Analysis of the immersed boundary method for turbulent fluid-structure interaction with Lattice Boltzmann method

Isabelle Cheylan, Tom Fringand, Jérôme Jacob, Julien Favier

Analysis of the immersed boundary method for turbulent fluid-structure interaction with Lattice Boltzmann method

Isabelle Cheylan *, Tom Fringand, Jérôme Jacob, Julien Favier

Aix Marseille Univ, CNRS, Centrale Marseille, M2P2 UMR 7340, 13451 Marseille cedex 13, France

ARTICLE INFO

Article history:
Received 9 September 2022
Received in revised form 3 June 2023
Accepted 5 August 2023
Available online 11 August 2023

Keywords:
Fluid-structure interaction
Lattice-Boltzmann method
Immersed-boundary method
Lagrangian weight
Predictor-corrector
Large eddy simulation

ABSTRACT

An efficient and new methodology to deal with fluid structure interaction at high Reynolds number flows is presented in this article. It relies on the coupling of the lattice Boltzmann method and the immersed boundary method with a second order predictor corrector model for the structure. The effect of the lagrangian weight of the immersed boundary method is also analyzed in the context of fluid structure interaction. Both curved and moving boundaries are considered, and several methods to calculate the lagrangian weight are compared on relevant test-cases: the laminar flow around a cylinder at Reynolds number $Re = 100$, the Poiseuille flow in a 2D channel and a 3D fluid-structure interaction test case: a deformable flapping flag immersed in a laminar flow. A convergence study in space and time is performed and the three following parameters are investigated: the number of lagrangian markers along the boundary, the value of the lagrangian weight and the shape of the discrete delta function used in the immersed boundary. Finally, the novelty of the paper is two-fold: a new expression of the lagrangian weight which is found to reduce the error by 20% in the case of fluid-structure interaction, and the coupling of a turbulence model with a second order predictor corrector model for improved stability and accuracy. This numerical methodology is found here to be accurate for challenging cases with high added mass effect and high Reynolds number flows.

1. Introduction

The need to accurately capture the motion of a solid in a fluid flow has proven necessary in a number of applications, such as wind turbines, aircraft wing oscillations or biological flows. Fluid-structure interaction deals with the coupled dynamics between a complex fluid flow and a moving or deforming structure, for instance when a fluid flow exerts a load onto a structure, which is significant enough to change the fluid flow. In literature two ways of coupling are encountered, namely one-way or two-way coupling. In the one-way coupling method, we consider that the fluid is barely affected by the small structural deformations which means that only the fluid pressure is transferred from the fluid domain to the structural domain. In the two-way coupling method, we consider that the fluid and solid domains are fully interacting, meaning that the fluid also receives information from the structure solver, namely the displacement and velocity of the structure. In this paper, we propose a two-way coupling between a Lattice-Boltzmann solver for the fluid and a finite element software for the structure. To couple both solvers, we use the immersed boundary method (IBM) with an explicit coupling (or weak

* Corresponding author.
E-mail address: isabelle.cheylan@univ-amu.fr (I. Cheylan).

coupling), and we focus on the derivation of a new expression of the lagrangian weight in the IBM, and its effect on the accuracy of the fluid structure interaction.

The immersed boundary method (IBM) is a useful tool in order to deal with moving or deformable solids in a fluid flow, especially for complex geometries. Indeed, the fluid mesh does not need to be fitted to the solid body, making it possible to handle complex moving and deformable shapes. The no-slip condition is enforced by imposing a fictitious force on the immersed boundary between the fluid and the solid. Peskin [1] introduced the immersed boundary method in the 1970s in order to simulate blood flow in a heart with flexible valves. He was followed by a number of authors who used the IBM for numerous applications, including multiphase flows [2], turbulent flows [3] and porous flows [4]. As detailed in section 2.2, the IBM is based on an interpolation of the macroscopic values followed by a spreading operation of the computed force. If the interpolation and spreading operators are not strictly mathematically reciprocal, fluid can penetrate into the solid, leading to a boundary slip error which is a well-known drawback of the immersed boundary method. A notorious effect of this error is the overprediction of the aerodynamic forces [5,6] even if the overall fluid mass and momentum are conserved in the domain. This issue has been investigated in the past, and several solutions have been proposed to overcome this problem, which are listed below.

In 2005, Uhlmann [7] was the first to propose a predictive step yielding a more regular intermediate velocity field, since there is no discontinuity related to the presence of a boundary in the predictive step. This improves the order of the interpolation with the discretized interpolation kernel which explains the order 2 reported in many works in the literature. He also derived a formula in order to find the number of lagrangian points to define the boundary, based on an evaluation of the lagrangian weight, denoted by W_k hereafter, which can be seen as the volume of influence of each lagrangian marker.

To obtain a reliable method, the total force integrated in the eulerian space must be equal to the total force integrated in the lagrangian space, as pointed by [1,7]. In the continuous space, this condition is not problematic, whereas when evaluating both integrals numerically, one must compute the Eulerian weight, which is simply equal to Δx^3 following the mid-point rule for the integral, and the Lagrangian weight, which cannot be computed directly because a dimension is missing: only the boundary is known in the Lagrangian space, and not the whole volume, by definition of the IBM. It is thus necessary to compute the Lagrangian weight in another way. There exist in literature many ways to tackle this problem, but most of them are not general. Concerning the order of the method, Peng and Luo [5] performed a comparison of the IBM with the interpolated bounce back scheme, as well as a grid convergence study, and concluded that the IBM is a first-order method, which does not strictly satisfy the no slip boundary condition at the boundary. In 2007, Luo [8] stated that the velocities on the Lagrangian points may not satisfy strictly the no-slip boundary condition, and they proposed a correction called “the multidirect forcing technique”, which consists in iterating the boundary force until the interpolated velocity matches the desired velocity at the boundary. Later on in 2009, Wu and Shu [9] considered that the IB force is regarded as an unknown (more precisely the unknown variables are the velocity corrections at the boundary). These unknown variables can be determined by enforcing the no-slip boundary condition, and solving a linear system. Although their approach is accurate, it is cumbersome to implement and to solve at each iteration. Pinelli et al. [10] and Favier et al. [11] proposed a correction factor ϵ, which can be assimilated to the lagrangian weight computed by resolving a linear system in order to ensure the reciprocity of the interpolation and spreading operators, which can be done by inverting a matrix or using a conjugate gradient algorithm to obtain an approximated solution. Vanella and Balaras [12] proposed a moving-least-square approximation to build the transfer functions between the eulerian and lagrangian grids and Kempe and Fröhlich [13] proposed an implicit correction based on a direct forcing approach of the IBM and on the Navier-Stokes equations. Seta et al. [14] also developed an implicit correction of the boundary slip error based on a two-relaxation-time collision operator in the Lattice Boltzmann framework. Akiki and Balachandar [15] expanded the area weights in scalar spherical harmonic modes for only the lowest modes that equal the number of independent equations in order to find the lagrangian weights, but this method adds complexity to the IBM algorithm making it more computationally demanding. In fact, introducing an implicit algorithm or solving a linear system heavily increases the cost of the method.

Another way to improve the accuracy of the no slip condition is to calibrate the boundary force using a multiplying coefficient, as proposed in 2019 by Tao et al. [16]. In 2020, a similar approach was reported by Peng and Wang [17] where optimal values for the coefficients were derived (8/3 for 4-point delta function and 2 for 3-point delta function) from ideal testcases such as Poiseuille flow. Nevertheless this approach cannot satisfy the no-slip condition perfectly. In order to overcome this issue, Gsell et al. [18] recently proposed an analytical correction leading to a very small boundary slip error for a negligible computational cost and without any major modification of the IBM. Zhou and Balachandar [19] did an analysis of the spatio-temporal resolution of the immersed boundary method with direct forcing and investigated the importance of the lagrangian weight in the IBM, which we will detail in section 2.2. They reported that the overprediction of the aerodynamic forces in the IBM is due to the error of the velocity profile near the solid-fluid interface because of the spreading operation and that it is linked to the value of the lagrangian weight which appears in the spreading operation. Finally, Qiuxiang et al. [20] implemented the IBM in non-iterative and iterative ways and discussed the non-iterative version without and with a velocity prediction step. They concluded that for numerical instabilities often encountered in low structure-to-fluid mass ratio cases, the numerical stability can be improved by adding a small feedback coefficient with several iterations. In addition, they also find that the iterative IBM with smaller feedback coefficient has better numerical stability.

These recent works show that the boundary slip error in the IBM is still a crucial topic of an on-going research to improve the reliability of the method, to which our work is contributing. By ensuring the reciprocity of the interpolation and spreading operators, the lagrangian weight is crucial to impose accurately the no slip boundary condition at the interface.

Here, we propose a theoretical development in section 2.4.2 based on a Taylor expansion for the expression of the lagrangian weight. It has the advantage of being independent of the mesh size ratio between the structure and the fluid, and can also be applied to complex geometries regardless of the delta function. Its computational cost is negligible as no iterative method or linear system resolution (i.e. matrix inversion) is needed. We also emphasize here quantitatively the importance of the lagrangian weight for fluid-structure interaction, which affects the accuracy and stability of the coupling.

Another important aspect of this work is the explicit coupling used here. The simultaneous solution of structural dynamics and fluid dynamics is not possible with the explicit approach because each solver (fluid or solid) performs its own iteration before exchanging data (forces or displacements) to the other. This staggering (or time lag) introduced in the coupling can lead to an imbalance of forces exchanged at the fluid/solid interface, which causes numerical instabilities detrimental to the stability. For this reason, Piperno and Farhat [21] proposed several prediction schemes for the structural displacement to counter the time lag introduced in the explicit coupling. With this predictive step, the accuracy and stability of the explicit coupling is improved and the numerical instabilities are decreased. Similarly to this idea, we use in this work a predictor-corrector model proposed by Miranda [22] in 1989, which is inspired by the Newmark method, but which presents better stability and accuracy properties. This results in an implicit-explicit method combining the computational efficiency of the explicit method, and the enhanced stability of the implicit method. This predictor-corrector model is effective in reducing the time lag of the explicit coupling, and it can therefore be used in challenging configurations, involving high added-mass effect for instance [23] or turbulent flows.

Very recently in 2022, Wang et al [24] performed a thorough analysis of the progress of the Lattice Boltzmann for fluid structure interaction. One of their conclusions is that "the further efforts to extend the LBM to two-way FSI at high Reynolds numbers and in 3D domain for realistic applications are highly desired". By including a turbulence model in our work, and validating our method on a complex 3D turbulent fluid structure interaction case, we address this recent issue. Indeed, LES is not well established in the FSI context up to now and only a few articles mention it, including [25] to which we will compare our results. They use an ALE formalism for their coupling scheme, meaning that the grid movement and refinement has to be solved following the movement of the solid. An advantage of our method is that by using the IBM, the grid stays fixed during the movement of the solid, hence we do not need to solve a supplementary equation for the movement of the grid, and this added complexity is avoided.

To summarize, we propose a theoretical development based on a Taylor expansion of the lagrangian weight, which is general and not restricted to any specific delta function, any lagrangian or eulerian grid size, or any specific geometry. It is thus applicable to a wide range of cases. We then perform an analysis of the effect of this lagrangian weight on a fluid structure interaction, which shows a reduction of the error by 20% in this case. Finally, we study a turbulent flow around a moving object with high added mass effect, which demonstrates the accuracy and robustness of our method. We also analyze two prediction schemes for the structural displacement to counteract the time lag introduced in the explicit coupling. Both moving and deformable objects immersed in laminar and turbulent flows are considered and we investigate the influence of the three following parameters:

- the resolution of the lagrangian mesh, *i.e.* the number of lagrangian markers along the boundary
- the value of the lagrangian weight
- the shape of the discrete delta function

The paper is organized as follows. First, the key features of the mathematical and numerical approach are given in Sect. 2. In particular, the reciprocity of the interpolating and spreading operators is also discussed in this section, as well as the details of the coupling between the structure and fluid solvers. A validation on both steady and moving boundary test cases is then presented in Sect. 3, with a final test case consisting in a fully two-way coupled turbulent fluid structure interaction configuration. Conclusions are then drawn in Sect. 4.

2. Mathematical description and numerical implementation

For the fluid, the Lattice-Boltzmann method (LBM) is used, which has been increasingly used in the past decades as an alternative approach to the Navier-Stokes (NS) equations. The LBM relies on a mesoscopic description of the fluid flow using particle distribution functions, and matches the solutions of the NS equations on a macroscopic level. Here, we use the Hybrid Recursive Regularized (HRR) collision model which shows excellent stability and accuracy properties [26].

A brief overview of the Lattice-Boltzmann method is presented here, together with the immersed boundary method and the correction of the no-slip boundary error.

2.1. Basic lattice-Boltzmann equations and collision model

The LBM is used with the D3Q19 lattice discretization as it is classically well-suited to simulate athermal weakly compressible flows. The macroscopic quantities (density ρ, momentum ρu_α, and momentum flux tensor $\Pi_{\alpha\beta}$) are computed as the zeroth order moment: $\rho = \sum_i f_i$, the first order moment: $\rho u_\alpha = \sum_i c_{i,\alpha} f_i$ and the second order moment:

$\Pi_{\alpha\beta} = \sum_i c_{i,\alpha} c_{i,\beta} f_i$ where the Greek letters α and β represent the cartesian coordinates. The Lattice-Boltzmann equation, discretized in space, time, and velocities, is given by

$$f_i(\boldsymbol{x} + \boldsymbol{c_i}\Delta_t, t + \Delta_t) = f_i^{eq}(\boldsymbol{x}, t) + (1 - \frac{\Delta t}{\tau}) f_i^{neq}(\boldsymbol{x}, t) + \frac{\Delta t}{2} h_i(\boldsymbol{x}, t) \tag{1}$$

This equation describes the evolution of a particle distribution function f that represents the probability density of particles with velocity c at time t and position x.

