A partitioned framework for coupling LBM and FEM through an implicit IBM allowing non-conforming time-steps: Application to fluid-structure interaction in biomechanics

Zhe Li, Guillaume Oger, David Le Touzé

To cite this version:

Zhe Li, Guillaume Oger, David 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. Journal of Computational Physics, 2022, 449, pp.110786. 10.1016/j.jcp.2021.110786. hal-04856027

HAL Id: hal-04856027
https://hal.science/hal-04856027v1

Submitted on 26 Dec 2024

A partitioned framework for coupling LBM and FEM through an implicit IBM allowing non-conforming time-steps: application to fluid-structure interaction in biomechanics

Zhe Li*, Guillaume Oger, David Le Touzé

[a]*LHEEA Research Department, Ecole Centrale Nantes (ECN and CNRS), Nantes, France*

Abstract

This paper presents a partitioned framework for the numerical simulation of fluid-structure interactions by coupling the lattice Boltzmann method (LBM) and the finite element method (FEM). The two numerical methods LBM and FEM are coupled with an implicit immersed boundary method (IBM) in a strong way, which ensures exactly the no-slip condition and the continuities of velocity and stress at the fluid-solid interface and each instant in time. In the proposed partitioned coupling procedure, the coupling system of equations are first established and then condensed to the interface. By solving the condensed coupling system of equations, the interface force field is obtained and sent to both solvers to accomplish time integrations in each sub-domain. In addition, two strategies based on linear interpolation in time are proposed to handle the cases with non-conforming time-steps in the fluid and solid sub-domains. Through several 2D and 3D numerical test-cases on the mechanical heart valve, the fluid-induced vibration of a deformable solid beam, the flapping flag, the proposed coupling framework is validated with good agreements with references. Finally, a test-case on the interaction between the blood flow and the aortic valve is carried out, showing the applicability of the present framework in realistic biomechanical applications.

Keywords: Strong partitioned coupling, lattice Boltzmann method, finite element method, implicit immersed boundary method, fluid-structure interaction, non-conforming time-steps

---

1. Introduction

Fluid-structure interactions (FSI) are widely present within the human body, for instance, in the heart between the pulsatile blood flow and the cardiac muscle and valves, the interaction between the urine and the bladder, etc. With the fast development of computational techniques, numerical simulation plays a more and more critical role in designing and assessing new patient-specific therapies. Indeed, as a secure analysis tool, the

---

*Corresponding author
Email address: `zhe.li@ec-nantes.fr` (Zhe Li)

numerical simulation based on computational fluid dynamics (CFD) and computational solid dynamics (CSD) can provide details which are unreachable today using experimental measurements.

Up to now, various types of numerical simulation methods have been proposed and applied in biomechanical studies involving FSI, such as blood flows through mechanical and bioprosthetic heart valves [6, 9, 12, 15, 32, 44, 45, 48, 65], cerebral aneurysms [2] and pulmonary airways [37], etc. Over the past decades, lattice Boltzmann method (LBM) [71] emerged as an efficient and reliable alternative to conventional CFD methods for simulating various types of flows. Recently, a thorough review on the theory of LBM for nearly-incompressible fluid flows has been given by Lallemand et al. [51]. Being highly parallelizable, LBM appears as a good candidate for modeling biomechanical problems with high performance computing (HPC) techniques. In the present work, we propose a coupling framework for the numerical simulation of FSI problems, in which the fluid sub-domain is solved using LBM, whereas the deformable structures are modeled with the finite element method (FEM) [3, 42] taking into account geometric non-linearities. Because of the radically different descriptions of LBM (Eulerian) and FEM (Lagrangian), we choose to couple these methods through the use of the immersed boundary method (IBM).

Initially proposed by Peskin [58] for simulating the blood flow in the heart, IBM and its variations have been applied for handling different types of FSI problems. With IBM, it is straightforward and easy to incorporate solid boundaries in fluid flows without changing the mesh of the fluid sub-domain. An insightful review on IBM was recently given by Huang & Tian [41]. More specifically, some previous works on IB-LBM can be found in [14, 24, 26, 43, 72, 77, 79]. In the present work, we adopt the implicit IB-LBM first proposed by Wu & Shu [79], of which the key is to obtain the IB-related force at all IB-points simultaneously by solving a linear system of equations at the fluid-solid interface. As a direct-forcing scheme, this implicit IBM does not require to tune any artificial stiffness coefficient and can ensure exactly the no-slip condition, in the sense that the interpolated fluid velocity is exactly equal to the solid velocity at the interface. Another reason why we prefer this implicit IBM of Wu & Shu [79] is that it does not need iterative procedures to enforce the no-slip condition.

As for the FSI coupling algorithms, with the classifications given in [25], they can be categorized as monolithic and partitioned procedures. In monolithic procedures, the fluid and solid discrete equations are established and then solved in the same entity (solver or code). Whereas in partitioned procedures, the fluid and solid sub-domains are solved separately in two entities and communications between these two entities are required. Usually but not always, partitioned procedures are less stable than monolithic ones, as they do not fulfill exactly the continuity conditions at the fluid-solid interface (velocity or stress continuity). However, partitioned procedures are more flexible to be used in realistic applications, as the modularities of different solvers are retained [25]. In addition, based on whether the continuity conditions are ensured at the interface and each instant or not, partitioned coupling procedures can be classified as strongly-coupled or weakly-coupled (loosely-coupled or staggered) algorithms [81]. A strong coupling can be obtained by means of sub-iterations during each time-step, which might induce the degradation of coupling efficiency, due to the extra-cost for sub-iterations and communications between the two solvers. To reduce this cost, some efforts have been made to significantly decrease the sub-iteration steps or completely remove the sub-iterations [31, 76, 80].

Probably due to its implicit feature, to the best of our knowledge, this implicit IBM [79] has not been applied in strong couplings between LBM and FEM. Wang et al. [78] coupled the non-linear FEM with the lattice Boltzmann flux solver (LBFS) via this implicit IBM, however, in a weak way. Some other LBM-FEM coupling algorithms can also be found in the literature, such as the ones in [13, 14, 20, 47], etc. However, they are either weakly coupled or strongly coupled with the help of sub-iterations. To fill this gap, in the present work we propose a coupling framework for simulating FSI problems involving deformable solid structures, in which the LBM and FEM are coupled in a strong and partitioned way. Similar to the monolithic coupling procedure, in the proposed partitioned coupling framework, the FSI coupling equations are expressed in a strong way without destroying the continuity conditions at the interface. However, this coupling system will not be solved directly in any solver, but will be firstly condensed to the interface. By solving this condensed coupling system, we can obtain the interface force field which is then sent to the fluid and solid solvers to accomplish the time integration in each sub-domain. The proposed coupling framework can ensure exactly the no-slip condition as well as the continuity conditions at the fluid-solid interface and each instant, while avoiding sub-iterations during each time-step. In the previous non-staggered coupling algorithms [55, 54], the IBM is based on the computation of the Lagrangian weight with the method proposed in [59]. However, it has been recently demonstrated that this Lagrangian weight cannot ensure exactly the no-slip condition [85, 86] in general cases, although it does remove the penetration of streamlines across the immersed boundary as shown in [56]. Moreover, in the present work, two strategies based on linear interpolation in time for using non-conforming time-steps in fluid and solid solvers are provided, which render the coupling framework more flexible and efficient to be used in realistic applications, such as the biomechanical problems involving the interaction between the blood flow and heart valves.

The rest of the paper is organized as follows. Sec. 2 presents briefly the adopted LBM and the projection-based regularization procedure. Sec. 3 provides necessary details about the used FEM and the hyperelastic material models. The implicit direct-forcing IBM as well as the evaluation of interfacial forces are presented in Sec. 4. The proposed coupling algorithm is shown in Sec. 5, compared with an ordinary weak coupling algorithm and a strong coupling one based on sub-iterations. In addition, two strategies based on linear interpolation in time are provided in Sec. 6 for the use of non-conforming time-steps in fluid and solid sub-domains. The proposed strong coupling framework is validated with several validation test-cases and applied in a biomechanical application case in Sec. 7. Finally, the conclusions are drawn in Sec. 8.

2. Lattice Boltzmann method for the fluid sub-domain

2.1. Lattice Boltzmann equation

The continuous Boltzmann equation with the BGK collision model [5] reads

$$\frac{\partial f}{\partial t}+\boldsymbol{\xi}\cdot\frac{\partial f}{\partial \boldsymbol{x}}+\boldsymbol{g}\cdot\frac{\partial f}{\partial \boldsymbol{\xi}}=-\frac{f-f^{eq}}{\mathcal{T}}, \tag{1}$$

where $f=f(\boldsymbol{x},\boldsymbol{\xi},t)$ is the single particle distribution function at the position $\boldsymbol{x}$ and time t for the microscopic velocity $\boldsymbol{\xi}$, $\boldsymbol{g}=\boldsymbol{g}(\boldsymbol{x},t)$ is the external body force per unit mass, $f^{eq}=f^{eq}(\boldsymbol{x},\boldsymbol{\xi},t)$ is the equilibrium distribution function and $\mathcal{T}$ is the relaxation time.

He & Luo [36] first demonstrated that the lattice Boltzmann equation (LBE) is a finite difference form of the continuous Boltzmann equation, which can be derived in an *a priori* way. In the present work, the LBE for simulating isothermal weakly-compressible single-phase fluid flows is adopted, which is written as

$$f_\alpha(\boldsymbol{x}+\boldsymbol{\xi}_\alpha\Delta t,t+\Delta t)=f_\alpha(\boldsymbol{x},t)-\frac{\Delta t}{\tau}\left(f_\alpha(\boldsymbol{x},t)-f_\alpha^{eq}(\boldsymbol{x},t)\right)+\Delta t\left(1-\frac{\Delta t}{2\tau}\right)\mathcal{F}_\alpha(\boldsymbol{x},t), \tag{2}$$

where $f_\alpha(\boldsymbol{x},t)$ denotes the distribution function (after a change of variables) in the αth lattice direction with $\boldsymbol{\xi}_\alpha$ being the corresponding lattice velocity, Δt is the time-step and the kinematic viscosity ν_f is related to the relaxation-time parameter τ as $\nu_f=c_f^2(\tau-0.5\Delta t)$ with $c_f=\sqrt{1/3}\Delta x_f/\Delta t$ being the fluid sound speed for the adopted types of lattices D2Q9 and D3Q19 [60] in the present work. In addition, Δx_f denotes the lattice spacing and $f_\alpha^{eq}(\boldsymbol{x},t)$ is referred to as the discrete equilibrium distribution function, which can be obtained by the Hermite polynomial expansion of the Maxwell-Boltzmann equilibrium distribution [66]

$$f_\alpha^{eq}=\rho_f w_\alpha\left(1+\frac{\boldsymbol{\xi}_\alpha\cdot\boldsymbol{v}_f}{c_f^2}+\frac{(\boldsymbol{\xi}_\alpha\cdot\boldsymbol{v}_f)^2}{2c_f^4}-\frac{\boldsymbol{v}_f\cdot\boldsymbol{v}_f}{2c_f^2}\right), \tag{3}$$

where $\rho_f=\rho_f(\boldsymbol{x},t)$ and $\boldsymbol{v}_f=\boldsymbol{v}_f(\boldsymbol{x},t)$ denote respectively the fluid density and macroscopic velocity, and w_α is the weight coefficient of the lattice. The last term $\mathcal{F}_\alpha(\boldsymbol{x},t)$ in Eq. (2) is the body force-related term given as [34]

$$\mathcal{F}_\alpha=\rho_f w_\alpha\left(\frac{\boldsymbol{\xi}_\alpha-\boldsymbol{v}_f}{c_f^2}+\frac{(\boldsymbol{\xi}_\alpha\cdot\boldsymbol{v}_f)\boldsymbol{\xi}_\alpha}{c_f^4}\right)\cdot\boldsymbol{g}. \tag{4}$$

With the adopted force scheme of Guo et al. [34, 35], the macroscopic fluid density and velocity are computed as

$$\rho_f=\sum_\alpha f_\alpha, \qquad \rho_f\boldsymbol{v}_f=\sum_\alpha \boldsymbol{\xi}_\alpha f_\alpha+\frac{\Delta t}{2}\rho_f\boldsymbol{g}. \tag{5}$$

Finally, the pressure in the fluid domain is computed as $p_f=\rho_f c_f^2$ (barotropic equation of state), which is sufficient for simulating isothermal weakly-compressible fluid flows.

2.2. Regularization procedure based on Hermite polynomial expansion

While being simple and straightforward to be implemented, the previously presented single-relaxation-time (SRT) LBE often suffers from numerical instabilities, especially for simulating fluid flows of high Reynolds numbers. To circumvent the instability issue of LBE, various collision models have been proposed in the literature [10], such as the multi-relaxation-time (MRT) models [16, 17, 29, 50], the cascaded LBM [28] and its central-moment version [21], the regularized LBM [11, 52, 84], the cumulant scheme [64] and the entropic LBM [1, 46]. In the present work, we adopt the regularized LBM [84] based on the projection of the non-equilibrium distribution function onto the sub-space spanned by the first N Hermite polynomials. The regularized discrete LB scheme is written as

$$f_\alpha(\boldsymbol{x}+\boldsymbol{\xi}_\alpha\Delta t, t+\Delta t) = f_\alpha^{eq}(\boldsymbol{x},t) + \left(1-\frac{\Delta t}{\tau}\right)\hat{f}_\alpha^{neq}(\boldsymbol{x},t) + \Delta t\left(1-\frac{\Delta t}{2\tau}\right)\mathcal{F}_\alpha(\boldsymbol{x},t), \tag{6}$$

where the regularized non-equilibrium part of the pre-collision distribution function $\hat{f}_\alpha^{neq}(\boldsymbol{x},t)$ is computed as

$$\hat{f}_\alpha^{neq}(\boldsymbol{x},t) = w_\alpha \sum_{n=0}^{N} \frac{1}{n!}\boldsymbol{Q}_{neq}^{(n)}(\boldsymbol{x},t) : \mathscr{H}^{(n)}(\boldsymbol{\xi}_\alpha), \tag{7}$$

with $\boldsymbol{Q}_{neq}^{(n)}(\boldsymbol{x},t)$ and $\mathscr{H}^{(n)}(\boldsymbol{\xi}_\alpha)$ being respectively the nth-order expansion coefficient and Hermite polynomial, which are computed as

$$\boldsymbol{Q}_{neq}^{(n)}(\boldsymbol{x},t) = \sum_\alpha f_\alpha^{neq}(\boldsymbol{x},t)\mathscr{H}^{(n)}(\boldsymbol{\xi}_\alpha) = \sum_\alpha \left(f_\alpha(\boldsymbol{x},t) - f_\alpha^{eq}(\boldsymbol{x},t)\right)\mathscr{H}^{(n)}(\boldsymbol{\xi}_\alpha), \tag{8}$$

and the first few Hermite polynomials are given as

$$\mathscr{H}^{(0)}(\boldsymbol{\xi}_\alpha) = 1, \quad \mathscr{H}^{(1)}(\boldsymbol{\xi}_\alpha) = \frac{\boldsymbol{\xi}_\alpha}{c_f}, \quad \mathscr{H}^{(2)}(\boldsymbol{\xi}_\alpha) = \frac{\boldsymbol{\xi}_\alpha \otimes \boldsymbol{\xi}_\alpha}{c_f^2} - \boldsymbol{I}, \tag{9}$$

with $\boldsymbol{I}$ being the second-order identity tensor.

By means of this regularization procedure (7), all the terms on the right hand side of Eq. (6) lie in the sub-space spanned by the first N Hermite polynomials. For simulating isothermal weakly-compressible fluid flows, it suffices to truncate at $N = 2$. In addition, it is noteworthy that the 1st-order expansion coefficient $\boldsymbol{Q}_{neq}^{(1)}$ is not equal to zero due to the use of the adopted force model [54].

3. Finite element method for the solid sub-domain

3.1. Total Lagrangian formulation

Using the principle of virtual work under the total Lagrangian formulation, the weak form of the dynamical equilibrium equation is written as

$$\int_{\Omega_0} \delta\boldsymbol{u}_s \cdot \left(\rho_s^0 \frac{\mathrm{d}^2\boldsymbol{u}_s}{\mathrm{d}t^2} - \nabla_0\cdot\boldsymbol{P}\right)\mathrm{d}\Omega_0 = 0, \tag{10}$$

where $\boldsymbol{u}_s = \boldsymbol{u}_s(\boldsymbol{X}, t)$ denotes the solid displacement field, $\rho_s^0 = \rho_s^0(\boldsymbol{X})$ is the initial solid density, and $\boldsymbol{P} = \boldsymbol{P}(\boldsymbol{X}, t)$ is the nominal stress tensor. In addition, Ω_0 denotes the initial solid configuration and $\nabla_0 \cdot$ is the divergence operator with respect to the material coordinate $\boldsymbol{X}$. Finally, the solid velocity $\boldsymbol{v}_s$ and acceleration $\boldsymbol{a}_s$ can be obtained as $\boldsymbol{v}_s = \mathrm{d}\boldsymbol{u}_s/\mathrm{d}t$ and $\boldsymbol{a}_s = \mathrm{d}^2\boldsymbol{u}_s/\mathrm{d}t^2$.

With the finite element discretization $\delta\boldsymbol{u}_s = N_I(\boldsymbol{X})\delta\boldsymbol{u}_I(t)$ and $\boldsymbol{a}_s = N_I(\boldsymbol{X})\boldsymbol{a}_I(t)$, from Eq. (10) one can obtain

$$\left(\int_{\Omega_0} \rho_s^0 N_I N_J \,\mathrm{d}\Omega_0\right) \boldsymbol{a}_J + \int_{\Omega_0} (\boldsymbol{\nabla}_0 N_I) \cdot \boldsymbol{P} \,\mathrm{d}\Omega_0 - \int_{\partial\Omega_0} N_I \boldsymbol{t}_0 \,\mathrm{d}\Gamma_0 = \boldsymbol{0}, \tag{11}$$

where N_I and N_J are the shape functions for the Ith and Jth nodes, $\boldsymbol{a}_J$ is the acceleration vector for the Jth node, and $\boldsymbol{t}_0$ is the surface force (per unit area) defined on the boundary $\partial\Omega_0$. Notice that the Einstein summation convention is adopted in Eq. (11). By defining the consistent mass matrix element M_{IJ}, the internal nodal force $\boldsymbol{q}_{int,I}$ and the external nodal force $\boldsymbol{q}_{ext,I}$, Eq. (11) can be rewritten for the Ith node as

$$M_{IJ}\boldsymbol{a}_J = \boldsymbol{q}_{ext,I} - \boldsymbol{q}_{int,I}, \tag{12}$$

with

$$M_{IJ} = \int_{\Omega_0} \rho_s^0 N_I N_J \,\mathrm{d}\Omega_0, \quad \boldsymbol{q}_{ext,I} = \int_{\partial\Omega_0} N_I \boldsymbol{t}_0 \,\mathrm{d}\Gamma_0, \quad \boldsymbol{q}_{int,I} = \int_{\Omega_0} (\boldsymbol{\nabla}_0 N_I) \cdot \boldsymbol{P} \,\mathrm{d}\Omega_0. \tag{13}$$

Finally, the whole system of equations for all FE-nodes can be obtained using the scatter-gather operation and is expressed as

$$\boldsymbol{M}_s \underline{\boldsymbol{a}}_s = \underline{\boldsymbol{q}}_{ext} - \underline{\boldsymbol{q}}_{int}, \tag{14}$$

where $\boldsymbol{M}_s$ is the global consistent mass matrix, and $\underline{\boldsymbol{a}}_s$, $\underline{\boldsymbol{q}}_{ext}$ and $\underline{\boldsymbol{q}}_{int}$ are the global acceleration, the global external and internal nodal force vectors for all FE-nodes, respectively. For instance, in 3D cases, the acceleration vector is $\underline{\boldsymbol{a}}_s = [a_1^x, a_1^y, , a_1^z, \ldots, a_I^x, a_I^y, a_I^z, \ldots, a_{N_s}^x, a_{N_s}^y, a_{N_s}^z]^\top$ where a_I^x is the acceleration component in x-direction and N_s denotes the total number of FE-nodes.

3.2. Hyperelastic material models

In the present work, to calculate the internal nodal force $\boldsymbol{q}_{int,I}$ in Eq. (13), we adopt hyperelastic material models in order to simulate moderately large deformations in solid structures. The nominal stress tensor $\boldsymbol{P}$ is computed as $\boldsymbol{P} = \boldsymbol{S} \cdot \boldsymbol{F}^\top$ where $\boldsymbol{F} = \partial\boldsymbol{x}/\partial\boldsymbol{X}$ is the deformation gradient tensor, $\boldsymbol{x}$ the spacial coordinate, and $\boldsymbol{S}$ the second Piola-Kirchhoff stress tensor related to the Green strain tensor $\boldsymbol{E} = (\boldsymbol{F}^\top \cdot \boldsymbol{F} - \boldsymbol{I})/2$ via a hyperelastic material model as $\boldsymbol{S} = \partial\Psi/\partial\boldsymbol{E}$ with Ψ being a strain energy density function.

