A pressure-corrected Immersed Boundary Method for the numerical simulation of compressible flows

H. Riahi, Marcello Meldi, Julien Favier, Eric Serre, Eric Goncalves da Silva

To cite this version:

H. Riahi, Marcello Meldi, Julien Favier, Eric Serre, Eric Goncalves da Silva. A pressure-corrected Immersed Boundary Method for the numerical simulation of compressible flows. Journal of Computational Physics, 2018, 374, pp.361-383. 10.1016/j.jcp.2018.07.033 . hal-01859760

HAL Id: hal-01859760
https://hal.science/hal-01859760v1

Submitted on 22 Aug 2018

A pressure-corrected Immersed Boundary Method for the numerical simulation of compressible flows

H. Riahi[a], M. Meldi*[a,b], J. Favier[b], E. Serre[b], E. Goncalves[a]

[a] *Institut Pprime, Department of Fluid Flow, Heat Transfer and Combustion, CNRS - ENSMA - Université de Poitiers, UPR 3346, SP2MI - Téléport, 211 Bd. Marie et Pierre Curie, B.P. 30179 F86962 Futuroscope Chasseneuil Cedex, France*
[b] *Aix-Marseille Univ., CNRS, Centrale Marseille, M2P2 Marseille, France*

Abstract

The development of an improved new IBM method is proposed in the present article. This method roots in efficient proposals developed for the simulation of incompressible flows, and it is expanded for compressible configurations. The main feature of this model is the integration of a pressure-based correction of the IBM forcing which is analytically derived from the set of dynamic equations. The resulting IBM method has been integrated in various flow solvers available in the CFD platform *OpenFOAM*. A rigorous validation has been performed considering different test cases of increasing complexity. The results have been compared with a large number of references available in the literature of experimental and numerical nature. This analysis highlights numerous favorable characteristics of the IBM method, such as precision, flexibility and computational cost efficiency.

Keywords: IBM, compressible flows, *OpenFOAM*

---

1. Introduction

Recent technological progress for aerospace engineering but also ground transportation with magnetic levitation trains (Maglev) promises to reduce the travel time with always increasing speed of the vehicles. Under this perspective,

---

*Corresponding author, *marcello.meldi@ensma.fr*

transport engineering advances are more and more related with compressible flow configurations.

The accurate simulation of the flow evolution around immersed bodies is arguably one of the most challenging open issues in transport engineering applications. Success in the flow state prediction allows for precise estimation of the aerodynamic forces acting on the vehicle, which provides fundamental insight for shape optimization. Gains in drag reduction of the order of percentage points will result in significantly reduced fuel consumption [1], and they will allow to remove barriers for consistent green energy usage in the coming years, in agreement with recent European laws for environment [2]. Additionally, a precise flow estimation is necessary to estimate other aspects such as the acoustic field produced, which may result in improved features of comfort and safety for the passengers. However, the state-of-the-art in numerical simulation still needs important development to become an efficient tool for advanced transport engineering applications. Two main critical issues must be challenged:

1. **The mesh representation of complex geometric shapes**. The representation of fine geometric features in classical body-fitted simulations may result in overly deformed / stretched elements, and unfavorable characteristics of the mesh quality. This problematic aspect may lead to poor predictive results.
2. **Moving immersed bodies**. Even simple prescribed movement laws for the immersed body may require several computational mesh updates during the numerical simulation. These updates entail prohibitive computational costs.

Among the numerous strategies proposed in the literature to overcome these critical issues, the *Immersed Boundary Method* (IBM) [3, 4] is an established high-performance tool for the analysis of flow configurations around complex moving bodies. The characteristic feature of the IBM is the representation of the body surface via a volume source effect which is integrated in the chosen mathematical set of equations. Thus, the computational mesh does not need any manipulation in the proximity of the body surface to conform to it. This implies that negative predictive effects such as mesh element deformation can be naturally excluded. In addition, body motion can be imposed or determined without any mesh recalculation. The way these effects are integrated within the numerical simulation may vary significantly, depending on the strategy employed. The IBM methods include a large spectrum of tools which operate using completely different procedures such as fictitious domain approaches [3], level-set methods [5], Lagrangian multipliers [6] and volume penalization [7]. Depending on the implementation strategy employed to determine the level of volume forcing representing the body surface, the IBM approaches reported in the literature are usually classified in two large families, namely the *continuous* methods and the *discrete* methods. The principal difference in the application depends on whether the IBM force is integrated in the continuous or discretized Navier–Stokes equations. The pioneering work proposed by Peskin [3, 8] is the first continuous forcing method reported in the literature. The flow evolution is investigated using an Eulerian system of coordinates whereas the immersed body is represented on a Lagrangian system. In these methods, markers define the immersed solid boundaries. Interpolation between the two grids is obtained via approximations of the δ delta distribution by smoother functions. Following this work, other strategies have been investigated. One notable example is the feedback forcing method, which relies on driving the boundary velocity to rest [9, 10]. Because of the integration of the IBM forcing in the continuous Navier–Stokes equations, the continuous methods are not sensitive to the numerical discretization. However, calibration of the free constants in their formulation is needed. In addition, they exhibit spurious oscillations and severe CFL restrictions, which are associated with the choice of stiffness constants [4]. The direct forcing method, usually referred to as the discrete approach, provides solutions to the drawbacks of the continuous forcing approach. In fact, the introduction of the force term at the discretization stage provides more stable and efficient algorithms [4]. These strategies, which were first investigated by Mohd-Yusof [11], have been further developed in following original research works [12, 13, 14, 15].

The main drawback of these methods is that they exhibit a natural sensitivity to the numerical discretization, especially for the time derivative for unstationary flow configurations.

In the present work, a discrete IBM method proposed for the analysis of incompressible flows on curvilinear grids [16, 17, 18] is extended for the analysis of compressible configurations. As previously discussed, these flows are a timely subject of investigation because of their relevance in environmental [19] / industrial [20] studies. To this aim, a pressure-based correction of the method is introduced, which dramatically improves the numerical prediction of the flow features. The IBM method developed is assessed via analysis of test cases exhibiting increasing complexity. In particular, the flow around a sphere is extensively investigated. This test case represents a classical choice for studies in aerodynamics around 3D bluff bodies, because a number of realistic features observed in flows around complex geometries can be here investigated with reduced computational resources. In addition, moderate Reynolds number configurations exhibit the emergence of different regimes for subsonic Ma values, which are extremely sensitive to fine features of the numerical representation.

The article is structured as follows. In Section 2 the mathematical and numerical background, including the analytic derivation of the new IBM method, is introduced and discussed. In Section 3 the practical implementation in the flow solvers considered is detailed. In Section 4 the IBM method is validated via analysis of classical two-dimensional test cases, encompassing a large range of Ma values. In Section 5 the flow around a sphere is analyzed. In Section 6 the analysis is extended to a sphere subjected to rotation. Finally, in Section 7 the concluding remarks are drawn.

2. Numerical ingredients and Immersed Boundary Method

In this section analytic and numerical details of the IBM algorithm are provided.

2.1. Governing equations

The general Navier–Stokes equations for a compressible fluid write:

$$\frac{\partial \rho}{\partial t} + div(\rho \mathbf{u}) = 0 \tag{1}$$

$$\frac{\partial \rho \mathbf{u}}{\partial t} + \mathbf{div}(\rho \mathbf{u} \otimes \mathbf{u}) = -\mathbf{grad} p + \mathbf{div} \overline{\overline{\tau}} + \mathbf{f} \tag{2}$$

$$\frac{\partial \rho e_t}{\partial t} + div(\rho\, e_t\, \mathbf{u}) = -div(p\, \mathbf{u}) + div(\overline{\overline{\tau}} \mathbf{u}) + div(\lambda(\theta) \mathbf{grad} \theta) + \mathbf{f} \cdot \mathbf{u} \tag{3}$$

where ρ is the density, p is the pressure, $\mathbf{u}$ is the velocity, $\overline{\overline{\tau}}$ is the tensor of the viscous constraints, e_t is the total energy, λ is the thermal conductivity, θ is the temperature and $\mathbf{f}$ is a general volume force term. For Newtonian fluids, the tensor $\overline{\overline{\tau}}$ becomes :

$$\overline{\overline{\tau}} = \mu(\theta) \left((\overline{grad} \mathbf{u} +^{t} \overline{grad} \mathbf{u}) - \frac{2}{3} div(\mathbf{u}) \right) \tag{4}$$

where μ is the dynamic viscosity. It is here calculated using the Sutherland's law as function of temperature θ. The total energy e_t is defined as:

$$e_t = e + \frac{1}{2} \rho \mathbf{u} \cdot \mathbf{u} = C_v \theta + \frac{1}{2} \rho \mathbf{u} \cdot \mathbf{u} \tag{5}$$

where e is the internal energy and C_v is the heat capacity at constant volume. This system is closed by the perfect gas equation of state $p = \rho r \theta$ where r is the specific gas constant.

2.2. Immersed Boundary Method for compressible flows: numerical formulation

The method here presented roots in previous works proposed by Uhlmann [16] and Pinelli et al. [17] which combine strengths of classical continuous forcing methods [3] and discrete forcing methods [21, 11]. In this framework, the numerical results obtained in the Eulerian mesh elements x_s are modified via a body force, which is calculated in a Lagrangian frame of reference defined by markers X_s. These Lagrangian markers describe a discretized shape for the immersed body. We will refer to physical quantities in the Lagrangian space using capital letters (or via the subscript L for Greek letters), while low case letters will be used for the Eulerian description.

2.2.1. Communication between the Eulerian and Lagrangian systems

Communication between the two frames of reference is performed via two steps, namely:

- the *interpolation*, where physical quantities in the Eulerian mesh are interpolated on the Lagrangian markers, in order to estimate the volume force
- the *spreading*, where the volume force previously calculated on the Lagrangian markers is spread back on the Eulerian mesh elements

Physical quantities in the two domains are communicated via interpolation, using δ functions originally proposed by Peskin. The case is now exemplified for the physical quantity $\rho\mathbf{u}$ available on the Eulerian mesh. The corresponding quantity $\rho_L\mathbf{U}$ on the s^{th} Lagrangian marker is determined via the interpolation operator $\mathcal{I}$ as:

$$\mathcal{I}[\rho\mathbf{u}]_{X_s} = [\rho_L\mathbf{U}](X_s) = \sum_{j\in D_s} (\rho\mathbf{u})_j^n \delta_h(\mathbf{x}_j - \mathbf{X}_s)\Delta\mathbf{x} \tag{6}$$

where D_s represents the set of points of the Eulerian mesh. $\Delta\mathbf{x}$ formally refers to an Eulerian quadrature, i.e. $\Delta\mathbf{x} = \Delta x\Delta y\Delta z$ for the case of a Cartesian uniform mesh. The interpolation kernel δ_h is the discretized delta function used in [17] :

$$\delta_h(r) = \begin{cases} \dfrac{1}{3}\left(1+\sqrt{-3r^2+1}\right) & 0 \le r \le 0.5 \\ \dfrac{1}{6}\left[5-3r-\sqrt{-3(1-r)^2+1}\right] & 0.5 \le r \le 1.5 \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

It is centered on each Lagrangian marker X_s and takes non-zero values inside a finite domain D_s, called the support of the s^{th} Lagrangian marker.

The backward communication from the Lagrangian markers to the Eulerian mesh is also performed using delta functions. This is done in the *spreading* step, where the value of the forcing F is distributed over the surrounding mesh. The value of the forcing term evaluated on the Eulerian mesh, $\mathbf{f}(\mathbf{x}_j)$, is given by:

$$\mathbf{f}(\mathbf{x}_j) = \sum_{k \in D_j} \mathbf{F}_k \delta_h(\mathbf{x}_j - \mathbf{X}_k)\boldsymbol{\epsilon}_k \tag{8}$$

The k-index refers to a loop over the Lagrangian markers whose support contains the Eulerian node j. $\boldsymbol{\epsilon}_k$ is the Lagrangian quadrature, which is calculated by solving a linear system to satisfy a partition of unity condition. As in [17] we have:

$$A\boldsymbol{\epsilon} = \mathbf{1} \tag{9}$$

where the vectors $\boldsymbol{\epsilon} = (\epsilon_1, \ldots, \epsilon_{N_s})^T$ and $\mathbf{1} = (1, \ldots, 1)^T$ have a dimension of N_s, N_s being the number of Lagrangian markers. A is the matrix defined by the product between the k^{th} and the l^{th} interpolation kernels such that:

$$A_{kl} = \sum_{j \in D_l} \delta_h(\mathbf{x}_j - \mathbf{X}_k)\delta_h(\mathbf{x}_j - \mathbf{X}_l) \tag{10}$$

2.2.2. Analytic form of the IBM forcing

The novelty of the present approach is represented by i) the extension to compressible flow configurations and ii) the addition of a numerical term which penalizes deviation from the expected behavior of the pressure gradient close to the surface of the body. In numerical simulation, the most classical choice of boundary condition for the pressure field is a homogeneous Neumann condition i.e. zero gradient in the wall normal direction [22]. The present investigation encompasses exclusively this basic condition, which is implemented in most of available open source CFD software. However, the proposed algorithm will allow for the implementation of more sophisticated and precise pressure boundary conditions [23] in future works. This could provide a significant improvement for the IBM, which is usually considered less precise in the resolution of near wall features when compared with body-fitted approaches. In this case, additional information in the form of a wall normal vector $\vec{e}_{ns}$ must be provided for each Lagrangian marker X_s.