The equilibrium function is written using the HRR (Hybrid Recursive Regularized) collision model [26]. The HRR model is used to enhance numerical stability, by enforcing a traceless lattice-Boltzmann tensor (by simply removing its trace). It is observed to yield accurate and robust results for a broad range of applications and flow physics, e.g. [27–31].

$$f_i^{eq} = \omega_i [\rho + \frac{c_{i\alpha}\rho u_\alpha}{c_s^2} + \frac{a_{\alpha\beta}^{(2),eq} H_{i\alpha\beta}^{(2)}}{2c_s^4} + \frac{a_{\alpha\beta\gamma}^{(3),eq} H_{i\alpha\beta\gamma}^{(3)}}{6c_s^6}] \tag{2}$$

In equation (1), h_i is the external source term representing the effect of the solid on the fluid. In the lattice space, we follow Guo's [32] work to take into account an external force so that h_i is written:

$$h_i = \rho\omega_i (1 - \frac{\Delta t}{2\tau}) [\frac{\boldsymbol{c_i} - \boldsymbol{u}}{c_s^2} + \frac{\boldsymbol{c_i}.\boldsymbol{u}}{c_s^4} \boldsymbol{c_i}].\boldsymbol{g} \tag{3}$$

In the presence of an external force, following [32], the macroscopic velocity is also modified as:

$$\rho \boldsymbol{u} = \sum_i \boldsymbol{c_i} f_i^{eq} + \frac{\Delta t}{2} \boldsymbol{g} \tag{4}$$

where $\boldsymbol{g}$ denotes the external macroscopic volumic force given by the IBM which will be defined in section 2.2.

2.2. Implementation of the immersed boundary method

The immersed boundary force is a fictitious force which takes into account the effects of the structure on the flow. The solid boundary Γ is geometrically described by a set of lagrangian markers X_k, whereas the eulerian fluid nodes occupy the volume Ω and are denoted by x_j. Capital letters indicate the variables at the lagrangian markers and u^* is the predicted velocity without the presence of solid. The spreading operator $S[\boldsymbol{G}](\boldsymbol{x})$ of a function $\boldsymbol{G}$ is written as:

$$S[\boldsymbol{G}](\boldsymbol{x}) = \int_\Gamma \boldsymbol{G}(r) \delta(\boldsymbol{x} - \boldsymbol{X}(r)) \Delta S(r) dr \tag{5}$$

where ΔS is the weight associated to the solid boundary element.

The interpolating operator $I[\phi](r)$ of a function ϕ on the boundary Γ uses the same δ function which will be described in section 2.3:

$$I[\phi](r) = \int_\Omega \phi(\boldsymbol{x}) \delta(\boldsymbol{x} - \boldsymbol{X}(r)) \Delta x^3 \tag{6}$$

The algorithm of the IB method is now described as derived in [18,33]:

1. Update of the position of the lagrangian markers $\boldsymbol{X_k}$ and the velocity of the solid $\boldsymbol{U^d}$
2. Algorithm of the LBM without solid on the eulerian fluid nodes to obtain u^*
   Start of the loop over lagrangian nodes:
3. Interpolation of ρ and $\rho\boldsymbol{u}^*$ on the lagrangian markers $\boldsymbol{X_k}$:

$$I[\rho](\boldsymbol{X_k}) = \sum_j \rho(\boldsymbol{x_j}) \delta(\boldsymbol{x_j} - \boldsymbol{X_k}) \Delta x^3 \tag{7}$$

$$I[\rho\boldsymbol{u}^*](\boldsymbol{X_k}) = \sum_j \rho\boldsymbol{u}^*(\boldsymbol{x_j}) \delta(\boldsymbol{x_j} - \boldsymbol{X_k}) \Delta x^3 \tag{8}$$

4. Computation of the immersed boundary force:

$$\boldsymbol{G} = \frac{2}{\Delta t} (I[\rho]\boldsymbol{U^d} - I[\rho\boldsymbol{u^*}]) \tag{9}$$

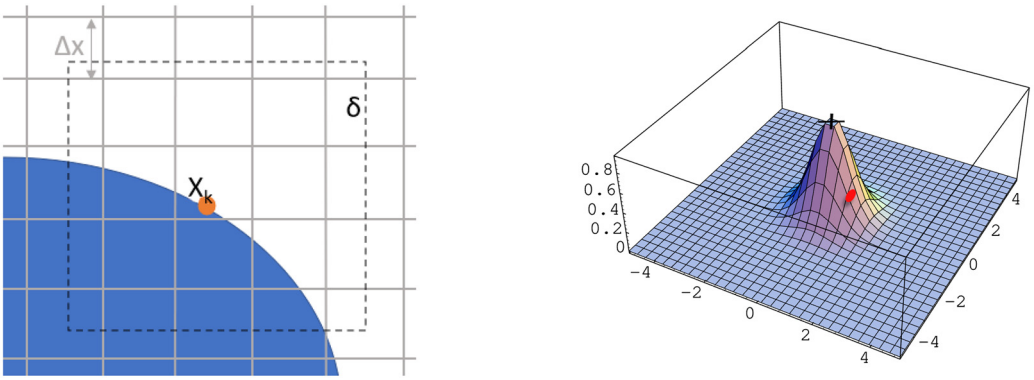

Fig. 1. Immersed boundary method and representation of the delta function. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

5. Spreading of the force on the surrounding eulerian fluid nodes

$$\boldsymbol{g}(\boldsymbol{x_j}) = S[\boldsymbol{G}](\boldsymbol{x_j}) = \sum_k \boldsymbol{G}(\boldsymbol{X_k})\delta(\boldsymbol{x_j} - \boldsymbol{X_k})\Delta S_k \tag{10}$$

End of the loop over lagrangian nodes

6. Update of $\boldsymbol{u}$ (eq. (4)) and f (eq. (1)) in the LBM with the previously computed force

The main advantage of the immersed boundary method is that the motion of the solid can be described with a single mesh and a moving frontier across this original mesh. Thus, there is no need to deform the mesh nor recreate a new mesh during the computation, which are well-known computationally expensive procedures.

2.3. Discrete delta function

In the interpolation or spreading operators, $\delta(x)$ is a mollifier or a smooth approximation to the Dirac delta function. It is the discrete equivalent of the Dirac delta function which serves to transport information from the lagrangian space to the eulerian space and vice versa. In Fig. 1, the dotted square line represents the application field of the δ mollifier function, the blue domain represents the solid and X_k is the lagrangian marker where the density and velocity fields are interpolated from the surrounding fluid nodes.

The delta function needs to verify the two following conditions:

$$\int_\Omega \delta(x - x_0)dx = 1 \tag{11}$$

$$\int_\Omega (x - x_0)^k \delta(x - x_0)^k dx = 0 \tag{12}$$

In other words, the moments of order k of the function δ all cancel except for $k = 0$, where the moment is equal to 1. This formulation yields an error $O((x - x_0)^{(n+1)})$ which allows to fix the accuracy of the method.

We consider and compare three delta functions. The first one is:

$$\delta(r) = \begin{cases} \frac{1}{2d}(1 + \cos(\frac{\pi r}{d})), & |r| \le d, \\ 0, & |r| > d. \end{cases} \tag{13}$$

This delta function is chosen for its simplicity and d is the radius of the delta function, set to 1.5 here, but more sophisticated formulas can be used such as the second one which is a four-point discrete delta function:

$$\delta(r) = \begin{cases} \frac{1}{8}(3 - 2|r| + \sqrt{1 + 4|r| - 4r^2}, & |r| \le 1, \\ \frac{1}{8}(5 - 2|r| - \sqrt{-7 + 12|r| - 4r^2}, & 1 < |r| \le 2 \\ 0, & |r| > 2.0 \end{cases} \tag{14}$$

Finally, the third one is the one initially chosen by Roma et al. [34], and in numerous other works that followed:

$$\delta(r) = \begin{cases} \frac{1}{3}(1 + \sqrt{-3r^2 + 1}, & |r| \le 0.5, \\ \frac{1}{6}(5 - 3|r| - \sqrt{-3(1 - |r|)^2 + 1}, & 0.5 < |r| \le 1.5 \\ 0, & |r| > 1.5 \end{cases} \tag{15}$$

In 2012, Breugem [35] compared different delta functions regarding the "grid locking" phenomenon, which is the appearance of undesired high-frequency oscillations in the force when the interface moves over the Eulerian grid [36]. He found that these oscillations originate from variations in the interpolated prediction velocity when the Lagrangian grid moves with the particle over the Eulerian grid. Uhlmann [7] showed that the amplitude of the spurious oscillations decreases when the width of the regularized Dirac delta function is increased. Breugem [35] showed that the regularized Dirac delta function of Roma et al. [34] with a width of three Eulerian grid cells is considered as effective for suppressing grid locking and its compact support as computationally efficient.

Here, as explained in the introduction, we use a predictive step which consists in solving the fluid without the presence of the solid points, which yields a more regular intermediate velocity field. The grid locking phenomenon is hence reduced since there is no discontinuity related to the presence of a boundary in the predictive step, and we will compare the accuracy of the three previous delta functions in this work.

2.4. Computation of the Lagrangian weight

After interpolating the fluid velocity on the immersed boundary, the force is distributed over the fluid volume using the same delta function, an operation referred to as spreading. Nevertheless, this procedure can lead to boundary slip error, and thus flow penetration. This error can be attributed to a dimension problem, which is inherent to the IBM: while the fluid is described as a volume (3 dimensions in the general case), the structure is described only through its boundary (2 dimensions in the general case). Thus, eulerian and lagrangian spaces do not have the same number of dimensions, and a dimension is missing in the lagrangian space to compute the term ΔS_k in the spreading step (see equation (10)).

We call the lagrangian weight W_k for each lagrangian marker k:

$$W_k = \Delta q \Delta r \Delta s = \Delta S_k \tag{16}$$

It can be seen as a volume of influence of each lagrangian marker. Some authors (including Uhlmann [7]) have chosen to set $W_k = dx^3$ [37], but the original idea of [1] is that the total force integrated numerically in the eulerian space must be equal to the total force integrated numerically in the lagrangian space:

$$\sum_j f_j dx^3 = \sum_k F_k W_k \tag{17}$$

Thus, one must consider the Eulerian weight, which is simply equal to Δx^3 following the mid-point rule for the integral, and the Lagrangian weight, which cannot be computed directly because a dimension is missing: only the boundary is known in the Lagrangian space, and not the whole volume, by definition of the IBM. It is thus necessary to compute the Lagrangian weight in another way. Moreover, as W_k and dx^3 belong to two different discretization spaces, there is no reason why they should be equal.

2.4.1. Linear system resolution

To compute the vector $\boldsymbol{W}$, we will enforce the condition that the interpolation and the spreading are reciprocal operations. In other words, for a given function ϕ defined in the lagrangian space, spreading it and interpolating the obtained eulerian field must give back the same function. For each lagrangian marker $l = 1...N$, this is written by:

$$\phi(l) = \sum_j \left(\sum_k \phi(k) \delta(\boldsymbol{x}_j - \boldsymbol{X}_k) W_k \right) \delta(\boldsymbol{x}_j - \boldsymbol{X}_l) \Delta x \Delta y \Delta z \tag{18}$$

where j is an indice in the eulerian space, and l and k are indices in the lagrangian space. This can be rewritten as:

$$\phi(l) = \Delta x \Delta y \Delta z \sum_k A_{kl} W_k \phi(k) \tag{19}$$

with:

$$A_{kl} = \sum_j \delta(\boldsymbol{x_j} - \boldsymbol{X}_k) \delta(\boldsymbol{x}_j - \boldsymbol{X}_l) \tag{20}$$

The matrix A defined by its generic term A_{kl} in the previous equation is thus constructed by the product of each interpolation kernel k defined at the kth lagrangian marker with the other kernels l defined at the other lagrangian markers l. Finally, if we are looking for the non dimensional weight $\frac{W}{\Delta x \Delta y \Delta z}$, it can be shown that the solution of the problem for a constant function ϕ (having the same values for all lagrangian markers) can be found by solving [10,11]:

$$\boldsymbol{AW} = \boldsymbol{I} \tag{21}$$

where I is a vector of ones of length the number of lagrangian markers. The solution of the system is done either by direct inversion of the matrix A, or by an iterative method, a stabilized biconjugate gradient. The two methods are tested in Pinelli et al. [10] and Favier et al. [11] as well as in the present article, and have been shown to produce results in good agreement with literature for general cases despite the hypothesis on the constant function ϕ.

2.4.2. A new theoretical development of the Lagrangian weight

As proposed by [18], it is possible to find an analytical solution of this problem: by considering that the curvature of the boundary is negligible and that the value of the force is small, one can linearize the expression (18) in the vicinity of each lagrangian marker (see [18] for details). Using the properties of the discrete delta function $\int r\delta(r)dr = 0$ and $\int \delta(r)dr = 1$, we can find that:

$$I[S(\boldsymbol{G})](\boldsymbol{X_m}) = \boldsymbol{G}(\boldsymbol{X_m}) \int \delta(r)^2 dr \tag{22}$$

and that the corrected force noted G^* which ensures the reciprocity of the interpolation and spreading operators, *i.e.* $I[S(\boldsymbol{G})](\boldsymbol{X_m}) = \boldsymbol{G}(\boldsymbol{X_m})$, is given by $\boldsymbol{G}^* = \frac{1}{\kappa}\boldsymbol{G}$, with the correction parameter κ defined by $\kappa = \int \delta(r)^2 dr$ [18]. Hence this parameter κ only depends on the employed regularized Delta function and can be calculated analytically or numerically. This approach is equivalent to defining a discretized lagrangian weight written as follows:

$$W_k = \frac{1}{\sum_j \delta(\boldsymbol{x}_j - \boldsymbol{X}_k)^2} \tag{23}$$

We can see here that in order to calculate the lagrangian weight of the lagrangian point k, we perform a sum on its eulerian support, i.e. the surrounding fluid nodes j of lagrangian point k.

