 ICCES, vol.3, no.3, pp.151-156, 2007

An immersed boundary technique using semi-structured grids for computing compressible viscous flows

M.D. de Tullio[1], P. De Palma[1], G. Pascazio[1] and M. Napolitano[1]

Summary

This paper provides a numerical method based on the immersed boundary approach for computing compressible viscous flows. The efficency of the method is enhanced by using a flexible local grid refinement technique which is obtained by coarsening a uniformly fine mesh far from high-gradient flow regions, such as boundary layers and shocks.

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]-[3] using Cartesian grids. Recently, some of the authors have extended the IB technique to compressible flows [4] using the preconditioned Navier–Stokes equations, which allow one to provide accurate and efficient solutions for a wide range of the Mach number. To date, IB methods employ structured grids, which allow only limited control on the distribution of the grid points in the computational domain; in fact, clustering of grid points is needed close to solid boundaries in order to describe its geometry accurately and, since mesh lines run through the entire computational domain, a high concentration of grid points is obtained also in regions away from the solid walls, where flow gradients are usually small. In order to cope with this problem, a flexible local grid refinement technique is to be employed, increasing the mesh resolution near the body.

Governing equations and numerical method

In this work, the Reynolds Averaged Navier–Stokes (RANS) equations, written in terms of Favre mass-averaged quantities, are solved in conjunction with the low-Reynolds number $k-\omega$ turbulence model [5]. Such equations are given in compact form as:

$$\Gamma\frac{\partial Q_v}{\partial \tau}+\frac{\partial Q}{\partial t}+\frac{\partial E}{\partial x}+\frac{\partial F}{\partial y}+\frac{\partial G}{\partial z}-\frac{\partial E_v}{\partial x}-\frac{\partial F_v}{\partial y}-\frac{\partial G_v}{\partial z}=D, \qquad (1)$$

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 for the primitive variable vector Q_v, which is related to Q by a Jacobian matrix, has been added to the left-hand-side of equation (1) in order to use a time marching approach for both steady and unsteady problems. The

[1]DIMeG & CEMeC, Politecnico di Bari, via Re David 200, 70125, Bari, Italy

preconditioning matrix, Γ, proposed in [6] is used to premultiply the pseudo-time derivative in order to enhance the efficiency of the solution procedure at all values of the Mach number. Equation (1) is discretized by an Euler implicit scheme in the pseudo-time and the physical-time derivative is approximated by a second-order-accurate three-point backward difference. The resulting algebraic system is diagonalized according to the procedure of Pulliam and Chaussee [7] and solved by a BiCGStab method [8]. A collocated cell-centred finite volume space discretization is used. The convective terms are discretized using either an upwind flux-difference-splitting scheme with first-, second- or third-order accuracy, or a second-order-accurate centred scheme, for low Reynolds number flows. When computing flows with shocks, a total variation diminishing approach is employed using the minmod limiter function and either a second- or a third-order accurate upwind scheme. The viscous terms are discretized by second-order-accurate centred differences.

Data structure and semi-structured grid generation

Local grid refinement allows for efficient clustering of cells close to the immersed boundary. The basic idea was recently introduced by Durbin and Iaccarino [9] for a finite difference discretization and extended to a finite volume formulation by Iaccarino et al. [10]. The following description of the algorithm refers to two dimensions, the extension to three-dimensions being straightforward. An auxiliary structured grid is employed to handle the data structure of the semi-structured locally refined grid (shown in figure 1). The auxiliary grid covers the whole computational domain employing the finest mesh size on the semi-structured grid. Therefore, each cell of the semi-structured grid is bounded by the lines passing through the vertices (i, j) and $(i+\Delta i, j+\Delta j)$, see figure 1, where the indeces $i = 0,\ldots,N_i$ and $j = 0,\ldots,N_j$ refer to the auxiliary (finest) grid and $\Delta i \geq 1$, $\Delta j \geq 1$ depend on (i, j). Therefore, 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 couples 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 former grid not employed in the latter 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. The total storage required for allocating $ID(i, j)$ is, therefore, $N_i \times N_j$ integers. The connectivity information for each cell is retrieved by querying the array $ID(i, j)$.

Within the IB method, the generation of the semi-structured grid is carried out by firstly creating the auxiliary (fine) grid and coarsening it in the regions away from the immersed boundary. The advantage of this approach is that all of the cell-tagging (ray-tracing) needed for using the IB technique can be performed on a

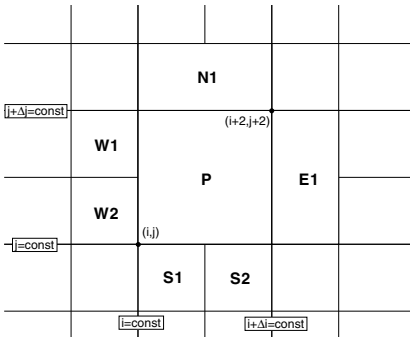

Figure 1: Locally refined grid showing a cell P and its neighbors.

structured grid, taking full advantage of the alignment of the cell centers and of the grid nodes. Moreover, it is possible to define regions of the computational domain to be refined, such as wake and shock regions.