In the following test-cases, two hyperelastic material models have been used. The first one is the Saint Venant-Kirchhoff isotropic hyperelastic model

$$\boldsymbol{S} = \lambda_{\text{Lamé}} \mathrm{trace}(\boldsymbol{E})\boldsymbol{I} + 2\mu_{\text{Lamé}}\boldsymbol{E}, \tag{15}$$

where $\lambda_{\text{Lamé}}$ and $\mu_{\text{Lamé}}$ are the two Lamé constants which are related to the Young modulus E_s and the Poisson ratio ν_s as

$$\lambda_{\text{Lamé}} = \frac{\nu_s E_s}{(1+\nu_s)(1-2\nu_s)}, \qquad \mu_{\text{Lamé}} = \frac{E_s}{2(1+\nu_s)}. \tag{16}$$

The second isotropic hyperelastic material model is the Neo-Hookean model adopted by Sigüenza et al. [67] for modeling the aortic heart valve. In this hyperelastic model, the strain energy function Ψ is given as

$$\Psi = \frac{G_s}{2}\left(J_0^{-2/3} I_1 - 3\right) + \frac{K_s}{2}\left(\ln J_0\right)^2, \tag{17}$$

where G_s and K_s denote respectively the shear and bulk moduli, of which the values are determined by means of a simulation-experiment fitting in a uniaxial tensile test [67]. In addition, $J_0 = J_0(\boldsymbol{X}, t) = \det(\boldsymbol{F})$ is the Jabobian determinant of the deformation gradient tensor and $I_1 = \operatorname{trace}(\boldsymbol{F}^\top \cdot \boldsymbol{F})$ is the first invariant of the right Cauchy-Green deformation tensor. Based on the strain energy density function given in Eq. (17), we can derive the second Piola-Kirchhoff stress tensor $\boldsymbol{S}$ as

$$\boldsymbol{S} = \frac{\partial \Psi}{\partial \boldsymbol{E}} = G_s J_0^{-2/3} \boldsymbol{I} + \left(K_s \ln J_0 - \frac{G_s}{3} I_1 J_0^{-2/3}\right) (\boldsymbol{F}^\top \cdot \boldsymbol{F})^{-1}. \tag{18}$$

For the sake of completeness of the paper, we briefly provide here the major procedures for calculating the internal nodal force vector $\underline{\boldsymbol{q}}_{int}$ in Eq. (14). As mentioned previously, in practical implementation, the internal nodal force is firstly calculated for the nodes of each element Ω_0^e by means of a Gauss quadrature as

$$\boldsymbol{q}^e_{int,I} = \int_{\Omega_0^e} (\boldsymbol{\nabla}_0 N_I) \cdot \boldsymbol{P}\, \mathrm{d}\Omega_0^e = \int_{\square} (\boldsymbol{\nabla}_0 N_I) \cdot \boldsymbol{P} J_\eta^0\, \mathrm{d}\square \simeq \sum_\gamma \omega_\gamma (\nabla_0 N_I)_\gamma \cdot \boldsymbol{P}_\gamma J^0_{\eta,\gamma}, \tag{19}$$

where $\square$ denotes the parent domain of each element, J_η^0 is the Jacobian between the material and element coordinates and ω_γ is the quadrature weight for the γth quadrature point $\boldsymbol{\eta}_\gamma$. The previously presented material models will be used to compute the nominal stress tensor $\boldsymbol{P}_\gamma = \boldsymbol{P}(\boldsymbol{\eta}_\gamma, t)$ at $\boldsymbol{\eta}_\gamma$.

Once all elemental internal nodal force $\underline{\boldsymbol{q}}^e_{int}$ vectors are obtained, one can carry out the gather operation in order to get the global internal nodal force vector $\underline{\boldsymbol{q}}_{int}$ as

$$\underline{\boldsymbol{q}}_{int} = \sum_e \boldsymbol{L}_e^\top \underline{\boldsymbol{q}}^e_{int}, \tag{20}$$

where $\boldsymbol{L}_e$ is the boolean connectivity matrix that gives the nodal values of a variable for the eth element from the global vector. For instance, the acceleration for all nodes in the eth element is obtained as $\underline{\boldsymbol{a}}^e_s = \boldsymbol{L}_e \underline{\boldsymbol{a}}_s$.

4. Implicit direct-forcing immersed boundary method

4.1. Implicit formulation

The implicit direct-forcing IBM consists in finding the forces at all Lagrangian points simultaneously by solving a linear system so as to ensure exactly the no-slip velocity boundary condition at the fluid-solid interface.

In FSI problems, this refers to imposing the velocity continuity condition at the interface in the sense that the interpolated fluid velocity is exactly equal to the local solid velocity. This idea was proposed and applied by Taira & Colonius [73] in the framework of a projection method. In the literature of LBM, the first implicit IBM was proposed by Wu & Shu [79], which is adopted in the present work for the coupling of LBM and FEM.

As shown in Eq. (5), in the framework of LBM, the fluid macroscopic velocity $\boldsymbol{v}_f$ can be computed as

$$\boldsymbol{v}_f = \frac{\sum\limits_{\alpha} \boldsymbol{\xi}_\alpha f_\alpha}{\sum\limits_{\alpha} f_\alpha} + \frac{\Delta t}{2}\boldsymbol{g}, \tag{21}$$

which is sometimes written as $\boldsymbol{v}_f = \boldsymbol{v}_f^* + \delta \boldsymbol{v}_f$ with $\boldsymbol{v}_f^* = \sum_\alpha \boldsymbol{\xi}_\alpha f_\alpha / \sum_\alpha f_\alpha$ being the intermediate velocity and $\delta \boldsymbol{v}_f = \Delta t \boldsymbol{g}/2$ being the velocity-correction term.

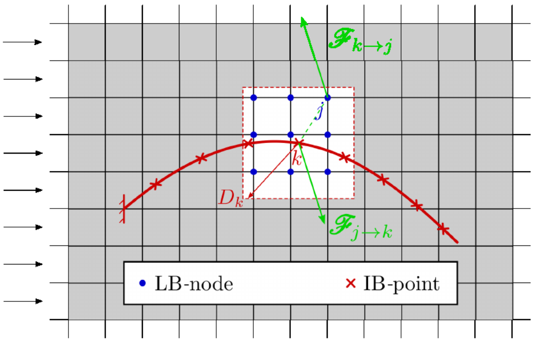


Figure 1: Immersed boundary method: the jth LB-node is located inside the support kernel D_k of the kth IB-point.

The idea of IBM is to impose that the interpolated fluid velocity is equal to the one of the solid boundary, by computing an appropriate body force field near the immersed boundary. For example, as shown in Fig. 1, the IBM imposes that the interpolated fluid velocity $\mathcal{I}[\boldsymbol{v}_f]_k$ at the kth Lagrangian point is equal to the solid velocity $\boldsymbol{v}_k$. Using Eq. (21), one has

$$\boldsymbol{v}_k = \mathcal{I}[\boldsymbol{v}_f]_k = \mathcal{I}\left[\frac{\sum\limits_{\alpha} \boldsymbol{\xi}_\alpha f_\alpha}{\sum\limits_{\alpha} f_\alpha}\right]_k + \frac{\Delta t}{2}\mathcal{I}[\boldsymbol{g}]_k, \tag{22}$$

where $\mathcal{I}[\phi]_k$ denotes the interpolation operator giving the interpolated value of a certain variable $\phi(\boldsymbol{x}, t)$ at the kth Lagrangian IB-point. The interpolation operator is defined as

$$\mathcal{I}[\phi]_k = \sum_{j\in D_k} \tilde{\delta}_{jk} \phi_j \Delta x_f^d, \tag{23}$$

where d is the number of dimensions, $\phi_j = \phi(\boldsymbol{x}_j, t)$ and $\tilde{\delta}_{jk} = \tilde{\delta}(\boldsymbol{x}_j - \boldsymbol{x}_k)$ denotes the kernel function with $\boldsymbol{x}_k$ and $\boldsymbol{x}_j$ being the coordinate of the kth IB-point and the jth LB-node within the support domain D_k of the kernel function, respectively. In the present work, the 3-points kernel function of Roma et al. [62] has been used for all numerical simulations.

In the adopted implicit direct-forcing IBM, the IB-related force is computed at the Lagrangian IB-points and then spread to the Eulerian LB-nodes. The body force term $\boldsymbol{g} = \boldsymbol{g}(\boldsymbol{x}, t)$ in Eq. (22) at LB-nodes is spread from the IB-points by means of a spreading operator $\boldsymbol{g}_j = \mathcal{S}[\boldsymbol{g}]_j$ defined as

$$\mathcal{S}[\phi]_j = \sum_{k \in D_j} \tilde{\delta}_{jk} \phi_k \Delta v_k, \tag{24}$$

where $\phi_k = \phi(\boldsymbol{x}_k, t)$ is the value of the variable ϕ at kth IB-point inside the support domain of the jth LB-node and Δv_k is the volume weight of the kth IB-point.

Applying the definitions (23) and (24) in Eq. (22) gives

$$\begin{aligned}
\mathcal{I}[\boldsymbol{v}_f]_k &= \mathcal{I}\left[\frac{\sum\limits_{\alpha} \boldsymbol{\xi}_\alpha f_\alpha}{\sum\limits_{\alpha} f_\alpha}\right]_k + \frac{\Delta t}{2} \sum_{j \in D_k} \tilde{\delta}_{jk} \boldsymbol{g}_j \Delta x_f^d, \\
&= \mathcal{I}\left[\boldsymbol{v}_f^*\right]_k + \frac{\Delta t}{2} \sum_{j \in D_k} \tilde{\delta}_{jk} \Delta x_f^d \sum_{k' \in D_j} \tilde{\delta}_{jk'} \boldsymbol{g}_{k'} \Delta v_{k'}, \\
&= \mathcal{I}\left[\boldsymbol{v}_f^*\right]_k + \frac{\Delta t}{2} \sum_{j \in D_k} \tilde{\delta}_{jk} \Delta x_f^d \sum_{k'=1}^{N_i} \tilde{\delta}_{jk'} \boldsymbol{g}_{k'} \Delta v_{k'}, \\
&= \mathcal{I}\left[\boldsymbol{v}_f^*\right]_k + \frac{\Delta t}{2} \sum_{j \in D_k} \tilde{\delta}_{jk} \Delta x_f^d \left(\tilde{\delta}_{j1} \boldsymbol{g}_1 \Delta v_1 + \tilde{\delta}_{j2} \boldsymbol{g}_2 \Delta v_2 + \ldots + \tilde{\delta}_{jN_i} \boldsymbol{g}_{N_i} \Delta v_{N_i}\right), \\
&= \mathcal{I}\left[\boldsymbol{v}_f^*\right]_k + \frac{\Delta t}{2} \Delta x_f^d \begin{bmatrix} \sum\limits_{j \in D_k} \tilde{\delta}_{jk} \tilde{\delta}_{j1} & \sum\limits_{j \in D_k} \tilde{\delta}_{jk} \tilde{\delta}_{j2} & \ldots & \sum\limits_{j \in D_k} \tilde{\delta}_{jk} \tilde{\delta}_{jN_i} \end{bmatrix} \begin{bmatrix} \boldsymbol{g}_1 \Delta v_1 \\ \boldsymbol{g}_2 \Delta v_2 \\ \vdots \\ \boldsymbol{g}_{N_i} \Delta v_{N_i} \end{bmatrix},
\end{aligned} \tag{25}$$

where N_i denotes the total number of the IB-points on the fluid-solid interface.

We shall now rewrite Eq. (25) for all IB-points $k \in [1, N_i]$ in a matrix form as

$$\underbrace{\begin{bmatrix} \mathcal{I}[\boldsymbol{v}_f]_1 \\ \mathcal{I}[\boldsymbol{v}_f]_2 \\ \vdots \\ \mathcal{I}[\boldsymbol{v}_f]_{N_i} \end{bmatrix}}_{\boldsymbol{L}_f \underline{\boldsymbol{v}}_f} = \underbrace{\begin{bmatrix} \mathcal{I}[\boldsymbol{v}_f^*]_1 \\ \mathcal{I}[\boldsymbol{v}_f^*]_2 \\ \vdots \\ \mathcal{I}[\boldsymbol{v}_f^*]_{N_i} \end{bmatrix}}_{\boldsymbol{L}_f \underline{\boldsymbol{v}}_f^*} + \frac{\Delta t}{2} \Delta x_f^d \underbrace{\begin{bmatrix} \sum\limits_{j \in D_1} \tilde{\delta}_{j1} \tilde{\delta}_{j1} & \sum\limits_{j \in D_1} \tilde{\delta}_{j1} \tilde{\delta}_{j2} & \ldots & \sum\limits_{j \in D_1} \tilde{\delta}_{j1} \tilde{\delta}_{jN_i} \\ \sum\limits_{j \in D_2} \tilde{\delta}_{j2} \tilde{\delta}_{j1} & \sum\limits_{j \in D_2} \tilde{\delta}_{j2} \tilde{\delta}_{j2} & \ldots & \sum\limits_{j \in D_2} \tilde{\delta}_{j2} \tilde{\delta}_{jN_i} \\ \vdots & \vdots & \ddots & \vdots \\ \sum\limits_{j \in D_{N_i}} \tilde{\delta}_{jN_i} \tilde{\delta}_{j1} & \sum\limits_{j \in D_{N_i}} \tilde{\delta}_{jN_i} \tilde{\delta}_{j2} & \ldots & \sum\limits_{j \in D_{N_i}} \tilde{\delta}_{jN_i} \tilde{\delta}_{jN_i} \end{bmatrix}}_{\boldsymbol{A}} \underbrace{\begin{bmatrix} \boldsymbol{g}_1 \Delta v_1 \\ \boldsymbol{g}_2 \Delta v_2 \\ \vdots \\ \boldsymbol{g}_{N_i} \Delta v_{N_i} \end{bmatrix}}_{\underline{\boldsymbol{G}}}, \tag{26}$$

or in a more concise way as

$$\boldsymbol{L}_f \underline{\boldsymbol{v}}_f = \boldsymbol{L}_f \underline{\boldsymbol{v}}_f^* + \frac{\Delta t}{2} \Delta x_f^d \boldsymbol{A} \underline{\boldsymbol{G}}, \tag{27}$$

where $\boldsymbol{L}_f$ is an interpolation matrix that relates the interpolated velocity on all IB-points with the global fluid velocity vector $\underline{\boldsymbol{v}}_f = [(\boldsymbol{v}_{f,1})^\top, (\boldsymbol{v}_{f,2})^\top, \ldots, (\boldsymbol{v}_{f,N_f})^\top]^\top$, where $(\boldsymbol{v}_{f,j})^\top = [v^x_{f,j}, v^y_{f,j}, v^z_{f,j}]$ denotes the fluid velocity at the jth LB-node for $j \in [1, N_f]$ with N_f being the total number of LB-nodes. In addition, $\underline{\boldsymbol{v}}^*_f$ is the global intermediate velocity vector.

Remark 1. *As will be shown subsequently, there is never need to provide explicitly the form of* $\boldsymbol{L}_f$ *in practical numerical simulations. The purpose of using Eq. (27) is to express the fluid coupling equations in a concise and clear way.*

Remark 2. *Due to the compactness of the kernel function, the matrix* $\boldsymbol{A}$ *in Eq. (27) is a large but sparse matrix. In FSI problems with moving solid boundaries, one needs to compute the matrix* $\boldsymbol{A}$ *at each time-step as long as the solid boundary moves. Nevertheless, the computation of* $\boldsymbol{A}$ *can be largely accelerated by means of a neighbor list for the Lagrangian IB-points. In addition, when discretizing the fluid-solid interface* Γ_i *using IB-points, it is suggested to verify that the distance between IB-points is slightly greater than the lattice spacing, i.e.* $\Delta x_s > \Delta x_f$*, for the sake of numerical stability.*

The linear system of equations (26) is the formulation of the implicit IBM proposed by Wu & Shu [79]. In a one-way FSI problem where the movement of the solid boundary is prescribed, $\boldsymbol{L}_f\underline{\boldsymbol{v}}_f$, $\boldsymbol{L}_f\underline{\boldsymbol{v}}^*_f$ and $\boldsymbol{A}$ are all known, hence one can compute the unknown vector $\underline{\boldsymbol{G}}$ by solving the linear system. It is here worth noting that there is no need to specify the volume weight Δv_k for each IB-point with $k \in [1, N_i]$, because $\boldsymbol{G}_k = \boldsymbol{g}_k \Delta v_k$ can be directly used in the spreading operation (24). As mentioned previously, this implicit IBM can ensure exactly the no-slip condition at the fluid-solid interface.

4.2. Evaluation of the interface force field $\underline{\boldsymbol{\Lambda}}$

Besides the velocity continuity (no-slip) condition, the force should also be continuous across the fluid-solid interface. Hence, the evaluation of the interface force field is particularly important in FSI simulations.

We shall now demonstrate how to obtain a conservative force formulation. To this end, let us consider the spreading stage (24) of the IBM, in which the fluid and solid communicate with each other by means of the body force term $\boldsymbol{g}$. As shown in Fig. 1 and Eq. (24), the IB-related acceleration (body force per unit mass) received by the jth LB-node from the kth IB-point is equal to

$$\boldsymbol{g}_{k\to j} = \tilde{\delta}_{jk}\boldsymbol{G}_k = \tilde{\delta}_{jk}\boldsymbol{g}_k\Delta v_k, \tag{28}$$

which means that the force exerted on the jth fluid control volume from the kth IB-point can be expressed as

$$\mathscr{F}_{k\to j} = \rho_{f,j}\boldsymbol{g}_{k\to j}\Delta x^d_f = \rho_{f,j}\tilde{\delta}_{jk}\boldsymbol{G}_k\Delta x^d_f, \tag{29}$$

where $\rho_{f,j} = \rho_f(\boldsymbol{x}_j, t)$ denotes the density of fluid at the jth LB-node.

By means of the Newton's third law, the force exerted on the kth IB-point by the jth LB-node is then

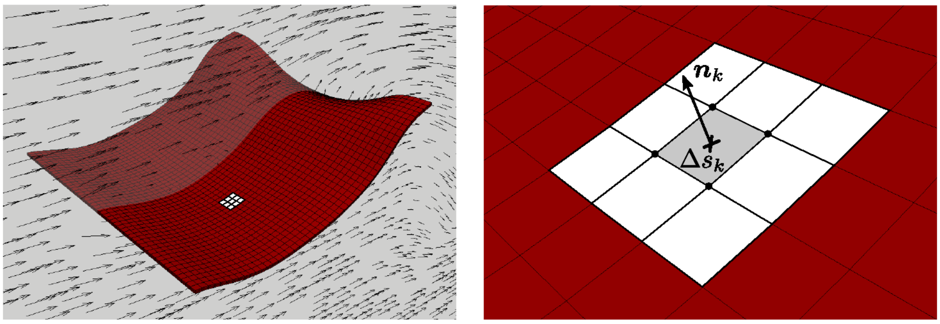
$$\mathscr{F}_{j\to k} = -\mathscr{F}_{k\to j} = -\rho_{f,j}\tilde{\delta}_{jk}\boldsymbol{G}_k\Delta x_f^d. \tag{30}$$

By taking into account the forces from all LB-nodes within the support kernel D_k of the kth IB-point, one can then compute the total force exerted on the kth IB-point from the fluid as

$$\mathscr{F}_{f\to k} = \sum_{j\in D_k}\mathscr{F}_{j\to k} = -\sum_{j\in D_k}\rho_{f,j}\tilde{\delta}_{jk}\boldsymbol{G}_k\Delta x_f^d = -\boldsymbol{G}_k\underbrace{\sum_{j\in D_k}\rho_{f,j}\tilde{\delta}_{jk}\Delta x_f^d}_{\mathcal{I}[\rho_f]_k}, \tag{31}$$

in which one may observe that the last term corresponds to the interpolated fluid density on the kth IB-point.

Hence, if one applies Eq. (31) to evaluate the force at the fluid-solid interface, one can ensure that this force formulation is conservative in the sense that the forces between the fluid and solid domains are equal and opposite. However, it is worth mentioning that this formulation does not conserve locally the angular momentum, because the forces $\mathscr{F}_{k\to j}$ and $\mathscr{F}_{j\to k}$ are generally not aligned with the vector $\boldsymbol{r}_{jk} = \boldsymbol{x}_j - \boldsymbol{x}_k$, as shown in Fig. 1. Nevertheless, as shown in the Appendix, the angular momentum is globally conserved with an approximation for the whole system including both the fluid and solid sub-domains.



(a) (b)

Figure 2: Discretization of the fluid-solid interface.

As done in our previous works [54, 55], we shall define an interface force $\boldsymbol{\lambda}$ in order to couple with the finite-element method. Supposing that the fluid-solid interface is discretized into a finite number N_i of elemental surfaces and the force is piece-wise constant on the interface, at the kth interface-element, one can get the surface force as $\boldsymbol{\lambda}_k = \mathscr{F}_{f\to k}/\Delta s_k = -\boldsymbol{G}_k\mathcal{I}[\rho_f]_k/\Delta s_k$ with Δs_k being the surface area of the kth interface-element, as shown in Fig. 2. In practice, $\underline{\boldsymbol{\Lambda}}$ and $\underline{\boldsymbol{G}}$ are related as $\underline{\boldsymbol{\Lambda}} = -\boldsymbol{B}\underline{\boldsymbol{G}}$

$$\underbrace{\begin{bmatrix} \boldsymbol{\lambda}_1 \\ \boldsymbol{\lambda}_2 \\ \vdots \\ \boldsymbol{\lambda}_{N_i} \end{bmatrix}}_{\underline{\boldsymbol{\Lambda}}} = - \underbrace{\begin{bmatrix} \frac{\mathcal{I}[\rho_f]_1}{\Delta s_1}\boldsymbol{I} & \boldsymbol{0} & \dots & \boldsymbol{0} \\ \boldsymbol{0} & \frac{\mathcal{I}[\rho_f]_2}{\Delta s_2}\boldsymbol{I} & \dots & \boldsymbol{0} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{0} & \boldsymbol{0} & \dots & \frac{\mathcal{I}[\rho_f]_{N_i}}{\Delta s_{N_i}}\boldsymbol{I} \end{bmatrix}}_{\boldsymbol{B}} \underbrace{\begin{bmatrix} \boldsymbol{G}_1 \\ \boldsymbol{G}_2 \\ \vdots \\ \boldsymbol{G}_{N_i} \end{bmatrix}}_{\underline{\boldsymbol{G}}}. \tag{32}$$

Substituting Eq. (32) into Eq. (27) yields

$$\boldsymbol{L}_f \underline{\boldsymbol{v}}_f + \frac{\Delta t}{2} \Delta x_f^d \boldsymbol{A}\boldsymbol{B}^{-1}\underline{\boldsymbol{\Lambda}} = \boldsymbol{L}_f \underline{\boldsymbol{v}}_f^*, \tag{33}$$

which will be used subsequently for solving the coupled system of equations.