Let us consider a general discretized form of the momentum equation 2 in the Eulerian frame of reference, represented by the mesh elements x_s. The time advancement between the time steps n and $n+1$ is considered:

$$a_s\left(\rho\mathbf{u}\right)^{n+1} = \phi^{n+1/2} - \mathbf{grad}p^{n+1/2} + \mathbf{f}^{n+1/2} \tag{11}$$

where a_s represents a discretization coefficient which is equal to $a_s = 1/\Delta t$ if an Euler discretization scheme for the time derivative is employed. The three right hand terms are calculated at an intermediate time $n+1/2$ [16]. In particular, the discretized term ϕ includes the convective and viscous terms, as well as the part of the discretization of the time derivative related with $(\rho\mathbf{u})^n$. So, if we indicate with the affix (d) the expected value of the solution at the instant $n+1$, the optimal value of the forcing in the Eulerian system is:

$$\mathbf{f}^{n+1/2} = a_s\left(\rho\mathbf{u}\right)^{d} - \phi^{n+1/2} + \mathbf{grad}p^{n+1/2} \tag{12}$$

In the frame of incompressible flows, Uhlmann [16] showed that the sum of the last two terms in the right part of equation 12 corresponds to the Eulerian solution $-a_s\left(\rho\mathbf{u}\right)^{n+1}$ at the time $n+1$ considering $\mathbf{f} = 0$. Following the work by Uhlmann, we now shift to the Lagrangian system via *interpolation*. Details are going to be provided in Section 2.2.1. Assuming that a_s is unchanged in the interpolation step (which is exactly true if a_s is a function of the time t only) and indicating in capital letters the physical quantities in the Lagrangian space, equation 12 is transformed in:

$$\mathbf{F}^{n+1/2} = a_s\left(\rho_L\mathbf{U}\right)^{d} - \mathbf{\Phi}^{n+1/2} + \mathbf{grad}P^{n+1/2} \tag{13}$$

where ρ_L is the density field interpolated into the Lagrangian space. We now project the term $\mathbf{grad}P^{n+1/2}$ in equation 13 in the direction of the Lagrangian wall normal $\vec{e}_{ns}$, obtaining

$$\mathbf{grad}P^{n+1/2} = \mathbf{grad}P^{n+1/2} \cdot \vec{e}_{ns} + \mathbf{grad}P^{n+1/2} \cdot \vec{e}_{ts}$$

$\vec{e}_{ts}$ represents the direction of the interpolated pressure gradient in the plane normal to $\vec{e}_{ns}$. In addition, the term $\mathbf{grad}P^{d} \cdot \vec{e}_{ns} = 0$ is included to the right hand of equation 13. This term represents the expected behavior (superscript

d) of the pressure field, which supposedly exhibits a zero-gradient condition in proximity of a wall. Equation 13 is then recast as:

$$\mathbf{F}^{n+1/2} = a_s \left(\rho_L \mathbf{U}\right)^d - \mathbf{\Phi}^{n+1/2} + \mathbf{grad} P^{n+1/2} \cdot \vec{e}_{ts} - \left(\mathbf{grad} P^d - \mathbf{grad} P^{n+1/2}\right) \cdot \vec{e}_{ns} \tag{14}$$

The term $-\mathbf{\Phi}^{n+1/2} + \mathbf{grad} P^{n+1/2} \cdot \vec{e}_{ts} = -a_s \overline{\left(\rho_L \mathbf{U}\right)}$ is a realistic estimation of a first time advancement of the flow field from n to $n+1$ using the momentum equation only, where the pressure gradient is evaluated using data available at the instant n. On the other hand, the term $\left(\mathbf{grad} P^d - \mathbf{grad} P^{n+1/2}\right) \cdot \vec{e}_{ns}$ measures the deviation from the expected behavior of the pressure gradient following this time advancement. Thus the total forcing in the Lagrangian system can be written as:

$$\mathbf{F}^{n+1/2} = a_s \left(\left(\rho_L \mathbf{U}\right)^d - \overline{\left(\rho_L \mathbf{U}\right)}\right) - \left(\mathbf{grad} P^d - \mathbf{grad} P^{n+1/2}\right) \cdot \vec{e}_{ns} \tag{15}$$

This more elaborated structure of the forcing F exhibits a number of interesting aspects:

1. it naturally fits segregated solvers, where the flow variables are not simultaneously resolved and they can be obtained via corrective loops. The new proposals exploits this feature via the separation of the pressure contribution and thus it is supposed to be efficient over a larger spectrum of CFD algorithms;
2. the calculation of the terms $\overline{\left(\rho_L \mathbf{U}\right)}$ and $\mathbf{grad} P^{n+1/2}$ is integrated within the classical formulation of the solver considered, and a full time step without the addition of the forcing is not required anymore [16]. This implies a significant reduction in the computational costs associated with the determination of the Lagrangian forcing F;
3. using this strategy, the behavior of the pressure field is guided towards an expected zero-gradient condition in the wall normal direction. This result is not granted by the classical integration of the forcing as in [16, 17] and

it is essential to capture important features of the flow, as shown in the following.

3. IBM implementation in OpenFOAM numerical solvers

The analytic development described in Section 2.2.2 suggests how the present formulation of the Lagrangian forcing F may be suitable for integration in a large spectrum of algorithmic architectures for fluid mechanics studies. This feature is extremely relevant for the simulation of compressible flows, where different resolution approaches must be employed depending on the values of the Ma number investigated. Thus, in order to validate this important feature of the proposed method, the implementation of the IBM model has been performed in the open source library OpenFOAM. With the target to be used further to investigate industrial configurations, this code provides an efficient coding and a suitable environment for the implementation of new algorithms. It has been identified as a convenient and efficient numerical platform because of the simplicity in implementation as well as the availability of numerous routines already integrated, including IBM for incompressible flows [18]. Two solvers available in the standard version of the code, which allow for the investigation over a very large range of Ma numbers, are considered in the present investigation:

- the segregated pressure-based solver with PIMPLE loop for compressible flows for low Mach numbers ($Ma \leq 0.3$) [24], namely *sonicFoam*.
- the segregated density-based solver with Kurganov and Tadmor divergence scheme for compressible flows for high Mach numbers ($Ma > 0.3$)[24], namely *rhoCentralFoam*.

Details about the algorithmic structure of *sonicFoam* and *rhoCentralFoam* are provided in the Appendix A. Core differences are observed in the practical resolution of the equations. These differences stem from ad-hoc strategies developed with respect to the envisioned range of application of Ma values. It will be shown in the following how the IBM method here developed naturally integrates within the structure of the two codes, exhibiting a very high level of flexibility. The integration of this new IBM strategy follows recent work by Constant et al. [18] dedicated to incompressible flows. The newly generated solvers will be referred to in the following as IBM versions of the initial solver modified and are now presented. A grid convergence analysis of the method is provided in the Appendix B.

3.1. IBM-sonicFoam

The structure of the code is very similar to the scheme presented in Appendix A. The algorithm goes through the following steps:

1. The discretized continuity and momentum equations A.1 - A.2 are resolved, providing a first time advancement of $\rho^\star$, $\mathbf{u}^\star$.
2. A first estimation of the updated pressure field $p^\star$ is obtained via equation A.6.
3. The fields calculated in steps 1 and 2 are *interpolated* on the Lagrangian markers in order to obtain the value of the forcing $\mathbf{F}$. This value is *spread* over the Eulerian mesh, to estimate a forcing term $\mathbf{f}$ for each mesh cell.
4. The whole system is resolved again, starting from stored quantities at the time step n and including the forcing term. Equations are resolved iteratively until convergence is reached:

$$\rho^{n+1} = \frac{\phi_\rho(\rho^\star, \mathbf{u}^\star)}{a_\rho} \tag{16}$$

$$\mathbf{u}^{n+1} = \frac{\phi_\mathbf{u}(\rho^{n+1}, \mathbf{u}^\star)}{a_\mathbf{u}} - \frac{\mathbf{grad} p^\star}{a_\mathbf{u}} + \frac{\mathbf{f}}{a_\mathbf{u}} \tag{17}$$

$$e^{n+1} = \frac{\phi_e(\rho^{n+1}, \mathbf{u}^{n+1}, e^\star)}{a_e} - \frac{div(p^\star \mathbf{u}^{n+1})}{a_e} + \frac{\phi_{fe}(\mathbf{f}, \mathbf{u}^{n+1})}{a_e} \tag{18}$$

$$p^{n+1} = \frac{\phi_p(p^\star, \rho^{n+1}, \mathbf{u}^{\mathbf{n+1}})}{a_p} + \frac{\phi_{fp}(\mathbf{f})}{a_p} \tag{19}$$

In this case, the term $\mathbf{f}$ is not recalculated during the PISO loop and is determined only one time at the beginning of the time step.

3.2. IBM-rhoCentralFoam

The integration of the IBM method in the solver rhoCentralFoam presented in the Appendix A is performed through the following steps :

1. The first predictive step resolving equations A.8 , A.9 , A.10 , A.11 and A.13 is performed to obtain $\rho^\star$, $e^\star$ and $\mathbf{u}^\star$ (and $p^\star$ via state equation). The volume forcing is here $\mathbf{f} = \mathbf{0}$.
2. The physical quantities $\rho^\star$, $p^\star$, $e^\star$ and $\mathbf{u}^\star$ are interpolated in the Lagrangian space and $\mathbf{F}$ is calculated. This field is *spread* over to the Eulerian mesh, so that the value of the forcing term $\mathbf{f}$ for each mesh cell is determined.
3. Equations A.9 , A.10 , A.11 and A.13 are resolved again including the IBM forcing:

$$\rho^{n+1} = \frac{\phi_\rho(\rho^\star, \mathbf{u}^\star)}{a_\rho} \tag{20}$$

$$(\rho\mathbf{u})^{\star\star} = \frac{\phi'_{\mathbf{u}}((\rho\mathbf{u})^\star)}{a_{\mathbf{u}}} - \frac{\mathbf{grad}p^\star}{a_{\mathbf{u}}} \tag{21}$$

$$(\mathbf{u})^{\star\star} = (\rho\mathbf{u})^{\star\star}/\rho^{n+1} \tag{22}$$

$$\rho^{n+1}\mathbf{u}^{n+1} = \rho^{n+1}\mathbf{u}^{\star\star} + \frac{\phi_{\mathbf{u}}(\rho^\star, \mathbf{u}^\star)}{a_{\mathbf{u}}} - \frac{\phi'_{\mathbf{u}}((\rho\mathbf{u})^\star)}{a_{\mathbf{u}}} + \frac{\mathbf{f}}{a_{\mathbf{u}}} \tag{23}$$

$$(\rho e_t)^{\star\star} = \frac{\phi'_{e_t}((\rho e_t)^\star, \mathbf{u}^{\star\star})}{a_{e_t}} - \frac{div(p^\star\mathbf{u}^\star)}{a_{e_t}} \tag{24}$$

$$e^{\star\star} = (\rho e_t)^{\star\star}/\rho^{n+1} - 0.5((\mathbf{u})^{\star\star}.(\mathbf{u})^{\star\star}) \tag{25}$$

$$\theta^{\star\star} = e^{\star\star}/c_v \tag{26}$$

$$\begin{aligned} \rho^{n+1}e^{n+1} &= \rho^{n+1}e^{\star\star} + \frac{\phi_e(\rho^{n+1}, \mathbf{u}^{n+1}, e^n)}{a_e} - \frac{div(\lambda(\theta^{\star\star})\mathbf{grad}(\theta^{\star\star}))}{a_e} \\ &- \frac{\phi'_{e_t}((\rho e_t)^\star, \mathbf{u}^{\star\star})}{a_{e_t}} + \frac{\phi_{fe}(\mathbf{f}, \mathbf{u}^{n+1})}{a_e} \end{aligned} \tag{27}$$

4. Finally, the temperature $\theta^{n+1} = e^{n+1}/C_v$ and the pressure $p^{n+1} = \rho^{n+1} \cdot (r\theta^{n+1})$ are updated.

4. Numerical validation of the IBM based algorithms

Validation of the new solvers is performed on the 2D flow around a circular cylinder. This classical test case has been extensively investigated in the literature for a large spectrum of values of Re and Ma, and numerous databases are available for comparison.

4.1. Test case - numerical details

The size of the computational domain is chosen to be $[-16D, 48D]$ in the streamwise (x) direction and $[-16D, 16D]$ in the vertical (y)-direction. D is the diameter of the cylinder. The physical domain has been determined from IBM results obtained for incompressible flows [18]. The center of the immersed circular cylinder is chosen to be in the origin of the frame of reference (Figure 1). Hexahedral mesh elements have been chosen for the discretization. The physical domain in the region $x \times y \in [-D,\, D] \times [-D,\, D]$ is discretized in homogeneous elements of size $\Delta x = \Delta y = 0.01D$. Outside this central region, a geometric coarsening of the elements is imposed (ratio between neighbor elements $r = 1.05$) in both x and y directions. The resulting total number of mesh elements is equal to 1.5×10^5. In addition, the boundary conditions have been carefully selected for each case accounting for the Ma number investigated, so that their effect over the predicted results may be considered negligible. Generally speaking, a velocity inlet condition is imposed upstream (left side), a mass conserving outlet condition is imposed downstream and slip / non reflective conditions are chosen in the normal direction.

For each case analyzed, the main physical quantities of interest are compared with available data of the literature. In particular the bulk flow coefficients are defined as:

$$C_D = \frac{2F_x}{\rho_\infty U_\infty^2} \; , \qquad C_L = \frac{2F_y}{\rho_\infty U_\infty^2} \tag{28}$$

where the forces F_x and F_y are directly calculated on the Lagrangian points and projected in the streamwise direction x and vertical direction y, respectively. ρ_∞ and U_∞ denote asymptotic physical quantities imposed at the inlet.