Immersed boundary treatment

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, the computational cells occupied entirely by the flow are tagged as *fluid* cells; those whose centres lie within the immersed body are tagged as *solid* cells; the remaining ones are finally tagged as *interface* cells. The application of the boundary conditions at the immersed surface is treated explicitly, by assigning the values of the variables at the interface cells. At *solid* cells, the flow variable ϕ (e.g., the velocity components and, in the case of isothermal surfaces, the temperature) is set to its wall value. At the interface cells, the nearby wall is modeled with an off-wall boundary condition which consists of an interpolation of the flow variables, using the computed values of the surrounding *fluid* cells and the imposed values at the wall. At each interface cell it is possible to find N_{nbr} neighbouring *fluid* cells and N_{ib} intersections of the faces of each cell with the immersed boundary, and the following interpolation formula is used:

$$\phi_{int} = \sum_{i}^{N_{nbr}} \frac{\alpha_i}{q}\phi_i + \sum_{j}^{N_{ib}} \frac{\beta_j}{q}\phi_{j,wall}, \quad q = \sum_{i}^{N_{nbr}} \alpha_i + \sum_{j}^{N_{ib}} \beta_j, \tag{2}$$

where $\phi_{j,wall}$ 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 between the surrounding cell centers and the interface cell center and between the wall intersections and the interface cell center, respectively. It can be shown that in the one-dimensional case, this procedure coincides with the linear interpolation scheme used in [2] and [4]. The pressure gradient along the normal to the immersed surface is set to zero by assigning the corresponding flow field value at the interface cell. In case of an

adiabatic surface, also the temperature gradient is set to zero.

Results

The unsteady two-dimensional low-Mach-number flow past a heated circular cylinder has been chosen in order to validate both the unsteady terms and the correct implementation of the energy equations, since experimental [11] and numerical [12] investigations indicate that the temperature field has a significant influence on the flow pattern, expecially when the ratio between the cylinder wall temperature T_w and the free-stream one T_∞, $T^* = T_w/T_\infty$ exceeds 1.1. It has been found that, for a given Re_∞, the vortex shedding frequency, f, and thus the Strouhal number $St = fD/U_\infty$, decreases for increasing values of T^*. The computational domain has

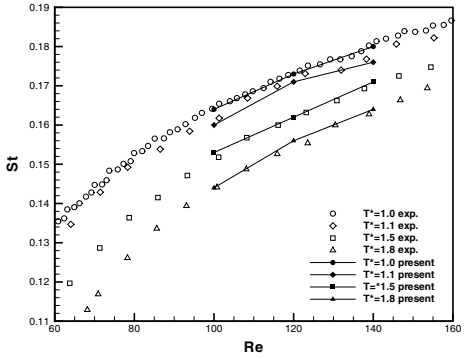

Figure 2: Strouhal number vs Reynolds number for the flow past a heated circular cylinder.

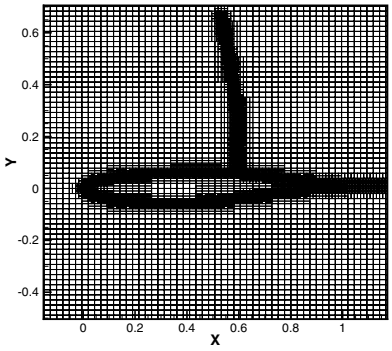

Figure 3: Local view of the semi-structured grid for the flow past the RAE 2822 airfoil.

the inlet and outlet boundaries located at $x_i = -10D$ and $x_o = 40D$, and the far-field boundaries located at $y_w = \pm 15D$, the origin of the box coinciding with the centre of the cylinder. Standard characteristic boundary conditions have been imposed at the inlet and outlet points, whereas free-shear wall boundary conditions are imposed at the far-field points. For such a low Re problem, centred differences are employed also for the advection terms. Computations have been performed using a semi-structured mesh with 41509 cells and 293647 faces. The grid is highly refined at the cylinder surface, in order to solve the thermal boundary layer, and inside a box surrounding the cylinder and the wake, so as to obtain a satisfactory resolution of the vortex shedding phenomenon. The auxiliary mesh is composed of 796×379 cells. The physical time step has been chosen in order to have about 500 steps per period; about 250 inner iterations are needed to reduce the unsteady residual to 10^{-6} at each time step. Figure 2 shows the computed values of the Strouhal number, St, for $Re_\infty = 100, 120, 140$ and $T^* = 1.0, 1.1, 1.5, 1.8$, as well as the experimental results provided by Wang et al. [11] and Sabanca & Durst [12]: a very good agree-

ment is obtained; moreover the present results are comparable with those obtained using the uniform auxiliary grid [4], having seven times more cells. The turbulent

