y}\right)\right]\Delta Q_{\mathrm{v}}^{(3)} = \mathbf{M}_y^{-1}\Delta Q_{\mathrm{v}}^{(2)}, \tag{24}$$

$$\Delta Q_{\mathrm{v}}^{(4)} = \mathbf{M}_y\Delta Q_{\mathrm{v}}^{(3)}, \tag{25}$$

$$\left[\mathbf{I}+\Delta\tau\frac{\partial}{\partial z}\left(\mathbf{\Lambda}_z - R_z\mathbf{I}\frac{\partial}{\partial z}\right)\right]\Delta Q_{\mathrm{v}}^{(5)} = \mathbf{M}_z^{-1}\Delta Q_{\mathrm{v}}^{(4)}, \tag{26}$$

$$\Delta Q_{\mathrm{v}} = \mathbf{M}_z\Delta Q_{\mathrm{v}}^{(5)}. \tag{27}$$

A collocated cell-centred finite volume space discretization is used. The convective terms at the right-hand side are discretized using either an upwind flux-difference-splitting scheme with second- or third-order accuracy, or a second-order-accurate centred scheme. The viscous terms are discretized by second-order-accurate centred differences. The LHS convective term is always discretized using a first-order-accurate upwind scheme, according to a deferred-correction approach, so as to guarantee the diagonal dominance of the matrices, while reducing their bandwidth. Finally, the boundary conditions are imposed explicitly, namely, using the flow variables computed at the old pseudo-time level.

2.1. Grid generation

The IB technique used in this work is based on that proposed in [3,4]. In a preliminary step, the geometry under consideration, which is described by a closed curve in two dimensions (a closed surface in three dimensions), is overlapped onto a Cartesian (non uniform) grid. Using the ray tracing technique based on the geometrical algorithms reported in [14], the computational cells occupied entirely by the flow are tagged as *fluid* cells; those whose centres lie inside the immersed body are tagged as *solid* cells; the remaining ones are tagged as *interface* cells. The present local grid refinement (LGR) procedure follows those introduced in [15,16]. The

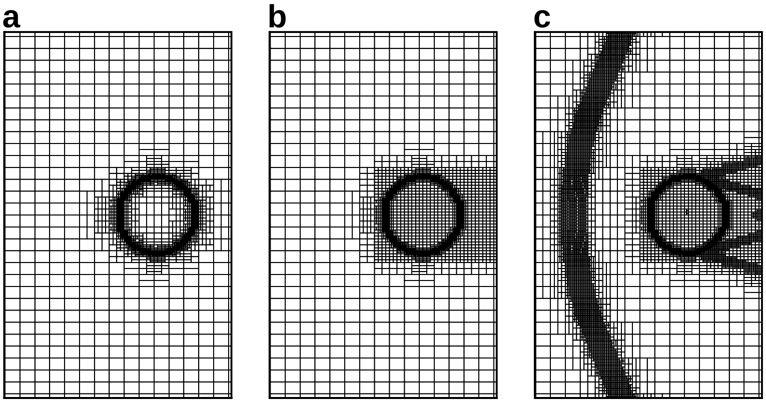


Fig. 1. Semi-structured grids with different refinements: (a) immersed boundary; (b) immersed boundary + wake; (c) immersed boundary + wake + shock.

following description of the algorithm refers to two dimensions, the extension to three dimensions being straightforward. Starting from an initial Cartesian coarse grid, the auxiliary grid is generated by recursively halving the mesh size, until an assigned target value is reached at the immersed boundary. In more detail, a tag function, generated using the ray tracing technique, is used to mark the cells inside and outside the immersed body: an integer value ±1 is assigned to *fluid* and *solid* cells, respectively. The numerical gradient of this function is different from zero only at the immersed boundary and depends on the local grid size. The components of such a gradient in the x and y directions are used to select the cells to be refined. The refinement process is terminated when a user specified resolution, namely, a prescribed value of the gradient of the tagging function, is achieved at the boundary. It is noteworthy that such a refinement extends to all rows and columns of the computational grid, so that it applies also to regions of high flow-gradients. Then, the semi-structured mesh is obtained by coarsening (grouping) the cells far from the boundaries and such regions until the maximum prescribed cell-size is achieved, see, e.g., Fig. 1. It is noteworthy that this approach is very efficient insofar as the cell tagging takes full advantage of the alignment of the cell centres and the grid nodes.

2.2. Data structure

The auxiliary (uniformly fine structured Cartesian) grid is employed to handle the data structure of the semi-structured locally refined grid (see Fig. 2). The auxiliary grid covers the whole computational domain

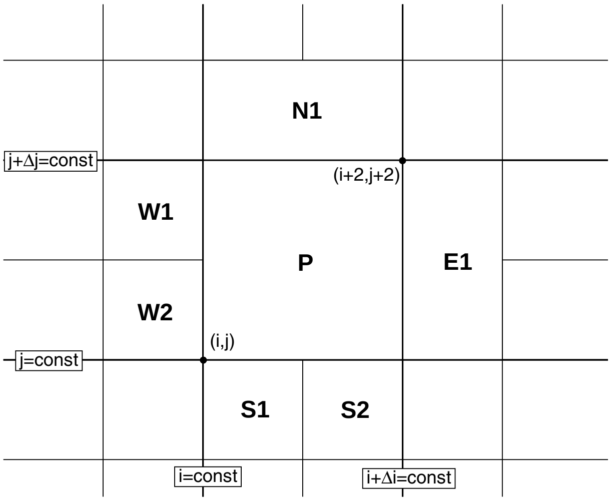


Fig. 2. Locally refined grid showing a cell P and its neighbours.

employing the finest mesh level of the semi-structured grid. Therefore, each cell of the semi-structured grid is bounded by the lines passing through the vertices (i,j) and $(i+\Delta i, j+\Delta j)$, where the indices $i=0,\ldots,N_i$ and $j=0,\ldots,N_j$ refer to the auxiliary grid and Δi and Δj are integers, equal to or greater than one, depending on (i,j), see Fig. 2. The major advantage of this approach with respect to classical OCTREE based [17] and fully-unstructured [18] ones lies in the economy and flexibility of storing and retrieving connectivity informations employing the auxiliary grid. In particular, having an auxiliary grid with $N_i \times N_j$ cells, the $N < N_i \times N_j$ cells belonging to the semi-structured grid are defined using the two couple of indices (i,j) and $(\Delta i, \Delta j)$, with a total memory requirement of $4N$ integers. In addition, an array of integers, $ID(i,j)$, is needed to store the correspondence between the cells of the auxiliary and semi-structured grids. All of the cells of the auxiliary grid not employed in the semi-structured one, namely, those included in the range $[i : i+\Delta i-1]$ and $[j : j+\Delta j-1]$, are tagged using the same cell number (see Figs. 2 and 3). The total storage required for allocating $ID(i,j)$ is equal to $N_i \times N_j$ integers. The connectivity information for each cell are retrieved by querying the array $ID(i,j)$.