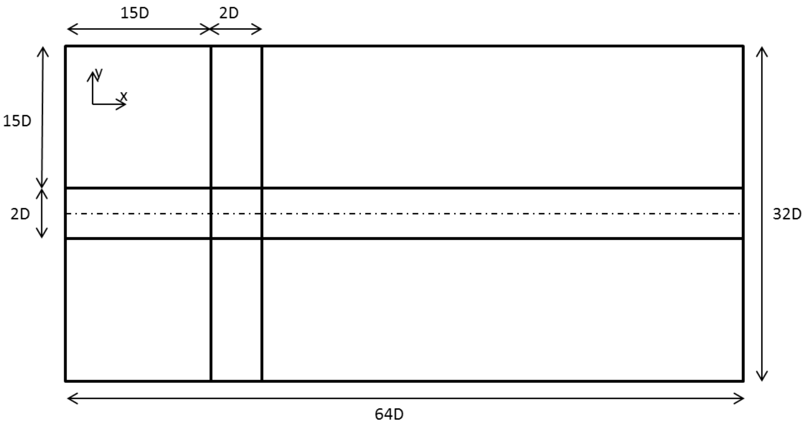


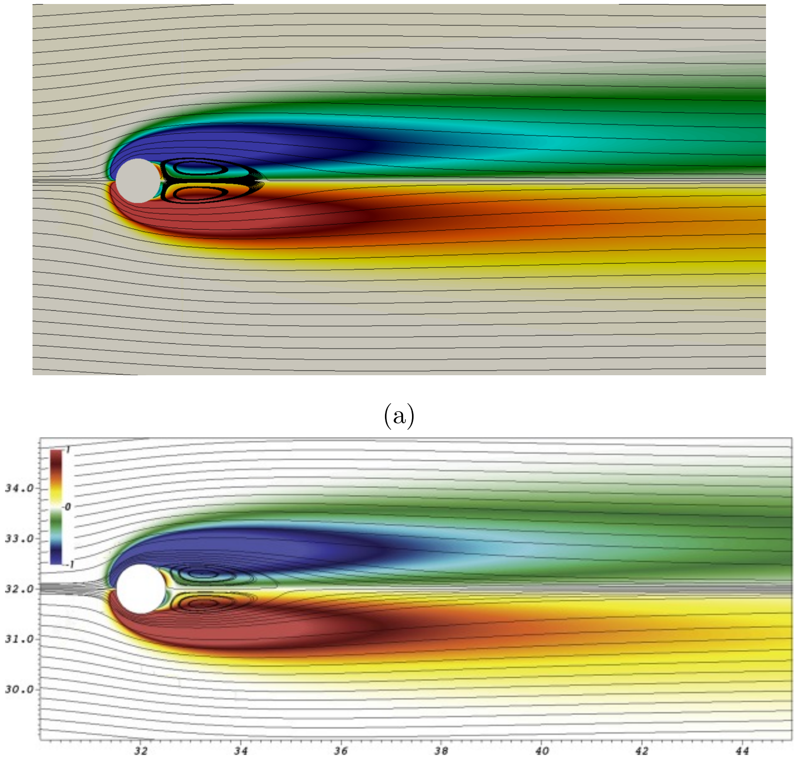
Figure 1: 2D computational domain used for IBM validation.

4.2. Nearly incompressible flow around a circular cylinder case ($Ma = 0.05$)

Because of the almost negligible contribution of compressibility effects in this case, the pressure based solver $IBM - sonicFoam$ is chosen for investigation. Two configurations are studied for Reynolds numbers $Re = 40$ and $Re = 100$.

For $Re = 40$ the flow is characterized by a laminar steady recirculating region (see Figure 2) as the critical point of Bénard - von Kármán instability is not reached. Qualitative comparison of the vorticity isocontours with data taken from the work of Al-Marouf et al. [29] indicate that the structural organization of the flow is well captured. In addition, all characteristic geometrical parameters and bulk flow quantities (drag coefficient C_D) compare very well with the data of literature reported in Table 1, assessing the present results. This includes the pressure coefficient $C_p = \frac{2\times(p-p_\infty)}{\rho_\infty \times \mathbf{U}_\infty^2}$ which is observed to be in good agreement with results by Al-Marouf et al. [29] as shown in Figure 3.

For $Re = 100$ an unstationary behavior characterized by a periodic von Kármán wake is observed. Results include as well the Strouhal number $S_t = \frac{fD}{U_\infty}$, where f is the shedding frequency computed using the time evolution of the lift coefficient C_L. Comparison shows a very good agreement with results available in the literature, see Table 1. For reference, a comparison of the instantaneous



(a)

(b)

Figure 2: Axial vorticity contours and velocity streamlines for the flow past a circular cylinder for $Ma = 0.05$ and $Re = 40$. A zoom around the recirculation region is shown. A qualitative comparison between (a) present IBM simulations and (b) a visualization from the work by Al-Marouf et al. [29] is shown.

axial vorticity isocontours obtained via IBM method with similar results reported in the literature [29] is shown in Figure 4. The time evolution of the lift coefficient is shown in Figure 5(a-d), where $t_a = D/U_\infty$ is the characteristic advection time.

Study	C_D	x_s	a	b	α_s	C_l^{rms}	S_t
Present (Re=40)	1.58	2.35	0.7	0.6	53.7	-	-
Tritton [32] (Exp.)	1.59	-	-	-	-	-	-
Le et al. [33] (Num.)	1.56	2.22	-	-	53.6	-	-
Dennis & Chang [34] (Num.)	1.52	2.35	-	-	53.8	-	-
Coutanceau & Bouard [35] (Exp.)	-	2.13	0.76	0.59	53.5	-	-
Gautier et al. [30] (Exp.)	1.49	2.24	0.71	0.59	53.6	-	-
Chiu et al. [36] (Num.)	1.52	2.27	0.73	0.6	53.6	-	-
Taira & Colonius [15] (Num.)	1.54	2.30	0.73	0.60	53.7	-	-
Brehm et al. [37] (Num.)	1.51	2.26	0.72	0.58	52.9	-	-
Present (Re=100)	1.35	-	-	-	-	0.237	0.164
Berger & Wille [38] (Exp.)	-	-	-	-	-	-	0.16-0.17
Le et al. [33] (Num.)	1.37	-	-	-	-	0.228	0.160
White [39] (Theo.)	1.46	-	-	-	-	-	
Stalberg et al. [31] (Num.)	1.32	-	-	-	-	0.233	0.166
Russell & Wang. [40] (Num.)	1.38	-	-	-	-	0.212	0.172
Chiu et al. [36] (Num.)	1.35	-	-	-	-	0.214	0.167
Liu et al. [41] (Num.)	1.35	-	-	-	-	0.240	0.165
Brehm et al. [37] (Num.)	1.32	-	-	-	-	0.226	0.165

Table 1: Comparison of bulk flow quantities for the flow past a circular cylinder with available data in the literature for $Ma = 0.05$. C_D is the drag coefficient, C_l is the lift coefficient, S_t the Strouhal number, x_s the recirculation length, (a, b) are the characteristic lengths of the vortex structural organization and α_s is the separation angle. Data are provided for (top) $Re = 40$ and (bottom) $Re = 100$, respectively.

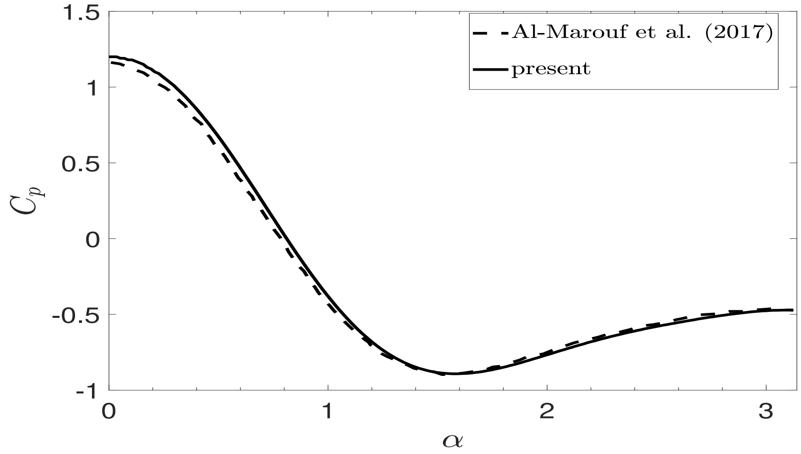
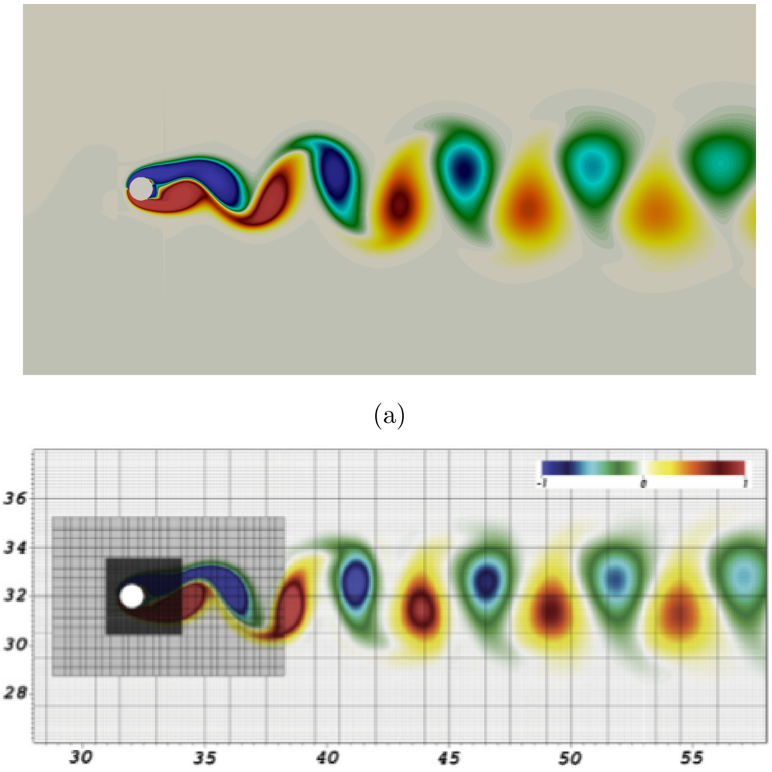


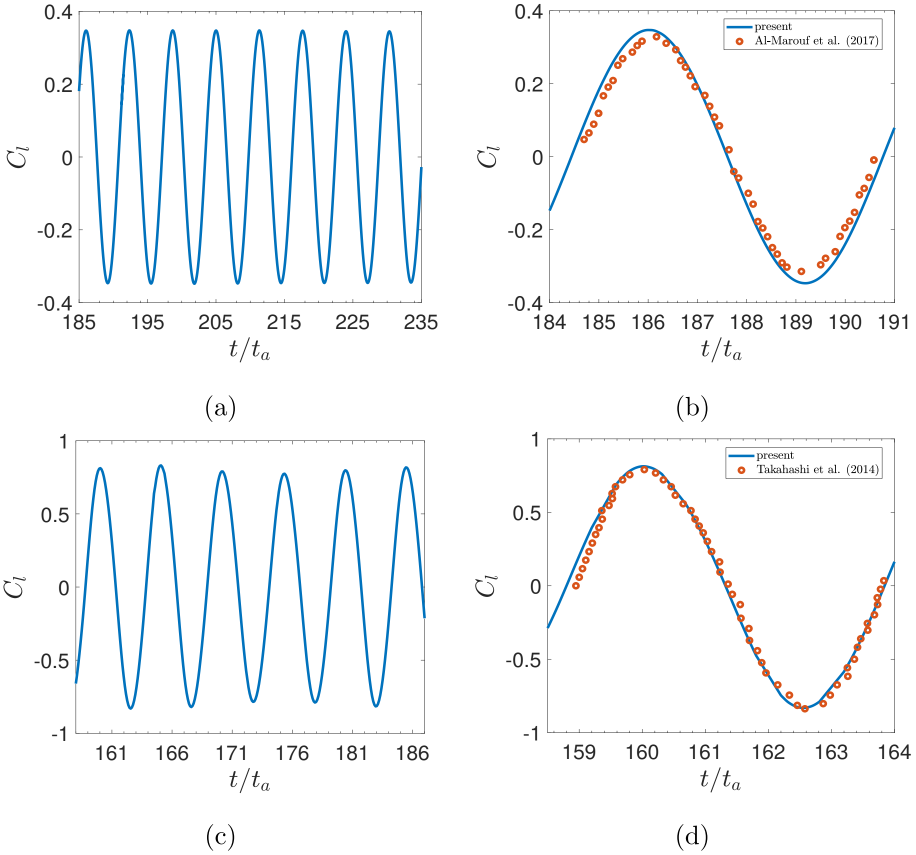
Figure 3: Pressure coefficient C_p along the cylinder surface, for the angle $\alpha \in [0,\, \pi]$. IBM results are compared with data available in the literature, steady solution for $Ma = 0.05$ and $Re = 40$.



(a)

(b)

Figure 4: Axial vorticity isocontours of the flow around a circular cylinder, unsteady solution for $Ma = 0.05$ and $Re = 100$. A qualitative comparison between (a) present IBM simulations and (b) a visualization from the work by Al-Marouf et al. [29] is shown.