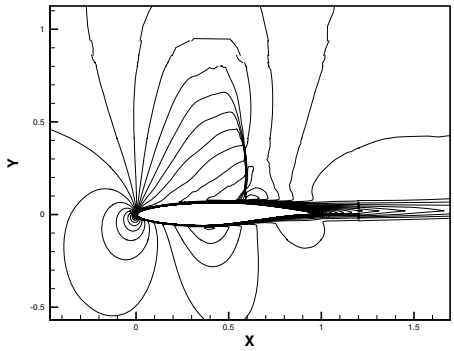

Figure 4: Mach number contours for the flow past the RAE 2822 airfoil;, $\Delta M = 0.05$.

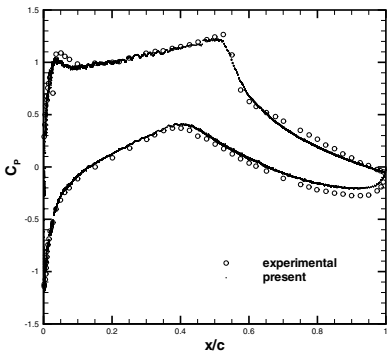

Figure 5: Pressure coefficient distributions for the flow past the RAE 2822 airfoil.

transonic flow past the supercritical RAE 2822-airfoil has finally been computed as a suitable test case involving shock/boundary layer interaction. The flow-condition case 10 [13] has been considered with: $M_\infty = 0.75$, $Re = 6.2 \times 10^6$ (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 flow resolution in the wall region and to the turbulence modeling. To compare the experimental data with the results of the simulation considering an isolated airfoil, corrections to the tunnel data are required. Here, the flow conditions used in the EUROVAL project [14] are employed: $M_\infty = 0.754$, $Re = 6.2 \times 10^6$, $\alpha = 2.57^\circ$. The shock location and the pressure distribution, particularly on the suction side, are influenced by the far-field boundary condition. Therefore, inlet and outlet boundary planes are located at $x_i = -20c$ and $x_o = 30c$, respectively, and the far-field boundaries are located at $y = \pm 20c$, the origin of the box coinciding with the leading-edge of the profile. Standard characteristic boundary conditions have been imposed at the far-field boundary. Numerical results are obtained using the TVD third-order-accurate upwind scheme. Computations have been performed using a mesh with 169312 cells and 1214772 faces (see figure 3). Mach number contours are shown in figure 4, whereas the pressure coefficient distribution along the profile is provided in figure 5 and is in good agreement with the experimental data.

References

1. C. S. Peskin, J. Comput. Phys. 10 (1972) 252.

2. E. A. Fadlun, R. Verzicco, P. Orlandi, J. Mohd-Yosuf, J. Comput. Phys. 161 (2000) 35.
3. R. Mittal, G. Iaccarino, Annu. Rev. Fluid Mech. 37 (2005) 239.
4. P. D. Palma, M. de Tullio, G. Pascazio, M. Napolitano, Comput. Fluids 35 (2006) 693.
5. D. C. Wilcox, Turbulence models for CFD, 2nd Edition, DCW Industries, Inc., 1998.
6. C. L. Merkle, in: M. Hafez, K. Oshima (Eds.), Computational Fluid Dynamics Review 1995, John Wiley & Sons, pp. 419–436 (1995).
7. T. H. Pulliam, D. S. Chaussee, J. Comput. Phys. 39 (1981) 347.
8. H. van der Vorst, SIAM J.Sci.Statist.Comput. 13 (1992) 361.
9. P. A. Durbin, G. Iaccarino, J. Comput. Phys. 128 (2002) 110.
10. G. Iaccarino, G. Kalitzin, P. Moin, B. Khalighi, Paper AIAA-2004-0586.
11. A.-B. Wang, Z. Trávníĉek, K.-C. Chia, Phys. Fluids 12 (2000) 1401.
12. M. Sabanca, F. Durst, Phys. Fluids 15 (2003) 1821.
13. P. H. Cook and M. A. McDonald and M. C. P. Firmin, AGARD AR 138, 1979.
14. W. Haase and F. Bradsma and E. Elsholz and M. Leschziner and D. Schwamborn, Notes on Numerical Fluid Mechanics 42, 1993.