

Simulation of hypersonic rarefied flows with the immersed-boundary method

D. Bruno*, P. De Palma[†] and M. D. de Tullio[†]

**Istituto di Metodologie Inorganiche e dei Plasmi, Consiglio Nazionale delle Ricerche,
via G. Amendola, 122/D, 70125 Bari, Italy*

*[†] Dipartimento di Ingegneria Meccanica e Gestionale & Centro di Eccellenza in Meccanica Computazionale,
Politecnico di Bari,
via Re David 200, 70125 Bari, Italy*

Abstract. This paper provides a validation of an immersed boundary method for computing hypersonic rarefied gas flows. The method is based on the solution of the Navier-Stokes equation and is validated versus numerical results obtained by the DSMC approach. The Navier–Stokes solver employs a flexible local grid refinement technique and is implemented on parallel machines using a domain-decomposition approach. Thanks to the efficient grid generation process, based on the ray-tracing technique, and the use of the METIS software, it is possible to obtain the partitioned grids to be assigned to each processor with a minimal effort by the user. This allows one to by-pass the expensive (in terms of time and human resources) classical generation process of a body fitted grid. First-order slip-velocity boundary conditions are employed and tested for taking into account rarefied gas effects.

Keywords: Immersed Boundary method, separation

PACS: 47.45.Gx: Slip flows and accommodation, 47.40.Nm: Shock wave interactions and shock effects, 47.40.Ki: Supersonic and hypersonic flows, 47.32.Ff: Separated flows, 47.20.Ib: Instability of boundary layers; separation, 47.11.Hj: Boundary element methods

INTRODUCTION

In recent years the immersed boundary (IB) method is emerging as a very appealing approach for solving flows past very complex geometries, like those occurring in most industrial applications. Its main, very significant, feature is the use of a Cartesian grid embodying the complex boundaries of the flow domain, which allows one to generate the computational mesh within a few minutes, whereas a very complicated body fitted grid may require several hours or even days. In the IB approach, the presence of a solid boundary is replaced by suitable forcing conditions modeling the effect of the body on the flow. The IB technique was originally developed for incompressible flows [1, 2, 3, 4, 5] using non-uniform Cartesian grids to take advantage of simple numerical algorithms. Two of the authors have contributed to the extension of the IB method to the preconditioned compressible Navier–Stokes (NS) equations in order to solve complex flows at any value of the Mach number [6], and equipped it with a local mesh refinement procedure to resolve boundary layers and regions with high flow gradients (e.g., shocks) [7]. In this work, the IB method is extended to the solution of flows with rarefaction effects and is validated against DSMC simulations of a shock wave-boundary layer interaction.

NUMERICAL METHODS

Immersed-boundary Navier–Stokes (IBNS) equation solver

The preconditioned Navier–Stokes (NS) equations for a steady laminar flow of a perfect gas are solved [7]. The Prandtl number is equal to 0.71 and a temperature exponent for the viscosity coefficient equal to 0.75 has been employed.

A pseudo-time derivative is added to the NS equations in order to use a time marching approach to compute the steady-state solution. The preconditioning matrix proposed in [8] is used to pre-multiply such a pseudo-time derivative to enhance the efficiency of the solution at all values of the Mach number. The equations are discretized by an implicit Euler scheme in the pseudo-time. The space discretization is based on a cell-centred finite volume approach. The

convective terms are discretized using a second-order-accurate upwind flux-difference-splitting scheme. The viscous terms are discretized by second-order-accurate centred differences. The resulting discrete system is solved direction-by-direction using a BiCGStab [9] approach, the boundary conditions being treated explicitly (see [7] for details).

An efficient local grid refinement technique is employed for clustering cells near the immersed boundary and at other high-flow-gradient regions. For each face, the contributions of the neighbor cells are collected to build the corresponding convective and diffusive operators for the cell, the maximum number of neighbors being limited to two for the present two-dimensional computations [7].

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 polygon in two dimensions (a closed surface in three dimensions), is overlapped onto a Cartesian grid. Using the ray tracing technique based on the geometrical algorithms reported in [10], 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*. Furthermore, the *fluid cells* neighboring *solid* ones are tagged as *interface fluid cells*, and the *solid cells* neighboring *fluid* ones are tagged as *interface solid cells*. Interface cells are used to enforce the boundary conditions between the solid body and the fluid.

