Received: 4 April 2023 | Revised: 29 September 2023 | Accepted: 5 November 2023

DOI: 10.1002/nme.7402

RESEARCH ARTICLE

WILEY

A hybrid Lagrangian–Eulerian particle finite element method for free-surface and fluid–structure interaction problems

Cheng Fu | **Massimiliano Cremonesi** | **Umberto Perego**

Department of Civil and Environmental Engineering, Politecnico di Milano, Milano, Italy

Correspondence
Massimiliano Cremonesi, Department of Civil and Environmental Engineering, Politecnico di Milano, piazza Leonardo da Vinci 32, 20133 Milano, Italy.
Email: massimiliano.cremonesi@polimi.it

Funding information
Tetra Pak Packaging Solutions; Ministero dell'Istruzione, dell'Università e della Ricerca, Grant/Award Number: 20173C478N

Abstract
The dynamics of fluid flows with free surfaces and interacting with highly deformable structures is a complex problem, attracting considerable attention. The Particle Finite Element Method (PFEM) is one of the various numerical methods recently proposed in the literature to simulate this type of problems. It is a mesh-based Lagrangian approach, particularly suited for problems with fast changes in the domain topology, since the fluid boundaries and the Fluid–Structure Interaction (FSI) interface are naturally tracked by the position of the mesh nodes. However, when nonhomogeneous boundary conditions are imposed on velocities or when there are regions where the topology varies moderately, for example, in confined portions of the fluid domain characterized by fixed boundaries, an Eulerian formulation turns out to be more convenient. To exploit the advantages of both formulations, an adaptive hybrid Lagrangian–Eulerian approach is presented in this work. According to the proposed method, nodes on the fluid free-surface and on the FSI interface are treated as Lagrangian, while the remaining nodes can be either Eulerian or Lagrangian. Furthermore, to increase the efficiency of the method, an algorithm to automatically detect runtime the transition zone between the two kinematic descriptions is devised. To validate the proposed approach, several numerical examples are developed and their results are compared to those available in the literature.

KEYWORDS
adaptive hybrid Lagrangian–Eulerian, explicit dynamics, fluid structure interaction, particle finite element method

1 | INTRODUCTION

The study of fluid flows and Fluid–Structure Interaction (FSI) problems involving fast domain evolution and large structural deformation is of great interest in many research areas, such as civil, aerospace, and biomedical engineering. The motion of viscous fluids is governed by the Navier–Stokes equations, a system of partial differential equations that can be formulated either in the Eulerian or in the Lagrangian framework.[1] In the former approach, the fluid problem is solved in a fixed control volume in which the material flows. Using a mesh-based approach, the mesh does not change in time and cannot undergo any distortion. This approach facilitates the treatment of fixed boundaries with prescribed fields (e.g., inflow and outflow, slip, symmetry conditions, …) and it is widely adopted in fluid dynamics for solving problems in confined domains.[1] However, the Eulerian description does not allow for a convenient free surface tracking unless ad hoc techniques are provided, such as Level Set[2] or Volume of Fluid[3] methods. Moreover, due to the relative motion between the material and the mesh, convective terms appear in the definition of the time derivative. These terms should be discretized and properly stabilized. On the contrary, in the Lagrangian approach the computational mesh moves following the fluid particles, with mesh nodes positions that are updated according to the fluid movement. In large-deformation problems, this can lead to excessive mesh distortion and accuracy loss or even compromise the computation. On the positive side, in Lagrangian approaches the nonlinearity related to the convective term disappears and the main difficulty remains the efficient treatment of mesh distortion.[4] While the Lagrangian approach is preferred in Structural Mechanics, it is also very appealing for treating fluid and FSI problems involving free surface flows and interactions with highly deformable structures.

Recently, thanks to improvements in mesh generation algorithms, Lagrangian mesh-based methods have been increasingly employed for fluid flow problems. Among these, the Particle Finite Element Method (PFEM) is a Lagrangian finite element approach based on triangular or tetrahedral elements, in which a new mesh is generated whenever the current one becomes too distorted. The PFEM can be effectively employed in a wide class of engineering problems. It has been initially introduced for solving free surface flows[5,6] but soon its application has been extended also to FSI problems,[7-12] granular,[13,14] and multifluid flows,[15,16] forming processes,[17,18] and thermal coupled analyses.[19,20]

To overcome the typical difficulties of Eulerian and Lagrangian approaches and to exploit their positive features, Arbitrary Lagrangian Eulerian (ALE) methods have been proposed.[1,21] In this unified formulation, the motion of the mesh nodes is decoupled from the motion of material particles. Therefore the computational mesh can move arbitrarily with respect to fluid particles and a convective term appears, depending on the relative velocity between mesh nodes and material particles. This formulation usually prescribes a fully Lagrangian description in correspondence of free surfaces whereas in the rest of the domain the mesh velocity is defined in such a way that mesh distortion is reduced.[1] It is important to recall that in the ALE approach an additional equation should be solved to define the mesh velocity. The ALE approach can be seen as a general kinematic description according to which fully Lagrangian and Eulerian models are obtained as special cases by setting the convective velocity equal to zero for the former and equal to the fluid velocity for the latter. However, the ALE method has important limitations for very large and unpredictable domain deformations which can lead also the ALE mesh to be too distorted.

In addition to ALE approaches, many studies have already been performed to combine Lagrangian and Eulerian approaches to fully exploit their benefits. The Material Point Method, originally introduced in Reference 22, combines the Lagrangian representation of the domain as a set of particles (called Material Points), with a background Eulerian computational mesh. It has the advantages of being able to handle large deformations and to accurately simulate the material behavior, as well as to easily impose complex boundary conditions. However, it may be costly from both the computational and implementation point of view, due to the information transfer at each time step, from material points to cell nodes and vice versa.[23] Another recently developed method is the PFEM-Second Generation known as PFEM-2.[24] It represents an alternative to PFEM for solving incompressible Navier–Stokes problems with larger time steps. It couples standard PFEM with X-IVAS (eXplicit integration following the Velocity and Acceleration Streamlines), the latter consisting of integrating the convective terms along the streamlines. Two versions of PFEM-2 have been introduced: one with a moving mesh and another one with a fixed mesh. The first version rebuilds the mesh when it becomes distorted, while the second one maps variables onto the existing mesh, without generating a new one.[4]

In this work, a hybrid Lagrangian–Eulerian kinematic formulation for the PFEM is presented. Unlike in the ALE approach, the proposed technique divides the domains into separate Eulerian and Lagrangian parts to fully exploit the advantages of the two approaches. The Lagrangian approach is used in regions with fluid free surfaces and fluid–structure interfaces, whereas the Eulerian approach with a fixed mesh is used in the remaining regions, with a reduction of the time spent in the mesh management with respect to the standard Lagrangian methods. In this way, unlike in the standard Eulerian approach, no additional equations are needed to describe the position of the free-surfaces and compared to the ALE approach, there is no need for an additional equation to describe the mesh motion. The price to pay for these advantages, is the management of the transition between Lagrangian and Eulerian regions. Moreover, in a general case, the two regions can evolve in time. Parts of the domain which are at the beginning Lagrangian can become Eulerian and vice versa.

The idea of the mixed Lagrangian–Eulerian approach is not new. Ryzhakov et al.[25] proposed a unified ALE model for treating FSI problems involving flows in flexible channels. Cremonesi et al.[13,26] introduced the mixed

Lagrangian–Eulerian formulation to deal with non-homogeneous boundary conditions, limiting the Eulerian description to boundaries. Masó et al.[27] proposed a strategy to combine a Lagrangian Navier–Stokes multifluid solver with an Eulerian method based on the Boussinesq shallow water equations for simulating landslide-generated wave events.

Starting from the ideas proposed in Reference 26, in this work the Eulerian approach is extended also to the inner regions of the fluid domain, adopting the Lagrangian description only where it is advantageous, such as in correspondence of free surfaces and FSI interfaces. The idea is first presented for the fluid problem alone and then extended also to FSI problems. In this latter case, a rectangular remeshing region enclosing the structure is introduced. Inside this region, to facilitate the definition of the FSI interface, nodes are treated as Lagrangian, while the remaining nodes are treated as Eulerian. During the analysis, the position and space extension of the remeshing region are updated to follow the solid body.

The paper is organized as follows. In Section 2 the PFEM is briefly described, space and time discretization are defined and the technique to deal with FSI problems is presented. The hybrid Lagrangian–Eulerian scheme proposed in this work is discussed in Section 3 for both fluid only and FSI problems. The proposed approach is validated in Sections 4 through two-dimensional (2D) and three-dimensional (3D) numerical examples. Finally, conclusions are highlighted in Section 5.