What we propose in this paper is a new calculation of the lagrangian weight, which does not only take into account the eulerian support of the current lagrangian point k, but also those of the neighboring lagrangian points l. The curvature of the boundary is hence taken into consideration, making the method applicable to a wide variety of cases. To do so, we use again the condition that for a given function ϕ defined in the lagrangian space, spreading it and interpolating the obtained eulerian field must give back the same function. For each lagrangian marker $l = 1...N$, this is expressed by equation (18). As for the LBM, $\Delta x = \Delta y = \Delta z$, we can write it as:

$$\phi(l) = \Delta x^3 \sum_j \sum_k \delta(\boldsymbol{x}_j - \boldsymbol{X}_k)\delta(\boldsymbol{x}_j - \boldsymbol{X}_l) W_k \phi(k) \tag{24}$$

We then perform a Taylor expansion of the function ϕ and the lagrangian weight W around the lagrangian point k, with l its lagrangian neighbor, and $|\boldsymbol{X}_k - \boldsymbol{X}_l|$ the distance between the two lagrangian points.

$$\phi(k) = \phi(l) + |\boldsymbol{X}_k - \boldsymbol{X}_l| \frac{\partial \phi}{\partial X} + o(|\boldsymbol{X}_k - \boldsymbol{X}_l|^2) \tag{25}$$

$$W_k = W_l + |\boldsymbol{X}_k - \boldsymbol{X}_l| \frac{\partial W}{\partial X} + o(|\boldsymbol{X}_k - \boldsymbol{X}_l|^2) \tag{26}$$

We now assume that the solid boundary is discretized with a sufficiently fine mesh, meaning that points k and l are close enough so that W_k vary little. We also assume that ϕ vary little on this mesh, which excludes functions with discontinuities at the wall (presence of attached shocks on the boundary for instance). Based on these assumptions, we can neglect the first order in both Taylor developments, and obtain by non-dimensionalizing by Δx^3:

$$\phi(l) = W_l \phi(l) \sum_k \sum_j \delta(\boldsymbol{x}_j - \boldsymbol{X}_k)\delta(\boldsymbol{x}_j - \boldsymbol{X}_l) \tag{27}$$

$$W_l = \frac{1}{\sum_k \sum_j \delta(\boldsymbol{x}_j - \boldsymbol{X}_k)\delta(\boldsymbol{x}_j - \boldsymbol{X}_l)} \tag{28}$$

Here, we do not consider that the normal distance between a lagrangian point and an eulerian point is constant along the boundary. Thus, the assumptions are less restrictive than in [38] because the curvature of the boundary is not neglected. This new expression of the lagrangian weight can be used regardless of the solid and fluid mesh size, and of the chosen delta function. Its computational cost is negligible and we analyze its effect on the global precision of the IBM hereafter.

2.4.3. Empirical solution

[19] performed a detailed analysis of the spatio-temporal resolution of the immersed boundary method with direct forcing and one of their findings is that as the number of lagrangian markers is increased, the largest volume-weight possible must be less than a critical value, obtained to maintain stable solutions for a spherical particle with uniform distribution of markers on the surface. They hence increased the value of the lagrangian weight until stability issues appeared, but we found that this stability criterion does not necessarily lead to the value of W_k computed as in section 2.4.2. This discrepancy may be due to the fact their stability condition is correct only for spherical particles, which is quite restrictive.

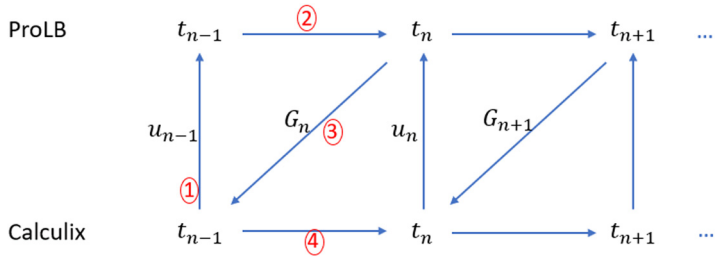

Fig. 2. CSS coupling between ProLB and Calculix.

2.5. Finite element method: predictor-corrector model

For fluid-structure interaction cases, we couple the fluid LBM solver to Calculix [39] which is an open source three-dimensional finite element solver for the structure. The explicit solver is used here with the Hilber-Hughes-Taylor method [40], which is an extension of the Newmark method. It is derived from the implicit-explicit time integration method proposed by Miranda et al. [22] in 1989 which consists in the following steps:

Predictive step:

$$\tilde{\boldsymbol{d}}_{n+1} = \boldsymbol{d}_n + \Delta t \boldsymbol{v}_n + \frac{1}{2}(1-2\beta)\Delta t^2 \boldsymbol{a}_n \tag{29}$$

$$\tilde{\boldsymbol{v}}_{n+1} = \boldsymbol{v}_n + (1-\gamma)\Delta t \boldsymbol{a}_n \tag{30}$$

Time discrete equation of motion:

$$\boldsymbol{M}\boldsymbol{a}_{n+1} + (1+\alpha)\boldsymbol{C}\tilde{\boldsymbol{v}}_{n+1} - \alpha \boldsymbol{C}\boldsymbol{v}_n + (1+\alpha)\boldsymbol{K}\tilde{\boldsymbol{d}}_{n+1} - \alpha \boldsymbol{K}\boldsymbol{d}_{\boldsymbol{n}} = \boldsymbol{G}_{\boldsymbol{n+1}} \tag{31}$$

Corrective step:

$$\boldsymbol{d}_{n+1} = \tilde{\boldsymbol{d}}_{n+1} + \beta \Delta t^2 \boldsymbol{a}_{n+1} \tag{32}$$

$$\boldsymbol{v}_{n+1} = \tilde{\boldsymbol{v}}_{n+1} + \gamma \Delta t \boldsymbol{a}_{n+1} \tag{33}$$

where $\boldsymbol{M}$ is the mass matrix, $\boldsymbol{C}$ is the damping matrix, $\boldsymbol{K}$ is the stiffness matrix, and $\boldsymbol{d}_n$, $\boldsymbol{v}_n$ and $\boldsymbol{a}_n$ are respectively the displacement, velocity and acceleration of the solid nodes. $\boldsymbol{G_n}$ are the fluid forces calculated with the IBM. The parameters α, β and γ control the accuracy and the stability of the method. Second-order accuracy can be achieved if they are set to:

$$\beta = \frac{1}{4}(1-\alpha)^2 \tag{34}$$

$$\gamma = \frac{1}{2} - \alpha \tag{35}$$

$$-\frac{1}{3} \leq \alpha \leq 0 \tag{36}$$

For $\alpha = 0$, the method reduces to the classical Newmark method and for $\alpha = -\frac{1}{3}$ there is maximum high frequency dissipation in the structure. This method proposes a predictive step which is analogous to an explicit step, and a corrective step analogous to an implicit step. By doing so, it combines the advantages of both time integration methods: it achieves the goal of increased stability in the implicit step, and the computational efficiency in the explicit step.

2.6. Explicit coupling algorithm

An explicit coupling between ProLB and Calculix is implemented: at each time step, ProLB sends the forces on each lagrangian marker to Calculix, which in return sends the displacement of the lagrangian markers to ProLB. The algorithm is schematized in Fig. 2 in which G_n is the lagrangian force on each node, and u_n represents the displacement of each lagrangian node. The red numbers refer to the operations performed during the coupling. This type of coupling is called "Conventional Sequential Staggered" algorithm (CSS) and was used for aeroacoustic applications in the 1990's by Piperno and Farhat ([41,42]). There exists a large variety of more sophisticated coupling algorithms in literature, including implicit [43] or semi-implicit [25] coupling algorithms, but for the present case the explicit CSS coupling was sufficient to obtain stable and accurate results.

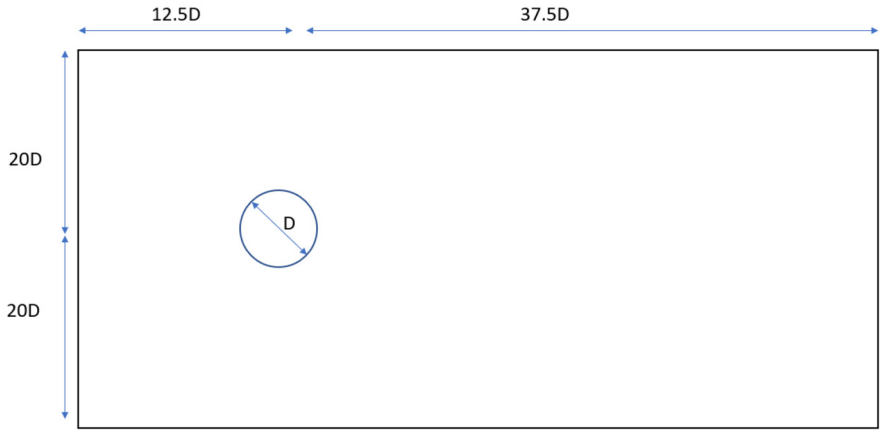

Fig. 3. Computational setup of the 2D cylinder case.

2.7. Turbulence modeling

For the turbulent flow modeling, because of the high computational cost of DNS (Direct Numerical Simulation) and the low fidelity accuracy of RANS (Reynolds Averaged Navier Stokes), LES (Large Eddy Simulation) has become a very popular numerical technique. To account for the unresolved scales of motion in the turbulent case, a subgrid model is used, and more precisely a variant of the Smagorinsky model called the "shear-improved Smagorinsky" proposed in [44]. In this model, the sub grid scale (SGS) Reynolds stress depends on the local strain rate and leads to the eddy viscosity assumption. The eddy viscosity arising from this model is calculated by:

$$\nu_t = (C_S^2 \Delta x)^2 (|S| - \mathcal{S}) \tag{37}$$

with C_S the Smagorinsky constant, Δx the local grid spacing and $|S|$ the norm of the rate of strain tensor. The correcting term $\mathcal{S}$ is the norm of the low-pass filtering (in time) of the rate of strain tensor. This filtering is explained in details in [44] and is independent from the test case. In the LBM, the turbulent viscosity is taken into account in the calculation of the relaxation time τ:

$$\tau = \frac{\nu + \nu_t}{c_s^2} + \frac{1}{2}\Delta t \tag{38}$$

with ν and ν_t the molecular viscosity and turbulent viscosity respectively.

3. Validation of the base LBM-IBM solver

The results presented hereafter will show that the value of the lagrangian weight used to perform the spreading highly impacts the accuracy of the fluid-structure interaction.

To examine the accuracy and reciprocity properties of the different methods, we consider different test cases of increasing complexity.

3.1. Sensitivity analysis of the IBM: effects of Lagrangian & Eulerian grid size, time step, delta function and weight of the Lagrangian markers

The laminar flow around a cylinder at Reynolds number $Re = 100$ is first considered. The computational domain is quasi 2D and is shown on Fig. 3. The mesh is cartesian uniform, D = 0.4 is the cylinder diameter and we perform a grid convergence study on Δx which is the eulerian mesh size. We fix the velocity at the inlet, top and bottom walls, and we fix the density at the outlet.

The L2 error $err_{L2}(F)$ of a quantity F is l by:

$$err_{L2}(F) = \sqrt{\frac{\sum_n (F_{ref}(t_n) - F_{num}(t_n))^2}{\sum_n (F_{ref}(t_n))^2}} \tag{39}$$

where n is the number of considered time-steps, F_{ref} is the reference solution, and F_{num} is the numerically computed solution. A number of authors have studied the flow around a cylinder at Reynolds 100 in the past, and despite its apparent simplicity, there is no clear consensus on the exact value of the mean aerodynamic forces for this test-case, as illustrated

Table 1
Comparison of the results by different studies for a cylinder at Re = 100.

Authors	$\overline{C_d}$	C_l^{max}	St
Braza et al. [45]	1.28	0.29	0.16
Zhou et al. [46]	1.48	0.31	0.162
Bourguet et al. [47]	1.32	0.32	0.164
Gsell et al. [38]	1.37	0.34	0.164

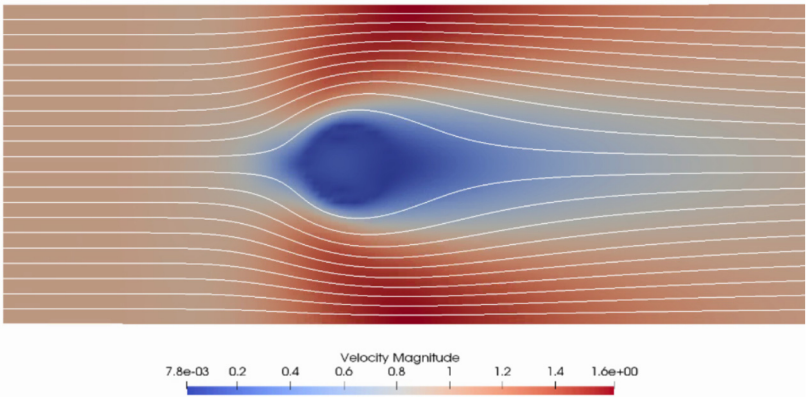

Fig. 4. Instantaneous velocity field and streamlines around the cylinder at Re = 100.

Table 2
Comparison of the accuracy and computational cost of the boundary slip correction methods.

Method	$err_{L2}(u)$	Cpu time
W_k	$1e^{-4}$	1
Bicgstab	$5e^{-5}$	1.6
Inverse matrix	$1e^{-5}$	2.25

in Table 1. The reference value used to compare with our results is taken as the average of the literature results of Table 1: 1.36, 0.315 and 0.162 respectively for drag force, peak lift coefficient and Strouhal number.

An instantaneous velocity field is shown on the Fig. 4. The velocity should be equal to zero on the immersed boundary of the cylinder, so the velocity slip error at the boundary gives an insight on the penetration of the fluid into the solid. In Table 2 which presents the L2 error on the velocity at the boundary, we compare in terms of accuracy and computational time the different methods presented in sections 2.4.1 and 2.4.2 to prevent the boundary slip error, namely the method using a matrix inversion (inverse matrix), the approximated solution using a biconjugate gradient method (bicgstab), and the analytical solution (W_k, see eq (28)). Fig. 5 compares the velocity profiles in the wake of the cylinder located at 0.8D behind the cylinder obtained with the different methods.

The three methods give almost identical results, with less than 1% discrepancy between the velocity profiles. The boundary slip errors are between $1e^{-4}$ and $1e^{-5}$, showing that the penetration of fluid in the solid is negligible. The computational time of the analytical method is the same as the baseline method, whereas the inversion of the matrix is the most expensive method, as expected. Indeed, for this simple 2D test case, the computational cost of the $N \times N$ matrix inversion is more than twice the one of the base method, meaning that for more complex flows and geometries involving a much higher number of lagrangian markers N, the cost will increase drastically. We will now compare the results in terms of mean aerodynamic forces, and frequencies to the reference solution.