In the present implementation, starting from a grid with uniform mesh size, a locally refined grid is generated by recursively halving the mesh size at the immersed boundary region, until an assigned target value is reached [11]. This automatic refinement is based on the following strategy. 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 gradient of this function is different from zero only at the immersed boundary and depends on the local grid size. The components of this gradient in the x and y directions are used to select the cells to be refined. The grid is refined until a user specified resolution is achieved at the boundary. One can define other regions of the computational domain to be refined, selecting the local resolution of the mesh, like the wake or wall regions and, finally, it is possible to refine on *void* surfaces, namely, surfaces without solid or interface points, like bow-shock regions. An anisotropic local grid refinement is allowed, namely, each cell can be refined independently in each Cartesian direction. This feature complicates the grid topology but renders the approach more flexible to handle complex geometries with a remarkable reduction of the memory requirement with respect to a standard OCTREE data structure. Then, employing the software METIS [12], the obtained mesh is divided into a number of blocks defined by the user, balancing the number of computational cells among them. Following a domain decomposition approach, each block is assigned to a CPU which performs the integration of the NS equations in parallel, exchanging the needed information with neighboring processors. Finally, all the necessary information about the grid (coordinates, metrics, pointers for the communications among cells and among processors, etc.) are allocated according to an edge-based data structure and the data are provided in output files to be read by each processor. All these operations are performed automatically and the user needs only to establish the number of processors employed for the computation. The code has been parallelized implementing the communication exchange among the processors based on the MPI protocol.

Concerning the boundary conditions at the solid surfaces, a linear reconstruction is employed near the wall to determine the flow variables at the interface cells imposing the values of the velocity and of the temperature at the solid surface [7]. Slip velocity conditions are computed using the first-order relation provided in [13], which allows one to capture the bulk flow-field away from the walls.

DSMC method

IBNS results have been compared to DSMC simulations. The latter use a standard DSMC implementation [14]. In the present case, a model with continuous internal energy is used. Energy in collision is distributed according to the Larsen-Borgnakke scheme with unitary collision relaxation number in order to simulate the constant- γ behavior. The code implements a two-dimensional geometry with a multi-block, stretched grid approach. Parallel computations, based on a domain decomposition approach and coded with MPI standard have been performed for the hypersonic ramp flow on a 16-cpu Linux cluster.

RESULTS

Slip model validation

As a first step, the slip flow model has been validated in a simple configuration. The gas is a Hard Sphere gas with the following properties:

mass: $m=28.9641$ amu

specific heat ratio: $\gamma = 1.4$

viscosity: $\eta = 5.6853 \cdot 10^{-6} \sqrt{\frac{T}{80}} \text{ Pa} \cdot \text{s}$

Prandtl number: $Pr = 0.71$

The domain is a two-dimensional channel, $4 \cdot 10^{-3} \text{ m}$ wide and periodic in the other direction. The walls are perfectly diffusing with wall temperature set to 150 K . The average density in the channel is $n = 9.13 \cdot 10^{21} \text{ m}^{-3}$. The gas is subject to a volume acceleration in the longitudinal direction equal to $a_x = 5 \cdot 10^6 \text{ m/s}^2$.

The Knudsen number, based on the average density and the channel half-length, is $Kn = 0.05$. The local Knudsen number, instead, based on the local density and on the velocity gradient can reach the value 0.9 near the wall. The flow is highly rarefied, in fact.

For the NS simulations, uniform grids with 32 and 64 mesh points in the normal direction to the boundary provide the same results within plotting accuracy. Computations have been performed in a two-dimensional domain with periodic boundary-conditions in the streamwise direction, slip-velocity boundary conditions at the bottom wall, and symmetry conditions at the top boundary corresponding to mid-channel height.

DSMC simulations have been performed on a simulation domain $8 \cdot 10^{-3} \text{ m}$ long with periodic boundary conditions in the streamwise direction and $2 \cdot 10^{-3} \text{ m}$ wide with a diffuse wall at $y = 0$ and symmetry condition at the upper limit. The simulation space has been discretized in 256×64 square cells $3.125 \cdot 10^{-5} \text{ m}$ in length. The time step is $dt = 5 \cdot 10^{-9} \text{ s}$ and the number of simulated particles is about $3.25 \cdot 10^5$. Time averaging has been performed in order to decrease the statistical error.