5. Coupling algorithms

5.1. Fluid coupling equations

In the fluid sub-domain, after the streaming step of the LBM, one can update the distribution function $f_\alpha^{n+1} = f_\alpha(\boldsymbol{x}, t^{n+1})$ at all LB-nodes so that the new macroscopic density $\rho_f^{n+1} = \rho_f(\boldsymbol{x}, t^{n+1})$ can be calculated with Eq. (5). However, the macroscopic velocity $\boldsymbol{v}_f^{n+1}$ cannot be computed, because the IB-related body force is not known yet and will be obtained after solving the coupling system of equations.

To get the fluid coupling equations, let us rewrite Eq. (5) in a matrix form for all LB-nodes as

$$\underbrace{\frac{2}{\Delta t}\begin{bmatrix} \rho_{f,1}^{n+1}\boldsymbol{I} & \boldsymbol{0} & \dots & \boldsymbol{0} \\ \boldsymbol{0} & \rho_{f,2}^{n+1}\boldsymbol{I} & \dots & \boldsymbol{0} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{0} & \boldsymbol{0} & \dots & \rho_{f,N_f}^{n+1}\boldsymbol{I} \end{bmatrix}}_{\boldsymbol{R}_f} \underbrace{\begin{bmatrix} \boldsymbol{v}_{f,1}^{n+1} \\ \boldsymbol{v}_{f,2}^{n+1} \\ \vdots \\ \boldsymbol{v}_{f,N_f}^{n+1} \end{bmatrix}}_{\underline{\boldsymbol{v}}_f^{n+1}} - \underbrace{\begin{bmatrix} \rho_{f,1}^{n+1}\boldsymbol{g}_1^{n+1} \\ \rho_{f,2}^{n+1}\boldsymbol{g}_2^{n+1} \\ \vdots \\ \rho_{f,N_f}^{n+1}\boldsymbol{g}_{N_f}^{n+1} \end{bmatrix}}_{\underline{\boldsymbol{\Phi}}^{n+1}} = \underbrace{\frac{2}{\Delta t}\begin{bmatrix} \sum_\alpha \boldsymbol{\xi}_\alpha f_{\alpha,1}^{n+1} \\ \sum_\alpha \boldsymbol{\xi}_\alpha f_{\alpha,2}^{n+1} \\ \vdots \\ \sum_\alpha \boldsymbol{\xi}_\alpha f_{\alpha,N_f}^{n+1} \end{bmatrix}}_{\underline{\boldsymbol{b}}_f}, \tag{34}$$

which can also be expressed in a concise form as

$$\boldsymbol{R}_f \underline{\boldsymbol{v}}_f^{n+1} - \underline{\boldsymbol{\Phi}}^{n+1} = \underline{\boldsymbol{b}}_f, \tag{35}$$

where only the velocity $\underline{\boldsymbol{v}}_f^{n+1}$ and force $\underline{\boldsymbol{\Phi}}^{n+1}$ vectors are unknown. In addition, the force vector $\underline{\boldsymbol{\Phi}}^{n+1}$ can be related with the interface force field $\underline{\boldsymbol{\Lambda}}^{n+1}$ as $\underline{\boldsymbol{\Phi}}^{n+1} = -\boldsymbol{C}_f \underline{\boldsymbol{\Lambda}}^{n+1}$ with $\boldsymbol{C}_f$ being a matrix related to the spreading operation (24). Like the interpolation matrix $\boldsymbol{L}_f$, one does not need to provide explicitly the matrix $\boldsymbol{C}_f$. As a result, Eq. (35) can then be rewritten as

$$\boxed{\boldsymbol{R}_f \underline{\boldsymbol{v}}_f^{n+1} + \boldsymbol{C}_f \underline{\boldsymbol{\Lambda}}^{n+1} = \underline{\boldsymbol{b}}_f}. \tag{36}$$

5.2. Solid coupling equations

5.2.1. Explicit Newmark time integrator

In the present work, we choose the explicit Newmark time integration scheme [57] to update the solid state. The major advantages of this choice are: (i) the solid displacement $\underline{\boldsymbol{u}}_s^{n+1}$ (geometry) can be explicitly updated to the next instant t^{n+1}; (ii) the 'single-step, single-solve' feature of the Newmark scheme [49]; (iii) there is no need to carry out sub-iterations, because the internal nodal force $\underline{\boldsymbol{q}}_{int}^{n+1}$ can be computed once $\underline{\boldsymbol{u}}_s^{n+1}$ is known. Notice that although these can significantly simplify the coupling procedure, the time-step Δt in the solid sub-domain is restricted due to the explicit feature.

In addition, instead of the consistent mass matrix $\boldsymbol{M}_s$ in Eq. (13), diagonal mass matrices $\boldsymbol{M}_s^{diag}$ are used for the sake of numerical stability in explicit dynamical simulations. For linear elements, we simply apply the row-sum method [22, 87] to get the diagonal mass matrix, whereas for quadratic elements, we adopt the diagonal scaling method [39] in order to avoid negative components in the mass matrix.

The explicit Newmark scheme is given as

$$\begin{cases} \underline{\boldsymbol{u}}_s^{n+1} = \underline{\boldsymbol{u}}_s^{n} + \Delta t \underline{\boldsymbol{v}}_s^{n} + \dfrac{\Delta t^2}{2} \underline{\boldsymbol{a}}_s^{n}, \\ \underline{\boldsymbol{v}}_s^{n+1} = \underline{\boldsymbol{v}}_s^{n} + \dfrac{\Delta t}{2} \left(\underline{\boldsymbol{a}}_s^{n+1} + \underline{\boldsymbol{a}}_s^{n} \right), \end{cases} \tag{37}$$

which will be used to solve the solid dynamical equilibrium equation (14) at the instant t^{n+1}

$$\boldsymbol{M}_s^{diag} \underline{\boldsymbol{a}}_s^{n+1} = \underline{\boldsymbol{q}}_{ext}^{n+1} - \underline{\boldsymbol{q}}_{int}^{n+1}. \tag{38}$$

5.2.2. External nodal force

As fluid-structure interactions happen at the fluid-solid interface $\mathit{\Gamma}_i$, the flow-induced force can be considered as an external surface force exerted on the solid structure. In FEM, the external nodal force for the Ith FE-node is computed as

$$\boldsymbol{q}_{ext,I} = \int_{\mathit{\Gamma}_i} N_I \boldsymbol{t} \, \mathrm{d}\mathit{\Gamma} = \sum_{k=1}^{N_i} \int_{\mathit{\Gamma}_i^k} N_I \boldsymbol{t} \, \mathrm{d}\mathit{\Gamma}_i^k = \sum_{k=1}^{N_i} \bar{\boldsymbol{t}}_k \int_{\mathit{\Gamma}_i^k} N_I \, \mathrm{d}\mathit{\Gamma}_i^k, \tag{39}$$

where $\boldsymbol{t}$ denotes the flow-induced surface force (per unit area) exerted on the fluid-solid interface $\mathit{\Gamma}_i$ which is constituted of N_i surface element $\mathit{\Gamma}_i^k$ with $k \in [1, N_i]$, as shown in Fig. 2. Additionally, $\bar{\boldsymbol{t}}_k$ is the surface-average value of $\boldsymbol{t}$, which can be approximated with $\boldsymbol{\lambda}_k$ defined in Sec. 4.2. Substituting $\bar{\boldsymbol{t}}_k \simeq \boldsymbol{\lambda}_k$ into Eq. (39), one may observe that the external nodal force vector $\underline{\boldsymbol{q}}_{ext}$ can be related with the interface force field $\underline{\boldsymbol{\mathit{\Lambda}}}$ as

$$\underline{\boldsymbol{q}}_{ext} = -\boldsymbol{C}_s \underline{\boldsymbol{\mathit{\Lambda}}}, \tag{40}$$

where the matrix $\boldsymbol{C}_s$ depends on the current geometry of the fluid-solid interface.

To obtain the components of the matrix $\boldsymbol{C}_s$, we follow the steps proposed in [55]: (i) impose a force boundary condition where $\boldsymbol{\lambda}_k = \boldsymbol{I}$ and $\boldsymbol{\lambda}_l = \boldsymbol{0}$ with $k, l \in [1, N_i]$ and $l \neq k$; (ii) compute the external nodal force vector $\underline{\boldsymbol{q}}_{ext}^k$ under this condition with Eq. (39); (iii) assign the kth column of the matrix $\boldsymbol{C}_s$ with the obtained $\underline{\boldsymbol{q}}_{ext}^k$. For instance, as shown in Fig. 2, if the kth interface element possesses four nodes, then only four lines have non-zero values in the kth column of $\boldsymbol{C}_s$.

With Eq. (37) and Eq. (40), we shall rewrite Eq. (38) as

$$\underbrace{\frac{2}{\Delta t}\boldsymbol{M}_s^{diag}}_{\boldsymbol{R}_s} \underline{\boldsymbol{v}}_s^{n+1} + \boldsymbol{C}_s \underline{\boldsymbol{\Lambda}}^{n+1} = \underbrace{\frac{2}{\Delta t}\boldsymbol{M}_s^{diag}\left(\underline{\boldsymbol{v}}_s^n + \frac{\Delta t}{2}\underline{\boldsymbol{a}}_s^n\right) - \underline{\boldsymbol{q}}_{int}^{n+1}}_{\underline{\boldsymbol{b}}_s}, \tag{41}$$

or in a more concise form as

$$\boxed{\boldsymbol{R}_s \underline{\boldsymbol{v}}_s^{n+1} + \boldsymbol{C}_s \underline{\boldsymbol{\Lambda}}^{n+1} = \underline{\boldsymbol{b}}_s}. \tag{42}$$

Remark 3. *In the present work, essential boundary conditions (fixed FE-nodes) are imposed by means of the elimination method. To this end, it suffices to set the nth rows of the matrix* $\boldsymbol{C}_s$ *and the vector* $\mathbf{b}_s$ *to zero, where 'n' denotes the number of the node that is fixed in space.*

5.3. *Coupling system of equations and partitioned coupling procedure*

At the fluid-solid interface, the no-slip condition is imposed, which implies

$$\boxed{\boldsymbol{L}_f \underline{\boldsymbol{v}}_f^{n+1} + \boldsymbol{L}_s \underline{\boldsymbol{v}}_s^{n+1} = \underline{\boldsymbol{0}}}, \tag{43}$$

where $\boldsymbol{L}_s$ is an interpolation matrix of dimension $dN_i \times dN_s$, which gives the opposite of the interpolated solid velocity at the center of each interface element.

Now, combining Eq. (36), Eq. (42) and Eq. (43) yields

$$\begin{bmatrix} \boldsymbol{R}_f & \boldsymbol{0} & \boldsymbol{C}_f \\ \boldsymbol{0} & \boldsymbol{R}_s & \boldsymbol{C}_s \\ \boldsymbol{L}_f & \boldsymbol{L}_s & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \underline{\boldsymbol{v}}_f^{n+1} \\ \underline{\boldsymbol{v}}_s^{n+1} \\ \underline{\boldsymbol{\Lambda}}^{n+1} \end{bmatrix} = \begin{bmatrix} \underline{\boldsymbol{b}}_f \\ \underline{\boldsymbol{b}}_s \\ \underline{\boldsymbol{0}} \end{bmatrix}, \tag{44}$$

which is the coupling system of equations for fluid-structure interaction problems by means of the coupling of LBM and FEM via the implicit IBM.

It is here worth noting that the system (44) possesses a large dimension, as the number of unknowns or equations is equal to $d(N_f + N_s + N_i)$. In a monolithic coupling procedure, the whole system of equations is built up and solved in the same solver or code. In the present work, we propose to use a partitioned coupling procedure to solve the system (44), which benefits from the modularities of LBM, IBM and FEM solver packages.

Instead of directly solving the system of equations (44), the proposed partitioned coupling procedure requires some mathematical operations to the system in advance. Multiplying Eq. (36) and Eq. (42) with $\boldsymbol{L}_f\boldsymbol{R}_f^{-1}$ and $\boldsymbol{L}_s\boldsymbol{R}_s^{-1}$, respectively, we can obtain

$$\begin{cases} \boldsymbol{L}_f\boldsymbol{R}_f^{-1}\boldsymbol{R}_f\underline{\boldsymbol{v}}_f^{n+1} + \boldsymbol{L}_f\boldsymbol{R}_f^{-1}\boldsymbol{C}_f\underline{\boldsymbol{\Lambda}}^{n+1} = \boldsymbol{L}_f\boldsymbol{R}_f^{-1}\underline{\boldsymbol{b}}_f, \\ \boldsymbol{L}_s\boldsymbol{R}_s^{-1}\boldsymbol{R}_s\underline{\boldsymbol{v}}_s^{n+1} + \boldsymbol{L}_s\boldsymbol{R}_s^{-1}\boldsymbol{C}_s\underline{\boldsymbol{\Lambda}}^{n+1} = \boldsymbol{L}_s\boldsymbol{R}_s^{-1}\underline{\boldsymbol{b}}_s. \end{cases} \tag{45}$$

Adding together the two equations in Eq. (45) gives

$$\underbrace{\boldsymbol{L}_f\underline{\boldsymbol{v}}_f^{n+1} + \boldsymbol{L}_s\underline{\boldsymbol{v}}_s^{n+1}}_{\underline{\boldsymbol{0}}} + \underbrace{\left(\boldsymbol{L}_f\boldsymbol{R}_f^{-1}\boldsymbol{C}_f + \boldsymbol{L}_s\boldsymbol{R}_s^{-1}\boldsymbol{C}_s\right)}_{\boldsymbol{H}}\underline{\boldsymbol{\Lambda}}^{n+1} = \underbrace{\boldsymbol{L}_f\boldsymbol{R}_f^{-1}\underline{\boldsymbol{b}}_f + \boldsymbol{L}_s\boldsymbol{R}_s^{-1}\underline{\boldsymbol{b}}_s}_{\underline{\boldsymbol{b}}}, \tag{46}$$

where the sum of the first two terms is equal to zero due to the no-slip condition (43). As a result, we obtain a condensed system of equations

$$\boldsymbol{H}\underline{\boldsymbol{\Lambda}}^{n+1} = \underline{\boldsymbol{b}}, \tag{47}$$

in which $\boldsymbol{H} = \boldsymbol{L}_f\boldsymbol{R}_f^{-1}\boldsymbol{C}_f + \boldsymbol{L}_s\boldsymbol{R}_s^{-1}\boldsymbol{C}_s$ is the condensed matrix of size $dN_i \times dN_i$, which is much smaller than the one in the original coupling system of equations (44).

By solving this condensed system of equations, one can obtain the interface force field $\underline{\boldsymbol{\Lambda}}^{n+1}$. To this end, we need to prepare

- $\boldsymbol{L}_s\boldsymbol{R}_s^{-1}\boldsymbol{C}_s$: $\boldsymbol{L}_s$ and $\boldsymbol{R}_s^{-1}$ are both constant, hence they can be calculated only once at the beginning of the simulation. In addition, $\boldsymbol{R}_s$ is diagonal and easy to inverse, due to the use of a diagonal mass matrix, and $\boldsymbol{C}_s$ can be calculated using the method presented previously in Sec. 5.2.2.

- $\boldsymbol{L}_s\boldsymbol{R}_s^{-1}\underline{\boldsymbol{b}}_s$: $\underline{\boldsymbol{b}}_s$ is the vector defined in Eq. (41), which depends partially on the internal nodal force vector $\underline{\boldsymbol{q}}_{int}^{n+1}$. Notice that $\underline{\boldsymbol{q}}_{int}^{n+1}$ can be calculated with Eq. (19) and Eq. (20), once the new displacement $\underline{\boldsymbol{u}}_s^{n+1}$ is updated using the explicit Newmark scheme (37).

- $\boldsymbol{L}_f\boldsymbol{R}_f^{-1}\boldsymbol{C}_f$: As mentioned previously, we do not need to compute explicitly the matrix $\boldsymbol{L}_f$, $\boldsymbol{R}_f^{-1}$ or $\boldsymbol{C}_f$, because by comparing the first equation in Eq. (45) with the implicit IBM formulation (33), we can obtain that $\boldsymbol{L}_f\boldsymbol{R}_f^{-1}\boldsymbol{C}_f = (\Delta t/2)\Delta x_f^d \boldsymbol{A}\boldsymbol{B}^{-1}$.

- $\boldsymbol{L}_f\boldsymbol{R}_f^{-1}\underline{\boldsymbol{b}}_f$: Similarly, from the comparison between Eq. (33) and Eq. (45), we can obtain that $\boldsymbol{L}_f\boldsymbol{R}_f^{-1}\underline{\boldsymbol{b}}_f = \boldsymbol{L}_f\underline{\boldsymbol{v}}_f^*$ which is just the interpolated intermediate velocity on the IB-points.

After solving Eq. (47), the interface force field $\underline{\boldsymbol{\Lambda}}^{n+1}$ will be sent to the fluid and solid solvers in order for them to accomplish the time-step in each sub-domain.

5.4. Summary of the coupling algorithm

A summary of the major steps in the proposed coupling algorithm is provided as follows:

(1) Separate simultaneous computations in the LBM (fluid) and FEM (solid) solvers

  (1.1) LBM solver:

    (1.1.1) Compute the equilibrium distribution function f_α^{eq} with ρ_f^n and $\boldsymbol{v}_f^n$ using Eq. (3)

    (1.1.2) Compute the force-related term $\mathcal{F}_\alpha$ with ρ_f^n, $\boldsymbol{v}_f^n$ and $\boldsymbol{g}^n$ using Eq. (4)

    (1.1.3) Compute the regularized non-equilibrium distribution function $\hat{f}_\alpha^{neq}$ using Eq. (7)

    (1.1.4) Collision and streaming with Eq. (6) to update the distribution function f_α^{n+1}

    (1.1.5) Compute the macroscopic density ρ_f^{n+1} and the intermediate velocity $\boldsymbol{v}_f^*$ with f_α^{n+1}

  (1.2) FEM solver:

    (1.2.1) Update the displacement vector $\underline{\boldsymbol{u}}_s^{n+1}$ with the explicit Newmark scheme (37)

    (1.2.2) Compute the internal nodal force vector $\underline{\boldsymbol{q}}_{int}^{n+1}$ with Eq. (19) and Eq. (20), and then $\underline{\boldsymbol{b}}_s$

(2) Solve the FSI couping system in the coupler

  (2.1) Receive $\underline{\boldsymbol{u}}_s^{n+1}$ from the solid solver and compute the new geometry of the interface

  (2.2) Compute the matrix $\boldsymbol{A}$ in Eq. (26) and the matrix $\boldsymbol{C}_s$ in Eq. (44)

  (2.3) Interpolation stage of the IBM to get $\mathcal{I}[\rho_f^{n+1}]_k$ and $\mathcal{I}[\boldsymbol{v}_f^*]_k$, and then $\boldsymbol{B}^{-1}$, $\boldsymbol{L}_f\underline{\boldsymbol{v}}_f^*$

  (2.4) Compute $\boldsymbol{L}_s\boldsymbol{R}_s^{-1}\boldsymbol{C}_s$, $\boldsymbol{L}_f\boldsymbol{R}_f^{-1}\boldsymbol{C}_f$, $\boldsymbol{L}_s\boldsymbol{R}_s^{-1}\underline{\boldsymbol{b}}_s$ and $\boldsymbol{L}_f\boldsymbol{R}_f^{-1}\underline{\boldsymbol{b}}_f$, and then $\boldsymbol{H}$ and $\underline{\boldsymbol{b}}$ in Eq. (47)

  (2.5) Solve the condensed coupling system of equations (47) to get $\underline{\boldsymbol{\Lambda}}^{n+1}$

  (2.6) Compute $\underline{\boldsymbol{G}}^{n+1} = -\boldsymbol{B}^{-1}\underline{\boldsymbol{\Lambda}}^{n+1}$ and $\underline{\boldsymbol{q}}_{ext}^{n+1} = -\boldsymbol{C}_s\underline{\boldsymbol{\Lambda}}^{n+1}$

(3) Finalize separately the time-steps in the LBM and FEM solvers

  (3.1) LBM solver:

    (3.1.1) Spreading stage of the IBM to get $\boldsymbol{g}^{n+1}$ at the LB-nodes near the interface

    (3.1.2) Update the macroscopic velocity $\boldsymbol{v}_f^{n+1} = \boldsymbol{v}_f^* + \Delta t\boldsymbol{g}^{n+1}/2$

  (3.2) FEM solver:

    (3.2.1) Update the acceleration vector $\underline{\boldsymbol{a}}_s^{n+1} = \left(\boldsymbol{M}_s^{diag}\right)^{-1}\left(\underline{\boldsymbol{q}}_{ext}^{n+1} - \underline{\boldsymbol{q}}_{int}^{n+1}\right)$

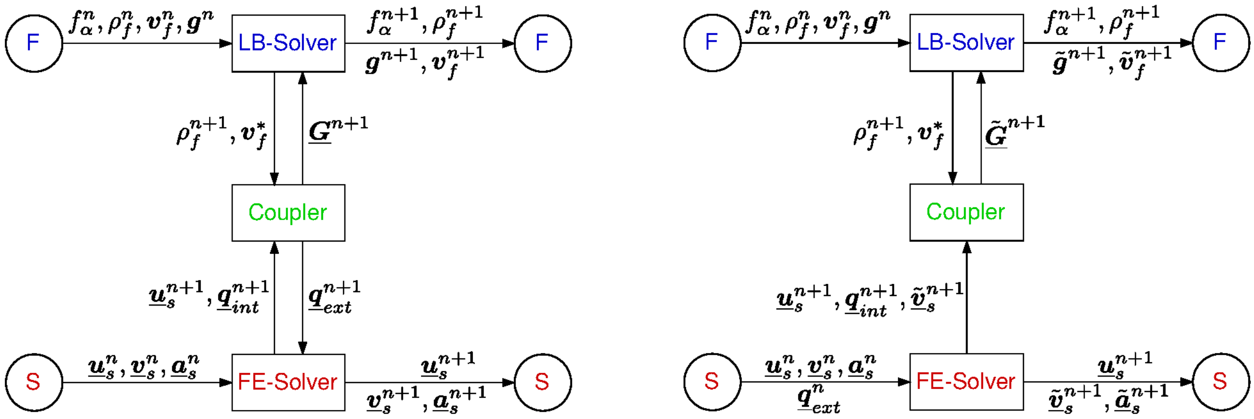
    (3.2.2) Update the velocity vector $\underline{\boldsymbol{v}}_s^{n+1} = \underline{\boldsymbol{v}}_s^n + \Delta t/2\left(\underline{\boldsymbol{a}}_s^{n+1} + \underline{\boldsymbol{a}}_s^n\right)$

5.5. *Comparison with a weakly-coupled algorithm*

It is worth noting that Eq. (44) is a strong coupling system of equations for FSI problems, because the solution to this system of equations satisfies both the velocity- and stress-continuity condition at the fluid-solid interface at each time instant. As shown in the previous summary, compared to a monolithic coupling procedure, the proposed method solves exactly the same system of equations (44), except in a partitioned way where the fluid and solid simulations are carried out separately and simultaneously in two solvers. Hence, the present method can be categorized as a strongly-coupled partitioned coupling algorithm.