2 | PARTICLE FINITE ELEMENT METHOD

The PFEM is a Lagrangian mesh-based method for solving fluid-dynamics problems, particularly effective for modeling free surface flows and breaking waves.[4-6,28] Due to its natural ability to track evolving interfaces, the method gained more and more attention also in FSI problems involving large structural displacements.[7-12,29]

The key feature of PFEM is the runtime mesh regeneration. Following the Lagrangian approach, mesh nodes move with the fluid and, consequently, the computational mesh becomes soon overly distorted leading to a loss of convergence and accuracy. Moreover, if an explicit time integration scheme is adopted, the stable time step size can be drastically reduced in the presence of highly distorted elements.[30] To overcome these issues, when the current mesh becomes overly distorted, a new one is created through the Delaunay triangulation[31] starting from the current node positions (Figure 1A). A global index of the mesh distortion, used to decide whether a new mesh should be built, is defined using the radius of the circle circumscribed to each element and the mean mesh size. Details can be found in Reference 9. To avoid an expensive and inaccurate mapping from the old mesh to the new one, all fluid variables and physical properties are stored at the nodes.

The 2D Delaunay tessellation generates optimal triangular meshes, in the sense that important geometrical properties, such as the minimization of the maximum radius of an element circumcircle and the maximization of the minimum angle among all elements, are guaranteed. However, the tessellation generates the convex hull of a given cloud of nodes (see Figure 1B), which does not guarantee the respect of internal and external boundaries. A way to overcome this issue is to apply the alpha-shape method,[4,6,32] consisting in the removal of nonphysical elements according to a geometric criterion.

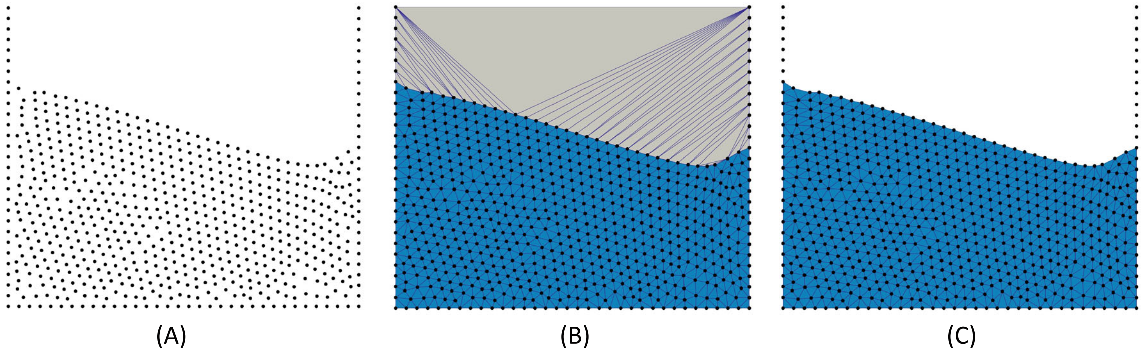


FIGURE 1 Mesh generation in particle finite element method (PFEM). starting from a set of fluid particles (A), a convex hull is obtained through the Delaunay triangulation (B). The real fluid domain is recovered by applying the alpha-shape method that removes the distorted elements (C). (A) Set of particles; (B) Delaunay triangulation; (C) Alpha-shape.

An element is considered too distorted, hence to be removed, if its characteristic length l_e is larger than a predefined threshold value:

$$l_e \geq \overline{\alpha}\, l_{\text{mean}}, \tag{1}$$

where l_{mean} is a mean characteristic size of the elements in the computational mesh and $\overline{\alpha}$ a geometrical distortion index. Elements that fulfill the relation (1) are removed from the computational mesh (Figure 1C). The parameter $\overline{\alpha}$ plays a fundamental role in defining the actual fluid domain. For a small value of $\overline{\alpha}$, few elements are preserved, whereas the original domain obtained by the Delaunay tessellation is recovered for a large value of $\overline{\alpha}$. As a consequence, the mesh quality and the parameter $\overline{\alpha}$ impact the overall mass conservation.[33] Although this issue is intrinsic to the remeshing process in the PFEM, it can be controlled with mesh refinement and by setting a proper $\overline{\alpha}$ value. As suggested in Reference 34, different parameters $\overline{\alpha}$ can be introduced in specific portions of the fluid domain, for example, for internal regions ($\overline{\alpha}_{\text{int}}$) and for free surface elements ($\overline{\alpha}_{\text{surf}}$). The former is adopted for the distortion control in the fluid bulk and it is generally assigned a higher value to avoid nonphysical voids inside the domain. The latter is used for removing distorted elements on the free surface and a smaller value is generally prescribed, so that a more accurate profile is obtained. In the applications discussed in this paper, the characteristic length of a considered element is taken as its circumcircle radius, while the two $\overline{\alpha}$-parameters are set to $\overline{\alpha}_{\text{int}} = 2$ and $\overline{\alpha}_{\text{surf}} = 1.2$. In all the examples, the fluid domain is discretized with almost homogeneous meshes with elements of approximately the same size. However, the technique presented in the what follows can also be used with nonhomogeneous meshes, without any specific modification.

2.1 | Fluid governing equations

Let $\boldsymbol{v}_f = \boldsymbol{v}_f(\boldsymbol{x}, t)$ be the velocity field, $\rho_f = \rho_f(\boldsymbol{x}, t)$ the fluid density, $p_f = p_f(\boldsymbol{x}, t)$ the fluid pressure and $\boldsymbol{\sigma}_f = \boldsymbol{\sigma}_f(\boldsymbol{x}, t)$ the Cauchy stress tensor. Denoting $\rho_f \boldsymbol{b}_f$ the fluid external body force and K_f the fluid bulk modulus, the Navier–Stokes equations for an evolving fluid domain Ω_f^t in the time interval $[0, T]$ can be conveniently written in the ALE framework:

$$\rho_f \left[\frac{\partial \boldsymbol{v}_f}{\partial t} + (\boldsymbol{c}_f \cdot \nabla_x) \boldsymbol{v}_f \right] = \nabla_x \cdot \boldsymbol{\sigma}_f + \rho_f \boldsymbol{b}_f \quad \text{in } \Omega_f^t \times [0, T], \tag{2}$$

$$\left[\frac{\partial p_f}{\partial t} + (\boldsymbol{c}_f \cdot \nabla_x) p_f \right] + K_f (\nabla_x \cdot \boldsymbol{v}_f) = 0 \quad \text{in } \Omega_f^t \times [0, T], \tag{3}$$

where ∇_x is the gradient operator and $\boldsymbol{c}_f = \boldsymbol{c}_f(\boldsymbol{x}, t)$ is the convective velocity defined as:

$$\boldsymbol{c}_f(\boldsymbol{x}, t) = \boldsymbol{v}_f(\boldsymbol{x}, t) - \boldsymbol{r}_f(\boldsymbol{x}, t), \tag{4}$$

being $\boldsymbol{r}_f(\boldsymbol{x}, t)$ the mesh velocity. Typically, the PFEM is formulated in the Lagrangian framework, however, for the subsequent developments the equations of motion (2) and mass conservation (3) have been written in the ALE framework. The Lagrangian and Eulerian formulations can be recovered as special cases by setting the convective velocity to $\boldsymbol{c}_f = \boldsymbol{0}$ or to $\boldsymbol{c}_f = \boldsymbol{v}_f$, respectively.

Even though fluids are generally considered incompressible, they actually exhibit a small compressibility. In the present method, the fluid is assumed to be a weakly compressible. The main advantage of this hypothesis is that the mass conservation equation (3) is hyperbolic and consequently the Poisson pressure equation should not be solved. Moreover, explicit methods can be conveniently used for time integration, allowing easy code parallelization and avoiding convergence problems. Even though the obtained time integration scheme is only conditionally stable, it is particularly appealing for treating fast dynamics and highly nonlinear problems.

To express the constitutive law, the Cauchy stress tensor $\boldsymbol{\sigma}_f$ is decomposed into its isotropic and deviatoric parts:

$$\boldsymbol{\sigma}_f = -p_f \boldsymbol{I} + \boldsymbol{\tau}_f, \tag{5}$$

with $\boldsymbol{I}$ denoting the identity tensor and $\boldsymbol{\tau}_f$ the deviatoric stress tensor. The latter is related to the deviatoric strain rate tensor $\dot{\boldsymbol{e}}_f$ through the fluid constitutive law:

$$\boldsymbol{\tau}_f = 2\mu_f(\dot{\boldsymbol{e}}_f)\dot{\boldsymbol{e}}_f, \tag{6}$$

where $\mu_f(\dot{\boldsymbol{\epsilon}}_f)$ represents the fluid viscosity and $\dot{\boldsymbol{\epsilon}}_f$ is defined as:

$$\dot{\boldsymbol{\epsilon}}_f = \frac{1}{2}(\nabla_x \boldsymbol{v}_f + \nabla_x \boldsymbol{v}_f^T) - \frac{1}{3}(\nabla_x \cdot \boldsymbol{v}_f)\boldsymbol{I}. \tag{7}$$

In the specific case of Newtonian fluid, the viscosity is constant $\mu_f(\dot{\boldsymbol{\epsilon}}_f) = \mu_f$.