The velocity profiles obtained by DSMC and IBNS with and without slip are reported in Fig. 1. In this plot, the position in the channel is divided by the channel half-length and the flow speed is divided by the maximum value at the center of the channel. The first-order slip model therefore allows to accurately reproduce DSMC results in this case even in conditions of strong rarefaction.

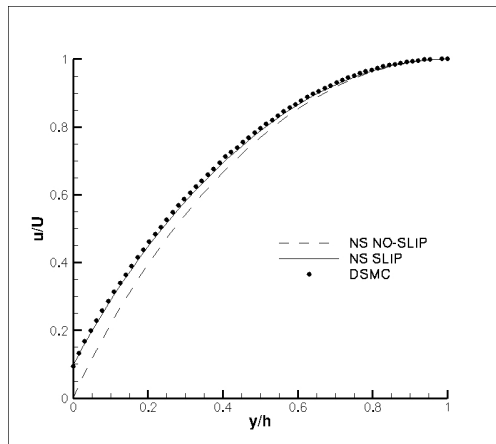


FIGURE 1. Channel flow: velocity profiles.

Hypersonic ramp flow

In order to validate the IBNS approach on a more complex test case, simulations are performed for the hypersonic flow over a compression ramp [15]. The ramp geometry is employed to study the effects of flap deflection on the flow

past a space vehicle. This configuration shows the typical features of shock wave-boundary layer interaction with flow separation and re-attachment.

The distance x_c , between the leading edge of the flat plate and the ramp corner is equal to 71.4 mm . The ramp angle is 35-deg and also its length is equal to 71.4 mm . Further details of the ramp geometry are provided in [15].

The gas is Nitrogen with $\gamma = 1.4$. The Prandtl number is equal to 0.71 and a temperature exponent for the viscosity coefficient equal to 0.75 has been employed. In the present work, steady flow simulations have been performed.

The flow of nitrogen gas is considered with the following free-stream conditions [15]: $\rho_\infty = 1.401 \times 10^{-4} \text{ kg/m}^3$, $V_\infty = 1521 \text{ m/s}$, $T_\infty = 9.06 \text{ K}$. The corresponding free-stream Mach number and Reynolds number are 24.8 and $12,020$, respectively. The wall is perfectly diffusing with wall temperature set to 403.2 K .

Two-dimensional IBNS computations have been performed using a locally refined grid with about 80000 cells clustered at the leading edge of the plate and in the recirculation region. The height of the first cell along the wall is 0.02 mm . About 20 minutes wall-clock CPU time are needed using 16 cores of a Linux cluster with $8\text{-core Intel Xeon @}2.80\text{Ghz}$ processors to obtain the numerical solution, including the grid generation process.

DSMC simulations used 16 cpus on a Linux cluster. The mesh is made of 5 blocks with 1.7 million cells. The timestep is $dt = 10^{-8} \text{ s}$ and the number of simulated particles is 22 millions.

Figure 2 shows the pressure and skin friction coefficients as obtained by DSMC and by IBNS method with slip conditions. Moss's results [15] are also reported for comparison. The IBNS results have been smoothed to avoid the typical small oscillations of the data at wall due to the linear reconstruction near the boundary. Concerning the DSMC results, a small residual statistical scatter is still visible after the time averaging procedure.

At the leading edge, the strong velocity gradient that induces nonlinear transport terms in the stress tensor is the likely cause of discrepancy between the particle methods on one side and the Navier-Stokes equation solver on the other.

Apart from this, the results compare favorably. A better agreement is found when using the slip conditions than when the standard no-slip conditions are considered (not shown in the figures). The skin friction plot shows that the separation zone is overestimated by the IBNS method. As a result, the flow velocity following re-attachment is underestimated. One possible reason for the discrepancy might be the use of slip coefficients for the Hard Sphere gas. Although, on general grounds, first-order slip coefficients are considered insensitive to the gas model [13], in the present case the position and the extent of the separation zone are found to be very sensitive to the slip coefficients. Further investigations are in progress in order to clarify this point.