To better illustrate the procedure and advantages of the proposed coupling method, we briefly provide here a weakly-coupled partitioned algorithm, in which the coupling system (44) is not solved. Instead, the interface force field $\underline{\boldsymbol{\Lambda}}$ is assumed to be constant during one time-step, so that one can approximately consider that $\underline{\boldsymbol{\Lambda}}^{n+1} \simeq \underline{\boldsymbol{\Lambda}}^{n}$, and thus $\underline{\boldsymbol{q}}_{ext}^{n+1} \simeq \underline{\boldsymbol{q}}_{ext}^{n} = -\boldsymbol{C}_s \underline{\boldsymbol{\Lambda}}^{n}$. This is a typical solution in a weakly-coupled or loosely-coupled algorithm, as the new interface force field $\underline{\boldsymbol{\Lambda}}^{n+1}$ is not available for the FEM solver to compute the external nodal force $\underline{\boldsymbol{q}}_{ext}^{n+1}$ at the instant t^{n+1}. Next, the solid solver can compute an approximate acceleration vector $\underline{\tilde{\boldsymbol{a}}}_s^{n+1}$, and then the velocity vector $\underline{\tilde{\boldsymbol{v}}}_s^{n+1}$. Finally, the fluid solver will use $\underline{\tilde{\boldsymbol{v}}}_s^{n+1}$ as a boundary condition to compute the body force $\tilde{\boldsymbol{g}}^{n+1}$ with the IBM, and finally update the macroscopic velocity $\tilde{\boldsymbol{v}}_f^{n+1}$. This weakly-coupled algorithm is also shown in Fig. 3 and compared with the proposed strongly-coupled algorithm.

Although this weak coupling strategy is simple and retains the modularities of the adopted solvers, it sometimes suffers from numerical instability issues, as the approximation $\underline{\boldsymbol{\Lambda}}^{n+1} \simeq \underline{\boldsymbol{\Lambda}}^{n}$ induces a time-lag between the time integrations of the fluid and solid sub-domains. In other words, the stress-continuity condition is not satisfied at the fluid-solid interface, because the interface force fields $\underline{\boldsymbol{\Lambda}}$ used by the fluid and solid solvers are not the same.



(a) (b)

Figure 3: Comparison between strong (left) and weak (right) coupling algorithms.

In order to improve the numerical stability of this weak coupling strategy, it is possible to add a loop of sub-iterations within each physical time-step. One example of sub-iteration-based coupling algorithms is shown in Fig. 4, in which the solid solver provides a first guess $\underline{\tilde{v}}_s^{n+1}$ of the 'real' solid velocity vector $\underline{v}_s^{n+1}$. After applying the IBM in the coupler, a new external nodal force vector $\underline{q}_{ext}^{\beta+1}$ can be obtained, where β is the current sub-iteration number. Next, one can get a new solid velocity vector $\underline{v}_s^{\beta+1}$ using $\underline{q}_{ext}^{\beta+1}$, which will give a new external nodal force vector. When the residual $\mathcal{R} = ||\underline{q}_{ext}^{\beta+1} - \underline{q}_{ext}^{\beta}||$ is small enough, the sub-iteration-loop stops and the fluid and solid solvers finalize the time-steps in both sub-domains.

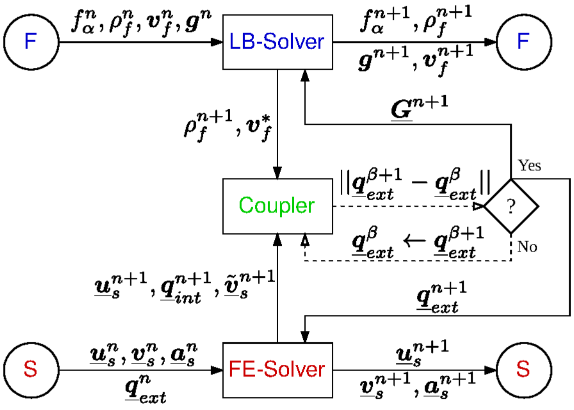


Figure 4: Strongly-coupled algorithm based on sub-iterations.

The sub-iterations within each time-step allow one to couple the fluid and solid solvers in a strong way, which means the velocity- and stress-continuity conditions can be ensured with a controllable precision. Hence, less instability issues will be encountered with sub-iteration-based coupling algorithms. However, as an extra computational cost, the sub-iteration-loop degrades the coupling efficiency, especially with a large number of sub-iterations. Moreover, when the FSI effects are significant or the first guess is far from the solution, the sub-iteration-loop might even not converge.

With these comparisons, we summarize here several features of the proposed coupling method: (i) it is categorized as a strongly-coupled algorithm; (ii) the strong coupling property maintains the numerical stability; (iii) the key point is to build and solve the condensed coupling system of equations; (iv) the modularities of solvers are preserved with the proposed partitioned coupling method; (v) the sub-iteration-free feature makes the coupling procedure more efficient.

6. Coupling strategies with non-conforming time-steps

In the previous section, the proposed strongly-coupled partitioned algorithm was introduced assuming identical time-steps for the fluid and solid solvers, i.e. $\Delta t_f = \Delta t_s$. However, due to the different material properties and heterogeneous features of the numerical schemes, the suitable time-steps for fluid and solid solvers are often different. Some previous works allowing the use of non-conforming time-steps can be found in the literature, e.g. the partitioned coupling between LBM and p-FEM [47], the SPH-FEM coupling algorithm involving free-surface flows [38], etc., however, few of them ensure a strong coupling.

In the following, we shall discuss first how to choose time-steps in the adopted LB- and FE-solvers and then propose a simple strategy for using non-conforming time-steps, based on a linear interpolation in the time domain.

6.1. Choice of time-steps in both sub-domains

6.1.1. Fluid sub-domain simulated with LBM

As recently reviewed in [51], one of the special features of LBM is that the physical lattice-spacing Δx_f^{phy} and the time-step Δt_f^{phy} are tied up in a way that the lattice speeds $\boldsymbol{\xi}_\alpha$ are directly related with the ratio $\Delta x_f^{phy}/\Delta t_f^{phy}$. In a LB-solver, the space and time rescaling factors are usually chosen as $\mathcal{C}_x = \Delta x_f^{phy}$ and $\mathcal{C}_t = \Delta t_f^{phy}$ such that the rescaled lattice-spacing and time-step are both equal to one, i.e. $\Delta x_f^{lat} = \Delta x_f^{phy}/\mathcal{C}_x = 1$ and $\Delta t_f^{lat} = \Delta t_f^{phy}/\mathcal{C}_t = 1$. This limits, to some extent, the choice of the time-step in LB solvers. Although several solutions such as the multi-grid techniques [33] have been proposed in the literature to overcome this constraint, the present paper focuses on providing a coupling strategy using non-conforming time-steps with the classic LBM.

Given $\mathcal{C}_x$ and $\mathcal{C}_t$, the characteristic velocity in the lattice scale U^{lat} is related to the physical one U^{phy} as $U^{lat} = U^{phy}/\mathcal{C}_v$, where $\mathcal{C}_v = \mathcal{C}_x/\mathcal{C}_t$ denotes the velocity rescaling factor. Additionally, the characteristic length L^{lat} in the lattice scale represents the number of LB-nodes used to discretize the physical characteristic length L^{phy}, as they are related as $L^{lat} = L^{phy}/\mathcal{C}_x = L^{phy}/\Delta x_f^{phy}$. Finally, the kinematic viscosity is computed as $\nu^{lat} = (c_f^{lat})^2(\tau^{lat} - 0.5)$, where $c_f^{lat} = 1/\sqrt{3}$ and τ^{lat} denote respectively the speed of sound and the relaxation time parameter in the lattice scale.

The stability condition for LBM requires that the relaxation time parameter τ^{lat} should be greater than 0.5. In practical simulations, τ^{lat} is usually tuned to impose the desired Re number, satisfying a constraint condition that $\tau^{lat} \geq \tau_{cri}^{lat}$ with τ_{cri}^{lat} being a critical value not too close to 0.5. With $Re = U^{lat}L^{lat}/\nu^{lat}$, this stability condition can be rewritten as

$$\tau^{lat} = \frac{3U^{lat}L^{lat}}{Re} + 0.5 = \frac{3(U^{phy}/\mathcal{C}_v)(L^{phy}/\mathcal{C}_x)}{Re} + 0.5 = \frac{3U^{phy}L^{phy}}{Re(\Delta x_f^{phy})^2}\Delta t_f^{phy} + 0.5 \geq \tau_{cri}^{lat}, \tag{48}$$

which indicates that, given a physical problem and a lattice (i.e. Re, U^{phy}, L^{phy} and Δx_f^{phy} are fixed), the physical time-step in LBM Δt_f^{phy} is limited as

$$\Delta t_f^{phy} \geq \frac{(\tau_{cri}^{lat} - 0.5)Re(\Delta x_f^{phy})^2}{3U^{phy}L^{phy}}. \tag{49}$$

Moreover, for weakly-compressible isothermal LB schemes, the Mach number $Ma = U^{lat}/c_f^{lat}$ in the lattice scale should be small enough to satisfy the weakly-compressible assumption. For instance, one may consider that the variation of density is negligible when $Ma \leq 0.1$, which implies that

$$Ma = \sqrt{3}U^{lat} = \sqrt{3}\frac{U^{phy}}{\Delta x_f^{phy}}\Delta t_f^{phy} \leq 0.1, \tag{50}$$

which is another constraint condition when choosing the physical time-step in a LB-solver.

Combining Eq. (49) and Eq. (50) gives

$$\frac{(\tau_{cri}^{lat} - 0.5)Re(\Delta x_f^{phy})^2}{3U^{phy}L^{phy}} \leq \Delta t_f^{phy} \leq \frac{0.1\Delta x_f^{phy}}{\sqrt{3}U^{phy}}, \tag{51}$$

which is the range of the acceptable time-step used in the adopted LB-solver.

6.1.2. Solid sub-domain simulated with FEM

The explicit Newmark time scheme is stable under a CFL-like condition

$$\Delta t_s \leq \mathcal{K}_c \frac{\Delta x_s^{min}}{\sqrt{E_s/\rho_s^0}}, \tag{52}$$

where $\mathcal{K}_c \in [0, 1]$ denotes the CFL coefficient and Δx_s^{min} is the minimal distance between two FE-nodes of the solid mesh.

As shown in Eq. (51) and Eq. (52), the time-steps in LBM and FEM are limited by different criteria, as a consequence, it is not always easy to use the same time-step in both solvers. In addition, depending on the physical problem solved, characteristic time variations could be of different scales in the two media, thus leading to very different optimal time-step choices in the two solvers. Hence, we propose subsequently two simple strategies based on linear interpolation in time allowing the use of non-conforming time-steps ($\Delta t_f \neq \Delta t_s$), which make the proposed coupling algorithm more flexible and efficient.

6.2. Coupling strategy for $\Delta t_f < \Delta t_s$

Due to the numerical limitations or different physical characteristics mentioned above, if one finds that the solid time-step Δt_s is larger than the fluid time-step Δt_f ($\Delta t_s = r\Delta t_f$ with $r \in \mathbb{Z}$ and $r > 1$), it is not necessary to carry out the FEM computation using the smaller time-step.

To save the computational resources, we propose to replace the FEM computation by a linear interpolation in time. To do so, we need to firstly update the new displacement vector $\underline{\boldsymbol{u}}_s^{n+1}$ using the explicit Newmark scheme (37) and the solid time-step Δt_s, based on which we can calculate the internal nodal force vector $\underline{\boldsymbol{q}}_{int}^{n+1}$. Next, we shall assume that both $\underline{\boldsymbol{u}}_s$ and $\underline{\boldsymbol{q}}_{int}$ vary linearly in time, which is equivalent to ignore the higher order terms in the Taylor expansion in time. As a consequence, we can obtain a reasonable approximation of $\underline{\boldsymbol{u}}_s$ and $\underline{\boldsymbol{q}}_{int}$ at the mth sub-instant within the time interval Δt_s as

$$\begin{cases} \underline{\boldsymbol{u}}_s^{n+m/r} = \underline{\boldsymbol{u}}_s^{n} + \dfrac{m}{r}\left(\underline{\boldsymbol{u}}_s^{n+1} - \underline{\boldsymbol{u}}_s^{n}\right), \\ \underline{\boldsymbol{q}}_{int}^{n+m/r} = \underline{\boldsymbol{q}}_{int}^{n} + \dfrac{m}{r}\left(\underline{\boldsymbol{q}}_{int}^{n+1} - \underline{\boldsymbol{q}}_{int}^{n}\right), \end{cases} \tag{53}$$

where $m \in \mathbb{Z}$ and $1 \le m \le r$.

With Eq. (53), the solid coupling equation (42) can be established at the mth sub-instant as

$$\boldsymbol{R}_s^m \underline{\boldsymbol{v}}_s^{n+m/r} + \boldsymbol{C}_s^m \underline{\boldsymbol{\Lambda}}^{n+m/r} = \underline{\boldsymbol{b}}_s^m, \tag{54}$$

with

$$\begin{cases} \boldsymbol{R}_s^m = \dfrac{2}{\Delta t_s^m}\boldsymbol{M}_s^{diag}, \\ \underline{\boldsymbol{b}}_s^m = \dfrac{2}{\Delta t_s^m}\boldsymbol{M}_s^{diag}\left(\underline{\boldsymbol{v}}_s^n + \dfrac{\Delta t_s^m}{2}\underline{\boldsymbol{a}}_s^n\right) - \underline{\boldsymbol{q}}_{int}^{n+m/r}, \end{cases} \tag{55}$$

where $\Delta t_s^m = m\Delta t_s/r$ and the matrix $\boldsymbol{C}_s^m$ can be obtained with $\underline{\boldsymbol{u}}_s^{n+m/r}$.

On the other hand, the fluid coupling equation (36) and the velocity condition (43) remain the same, except that they are established at the mth sub-instant with Δt_f, e.g. $\boldsymbol{L}_f \boldsymbol{R}_f^{-1} \boldsymbol{C}_f = (\Delta t_f/2)\Delta x_f^d \boldsymbol{A}\boldsymbol{B}^{-1}$. By solving the coupling system, we can obtain the interface force field $\underline{\boldsymbol{\Lambda}}^{n+m/r}$ and $\underline{\boldsymbol{G}}^{n+m/r}$ at each sub-instant, which is then used at the spreading stage to accomplish the time-steps in the fluid sub-domain. It is important to note that $\underline{\boldsymbol{\Lambda}}^{n+m/r}$ is used to update the solid velocity $\underline{\boldsymbol{v}}_s^{n+1}$ and acceleration $\underline{\boldsymbol{a}}_s^{n+1}$ vectors only at the last sub-step, i.e. when $m = r$.

6.3. *Coupling strategy for $\Delta t_f > \Delta t_s$*

Conversely, if the fluid time-step Δt_f is greater than the solid time-step Δt_s ($\Delta t_f = r\Delta t_s$ with $r \in \mathbb{Z}$ and $r > 1$), it is then unnecessary and sometimes even impossible to carry out the LB computations using the smaller time-step.

In this case, the fluid coupling equation (36) is established at the mth sub-instant as

$$\boldsymbol{R}_f^m \underline{\boldsymbol{v}}_f^{n+m/r} + \boldsymbol{C}_f^m \underline{\boldsymbol{\Lambda}}^{n+m/r} = \underline{\boldsymbol{b}}_f^m, \tag{56}$$

in which $\boldsymbol{R}_f^m$, $\boldsymbol{C}_f^m$ and $\underline{\boldsymbol{b}}_f^m$ are not explicitly required. Instead, we need to compute

$$\begin{cases} \boldsymbol{L}_f\left(\boldsymbol{R}_f^m\right)^{-1}\boldsymbol{C}_f^m = \dfrac{\Delta t_f}{2}\Delta x_f^d \boldsymbol{A}\boldsymbol{B}_m^{-1}, \\ \boldsymbol{L}_f\left(\boldsymbol{R}_f^m\right)^{-1}\underline{\boldsymbol{b}}_f^m = \boldsymbol{L}_f \underline{\boldsymbol{v}}_f^{*,m}, \end{cases} \tag{57}$$

where $\boldsymbol{B}_m^{-1}$ and $\boldsymbol{L}_f \underline{\boldsymbol{v}}_f^{*,m}$ require the fluid density $\rho_f^{n+m/r}$ and the intermediate velocity $\boldsymbol{v}_f^{*,n+m/r}$ at the mth sub-instant, which are obtained by means of a linear interpolation in time as

$$\begin{cases} \rho_f^{n+m/r} = \rho_f^n + \dfrac{m}{r}\left(\rho_f^{n+1} - \rho_f^n\right), \\ \boldsymbol{v}_f^{*,n+m/r} = \boldsymbol{v}_f^{*,n} + \dfrac{m}{r}\left(\boldsymbol{v}_f^{*,n+1} - \boldsymbol{v}_f^{*,n}\right), \end{cases} \tag{58}$$

in which ρ_f^{n+1} and $\boldsymbol{v}_f^{*,n+1}$ can be calculated, once the distribution function f_α^{n+1} is updated with the streaming step (1.1.4) in Section 5.4.

The solid coupling equation (42) and the velocity condition (43) remain the same and are established at the mth sub-instant with Δt_s. Solving the coupling system gives the interface force field $\underline{\boldsymbol{\Lambda}}^{n+m/r}$, which is used to compute the external nodal force $\underline{\boldsymbol{q}}_{ext}^{n+m/r}$ in order for the FE-solver to update the solid state. Similarly, $\underline{\boldsymbol{\Lambda}}^{n+m/r}$ is used to update the fluid macroscopic velocity $\boldsymbol{v}_f^{n+1}$ only at the last sub-step, i.e. when $m = r$.

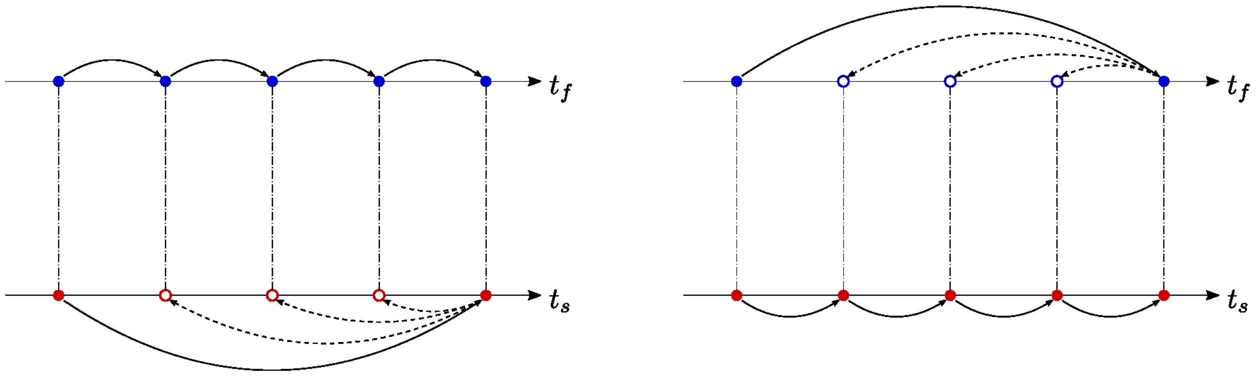
The two proposed strategies in Sections 6.2 and 6.3 are graphically illustrated in Fig. 5, in which the ratio of time-steps is equal to 4 as an example. The solid lines represent the LBM and FEM computations (filled circles), whereas the dashed lines correspond to the linear interpolations in time (empty circles), and the dash-dot lines denote the solution of the FSI coupling equations.

In summary, first, these two strategies allow the use of non-conforming time-steps, which makes the proposed framework more flexible for coupling LBM and FEM for FSI simulations. Second, because costly computations, for instance, the collision-streaming steps in LBM for the whole fluid sub-domain and the calculation of the internal nodal force vector in FEM, are replaced by fast linear interpolations, the proposed strategies can help to save computational resources in the FSI simulations with non-conforming time-steps. Third, thanks to the FSI coupling at each sub-instant, the numerical stability can be retained so long as the variation in time of the interpolated variables are almost linear. In other words, if the ratio of time-steps is too large or the simulated physical problem contains sudden changes in time, the coupling scheme is no longer strong and might encounter instabilities. In this situation, the conservative feature of the force formulation presented in Sec. 4.2 can no longer be guaranteed either. However, the maximal value of the ratio highly depends on each investigated test-case. Hence, it is impossible to find out a universal maximal ratio and it does not make much sense to attempt to obtain the maximal ratio in a specific test-case, as the coupling scheme might fail to remain stable with this maximal ratio in other test-cases.