The set of Equations (2) and (3) must be completed with initial and boundary conditions. Denoting $\Gamma_f^t = \partial\Omega_f^t$ the fluid boundary at time t, which can be divided into nonoverlapping Dirichlet $\Gamma_{f,D}^t$ and Neumann $\Gamma_{f,N}^t$ boundaries, the following conditions are imposed:

$$\boldsymbol{v}_f(\boldsymbol{x}, t) = \bar{\boldsymbol{v}}_f(\boldsymbol{x}, t) \quad \text{on } \Gamma_{f,D}^t, \tag{8}$$

$$\boldsymbol{\sigma}_f(\boldsymbol{x}, t) \cdot \boldsymbol{n_f} = \boldsymbol{g}(\boldsymbol{x}, t) \quad \text{on } \Gamma_{f,N}^t, \tag{9}$$

where $\bar{\boldsymbol{v}}_f(\boldsymbol{x}, t)$ is an assigned velocity function, $\boldsymbol{g}(\boldsymbol{x}, t)$ represents the applied surface tractions and $\boldsymbol{n_f}$ stands for the outward normal to the boundary $\Gamma_{f,N}^t$. Furthermore, the initial conditions may be expressed as:

$$\boldsymbol{v}_f(\boldsymbol{x}, t = 0) = \boldsymbol{v}_0(\boldsymbol{x}) \quad \text{in } \Omega_f^0, \tag{10}$$

$$p(\boldsymbol{x}, t = 0) = p_0(\boldsymbol{x}) \quad \text{in } \Omega_f^0, \tag{11}$$

where $\boldsymbol{v}_0$ and p_0 are initial velocity and pressure fields.

When the fluid is modeled as an incompressible medium, the density assumes a constant value in time. In the case of the weakly compressible fluid, an equation of state relating pressure to density is required. Under the assumption of isothermal and barotropic flows, the Tait equation of state[35-37] is adopted:

$$p(\rho_f) = p_0 + \frac{K_f}{\gamma}\left[\left(\frac{\rho_f}{\rho_{f,0}}\right)^{\gamma} - 1\right], \tag{12}$$

where the subscript 0 indicates the reference value and γ is the specific heat ratio (set to $\gamma = 7$ in all the examples considered in this paper).

2.2 | Space discretization

A standard Galerkin finite element approach is adopted for the space discretization of Equations (2) and (3). Let S^v and S^p be the spaces of admissible functions for velocities and pressure, respectively, compatible with the assigned boundary conditions. The weak form of the momentum conservation can be obtained by multiplying Equation (2) by a vector of test function $\boldsymbol{w}_f \in S^v$. Integrating then the equation and exploiting the Green formula, the following expression is obtained:

$$\begin{aligned} \int_{\Omega_f^t} \rho_f \boldsymbol{w}_f \cdot \left[\frac{\partial \boldsymbol{v_f}}{\partial t} + (\boldsymbol{c}_f \cdot \nabla_x)\boldsymbol{v_f}\right] d\Omega_f = \int_{\Omega_f^t} p_f (\nabla_x \cdot \boldsymbol{w}_f) d\Omega_f + \\ - \int_{\Omega_f^t} \nabla_x \boldsymbol{w}_f : \boldsymbol{\tau_f} d\Omega_f + \int_{\Omega_f^t} \boldsymbol{w}_f \cdot \rho_f \boldsymbol{b} d\Omega_f + \int_{\Gamma_f^t} \boldsymbol{w}_f \cdot \boldsymbol{g} d\Gamma_f \quad \forall \boldsymbol{w}_f \in S^v. \end{aligned} \tag{13}$$

Let q_f be a scalar test function, with $q_f \in S^p$, the weak form of the mass conservation (3) reads:

$$\int_{\Omega_f^t} q_f \left[\frac{\partial p_f}{\partial t} + (\boldsymbol{c}_f \cdot \nabla_x) p_f\right] d\Omega_f + \int_{\Omega_f^t} K_f q_f (\nabla_x \cdot \boldsymbol{v}_f) d\Omega_f = 0 \quad \forall q_f \in S^p. \tag{14}$$

Following the standard Galerkin approach, test and trial velocity and pressure fields are discretized by means of linear shape functions[*]. Velocities and pressures can be written as a function of their nodal values as:

$$\boldsymbol{v}_f(\boldsymbol{x},t)=\boldsymbol{N}_f^v(\boldsymbol{x})\boldsymbol{V}_f(t), \tag{15}$$

$$\boldsymbol{c}_f(\boldsymbol{x},t)=\boldsymbol{N}_f^v(\boldsymbol{x})\boldsymbol{C}_f(t), \tag{16}$$

$$\boldsymbol{p}_f(\boldsymbol{x},t)=\boldsymbol{N}_f^p(\boldsymbol{x})\boldsymbol{P}_f(t), \tag{17}$$

where $\boldsymbol{N}^v(\boldsymbol{x})$ and $\boldsymbol{N}^p(\boldsymbol{x})$ contain linear shape functions for the velocity and pressure field, whereas $\boldsymbol{V}_f(t)$, $\boldsymbol{C}_f(t)$ and $\boldsymbol{P}_f(t)$ are vectors of nodal fluid velocity, convective velocity and pressure, respectively. It the present work, the convective velocity $\boldsymbol{C}_f(t)$ is set either to zero (Lagrangian) or to the fluid velocity (Eulerian). The semi-discretized momentum and mass conservation equations read:

$$\boldsymbol{M}_{v,f}\frac{d\boldsymbol{V}_f}{dt}=-(\boldsymbol{K}_\mu+\boldsymbol{K}_c^v)\boldsymbol{V}_f+\boldsymbol{D}^T\boldsymbol{P}_f+\boldsymbol{F}_{\text{ext},f}=\boldsymbol{F}_f, \tag{18}$$

$$\boldsymbol{M}_{p,f}\frac{d\boldsymbol{P}_f}{dt}=-\boldsymbol{K}_c^p\boldsymbol{P}_f-K_f\boldsymbol{D}\boldsymbol{V}_f, \tag{19}$$

where $\boldsymbol{M}_{v,f}$ and $\boldsymbol{M}_{p,f}$ are fluid mass and volume-like matrices, respectively, $\boldsymbol{K}_\mu$ is the viscosity matrix and $\boldsymbol{D}$ is the discretization of the divergence operator. The vector $\boldsymbol{F}_{ext,f}$ contains external loads, whereas $\boldsymbol{F}_f$ includes the total forces, given by the sum of the internal and external ones. The terms $\boldsymbol{K}_c^v$ and $\boldsymbol{K}_c^p$ are derived from the convective term in the momentum and mass conservation, respectively (see Reference 26 for a more detailed description). It is important to recall that in the Lagrangian part of the domain, the geometry is updated and the mesh is re-triangulated whenever it becomes too distorted. Consequently, the matrices in Equations (18) and (19) change in time. In contrast, in the Eulerian part of the domain the mesh is fixed in time, so that the elemental matrices and operators do not change and can be defined only once, at the beginning of the analysis.

Since linear shape functions are used for both velocity and pressure fields in the space discretization, a stabilization technique is required to prevent pressure spurious oscillations.[38] In the present work, the Direct Pressure Stabilization has been used.[39,40] Details can be found in the Appendix A. Moreover, in convection-dominated problems, the Eulerian formulation can suffer instabilities. To overcome this issue, standard SUPG stabilization terms have been added to the formulation.[41]

2.3 | Time discretization

The time integration is here performed with an explicit approach, which is very appealing for highly nonlinear and fast dynamic problems. Moreover, since in the explicit scheme the equations are decoupled, the parallelization is straightforward and computationally very effective.

The discretization in time of Equations (13) and (14) is obtained by dividing the time domain $[0,T]$ into N time steps $\Delta t^{n+1}=t^{n+1}-t^n$, such that $T=\sum_{n=1}^{N}\Delta t^n$. Adopting an explicit time integration with the Central Difference Scheme and after computing the matrices $\boldsymbol{M}_{pf}$, $\boldsymbol{K}_\mu$, $\boldsymbol{K}_c^v$, $\boldsymbol{K}_c^p$ and $\boldsymbol{D}$ on the current geometry and mesh[†], the equations are solved with the following steps:

1. Evaluate the mid-step velocities:

$$\boldsymbol{V}_f^{n+\frac{1}{2}}=\boldsymbol{V}_f^n+\frac{1}{2}\boldsymbol{A}_f^n\Delta t^{n+1}, \tag{20}$$

being $\boldsymbol{A}_f^n$ the vector of nodal accelerations.