3.1.1. Grid and time step convergence study

Following Luo et al. [5] and Zhou et al. [19] for instance, who pointed out that the accuracy of the IBM depends on the eulerian grid resolution, we perform a grid convergence study in Table 3.

With case 15, the results show a 3.5% error on the drag force by decreasing the time step and the grid size sufficiently. This is in accordance with [19] which stated that an extremely high eulerian grid resolution and small time step have to be used to obtain high precision simulation results, and that the boundary slip error on velocity is proportional to the time step when the time step is relatively small. Indeed, as reported in [38], this error scales with the Courant number $C = U_0 \frac{\Delta t}{\Delta x}$. (See Table 4.)

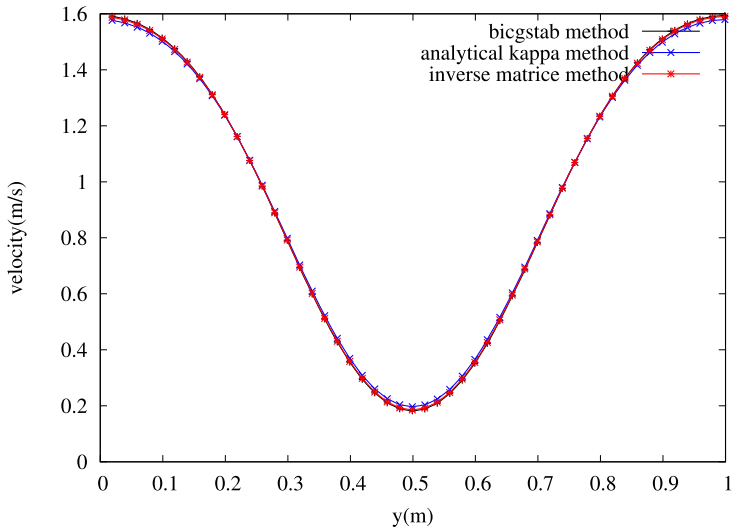

Fig. 5. Velocity profiles along the y axis in the wake of the cylinder.

Table 3
Grid convergence.

Case	Δx	Δs	$\overline{C_d}$	C_l^{max}	St	$err_{L2}(\overline{C_d})$	$err_{L2}(C_l^{max})$	$err_{L2}(St)$
1	0.08	0.0837	2.07	0.6	0.134	0.52	0.9	0.17
2	0.04	0.0628	1.74	0.47	0.156	0.28	0.49	0.037
3	0.02	0.02512	1.56	0.395	0.163	0.15	0.25	0.006
4	0.01	0.01256	1.45	0.36	0.164	0.066	0.14	0.012
15	0.005	0.00628	1.408	0.33	0.166	0.035	0.047	0.02
Ref	-	-	1.36	0.315	0.162			

Table 4
Effect of the time step.

Case	Δx	Δs	dt	$\overline{C_d}$	C_l^{max}	St	$err_{L2}(\overline{C_d})$	$err_{L2}(C_l^{max})$	$err_{L2}(St)$
5	0.04	0.0419	0.00046188	1.68	0.425	0.153	0.24	0.35	0.055
13	0.04	0.0419	0.00015396	1.64	0.425	0.152	0.2	0.35	0.06
6	0.02	0.0209	0.00023094	1.52	0.385	0.165	0.12	0.22	0.018
14	0.02	0.0209	7.698e-05	1.45	0.36	0.162	0.06	0.14	0
Ref	-	-	-	1.36	0.315	0.162			

Table 5
Effect of the type of discrete delta function.

Case	Δx	Δs	δ function	$\overline{C_d}$	C_l^{max}	St	$err_{L2}(\overline{C_d})$	$err_{L2}(C_l^{max})$	$err_{L2}(St)$
5	0.04	0.0419	eq. (13)	1.68	0.425	0.153	0.23	0.35	0.055
7	0.04	0.0419	eq. (14)	1.8	0.5	0.152	0.32	0.59	0.06
8	0.04	0.0419	eq. (15)	1.68	0.42	0.154	0.23	0.33	0.05
Ref	-	-	-	1.36	0.315	0.162			

3.1.2. Discrete delta function

For the case 5 corresponding to an error of 23%, we test three different shapes of discrete delta functions. (See Table 5.)

We find here that a flatter function gives a bigger error on the no-slip condition (hence on the aerodynamic forces). Indeed, a narrow support of the delta function gives a sharper and more accurate description of the boundary. Both equations (13) and (15) yield more accurate results. Some authors [35,36] also choose different delta functions to reduce the grid locking effect, but as already discussed in this paper, the predictive step allows us to alleviate this drawback.

3.1.3. Number of Lagrangian markers

We analyze the effect of the markers density along the immersed boundary, in the general case where they are uniformly distributed (see Table 6). Indeed, a non-uniform distribution of markers can yield a better computational efficiency [15] but it is case-dependent, and thus less applicable in a general framework.

Table 6
Effect of the number of lagrangian markers.

Case	Δx	Δs	Nb markers	$\overline{C_d}$	C_l^{max}	St	$err_{L2}(\overline{C_d})$	$err_{L2}(C_l^{max})$	$err_{L2}(St)$
2	0.04	0.0628	20	1.74	0.47	0.156	0.28	0.49	0.037
5	0.04	0.0419	30	1.68	0.425	0.153	0.24	0.35	0.055
9	0.04	0.03925	32	1.68	0.428	0.153	0.23	0.36	0.055
10	0.04	0.0314	40	1.66	0.416	0.153	0.22	0.32	0.055
3	0.02	0.02512	50	1.56	0.395	0.163	0.15	0.25	0.006
6	0.02	0.0209	60	1.52	0.385	0.165	0.12	0.22	0.018
11	0.02	0.01256	100	1.48	0.395	0.163	0.09	0.25	0.006
4	0.01	0.01245	100	1.45	0.36	0.164	0.066	0.14	0.012
12	0.01	0.00628	200	1.44	0.38	0.165	0.059	0.2	0.018
Ref	-	-	-	1.36	0.315	0.162			

Table 7
Effect of the weight of the lagrangian markers.

Case	Δx	Δs	W_k	$\overline{C_d}$	C_l^{max}	St	$err_{L2}(\overline{C_d})$	$err_{L2}(C_l^{max})$	$err_{L2}(St)$
14	0.02	0.0209	1	1.45	0.36	0.162	0.066	0.14	0
16	0.02	0.0209	2.35	1.42	0.35	0.162	0.044	0.11	0
4	0.01	0.01256	1	1.45	0.36	0.164	0.066	0.14	0.012
20	0.01	0.01256	2.35	1.41	0.345	0.165	0.037	0.095	0.055
Reference	-	-	-	1.36	0.315	0.162			

Table 8
Stability limit with the weight of the lagrangian markers.

Case	Δx	Δs	$\Delta s/\Delta x$	W_k	$\overline{C_d}$	C_l^{max}	St
18	0.04	0.0314	0.785	2.2	x	x	x
19	0.04	0.02512	0.62	1.8	x	x	x

In literature, there is no clear consensus on the optimal lagrangian markers spacing with respect to the eulerian spacing. As mentioned by Su et al. [48], the optimum marker spacing needs to be proportional to the grid spacing, which is also what we report here, but the value of the proportionality coefficient, generally close to 1, varies from one study to the other. In his original paper, Uhlmann [7] suggested that $\Delta s = \Delta x$ by default which means that the Lagrangian points spacing should be equal to the Eulerian grids step. Su et al. [48] and Kang and Hassan [49] adopted $\Delta s = \Delta x = 1/1.5$, while Silva et al. [50] used $\Delta s/\Delta x < 0.9$. Jiang [51] in 2019 indicated that Δs can be further controlled by another parameter, the ratio of $\Delta s/drf$, in which drf is the diffuse range of the regularized delta function. Later on, [19] uses the minimum number of lagrangian markers and the largest time step possible to achieve the best computational efficiency, which leads to a value of the coefficient larger than 1, while Qiuxiang et al. [20] use a markers spacing equal to half of the lattice spacing in order to avoid fluid volume leakage. These differences come from the fact that many versions of the IBM exist, and their sensitivity to the relative spacing of lagrangian markers is different for each formulation. In our case, we find that the maximum number of lagrangian markers allows the best accuracy, to the detriment of the computational efficiency, which is corroborated by the fact that a higher number of lagrangian markers enhances the boundary definition, and therefore the no slip condition. Nevertheless, when applying the lagrangian weight, the number of markers is limited by a stability threshold, meaning that we can not increase the number of markers indefinitely. This is the subject of the next section.

3.1.4. Weight of the Lagrangian markers

In order to find the optimal values of the lagrangian weights, we proceed as follows: we find the value of the lagrangian weight using the new analytical formula (see section 2.4.2, eq (28)) and we compare it to the value obtained with the matrix inversion (see section 2.4.1), knowing that methods should give similar results. If the solid mesh and the fluid lattice are uniform, which is the case here, each lagrangian marker has approximately the same number of surrounding fluid nodes. For this reason, the local value of the lagrangian weight W_k does not vary much, and for the test we can use the same mean value of W_k for all lagrangian markers. The results are presented in Table 7 and 8.

In Table 8, we observe that when the markers are too close from each other, i.e. when $\Delta s/\Delta x < 0.785$ with Δs the lagrangian grid spacing and Δx the eulerian grid spacing, the code eventually crashes. This can be attributed to the fact that when two markers are too close to each other, their respective discrete function gets identical, and thus two lines of the linear system defined in section 2.4.1 are identical, which leads to an ill-defined problem.

This stability criterion is illustrated in Fig. 6 which shows the drag force as a function of time, with and without the weight. Oscillations appear at the beginning of the computation when the lagrangian weight is taken into account, but the oscillations decrease over time so that the final value of the drag force is similar (but different) with and without the lagrangian weight in the end as we can be in Fig. 7. However, if the lagrangian weight is higher than a certain threshold, the stability limit is reached and the computation crashes.

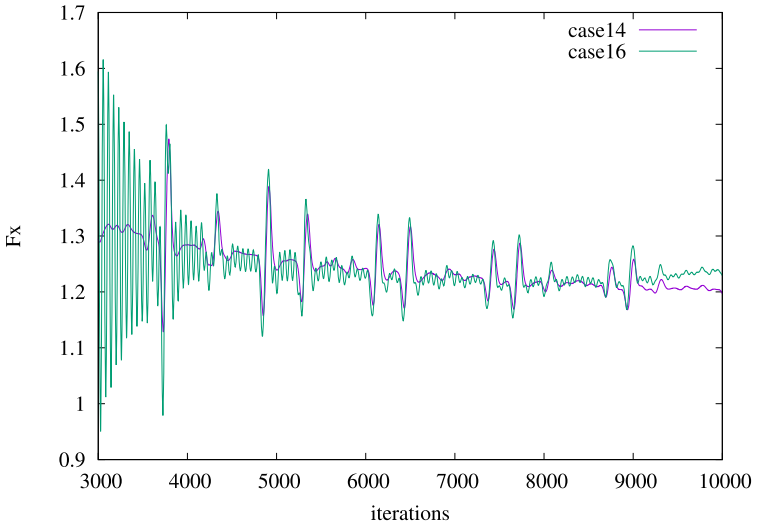

Fig. 6. Drag force as a function of the number of iterations with (green line) and without (purple line) calculation of the lagrangian weight W_k.

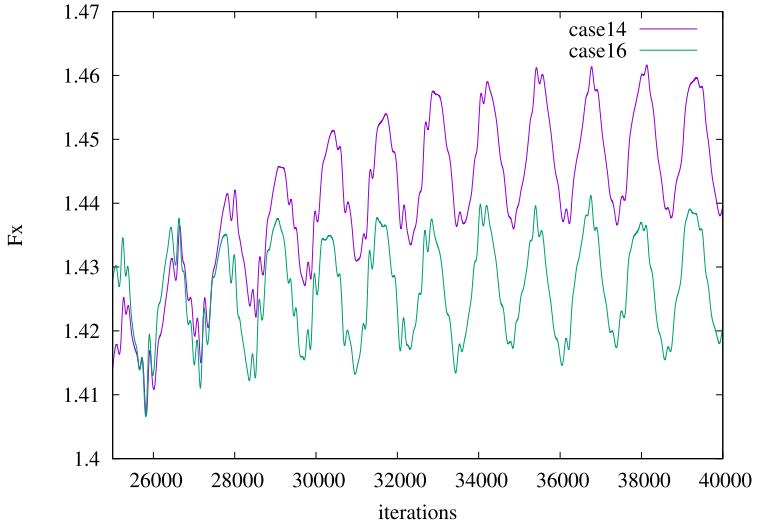

Fig. 7. Drag force w.r.t iterations with (green line) and without (purple line) W_k.

In summary, the most efficient method to compute the lagrangian weight is to use the analytical method (eq. (28)), which leads to a slip error on the boundary of less than a few percent. However, this lagrangian weight must not exceed a certain stability threshold which appears when the lagrangian spacing is decreased to less than 0.785 Δx. Also, unlike what is reported in [19], by comparing cases 4 and 20, we find that the lagrangian weight has a significant influence on the computation of the total forces as it decreases the error by 3%.

3.1.5. Discussion

This section highlights the two cases for which we obtain the best results in terms of accuracy and proposes guidelines on how to use the IBM.

Our findings here are that a high grid resolution is necessary in order to obtain accurate results, as well as a small time step, which are in agreement with the results of [19]. Nevertheless there exists in literature discrepancies in the errors reported by various authors, due to the large diversity of numerical formulations of the IBM encountered. For instance, Luo et al. [5] studied the flow around a cylinder at Re = 100 and showed that the interpolated bounce-back method is more accurate than the immersed boundary method which is a first order method, although interestingly both methods exhibit a second-order rate of convergence in L2-norm in space. More recently in 2020, Zhou et al. [19] studied the flow around a sphere at Re = 100 and found an error of the drag force which decreases as the fluid and solid grid spacing is reduced. Here, we also find that the error scales as the Courant number, and we obtain an error of 3.5% on the drag force with the configuration shown in the Table 9.

Table 9
Optimal parameters for the IBM method applied to the flow around a cylinder at Re = 100.