7. Numerical validations and applications

7.1. Rigid mechanical heart valve

The first validation test-case is about the transient interaction between a rigid mechanic heart valve and the blood flow, which was initially introduced by Stijnen et al. [70] in order to evaluate a fictitious domain method

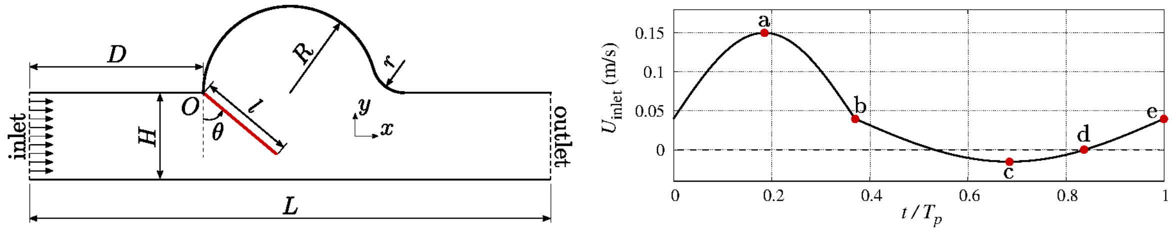


(a) (b)

Figure 5: Coupling strategies for non-conforming time-steps based on linear interpolation in time: (a) $\Delta t_f < \Delta t_s$; (b) $\Delta t_f > \Delta t_s$.

by means of comparison with experimental measurements. This test-case has been adopted in several previous works [23, 27, 30, 83] for validating different numerical methods, e.g. an ALE-dynamic mesh formulation in [23] and an iterative IB-LB coupling scheme in [83].

7.1.1. Configuration and numerical setup



(a) (b)

Figure 6: Rigid valve test-case: (a) geometric configuration; (b) inlet velocity boundary condition.

The configuration of this test-case is shown in Fig. 6-(a), where the fluid flows through a 2D channel of height $H = 0.02$ m and length $L = 6H$. A cylindrical cavity of radius $R = H$ is located halfway along the length of the channel, with a distance from the inlet $D = 2H$, which represents the sinus cavity of the aortic valve. The downstream side of the cavity is connected to the outflow channel with a rounded transition of radius $r = 0.0075$ m. A rigid valve leaflet of length $l = 1.07H$ is installed at the point O on the upstream side of the cavity with an initial inclination angle of $\theta_0 = 22°$. In the experiment carried out in [70], the rigid leaflet has a thickness of $t = 0.001$ m and a density of $\rho_s = 1100$ kg/m^3, which gives the moment of inertia $I_O = \rho_s l^3 t/3 \simeq 3.9 \times 10^{-6}$ kg.m^2 with respect to the point O. The dynamic equilibrium equation for the rotating rigid valve is written as

$$I_O\ddot{\theta} = M_O, \tag{59}$$

where $\ddot{\theta} = \ddot{\theta}(t) = \mathrm{d}\dot{\theta}/\mathrm{d}t$ and $\dot{\theta} = \dot{\theta}(t) = \mathrm{d}\theta/\mathrm{d}t$ denote the first and second derivatives in time of the inclination angle $\theta = \theta(t)$, respectively. Additionally, M_O is the total moment of force exerted on the immersed moving solid valve $\varGamma_i$, which is defined as

$$M_O = \boldsymbol{e}_z \cdot \left(\int_{\varGamma_i} \boldsymbol{p} \times \boldsymbol{\lambda}\, \mathrm{d}\varGamma \right), \tag{60}$$

where $\boldsymbol{e}_z$ denotes the unit vector of the rotation axis, $\boldsymbol{p} = \boldsymbol{x} - \boldsymbol{x}_O$ with $\boldsymbol{x}_O$ being the position of the point O, and $\boldsymbol{\lambda}$ is the interface force per unit length in this 2D test-case. The total moment of force M_O can be numerically computed as

$$M_O \simeq \boldsymbol{e}_z \cdot \left(\sum_{k=1}^{N_i} \boldsymbol{p}_k \times \boldsymbol{\lambda}_k \Delta s_k \right) = \sum_{k=1}^{N_i} \left(p_k^x \lambda_k^y - p_k^y \lambda_k^x \right) \Delta s_k = -\boldsymbol{C}_s \underline{\boldsymbol{\varLambda}}, \tag{61}$$

where $\boldsymbol{p}_k$, $\boldsymbol{\lambda}_k$ and Δs_k denote the relative vector, the local interface force per unit length and the length of the kth IB-segment, respectively. Moreover, the matrices $\boldsymbol{C}_s$ and $\underline{\boldsymbol{\varLambda}}$ are expressed as

$$\begin{cases} \boldsymbol{C}_s = \begin{bmatrix} p_1^y \Delta s_1 & -p_1^x \Delta s_1 & p_2^y \Delta s_2 & -p_2^x \Delta s_2 & \dots & p_{N_i}^y \Delta s_{N_i} & -p_{N_i}^x \Delta s_{N_i} \end{bmatrix}, \\ \underline{\boldsymbol{\varLambda}} = \begin{bmatrix} \lambda_1^x & \lambda_1^y & \lambda_2^x & \lambda_2^y & \dots & \lambda_{N_i}^x & \lambda_{N_i}^y \end{bmatrix}^\top. \end{cases} \tag{62}$$

As for the modeling of the blood flow, the fluid has a density of $\rho_f = 1090$ kg/m^3 and a kinematic viscosity of $\nu_f = 4 \times 10^{-6}$ m^2/s. At the inlet of the channel, a time-varying uniform velocity profile is imposed as

$$U_{\text{inlet}}(t) = \begin{cases} U_{avg} + U_{amp} \sin\left(\dfrac{2\pi t/T_p}{0.74} \right), & \text{if } 0 < t/T_p \leq 0.37, \\ U_{avg} + \dfrac{U_{amp}}{2} \sin\left(\dfrac{2\pi (t/T_p + 0.26)}{1.26} \right), & \text{if } 0.37 < t/T_p \leq 1, \end{cases} \tag{63}$$

where $U_{avg} = 0.04$ m/s and $U_{amp} = 0.11$ m/s denote respectively the average value and amplitude of the inlet velocity during the time period $T_p = 2.45$ s, which give a peak value of $U_{max} = 0.15$ m/s. Meanwhile, Fig. 6-(b) shows the time evolution of the inlet velocity during one period. These parameters lead to a Reynolds number of $Re = U_{max} l/\nu_f \simeq 750$ and a Strouhal number of $St = H/(U_{max} T_p) \simeq 0.055$.

The numerical velocity inlet and pressure outlet boundary conditions are imposed by means of the Zou-He's method [88]. For the lower and upper fixed rigid solid walls, we adopt the linear interpolated bounce-back scheme [7] with the mass-conserving correction [82].

The objective of using this test-case is to validate the capability of the proposed coupling framework for handling FSI problems with rigid solid bodies. To this end, it suffices to modify the solid coupling equations, while retaining the fluid coupling equations and the solving procedure as presented in Sec. 5. Combined with the explicit Newmark scheme (37), the solid dynamic equilibrium equation (59) can be written at the instant t^{n+1} as

$$\begin{cases} I_O\ddot{\theta}^{n+1} = M_O^{n+1}, \\ \theta^{n+1} = \theta^n + \Delta t_s\dot{\theta}^n + \dfrac{\Delta t_s^2}{2}\ddot{\theta}^n, \\ \dot{\theta}^{n+1} = \dot{\theta}^n + \dfrac{\Delta t_s}{2}\left(\ddot{\theta}^{n+1} + \ddot{\theta}^n\right), \end{cases} \tag{64}$$

which can then be rewritten in a concise form as

$$\underbrace{\left(\frac{2}{\Delta t_s}I_O\right)}_{R_s}\dot{\theta}^{n+1} + \boldsymbol{C}_s\underline{\boldsymbol{\Lambda}}^{n+1} = \underbrace{I_O\left(\frac{2}{\Delta t_s}\dot{\theta}^n + \ddot{\theta}^n\right)}_{b_s}. \tag{65}$$

It is here worth noting that Eq. (65) has the same form as Eq. (42), hence the FSI-solving procedure remains the same as the one presented in Eqs. (46) and (47). In order to compute the condensed matrix $\boldsymbol{H}$, we need to prepare the operator $\boldsymbol{L}_s$ relating the solid velocity vector $\underline{\boldsymbol{v}}_s$ at all IB-points to the rotation speed $\dot{\theta}$ as $\underline{\boldsymbol{v}}_s = -\boldsymbol{L}_s\dot{\theta}$ with $\boldsymbol{L}_s = [p_1^y, -p_1^x, p_2^y, -p_2^x, \ldots, p_{N_i}^y, -p_{N_i}^x]^\top$.

7.1.2. Numerical results, mesh convergence and comparison with references

We first carry out a mesh convergence study using three lattice resolutions: $H = 80\Delta x_f$, $100\Delta x_f$ and $160\Delta x_f$. During the mesh convergence, the number of IB-points is changed accordingly in order to maintain a constant ratio between Δx_s and Δx_f. Based on the suggestion given in Remark 2, we choose to use a value of $\Delta x_s/\Delta x_f \simeq 1.11$ for all the three lattice resolutions.

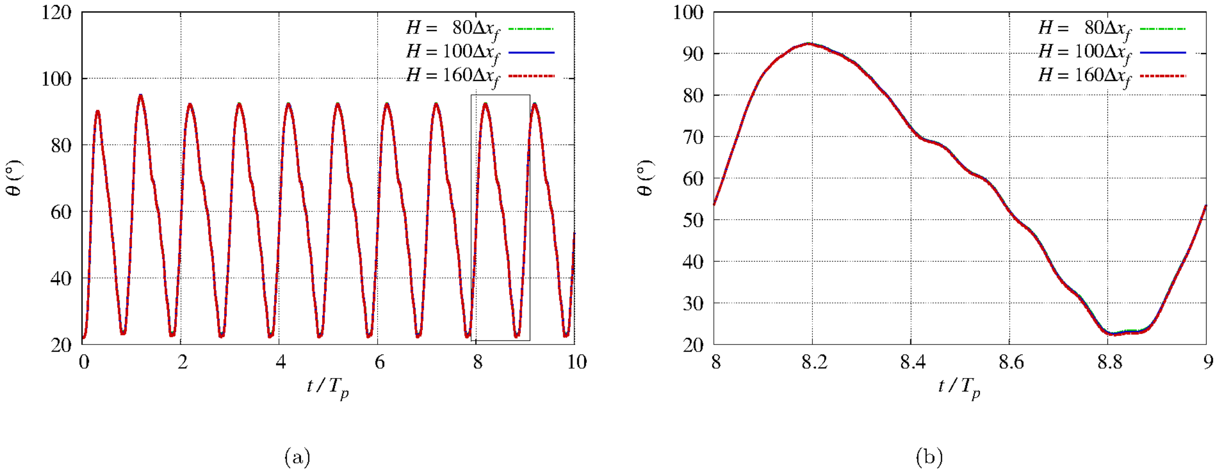
Initially, the fluid and solid sub-domains are both at rest. However, as shown in Fig. 6-(b), the inlet velocity is not equal to zero at $t = 0$. Hence, in order to avoid sudden changes of boundary condition, we multiply the inlet velocity U_{inlet} with a coefficient $\mathcal{K}_{\text{inlet}}$, which is set as

$$\mathcal{K}_{\text{inlet}} = \begin{cases} \dfrac{1}{2} - \dfrac{1}{2}\cos\left(\dfrac{\pi t}{T_{ini}}\right), & \text{if } t \le T_{ini}, \\ 1, & \text{if } t > T_{ini}, \end{cases} \tag{66}$$

where $T_{ini} = 1$ s is the duration of the starting stage.

The time evolutions of the inclination angle $\theta(t)$ are given in Fig. 7, from which one may observe that the movement of the rigid valve leaflet becomes stabilized (periodic) after two periods ($t/T_p > 2$). In the zoomed area during one period, the three lattice resolutions give very similar results, which means that the numerical simulations converge, as the lattice is refined and that visual-accuracy convergence is already obtained. We will use the lattice resolution $H = 100\Delta x_f$ in the following analysis and comparisons.

A comparison of the time evolution of the inclination angle $\theta(t)$ between the present result and some references is shown in Fig. 8. In this comparison, we observe that all the numerical results agree well with



(a)

(b)

Figure 7: Mesh convergence study in the rigid valve test-case: (a) time evolutions of the inclination angle; (b) zoomed area for one period.

each other, although different numerical methods have been applied: a fictitious domain method in [70] and an ALE formulation in [23]. However, an obvious discrepancy can be observed between the numerical results and the experimental measurement carried out in [70]. As commented in [70], this mismatch may be due to the leakage of fluid over the top of the valve between the strips of adhesive tape that are used to fix the valve leaflet.

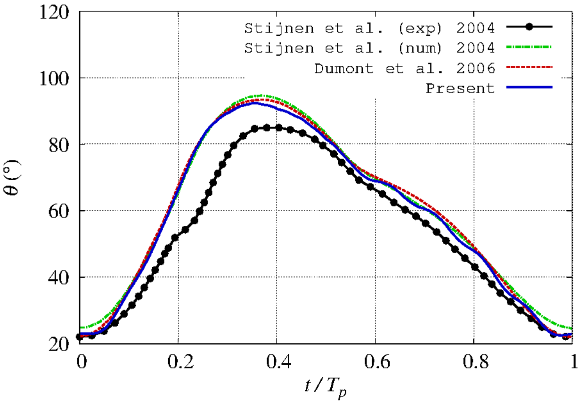


Figure 8: Comparison of the evolution in time of the angle $\theta(t)$ in the rigid valve test-case.

Moreover, the velocity field in the fluid domain is compared in Fig. 9 between the present numerical result and the experimental PIV measurement carried out in [70]. The discrepancy mentioned previously can be easily seen in the sub-figure (a) corresponding to the moment when the valve is fully opened. Apart from this discrepancy, the present numerical result agrees well with the PIV measurement, for instance, the double-vortex flow pattern has been successfully captured as shown in the sub-figures (c) and (d).

Interestingly, in the present numerical simulations, we observe that the rigid valve leaflet stops itself due to the hydrodynamic force before touching the lower solid wall. Hence, no contact model was needed in this test-case.

7.1.3. Strong coupling vs. weak coupling

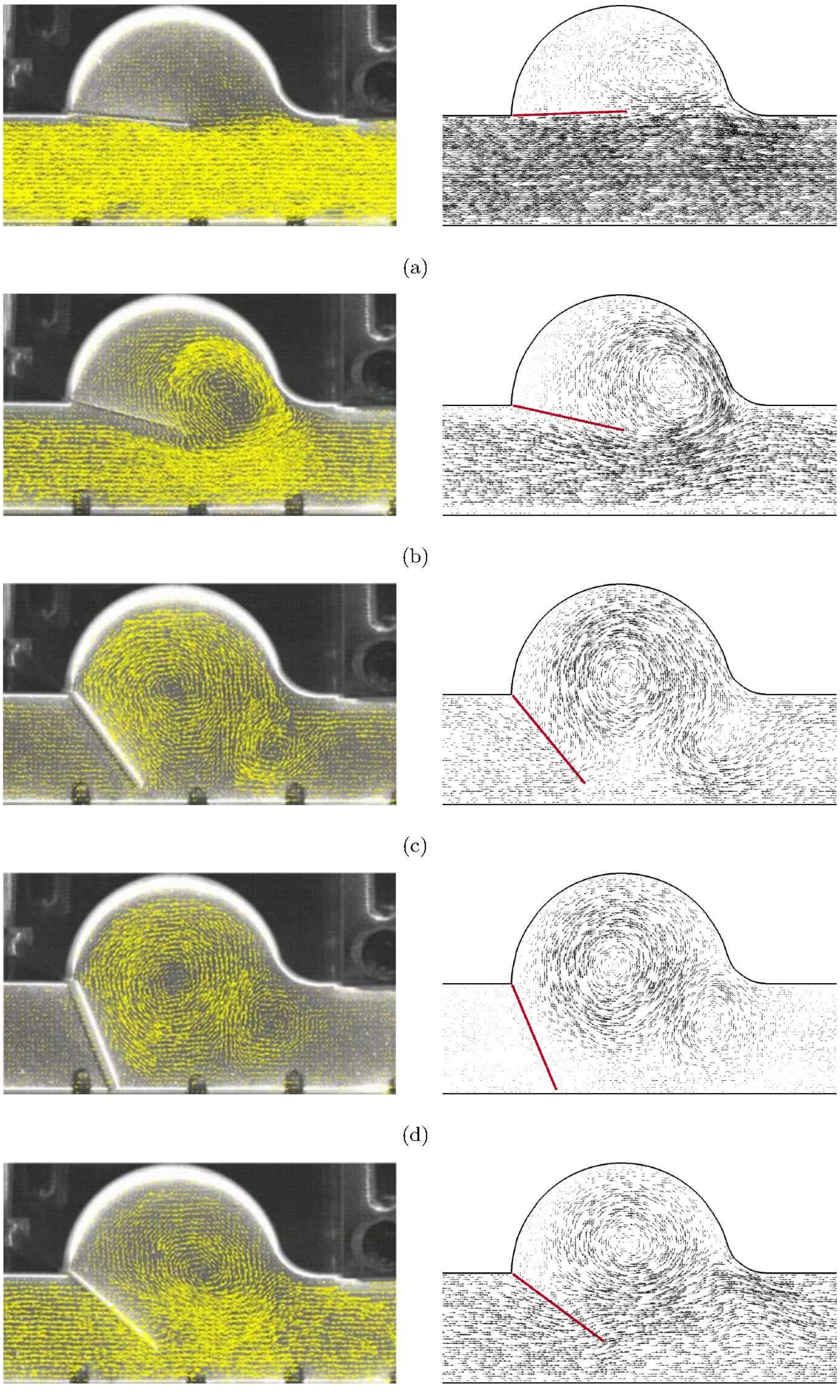
To show the importance of the strong coupling feature, we shall compare the proposed strong coupling procedure with a weak one, such as the weak coupling algorithm presented in Fig. 3-(b). In this case, the total moment is assumed to be constant during one time-step, i.e. $M_O^{n+1} \simeq M_O^n$, which allows the coupler to get an approximate body force vector $\tilde{\underline{\boldsymbol{G}}}^{n+1}$ for the LB-solver to update the macroscopic velocity after the spreading step of the implicit IBM.

The numerical result obtained with this ordinary weak coupling algorithm (without synchronization technique or sub-iterations) is compared with the one obtained using the present strong coupling algorithm in Fig. 10-(a). Some numerical oscillations appeared after about 1500 time-steps with the weak coupling algorithm, which finally induced the divergence of the simulation due to the numerical instability. In the meantime, the present strong coupling algorithm remained stable during the whole period of the numerical simulation. It is here worth noting that the density ratio between solid and fluid in the present test-case is almost equal to one, i.e. $\rho_s/\rho_f \simeq 1$, which means the FSI effect cannot be ignored in both sub-domains and a weak coupling algorithm may often suffer from numerical instabilities.

As presented in Sec. 5.5, one may add sub-iterations within each time-step, as shown in Fig. 4, in order to improve the numerical stability. In the present test-case, we define a residual $\mathcal{R} = |M_O^{\beta+1} - M_O^{\beta}|$ with β being the current sub-iteration number. Notice that the first guess M_O^0 is equal to M_O^n and $M_O^{\beta+1}$ is obtained at the interpolation stage of the IBM using the rotation speed $\dot{\theta}^{\beta+1}$. When the residual $\mathcal{R}$ is sufficiently small, the procedure becomes a strong coupling algorithm based on sub-iterations. In the present work, we choose to stop the sub-iterations when $\mathcal{R}/(0.5\rho_f U_{max}^2 H^2) < 10^{-10}$.

Thanks to the sub-iterations, the numerical simulation is stable during the whole period and the result is extremely close to the one of the proposed sub-iteration-free strong coupling algorithm. However, as shown in Fig. 10-(b), the number of sub-iterations is around 12 for each time-step, which degrades the coupling efficiency due to the extra cost related to the computations during sub-iterations and the communications between both solvers.

In summary, this test-case showed that (i) the proposed coupling framework can also be applied to simulate FSI problems involving rigid solid bodies; (ii) a good agreement has been found between the present result and



(a)

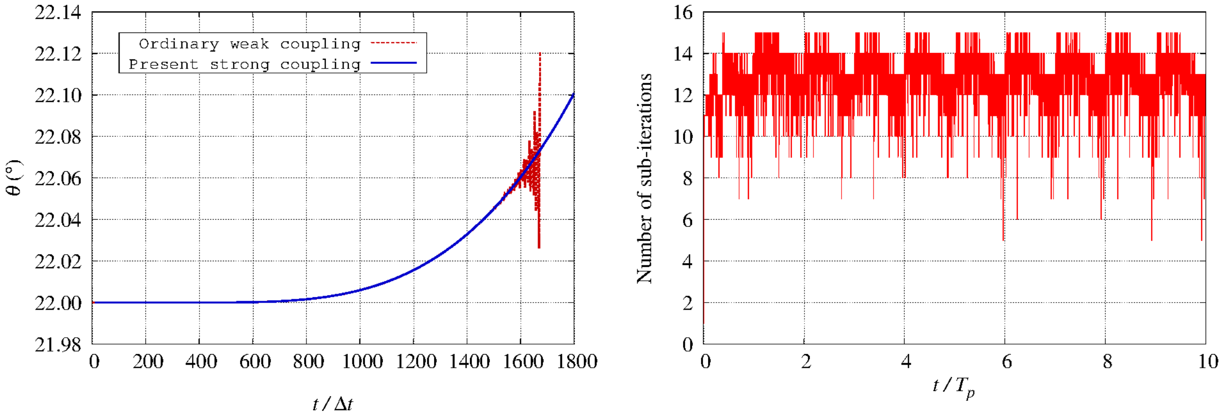
(b)

(c)

(d)

(e)

Figure 9: Comparison of the velocity field between the present numerical result and the experimental PIV measurement [70] in the rigid valve test-case (the figures in [70] are reprinted with permission from the publisher). The sub-figures from (a) to (e) correspond to the five instants indicated in Fig. 6-(b).