Figures 3 and 4 provide the non-dimensional density contours, ρ/ρ_∞ , obtained by the IBNS and DSMC methods, respectively. The peak value of density is found about in the same region and is equal to 12.4 for the IBNS approach and to 14.3 for the DSMC, whereas the value reported in [15] is 14.8 . Finally, figures 5 and 6 report the streamlines (superposed to the density contours) obtained by the IBNS and DSMC methods, respectively. The agreement is satisfactory both qualitatively and quantitatively apart from the small discrepancy in the separation zone already discussed.

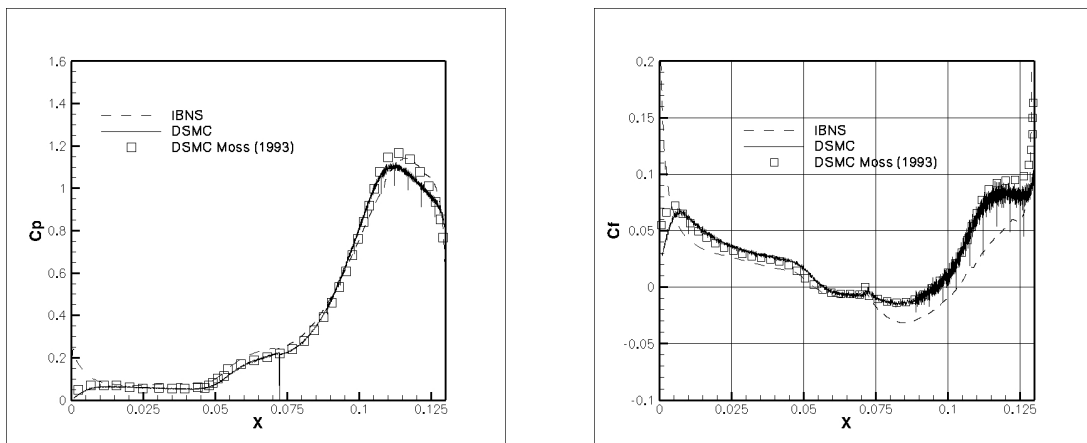


FIGURE 2. Hypersonic ramp flow: pressure coefficient (left); skin-friction coefficient (right).

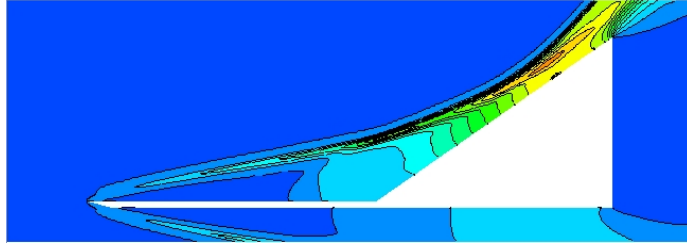


FIGURE 3. Hypersonic ramp flow; Navier–Stokes equations: density contours (ρ/ρ_∞ ; $\Delta = 1$).

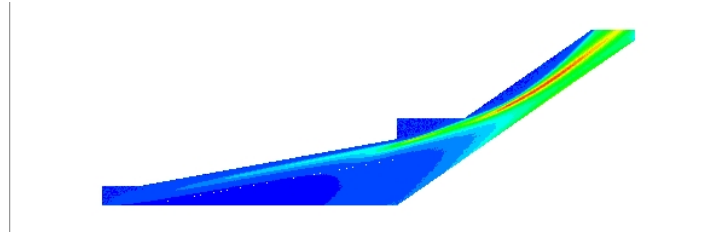


FIGURE 4. Hypersonic ramp flow; DSMC: density contours (ρ/ρ_∞ ; $\Delta = 1$).

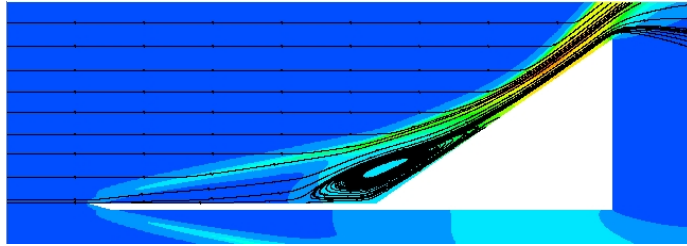


FIGURE 5. Hypersonic ramp flow; Navier–Stokes equations: streamlines.