(a) (b)

(c) (d)

Figure 5: Time evolution of the lift coefficient C_l for the flow around a circular cylinder for (top row) $Ma = 0.05$, $Re = 100$ and (bottom row) $Ma = 0.05$, $Re = 300$. Present IBM results are shown (left column) over a number of shedding cycles and (right column) compared with data in red markers sampled from the works reported in the literature for a single shedding cycle.

4.3. Subsonic flow around around a circular cylinder, $Ma = 0.3$ and $Re = 300$

The *IBM-sonicFoam* solver is used to perform the present investigation. For this case compressibility effects are not negligible anymore, albeit they do not drive the flow evolution. One notable established observation is that the unstationary vortex shedding does not exhibit a three-dimensional behavior in this case, contrarily to what is obtained for incompressible flows at the same Re. The axial vorticity isocontours are plotted in Figure 6 and the time evolution of the lift coefficient is shown in Figure 5 (c-d).

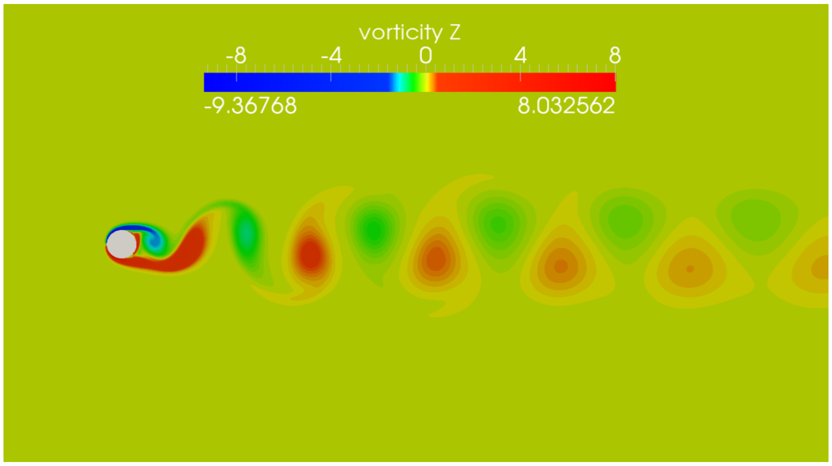


Figure 6: Vorticity isocontours for the flow around a circular cylinder, $Ma = 0.3$ and $Re = 300$.

Present results for the bulk flow quantities are compared with a classical body fitted simulation available in the literature [42] in Table 2. The bulk flow coefficients exhibit a good match with the available reference, assessing the precision of the IBM solver.

4.4. Supersonic flow around cylinder, $Ma = 2.0$ and $Re = 300$

The strong compressibility effects provide a regularization of the flow, which is known to be in this case stationary and two-dimensional. The density isocontours in the near field of the cylinder are shown in Figure 7(a) and compared

Case	C_D	C_l^{rms}	$\triangle_{shock}$
Present (Ma=0.3)	1.5	0.566	-
Takahashi et al. [42] (Num.)	1.444	0.573	
Present (Ma=2)	1.51	-	0.69
Takahashi et al. [42] (Num.)	1.55	-	-
Billig. [43] (Theo.)	-	-	0.62

Table 2: Drag coefficient C_D, standard deviation of the lift coefficient C_l^{rms} and standoff distance $\triangle_{shock}$, computed for the flow around a circular cylinder for $Re = 300$, $Ma = 0.3$ and $Ma = 2$. Present IBM results are compared with data available in the literature.

with a similar representation by Takahashi et al.[42] (Figure 7(b)). The IBM-rhoCentralFoam solver successfully captures the physical behavior of the flow, which exhibits a stationary and symmetric behavior. In addition, a bow shock before the circular cylinder is clearly obtained as shown in Figure 7 (c).

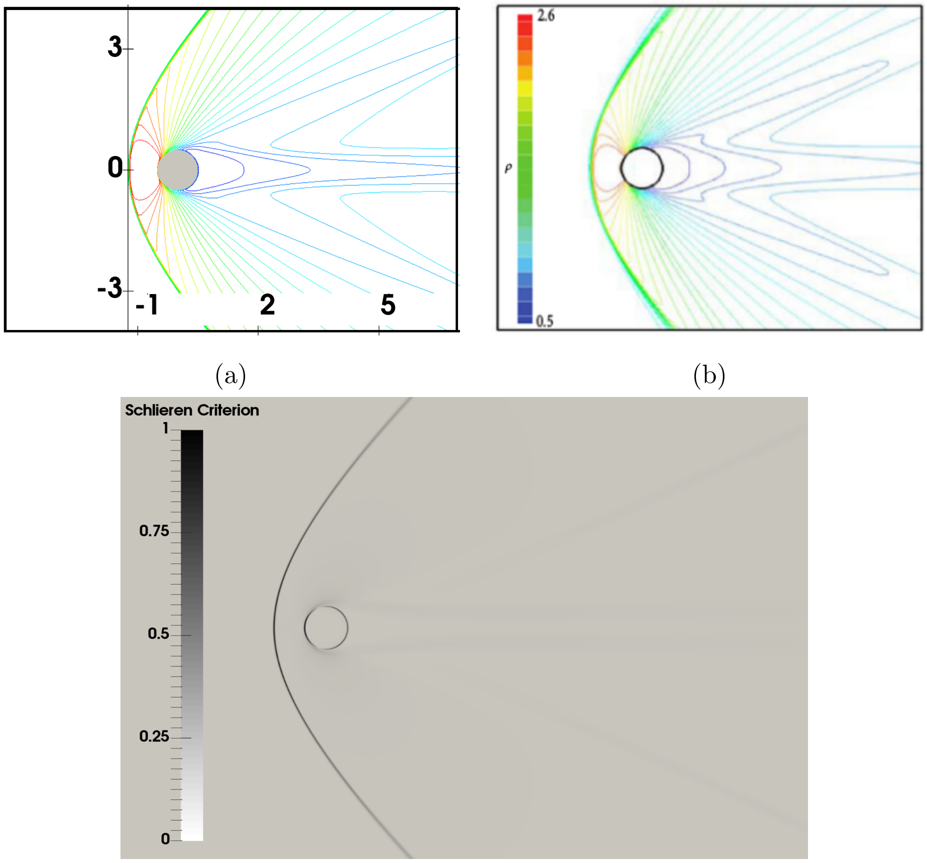
The comparison of the drag coefficient C_D and standoff distance $\triangle_{shock}$ with available data in the literature [42, 43] reported in Table 2 again indicates that a successful prediction of the flow is obtained. The standoff distance $\triangle_{shock}$ is the minimum separation from the shock and the immersed body. Additionally, the pressure coefficient distribution is compared with data from Takahashi et al. [42] in Fig. 8, showing again, a very good match with available reference.

4.5. Effects of the pressure gradient correction in the IBM forcing

At last, the effects of the newly introduced term in the IBM formulation are assessed. To do so, three different numerical settings have been considered:

- Complete IBM forcing as in equation 15
- IBM forcing without pressure correction (i.e. first term of equation 15)
- Body fitted

The three strategies have been applied to the analysis of the flow around a circular cylinder for different values of $Ma \in [0.05, 2]$ and $Re \in [40, 300]$.



(a)

(b)

(c)

Figure 7: Density ρ isocontours for the flow around a circular cylinder for $Ma = 2.0$ and $Re = 300$. (a) Present normalized IBM results are compared with (b) visualizations taken from the work of Takahashi et al.[42]. The legend for ρ is the same for the two figures and the size of the zoom in D units is almost identical, allowing for direct qualitative comparison. (c) Visualization via the normalized Schlieren criterion of the bow shock.

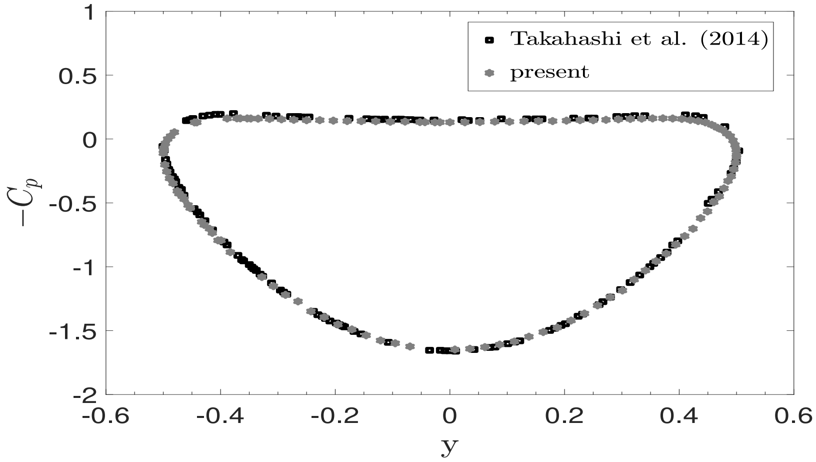


Figure 8: Pressure coefficient distributions for the case $Ma = 2.0$, $Re = 300$. Present IBM results are compared with data available in the literature.

The mesh resolution around the cylinder and in the wake area is roughly the same for the three simulations and it was verified that convergence of the results was reached. In addition, the time step for the simulations is the same and it has been set to comply with the relation $max(Co) = 0.1$, where Co is the Courant number. Results are shown in figure 9 for two different simulations for $Re = 40$, $Ma = 0.05$ and $Ma = 2$. The comparison of the surface distribution of the pressure coefficient C_p is shown. The two configurations have been chosen to highlight the behavior of the IBM forcing over the parametric space investigated. Counter-intuitively, the configuration for $Ma = 2$ is the easiest to predict, because the presence of the bow shock regularizes the wall pressure behavior. In this case, as shown in figure 9(a), the three simulations obtain an almost identical behavior for C_p. For more complex configurations, such as the case for $Ma = 0.05$ in figure 9(b), one can observe that the IBM without pressure correction exhibits accuracy issues. On the other hand, the quality of the prediction using the IBM pressure corrected scheme systematically matches the body-fitted prediction for every configuration investigated. Thus, the inclusion of this term provides a beneficial effect in particular for complex configurations, preventing a degradation of the IBM performance for these applications. The analysis of the database did not indicate that the pressure correction is more beneficial for low or high Ma configurations. It just prevents loss of accuracy for complex applications. Thus, the inclusion of the pressure correction term in the IBM formalism dramatically improves the robustness for the calculation of the pressure field in the near wall region. The precise calculation of the wall pressure distribution is an essential feature governing the emergence and evolution of different dynamic regimes, which are going to be studied for three-dimensional immersed bodies in the next sections.

5. Compressible flow regimes around a sphere

The three-dimensional flow around a sphere is now investigated. As previously mentioned, this investigation encompasses a large range of Ma and Re values, representing a challenging test case for validation.

5.1. Computational grids

The computational domain is here set to $x \times y \times z = [-16D, 48D] \times [-16D, 16D] \times [-16D, 16D]$ where D is the diameter of the sphere. Again, the center of the body is set in the origin of the system. Two computational meshes have been employed to investigate this test case. A first coarser mesh, which will be referred to as *sphereA*, is made by hexahedral uniform elements which are progressively refined approaching the sphere region (see figure 10). The size of the elements is refined by a factor two in each space direction when crossing the prescribed interfaces between regions at different resolution. The central most refined region is defined by the coordinates $x \times y \times z = [-1.25D, 1.25D] \times [-1.25D, 1.25D] \times [-1.25D, 1.25D]$. Within this region, the mesh resolution is $\Delta x = \Delta y = \Delta z = 1/64D$. This mesh is composed by a total of 5×10^6 elements.

A second more refined mesh, which will be referred to as *sphereB* , has been employed to perform a more accurate analysis of the near wall features for a limited number of targeted values of Ma, Re. The mesh is almost identical to *sphereA*, but a higher resolution region is included for $x \times y \times z = [-D,\, D] \times [-D,\, D] \times [-D,\, D]$. Within this region, a resolution $\Delta x = \Delta y = \Delta z = 1/128D$ has been imposed. The total number of mesh elements is in this case 16×10^6.

The size of the mesh elements in the near wall region has been selected accordingly to the recommendations of Johnson and Patel [46]:

$$\Delta x_{min} = \Delta y_{min} = \Delta z_{min} \approx \frac{1.13}{\sqrt{Re} \times 10.0} \tag{29}$$

In order to provide a suitable representation of the physical features of the flow, the *IBM-sonicFoam* solver is used for $Ma \leq 0.3$ and conversely the *IBM-rhoCentralFoam* solver is employed for $Ma > 0.3$.

The numerical simulations have been performed using the native mpi parallelization software available in OpenFOAM and the physical domain has been partitioned in 40 and 64 sub-domains for *SphereA* and *SphereB*, respectively. For the simulation of steady cases, flow convergence is obtained after approximately 90 scalar hours for simulations using mesh *sphereA* and 150 hours for simulations using mesh *sphereB*. For unsteady simulations, the CFL number has been fixed to 0.1. The computational resources demanded to perform a full shedding cycle in an established regime is on average equal to 48 - 84 scalar hours for the mesh *sphereA* and for the mesh *sphereB*, respectively.

5.2. Physical regimes observed for moderate Re

This test case has been chosen because of the emergence of different regimes which exhibit a very high sensitivity to the asymptotic values of Ma and Re prescribed at the inlet, representing a challenging test case of validation. In particular, if very low Ma configurations are considered, the flow undergoes a transition from a steady axisymmetric state to a steady planar-symmetric configuration and finally an unsteady regime with progressively higher Reynolds numbers. The two transitions are observed for $Re \approx 210$ and $Re \approx 280$, respectively. For Ma progressively higher, the two threshold Re values increase but they get progressively closer, finally superposing for $Ma \approx 1$. For higher Ma

values, a steady planar-symmetric regime is not observed anymore. A representation of the qualitative features of these three regimes ($Ma = 0.4$) is shown in Figure 11 using vorticity isocontours.