(a) (b)

Figure 10: Rigid valve test-case: (a) comparison between the strong and weak coupling results; (b) number of sub-iterations within each time-step for the sub-iteration-based strong coupling algorithm.

the ones obtained using different numerical methods in the references; (iii) the strong coupling feature maintains the numerical stability, while avoiding sub-iterations. Notice that, in the following, all strong coupling results are carried out with the proposed algorithm without sub-iterations.

7.2. 2D flow-induced vibration of an elastic beam

Next, we choose to validate the proposed coupling framework in the presence of deformable solid structures with a widely used benchmark, which was initially introduced by Turek & Hron [75].

7.2.1. Configuration and numerical setup

A fluid flow is simulated within a 2D channel of length $L = 2.5$ m and height $H = 0.41$ m. A fixed rigid cylinder of diameter $D = 0.1$ m is located 0.2 m from the inlet of the channel. A deformable solid beam of length $l = 0.35$ m and thickness $h = 0.02$ m is attached behind the cylinder. A parabolic velocity profile is imposed at the inlet and the pressure is kept constant at the outlet of the channel. The velocity inlet, pressure outlet and no-slip wall boundary conditions are imposed with the Zou-He's method [88].

In the present work, we have adopted the FSI2 ($Re = 100$) and FSI3 ($Re = 200$) test-cases in [75], in which the Reynolds number is defined as $Re = U_0 D/\nu_f$ with $U_0 = 1$ m/s (FSI2) and 2 m/s (FSI3) being the mean inlet velocity. In addition, the material parameters are provided in Table 1. The fluid flow is simulated by the LBM and the solid deformable beam is modeled using the FEM with the Saint Venant-Kirchhoff constitutive model (15).

As for the numerical discretization, the fluid sub-domain is discretized using a lattice of size 1251×206

Table 1: Material parameters in the 2D flow-induced vibration test-case.

	FSI2 ($Re = 100$)	FSI3 ($Re = 200$)
Initial fluid density (ρ_f^0)	10^3 kg/m^3	10^3 kg/m^3
Fluid kinematic viscosity (ν_f)	10^{-3} m^2/s	10^{-3} m^2/s
Initial solid density (ρ_s^0)	10^4 kg/m^3	10^3 kg/m^3
Solid Poisson ratio (ν_s)	0.4	0.4
Solid Young's modulus (E_s)	1.4×10^6 Pa	5.6×10^6 Pa
Solid-to-fluid density ratio (ρ_s^0/ρ_f^0)	10	1

LB-nodes for both of the FSI2 and FSI3 test-cases. The solid beam is discretized using 160×9 linear FE-elements. Based on the criteria presented in Sec. 6.1, the time-steps are chosen as $\Delta t_s = \Delta t_f = 10^{-4}$ s and 1.6×10^{-5} s in the FSI2 and FSI3 test-cases, respectively. The CFL numbers in Eq. (52) for the solid sub-domain are approximately equal to 0.6 for both of the FSI2 and FSI3 test-cases.

7.2.2. Numerical results and comparisons with references

The snapshots of the vorticity fields in the FSI2 and FSI3 test-cases are shown in Fig. 11, in which one may observe that the vortex intensity in FSI3 is higher than in FSI2. Moreover, the time evolutions of the vertical tip displacement u_y of the beam are shown and compared with the reference [75] in Fig. 12. Finally, a quantitative comparison with several references [4, 74, 75] is given in Table 2, in which $A = (\max(u_y) - \min(u_y))/2$ denotes the oscillation amplitude, $St = fD/U_0$ is the Strouhal number with f being the oscillation frequency, and $\bar{C}_D$ is the average drag coefficient. As shown in Fig. 12 and Table 2, good agreements can be found with the references for both the FSI2 and FSI3 test-cases.

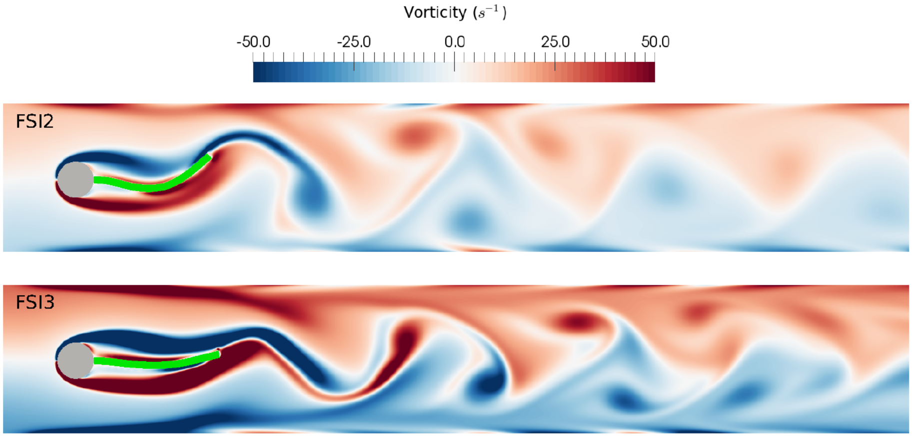
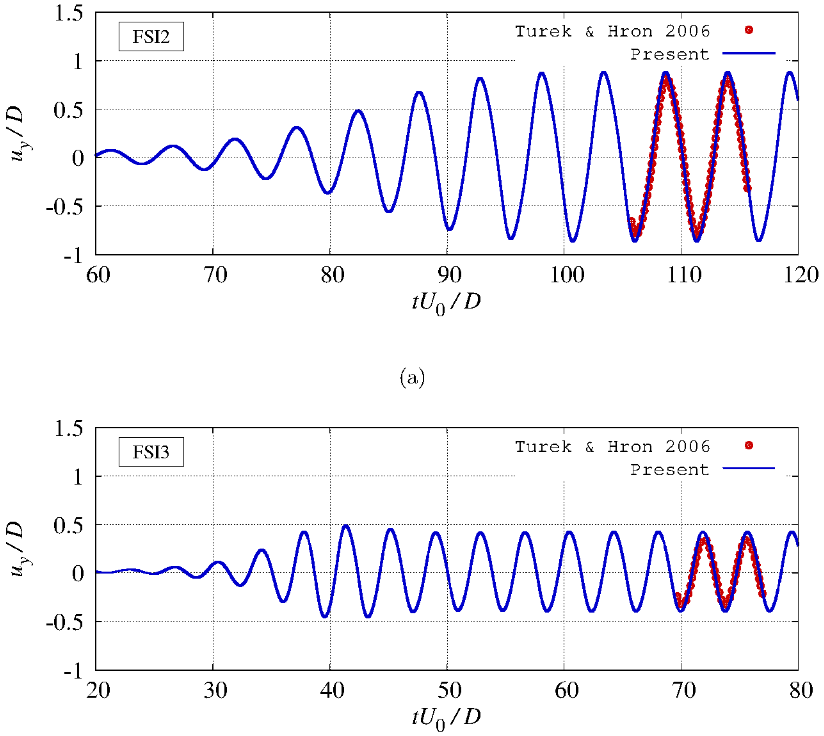


Figure 11: Vorticity fields in the 2D flow-induced vibration test-case.



(a)

(b)

Figure 12: Time evolution of the vertical tip displacement of the beam in the 2D flow-induced vibration test-case.

Table 2: Numerical results of the 2D flow-induced vibration test-case.

	FSI2 ($\rho_s^0/\rho_f^0 = 10$, $Re = 100$)			FSI3 ($\rho_s^0/\rho_f^0 = 1$, $Re = 200$)		
	A/D	St	$\bar{C}_D$	A/D	St	$\bar{C}_D$
Turek & Hron 2006 [75]	0.83	0.19	4.13	0.36	0.26	2.30
Bhardwaj & Mittal 2012 [4]	0.92	0.19	3.56	0.41	0.28	2.20
Tian et al. 2014 [74]	0.78	0.19	4.11	0.32	0.29	2.16
Present	0.87	0.19	4.39	0.41	0.26	2.44

7.2.3. *Non-conforming time-steps*

Now, we shall make use of the Turek-Hron's test-cases to validate the proposed temporal interpolation strategies in Sec. 6 for using non-conforming time-steps in the coupling of LBM and FEM.

After analyzing the acceptable time-steps with the criteria given in Sec. 6.1, we found that it is more suitable to use $\Delta t_s > \Delta t_f$ in FSI2 and $\Delta t_s < \Delta t_f$ in FSI3 test-cases. Additionally, the ratio of time-steps cannot be too large due to the constraints shown in Sec. 6.1. As a consequence, we have carried out the simulations with the ratios of time-steps up to 4.

In the present validation, the physical parameters and meshes remain unchanged. The time-steps and other numerical parameters are provided in Table 3. As one may observe in Table 3, we modify the fluid time-steps Δt_f to have different ratios, while fixing the solid time-step Δt_s. The purpose of doing so is to minimize the changing parameters in order to have a meaningful comparison.

Table 3: Numerical parameters used in the 2D flow-induced vibration test-case with non-conforming time-steps.

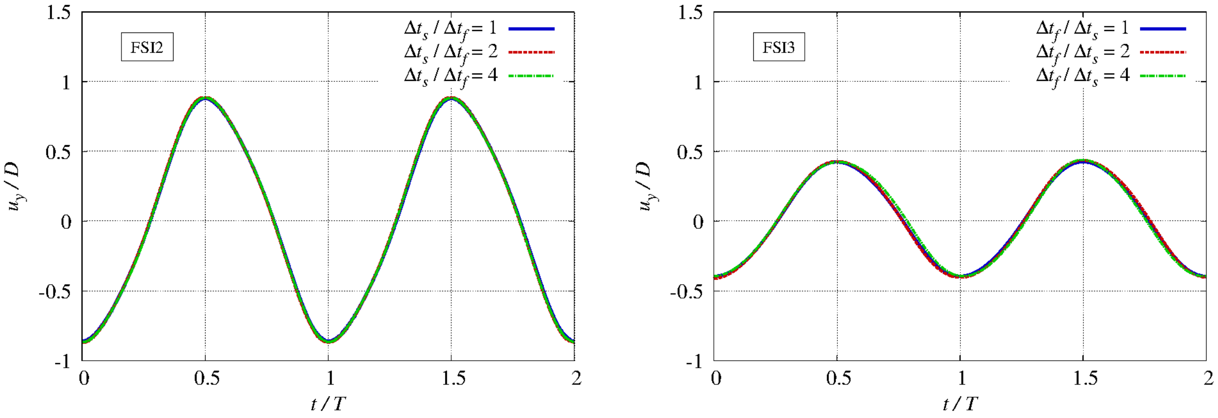
Ratio $r = \Delta t_s/\Delta t_f$	FSI2			FSI3		
	$r = 1$	$r = 2$	$r = 4$	$1/r = 1$	$1/r = 2$	$1/r = 4$
Δt_s (s)	1.0×10^{-4}	1.0×10^{-4}	1.0×10^{-4}	1.6×10^{-5}	1.6×10^{-5}	1.6×10^{-5}
Δt_f (s)	1.0×10^{-4}	5.0×10^{-5}	2.5×10^{-5}	1.6×10^{-5}	3.2×10^{-5}	6.4×10^{-5}
τ^{lat}	0.57500	0.52400	0.51875	0.51200	0.52400	0.54800
U^{lat}/c_f^{lat}	0.08660	0.04330	0.02165	0.02771	0.05542	0.11085

The numerical results using non-conforming time-steps are shown in Fig. 13, in which one may observe a good agreement with the result using the same time-step (ratio = 1), especially for the FSI2 test-case. A slight discrepancy can be found in the FSI3 test-case for the ratio $\Delta t_f/\Delta t_s = 4$, which might be due to the high Mach number in the lattice scale $Ma^{lat} = U^{lat}/c_f^{lat} = 0.11085$, as shown in Table 3. As stated previously in Sec. 6.1.1, when this Mach number is not small enough, the fluid flow simulated by the adopted LBM cannot be assumed to be weakly-compressible or nearly-incompressible.

By means of the present test-case, we have validated the proposed LBM-FEM coupling framework in the presence of deformable solid structures. The proposed temporal interpolation strategies for using non-conforming time-steps have also been assessed and validated, which make the coupling framework more flexible in realistic applications.

7.3. *3D flapping flag*

A 3D flapping flag test-case is considered here to further validate the proposed coupling framework for handling 3D FSI problems. The adopted 3D test-case involves the transient interaction between a thin elastic



(a) (b)

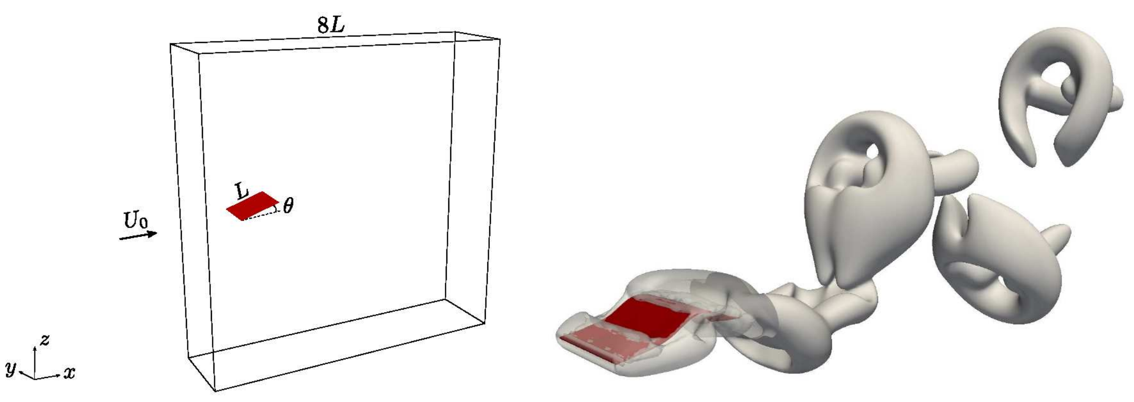
Figure 13: Numerical results of the 2D flow-induced vibration test-cases using non-conforming time-steps: (a) $\Delta t_f < \Delta t_s$; (b) $\Delta t_f > \Delta t_s$.

flag and a uniform incoming fluid flow. The initial configuration is shown in Fig. 14-(a), in which L denotes the size of the square flag in a fluid domain of size $8L \times 2L \times 8L$ in x-, y- and z-directions, respectively. The flag is fixed at its leading edge and initially inclined with an angle $\theta = 0.1\pi$ from the x-y plane. A uniform velocity U_0 is imposed at the inlet, upper and lower boundaries of the fluid domain. Periodic boundary condition is applied at the lateral surfaces in the spanwise direction. The pressure is fixed at the outlet of the fluid domain.

Initially conducted by Huang & Sung [40], this flapping flag test-case has been widely used for validations of various different numerical methods [15, 20, 53, 74]. In the present work, we choose the case with a Reynolds number $Re = U_0 L/\nu_f = 200$. As done in [74] with the second flag model, we set the thickness of the flag as $h = 0.01L$ in the present 3D FE-simulation. In addition, the solid Poisson ratio is $\nu_s = 0.4$ and the Young modulus E_s is computed such that the bending rigidity of the flag is equal to $E_s h^3/(12(1-\nu_s^2)\rho_f^0 U_0^2 L^3) = 0.0001$, where ρ_f^0 is the reference fluid density giving a unit mass ratio $\rho_s^0 h/(\rho_f^0 L) = 1$.

As for the discretization parameters, the fluid domain is discretized by means of a lattice of size $401 \times 101 \times 401$ LB-nodes, which means that $L = 50\Delta x_f$. Choosing $U_0 = 0.04\Delta x_f/\Delta t_f$, the relaxation time in the LB-solver is set as $\tau^{lat} = 0.53$ for $Re = 200$. The solid structure is discretized with a mesh of size $49 \times 49 \times 2$ FE-elements. In the present test-case, the fluid and solid time-steps are the same and set as $\Delta t_f = \Delta t_s = 0.0008L/U_0$ satisfying the criteria shown in Sec. 6.1 for both of the LB- and FE-solvers.

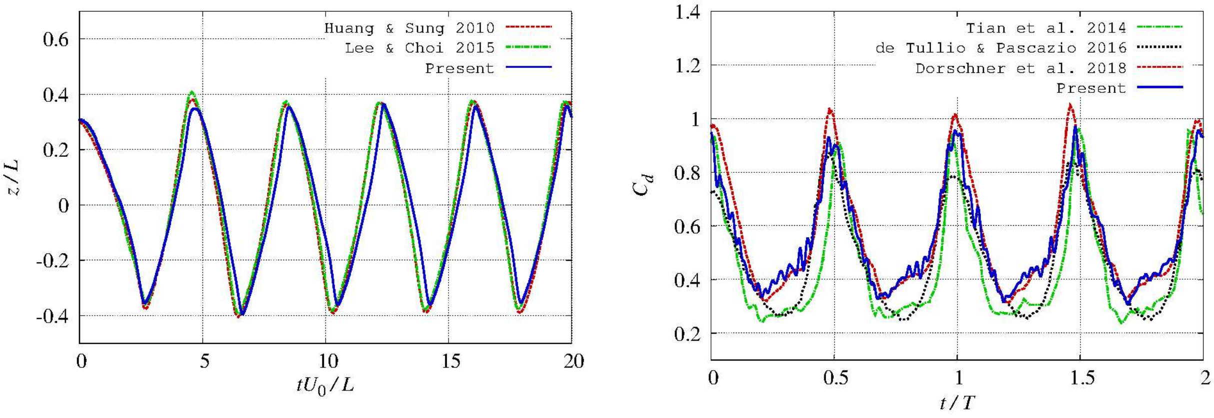
The vortical structures in the fluid domain are shown in Fig. 14-(b), which are obtained with the iso-surface of the Q-criterion. In addition, the time evolution of the position in z-direction of the middle point at the trailing edge is shown and compared with the references in Fig. 15-(a) and the drag coefficient C_d during two periods of flapping cycle is shown in Fig. 15-(b). Moreover, Table 4 shows the results of the flapping amplitude



(a) (b)

Figure 14: 3D flapping flag test-case: (a) initial configuration; (b) vortical structures in the fluid domain.

A/L and the Strouhal number $St = fL/U_0$ with f being the flapping frequency. In these comparisons, one may observe a good agreement between the present results and the references, which validate the proposed coupling framework in this test-case of a flapping flag in the presence of a 3D deformable structure.



(a) (b)

Figure 15: 3D flapping flag test-case: (a) time history of the position in z-direction of the middle point at the trailing edge; (b) evolution in time of the drag coefficient.

7.4. Blood flow through the aortic valve

At last, we apply the present LBM-FEM coupling framework to simulate a 3D blood flow passing through the aortic valve. This biomechanical test-case was initially conducted by Sigüenza et al. [67] in a combined experimental and numerical study, in which one of the purposes is to assess the immersed thick boundary method previously developed by Sigüeneza et al. [68].

Table 4: Numerical results of the 3D flapping flag test-case at $Re = 200$.

	A/L	St
Huang & Sung 2010 [40]	0.780	0.260
Tian et al. 2014 [74]	0.806	0.266
Lee & Choi 2015 [53]	0.752	0.265
de Tullio & Pascazio 2016 [15]	0.795	0.265
Present	0.711	0.269

The configuration is shown in Fig. 16, where we consider to simulate the blood flow inside a rigid duct composed of three parts of diameters $D_1 = 25$ mm and $D_3 = 31$ mm, and of lengths $L_1 = 2D_1$, $L_2 = D_1$ and $L_3 = 2D_3$. The first part starts with the inlet of the duct, which corresponds to the left ventricle side of the heart, the second part represents the aortic root and the third part is the ascending aorta ending with the outlet. The geometry of the three sinuses of Valsalva is computed with the method given by Reul et al. [61]. The aortic valve (red) composed of three deformable leaflets of thickness $e_l = 0.15$ mm is installed on a rigid frame (green) of thickness $e_f = 1.45$ mm representing the inter-leaflet triangle. It has to be noted here that some geometric details are not provided in [67], hence some parts used in the present work, e.g. the rigid frame and the sinus, do not have exactly the same geometries as the ones in [67], although they look like quite similar.