2. Update the displacements:

$$\boldsymbol{U}_f^{n+1}=\boldsymbol{U}_f^n+\boldsymbol{V}_f^{n+\frac{1}{2}}\Delta t^{n+1}. \tag{21}$$

3. Compute the pressures:

$$\boldsymbol{M}_{pf}\boldsymbol{P}_f^{n+1}=\boldsymbol{M}_{pf}\boldsymbol{P}_f^n-\boldsymbol{S}_f^n\boldsymbol{P}_f^n-\Delta t^{n+1}(\boldsymbol{K}_c^p\boldsymbol{P}_f^n+K_f\boldsymbol{D}\boldsymbol{V}_f^{n+\frac{1}{2}}), \tag{22}$$

where $\boldsymbol{S}_f^n$ is a stabilization matrix (see Appendix A for details).

4. Calculate the pointwise densities by inverting Equation (12). For the ith node, that is:

$$\rho_{f,i}^{n+1} = \rho_{f,0}\left[\frac{\gamma}{K_f}(P_{f,i}^{n+1} - P_{f,0}) + 1\right]^{1/\gamma}. \tag{23}$$

5. Compute the total vector $\boldsymbol{F}_f^{n+1}$ of equivalent nodal forces

$$\boldsymbol{F}_f^{n+1} = -(\boldsymbol{K}_\mu + \boldsymbol{K}_c^v)\boldsymbol{V}_f^{n+\frac{1}{2}} + \boldsymbol{D}^T\boldsymbol{P}_f^{n+1} + \boldsymbol{F}_{\text{ext},f}^{n+1}. \tag{24}$$

6. Compute the mass matrix $\boldsymbol{M}_{v,f}^{n+1}$ using the updated density ρ_f^{n+1}.
7. Evaluate the nodal accelerations $\boldsymbol{A}_f$ through:

$$\boldsymbol{A}_f^{n+1} = (\boldsymbol{M}_{v,f}^{n+1})^{-1}\boldsymbol{F}_f^{n+1}. \tag{25}$$

8. Compute the end step velocities:

$$\boldsymbol{V}_f^{n+1} = \boldsymbol{V}_f^{n+\frac{1}{2}} + \boldsymbol{A}_f^{n+1}\Delta t^{n+1}. \tag{26}$$

The proposed method is conditionally stable and consequently, the size of the time increment Δt should respect the Courant–Friedrichs–Lewy (CFL) stability condition[42]:

$$\Delta t = C_N \min_e\left(\frac{h_e}{c}\right), \tag{27}$$

where h_e is the characteristic element dimension which changes in time, c represents the speed of the dilatational wave in the fluid and C_N is the Courant number, here assumed to be $C_N = 0.9$.

2.4 | FSI problems: coupling PFEM with a FEM solid solver

Let Ω^t be the whole analysis domain defined in the interval $[0, T]$, composed of two non-overlapping Ω_f^t and Ω_s^t subdomains. The subscripts f and s stand for fluid and structure, respectively. Denoting the subdomains boundaries $\Gamma_i^t = \partial\Omega_i^t$, with $i = f, s$, the fluid–structure interface is defined as $\Gamma_{\text{FSI}}^t = \Gamma_f^t \cap \Gamma_s^t$. In each subdomain, Dirichlet and Neumann boundary conditions can be applied on $\Gamma_{D,i}^t$ and $\Gamma_{N,i}^t$, respectively. The FSI problem setting is depicted in Figure 2.

In the fluid domain Ω_f^t, the motion is described by the Navier–Stokes equations presented in the previous paragraph. The motion of the solid domain Ω_s^t is governed by the momentum conservation equation:

$$\rho_s\frac{d^2\boldsymbol{u}_s}{dt^2} = \nabla_x \cdot \boldsymbol{\sigma}_s + \rho_s\boldsymbol{b}_s \quad \text{in } \Omega_s^t \times [0, T], \tag{28}$$

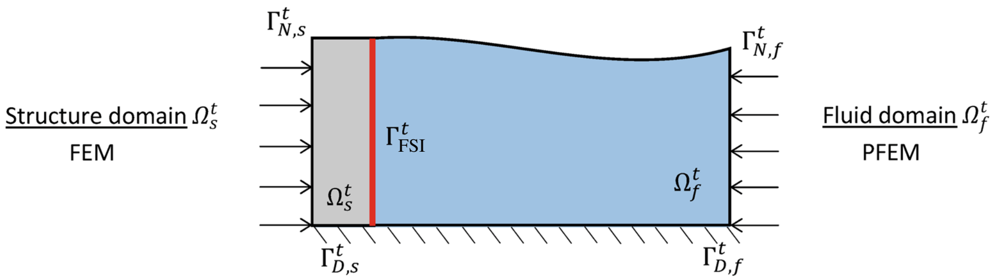


FIGURE 2 PFEM-FEM FSI problem setting.

where $\boldsymbol{u}_s(\boldsymbol{x}, t)$ represents the solid displacement field. Equation (28) is complemented with standard initial and boundary conditions. The weak form of Equation (28) is given by:

$$\int_{\Omega_s^t} \boldsymbol{v}_s \cdot \rho_s \frac{d^2\boldsymbol{u}_s}{dt^2} d\Omega_s = -\int_{\Omega_s^t} \boldsymbol{\sigma}_s : \nabla_x \boldsymbol{v}_s d\Omega_s + \int_{\Omega_s^t} \boldsymbol{v}_s \cdot \rho_s \boldsymbol{b}_s d\Omega_s + \int_{\Gamma_{s,N}^t} \boldsymbol{v}_s \cdot \boldsymbol{h}_s d\Gamma$$

$$\forall \boldsymbol{v}_s | \boldsymbol{v}_s = \boldsymbol{0} \text{ on } \Gamma_{s,D}^t, \tag{29}$$

where $\boldsymbol{h}_s, \boldsymbol{n}_s$ are the surface tractions and the outward normal to the boundary $\Gamma_{s,N}^t$, respectively. Following a standard discretization, the displacement field is expressed as

$$\boldsymbol{u}_s(\boldsymbol{x}, t) = \boldsymbol{N}_s^u(\boldsymbol{x})\boldsymbol{U}_s(t), \tag{30}$$

being $\boldsymbol{N}_s^u$ the matrix of shape functions and $\boldsymbol{U}_s(t)$ the vector of solid nodal displacements. The resulting semi-discretized governing equation is given by:

$$\boldsymbol{M}_s \frac{d^2\boldsymbol{V}_s}{dt^2} = \boldsymbol{F}_{\text{ext},s} - \boldsymbol{F}_{\text{int},s} = \boldsymbol{F}_s, \tag{31}$$

where $\boldsymbol{M}_s$ denotes the solid mass matrix; $\boldsymbol{F}_{\text{ext},s}, \boldsymbol{F}_{\text{int},s}, \boldsymbol{F}_s$ are the external, internal, and resultant equivalent nodal force vectors, respectively.

The FSI problem is here addressed with a partitioned approach following the scheme presented in Reference 11. The scheme exploits the PFEM for fluid parts and a standard FEM for solid parts. The two solvers are coupled through a domain decomposition approach based on the Gravouil and Comberscure (G&C) algorithm which was originally developed for structural dynamics[43] and then extended to FSI problems.[11,44,45] The approach ensures strong coupling and stability at the fluid–structure interfaces.[46,47] In the examples of Section 4, the solid part will be solved with the commercial software ABAQUS/Explicit allowing for an accurate modeling of complex problems, thanks to advanced material constitutive laws and rich finite element libraries.

The G&C scheme[46,47] consists in solving separately the fluid and solid domains first, as if there were no interaction between them (*free* problem). Then, the two sets of solutions are used to evaluate the interface tractions, which play the role of Lagrangian multipliers, to impose the velocity continuity at the fluid–structure interface (*link* problem). Finally, the real FSI solutions are obtained by summing the contributions from the *free* and *link* subproblems. The solution scheme is sketched below.