In order to perform a rigorous investigation of this test case, a database of 120 numerical simulations has been performed in the parametric space $[Ma] \times [Re] = [0.3, 2] \times [50, 600]$ using the coarser mesh *sphereA*. Results are compared with recent data reported in the literature for body-fitted numerical simulations using high order discretization schemes [44, 45].

5.3. Emergence of different characteristic regimes: a parametric study

The emergence of different flow regimes with variations in the prescribed values of (Ma, Re) is here investigated. The resulting regimes observed via analysis of the database of 120 simulations performed using the mesh *sphereA* are summarized in Figure 12. The comparison with high precision data by Sansica et al. [45] indicates that very similar thresholds for the transition between dynamic regimes are obtained as shown in Figure 13. Maximum differences observed are of the order of $\approx 8\%$ of the Reynolds number. These maximum differences are observed for $Ma \approx 0.8$, $Re \approx 250$ where Sansica et al. [45] hypothetised a linear evolution of the threshold value which was determined via stability analysis from a limited number of simulations. Thus, it is arguable that this relatively small difference in the results could simply be associated with the strategy of investigation. In particular, the very larger number of IBM numerical simulations here performed around the parameter value $Ma \approx 1$ suggests that the disappearence of the steady planar-symmetric regime is rather abrupt and not linearly progressive.

The database results have been employed to perform as well quantitative analyses of the main bulk quantities characterizing the flow regimes. A map of the drag coefficient C_d, the separation angle α_s and the recirculation length x_s as a function of Re and Ma are shown in Figures 14. The comparison of the present results with data by Nagata et al. [44] further assesses the precision of the proposed IBM method.

	Studies	$\overline{C_D}$	x_s	St	Δ_{shock}
Ma=0.3	**Present (Re=300)** *sphereA*	0.72	2.15	0.118	-
	Present (Re=300)	0.703	2.05	0.123	-
	Nagata [44] (Num.)	0.68	2	0.128	-
	Present (Re=600) *sphereA*	0.605	2.2	0.135	-
	Present (Re=600)	0.58	2.1	0.143	-
	Krumins [47] (Exp.)	0.54	-	-	-
Ma=0.95	**Present. (Re=50)**	2.116	1.15	-	-
	Present (Re=600)	0.91	4.1	0.138	-
	Krumins [47] (Exp.)	0.9	-	-	-
Ma=2	**Present (Re=300)**	1.39	1	-	0.2
	Nagata [44] (Num.)	1.41	1	-	0.2
	Present (Re=600)	1.27	1.1	-	0.18
	Krumins [47] (Exp.)	1.17	-	-	-

Table 3: Bulk flow quantities for the flow past a sphere, obtained via IBM simulation. The refined grid *sphereB* is used for all but two cases, where *sphereA* has been chosen. Present results are compared with available data in the literature. C_D is the time-averaged drag coefficient, x_s is the recirculation length, S_t is the Strouhal number and Δ_{shock} is the shock distance.

5.4. Investigation of the subsonic flow around a sphere

The unsteady flow configurations are analysed using the mesh *sphereB* for the two sets of parameters (Ma, Re)=(0.3, 300) and (0.3, 600). For these two cases, an unsteady behavior is obtained as shown by the time evolution of the lift coefficient C_l shown in Figure 15. The drag coefficient C_d and the Strouhal number St are reported in Table 3. The comparison of these quantities with data from the literature [44, 47] assesses the high level of performance of the proposed IBM-solver. In addition, the comparison with results using the coarse grid *sphereA* highlights very limited differences, which assesses the robustness of the criteria employed to determine the mesh refinement.

5.5. *Finer analysis of transonic regimes*

A limited number of numerical simulations have been performed using the mesh *sphereB* to further investigate the emergence of different dynamic regimes for $Ma = 0.95$. We remind that this threshold value for the Ma number corresponds to an abrupt transition from the steady axisymmetric state to the unsteady regime. Two higher-resolution numerical simulation are performed for $Re = 50$ and $Re = 600$. Isocontours of Ma are shown in Figure 16 (a-b) for the two cases. For the latter, a detached shock can be clearly observed via Q-criterion and Schlieren criterion, which is reported in Figure 16 (c-d). In addition, the comparison of the bulk flow quantities with data from the literature [44, 47], which are reported in table 3, again assesses the precision of the proposed IBM method.

5.6. *Investigation of the supersonic flow around a sphere*

The supersonic flows for $Ma = 2$ are investigated using the refined mesh. In this case, the numerical simulations are performed for $Re = 300$ and $Re = 600$. In this case compressibility effects are very strong and a steady axisymmetric configuration is observed in both cases. The analysis of the main bulk flow quantities, which is reported in table 3, indicates that all the physical features are accurately captured, when compared with data in the literature [47, 44]. Qualitative representations via isocontours of the Ma number and the Schlieren criterion are shown in Figure 17(a) and in Figure 17(b), respectively. These results assess the correct representation of the physical features of the flow via IBM.

6. Flow around a sphere under rotation

In this section, a flow configuration including an immersed moving body is studied. In order to consistently advance with respect to the analyses in the previous sections, the flow around a rotating sphere is investigated. The sphere rotates with constant angular velocity ω around the z axis. The asymptotic inlet

Mach number Ma_∞ of the flow in the streamwise x direction and the rotational Mach number $Ma_\omega = \omega D/2$ characterizing rotation are:

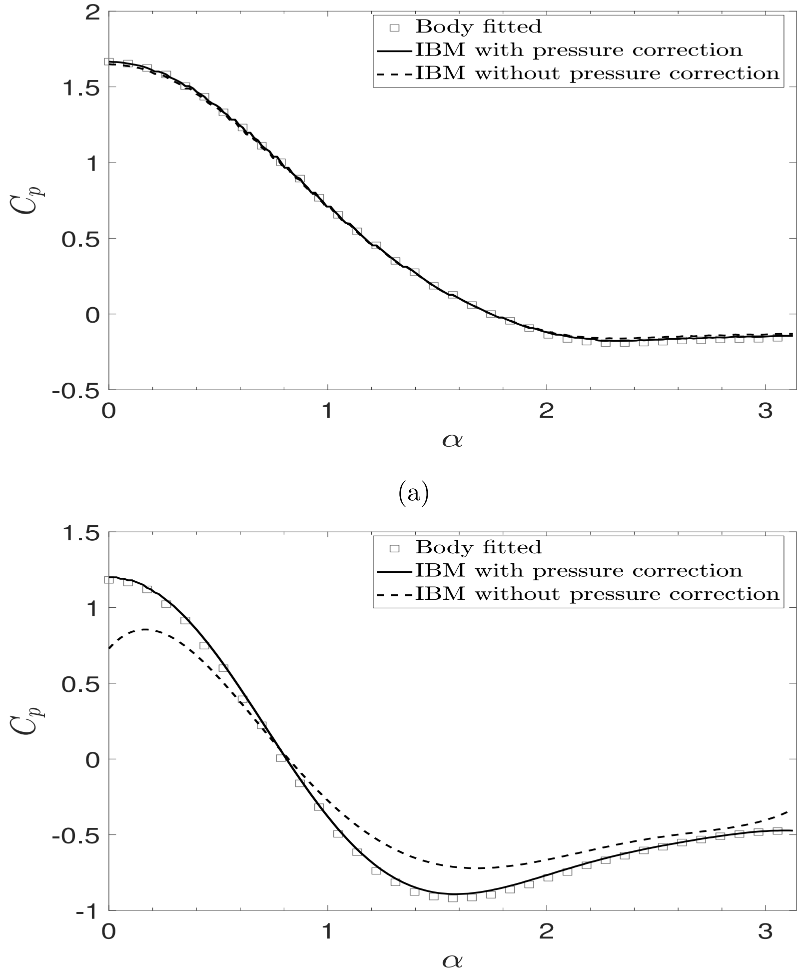
$$Ma_\infty = 0.5 \tag{30}$$

$$Ma_\omega = 0.5 \tag{31}$$

Two simulations are performed for $Re = 200$ and $Re = 300$, respectively. They are compared with correspondent IBM simulations of the flow around a fixed sphere i.e. $Ma_\omega = 0$. The comparison between the four simulations is reported in Figures 18 , 19 and 20 using the Q criterion, velocity streamlines and Ma isocontours. For the case $Re = 200$, the flow without imposed rotation is stationary. However, the sphere rotation triggers the emergence of an unsteady regime, where coherent structures are periodically advected downstream. Additionally, the streamlines behind the sphere lose their symmetric behavior. For $Re = 300$, both flow configurations are unstationary. However, the effect of the rotation is clearly visible in the evolution of the flow quantities. In particular, the recirculation bubble is not symmetrical anymore, and a lift effect is obtained. More interesting features can be deduced by the analysis of the bulk flow coefficients reported in table 4. Generally speaking, the rotation is responsible for an increased value of the drag coefficient C_D and the Strouhal number S_t. However, the generation of a lift force is as well observed, which is usually referred to as Magnus effect. The analysis of the present results indicates that the IBM model successfully captures this physical feature.

Studies	C_D	C_l	C_l^{rms}	S_t
Re = 200				
$Ma_\omega = 0$	0.87	0	-	-
$Ma_\omega = 0.5$	1.02	0.5	0.46	0.17
Re = 300				
$Ma_\omega = 0$	0.77	0.08	0.068	0.12
$Ma_\omega = 0.5$	0.92	0.47	0.45	0.22

Table 4: Bulk flow quantities for the flow around a sphere under rotation.



(a)

(b)

Figure 9: Distribution of the pressure coefficient C_p obtained via body-fitted and IBM numerical simulations. Data are visualized with respect to the angle $\alpha \in [0,\, \pi]$. The case of the stationary flow around a circular cylinder for $Re = 40$ is investigated. Configurations for (a) $Ma = 2$ and (b) $Ma = 0.05$ are shown, respectively.

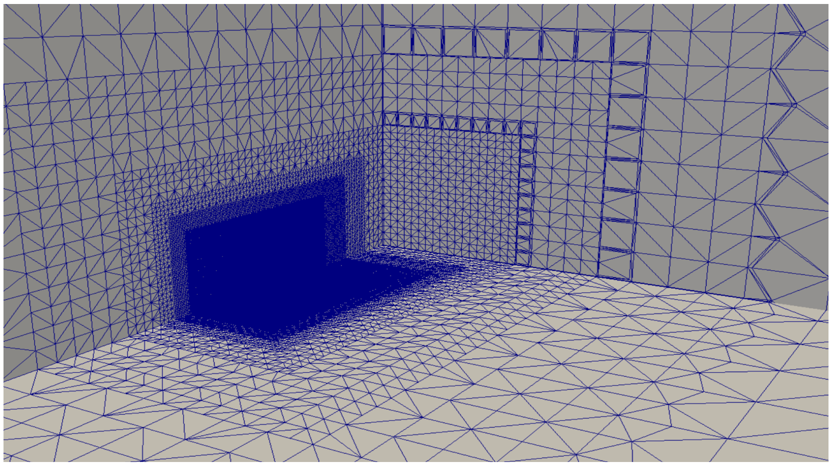
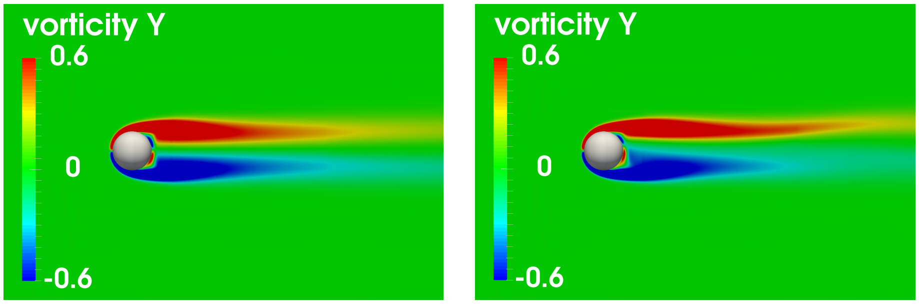
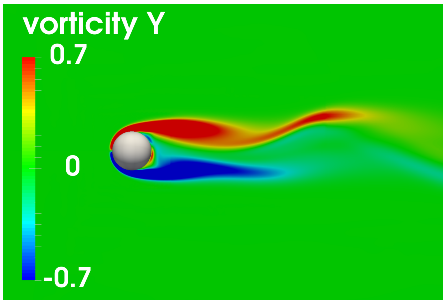


Figure 10: Visualization of cutting planes inside the 3D mesh used for the calculation of the flow around a sphere.



(a) (b)



(c)

Figure 11: Vorticity contours for the flow around a sphere for $Ma = 0.4$, (a) $Re = 205$ (steady axisymmetric state), (b) $Re = 250$ (steady planar-symmetric configuration) and (c) $Re = 300$ (unsteady regime). The vorticity component around the y axis is shown.