2.3. Flux evaluation in the semi-structured grid

For each face, the contributions of the neighbouring cells are collected to build the corresponding convective operators for the cell. For example, referring to the cell L_1 in Fig. 4, the flux balance along the horizontal x direction reads:

$$\int_V \frac{\partial E}{\partial x}\delta V = (E^{\mathrm{r}}_{\mathrm{L}_1} - E^{\mathrm{l}}_{\mathrm{L}_1})S_x, \tag{28}$$

where r and l indicate the right and left faces of the cell, having measure S_x. In particular, $E^{\mathrm{r}}_{\mathrm{L}_1}$ is evaluated as follows:

$$\begin{aligned} E^{\mathrm{r}}_{\mathrm{L}_1} = \frac{1}{N^{\mathrm{r}}_{\mathrm{L}_1}} \sum_{j=1}^{N^{\mathrm{r}}_{\mathrm{L}_1}} \Bigg\{ & \frac{1}{2}(E_{\mathrm{R}_j} + E_{\mathrm{L}_1}) - \phi_1(\mathbf{\Gamma M}_x|\mathbf{\Lambda}_x|)^{\mathrm{r}}_{\mathrm{L}_1,j}(\mathbf{M}_x^{-1})^{\mathrm{r}}_{\mathrm{L}_1,j}(Q_{v,\mathrm{R}_j} - Q_{v,\mathrm{L}_1}) - \frac{\phi_2}{N^{l}_{\mathrm{L}_1}} \\ & \times (\mathbf{\Gamma M}_x\mathbf{\Lambda}_x^{+})^{\mathrm{r}}_{\mathrm{L}_1,j}\left[\sum_{i=1}^{N^{l}_{\mathrm{L}_1}}(\mathbf{M}_x^{-1})^{l}_{\mathrm{L}_1,i}(Q_{v,\mathrm{L}_1} - Q_{v,\mathrm{L}_{1i}})\right] + \frac{\phi_2}{N^{\mathrm{r}}_{\mathrm{R}_j}}(\mathbf{\Gamma M}_x\mathbf{\Lambda}_x^{-})^{\mathrm{r}}_{\mathrm{L}_1,j}\left[\sum_{i=1}^{N^{\mathrm{r}}_{\mathrm{R}_j}}(\mathbf{M}_x^{-1})^{\mathrm{r}}_{\mathrm{R}_j,i}(Q_{v,\mathrm{R}_{ji}} - Q_{v,\mathrm{R}_j})\right]\Bigg\}. \end{aligned} \tag{29}$$

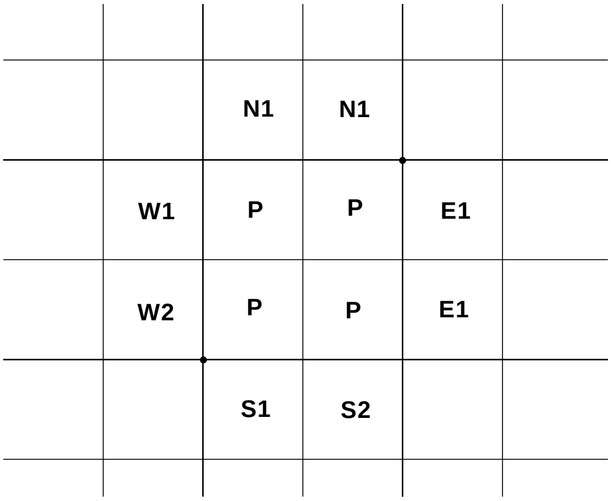


Fig. 3. Cell identification array on the fine underlying grid (ID) showing one-to-one connectivity.

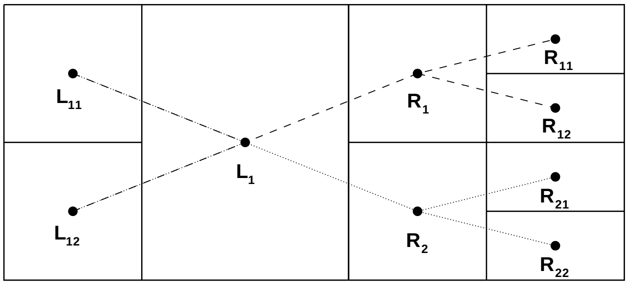


Fig. 4. Flux evaluation in the x-direction at the right face of the cell L_1.

In the equation above, R and L indicate the cell centres at the right and left sides of the face, respectively, one and two subscripts being employed for the first and second row of cells (see Fig. 4); moreover, N_X^y is the number of the neighbouring cells of the cell X on the y side. Matrices $\mathbf{\Gamma}$, $\mathbf{M}_x$, $\mathbf{\Lambda}_x^{\pm}$ and $\mathbf{M}_x^{-1}$ are calculated using the averaged variables according to Roe's linearization [19] and the values of ϕ_1 and ϕ_2 are provided in Table 1 for the three schemes used in this work. The maximum number of neighbours for each face is limited to two for 2D computations and four for 3D ones, each face being split into equal parts. A similar approach is employed to build the centred diffusive operator. When computing flows with shocks, a total variation diminishing scheme, with minmod limiter function, is employed in conjunction with the second- and third-order-accurate upwind schemes.

A final comment is warranted. The local refinement strategy, handled by the data structure described above, allows one to reduce dramatically the number of cells with respect to the Cartesian grid covering the whole computational domain with the same finest mesh level, namely, the auxiliary grid. In fact, in the present calculations the reduction of the number of cells varies in the range 85–98%, depending on the test case. On the other hand, the CPU overhead per iteration and cell, with respect to the same code using the (uniformly fine) auxiliary grid is always less then 10%. Considering that local coarsening also accelerates convergence, the efficiency gain is paramount.