Case	Δx	Δs	W_k	$\overline{C_d eul}$	$C_l^{max} eul$	St	$err_{L2}(\overline{C_d})$	$err_{L2}(C_l^{max})$	$err_{L2}(St)$
15	0.005	0.00628	1	1.408	0.33	0.166	0.035	0.047	0.02
20	0.01	0.01256	2.35	1.41	0.345	0.165	0.037	0.095	0.055
Reference	-	-	-	1.36	0.315	0.162			

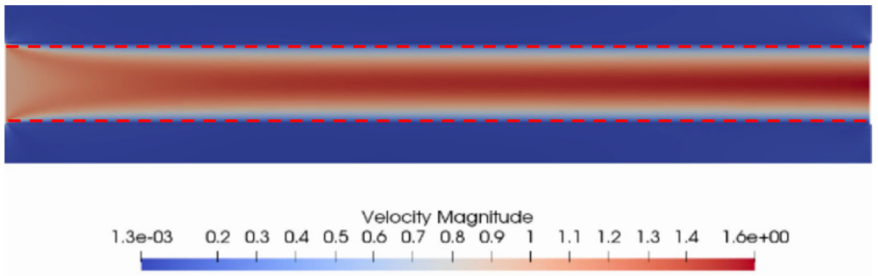

Fig. 8. Contours of fluid velocity in a straight channel at $Re = 100$ simulated with the IBM.

Table 10
Comparison of the local and global W_k.

Δx	Δs	W_k	Q (kg/s)	$err_{L2}(Q)$
0.0025	0.0025	local	0.118298	0.0141
0.0025	0.0025	global	0.118259	0.0145

We also find that a sharper discrete delta function gives better results than a flatter delta function, because the immersed boundary is more accurately captured with a sharp function. As for the numbers of lagrangian markers, our study shows that the optimal configuration is to use a similar fluid and solid grid spacing. In fact, if the number of lagrangian markers is too high, we reach a stability limit and the computation crashes as explained previously. On the other hand, if the number of lagrangian markers is too low, we observe fluid leakage across the boundary due to a poor definition of the boundary. Our best setting here is hence to use a lagrangian grid spacing set as $0.8 < \Delta s/\Delta x < 1.2$ with Δs and Δx the lagrangian and eulerian grid spacings respectively. Finally, the weight of the lagrangian marker can be computed with the analytical expression, whose assumptions are less restrictive than the analytical or empirical approaches presented in section 2.4.2. In their study, Zhou et al. [19] concluded that the lagrangian weight has little impact on the eulerian force, but we showed here that the solution with the weight reduces the error by a few percent. The two cases which gave the most accurate results here are case 15 which is the one with the highest grid resolution, and case 20 which is the case with the analytical solution (eq. (28)) for the computation of the lagrangian weight. We will show in the reminder of the paper that the precise computation of the lagrangian weight is highly important when the fluid is coupled to a moving and deformable structure.

3.2. Poiseuille flow

This second test case focuses on the Poiseuille flow in a straight channel with walls which are simulated with the immersed boundary method. The Reynolds number in the channel is equal to $Re = 100$, so that the flow is laminar and we compare the velocity profile to the theoretical profile obtained from the Poiseuille flow solution:

$$V(r) = V_{max}(1 - \frac{r}{R}) \tag{40}$$

with V_{max} the maximum velocity in middle of the channel, R the channel radius and r the local radius along the profile.

The channel is included in a larger simulation domain, and the top and bottom walls of the channel are simulated with the IBM (in dotted red lines in Fig. 8), whereas periodic conditions are imposed at the top and bottom boundaries of the larger simulation domain. A constant velocity is applied at the inlet, which is why the velocity profile needs to develop in space to reach a parabolic profile at the end of the channel. We chose to use the analytical solution to compute W_k here because of its ease of implementation and computational efficiency. In order to check that using a constant value of W_k on all points is accurate, we plot the values of W_k found with the analytical solution (see section 2.4.2) on all lagrangian points on Fig. 9.

We compare the flow rate Q obtained with a local value of W_k on each lagrangian point (meaning each lagrangian point k is assigned its lagrangian weight W_k), to a global value W used on all points (equal to the mean value of all W_k), and we check the accuracy of the global and local approaches.

The Fig. 9 shows that the value of W obtained with the analytical solution does not vary much along the immersed boundary, and Table 10 shows that the L2 error on the flow rate using the two methods vary by 0.04%. For this reason,

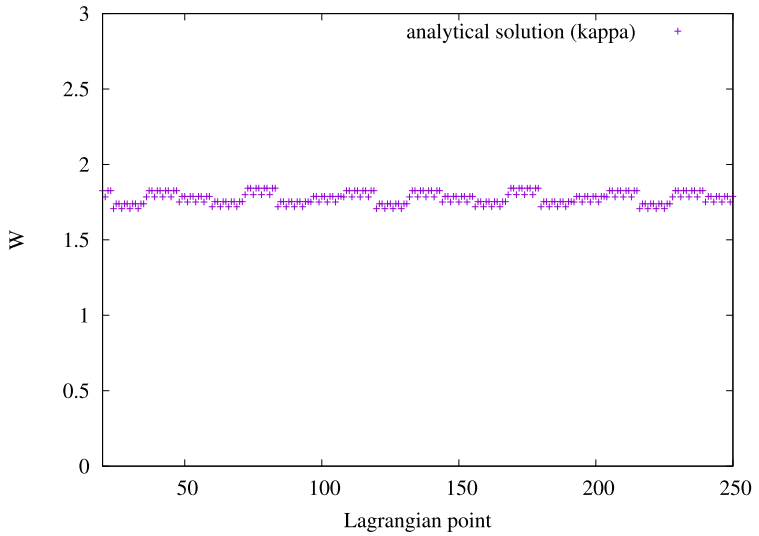

Fig. 9. Local value of W_k on all lagrangian points.

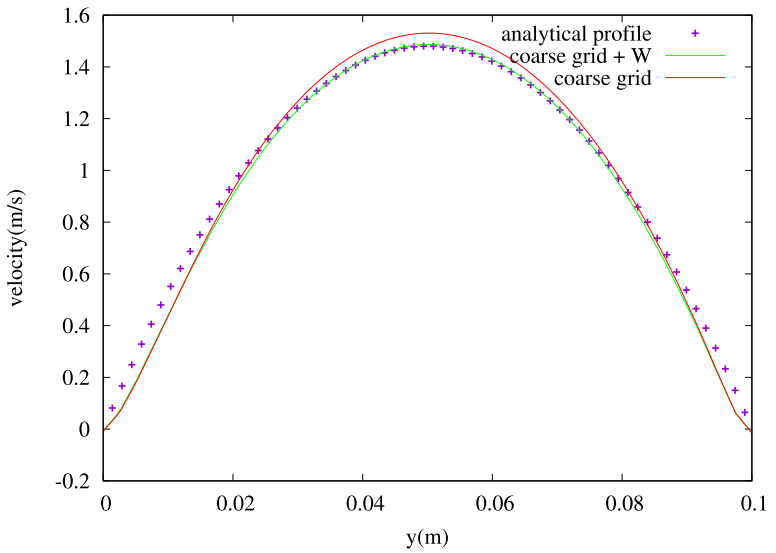

Fig. 10. Velocity profile with and without the lagrangian weight compared to the analytical profile.

we decide to use a constant value of W on all points, equal to the mean value of all lagrangian weights W_k. We compare the results to the analytical profile with and without the lagrangian weight calculated with the analytical solution (see section 2.4.2), and we also perform a grid convergence study.

The Fig. 10 presents the velocity profiles obtained at the middle of the channel for various simulations, intentionally under-resolved to better highlight the errors. We can observe that the simulation with the coarse grid and without the lagrangian weight gives the largest errors on the velocity at the side of the channels as well as in the middle of the channel. When the lagrangian weight is taken into account, the error on the velocity in the middle of the channel is reduced, but the error at the side of the channel is still present. This is due to the fact that the position of the immersed boundary is somehow not known exactly, because of the error of interpolation and smoothing of the velocity field by IBM, as also reported by Seta et al. [14]. Special treatments exist to fix this issue, for instance an implicit method introduced by Seta et al. [14] (see section 1 for more details).

When a finer is used together with a computation of the lagrangian weight W, this error in the near wall region can be reduced to a value of less than 1%, as illustrated in Fig. 11.

The L2 error on the mass flow is summarized in Table 11. We can observe that the correct setting of the lagrangian weight reduces the error on the mass flow by a factor of 3 approximately, and that using a fine mesh with the lagrangian weight gives the smallest error on the velocity.

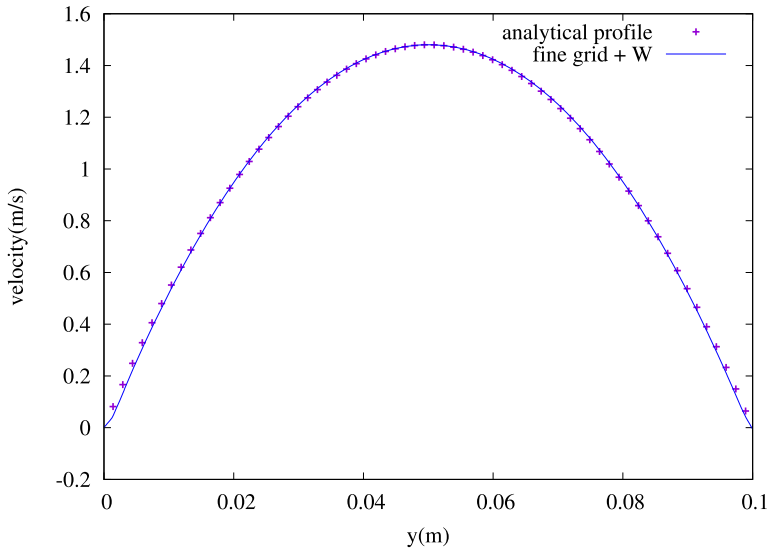

Fig. 11. Grid convergence study on the Poiseuille flow.

Table 11
L2 error on the mass flow with and without the lagrangian weight.

Case	Δx	Δs	W	$err_{L2}(Q)$
1	0.0025	0.0025	1	0.042
2	0.0025	0.0025	2	0.014
3	0.00125	0.00125	2	0.0083

As a final remark, we agree with [19] that while the far field solution is relatively insensitive to the resolution near the immersed boundary, the solution near the boundary is first order with respect to the grid size. Nevertheless, we report the lagrangian weight has a significant effect on the boundary slip error on velocity, even for steady fluid flows. In fact, the lagrangian weight effectively corrects the boundary slip error at the interface, but on the other hand, it does not correct IBM's main flaw which is the velocity gradient at the wall. This is in agreement with Peng's work [37] who found that the sharp fluid- solid interface is diffused by the regularized delta-functions, which results in a smaller velocity gradient near the interface. Indeed, the method assumes that the velocity field and its gradient are continuous across the wall, but the second assumption is generally not verified for typical velocity profiles at the wall. For this reason, the accuracy of the immersed boundary method is highly dependent on a high eulerian grid resolution to capture the near-wall solution (i.e. the velocity gradient near the wall). A very small time step is needed as well because the velocity slip error scales with the Courant number.

3.3. Fluid structure interaction case: influence of the Lagrangian weight on a 3D flapping flag in a uniform flow

The 3D flapping flag is a well known reference benchmark case used for the validation of fluid-structure interaction solvers [52–56]. The laminar flow at Reynolds number $Re = \frac{U_0 L}{\nu}$ = 200 around a square flag of length L in a computational domain of 8 L x 8 L x 2 L is studied. A Dirichlet condition on the velocity is applied at the inlet and upper and lower boundary conditions, and a Dirichlet condition on the density is applied at the outlet and periodic boundary conditions are applied at the lateral surfaces in the spanwise direction. We use a 3D body with thickness h=0.01 L to model the flag, and the minimum lattice spacing is equal to 0.01 L. On the flag, the grid spacing is equal to 0.013 L. The mass ratio $\frac{\rho_s h}{\rho_f L}$ is equal to 1, the Young modulus E is set to 1.008e+6 $\frac{kg}{m^1 s^2}$ and the Poisson ratio is equal to 0.4. The flag is initially set at an angle of 0.1π from the xz plane, the leading edge of the flag is pinned, and the other edges are freely moving. The solid and fluid time step is set to 1e-3s. Gravity is neglected, and we compare the time histories of the aerodynamic forces (the drag and lift profiles), as well as the trailing edge position of the flag, to the literature data.

The Fig. 12 shows the profiles of the lift and drag coefficients, without the lagrangian weight, and the Fig. 13 shows the same profiles but with the lagrangian weight calculated with the analytical solution (see section 2.4.2).

A first comment is that without the lagrangian weight, the aerodynamic profiles present a larger error than with the lagrangian weight, which corroborates our findings that the lagrangian weight is highly important when studying a fluid-structure interaction case. Because the lagrangian weight effectively corrects the error of the unknown quadrature in the lagrangian space, the boundary slip error can be corrected, and the forces sent to the structural solver are correctly com-

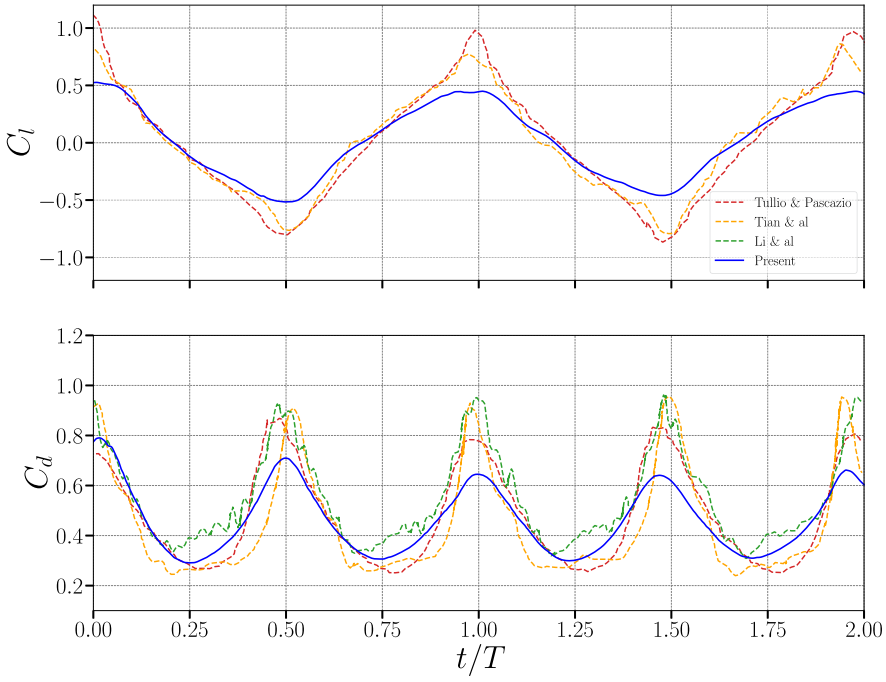

Fig. 12. Aerodynamic forces as a function of time, without W_k.