1. solution of the *free* problem

$$\boldsymbol{M}_{v,f}^{n+1}\boldsymbol{A}_{\text{free},f}^{n+1} = \boldsymbol{F}_f^{n+1} \quad \text{in } \Omega_f^{n+1}. \tag{32}$$

$$\boldsymbol{M}_s\boldsymbol{A}_{\text{free},s}^{n+1} = \boldsymbol{F}_s^{n+1} \quad \text{in } \Omega_s^{n+1}. \tag{33}$$

2. solution of the *link* problem

$$\boldsymbol{M}_{v,f}^{n+1}\boldsymbol{A}_{\text{link},f}^{n+1} = -\boldsymbol{C}_f^T\boldsymbol{\Lambda}^{n+1} \quad \text{on } \Gamma_{\text{FSI}}^{n+1}, \tag{34}$$

$$\boldsymbol{M}_s\boldsymbol{A}_{\text{link},s}^{n+1} = \boldsymbol{C}_s^T\boldsymbol{\Lambda}^{n+1} \quad \text{on } \Gamma_{\text{FSI}}^{n+1}, \tag{35}$$

where $\boldsymbol{C}_i$ is the boolean matrix extracting the nodal interface variables from the global vector and $\boldsymbol{\Lambda}$ the vector containing the nodal Lagrangian multipliers, computed through:

$$\boldsymbol{H}\boldsymbol{\Lambda}^{n+1} = \boldsymbol{C}_f\boldsymbol{V}_{\text{free},f}^{n+1} - \boldsymbol{C}_s\boldsymbol{V}_{\text{free},s}^{n+1}, \tag{36}$$

being $\boldsymbol{H}$ the dynamic condensation operator, defined as:

$$\boldsymbol{H} = \boldsymbol{C}_f(\boldsymbol{M}_{v,f}^{n+1})^{-1}\boldsymbol{C}_f^T\frac{\Delta t^{n+1}}{2} + \boldsymbol{C}_s\boldsymbol{M}_s^{-1}\boldsymbol{C}_s^T\frac{\Delta t^{n+1}}{2}. \tag{37}$$

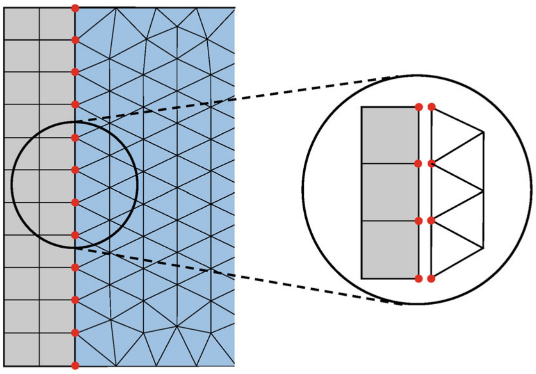


FIGURE 3 PFEM-FEM fluid-structure conforming mesh at the interface.

3. solution of the coupled FSI problem

$$\boldsymbol{A}_f^{n+1} = \boldsymbol{A}_{\text{free},f}^{n+1} + \boldsymbol{A}_{\text{link},f}^{n+1} \quad \text{in } \Omega_f^{n+1}. \tag{38}$$

$$\boldsymbol{A}_s^{n+1} = \boldsymbol{A}_{\text{free},s}^{n+1} + \boldsymbol{A}_{\text{link},s}^{n+1} \quad \text{in } \Omega_s^{n+1}. \tag{39}$$

The proposed approach can be generalized also to different time step sizes in the fluid and solid domains, see for example, Reference 43. In the present work, the remeshing process is performed only in the fluid sub-domain, whereas nodal connectivities are not changed throughout the analysis in the solid subdomain. Moreover, conforming fluid and solid meshes at the FSI interface are considered (see Figure 3). More details on the coupling scheme and nonconforming meshes can be found in Reference 11.

> *Remark* 1. It is important to recall that the considerations that will follow in the next sections do not depend on the used coupling scheme. The partitioned approach allows to solve separately the fluid and the structure and, hence, to use the hybrid Lagrangian–Eulerian approach only in the fluid part, without affecting the structural solver.

3 | HYBRID LAGRANGIAN–EULERIAN SCHEME

3.1 | Fluid problems

As already underlined, in some specific applications the accurate tracking of free-surfaces and interfaces between different fluids is fundamental. Typically, in the Eulerian framework free-surfaces and interfaces are identified using specific techniques like Level Set[2] or Volume of Fluid.[3] On the contrary, in the Lagrangian framework, interfaces are naturally tracked by nodal positions. The price to pay for this simple treatment of free-surfaces and interfaces is mesh distortion, calling for a continuous domain remeshing. Typically, a global remeshing should be performed even when large distortions of the fluid mesh appear only in limited regions of the domain. Localized remeshing is difficult to perform in the PFEM and up to now it has been used only in very few and specific situations. To overcome this difficulty, in the part of the fluid domain close to free-surfaces or to evolving interfaces, a Lagrangian approach is used, while in regions where the domain is not varying significantly, an Eulerian approach is adopted.

The proposed approach is computationally very efficient. On one hand, compared to standard Eulerian methods, the present approach avoids the difficulties connected with the tracking of free-surfaces (or of interfaces). On the other hand, with respect to the standard Lagrangian approach, remeshing is performed only in a part of the fluid domain, significantly limiting the time spent in mesh generation. The computational gain will be significantly more impacting in 3D analyses.

The simpler approach for the implementation of the proposed technique is to define ab-initio the Lagrangian (where remeshing is performed) and Eulerian regions in the domain (see Figure 4) and to keep them fixed in time. Alternatively, the two regions can evolve in time, driven by the position of free-surfaces and interfaces, as it will be discussed in the next section for the case of FSI problems.

Remark 2. In the proposed formulation, a node can be either Lagrangian or Eulerian. Therefore, there will be Lagrangian elements in which the convective term does not appear ($\boldsymbol{c}_f = \boldsymbol{0}$ or $\boldsymbol{r}_f = \boldsymbol{v}_f$), Eulerian elements in which the convective term is present ($\boldsymbol{c}_f = \boldsymbol{v}_f$ or $\boldsymbol{r}_f = \boldsymbol{0}$) and hybrid elements in which some nodes are Eulerian and others are Lagrangian. These elements are assembled using an ALE formulation by setting the convective velocity to zero in the Lagrangian nodes and to the fluid velocity in the Eulerian nodes.

Remark 3. In the ALE framework, an additional equation is typically needed to govern the mesh velocity evolution. In the present approach, this equation is not required because the mesh velocity is set equal to the fluid velocity in the Lagrangian nodes and zero in the Eulerian ones.

3.2 | FSI problems

In complex FSI problems involving fast fluid domain evolution and large structural deformations, the proposed Lagrangian–Eulerian formulation represents an efficient way to overcome the limits related to each of the two kinematic models. In the present multidomain PFEM-FEM FSI framework, the overall fluid computational domain is divided into an Eulerian region and a Lagrangian remeshing one. The latter, as shown in Figure 5, contains the solid body throughout the analysis and its position is updated when the solid body comes close to the interface between the two subregions. Inside the remeshing zone, all nodes are treated as Lagrangian making the automatic definition of the evolving fluid–solid interface simple. By contrast, all the outside nodes are set to be Eulerian, hence fixed in time. The mesh regeneration process, typical of the standard PFEM approach, will be performed only in the Lagrangian remeshing region, hence in a reduced portion of the overall fluid domain.

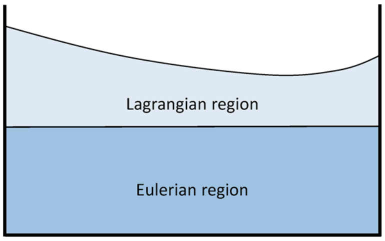


FIGURE 4 Fluid domain partition in hybrid Lagrangian–Eulerian formulation.

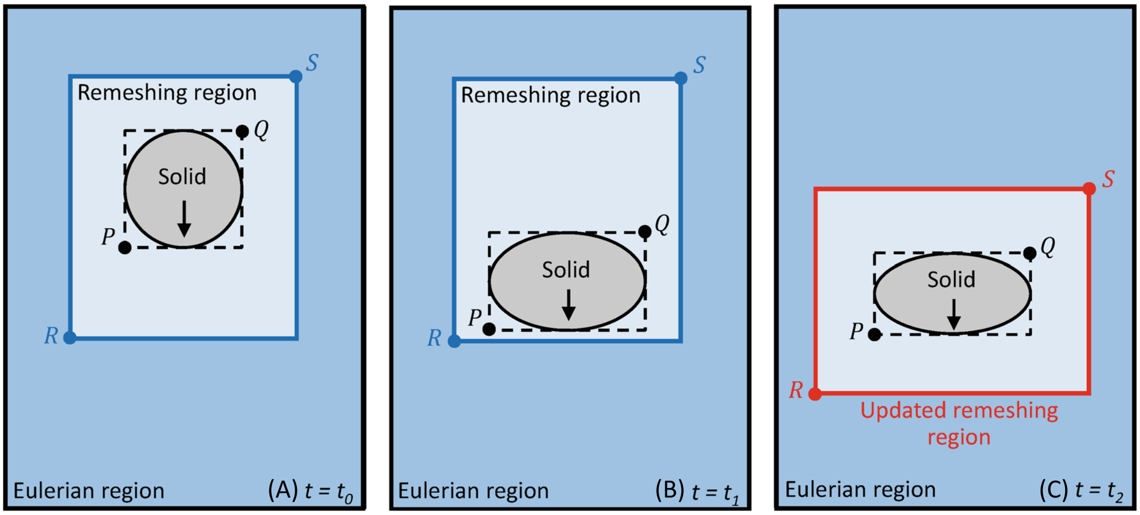


FIGURE 5 Domain partition in hybrid Lagrangian–Eulerian fluid–structure interaction problems. The Lagrangian remeshing region is identified by extending the bounding box enclosing the solid and it is updated when the solid reaches its borders (B, C). Eulerian description is adopted elsewhere. (A) $t = t_0$; (B) $t = t_1$; (C) $t = t_2$.