2.4. Boundary conditions

The application of the boundary conditions at the immersed boundary is treated explicitly by assigning the values of the variables at the *interface* cells. At *solid* cells, the velocity components are set to zero and, in the case of assigned surface temperature, the temperature is set equal to the wall value. At the *interface* cells, the direct forcing approach is used, namely, the flow variables are computed by interpolating the values at the surrounding *fluid* cells, at the old pseudo-time level, and the assigned values at the wall. For each *interface* cell it is possible to find N_{nbr} contiguous *fluid* cells and N_{ib} intersections of the faces of the cell with the immersed boundary. For the case of Dirichlet boundary conditions, the following interpolation formula is used:

$$\phi_{\text{int}} = \sum_{i}^{N_{nbr}} \frac{\alpha_i}{q} \phi_i + \sum_{j}^{N_{ib}} \frac{\beta_j}{q} \phi_{j,\text{wall}}, \tag{30}$$

where $\phi_{j,\text{wall}}$ is the value of the flow variable to be imposed at the immersed surface, $\alpha_i = 1/d_i$ and $\beta_j = 1/D_j$, d_i and D_j being the distances of each *interface* cell centre from each surrounding cell centre and from each wall intersection, respectively, and

Table 1
Coefficients for the three schemes

Order	ϕ_1	ϕ_2
2nd-Upwind	1/2	−1/2
2nd-Centred	0	0
3rd-Upwind	1/6	−1/6

$$q = \sum_{i}^{N_{nbr}} \alpha_i + \sum_{j}^{N_{ib}} \beta_j. \tag{31}$$

It is noteworthy that, in the one-dimensional case, this procedure recovers the linear interpolation scheme used in [3,6]. For the case of Neumann boundary conditions, the prescribed normal derivative is imposed at the interface cells and $\phi_{j,\mathrm{wall}}$ and ϕ_{int} are related by assuming a linear reconstruction for ϕ and approximating the normal direction with the direction along which the distance D_j is calculated, to give:

$$\phi_{j,\mathrm{wall}} = \phi_{\mathrm{int}} - \left(\frac{\partial \phi}{\partial n}\right)_{\mathrm{int}} D_j. \tag{32}$$

Substituting Eq. (32) into Eq. (30), the value of the flow variable to be imposed at the *interface* cell is finally obtained as:

$$\phi_{\mathrm{int}} = \frac{\sum_{i}^{N_{nbr}} \frac{\alpha_i}{q} \phi_i - \left(\frac{\partial \phi}{\partial n}\right)_{\mathrm{int}} \frac{1}{q} N_{ib}}{1 - \sum_{j}^{N_{ib}} \frac{\beta_j}{q}}. \tag{33}$$

For the case of zero normal derivative at the wall one has $(\partial \phi/\partial n)_{\mathrm{int}} = 0$ and the final equation reads:

$$\phi_{\mathrm{int}} = \frac{\sum_{i}^{N_{nbr}} \frac{\alpha_i}{q} \phi_i}{1 - \sum_{j}^{N_{ib}} \frac{\beta_j}{q}}. \tag{34}$$

Such a condition is always imposed to the pressure, according to the boundary layer theory, and also to the temperature, in the case of adiabatic walls. As far as the other boundary conditions are concerned, the computational domain, which extends far enough away from the body, is limited by inlet and outlet boundaries, where standard characteristic boundary conditions are imposed, and far-field boundaries, where free-shear boundary conditions are imposed.

3. Results

Several well documented test cases have been used to validate the proposed methodology. In all calculations, the L_2 norm of the maximum residual has been reduced to 10^{-7} and the second-order-accurate upwind scheme has been used for the inviscid fluxes, unless otherwise specified.

3.1. Incompressible flow past a circular cylinder

The two-dimensional incompressible flow past a circular cylinder has been considered to test both the preconditioning strategy and the immersed boundary method versus steady flows at very low Mach numbers. The free-stream Mach and Reynolds numbers, the latter being based on the cylinder diameter, D, are: $M = 0.03$ and $Re = 40$. A rectangular computational domain has been used with the inlet and outlet vertical boundaries located at $x_{\mathrm{i}} = -40D$ and $x_{\mathrm{o}} = 80D$ and the two horizontal far-field boundaries located at $y_{\mathrm{ff}} = \pm 40D$, respectively, the origin coinciding with the centre of the cylinder. The second-order-accurate centred scheme has been employed for the inviscid fluxes. Computations have been performed using a semi-structured grid with 39,964 cells and 283,106 faces, based on an auxiliary grid with 822×407 cells. A local view of the mesh is given in Fig. 5. The computed length of the symmetrical separation region, L, angle of separation, θ, both measured from the trailing edge of the cylinder, and drag coefficient, C_{D}, are provided in Table 2, together with the corresponding experimental [20,21] and numerical [22–24] results available in the literature. The agreement is quite satisfactory.

3.2. Incompressible flow past a sphere

The flow past a sphere has been computed to test the flow solver and the immersed boundary method versus a three-dimensional application. A single value of the free-stream Mach number, $M = 0.03$, and four values of

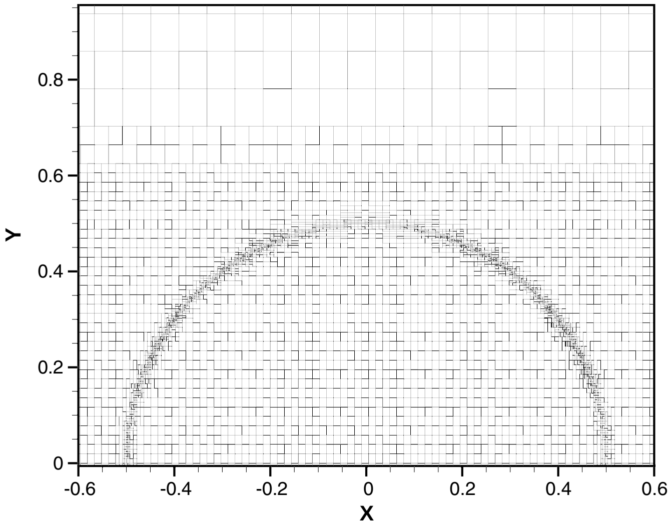


Fig. 5. Local view of the grid.