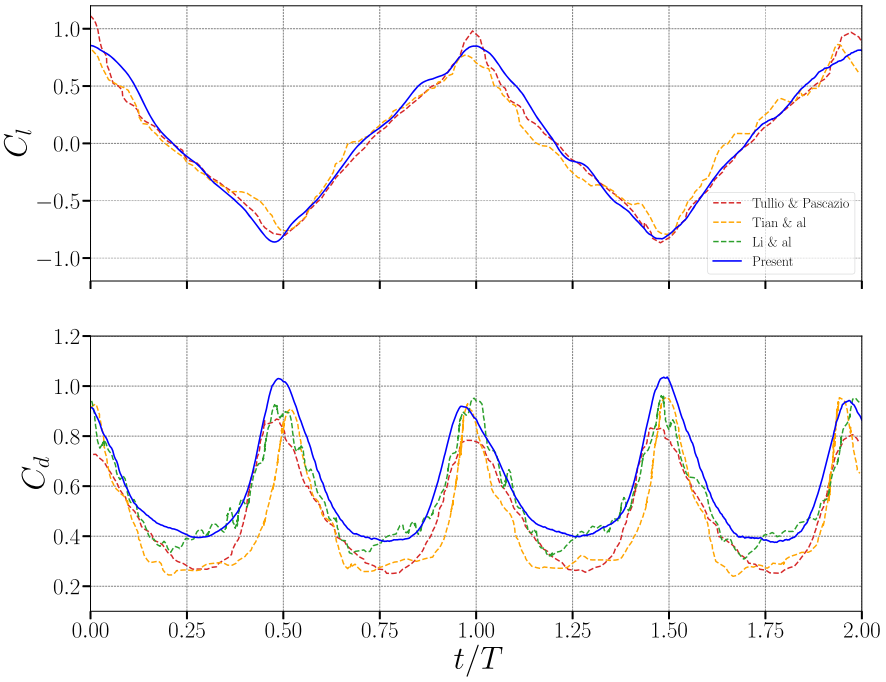

Fig. 13. Aerodynamic forces as a function of time, with W_k.

puted. With the lagrangian weight, the aerodynamic profiles match quite well to the reference data as can be seen in Fig. 13. The drag forces are nevertheless slightly overestimated as has already been reported in our first test case. To decrease further this error, a finer mesh can be used.

A second analysis of these results is that we obtain a lower level of high frequency noises in the aerodynamic forces compared to most previous literature studies [55,53]. This can be attributed to several reasons including the use of an improved collision model for the LBM solver, the HRR model, which is known for its stabilizing properties on the fluid flow alone. Also, the formulation of the IBM we are using here, together with an appropriate computation of the lagrangian

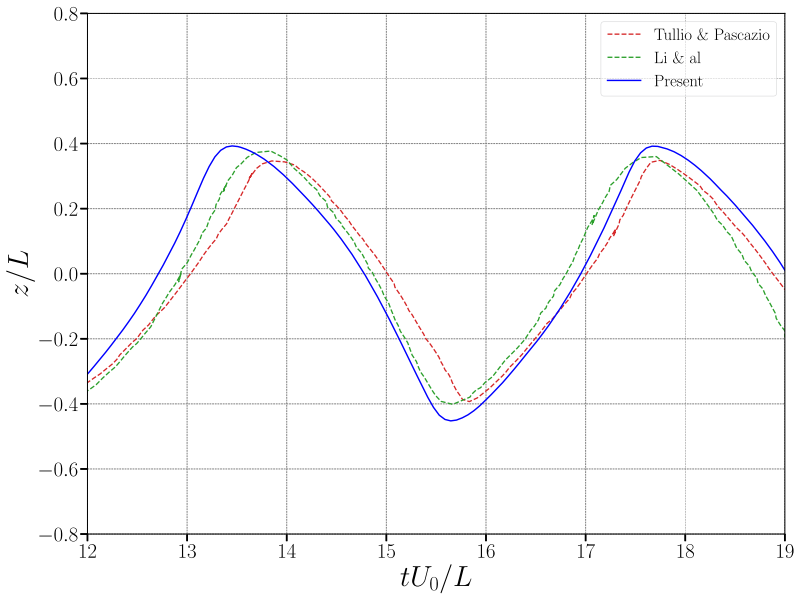

Fig. 14. Trailing edge position w.r.t time.

weight certainly plays a role in the smoothness of the time signals we obtain. A third reason is the stabilizing properties of the Hilber-Hughes-Taylor method used for the structure solver.

The presence of these instabilities in the coupling of fluid and structure schemes is a well-known issue which has been investigated by numerous authors in the past [57–59]. Although several implicit coupling methods are employed in literature to overcome this problem [43,25], here the use of a relatively simple weak coupling (defined in section 2.6) between the fluid and structure solvers is nevertheless sufficient to obtain robust and accurate results.

Fig. 14 shows the time history of the trailing edge position of the flag compared the reference.

We observe an excellent agreement with literature, both in terms of flapping frequency and amplitude. We can also observe in Fig. 15 the Q criterion around the flapping flag at time $t = 16.2s$ showing the characteristic hairpin-like structure shed at each flapping [56]. The flow pattern seems qualitatively consistent with the results obtained by Huang et al. [52], with the formation of the structures at the flag side edges which are then shed by the trailing edge. The vortex shed from the trailing edge and the vortices from the side edges are joined together to form a hairpin-like structure similar to the one obtained by [52] and later on by Tian et al. [53].

To summarize here, the lagrangian weight corrects the error inherently introduced in the quadrature of the lagrangian space, and produces more accurate results. Its importance is particularly highlighted when the fluid is coupled to the structure. We also obtain very satisfactory results in terms of robustness, as the time instabilities usually encountered in fluid structure interaction are reduced. This is due to the stabilizing properties of the fluid, structure, and IBM solver formulations we are using here. Thus, the weak coupling used here is found to be robust enough to be used in a 3D complex model. The computational efficiency of a weak coupling is highly interesting compared to a strong coupling which often needs implicit correction and subcycling iterations in order to converge.

3.4. Turbulent fluid structure interaction case: elastic beam attached to a fixed cylinder at Re = 10 000

This last test case deals with a high Reynolds number turbulent flow around a fixed cylinder to which is attached a flexible beam as can be seen on Fig. 16. The Reynolds number is 10 000, meaning that the boundary layers at the cylinder separate laminarly and large vortices are observed in the wake, originating from the shedding process as shown on Fig. 17.

This test case follows the work of [60], and the dimensions are the following: L = 2.5 m, H = 0.41 m, r = 0.05 m, l = 0.35 m, h = 0.02 m. The coordinates of point C are (0.2 m;0.2 m). There are 40 nodes in the Z-direction in order to create a 3D geometry to observe the turbulent behavior. We consider a fluid-to-density ratio of 1, meaning that the added mass effect is important and the explicit coupling is numerically challenged with this configuration. Indeed, when a solid body moves through a fluid or when a fluid flows around a solid structure, the fluid particles in the vicinity of the body or structure are set in motion. As a result, the fluid exerts forces on the solid, and these forces contribute to the effective mass experienced by the structure. The added mass effect hence refers to the phenomenon where a fluid exerts an additional mass on a solid structure and it is particularly significant when the fluid density is comparable to the solid density which is the case here. Explicit coupling schemes have been known to struggle with high added-mass effect as pointed out by [23] and [61]. Nevertheless the methodology proposed here effectively deals with this added mass effect as well as with the numerical instabilities that can be produced because of the time lag inherently present in the explicit (or weak) coupling.

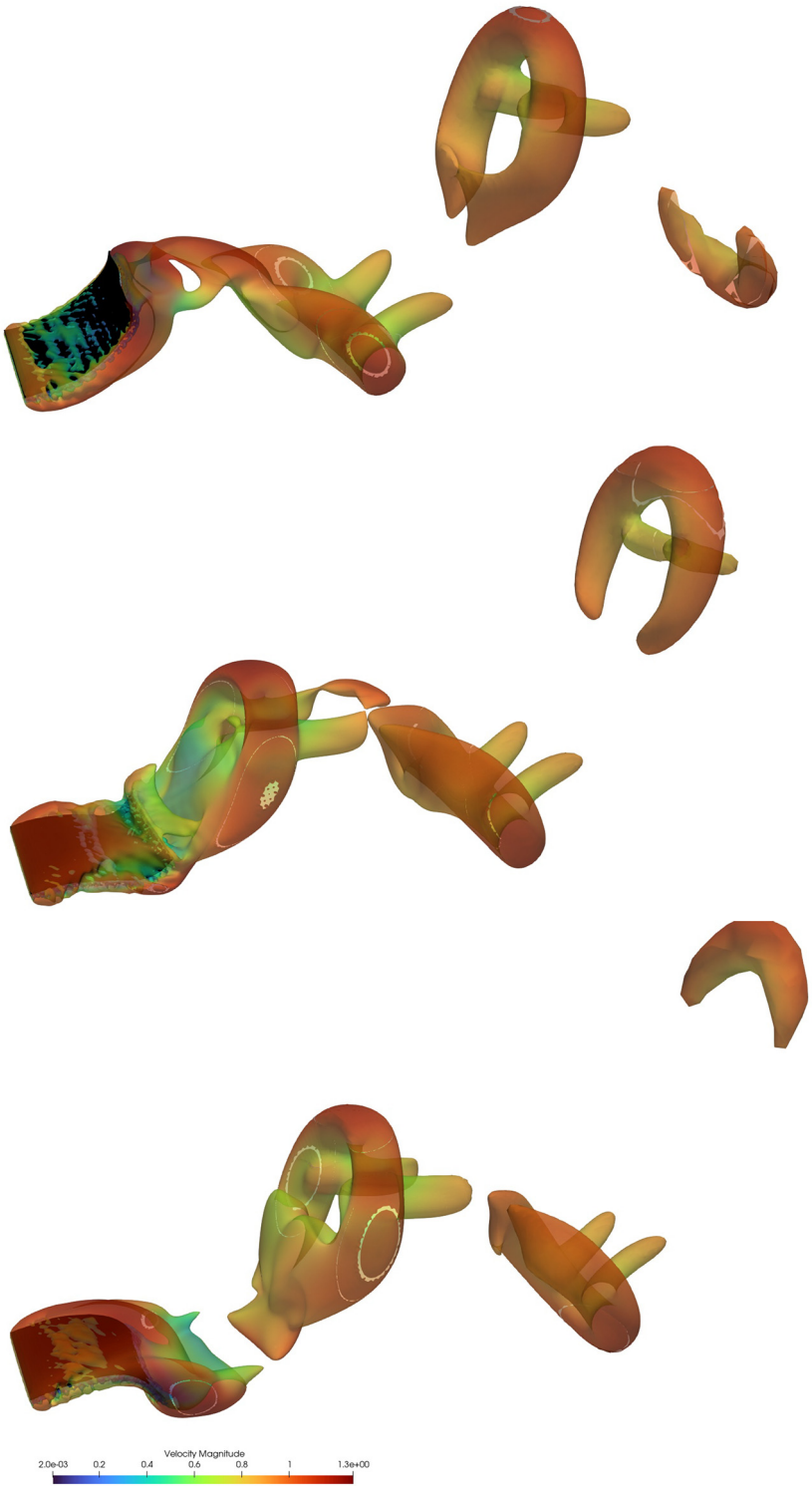

Fig. 15. Iso-surface of Q-criterion colored by velocity magnitude.

Different prediction schemes are compared for the prediction in time of the velocity of the solid points, namely a first order and a second order scheme. Here, we will show how this affects the numerical instabilities, and compare our results to [25]. The physical and numerical parameters are summarized in Table 12 and the lagrangian weight is equal to 2 for this case.

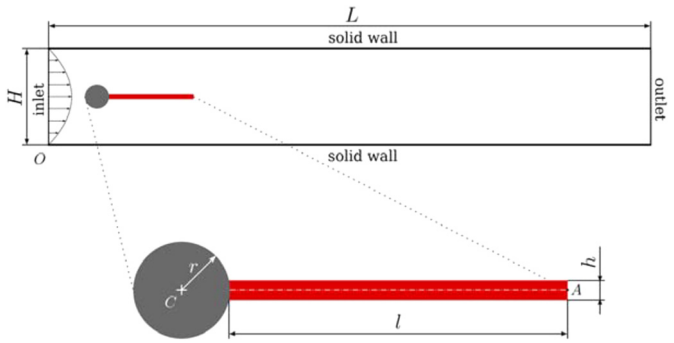

Fig. 16. Setup of the case.

Table 12
Physical and numerical parameters of the 3D turbulent deformable test case.

	$Re = 10000$
Fluid density (ρ_f)	10^3 kg/m^3
Fluid kinematic viscosity (ν_f)	2×10^{-5} m^2/s
Solid density (ρ_s)	10^4 kg/m^3
Solid Poisson ratio (ν_s)	0.4
Solid Young's modulus (E_s)	5.6×10^6 Pa
Solid-to-fluid density ratio (ρ_s/ρ_f)	1
Fluid velocity $(\bar{U})$	2 m/s
Mesh size $(dx_s/dx_f = 1)$	0.002 m
Time step (dt)	1.15×10^4 s

Table 13
Strouhal numbers compared to the reference.

	Present solver	Breuer 2012 [25]
Primary Strouhal number	0.185	0.175
Secondary Strouhal number	0.101	0.103

Fig. 17 shows the Q criterions in the flow on a x-y plane colored by the velocity magnitude. Large vortices are created from the shedding process and transition to turbulence occurs in the free shear layers. As pointed out in Breuer et al. [25], this shedding process is attributed to Kelvin-Helmhotz instabilities and a transition to turbulence in the free shear layers. For the turbulence modeling, we use a Large Eddy Simulation model based on the Smagorinsky model (see section 2.7) which is adapted to the separated shear layer turbulence modeling.