The new mesh generated in the Lagrangian part should respect the boundaries at the Lagrangian–Eulerian borders. A possibility to guarantee the respect of these boundaries is the use of a constrained Delaunay Tessellation. Alternatively, if one wants to use the standard Delaunay Tessellation, the Lagrangian region should be convex, to avoid any possible overlapping between the two regions. In this work, this last option has been chosen leading to a rectangular Lagrangian region in 2D and a parallelepiped in 3D.

To completely contain the structure throughout the computation, the position and spatial extension of the remeshing region have to be updated when the structure approaches its borders. This is achieved by means of the technique explained below and illustrated in Figure 5. Since the Lagrangian formulation is adopted in the region containing the structure, the FSI boundaries are automatically defined at each time step. Exploiting this feature, the region occupied by the solid at time t is identified through the position of the two points $P(x_P(t), y_P(t))$ and $Q(x_Q(t), y_Q(t))$ (Figure 5A,B), where $x_P(t)$ and $x_Q(t)$ are the minimum and maximum horizontal coordinate of the FSI interface and $y_P(t)$ and $y_Q(t)$ are the minimum and maximum vertical coordinate of the FSI interface. Using the points $P(t)$ and $Q(t)$, a bounding box exactly enclosing the solid body is defined. It is important to recall that this bounding box varies with time, because the coordinates of points $P(t)$ and $Q(t)$ follow the FSI interface. In order to define the position and size of the remeshing region containing the moving solid body, user-defined enlarging factors in horizontal β_x^+, β_x^- and vertical directions β_y^+, β_y^- are introduced. The rectangular remeshing region can be defined through its two opposite corners $R(x_R(t), y_R(t))$ and $S(x_S(t), y_S(t))$ (Figure 5C). Their position is obtained by increasing the extension of the solid bounding box by a factor depending on the mesh mean element size l_{mean}:

$$R(t) : \begin{cases} x_R(t) = x_P(t) - \beta_x^- l_{\text{mean}} \\ y_R(t) = y_P(t) - \beta_y^- l_{\text{mean}} \end{cases} \quad S(t) : \begin{cases} x_S(t) = x_Q(t) + \beta_x^+ l_{\text{mean}} \\ y_S(t) = y_Q(t) + \beta_y^+ l_{\text{mean}} \end{cases} \tag{40}$$

In the proposed method, the remeshing region is updated whenever the minimum distance between the solid bounding box and the boundaries of the remeshing region is less than a predefined threshold. In the examples considered here, this threshold is set to 1.2 l_{mean}, while different enlarging factors β are used. Four enlarging factors have been introduced in order to provide higher flexibility, so that the remeshing region can be modified differently in different directions, depending on the problem. In the special case where $\beta_x^+ = \beta_x^- = \beta_y^+ = \beta_y^-$, a uniform enlargement is obtained leading to a square remeshing region.

The nodes on the boundary of the remeshing region (blue rectangle in Figure 6A) are Eulerian and therefore in general do not move. However, following the movement of the structure, the position and size of the remeshing region changes (red box in Figure 6B). Now, the new boundaries of the red box (or part of them) may pass through the previous mesh, cutting some elements (see Figure 6B). On these parts, the Lagrangian–Eulerian border needs to be straightened and Eulerian nodes placed onto it. Exploiting the standard PFEM mesh optimization algorithms, a borders straightening operation and local mesh optimization are performed in correspondence of Lagrangian-Eulerian interfaces. The former moves a suitable number of nodes to the boundaries of the new remeshing region, with an offset distance equal to the mesh characteristic length (Figure 6C). The latter improves the mesh quality in the transition region by removing elements that result overly distorted due to the straightening operation, so that the stable time increment is not compromised

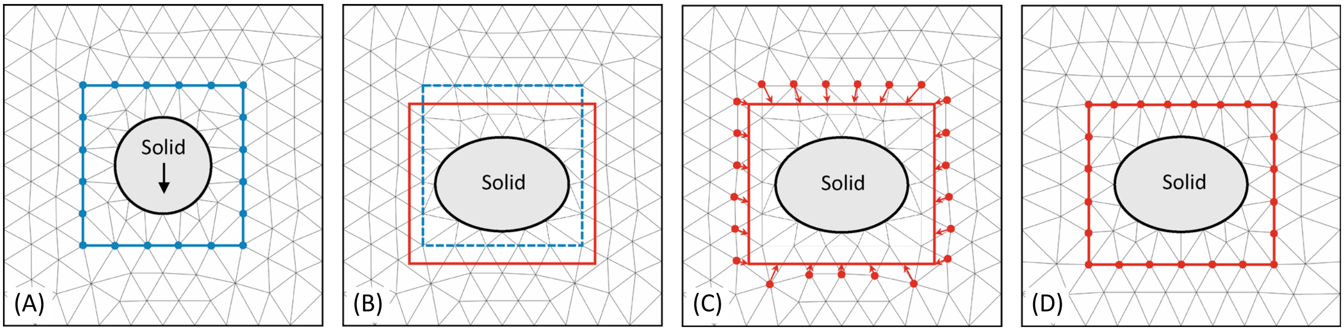


FIGURE 6 Two-dimensional (2D) Lagrangian–Eulerian interface straightening. Blue box in (A defines initial remeshing region. Solid bounding box not shown. Starting from a configuration (A), the Lagrangian–Eulerian interface moves to a new position (red box in (B)), nodes are moved on the new interface (C) and a new high-quality mesh is created (D).

Algorithm 1. Remeshing region update

```
create first mesh
compute position of points $P(t_0)$ and $Q(t_0)$ defining the solid bounding box at $t_0$
identify corners $R(t_0)$ and $S(t_0)$ of remeshing region through enlarging factors at $t_0$
for $t = 0, \dots, T$ do
  if solid is too close to borders of remeshing region then
    identify new corners $R(t)$ and $S(t)$ of remeshing region through enlarging factors
    where needed, straighten interfaces between Lagrangian and Eulerian regions
    generate new mesh
    locally improve mesh where needed
  end if
  regenerate Lagrangian mesh if too distorted
  solve FSI problem by keeping Eulerian mesh fixed
  update position of points $P(t)$ and $Q(t)$
end for
```

(Figure 6D). Even though the two operations are needed, they are performed only few times, when the remeshing region is updated. The proposed technique has also been sketched in the following Algorithm 1.

It is important to highlight that the solid bounding box is determined based on the solid boundaries. Consequently, if there are multiple solids present, each solid will have its own box. In cases where two boxes are sufficiently close to each other (e.g., distance lower than the predefined threshold), they will merge together. On the other hand, if the bodies start moving away from each other, the merged boxes will separate once again.

When the remeshing zone consists of a limited portion of the overall domain, being the only part containing Lagrangian nodes, element distortion occurs only in this subregion, drastically reducing the computational cost related to mesh generation with respect to standard PFEM approaches. Moreover, bad quality elements, which reduce the stable time step size due to small characteristic lengths in explicit analyses, are localized in the remeshing region only. Therefore, mesh quality improvements are performed in this limited zone, further reducing the total computational burden.[30]

For the sake of simplicity, the definition of the remeshing region and the straightening operation have been described for 2D cases. The extension to 3D problems is straightforward: there will be a parallelepiped instead of a rectangle. As in 2D analyses, the parallelepiped dimension and spatial extension are updated at each time step in order to fully enclose the moving 3D solid body.

4 | NUMERICAL EXAMPLES

This section reports the results of numerical tests performed with the proposed hybrid Lagrangian–Eulerian approach. In all the presented examples, a weakly compressible Newtonian fluid with a dilatational wave speed of $c = 350\,\text{m/s}$ is used. For all the examples, the reported computational costs refer to a standard workstation with an OPENMP parallelized solver with eight threads. Fluid flow and FSI problems are analyzed comparing the obtained results to those available in the literature.