Table 2
Incompressible steady flow past a circular cylinder at $Re = 40$

	L	θ (°)	C_D
Fornberg [22]	2.24	55.6	1.50
Dennis and Chang [23]	2.35	53.8	1.52
Coutanceau and Bouard [20]	2.13	53.5	–
Tritton [21]	–	–	1.59
Linnick and Fasel [24]	2.28	53.6	1.54
Present	2.23	53.7	1.49

the Reynolds number (based on the sphere diameter, D), namely, 40, 60, 80, and 100, have been considered. The inlet and outlet boundary planes are located at $x_i = -40D$ and $x_o = 80D$ and the far-field boundaries are located at $y_{ff} = \pm 40D$ and $z_{ff} = \pm 40D$, the origin of the computational box coinciding with the centre of the

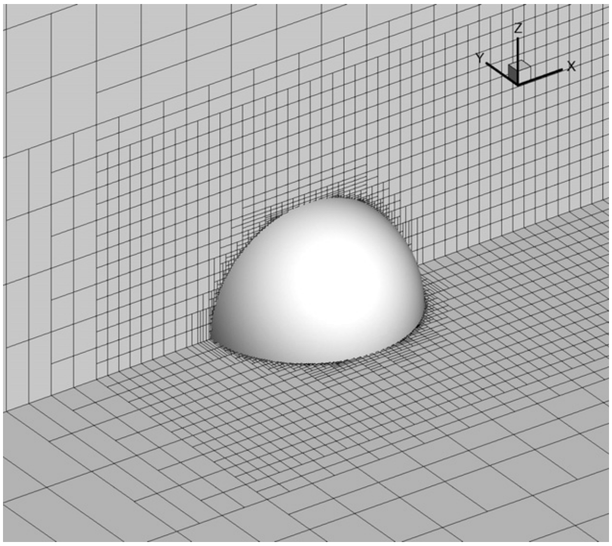


Fig. 6. Local view of the grid.

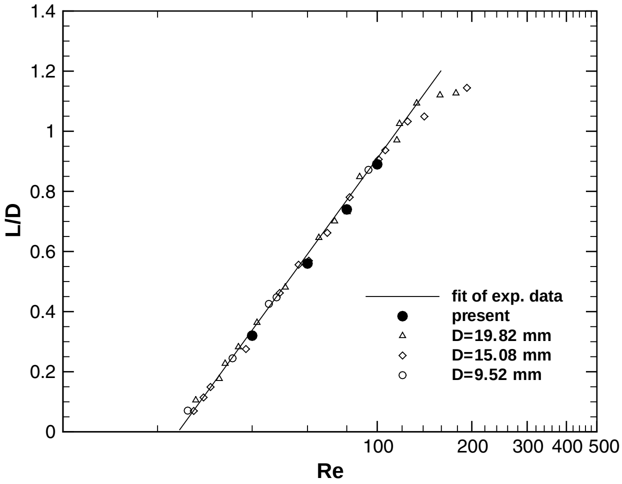


Fig. 7. Length of the separation bubble vs *Re*.

sphere. Computations have been performed using the third-order-accurate upwind scheme and a semi-structured mesh with 231,031 cells and 2,182,547 faces, the corresponding auxiliary mesh containing 446 × 199 × 185 cells, namely, 70 times more cells. The grid is refined at the sphere surface and in a box surrounding the sphere and the wake, in order to describe accurately the boundary layer, the separation region, and the wake. A local view of the mesh is given in Fig. 6. According to Batchelor [25], the flow around a sphere firstly separates at $Re \simeq 24$ and the axial length of the separation bubble grows linearly up to $Re \simeq 100$, see Fig. 7, where the present results are seen to agree well with such an experimental finding. Moreover, Figs. 8 and 9 show the drag coefficient versus *Re* and the computed pressure coefficient profile along the sphere surface for $Re = 100$, as compared with the experimental data of Clift et al. [26] and the benchmark solution of Fornberg [22], respectively.

3.3. Unsteady flow past a heated circular cylinder

The unsteady two-dimensional low-Mach-number flow past a heated circular cylinder has been used as an interesting test case to validate both the time accuracy of the method and the correct implementation of the

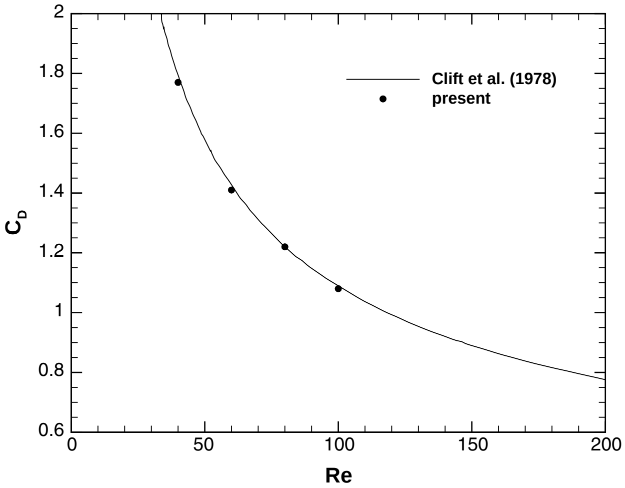


Fig. 8. Drag coefficient vs *Re*.

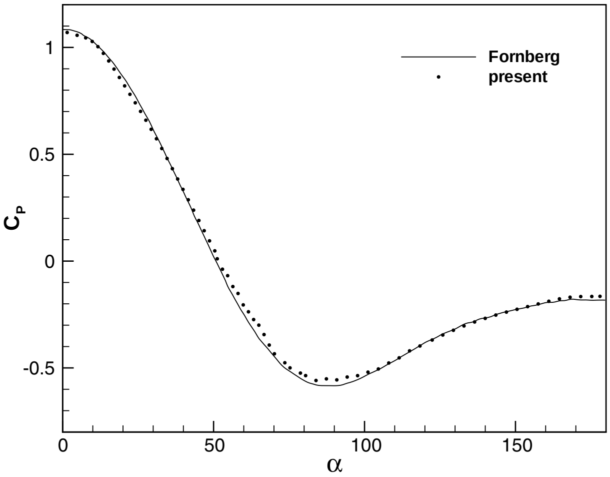


Fig. 9. Pressure coefficient distribution for $Re = 100$.

energy equation, insofar as the temperature fields has a significant influence on the flow pattern, especially when the ratio between the cylinder wall temperature, and the free-stream one, $T^* = T_w/T_\infty$, exceeds 1.1 [27,28]. Moreover, for a given Re, the Strouhal number, $St = fD/U_\infty$, decreases for increasing values of T^* [27].