The lift signal is depicted on Fig. 18 and compared to [25]. Because of the turbulent behavior of the case, a perfect superimposition of the signals is not expected. Nevertheless, we obtain a similar order of magnitude between the two signals. To more precisely compare our results, we perform a Fast Fourier Transform (FFT) to obtain the main frequencies of the lift signal.

The weak coupling is known to induce numerical instabilities in the coupling because of the time lag introduced between the fluid solver and the solid solver. This is for example noticeable in [25] in the time history of the lift coefficient. To enhance the stability of the coupling, Piperno and Farhat [57,41] proposed in the 2000s several prediction schemes for the advancement in time of the structural points to counter the time lag introduced by the explicit coupling. Based on this idea, we compare here two prediction schemes for the advancement in time of the velocity of the solid points. The first one is a first order scheme in which the velocity is very simply retrieved by $v_n = \frac{d_{n+1} - d_n}{\Delta t}$ and the second one is a second order scheme presented in section 2.5 and which comes from a predictor-corrector model. We compare the FFT of the time history of the lift force for both models in Fig. 19.

Two main frequencies are found in Fig. 19 whose corresponding Strouhal numbers are given in Table 13. A first remark is that the FFT of the 2nd order model is less noisy than the 1st order model as expected, meaning that the numerical noise is decreased with the predictor-corrector model. The weak coupling proposed here combined to the predictor corrector model for the structure can therefore be used for challenging cases with high added mass effect and high Reynolds number flows.

It is also interesting to notice in Table 13 that we obtain a primary frequency leading to a Strouhal number of 0.185 compared to 0.175 for [25] for their fixed case, and a secondary frequency leading to St = 0.101 compared to 0.103 for [25] for their oscillating case. Similarly to our previous work [6], we observe that the primary frequency is linked to the natural vortex shedding frequency observed on a fixed cylinder at certain Reynolds numbers, and the secondary frequency is related to the oscillating motion of the beam. The difference with [6] is that we now include a finite element solver

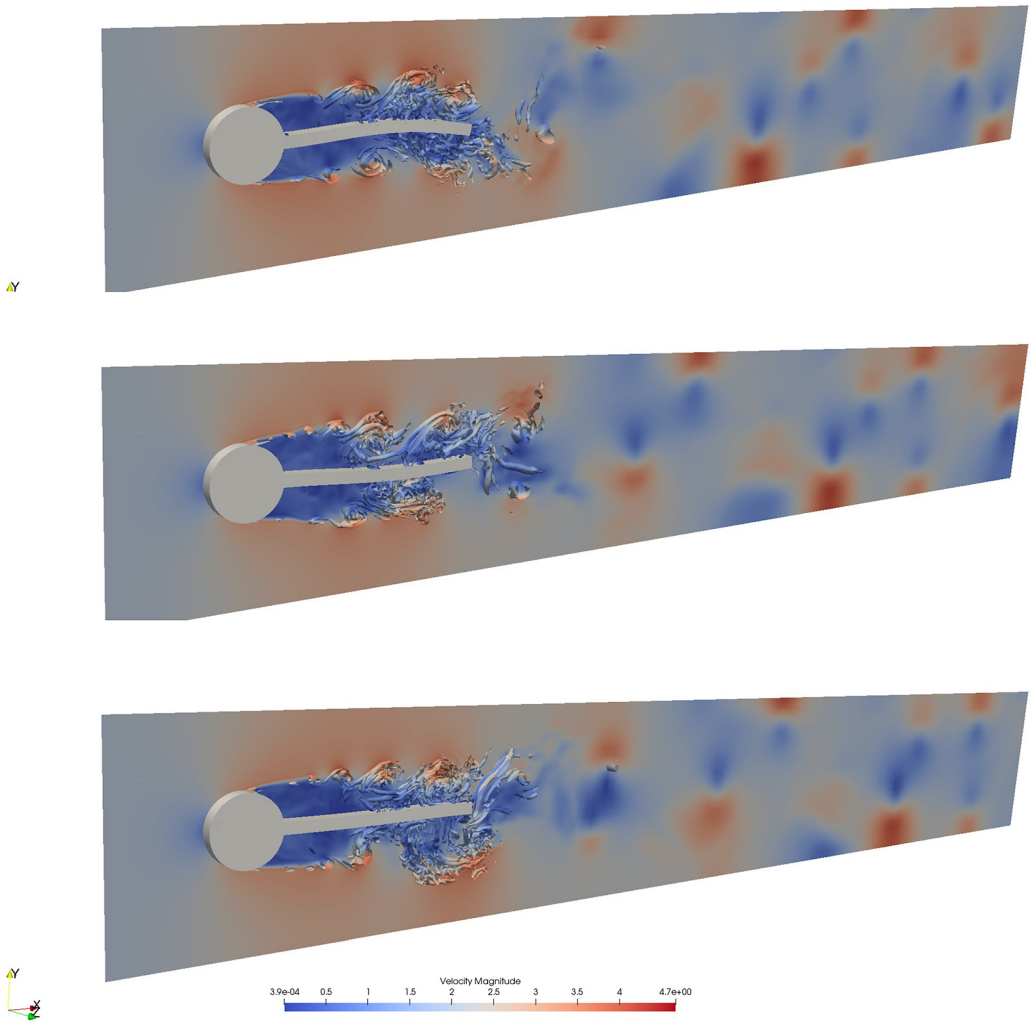

Fig. 17. Iso-surface of Q-criterion colored by velocity magnitude.

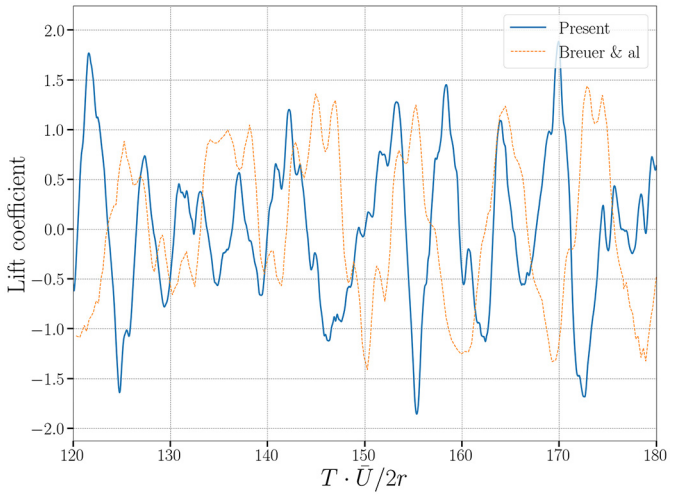

Fig. 18. Lift signal.

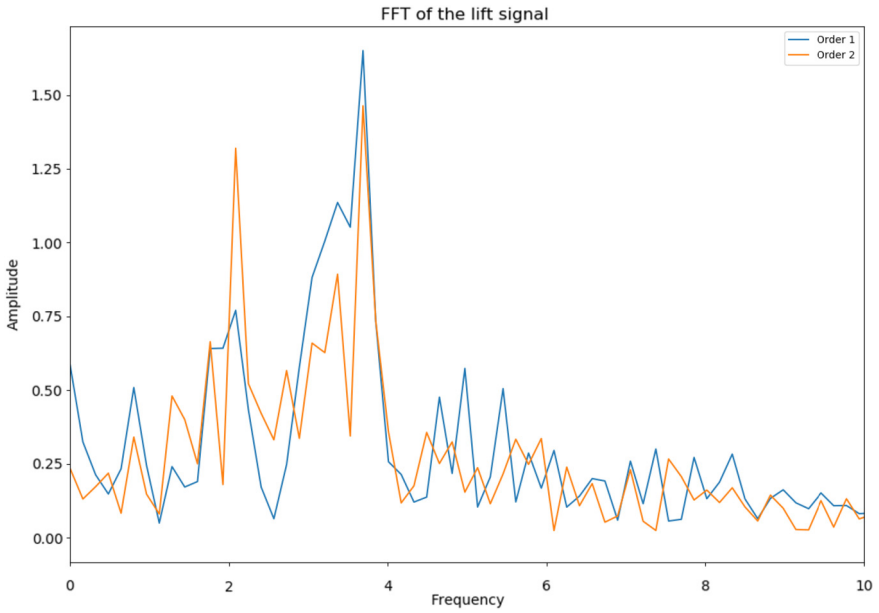

Fig. 19. FFT of the lift signal.

making it possible to study deformable solids, contrarily to [6] which was restricted to forced movements of rigid structures such as translating or oscillating cylinders. The lock-in phenomenon which happens when the vortex shedding frequency is close to the natural frequency of vibration of a structure could be interesting to study for future works as it can lead to large and damaging vibrations.

4. Conclusions

In this work, we analyze the effect of the lagrangian weight of the immersed boundary method in the context of fluid-structure interaction configurations. We also compare two prediction schemes for the structural displacement to counteract the time lag introduced in the explicit coupling. The second order predictor corrector scheme is found to be stable and accurate and is coupled to a LES turbulence model for our last application on a high-Reynolds number flows around a moving object with a high-added mass effect. Turbulence is thus taken into account in fluid structure interaction, which has rarely been studied so far [24].

We use different methods to compute the lagrangian weight, namely a matrix inversion, an approximated solution using a biconjugate gradient method and a new analytical solution. Relevant test-cases are proposed, from fixed boundaries to a two-way fluid-structure interaction test case to assess the influence of the lagrangian weight on large range of configurations. Several findings can be drawn from this study:

- A new theoretical development of the lagrangian weight is proposed, whose implementation is straightforward and with the advantage of being independent of the mesh size and the delta function.
- The lagrangian weight corrects the error inherently introduced in the quadrature of the lagrangian space, and produces more accurate results. Its importance is particularly highlighted when the fluid is coupled to the structure for which the error is decreased by 20%-30%.
- The second order predictor corrector model for the structural displacement can be used with the explicit coupling for challenging cases such as high-Reynolds number flows and high added-mass effect. The advantage of the explicit coupling (i.e. computational efficiency) can therefore be preserved and used for complex cases generally demanding implicit coupling for stability reasons.
- The coupling of a LES turbulence model to the FSI framework is validated with the explicit coupling proposed here.

This work provides insight and useful guidelines for the use of the immersed boundary method. The novelty is to propose a detailed analysis of the immersed boundary method in the context of fluid structure interaction, and to couple a predictor-corrector model to the Smagorinsky model for complex cases involving high Reynolds number flows and high added-mass effect for which the fluid-to-solid density ratio is equal to 1. Future works include the application to human aortic valve configurations for which turbulence is important and the density ratio is close to 1.

CRediT authorship contribution statement

Isabelle Cheylan: Methodology, Software & Writing. **Tom Fringand:** Software, Investigation & Visualization. **Jérôme Jacob:** Software, Resources & Validation. **Julien Favier:** Conceptualization, Supervision & Writing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The data that support the findings of this study are available from the corresponding author upon reasonable request.

Acknowledgements

Part of this research was supported by ANR, Renault, Airbus and SafranTech by the Industrial Chair Program ALBUMS (ANR-CHIND-18-ALBUMS). Centre de Calcul Intensif d'Aix-Marseille is acknowledged for granting access to its high performance computing resources. Simon Gsell and Iko Midani are acknowledged for fruitful discussions on the article.