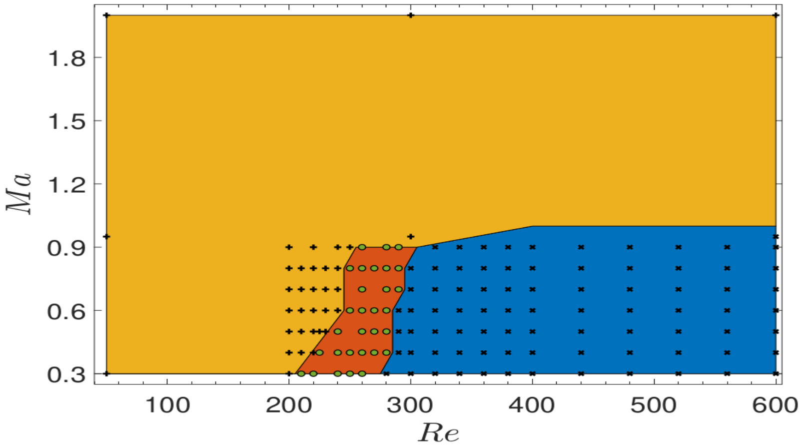


Figure 12: Emergence of different characteristic regimes for the flow around a sphere, as a function of *Re* and *Ma*: (+) steady axisymmetric flow, (●) steady planar-symmetric flow, (×) unsteady periodic flow.

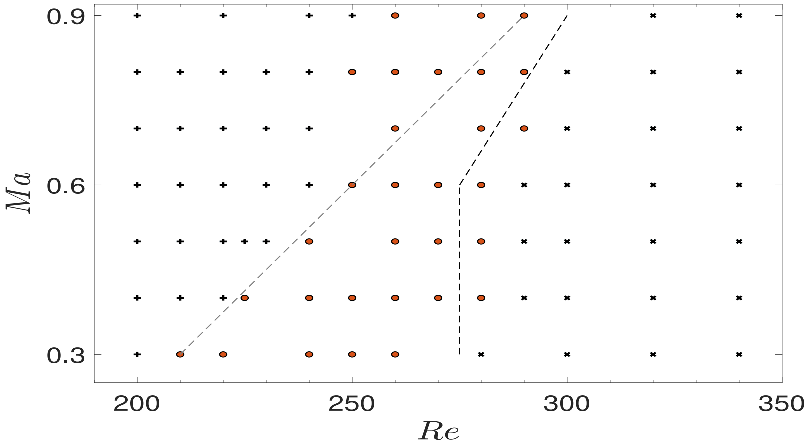
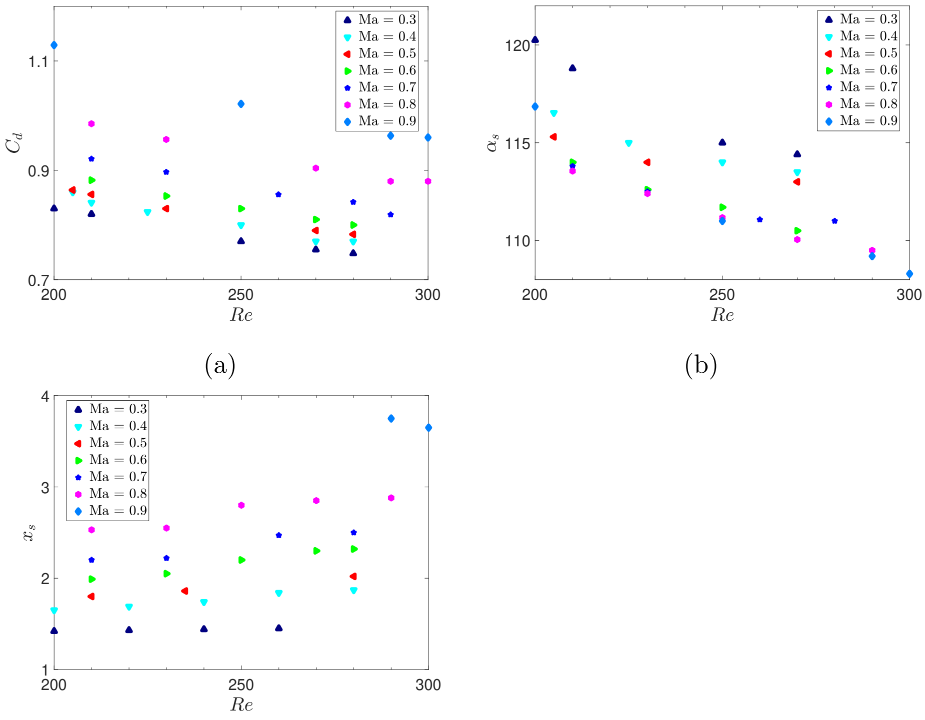


Figure 13: Type of flow field for subsonic regime: (+) Steady axisymmetric flow, (●) Steady planar-symmetric flow, (×) Unsteady periodic flow. Dashed lines represent threshold values for the change in dynamic regime, as calculated via stability analysis by Sansica et al. [45].



(a)

(b)

(c)

Figure 14: (a) Drag coefficient C_d, (b) separation angle α_s and (c) recirculation length x_s as a function of Re and Ma. Data are sampled from a database of 120 simulations.

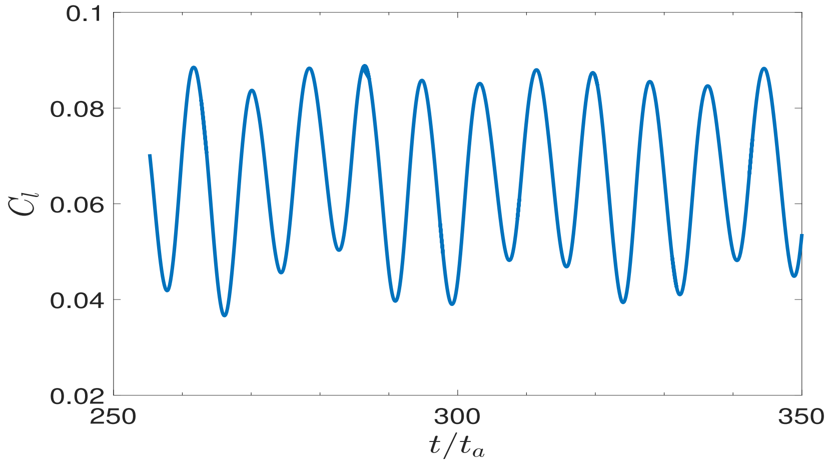
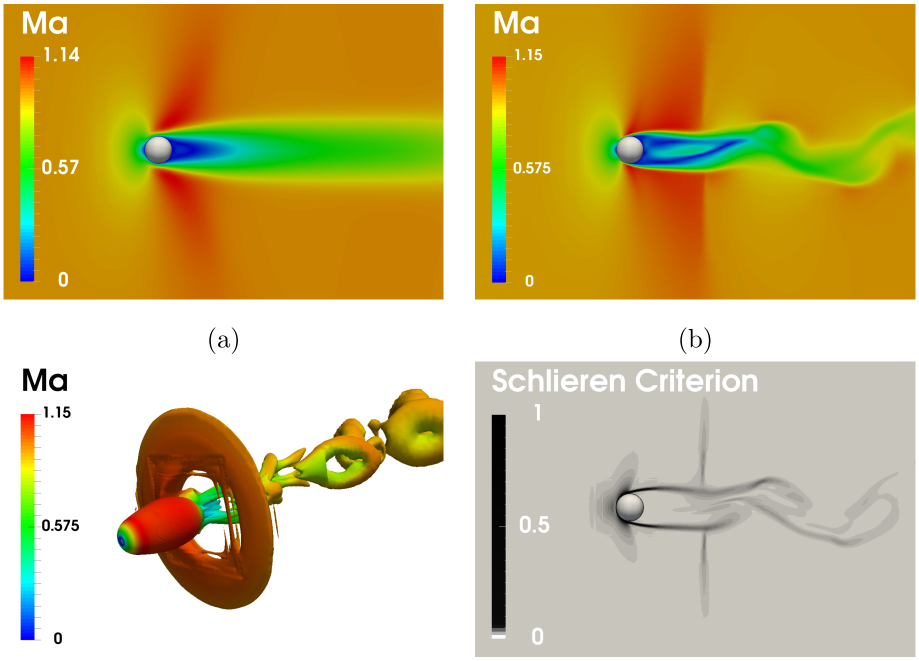


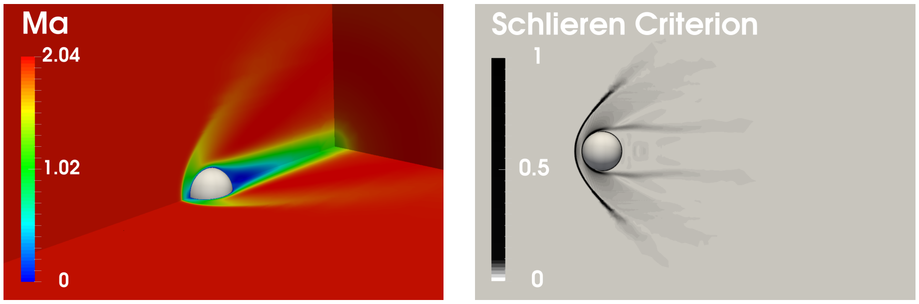
Figure 15: Time evolution of the lift coefficient C_l for the flow around a sphere for $Ma = 0.3$ and $Re = 300$.



(a) (b)

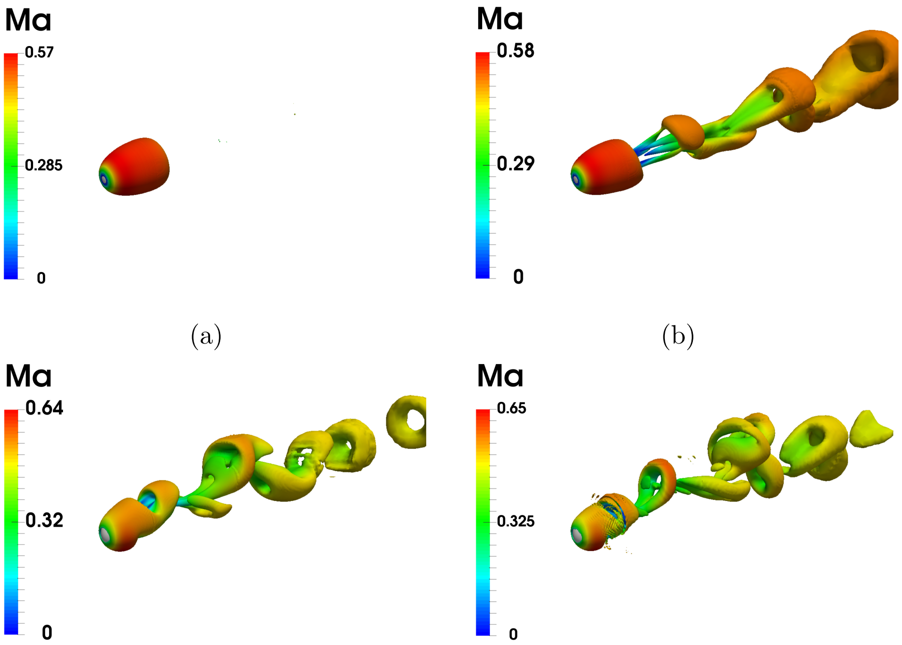
(c) (d)

Figure 16: Numerical simulation of the transonic flow around a sphere for $Ma = 0.95$, using the high resolution mesh *sphereB*. Isocontours of the Ma number are shown for (a) $Re = 50$ and (b) $Re = 600$, respectively. For the latter case, a detached shock is observed via (c) Q-criterion and (d) normalized Schlieren criterion is shown.



(a) (b)

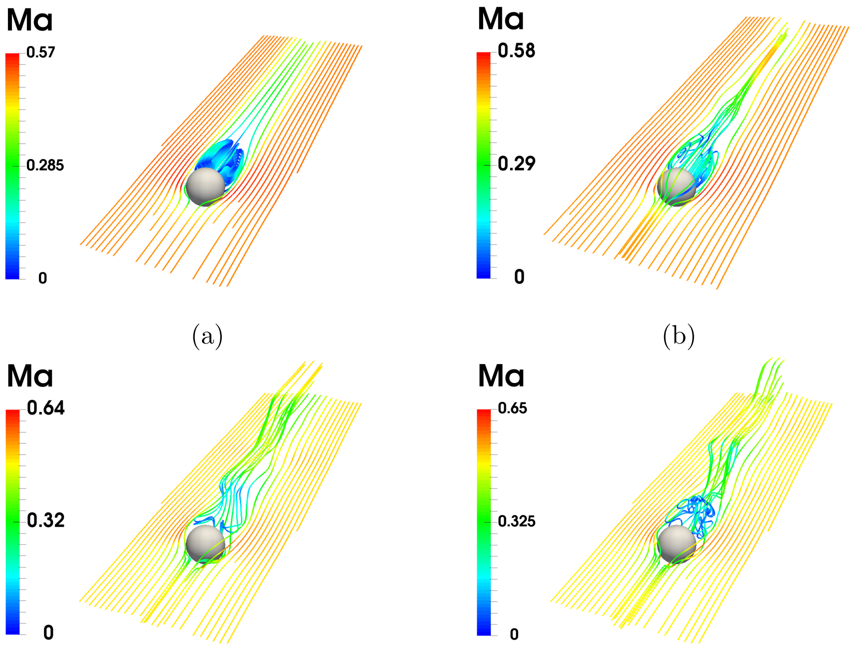
Figure 17: Numerical simulation of the supersonic flow around a sphere for $Ma = 2$, using the high resolution mesh *sphereB*. (a) Isocontours of the Ma number are shown for $Re = 300$ and (b) the normalized Schlieren criterion is presented for $Re = 600$.