The computational domain has the inlet and outlet vertical boundary planes located at $x_i = -10D$ and $x_o = 40D$, and the far-field horizontal boundaries located at $y_{ff} = \pm 15D$, the origin of the box coinciding with the centre of the cylinder. Computations have been performed using a centred space discretization and a semi-structured mesh with 41,509 cells and 293,647 faces, based on an auxiliary grid with 796 × 379 cells. The grid is refined at the body surface and in a box surrounding the cylinder and the wake, in order to solve accurately the thermal boundary layer and the vortex shedding. The physical time step has been chosen in order to have about 500 steps per period; about 250 pseudo time steps are needed to reduce the maximum L_2 norm of the residual to 10^{-6} at each physical time level. Fig. 10 shows the computed values of the Strouhal number, St, for $Re = 100, 120, 140$ and $T^* = 1.0, 1.1, 1.5, 1.8$, together with the experimental data provided in [27,28]: a very good agreement is obtained – also with the numerical results obtained with the structured version of the

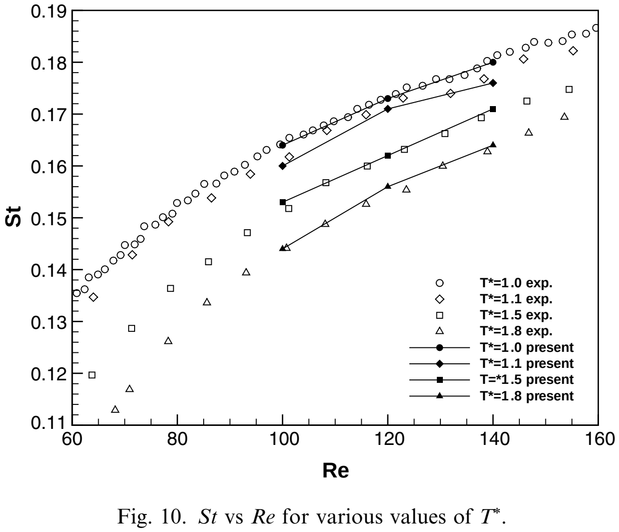
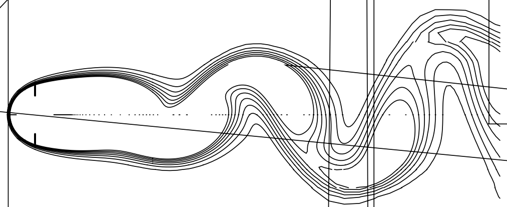


Fig. 10. St vs Re for various values of T^*.



code [6], not shown in the figure. Figs. 11 and 12 show the temperature and density contours for $Re = 100$ and $T^* = 1.8$.

3.4. Flow past an NACA-0012 airfoil

The subsonic flow past an NACA-0012 airfoil at $\alpha = 0$, $M = 0.5$, and $Re = 5000$ has been considered since it experiences a small separation region at the trailing edge of the airfoil. Therefore, it is appropriate for validating numerical methods in general [29] and the accuracy of the present IB one, in particular.

Three grids have been used to discretize the computational domain $[-20c; 21c] \times [-20c; 20c]$, c being the chord-length of the airfoil, whose leading edge is located at the origin. The grids employ 11,275 (grid01), 18,485 (grid02), and 34,350 (grid03) cells and the mesh sizes at the wall are equal to $5 \times 10^{-3}c$, $10^{-3}c$, and $5 \times 10^{-4}c$, respectively. Such grids correspond to three auxiliary grids with 58×334, 679×750, and 1404×1676 cells, respectively. Figs. 13 and 14 provide the computed skin friction coefficient distributions along the airfoil and close to the separation point, respectively. The solutions obtained using the two finest grids are shown to coincide within plotting accuracy and are in good agreement with the numerical results

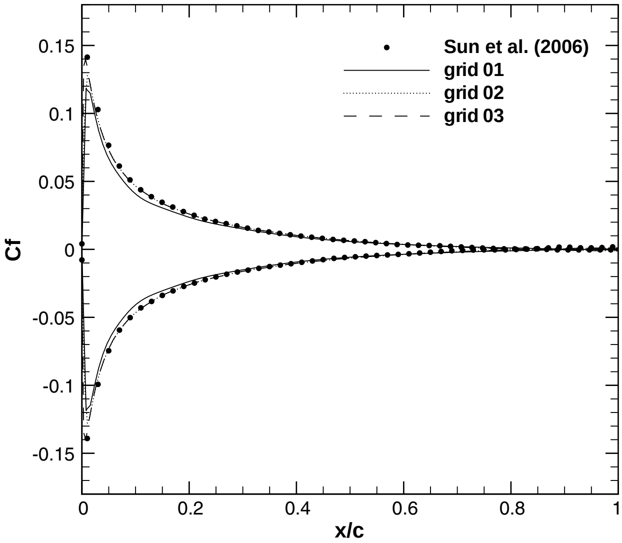


Fig. 13. Skin friction coefficient distribution along the surface.

of [29], also given in the figures. It is indeed remarkable that an accurate prediction of the separation point is obtained by an IB method using only about 18,000 cells.

3.5. Supersonic flow past a circular cylinder

The steady turbulent supersonic flow past a circular cylinder has been considered as a suitable test case to validate the method for compressible flows with shocks. Two cases have been computed with $M = 1.3$ and 1.7, the free-stream Reynolds number and the inlet values of the turbulence kinetic energy and specific dissipation rate being $Re = 2 \times 10^5$, $k/U_\infty^2 = 0.0009$, and $\omega D/U_\infty = 40$, respectively. The free stream flow encounters a bow shock ahead of the cylinder, then it accelerates along the cylinder surface forming a supersonic-flow region which envelops the subsonic recirculation region behind the cylinder and generates two symmetric tail

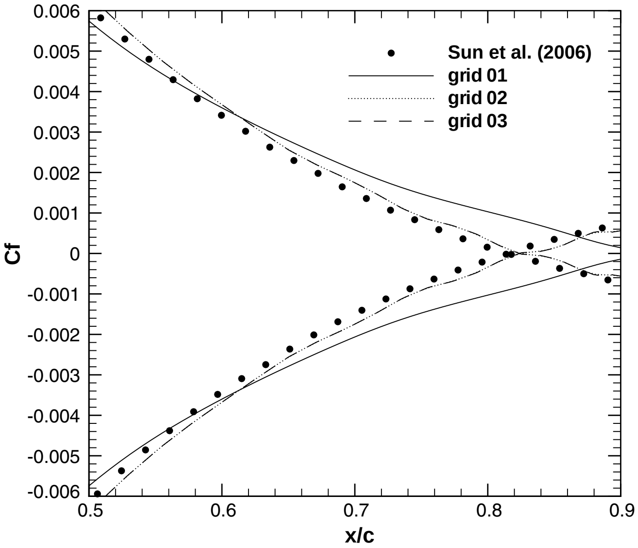
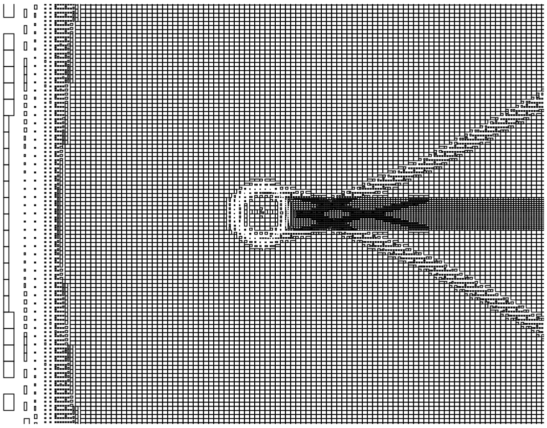
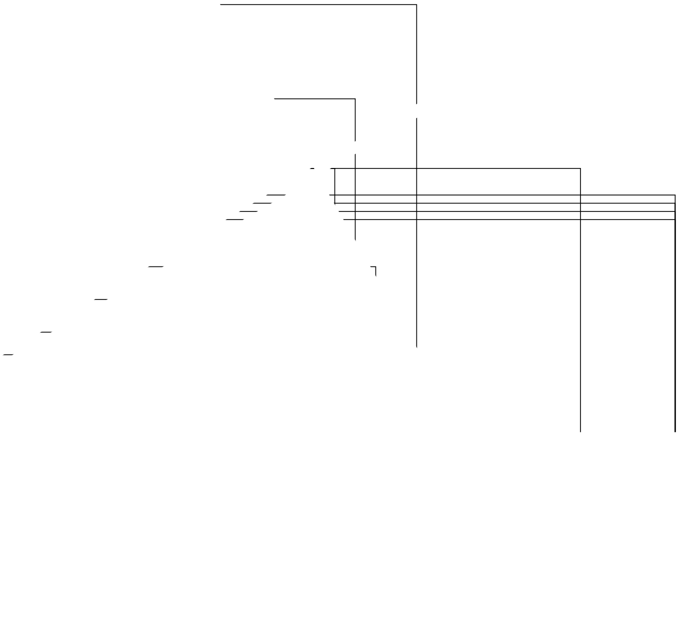