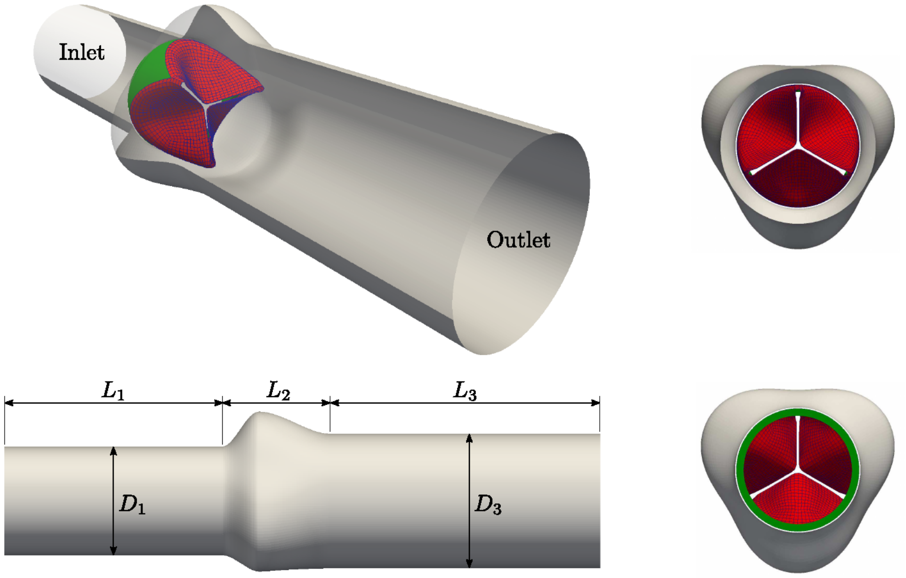
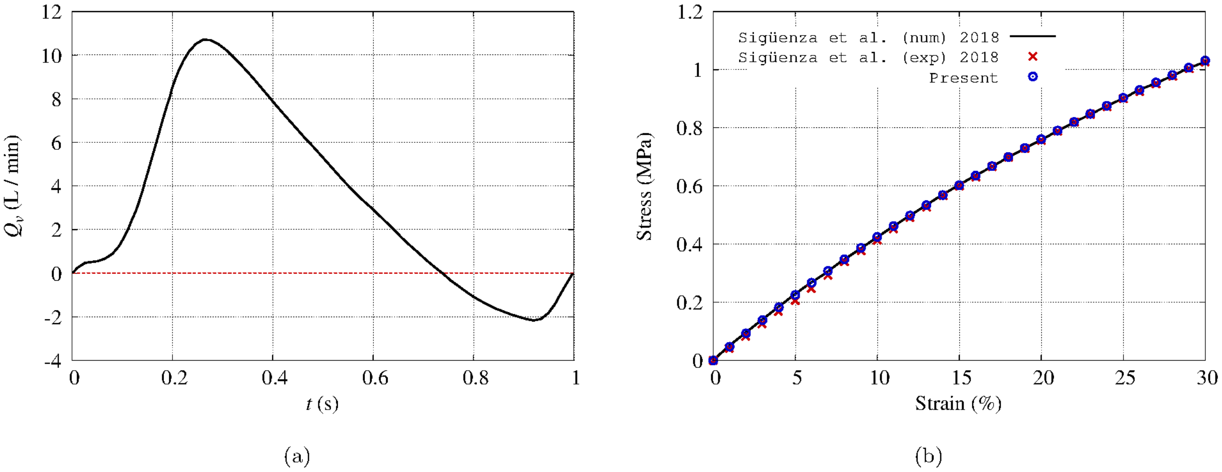


Figure 16: Configuration of the blood flow through the aortic valve 3D application case.

At the inlet of the fluid sub-domain simulated with LBM, a pulsatile and periodic volume flow rate Q_v is imposed and shown in Fig. 17-(a), where one may observe that each cycle of period $T = 1$ s starts with a forward flow ($Q_v > 0$) and ends with a backward flow ($Q_v < 0$). The peak value of the Reynolds number $Re = \rho_f^0 U_d D_1 / \mu_f$ with $U_d = 4Q_v/(\pi D_1^2)$ is about 2800, when the aortic valve is fully opened. In addition, the pressure is fixed at the outlet. The deformable aortic valve is modeled using FEM with the Neo-Hookean hyperelastic material model previously presented in Sec. 3.2 by Eq. (18). The material parameters are given in Table 5. In order to validate the hyperelastic material model implemented in the present FE-solver, we have carried out a numerical uniaxial tensile test and compare the results with the ones presented in [67]. As shown in Fig. 17-(b), a good agreement can be found between the present result and the numerical and experimental results in [67].

Table 5: Material parameters in the blood flow through the aortic valve test-case.

Parameter	Value
Fluid density (ρ_f^0)	1100 kg/m^3
Dynamic viscosity (μ_f)	3.6×10^{-3} Pa.s
Solid density (ρ_s^0)	1000 kg/m^3
Shear modulus (G_s)	2.4 MPa
Bulk modulus (K_s)	1.6 MPa



(a)

(b)

Figure 17: Blood flow through the aortic valve test-case: (a) pulsatile and periodic volume flow rate at the inlet; (b) results in the uniaxial tensile test.

A rectangular cuboid including the rigid duct is used as the computational fluid sub-domain, which is discretized by a uniform lattice of size 89 × 89 × 275 LB-nodes with $\Delta x_f = 0.5$ mm. Notice that only the LB-nodes inside the duct are involved in the computation. No-slip condition is applied on all rigid walls, which is imposed by means of the interpolated bounce-back scheme [7] combined with the mass-conservation correction [82]. In addition, the aortic leaflets are discretized with 1536 quadratic hexahedral FE-elements, leading to the same mesh resolution as in [67].

As for the modeling of turbulent effects, we choose to adopt the LBM-LES model [18, 19, 63] based on the Smagorinsky eddy-viscosity ν_t [69], which is computed as

$$\nu_t = (C_S \Delta x_f)^2 \sqrt{2\bar{S}_{ab}\bar{S}_{ab}} \quad \text{with} \quad \bar{S}_{ab} = \frac{1}{2}\left(\frac{\partial \bar{u}_a}{\partial x_b} + \frac{\partial \bar{u}_b}{\partial x_a}\right), \tag{67}$$

where the coefficient $C_S = 0.17$ in the present work and $\bar{u}_a$ denotes the filtered macroscopic velocity component in the ath direction. At each LB-node, after obtaining the eddy-viscosity ν_t, the total viscosity ν_{tot} will be computed as the harmonic average $\sqrt{\nu_f^2 + \nu_t^2}$, which is then used to obtain the relaxation time τ in the collision step of the LB-solver.

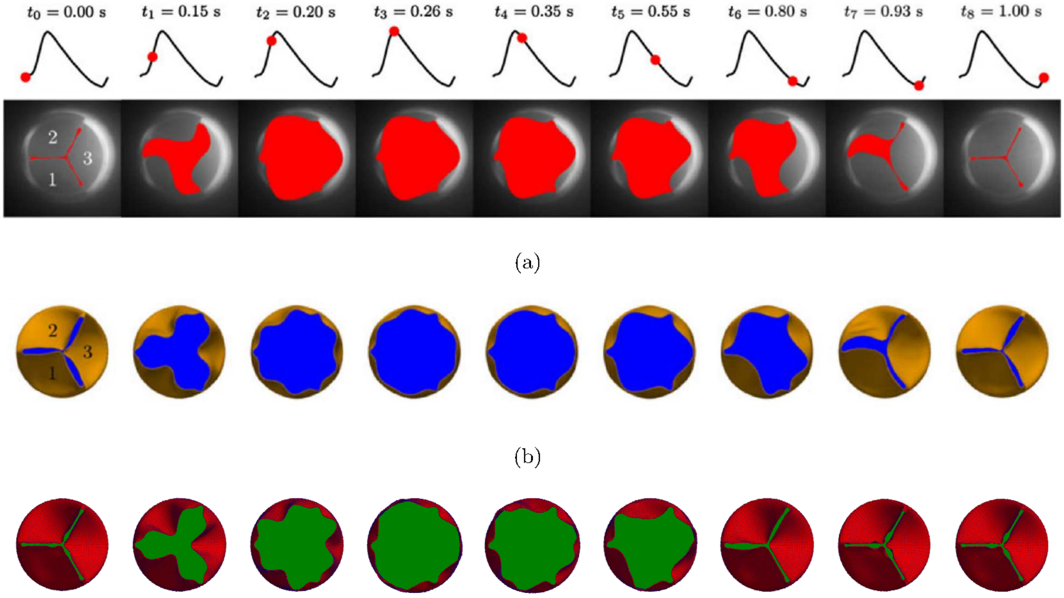
As done in [67], the initial shapes of the valve leaflets are obtained after a non-linear static FE-computation with a constant normal pressure loading on the outer surface of the leaflets. The initial state without stress in the valve leaflets is shown in Fig. 16. Furthermore, it is worth noting that during the closing phase of the aortic valve, significant contact may happen between the leaflets. Various types of contact models can be adopted to take into account the contact effects, such as the geometrical approach in [15] and the penalty force models in [6, 8, 45]. In the present work, we applied a simple strategy based on a repulsive force computed as

$$\mathscr{F}_{\text{contact}} = \begin{cases} 0, & \text{if } d_{vp} - d_0 > 0, \\ -k_c(d_{vp} - d_0)\boldsymbol{n}_{vp}, & \text{if } d_{vp} - d_0 \leq 0, \end{cases} \tag{68}$$

where d_{vp} is the distance between the FE-node and the nearest trileaflet-symmetry virtual plane [12] with $\boldsymbol{n}_{vp}$ being its normal vector pointing towards the FE-node side. In addition, $k_c = 0.1$ N/mm and $d_0 = 0.25$ mm are two constant coefficients which are chosen such that no penetration occurs between the leaflets.

In the present test-case, the solid time-step is 80 times smaller than the fluid time-step, i.e. $\Delta t_f = 80\Delta t_s = 5 \times 10^{-5}$ s, due to the thin leaflet of thickness $e_l = 0.15$ mm and the use of the explicit Newmark time integrator in the solid sub-domain. In this situation, we applied the coupling strategy presented in Sec. 6.3 for using non-conforming time-steps. Here, it is worth noting that the proposed strategies render the coupling framework more flexible and efficient, as they prevent the LB-solver from using the same time-step as the FE-solver, which is unnecessarily too small for the numerical simulation in the fluid sub-domain.

Starting from the initial state presented previously, the FSI simulation has been carried out for five cardiac cycles ($5T = 5$ s) and we observed that the opening-closing motion of the aortic valve became almost periodic after three cycles. The shapes of the aortic valve at nine instants $t = 0.00$ s, 0.15 s, 0.20 s, 0.26 s, 0.35 s, 0.55 s, 0.80 s, 0.93 s and 1.00 s during one cycle of period $T = 1$ s are shown in Fig. 18. Moreover, the time evolution of the opening area is shown and compared with the reference [67] in Fig. 19. By comparing Fig. 18-(b) and Fig. 18-(c) at $t = 0.15$ s and 0.20 s, we observed that the shape of the opening valve in the present result is quite similar to the one of the numerical simulation in [67], except that the opening area is slightly smaller than that of [67]. This discrepancy can be easily seen in the comparison shown in Fig. 19. Notice that when the valve is fully opened, the shapes as well as the opening areas shown in Fig. 18 and Fig. 19 agree well between the present and reference results. This means that the opening of the valve is slightly delayed in the present numerical result. Additionally, we found that the valve is closed a little earlier in the present result. As we used the same mesh resolution and the same material model for the deformable valve leaflets, one of the possible reasons might be the difference in the geometry of rigid walls such as the sinus of Valsalva, as mentioned previously. Nevertheless, despite these discrepancies, a globally good agreement can be found between the present numerical result and the reference, at least for 0.00 s $\leq t \leq$ 0.6 s.



(a)

(b)

(c)

Figure 18: Shapes of the aortic valve at different instants: (a) experimental measurement in [67]; (b) numerical simulation in [67]; (c) present numerical result (the figures in [67] are reprinted here with permission from the publisher).

Finally, the vortical structures in the blood flow are visualized using the iso-surface of the Q-criterion in Fig. 20 at four instants: early-systole (ES) at t = 0.20 s, peak-systole (PS) at t = 0.26 s, mid-systole (MS) at t = 0.35 s and late-systole (LS) at t = 0.55 s. From this comparison, one may observe that the vortical structures are quite similar between the present result and the reference simulation result. However, as shown in the ES and PS instants, the front of the vortical structures moves slightly faster in the present numerical simulation, which is related to the delayed opening of the valve. Since the volume flow rate Q_v is imposed the same as in the reference, when the opening area is smaller, the local mean velocity is greater, which explains

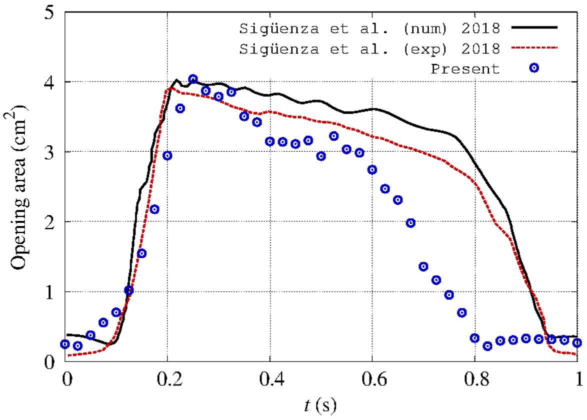


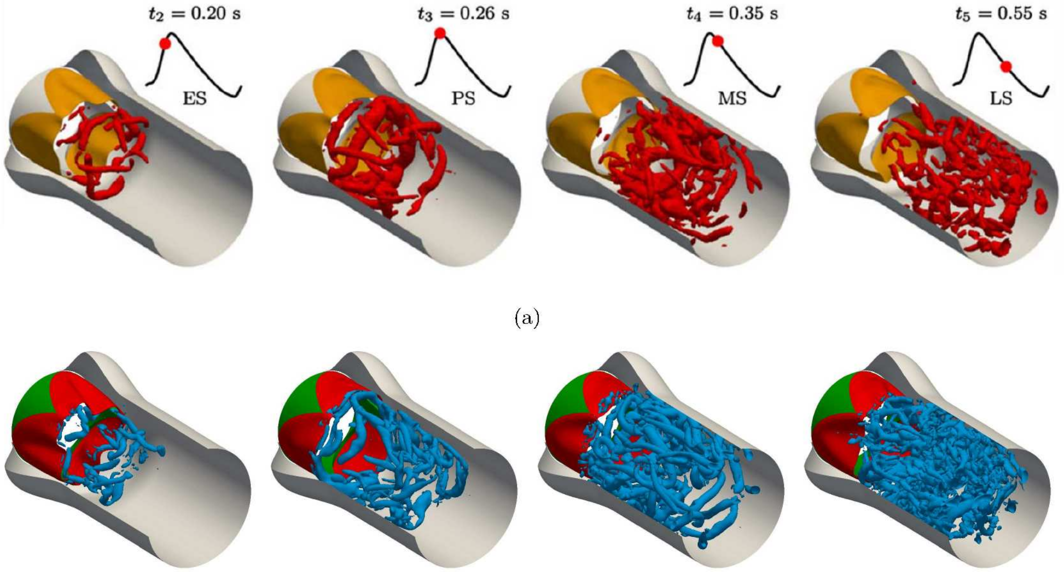
Figure 19: Time evolution of the opening area of the aortic valve during one cycle of period $T = 1$ s.

why the front moves slightly faster in the present result. At last, the pressure fields $p - p_o$ (with p_o being the pressure at the outlet) in a slice plane are shown at $t = 0.1$ s and $t = 0.9$ s in Fig. 21, corresponding to the opening and closing instants.

8. Conclusions

In the present work we proposed a coupling framework for the numerical simulation of fluid-structure interactions, in which the regularized lattice Boltzmann method (LBM) and the non-linear finite element method (FEM) are coupled via an implicit immersed boundary method (IBM) in a partitioned but strong way. Thanks to the use of the implicit IBM, the no-slip condition can be exactly ensured at the fluid-solid interface. In addition, while avoiding sub-iterations during each time-step, the proposed strong coupling algorithm satisfies the velocity and stress continuity condition across the interface. Moreover, in order to use non-conforming time-steps, we proposed two temporal interpolation strategies by assuming that several quantities vary linearly in time, which avoid LB- or FE-computations with unnecessarily small time-steps.

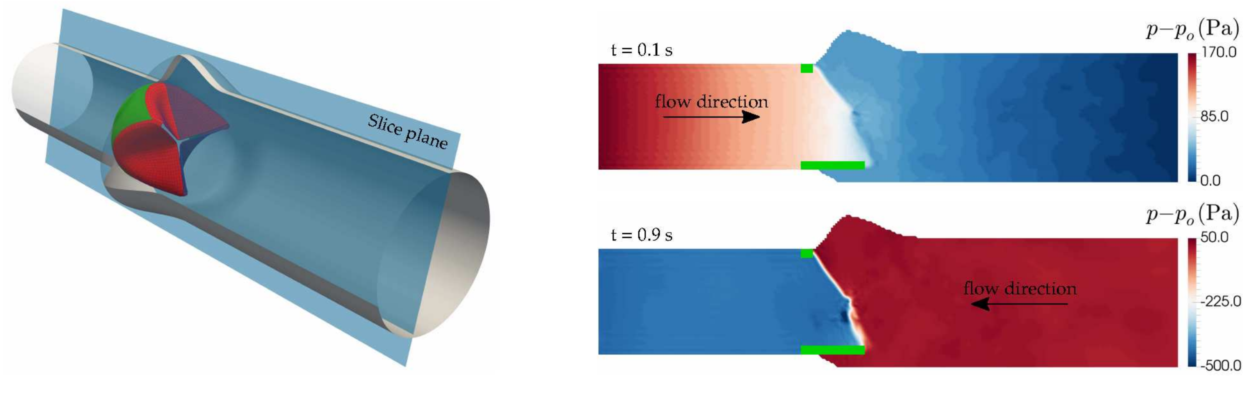
In the first test-case involving the interaction between the blood flow and a rigid mechanical heart valve, the coupling framework is shown to be capable of handling not only deformable solid structures but also rigid thin solid objects. The flow patterns and the valve motion have been correctly reproduced, compared with the reference numerical and experimental results. Then, the proposed coupling algorithm is validated in a 2D fluid-induced vibration test-case, followed by an assessment of the validity and correctness of the temporal interpolation strategies. A series of tests using different time-steps showed that the proposed strategies give good results, which are quite close to the ones using the same time-step. After being validated in a 3D flapping flag test-case, the proposed coupling framework has been applied in a 3D biomechanical case involving the



(a)

(b)

Figure 20: Vortical structures plotted with the iso-surface of the Q-criterion of value 5000 s^{-2} in the blood flow through the aortic valve test-case: (a) numerical result of Sigüenza et al. [67]; (b) present numerical result (the figures in [67] are reprinted here with permission from the publisher).



(a)

(b)

Figure 21: Pressure fields at the opening and closing instants: (a) slice plane; (b) pressure fields at $t = 0.1$ s and $t = 0.9$ s.

pulsatile blood flow through the deformable aortic valve in realistic physiological conditions. A time-step ratio of 80 was used in this test-case, which showed the flexibility and robustness of the proposed LBM-FEM coupling framework. However, it is noteworthy that shell-FEM seems to be more appropriate for modeling the thin heart valves, because the solid time-step would be less restrictive in shell-FEM. Hence, incorporating shell-FEM in the coupling framework will be considered in the future work in order to make it more efficient for simulating blood flows through heart valves.

Acknowledgement

This work was performed by using HPC resources of the Centrale Nantes Supercomputing Centre on the cluster Liger. Mr. Kevin Ley is acknowledged for his work during the internship of three months in 2019, which is helpful for the test-case about the mechanical heart valve in the present work.

References

[1] S. Ansumali and I.V. Karlin. Single relaxation time model for entropic lattice Boltzmann methods. *Physical Review E*, 65:056312, 2002.

[2] Y. Bazilevs, M.-C. Hsu, Y. Zhang, W. Wang, X. Liang, T. Kvamsdal, R. Brekken, and J.G. Isaksen. A fully-coupled fluid-structure interaction simulation of cerebral aneurysms. *Computational Mechanics*, 46:3–16, 2010.

[3] T. Belytschko, W.K. Liu, B. Moran, and K. Elkhodary. *Nonlinear finite elements for continua and structures.* John Wiley & Sons, 2013.

[4] R. Bhardwaj and R. Mittal. Benchmarking a Coupled Immersed-Boundary-Finite-Element Solver for Large-Scale Flow-Induced Deformation. *AIAA Journal*, 50:1638–1642, 2012.

[5] P.L. Bhatnagar, E.P. Gross, and M. Krook. A model for collision processes in gases. I: small amplitude processes in charged and neutral one-component system. *Physical Review*, 94:511, 1954.

[6] I. Borazjani. Fluid–structure interaction, immersed boundary-finite element method simulations of bio-prosthetic heart valves. *Computer Methods in Applied Mechanics and Engineering*, 257:103–116, 2013.

[7] M. Bouzidi, M. Firdaouss, and P. Lallemand. Momentum transfer of a Boltzmann-lattice fluid with boundaries. *Physics of Fluids*, 13:3452–3459, 2001.

[8] Y. Chen and H. Luo. A computational study of the three-dimensional fluid–structure interaction of aortic valve. *Journal of Fluids and Structures*, 80:332–349, 2018.

[9] Ye Chen and H. Luo. A computational study of the three-dimensional fluid–structure interaction of aortic valve. *Journal of Fluids and Structures*, 80:332–349, 2018.

[10] C. Coreixas, B. Chopard, and J. Latt. Comprehensive comparison of collision models in the lattice Boltzmann framework: Theoretical investigations. *Physical Review E*, 100:033305, 2019.

[11] C. Coreixas, G. Wissocq, G. Puigt, J.F. Boussuge, and P. Sagaut. Recursive regularization step for high-order lattice Boltzmann methods. *Physical Review E*, 96:033306, 2017.

[12] J. De Hart, G.W.M. Peters, P.J.G. Schreurs, and F.P.T. Baaijens. A three-dimensional computational analysis of fluid–structure interaction in the aortic valve. *Journal of Biomechanics*, 36:103–112, 2003.

[13] A. De Rosis, G. Falcucci, S. Ubertini, and F. Ubertini. A coupled lattice Boltzmann-finite element approach for two-dimensional fluid–structure interaction. *Computers & Fluids*, 86:558–568, 2013.

[14] A. De Rosis, S. Ubertini, and F. Ubertini. A partitioned approach for two-dimensional fluid–structure interaction problems by a coupled lattice boltzmann-finite element method with immersed boundary. *Journal of Fluids and Structures*, 45:202–215, 2014.

[15] M.D. de Tullio and G. Pascazio. A moving-least-squares immersed boundary method for simulating the fluid-structure interaction of elastic bodies with arbitrary thickness. *Journal of Computational Physics*, 325:201–225, 2016.

[16] D. d'Humières. Generalized Boltzmann equations. *Prog. Aeronaut. Astronaut.*, 159:450–458, 1992.