References

[1] C. Peskin, Flow patterns around heart valves: a numerical method, J. Comput. Phys. 10 (2) (Oct. 1972) 252–271, https://doi.org/10.1016/0021-9991(72)90065-4.
[2] Z.-G. Feng, E.E. Michaelides, The immersed boundary-lattice Boltzmann method for solving fluid–particles interaction problems, J. Comput. Phys. 195 (2) (2004) 602–628, https://doi.org/10.1016/j.jcp.2003.10.013.
[3] M. Ma, W.X. Huang, C.X. Xu, A dynamic wall model for large eddy simulation of turbulent flow over complex moving boundaries based on the immersed boundary method, Phys. Fluids 31 (2019) 115101, https://doi.org/10.1063/1.5126853.
[4] M. Pepona, J. Favier, A coupled immersed boundary – lattice Boltzmann method for incompressible flows through moving porous media, J. Comput. Phys. 321 (2016) 1170–1184, https://doi.org/10.1016/j.jcp.2016.06.026.
[5] Y. Peng, L.-S. Luo, A comparative study of immersed-boundary and interpolated bounce-back methods in LBE, Prog. Comput. Fluid Dyn. 8 (2008) 156–167, https://doi.org/10.1504/PCFD.2008.018086.
[6] I. Cheylan, J. Favier, P. Sagaut, Immersed boundary conditions for moving objects in turbulent flows with the lattice-Boltzmann method, Phys. Fluids 33 (2021) 095101, https://doi.org/10.1063/5.0062575.
[7] Markus Uhlmann, An immersed boundary method with direct forcing for the simulation of particulate flows, J. Comput. Phys. (ISSN 0021-9991) 209 (2) (2005) 448–476, https://doi.org/10.1016/j.jcp.2005.03.017, https://www.sciencedirect.com/science/article/pii/S0021999105001385.
[8] Kun Luo, et al., Full-scale solutions to particle-laden flows: multidirect forcing and immersed boundary method, Phys. Rev. E 76 (Dec. 2007) 066709, https://doi.org/10.1103/PhysRevE.76.066709, https://link.aps.org/doi/10.1103/PhysRevE.76.066709.
[9] J. Wu, C. Shu, Particulate flow simulation via a boundary condition-enforced immersed boundary-lattice Boltzmann scheme, Commun. Comput. Phys. (ISSN 1991-7120) 7 (4) (2010) 793–812, https://doi.org/10.4208/cicp.2009.09.054, http://global-sci.org/intro/article_detail/cicp/7655.html.
[10] A. Pinelli, et al., Immersed-boundary methods for general finite-difference and finite-volume Navier–Stokes solvers, J. Comput. Phys. 229 (24) (2010) 9073–9091, https://doi.org/10.1016/j.jcp.2010.08.021.
[11] J. Favier, A. Revell, A. Pinelli, A lattice Boltzmann–immersed boundary method to simulate the fluid interaction with moving and slender flexible objects, J. Comput. Phys. 261 (2014) 145–161, https://doi.org/10.1016/j.jcp.2013.12.052.
[12] M. Vanella, E. Balaras, A moving-least-squares reconstruction for embedded-boundary formulations, J. Comput. Phys. (ISSN 0021-9991) 228 (18) (2009) 6617–6628, https://doi.org/10.1016/j.jcp.2009.06.003.
[13] T. Kempe, J. Fröhlich, An improved immersed boundary method with direct forcing for the simulation of particle laden flows, J. Comput. Phys. 231 (9) (2012) 3663–3684, https://doi.org/10.1016/j.jcp.2012.01.021.
[14] T. Seta, et al., Implicit-correction-based immersed boundary–lattice Boltzmann method with two relaxation times, Phys. Rev. E 89 (Feb. 2014) 023307, https://doi.org/10.1103/PhysRevE.89.023307.
[15] G. Akiki, S. Balachandar, Immersed boundary method with non-uniform distribution of Lagrangian markers for a non-uniform Eulerian mesh, J. Comput. Phys. 307 (2016) 34–59, https://doi.org/10.1016/j.jcp.2015.11.019.
[16] Shi Tao, et al., A non-iterative direct-forcing immersed boundary method for thermal discrete unified gas kinetic scheme with Dirichlet boundary conditions, Int. J. Heat Mass Transf. (ISSN 0017-9310) 137 (2019) 476–488, https://doi.org/10.1016/j.ijheatmasstransfer.2019.03.147, https://www.sciencedirect.com/science/article/pii/S0017931018358034.
[17] Cheng Peng, Lian-Ping Wang, Force-amplified, single-sided diffused-interface immersed boundary kernel for correct local velocity gradient computation and accurate no-slip boundary enforcement, Phys. Rev. E 101 (May 2020) 053305, https://doi.org/10.1103/PhysRevE.101.053305, https://link.aps.org/doi/10.1103/PhysRevE.101.053305.
[18] S. Gsell, U. D'Ortona, J. Favier, Explicit and viscosity-independent immersed-boundary scheme for the lattice Boltzmann method, Phys. Rev. E 100 (3) (Sept. 2019), https://doi.org/10.1103/PhysRevE.100.033306.
[19] K. Zhou, S. Balachandar, An analysis of the spatio-temporal resolution of the immersed boundary method with direct forcing, J. Comput. Phys. 424 (2021) 109862, https://doi.org/10.1016/j.jcp.2020.109862.
[20] Q. Huang, et al., Streamline penetration, velocity error and consequences of the feedback immersed boundary method, Phys. Fluids (2022), https://doi.org/10.1063/5.0101584.
[21] Serge Piperno, Charbel Farhat, Design and Evaluation of Staggered Partitioned Procedures for Fluid-Structure Interaction Simulations, Tech. Rep. RR-3241, INRIA, Sept. 1997, https://inria.hal.science/inria-00073448.
[22] Isidoro Miranda, Robert M. Ferencz, Thomas J.R. Hughes, An improved implicit-explicit time integration method for structural dynamics, Earthq. Eng. Struct. Dyn. 18 (5) (1989) 643–653, https://doi.org/10.1002/eqe.4290180505, https://onlinelibrary.wiley.com/doi/abs/10.1002/eqe.4290180505.
[23] P. Causin, J.F. Gerbeau, F. Nobile, Added-mass effect in the design of partitioned algorithms for fluid–structure problems, Comput. Methods Appl. Mech. Eng. (ISSN 0045-7825) 194 (42) (2005) 4506–4527, https://doi.org/10.1016/j.cma.2004.12.005.

[24] Li Wang, Zhengliang Liu, Methma Rajamuni, Recent progress of lattice Boltzmann method and its applications in fluid-structure interaction, Proc. Inst. Mech. Eng., Part C, J. Mech. Eng. Sci. 237 (11) (2023) 2461–2484, https://doi.org/10.1177/09544062221077583.
[25] M. Breuer, et al., Fluid–structure interaction using a partitioned semi-implicit predictor–corrector coupling scheme for the application of large-eddy simulation, J. Fluids Struct. 29 (2012) 107–130, https://doi.org/10.1016/j.jfluidstructs.2011.09.003.
[26] J. Jacob, O. Malaspinas, P. Sagaut, A new hybrid recursive regularised Bhatnagar–Gross–Krook collision model for lattice Boltzmann method-based large eddy simulation, J. Turbul. 19 (11–12) (2018) 1051–1076, https://doi.org/10.1080/14685248.2018.1540879.
[27] J. Jacob, P. Sagaut, Wind comfort assessment by means of large eddy simulation with lattice Boltzmann method in full scale city area, Build. Environ. 139 (2018) 110–124, https://doi.org/10.1016/j.buildenv.2018.05.015.
[28] L. Merlier, J. Jacob, P. Sagaut, Lattice-Boltzmann large-eddy simulation of pollutant dispersion in complex urban environment with dense gas effects: model evaluation and flow analysis, Build. Environ. 148 (2019) 634–652, https://doi.org/10.1016/j.buildenv.2018.11.009.
[29] Y. Feng, et al., Hybrid recursive regularized lattice Boltzmann simulation of humid air with application to meteorological flows, Phys. Rev. E 100 (2) (2019) 023304, https://doi.org/10.1103/PhysRevE.100.023304.
[30] Y. Feng, et al., Hybrid recursive regularized thermal lattice Boltzmann model for high subsonic compressible flows, J. Comput. Phys. 394 (2019) 82–99, https://doi.org/10.1016/j.jcp.2019.05.031.
[31] S. Guo, et al., An efficient lattice Boltzmann method for compressible aerodynamics on D3Q19 lattice, J. Comput. Phys. 418 (2020) 109570, https://doi.org/10.1016/j.jcp.2020.109570.
[32] Z. Guo, C. Zheng, B. Shi, Discrete lattice effects on the forcing term in the lattice Boltzmann method, Phys. Rev. E 65 (Apr. 2002) 046308, https://doi.org/10.1103/PhysRevE.65.046308.
[33] Z. Li, et al., An immersed boundary-lattice Boltzmann method for single and multi-component fluid flows, J. Comput. Phys. 304 (2016) 422–440, https://doi.org/10.1016/j.jcp.2015.10.026.
[34] A. Roma, C. Peskin, M. Berger, An adaptive version of the immersed boundary method, J. Comput. Phys. 153 (2) (1999) 509–534, https://doi.org/10.1006/jcph.1999.6293.
[35] Wim-Paul Breugem, A second-order accurate immersed boundary method for fully resolved simulations of particle-laden flows, J. Comput. Phys. (ISSN 0021-9991) 231 (13) (2012) 4469–4498, https://doi.org/10.1016/j.jcp.2012.02.026, https://www.sciencedirect.com/science/article/pii/S0021999112001374.
[36] Xiaolei Yang, et al., A smoothing technique for discrete delta functions with application to immersed boundary method in moving boundary simulations, J. Comput. Phys. (ISSN 0021-9991) 228 (20) (2009) 7821–7836, https://doi.org/10.1016/j.jcp.2009.07.023, https://www.sciencedirect.com/science/article/pii/S0021999109004136.
[37] Cheng Peng, Orlando M. Ayala, Lian-Ping Wang, A comparative study of immersed boundary method and interpolated bounce-back scheme for no-slip boundary treatment in the lattice Boltzmann method: part I, laminar flows, Comput. Fluids (ISSN 0045-7930) 192 (2019) 104233, https://doi.org/10.1016/j.compfluid.2019.06.032, https://www.sciencedirect.com/science/article/pii/S0045793019301987.
[38] S. Gsell, J. Favier, Direct-forcing immersed-boundary method: a simple correction preventing boundary slip error, J. Comput. Phys. 435 (Mar. 2021) 110265, https://doi.org/10.1016/j.jcp.2021.110265.
[39] G. Dhondt, The Finite Element Method for Three-Dimensional Thermomechanical Applications, John Wiley & Sons, 2004.
[40] H. Hilber, T. Hughes, R. Taylor, Improved numerical dissipation for time integration algorithms in structural dynamics, Earthq. Eng. Struct. Dyn. 5 (3) (1977) 283–292, https://doi.org/10.1002/eqe.4290050306.
[41] C. Farhat, M. Lesoinne, Two efficient staggered algorithms for the serial and parallel solution of three-dimensional nonlinear transient aeroelastic problems, Comput. Methods Appl. Mech. Eng. 182 (3) (2000) 499–515, https://doi.org/10.1016/S0045-7825(99)00206-6.
[42] S. Piperno, C. Farhat, B. Larrouturou, Partitioned procedures for the transient solution of coupled aroelastic problems part I: model problem, theory and two-dimensional application, Comput. Methods Appl. Mech. Eng. 124 (1) (1995) 79–112, https://doi.org/10.1016/0045-7825(95)92707-9.
[43] C. Kassiotis, et al., Nonlinear fluid-structure interaction problem. Part I: implicit partitioned algorithm, nonlinear stability proof and validation examples, Comput. Mech. 47 (Mar. 2011) 305–323, https://doi.org/10.1007/s00466-010-0545-6.
[44] E. Leveque, et al., Shear-improved Smagorinsky model for large-eddy simulation of wall-bounded turbulent flows, J. Fluid Mech. 570 (2007) 491–502, https://doi.org/10.1017/S0022112006003429.
[45] M. Braza, P. Chassaing, H. Ha Minh, Numerical study and physical analysis of the pressure and velocity fields in the near wake of a circular cylinder, J. Fluid Mech. 165 (1986) 79–130, https://doi.org/10.1017/S0022112086003014.
[46] C.-Y. Zhou, R.-M. So, K. Lam, Vortex-induced vibrations of an elastic circular cylinder, J. Fluids Struct. (ISSN 0889-9746) 13 (2) (1999) 165–189, https://doi.org/10.1006/jfls.1998.0195.
[47] R. Bourguet, D. Lo Jacono, Flow-induced vibrations of a rotating cylinder, J. Fluid Mech. 740 (2014) 342–380, https://doi.org/10.1017/jfm.2013.665.
[48] S.-W. Su, M.-C. Lai, C.-A. Lin, An immersed boundary technique for simulating complex flows with rigid boundary, Comput. Fluids 36 (2) (2007) 313–324, https://doi.org/10.1016/j.compfluid.2005.09.004.
[49] Shin K. Kang, Yassin A. Hassan, A comparative study of direct-forcing immersed boundary-lattice Boltzmann methods for stationary complex boundaries, Int. J. Numer. Methods Fluids 66 (9) (2011) 1132–1158, https://doi.org/10.1002/fld.2304, https://onlinelibrary.wiley.com/doi/abs/10.1002/fld.2304.
[50] A.L.F. Lima, E. Silva, A. Silveira-Neto, J.J.R. Damasceno, Numerical simulation of two-dimensional flows over a circular cylinder using the immersed boundary method, J. Comput. Phys. (ISSN 0021-9991) 189 (2) (2003) 351–370, https://doi.org/10.1016/S0021-9991(03)00214-6, https://www.sciencedirect.com/science/article/pii/S0021999103002146.
[51] Maoqiang Jiang, Zhaohui Liu, A boundary thickening-based direct forcing immersed boundary method for fully resolved simulation of particle-laden flows, J. Comput. Phys. (ISSN 0021-9991) 390 (2019) 203–231, https://doi.org/10.1016/j.jcp.2019.03.047, https://www.sciencedirect.com/science/article/pii/S002199911930230X.
[52] W.-X. Huang, H.-J. Sung, Three-dimensional simulation of a flapping flag in a uniform flow, J. Fluid Mech. 653 (June 2010) 301–336, https://doi.org/10.1017/S0022112010000248.
[53] F.-B. Tian, et al., Fluid–structure interaction involving large deformations: 3D simulations and applications to biological systems, J. Comput. Phys. 258 (2014) 451–469, https://doi.org/10.1016/j.jcp.2013.10.047.
[54] I. Lee, H. Choi, A discrete-forcing immersed boundary method for the fluid-structure interaction of an elastic slender body, J. Comput. Phys. 280 (Nov. 2015) 529–546, https://doi.org/10.1016/j.jcp.2014.09.028.
[55] Z. Li, G. Oger, D. Le Touzé, A partitioned framework for coupling LBM and FEM through an implicit IBM allowing non-conforming time-steps: application to fluid-structure interaction in biomechanics, J. Comput. Phys. 449 (2022) 110786, https://doi.org/10.1016/j.jcp.2021.110786.
[56] M.D. de Tullio, G. Pascazio, A moving-least-squares immersed boundary method for simulating the fluid–structure interaction of elastic bodies with arbitrary thickness, J. Comput. Phys. (ISSN 0021-9991) 325 (2016) 201–225, https://doi.org/10.1016/j.jcp.2016.08.020.
[57] S. Piperno, Explicit/implicit fluid/structure staggered procedures with a structural predictor and fluid subcycling for 2D inviscid aeroelastic simulations, Int. J. Numer. Methods Fluids 25 (10) (1997) 1207–1226, https://doi.org/10.1002/(SICI)1097-0363(19971130)25:10<1207::AID-FLD616>3.0.CO;2-R.
[58] M. Haussmann, et al., Galilean invariance study on different lattice Boltzmann fluid–solid interface approaches for vortex-induced vibrations, Comput. Math. Appl. 80 (5) (2020) 671–691, https://doi.org/10.1016/j.camwa.2020.04.022.
[59] K. Wong, et al., Numerical stability of partitioned approach in fluid-structure interaction for a deformable thin-walled vessel, Comput. Math. Methods Med. 2013 (2013), https://doi.org/10.1155/2013/638519.

[60] Michael Breuer, Guillaume De Nayer, Manuel Münsch, Fluid–Structure Interaction of a Flexible Structure in a Turbulent Flow Using LES, vol. 15, ISBN 978-94-007-2481-5, Dec. 2010, pp. 449–454.
[61] C. Forster, W. Wall, E. Ramm, Artificial added mass instabilities in sequential staggered coupling of nonlinear structures and incompressible viscous flows, Comput. Methods Appl. Mech. Eng. 196 (Jan. 2007) 1278–1293, https://doi.org/10.1016/j.cma.2006.09.002.