4.1 | Lid-driven cavity flow

The lid-driven cavity flow considers a 2D square of length $L = 1\,\text{m}$ containing a Newtonian fluid initially at rest. The geometry of the problem is depicted in Figure 7. The fluid is characterized by the initial density $\rho_{f,0} = 1000\,\text{kg/m}^3$ and dynamic viscosity $\mu_f = 1$ Pa s. A unitary horizontal velocity is imposed at the top edge of the square $v_x = \bar{v} = 1\,\text{m/s}$ and $v_y = 0\,\text{m/s}$, while the remaining edges are kept fixed. As pointed out in Reference 26, a pure Lagrangian description of the problem would lead to displacements of the nodes at the top edge and the Lagrangian definition of the boundary would be lost. Although there exist other possibilities to treat this problem with a fully Lagrangian approach (see e.g., Reference 48), in this work nodes lying on the top edge are simply treated as Eulerian. Three different cases are considered, see

Figure 8: (a) Lagrangian domain with Eulerian top boundary only (*LagDom*, Figure 8A); (b) entire domain treated as Eulerian (*EulDom*, Figure 8B); (c) hybrid domain with Lagrangian and Eulerian subdomains, with separating interface placed at the horizontal centerline (*HybDom*, Figure 8C).

The computational mesh is composed of 28,224 fluid nodes with approximately 55k elements. In all three cases, the same initial mesh is adopted. No mesh distortion occurs in the second case since all nodes are fixed in time. By contrast, as the analysis proceeds, mesh distortion occurs in the entire and in the lower part of the domain for the first and third cases, respectively. Horizontal v_x and vertical v_y velocity profiles along the centerlines at the steady state are plotted in Figure 9. A good agreement is achieved between different velocity profiles and those presented in References 26 and 49, confirming that the proposed technique leads to correct and accurate results. Furthermore, the interface between the Lagrangian and Eulerian domains is practically indistinguishable in the *HybDom* case (see Figure 10).

> *Remark* 4. In the current example, there is no specific reason to use a hybrid approach, the solution can be obtained effectively with a full Eulerian (or also a full Lagrangian) approach. The example is here used to validate the approach and to show that in the transition regions the unknown fields are very accurately represented.

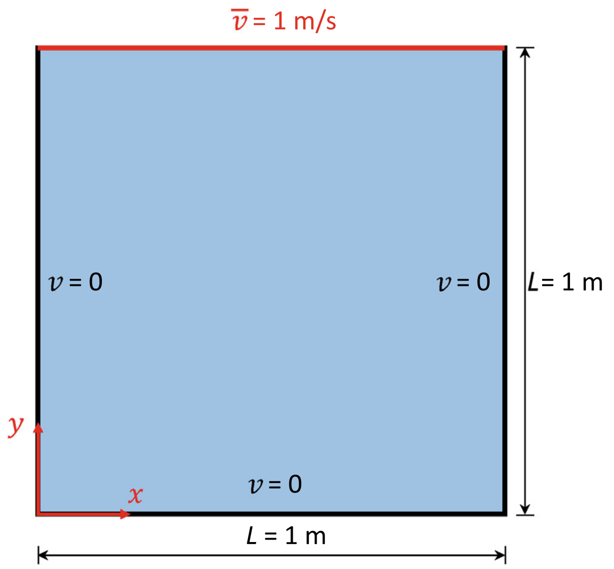


FIGURE 7 Lid-driven cavity flow. Geometry of the problem.

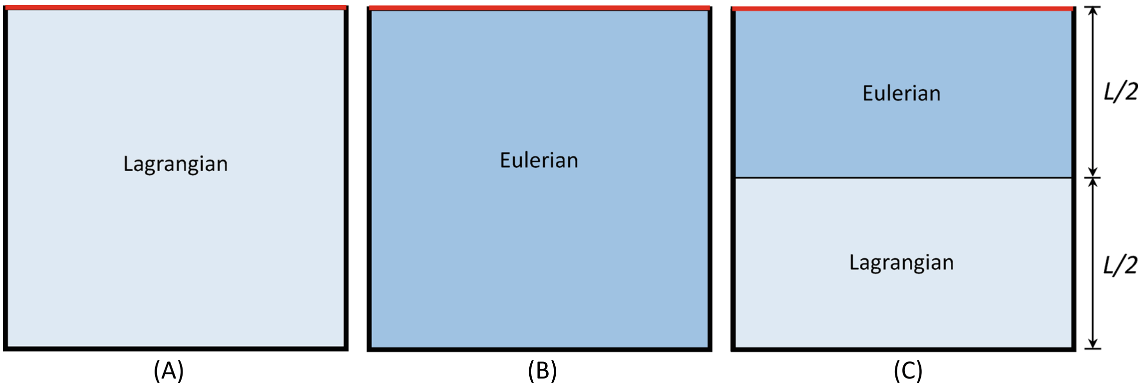


FIGURE 8 Lid-driven cavity flow. Three different cases analyzed. (A) Lagrangian domain with Eulerian top boundary (*LagDom*), (B) Eulerian domain (*EulDom*), (C) hybrid domain with Lagrangian and Eulerian subdomains and interface placed at the horizontal centerline (*HybDom*). A) *LagDom*; (B) *EulDom*; (C) *HybDom*.

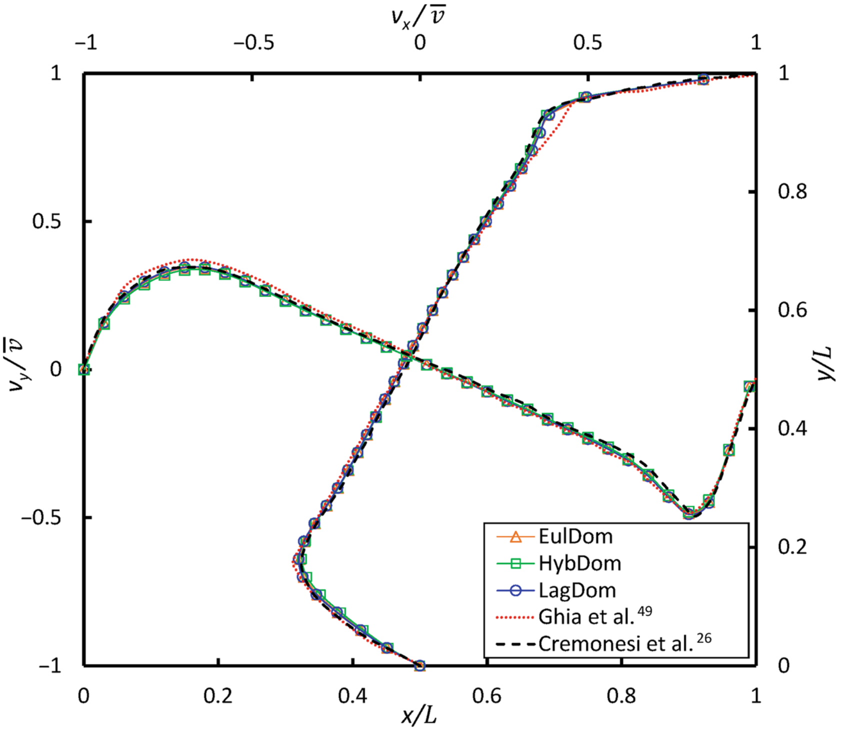


FIGURE 9 Lid-driven cavity flow. Velocity profiles at horizontal and vertical centerlines.

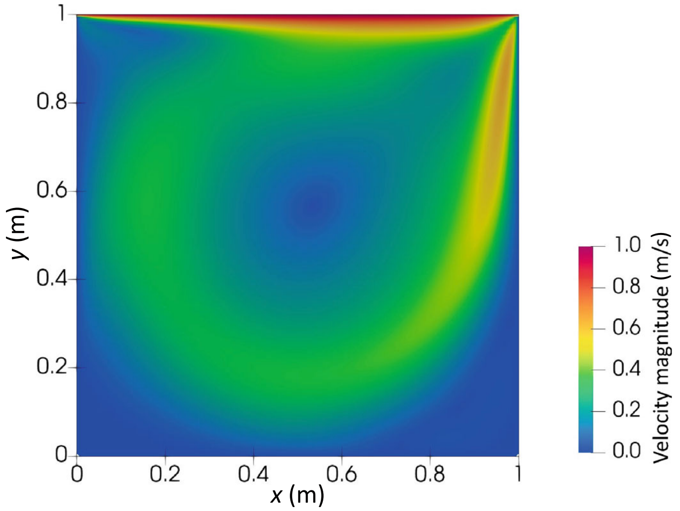


FIGURE 10 Lid-driven cavity flow. Velocity contour plot at steady state (*HybDom*, $t = 50$ s).