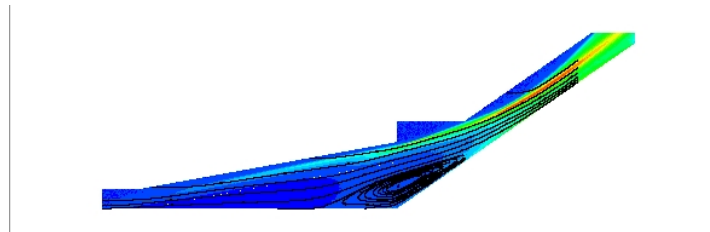


FIGURE 6. Hypersonic ramp flow; DSMC: streamlines.

CONCLUSIONS

This paper provides a validation of a very efficient immersed boundary method based on the solution of the Navier–Stokes equations for computing hypersonic rarefied gas flows. First-order slip-velocity boundary conditions are employed and tested for taking into account rarefied gas effects. The method employs a flexible local grid refinement technique and is implemented on parallel machines using a domain-decomposition approach. The comparison with the results obtained with DSMC simulations is satisfactory. Some discrepancies are found in the prediction of the separation bubble extension which need further investigations to be fully explained.

ACKNOWLEDGMENTS

This work has been supported by the MIUR and the Politecnico di Bari, Cofinlab 2000 and by PRIN-2007 grants.

REFERENCES

1. C. S. Peskin, Flow patterns around heart valves: A numerical method, *J. Comput. Phys.*, **10**, 252, (1972).
2. J. Mohd-Yusof *Combined Immersed Boundaries/B-Splines Methods for Simulations of Flows in Complex Geometries*. CTR Annual Research Briefs, NASA Ames / Stanford University (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**, 35 (2000).
4. G. Iaccarino, R. Verzicco, Immersed Boundary Technique for Turbulent Flow Simulations, *Appl. Mech. Rev.*, **56**, 331-347, 2003.
5. R. Mittal, G. Iaccarino, Immersed Boundary Methods. *Annu. Rev. Fluid Mech.*, **37**, 239 (2005).
6. P. De Palma, M. D. de Tullio, G. Pascazio, M. Napolitano, An immersed boundary method for compressible viscous flows, *Comput. Fluids*, **35**, 693, (2006).
7. M. de Tullio, P. De Palma, G. Iaccarino, G. Pascazio, M. Napolitano, An immersed boundary method for compressible flows using local grid refinement, *J. Comput. Phys.*, **225**, 2098-2117, (2007).
8. C. L. Merkle, Preconditioning methods for viscous flow calculations, in *Computational Fluid Dynamics Review 1995*, M. Hafez, K. Oshima eds., Wiley, New York, 419, 1995.
9. 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**, 361 (1992).
10. J. O'Rourke, *Computational Geometry in C*, Cambridge University Press, Cambridge (1998).
11. S. Kang, G. Iaccarino, F. Ham, DNS of buoyancy-dominated turbulent flows on a bluff body using the immersed boundary method, *J. Comput. Phys.*, **228**, 3189 (2009).
12. G. Karypis, V. Kumar, A fast and high quality multilevel scheme for partitioning irregular graphs, *SIAM J. Sci. Comput.*, **20**, 359 (1998).
13. N. G. Hadjiconstantinou, The limits of Navier-Stokes theory and kinetic extensions for describing small-scale gaseous hydrodynamics, *Physics of Fluids*, **18**, 111301, (2006).
14. D. Bruno, M. Capitelli, F. Esposito, S. Longo, P. Minelli, Direct simulation of non-equilibrium kinetics under shock conditions in nitrogen, *Chem. Phys. Lett.*, **360**, 31, (2002).
15. J. N. Moss, D. F. G. Rault, J. M. Price, Direct Monte Carlo simulations of hypersonic viscous interactions including separation, in *Rarefied Gas Dynamics: Space Science and Engineering*, edited by B. D. Shizgal and D. P. Weaver, **160** of Progress in Aeronautics and Astronautics, AIAA, Washington, 1993, pp. 209-220.