[17] D. d'Humières, I. Ginzburg, M. Krafczyk, P. Lallemand, and L.-S. Luo. Multiple-relaxation-time lattice Boltzmann models in three dimensions. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 360:437–451, 2002.

[18] Y.-H. Dong and P. Sagaut. A study of time correlations in lattice Boltzmann-based large-eddy simulation of isotropic turbulence. *Physics of Fluids*, 20:035105, 2008.

[19] Y.-H. Dong, P. Sagaut, and S. Marié. Inertial consistent subgrid model for large-eddy simulation based on the lattice Boltzmann method. *Physics of Fluids*, 20:035104, 2008.

[20] B. Dorschner, S.S. Chikatamarla, and I.V. Karlin. Fluid-structure interaction with the entropic lattice Boltzmann method. *Physical Review E*, 97:023305, 2018.

[21] F. Dubois, T. Fevrier, and B. Graille. Lattice Boltzmann schemes with relative velocities. *Communications in Computational Physics*, 17:1088–1112, 2015.

[22] S. Duczek and H. Gravenkamp. Critical assessment of different mass lumping schemes for higher order serendipity finite elements. *Computer Methods in Applied Mechanics and Engineering*, 350:836–897, 2019.

[23] K. Dumont, J.M.A. Stijnen, J. Vierendeels, F.N. van de Vosse, and P.R. Verdonck. Validation of a fluid–structure interaction model of a heart valve using the dynamic mesh method in Fluent. *Computer Methods in Biomechanics and Biomedical Engineering*, 7:139–146, 2004.

[24] J. Favier, A. Revell, and A. Pinelli. A Lattice Boltzmann–Immersed Boundary method to simulate the fluid interaction with moving and slender flexible objects. *Journal of Computational Physics*, 261:145–161, 2014.

[25] C.A. Felippa, K.C. Park, and C. Farhat. Partitioned analysis of coupled mechanical systems. *Computer Methods in Applied Mechanics and Engineering*, 190(24):3247–3270, 2001.

[26] Z.-G. Feng and E.E. Michaelides. The immersed boundary-lattice Boltzmann method for solving fluid–particles interaction problems. *Journal of Computational Physics*, 195:602–628, 2004.

[27] N. Forsythe and J.-D. Mueller. Validation of a fluid–structure interaction model for a bileaflet mechanical heart valve. *International Journal of Computational Fluid Dynamics*, 22:541–553, 2008.

[28] M. Geier, A. Greiner, and J.G. Korvink. Cascaded digital lattice Boltzmann automata for high Reynolds number flow. *Physical Review E*, 73:066705, 2006.

[29] I. Ginzburg, F. Verhaeghe, and D. d'Humières. Study of simple hydrodynamic solutions with the two-relaxation-times lattice Boltzmann scheme. *Communications in Computational Physics*, 3:519–581, 2008.

[30] V. Gkanis and C. Housiadas. A time-dependent numerical analysis of flow in a mechanical heart valve: Comparison with experimental results. *International Journal of Computational Fluid Dynamics*, 24:157–168, 2010.

[31] A. Goza and T. Colonius. A strongly-coupled immersed-boundary formulation for thin elastic structures. *Journal of Computational Physics*, 336:401–411, 2017.

[32] B.E. Griffith, X. Luo, D.M. McQueen, and C.S. Peskin. Simulating the fluid dynamics of natural and prosthetic heart valves using the immersed boundary method. *International Journal of Applied Mechanics*, 01:137–177, 2009.

[33] S. Gsell, U. D'Ortona, and J. Favier. Multigrid dual-time-stepping lattice Boltzmann method. *Physical Review E*, 101:023309, 2020.

[34] Z. Guo, C. Zheng, and B. Shi. Discrete lattice effects on the forcing term in the lattice Boltzmann method. *Physical Review E*, 65:046308, 2002.

[35] Z. Guo, C. Zheng, B. Shi, and T.S. Zhao. Thermal lattice Boltzmann equation for low Mach number flows: Decoupling model. *Physical Review E*, 75:036704, 2007.

[36] X. He and L.-S. Luo. Theory of the lattice Boltzmann method: From the Boltzmann equation to the lattice Boltzmann equation. *Physical Review E*, 56:6811–6817, 1997.

[37] M. Heil and A.L. Hazel. Fluid-Structure Interaction in Internal Physiological Flows. *Annual Review of Fluid Mechanics*, 43:141–162, 2011.

[38] C. Hermange, G. Oger, Y. Le Chenadec, and D. Le Touzé. A 3D SPH–FE coupling for FSI problems and its application to tire hydroplaning simulations on rough ground. *Computer Methods in Applied Mechanics and Engineering*, 355:558–590, 2019.

[39] E. Hinton, T. Rock, and O.C. Zienkiewicz. A note on mass lumping and related processes in the finite element method. *Earthquake Engineering & Structural Dynamics*, 4:245–249, 1976.

[40] W.-X. Huang and H.J. Sung. Three-dimensional simulation of a flapping flag in a uniform flow. *Journal of Fluid Mechanics*, 653:301–336, 2010.

[41] W.-X. Huang and F.-B. Tian. Recent trends and progress in the immersed boundary method. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, 233:7617–7636, 2019.

[42] T.J.R. Hughes. *The finite element method: linear static and dynamic finite element analysis.* Courier Corporation, 2012.

[43] T. Inamuro. Lattice Boltzmann methods for moving boundary flows. *Fluid Dynamics Research*, 44:024001, 2012.

[44] F. Jiang, K. Matsumura, J. Ohgi, and X. Chen. A GPU-accelerated fluid–structure-interaction solver developed by coupling finite element and lattice Boltzmann methods. *Computer Physics Communications*, 259:107661, 2021.

[45] D. Kamensky, M.-C. Hsu, D. Schillinger, J.A. Evans, A. Aggarwal, Y. Bazilevs, M.S. Sacks, and T.J.R. Hughes. An immersogeometric variational framework for fluid–structure interaction: Application to bioprosthetic heart valves. *Computer Methods in Applied Mechanics and Engineering*, 284:1005–1053, 2015.

[46] I.V. Karlin, A.N. Gorban, S. Succi, and V. Boffi. Maximum entropy principle for lattice kinetic equations. *Physical Review Letters*, 81:6–9, 1998.

[47] S. Kollmannsberger, S. Geller, A. Düster, J. Tölke, C. Sorger, M. Krafczyk, and E. Rank. Fixed-grid fluid–structure interaction in two dimensions based on a partitioned Lattice Boltzmann and p-FEM approach. *International Journal for Numerical Methods in Engineering*, 79:817–845, 2009.

[48] M. Krafczyk, M. Cerrolaza, M. Schulz, and E. Rank. Analysis of 3D transient blood flow passing through an artificial aortic valve by Lattice–Boltzmann methods. *Journal of Biomechanics*, 31:453–462, 1998.

[49] S. Krenk. Energy conservation in newmark based time integration algorithms. 195(44):6110–6124.

[50] P. Lallemand and L.-S. Luo. Theory of the lattice Boltzmann method: Dispersion, dissipation, isotropy, Galilean invariance, and stability. *Physical Review E*, 61:6546–6562, 2000.

[51] P. Lallemand, L.-S. Luo, M. Krafczyk, and W.-A. Yong. The Lattice Boltzmann Method for Nearly Incompressible Flows. *Journal of Computational Physics*, page 109713, 2020.

[52] J. Latt and B. Chopard. Lattice Boltzmann method with regularized pre-collision distribution functions. *Mathematics and Computers in Simulation*, 72:165 – 168, 2006.

[53] I. Lee and H. Choi. A discrete-forcing immersed boundary method for the fluid-structure interaction of an elastic slender body. *Journal of Computational Physics*, 280:529–546, 2015.

[54] Z. Li, W. Cao, and D. Le Touzé. On the coupling of a direct-forcing immersed boundary method and the regularized lattice Boltzmann method for fluid-structure interaction. *Computers & Fluids*, 190:470–484, 2019.

[55] Z. Li and J. Favier. A non-staggered coupling of finite element and lattice boltzmann methods via an immersed boundary scheme for fluid-structure interaction. *Computers & Fluids*, 143:90–102, 2017.

[56] Z. Li, J. Favier, U. D'Ortona, and S. Poncet. An immersed boundary-lattice boltzmann method for single- and multi-component fluid flows. *Journal of Computational Physics*, 304:424–440.

[57] N.M. Newmark. A method of computation for structural dynamics. *Journal of the Engineering Mechanics Division*, 85, 1959.

[58] C.S. Peskin. Flow patterns around heart valves: a numerical method. *Journal of Computational Physics*, 10:252–271, 1972.

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

[60] Y.H. Qian, D. D'Humières, and P. Lallemand. Lattice BGK Models for Navier-Stokes Equation. *EPL (Europhysics Letters)*, 17:479, 1992.

[61] H. Reul, A. Vahlbruch, M. Giersiepen, T. Schmitz-Rode, V. Hirtz, and S. Effert. The geometry of the aortic root in health, at valve disease and after valve replacement. *Journal of Biomechanics*, 23:181–191, 1990.

[62] A.M. Roma, C.S. Peskin, and M.J. Berger. An adaptive version of the immersed boundary method. *Journal of Computational Physics*, 153:509–534, 1999.

[63] P. Sagaut. Toward advanced subgrid models for Lattice-Boltzmann-based Large-eddy simulation: Theoretical formulations. *Computers & Mathematics with Applications*, 59:2194–2199, 2010.

[64] S. Seeger and H. Hoffmann. The cumulant method for computational kinetic theory. *Continuum Mechanics and Thermodynamics*, 12:403–421, 2000.

[65] J.-H. Seo, C. Zhu, J. Resar, and R. Mittal. Flow physics of normal and abnormal bioprosthetic aortic valves. *International Journal of Heat and Fluid Flow*, 86:108740, 2020.

[66] X. Shan, X.F. Yuan, and H. Chen. Kinetic theory representation of hydrodynamics: a way beyond the Navier-Stokes equation. *Journal of Fluid Mechanics*, 550:413–441, 2006.

[67] J. Sigüenza, D. Pott, S. Mendez, S.J. Sonntag, T.A.S. Kaufmann, U. Steinseifer, and F. Nicoud. Fluid-structure interaction of a pulsatile flow with an aortic valve model: A combined experimental and numerical study. *International Journal for Numerical Methods in Biomedical Engineering*, 34:e2945, 2018.

[68] J. Sigenza, S. Mendez, D. Ambard, F. Dubois, F. Jourdan, R. Mozul, and F. Nicoud. Validation of an immersed thick boundary method for simulating fluid-structure interactions of deformable membranes. *Journal of Computational Physics*, 322:723–746, 2016.

[69] J. Smagorinsky. General Circulation Experiments with the Primitive Equations: I. The Basic Experiment. *Monthly Weather Review*, 91:99–164, 1963.

[70] J.M.A. Stijnen, J. de Hart, P.H.M. Bovendeerd, and F.N. van de Vosse. Evaluation of a fictitious domain method for predicting dynamic response of mechanical heart valves. *Journal of Fluids and Structures*, 19:835–850, 2004.

[71] S. Succi. *The Lattice Boltzmann Equation: For Complex States of Flowing Matter.* Oxford University Press, 2018.

[72] K. Suzuki and T. Inamuro. Effect of internal mass in the simulation of a moving body by the immersed boundary method. *Computers & Fluids*, 49:173–187, 2011.

[73] K. Taira and T. Colonius. The immersed boundary method: A projection approach. *Journal of Computational Physics*, 225:2118–2137, 2007.

[74] F.B. Tian, H. Dai, H. Luo, J.F. Doyle, and B. Rousseau. Fluid-structure interaction involving large deformations: 3D simulations and applications to biological systems. *Journal of Computational Physics*, 258:451–469, 2014.

[75] S. Turek and J. Hron. *Proposal for Numerical Benchmarking of Fluid-Structure Interaction between an Elastic Object and Laminar Incompressible Flow*, volume 53 of *Lecture Notes in Computational Science and Engineering.* Springer Berlin Heidelberg, 2006.

[76] C. Wang and J.D. Eldredge. Strongly coupled dynamics of fluids and rigid-body systems with the immersed boundary projection method. *Journal of Computational Physics*, 295:87–113, 2015.

[77] L. Wang, F.-B. Tian, and J.C.S. Lai. An immersed boundary method for fluid–structure–acoustics interactions involving large deformations and complex geometries. *Journal of Fluids and Structures*, 95:102993, 2020.

[78] Y. Wang, C. Shu, T.G. Wang, and P. Valdivia y Alvarado. A generalized minimal residual method-based immersed boundary-lattice Boltzmann flux solver coupled with finite element method for non-linear fluid-structure interaction problems. *Physics of Fluids*, 31:103603, 2019.

[79] J. Wu and C. Shu. Implicit velocity correction-based immersed boundary-lattice Boltzmann method and its applications. *Journal of Computational Physics*, 228:1963–1979, 2009.

[80] L. Xu, F.-B. Tian, J. Young, and J.C.S. Lai. A novel geometry-adaptive Cartesian grid based immersed boundary–lattice Boltzmann method for fluid–structure interactions at moderate and high Reynolds numbers. *Journal of Computational Physics*, 375:22–56, 2018.

[81] J. Yang, S. Preidikman, and E. Balaras. A strongly coupled, embedded-boundary method for fluid–structure interactions of elastically mounted rigid bodies. *Journal of Fluids and Structures*, 24:167–182, 2008.

[82] Y. Yu, Q. Li, and Z.X. Wen. Modified curved boundary scheme for two-phase lattice Boltzmann simulations. *Computers & Fluids*, 208:104638, 2020.

[83] C. Zhang, Y. Cheng, L. Zhu, and J. Wu. Accuracy improvement of the immersed boundary–lattice Boltzmann coupling scheme by iterative force correction. *Computers & Fluids*, 124:246–260, 2016.

[84] R. Zhang, X. Shan, and H. Chen. Efficient kinetic method for fluid simulation beyond the Navier-Stokes equation. *Physical Review E*, 74:046703, 2006.

[85] K. Zhou and S. Balachandar. An analysis of the spatio-temporal resolution of the immersed boundary method with direct forcing. *Journal of Computational Physics*, 424:109862, 2021.

[86] K. Zhou, Z. Ding, and K. Sun. Is Lagrangian weight crucial in direct forcing immersed boundary method? *Journal of Physics: Conf. Series*, 1324, 2019.

[87] O.C. Zienkiewicz and R.L. Taylor. *The Finite Element Method: Volume 1, The Basis.* Butterworth Heinemann, 2000.

[88] Q. Zou and X. He. On pressure and velocity boundary conditions for the lattice boltzmann BGK model. *Physics of Fluids*, 9:1591–1598, 1997.

Appendix: Conservation of the angular momentum

As mentioned in Sec. 4.2, although the adopted force formulation (31) conserves exactly the linear momentum, it does not conserve locally the angular momentum. Now, we shall briefly show that the angular momentum can be globally conserved with an approximation and it tends to be zero as the mesh is refined.

With the help of Eq. (29), one can compute the torque $\boldsymbol{T}_{k\to f}$ applied on the whole fluid sub-domain by the kth IB-point as

$$\boldsymbol{T}_{k\to f} = \sum_{j\in D_k} \boldsymbol{T}_{k\to j} = \sum_{j\in D_k} \boldsymbol{x}_j \times \boldsymbol{\mathscr{F}}_{k\to j} = \underbrace{\left(\sum_{j\in D_k} \boldsymbol{x}_j \rho_{f,j}\tilde{\delta}_{jk}\Delta x_f^d\right)}_{\mathcal{I}[\boldsymbol{x}_f\rho_f]_k} \times \boldsymbol{G}_k. \tag{A.1}$$

Meanwhile, the torque $\boldsymbol{T}_{f\to k}$ applied on the kth IB-point from the fluid can be computed with Eq. (31) as

$$\boldsymbol{T}_{f\to k} = \boldsymbol{x}_k \times \boldsymbol{\mathscr{F}}_{f\to k} = -\underbrace{\left(\boldsymbol{x}_k \sum_{j\in D_k} \rho_{f,j}\tilde{\delta}_{jk}\Delta x_f^d\right)}_{\boldsymbol{x}_k\mathcal{I}[\rho_f]_k} \times \boldsymbol{G}_k. \tag{A.2}$$

Summing up Eq. (A.1) and Eq. (A.2) gives the local contribution from the kth IB-point to the variation of the total angular momentum for the whole system as

$$\delta\boldsymbol{T}_{k\leftrightarrow f} = \boldsymbol{T}_{k\to f} + \boldsymbol{T}_{f\to k} = \left(\mathcal{I}\left[\boldsymbol{x}_f\rho_f\right]_k - \boldsymbol{x}_k\mathcal{I}\left[\rho_f\right]_k\right) \times \boldsymbol{G}_k, \tag{A.3}$$

which is approximately equal to zero, as $\mathcal{I}\left[\boldsymbol{x}_f\rho_f\right]_k$ and $\boldsymbol{x}_k\mathcal{I}\left[\rho_f\right]_k$ are not strictly the same, even though they are quite close.

Here, it is important to notice that, if the fluid is strictly incompressible with a constant density $\rho_f = \rho_f^0$, then one has $\delta\boldsymbol{T}_{k\leftrightarrow f} = \rho_f^0(\mathcal{I}[\boldsymbol{x}_f]_k - \boldsymbol{x}_k) \times \boldsymbol{G}_k = \boldsymbol{0}$, because the adopted approximate delta function $\tilde{\delta}$ ensures that the interpolation operator $\mathcal{I}[\bullet]_k$ can give the exact value of a linear function, i.e. $\mathcal{I}[\boldsymbol{x}_f]_k = \boldsymbol{x}_k$.

However, in the present weakly-compressible LB scheme, the fluid density is not a constant, hence $\delta\boldsymbol{T}_{k\leftrightarrow f} \neq 0$ in general cases, which cannot conserve strictly the angular momentum.

To assess the order of approximation about the angular momentum conservation, let us first look at the following proposition regarding the interpolation operator.

Proposition 1. *Given two smooth functions $\chi = \chi(\boldsymbol{x})$ and $\phi = \phi(\boldsymbol{x})$, with the interpolation operator $\mathcal{I}[\bullet]_k$, one can have*

$$\mathcal{I}[\chi\phi]_k - \chi_k\mathcal{I}[\phi]_k = \mathscr{E}_k\Delta x_f^2 + \mathcal{O}(\Delta x_f^3), \tag{A.4}$$

where $\chi_k = \chi(\boldsymbol{x}_k)$ and $\mathscr{E}_k$ is a coefficient independent of Δx_f.

Proof. Given a smooth function $\phi = \phi(\boldsymbol{x})$, its interpolated value at $\boldsymbol{x}_k$ is computed by the interpolation operator $\mathcal{I}[\bullet]_k$ as

$$\mathcal{I}[\phi]_k = \sum_{j \in D_k} \phi_j \tilde{\delta}_{jk} \Delta x_f^d, \tag{A.5}$$

where ϕ_j can be replaced by its Taylor expansion around $\boldsymbol{x}_k$. Thanks to the properties of the approximate delta function [62], Eq. (A.5) can be rewritten as

$$\mathcal{I}[\phi]_k = \phi_k + D_\phi \Delta x_f^2 + \mathcal{O}(\Delta x_f^3), \tag{A.6}$$

where D_ϕ is defined as

$$D_\phi = C_{xx} \left.\frac{\partial^2 \phi}{\partial x^2}\right|_k + C_{xy} \left.\frac{\partial^2 \phi}{\partial x \partial y}\right|_k + C_{yy} \left.\frac{\partial^2 \phi}{\partial y^2}\right|_k, \tag{A.7}$$

in which C_{xx}, C_{xy} and C_{yy} are three coefficients depending only on $\boldsymbol{x}_k$.

Now, let us apply the interpolation operator to the product of two smooth functions χ and ϕ, which gives

$$\mathcal{I}[\chi\phi]_k = \chi_k \phi_k + D_{\chi\phi} \Delta x_f^2 + \mathcal{O}(\Delta x_f^3), \tag{A.8}$$

with

$$D_{\chi\phi} = C_{xx} \left.\frac{\partial^2 (\chi\phi)}{\partial x^2}\right|_k + C_{xy} \left.\frac{\partial^2 (\chi\phi)}{\partial x \partial y}\right|_k + C_{yy} \left.\frac{\partial^2 (\chi\phi)}{\partial y^2}\right|_k. \tag{A.9}$$

Finally, one can have

$$\begin{aligned} \mathcal{I}[\chi\phi]_k - \chi_k \mathcal{I}[\phi]_k &= \chi_k \phi_k + D_{\chi\phi} \Delta x_f^2 - \chi_k (\phi_k + D_\phi \Delta x_f^2) + \mathcal{O}(\Delta x_f^3), \\ &= \underbrace{(D_{\chi\phi} - \chi_k D_\phi)}_{\mathscr{E}_k} \Delta x_f^2 + \mathcal{O}(\Delta x_f^3), \end{aligned} \tag{A.10}$$

where $\mathscr{E}_k$ is independent of Δx_f. ☐

Now, if we use Proposition 1 to Eq. (A.3) with $\chi = \boldsymbol{x}_f$ and $\phi = \rho_f$, we can observe that $\delta \boldsymbol{T}_{k \leftrightarrow f} = \mathcal{O}(\Delta x_f^2)$, i.e. the local contribution from the kth-IB point to the variation of the total angular momentum of the whole system including both fluid and solid is approximately equal to zero and of order $\mathcal{O}(\Delta x_f^2)$.

As a consequence, by summing up $\delta \boldsymbol{T}_{k \leftrightarrow f}$ for all the IB-points ($k \in [1, N_i]$), one has the variation of the total angular momentum

$$\delta \boldsymbol{T}_{s \leftrightarrow f} = \sum_k^{N_i} \delta \boldsymbol{T}_{k \leftrightarrow f} = \mathcal{O}(\Delta x_f), \tag{A.11}$$

which is generally small and tends to be zero as $\Delta x_f \to 0$.