Fig. 14. Local view of the skin friction coefficient distribution near the separation point.



shocks. Results have been obtained using the TVD third-order-accurate upwind scheme and a rectangular computational domain with dimensions $[-10D; 15D] \times [-10D; 10D]$, D being the diameter of the cylinder centred at the origin. Coarse-grid preliminary computations have been performed to locate the shocks, approximately. Then, the coarse meshes have been refined around the shocks and in the wake, to obtained the final semi-structured grids having 107,305 cells and 771,180 faces, corresponding to an auxiliary 1947×2159 mesh, for the case $M = 1.3$, and 75,556 cells and 545,699 faces, corresponding to an auxiliary 1805×2159 mesh, for the case $M = 1.7$. Local views of such grids are shown in Figs. 15 and 16, the Mach number contours are shown in Figs. 17 and 18, where all high-flow-gradient regions appear well resolved, and the computed pressure coefficient distributions along the surface of the cylinder are provided in Figs. 19 and 20, together with the experimental data of [30]. The computed separation angles, measured clockwise from the leading edge, are equal to 105° and 113° for $M = 1.3$ and 1.7, respectively, which agree well with the corresponding experimental data of [30], namely, 103° and 112°. Moreover, the computed and experimental drag coefficients are equal to 1.45, 1.48 for $M = 1.3$ and 1.41, 1.43 for $M = 1.7$. In conclusion, all numerical results agree reasonably well with the experimental data and are in good agreement with the numerical solutions obtained using the same numerical method and a body-fitted grid [6].



[redacted]

3.6. *Transonic flow past an RAE 2822 airfoil*

The turbulent transonic flow past the supercritical RAE 2822-airfoil has been computed as a suitable test case involving shock/boundary layer interaction. The flow-condition case 10 [31] has been considered with: $M = 0.75$, $Re = 6.2 \times 10^6$ (based on the far-field conditions and on the chord length, c), incidence angle $\alpha = 3.19^\circ$. Such a test case is particularly severe since a separation bubble occurs close to the shock location; therefore, the separation length and the position of the shock are highly sensitive to the near-wall flow resolution and to the turbulence modeling. The flow conditions used in the EUROVAL project [32] are employed here to correct the tunnel data for wall-effects, namely: $M = 0.754$, $Re = 6.2 \times 10^6$, $\alpha = 2.57^\circ$. The inlet and outlet boundary planes are located at $x_\mathrm{i} = -20c$ and $x_\mathrm{o} = 30c$, respectively, and the far-field boundaries are located at $y_\mathrm{ff} = \pm 20c$, the origin of the box coinciding with the leading-edge of the airfoil. Computations have been performed using the TVD third-order-accurate upwind scheme and a mesh with 169,312 cells and 1,214,772 faces. The Mach number contours are shown in Fig. 21, whereas the pressure coefficient distribution along the profile is provided in Fig. 22, together with the experimental data: an excellent agreement is obtained.

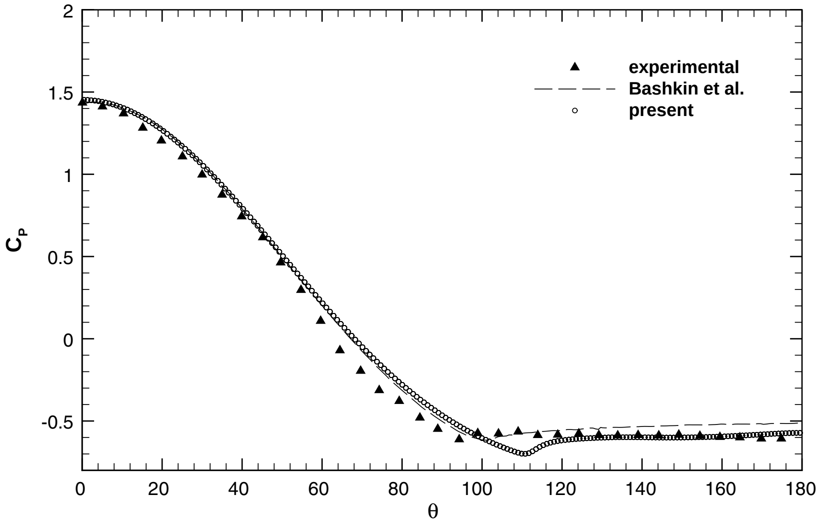


Fig. 19. Pressure coefficient distribution along the surface of the cylinder for $M = 1.3$.