(a) (b)

(c) (d)

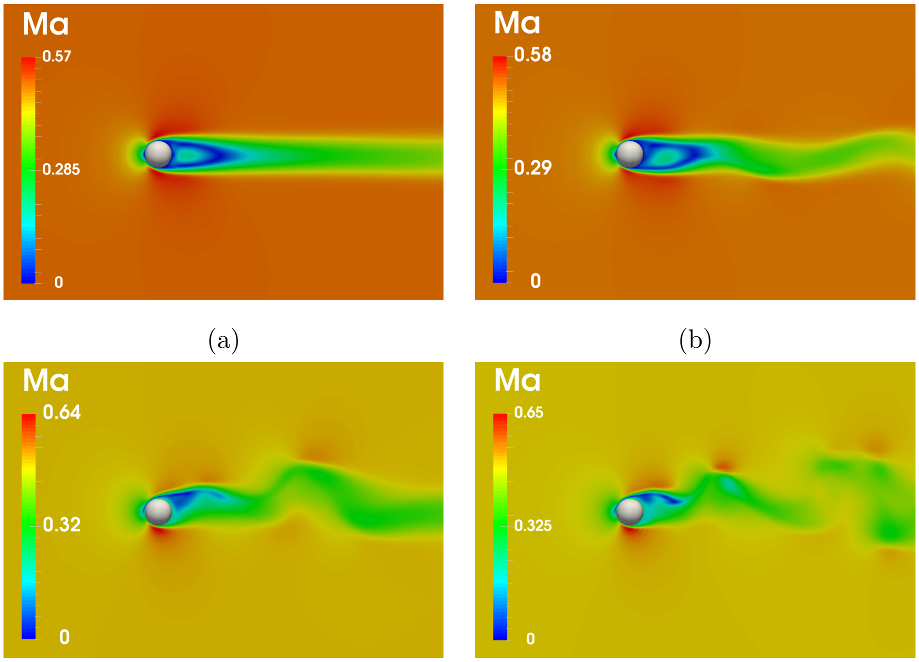
Figure 18: Vortex structures for the flow around a sphere under rotation for $Ma_\infty = 0.5$. The configurations (a) $Ma_\omega = 0$, $Re = 200$, (b) $Ma_\omega = 0$, $Re = 300$, (c) $Ma_\omega = 0.5$, $Re = 200$ and (d) $Ma_\omega = 0.5$, $Re = 300$ are investigated, respectively.



(a) (b)

(c) (d)

Figure 19: Streamlines for the flow around a sphere under rotation for $Ma_\infty = 0.5$. The configurations (a) $Ma_\omega = 0$, $Re = 200$, (b) $Ma_\omega = 0$, $Re = 300$, (c) $Ma_\omega = 0.5$, $Re = 200$ and (d) $Ma_\omega = 0.5$, $Re = 300$ are investigated, respectively.



(a) (b)

(c) (d)

Figure 20: Isocontours of the Mach number for the flow around a sphere under rotation for $Ma_\infty = 0.5$. The configurations (a) $Ma_\omega = 0$, $Re = 200$, (b) $Ma_\omega = 0$, $Re = 300$, (c) $Ma_\omega = 0.5$, $Re = 200$ and (d) $Ma_\omega = 0.5$, $Re = 300$ are investigated, respectively.

7. Conclusions

The development of an improved IBM method is proposed in the present article. This method is based on previous works for incompressible flows and it is expanded towards the analysis of compressible configurations. The most essential feature of this model is the integration of a pressure-based correction of the IBM forcing which is analytically derived from the dynamic set of equations. The resulting IBM method has been integrated in different flow solvers available in the CFD platform *OpenFOAM*. A rigorous validation has been performed considering different test cases of increasing complexity. The results have been compared with a large number of references available in the literature of experimental and numerical nature. The analysis highlights numerous favorable characteristics of the IBM method:

- **precision**. The validation process has encompassed different test cases over a large spectrum of dynamic regimes in the range of investigation $Ma \in [0.05, 2]$, $Re \in [40, 600]$. For each case investigated, the IBM simulation successfully predicted the physical quantities investigated. This level of precision is intimately tied with the pressure correction term, which allows for prescribing more sophisticated condition in the near wall region. Even if classical choices have been employed in the present research work, this observation open new research paths for IBM advancement.

- **flexibility**. The IBM method proved to work remarkably well when implemented in two completely different flow solvers. This aspect indicates that an efficient performance should be granted even with implementation to other codes available for CFD investigation.

- **computational costs**. The determination of the IBM forcing exploits the recursive calculation features of the numerical algorithms, so that a whole time advancement without IBM forcing is not needed anymore. This aspect provides a computational advancement with respect to early development of similar IBM strategies.

The application of the proposed IBM method to the analysis of three-dimensional flows confirmed its capabilities to capture fine physical features of the emerging wake dynamic regimes. Comparison of the present results with body-fitted DNS using high order schemes highlighted minimal differences, which is a signature of the precision of the proposed method in the representation of flow configurations exhibiting flow separation. This class of flow is observed in a large number of industrial flows and transport engineering applications.

The research work has been developed employing computational resources within the framework of the project gen7590-A0012A07590 DARI-GENCI and Mesocentre of Poitiers.

8. References

[1] S.L. Brunton, B.R. Noack, Closed-Loop Turbulence Control: Progress and Challenges, *Applied Mechanics Reviews* **67**(5): 050801 (2015).

[2] O. Ellabban, H. Abu-Rub, F. Blaabjerg, Renewable energy resources: Current status, future prospects and their enabling technology, *Renewable and Sustainable Energy Reviews* **39**:748–764 (2014).

[3] C.S. Peskin, Flow Patterns Around Heart Valves: A Numerical Method, *Journal of Computational Physics* **10**:252–271 (1972).

[4] R. Mittal, G. Iaccarino, Immersed Boundary Methods, *Annual Review of Fluid Mechanics* **37**:239–261 (2005).

[5] Y. Cheny, O. Botella, Set Method for the Computation of Incompressible Viscous Flows in Complex Moving Geometries with Good Conservation Properties, *Journal of Computational Physics* **229**:1043-1076 (2010).

[6] R. Glowinski, T.-W. Pan, T. Hesla, A distributed Lagrange multiplier/fictitious domain method for particulate flows, *International Journal of Multiphase Flow* **25**:755-794 (1999).

[7] L. Isoardi, G. Chiavassa, G. Ciraolo, P. Haldenwang, E. Serre, P. Ghendrih, Y. Sarazin, F. Schwander, P. Tamain, Penalization modeling of a limiter in the Tokamak edge plasma, *Journal of Computational Physics* **229**:2220-2235 (2010).

[8] C.S. Peskin, Numerical analysis of blood flow in the heart , *Journal of Computational Physics* **25 (3)**:220-252 (1977).

[9] R. Beyer, R. LeVeque, Analysis of a One-Dimensional Model for the Immersed Boundary Method, SIAM *Journal on Numerical Analysis* **29 (2)**:332-364 (1992).

[10] D. Goldstein, R. Handler, L. Sirovich, Modeling a No-Slip Flow Boundary withan External Force Field, *Journal of Computational Physics* **105 (2)**:354-366 (1993).

[11] J.P. Mohd-Yusof, Combined Immersed-Boundary/B-spline methods for simulations of flow in complex geometries, *Center for Turbulence Re-search - Annual REsearch Briefs* **1**:317-327 (1997).

[12] E. Fadlun, R. Verzicco, P. Orlandi, J. Mohd-Yusof, Combined Immersed-Boundary Finite-Difference Methods for Three-Dimensional Complex Flow Simulations, *Journal of Computational Physics* **161 (1)**:35-60 (2000).

[13] J. Kim, D. Kim, H. Choi, An Immersed-Boundary Finite-Volume Method for Simulations of Flow in Complex Geometries, *Journal of Computational Physics* **171 (1)**:132-150 (2001).

[14] E. Balaras, Modeling complex boundaries using an external force field on fixed Cartesian grids in large-eddy simulations, *Computers and Fluids* **33 (3)**:375-404 (2004).

[15] K. Taira, T. Colonius, The immersed boundary method: A projection approach, *Journal of Computational Physics* **225 (10)**:2118-2137 (2007).

[16] M. Uhlmann, An immersed boundary method with direct forcing for the simulation of particulate flows, *Journal of Computational Physics* **209**:448–476 (2005).

[17] A. Pinelli, I.Z. Naqavi, U. Piomelli, J. Favier, Immersed-boundary methods for general finite-difference and finite-volume NavierStokes solvers, *Journal of Computational Physics* **229**:9073–9091 (2010).

[18] E. Constant, J. Favier, M. Meldi, P. Meliga, E. Serre, An Immersed Boundary Method in OpenFOAM : verification and validation, *Computers & Fluids* **157**:55–72 (2017).

[19] T. Esposti Ongaro, C. Cavazzoni, G. Erbacci, A. Neri, M.V. Salvetti, A parallel multiphase flow code for the 3D simulation of explosive volcanic eruptions, *Parallel Computing* **33 (7)**:541–560 (2007).

[20] R. Boukharfane, F. Henrique, E. Ribeiro, Z. Bouali, A. Mura, A combined ghost-point-forcing / direct-forcing immersed boundary method (IBM) for compressible flow simulations, *Computers & Fluids* **162**:91–112 (2018).

[21] R.P. Beyer, R.J. Leveque, Analysis of a one-dimensional model for the immersed boundary method, *SIAM Journal on Numerical Analysis* **29**:332–364 (1992).

[22] J.H. Ferziger, M. Perić, Computational Methods for Fluid Dynamics, *Springer* (2002).

[23] I. Raspo, S. Hugues, E. Serre, A. Randriamampianina, P. Bontoux, A spectral projection method for the simulation of complex three-dimensional rotating flows, *Computers & Fluids* **31**:745–767 (2002).

[24] L. Marcantoni, J. Tamagno, S. Elaskar, High speed flow simulation using OpenFOAM, *Mecánica Computacional* **1**:2939–2959 (2012).

[25] R.I. Issa, Solution of the Implicit Discretized Fluid Flow Equations by Operator Splitting, *Journal of Computational Physics* **62**:40–65 (1986).

[26] M. Meldi, A. Poux, A reduced order model based on Kalman Filtering for sequential Data Assimilation of turbulent flows, *Journal of Computational Physics* **347**:207–234 (2017).

[27] A. Kurganov, E. Tadmor, New high-resolution central schemes for nonlinear conservation laws and convection-diffusion equations, *Journal of Computational Physics* **160**:241–282 (2000).

[28] A. Kurganov, S. Noelle, G. Petrova, Semidiscrete central-upwind schemes for hyperbolic conservation laws and hamilton-jacobi equations, *Journal on Scientific Computing* **23**:707–740 (2001).

[29] M. Al-Marouf, R. Samtaney, A versatile embedded boundary adaptive mesh method for compressible flow in complex geometry, *Journal of Computational Physics* **337**:339–378 (2017).

[30] R. Gautier, D. Biau, E. Lamballais, A reference solution of the flow over a circular cylinder at $Re = 40$, *Computers & Fluids* **75**:103–111 (2013).

[31] E. Stålberg, A. Brüger, P. Lötstedt, A. Johansson, D. Henningson, High order accurate solution of flow past a circular cylinder, *Journal of Scientific Computing* **27**:431–441 (2006).

[32] D. Tritton, Experiments on the flow past a circular cylinder at low Reynolds numbers, *Journal Fluid Mechanics* **6**:547–567 (1959).

[33] D. Le, B. Khoo, J. Peraire, An immersed interface method for viscous incompressible flows involving rigid and flexible boundaries, *Journal of Computational Physics* **220**:109–138 (2006).

[34] S. Dennis, G.-Z. Chang, Numerical solutions for steady flow past a circular cylinder at Reynolds numbers up to 100, *Journal Fluid Mechanics* **42**:471–189 (1970).

[35] M. Coutanceau, R. Bouard, Experimental determination of the main features of the viscous flow in the wake of a circular cylinder in uniform translation. I. Steady flow, *Journal Fluid Mechanics* **79**:231–256 (1977).

[36] P. Chiu, R. Lin, T.W. Sheu, A differentially interpolated direct forcing immersed boundary method for predicting incompressible Navier Stokes equations in time-varying complex geometries, *Journal of Computational Physics* **12**:4476–4500 (2010).

[37] C. Brehm, C. Hader, H. Fasel, A locally stabilized immersed boundary method for the compressible Navier Stokes equations, *Journal of Computational Physics* **295**:475–504 (2015).

[38] E. Berger, R. Wille, Periodic flow phenomena, *Annual Review of Fluid Mechanics* **4**:313–340 (1972).

[39] F. White, Viscous Fluid Flow, *(3rd edition), McGrawHill, New York* (1991).

[40] D. Russell, Z. Wang, A Cartesian grid method for modeling multiple moving objects in 2D incompressible viscous flow, *Journal of Computational Physics* **191**:177–205 (2003).

[41] C. Liu, X. Zheng, C. Sung, Preconditioned multigrid methods for unsteady incompressible flows, *Journal of Computational Physics* **139**:35–57 (1998).

[42] S. Takahashi, T. Nonomura, K. Fukuda, A numerical scheme based on an immersed boundary method for compressible turbulent flows with shocks applications to two-dimensional flows around cylinders, *Journal of Applied Mathematics* **2014**:1–21 (2014).

[43] F. Billig, Shock-wave shapes around spherical and cylindrical-nosed bodies, *Journal of Spacecraft and Rockets* **4**:822–823 (1967).

[44] T. Nagata, T. Nonomura, S. Takahashi, Y. Mizuno, K. Fukuda, Investigation on subsonic to supersonic flow around a sphere at low Reynolds number of between 50 and 300 by direct numerical simulation, *Physics of Fluids* **28**:056101-1–056101-20 (2016).