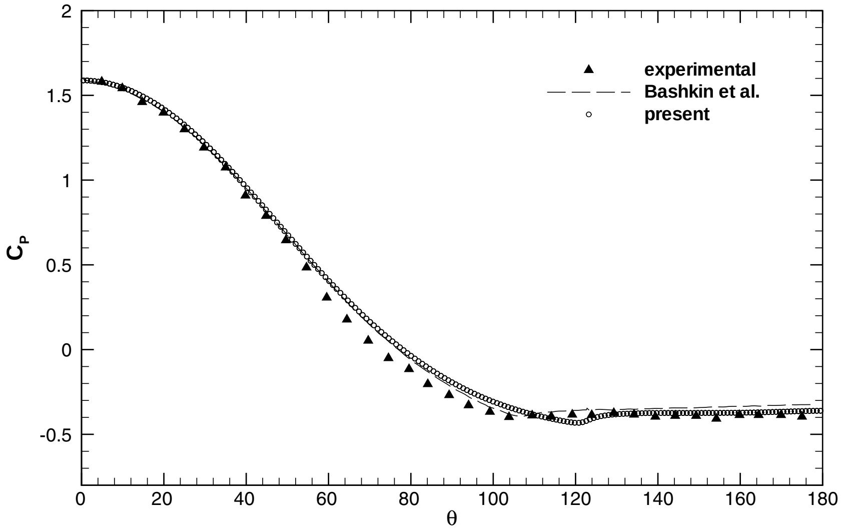


Fig. 20. Pressure coefficient distribution along the surface of the cylinder for $M = 1.7$.

3.7. Mesh-refinement study

Finally, the steady compressible flow past an NACA-0012 airfoil with free-stream conditions $M = 0.8$, $Re = 20$ and angle of attack equal to 10° has been computed to verify the order of accuracy of the method. Five grids have been used to discretize the computational domain $[-10c; 11c] \times [-10c; 10c]$, c being the chord-length of the airfoil, whose leading edge is located at the origin. Starting from the coarsest grid, which is locally refined near the airfoil, each finer grid has been obtained by dividing each cell into four identical sub-cells, thus preserving the topology of the semi-structured grid. The numbers of cells for each grid are reported in Table 3. Richardson's extrapolation has been applied to find the reference *exact* solution. Considering the two finest grids and assuming a second-order-accurate solution for sufficiently small mesh sizes, for any flow variable, ϕ, the extrapolated value ϕ_e has been computed as:

$$\phi_e = \frac{1}{3}(4\phi_5 - \phi_4), \tag{35}$$

where ϕ_4 is the bilinearly interpolated ϕ from grid 4 onto grid 5. Then, the *exact* solution has been interpolated bilinearly to provide the *exact* values at all cell-centres of grids 1–4 needed to evaluate the error norms

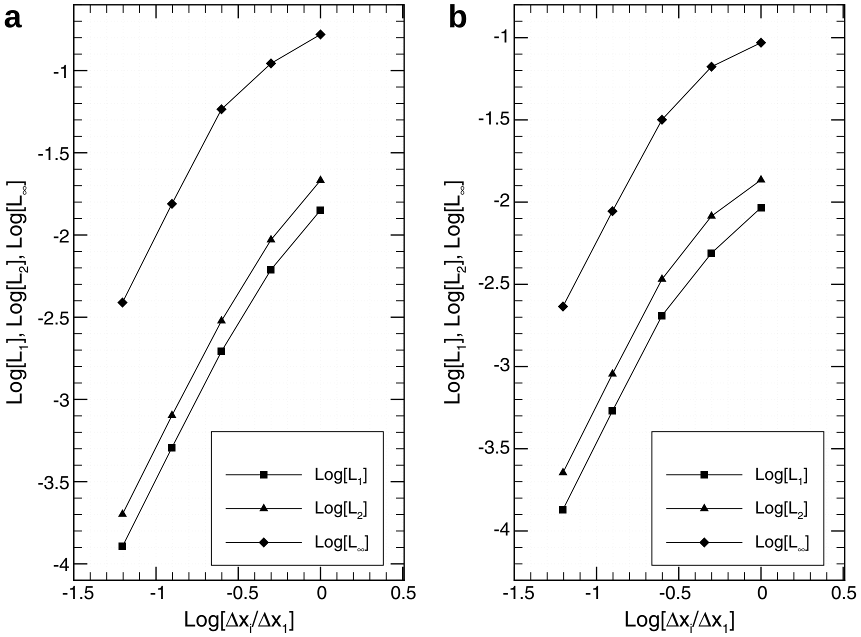


Fig. 23. L_1, L_2, L_∞ norms of the errors considering all of the *fluid* and *interface* cells: (a) horizontal velocity component; (b) vertical velocity component.

for all grids. The L_1, L_2, and L_∞ norms of the errors computed over all *fluid* and *interface* cells for the flow variables u, v, p, and T, non-dimensionalized by the free-stream values, are reported in Figs. 23 and 24. The convergence rates of the errors show second-order accuracy even for the L_∞ norms. Therefore, the forcing

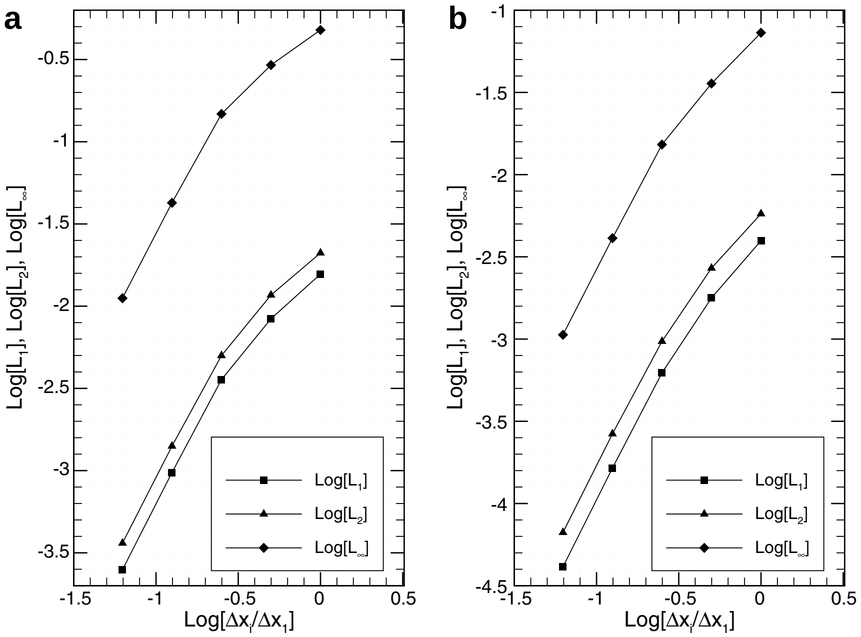


Fig. 24. L_1, L_2, L_∞ norms of the errors considering all of the *fluid* and *interface* cells: (a) pressure; (b) temperature.

procedure based on the linear reconstruction does not spoil the order of accuracy of the space discretization and the present LGR-IB method is indeed second-order-accurate.

4. Conclusions

This paper provides a numerical method for solving the three-dimensional preconditioned compressible Navier–Stokes and Reynolds-averaged Navier–Stokes equations using a Cartesian locally refined semi-structured grid. The proposed immersed boundary method is suitable for computing complex flows of industrial interest in a wide range of the Mach number, thanks to its local grid refinement strategy, which allows one to achieve high resolution near the body and in regions of high flow-gradients, while saving cells in other regions of the computational domain. The method has been validated versus well documented steady and unsteady test problems, in the highly subsonic, transonic, and supersonic flow regimes, demonstrating its versatility as well as its accuracy for moderate values of the Reynolds number. Most importantly, for the laminar compressible steady flow past an NACA-0012 airfoil, the method is shown to be second-order accurate by means of a thorough mesh-refinement study.

Acknowledgments

The authors acknowledge the Center for Turbulence Research (CTR) of Stanford University (USA) where the first author has performed part of his work. This research has been supported by the Ministry of the University and Research and by the Politecnico di Bari (Italy), grant CofinLab 2000.

References

[1] C.S. Peskin, Flow patterns around heart valves: a numerical method, J. Comput. Phys. 10 (1972) 252.
[2] J. Mohd-Yusof, Combined immersed boundaries/B-splines methods for simulations of flows in complex geometries, Technical Report, NASA Ames/Stanford University, CTR Annual Research Briefs, 1997.
[3] E.A. Fadlun, R. Verzicco, P. Orlandi, J. Mohd-Yosuf, Combined immersed-boundary finite-difference methods for three-dimensional complex flow simulations, J. Comput. Phys. 161 (2000) 35.
[4] G. Iaccarino, R. Verzicco, Immersed boundary technique for turbulent flow simulations, Appl. Mech. Rev. 56 (2003) 331.
[5] R. Mittal, G. Iaccarino, Immersed boundary methods, Annu. Rev. Fluid Mech. 37 (2005) 239.
[6] P. De Palma, M.D. de Tullio, G. Pascazio, M. Napolitano, An immersed boundary method for compressible viscous flows, Comput. Fluids 35 (2006) 693.
[7] D.C. Wilcox, Turbulence Models for CFD, second ed., DCW Industries, Inc., 1998.
[8] S. Venkateswaran, S. Weiss, C.L. Merkle, Y.H. Choi, Propulsion related flow fields using the preconditioned Navier–Stokes equations, AIAA Paper 92-3437.
[9] C.L. Merkle, Preconditioning methods for viscous flow calculations, in: M. Hafez, K. Oshima (Eds.), Computational Fluid Dynamics Review 1995, Wiley, New York, 1995, pp. 419–436.
[10] S. Venkateswaran, C.L. Merkle, Dual time stepping and preconditioning for unsteady computations, AIAA Paper 95-0078.
[11] D.A. Schwer, Numerical study of unsteadiness in non-reacting and reacting mixing layers, Ph.D. Thesis, Department of Mechanical Engineering, The Pennsylvania State University, 1999.
[12] T.H. Pulliam, D.S. Chaussee, A diagonal form of an implicit factorization algorithm, J. Comput. Phys. 39 (1981) 347.
[13] H. van der Vorst, Bi-CGSTAB: a fast and smoothly converging variant of Bi-CG for the solution of non-symmetric linear systems, SIAM J. Sci. Statist. Comput. 13 (1992) 361.
[14] J. O'Rourke, Computational Geometry in C, Cambridge University Press, Cambridge, 1998.
[15] P.A. Durbin, G. Iaccarino, Adaptive grid refinement for structured grids, J. Comput. Phys. 128 (2002) 110.
[16] G. Iaccarino, G. Kalitzin, P. Moin, B. Khalighi, Local grid refinement for an immersed boundary RANS solver, Paper AIAA-2004-0586.
[17] M. Berger, M. Aftosmis, Aspects (and aspect ratios) of Cartesian mesh methods, in: C.-H. Bruneau (Ed.), Sixteenth International Conference on Numerical Methods in Fluid Dynamics, Springer, Berlin, 1998, pp. 1–12.
[18] F.E. Ham, F.S. Lien, A.B. Strong, A Cartesian grid method with transient anisotropic adaptation, J. Comput. Phys. 179 (2002) 469.
[19] P.L. Roe, Approximate Riemann solvers, parameter vectors and difference schemes, J. Comput. Phys. 43 (1981) 357.
[20] M. Coutanceau, R. Bouard, Experimental determination of the main features of the viscous flow in the wake of a circular cylinder in uniform translation. Part 1. Steady flow, J. Fluid Mech. 79 (1977) 231.
[21] D.J. Tritton, Experiments on the flow past a circular cylinder at low Reynolds number, J. Fluid Mech. 6 (1959) 547.
[22] B. Fornberg, A numerical study of the steady viscous flow past a circular cylinder, J. Fluid Mech. 98 (1980) 819.

[23] S.C.R. Dennis, G.-Z. Chang, Numerical solutions for steady flow past a circular cylinder at Reynolds number up to 100, J. Fluid Mech. 42 (1970) 471.

[24] M.N. Linnick, H.F. Fasel, A high-order immersed interface method for simulating unsteady incompressible flows on irregular domains, J. Comput. Phys. 204 (2005) 157.

[25] G.K. Batchelor, An Introduction to Fluid Mechanics, Cambridge University Press, Cambridge, 1967.

[26] R. Clift, J.R. Grace, M. Weber, Bubbles, Drops and Particles, Academic Press, New York, 1978.

[27] A.-B. Wang, Z. Trávnîĉek, K.-C. Chia, On the relationship of effective Reynolds number and Strouhal number for the laminar vortex shedding of a heated circular cylinder, Phys. Fluids 12 (2000) 1401.

[28] M. Sabanca, F. Durst, Flows past a tiny circular cylinder at high temperature ratios and slight compressible effects on the vortex shedding, Phys. Fluids 15 (2003) 1821.

[29] Y. Sun, Y.L.Z.J. Wang, Spectral (finite) volume method for conservation laws on unstructured grids VI: Extension to viscous flow, J. Comput. Phys. 215 (2006) 41.

[30] V.A. Bashkin, A.V. Vaganov, I.V. Egorov, D.V. Ivanov, G.A. Ignatova, Comparison of calculated and experimental data on supersonic flow past a circular cylinder, Fluid Dyn. 37 (2002) 473.

[31] P.H. Cook, M.A. McDonald, M.C.P. Firmin, Aerofoil 2822-pressure distributions, boundary layer and wake measurements, AGARD AR 138.

[32] W. Haase, F. Bradsma, E. Elsholz, M. Leschziner, D. Schwamborn, EUROVAL – an European initiative on validation of CFD codes, Notes Numer. Fluid Mech. (1993) 42.