[45] A. Sansica, J.Ch. Robinet, F. Alizard, E. Goncalves, Three-dimensional instability of the flow around a sphere : Mach evolution of the first and second bifurcations, Submitted to *Journal of Fluid Mechanics* (2018).

[46] T. Johnson, V. Patel, Flow past a sphere up to a Reynolds number of 300, *Journal Fluid Mechanics* **378**:19–70 (1999).

[47] V. Krumins, A review of sphere drag coefficients applicable to atmospheric density sensing, *Naval Ordance Laboratory* (1972).

Appendix A. Native OpenFOAM solvers - algorithmic structure

The solver *sonicFoam* is described first. As previously mentioned, this tool is a segregated, pressure-based solver relying on implicit discretization of the time derivative and a pressure implicit step using a splitting of operators (PISO) and an iterative resolution [25, 26]. The different steps of the algorithm are now described for the time evolution from the time step n to $n+1$. First estimations of the quantities $\rho^\star$, $U^\star$ and $e^\star$ are derived via finite volume discretization of equations 1 , 2 and 3 , respectively:

$$\rho^\star = \frac{\phi_\rho(\rho^n, \mathbf{u}^n)}{a_\rho} \tag{A.1}$$

$$\mathbf{u}^\star = \frac{\phi_{\mathbf{u}}(\rho^\star, \mathbf{u}^n)}{a_{\mathbf{u}}} - \frac{\mathbf{grad} p^n}{a_{\mathbf{u}}} \tag{A.2}$$

$$e_t^\star = \frac{\phi_{et}(\rho^\star, \mathbf{u}^\star, e_t^n)}{a_{et}} - \frac{div(p^n \mathbf{u}^\star)}{a_{et}} \tag{A.3}$$

Here, the terms ϕ_ρ, $\phi_{\mathbf{u}}$ and ϕ_e represent the results of the finite volume discretization for every term with the exception of the pressure related terms and the volume forcing term (which is for the moment considered to be zero for sake of clarity). The terms a_ρ, $a_{\mathbf{u}}$ and a_e include coefficients resulting from the time discretization and possibly turbulence / subgridscale modeling. Two important aspects must be highlighted:

- equations A.1, A.2 and A.3 are not solved simultaneously, but they are strictly resolved in the presented order because of their nested structure;

- the equations are solved using the pressure field calculated at the previous time step n. This feature will be exploited for IBM implementation, as already indicated in the decomposition presented in equations 13 - 15.

The prediction of the new pressure field is obtain via manipulation of the vectorial momentum equation 2 via application of divergence operator. The resulting Poisson equation for the pressure field is:

$$div(\mathbf{grad}p) = -\frac{\partial}{\partial t} div(\rho\mathbf{u}) - div(\mathbf{div}(\rho\mathbf{u}\otimes\mathbf{u})) \tag{A.4}$$

The term $\partial div(\rho\mathbf{u})/\partial t$, which is equal to zero in incompressible flows, can be manipulated using equation 1:

$$\frac{\partial}{\partial t} div(\rho\mathbf{u}) = -\frac{\partial\rho}{\partial t} = -\frac{\partial\rho}{\partial p}\frac{\partial p}{\partial t} = -\Psi\frac{\partial p}{\partial t} \tag{A.5}$$

where the normalized compressibility coefficient $\Psi = \partial\rho/\partial p$ is included. Combining equations A.4 and A.5 provides an evolution equation for p, which can be discretized in the following form:

$$p^{\star} = \frac{\phi_p(p^n, \rho^{\star}, \mathbf{u}^{\star})}{a_p} \tag{A.6}$$

Equation A.6 provides a time advancement for p. The PISO loop consists of:

1. a resolution of equation A.6, which allows to update the pressure field to a state $p^{\star}$
2. equations A.1, A.2 and A.3 are solved using the updated value $p^{\star}$. The new flow quantities are used to provide a new estimation for the pressure field

This loop continues until a suitable convergence criterion set by the user is satisfied. Because of the use of a quasi-Poisson equation to determine the pressure p, this algorithm works best for lower Ma numbers, where compressibility effects are not dominant.

The solver *rhoCentralFoam* is now described. Here KT [27] and KNP [28] numerical schemes are employed, which allow for capturing discontinuity / shock features while conserving a general second order central scheme formulation. The numerical scheme allows the transport of fluid properties by both the flow and the acoustic waves. The integration of the convective term on a control volume V is written:

$$\sum_f \pi_f \sigma_f = \sum_f \beta \pi_{f+} \sigma_{f+} + (1-\beta) \pi_{f-} \sigma_{f-} + \omega_f (\sigma_{f-} - \sigma_{f+}) \tag{A.7}$$

with:

1. the mass flux π_f
2. the volumetric unknown $\sigma = (\rho \mathbf{u}); (\mathbf{u}(\rho \mathbf{u})); (\mathbf{u}(\rho e_t))$
3. f_+ and f_- indicate the two directions of incoming flux and outgoing flux, respectively
4. β the weighted coefficient of f_+ and f_-
5. the diffusive mass flux of the maximum speed of propagation of any discontinuity ω_f

The numerical resolution is here performed following a nested cycle. Initially, weak terms of the evolution equations are neglected. Following this first prediction, progressively more complete evolution equations are considered. With respect to this point, the matrices $\phi'_{\mathbf{u}}$, $\phi_{\mathbf{u}}$, ϕ'_{e_t} and ϕ_e used below represent the finite volume discretization for:

1. the momentum equation excluding the viscous term, the pressure term and the volume forcing term
2. the momentum equation excluding the pressure term and the volume forcing term
3. the total energy equation excluding the heat flux term, the pressure-velocity term and the volume forcing-velocity term

4. the internal energy equation excluding the pressure-velocity term and the volume forcing-velocity term

As previously explained, the coefficient $a_{\mathbf{u}}, a_{e_t}$ and a_e result from the numerical discretization. The governing equations are solved from the time step n to the time step $n+1$ in the following order:

1. The continuity equation 1 for the density ρ^{n+1}:
$$\rho^{n+1} = \frac{\phi_\rho(\rho^n, \mathbf{u}^n)}{a_\rho} \tag{A.8}$$
2. The momentum equation 2 for an intermediate estimate of the momentum $(\rho\mathbf{u})^\star$. In this step, viscous stresses are excluded:
$$(\rho\mathbf{u})^\star = \frac{\phi'_{\mathbf{u}}((\rho\mathbf{u})^n)}{a_{\mathbf{u}}} - \frac{\mathbf{grad} p^n}{a_{\mathbf{u}}} \tag{A.9}$$
3. The velocity field is calculated as $\mathbf{u}^\star = (\rho\mathbf{u})^\star/\rho^{n+1}$
4. The momentum equation, including the viscous stresses, is solved again by combining with equation A.9 :
$$\rho^{n+1}\mathbf{u}^{n+1} = \rho^{n+1}\mathbf{u}^\star + \frac{\phi_{\mathbf{u}}(\rho^n, \mathbf{u}^n)}{a_{\mathbf{u}}} - \frac{\phi'_{\mathbf{u}}((\rho\mathbf{u})^n)}{a_{\mathbf{u}}} \tag{A.10}$$
5. Update momentum : $(\rho\mathbf{u})^{n+1} = \rho^{n+1}\mathbf{u}^{n+1}$
6. The energy equation 3 for the total energy $(\rho e_t)^\star$ is resolved excluding the heat flux term.
$$(\rho e_t)^\star = \frac{\phi'_{e_t}(\mathbf{u}^{n+1}, (\rho e_t)^n)}{a_{e_t}} - \frac{div(p^n\mathbf{u}^n)}{a_{e_t}} \tag{A.11}$$
7. Update of an intermediate estimate of internal energy $e^\star$ associated with $(\rho e_t)^\star$:
$$e^\star = \frac{(\rho e_t)^\star}{\rho^{n+1}} - 0.5(\mathbf{u}^{n+1}\cdot\mathbf{u}^{n+1}) \tag{A.12}$$
and an intermediate estimation of the temperature $\theta^\star = e^\star/c_v$
8. Resolution of the energy equation for the internal energy e^{n+1} including the heat flux term:
$$\rho^{n+1}e^{n+1} = \rho^{n+1}e^\star + \frac{\phi_e(\rho^{n+1}, \mathbf{u}^{n+1}, e^n)}{a_e} - \frac{div(\lambda(\theta^\star)\mathbf{grad}(\theta^\star))}{a_e} - \frac{\phi'_{e_t}(\mathbf{u}^{n+1}, (\rho e_t)^n)}{a_{e_t}} \tag{A.13}$$
9. Then final update of the temperature $\theta^{n+1} = e^{n+1}/c_v$ and the pressure $p^{n+1} = \rho^{n+1}\cdot(r\theta^{n+1})$.

Appendix B. Grid convergence analysis

The accuracy of the proposed IBM method is assessed via the analysis of the flow around a circular cylinder for $Ma = 0.05$ and $Re = 300$. Details of the test case investigated are reported in section 4. Data from the work of Gautier et al. [30] is used as a reference solution. The precision of the IBM method is investigated using L_2 and L_∞ norms so that, for a physical quantity ϕ, the error is estimated as:

$$e_{\phi_{L_2}} = \parallel \phi_{ref} - \phi_G \parallel_2 \tag{B.1}$$

$$e_{\phi_{L_\infty}} = \parallel \phi_{ref} - \phi_G \parallel_\infty \tag{B.2}$$

where ϕ_G is the reference solution [30]. The grid convergence analysis is performed evaluating results from four different grids. The mesh resolution in the near cylinder region is imposed to be $\Delta x = \Delta y = \{\frac{D}{80}; \frac{D}{96}; \frac{D}{112}; \frac{D}{128}\}$ where D is the diameter of the cylinder. The corresponding number of Lagrangian markers employed is $\{252; 302; 352; 402\}$, respectively.

Results for the streamwise velocity u are shown in Figure B.21. The error is calculated selecting points at a distance of $0.52\,D$ from the center of the cylinder. One can observe that order of the grid convergence is almost 2 for the L_2 norm and 1 for the L_∞ norm. These results indicate that the precision of the original method [17] is conserved via the current implementation and it is perhaps even improved when compared with previous analyses using the initial OpenFOAM formulation [18].

The behavior of error in the prediction of the drag coefficient C_D is shown in Figure B.22. In the framework of this IBM method, the drag coefficient is directly calculated using information available on the Lagrangian markers. For this quantity, the rate of convergence is slightly faster than first order.

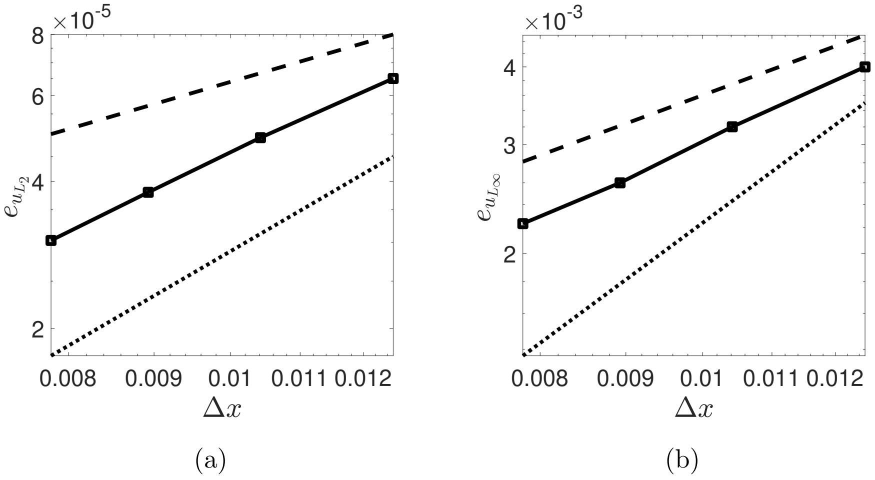


Figure B.21: Convergence rate in the prediction of the streamwise velocity u via the pressure-corrected IBM method. The error is calculated using (a) a L_2 norm and (b) a L_∞ norm, respectively. Dashed lines (first order accuracy) and dotted lines (second order accuracy) are included to highlight the error behavior.

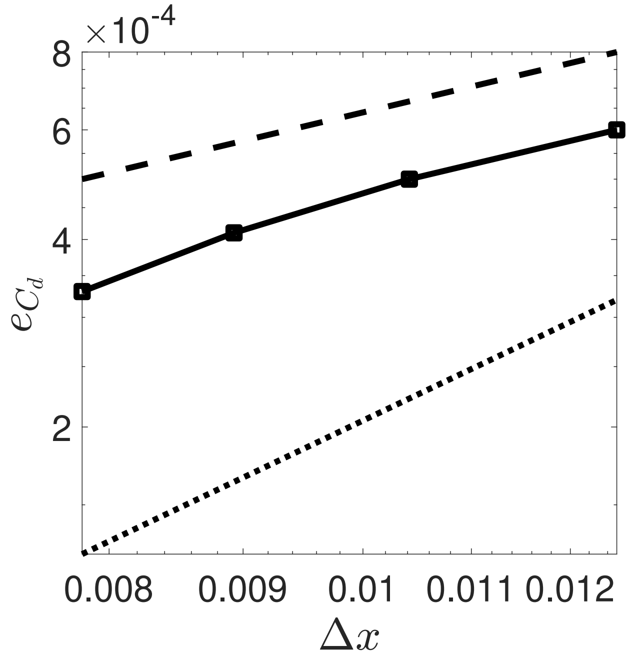


Figure B.22: Convergence rate in the prediction of the drag coefficient C_D via the pressure-corrected IBM method.