4.2 | 2D free sloshing

The 2D sloshing test presented in Reference 50 is considered. A fixed rigid tank contains an inviscid Newtonian fluid characterized by an initial density $\rho_{f,0} = 1000\ \text{kg/m}^3$. The fluid initial configuration is depicted in Figure 11 with the free surface profile $H(x, t = 0)$ described by the following expression:

$$H(x, t = 0) = A - a\ \cos[k(x + \lambda/2)], \tag{41}$$

with the constant $A = 1$ m, the wave amplitude $a = 0.1$ A, the wave length $\lambda = 2$ m and $k = 2\pi/\lambda$.

The fluid domain is discretized by 22,044 nodes and approximately 43k triangular elements whose mean size is 0.01 m. Considering that the motion of the upper region of the domain is governed by the time evolution of the free surface, the upper part of the domain is modeled with a Lagrangian approach. On the contrary, the lower part represents a confined region characterized by fixed geometry without free surfaces and consequently is described with an Eulerian approach. The transition line between the two subdomains is fixed and set at $y = h = 0.4\ m$.

Figure 12 shows the vertical position in time of the midpoint P (see Figure 11) as computed by the proposed approach and as resulting from the analytical and numerical solutions reported in Reference 50. The problem has been analyzed with both the hybrid Lagrangian-Eulerian approach (*HybDom* curve in Figure 12) and the fully Lagrangian PFEM (*LagDom* curve in Figure 12). The two approaches show a good agreement, with the curve of the proposed hybrid approach almost perfectly superimposed to the standard fully Lagrangian technique and very close to the analytical an numerical solutions of Reference 50. Furthermore, pressure contour plots from the two approaches at time $t = 0.4$ s are depicted in Figure 13. It is worth noticing that the transition between the Lagrangian and Eulerian regions is practically indistinguishable, proving the effectiveness of the presented method.

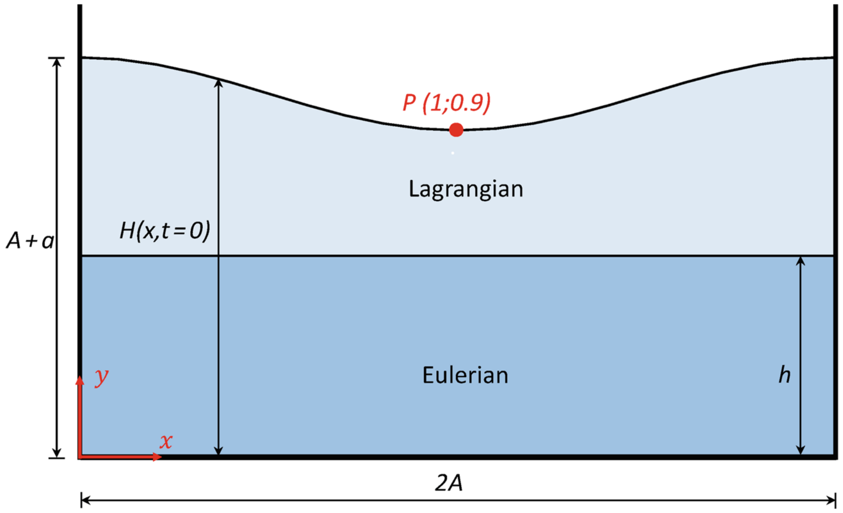


FIGURE 11 Two-dimensional free sloshing. Initial problem configuration.

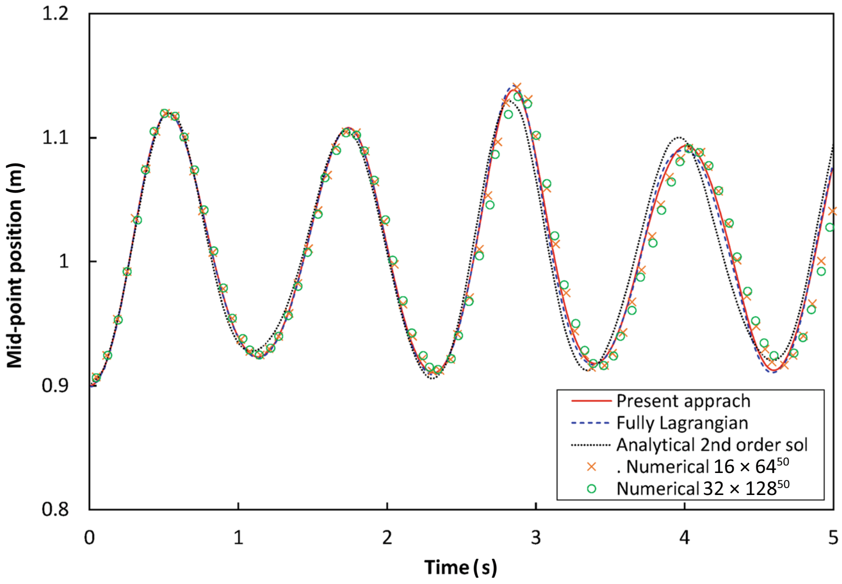
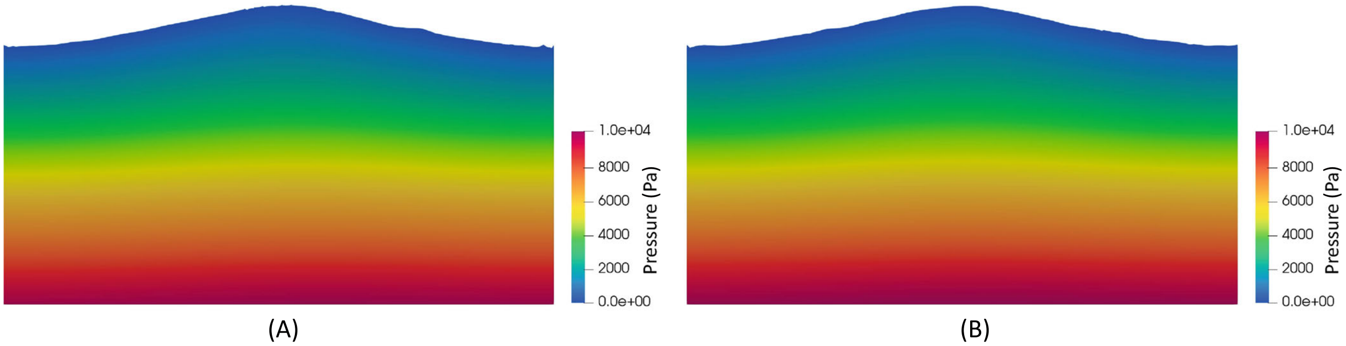


FIGURE 12 Two-dimensional free sloshing. Free surface height evolution in time at the midpoint P.



(A)

(B)

FIGURE 13 Two-dimensional free sloshing. Pressure contour plots at time $t = 0.4$ s obtained from the proposed approach (A) and standard fully Lagrangian approach (B). (A) *HybDom*; (B) *LagDom*.

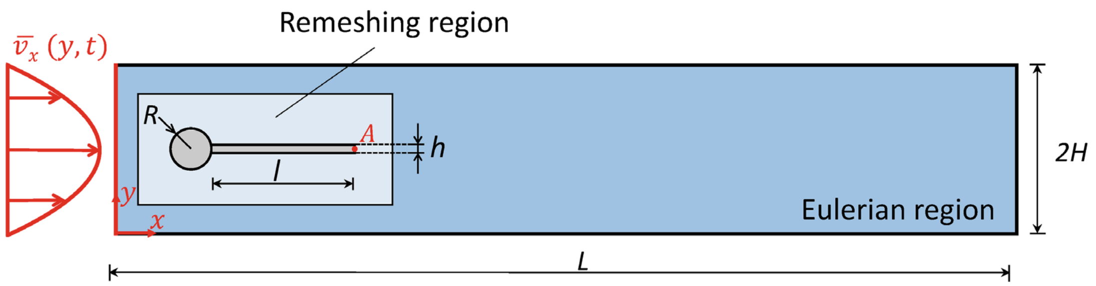


FIGURE 14 Turek–Hron benchmark. Problem geometry.

4.3 | Turek–Hron FSI benchmark

The Turek–Hron test introduced in Reference 51 represents a classical benchmark example for FSI validation. The problem consists of a 2D fluid channel hosting a rigid fixed cylinder of radius R and an elastic beam clamped to the cylinder. The channel contains a Newtonian fluid. The geometry is reported in Figure 14 whereas the geometrical and physical data are listed in Table 1. The test here addressed corresponds to the case *FSI2* presented in Reference 51, in which the ratio between the structural and fluid density $\rho_s/\rho_{f,0} = 10$. A parabolic velocity profile $\bar{v}_x(y,t)$ is imposed to the Eulerian nodes at the inlet section $x = 0$.

$$\bar{v}_x(y,t) = \left[1.5\bar{V}\frac{y(2H-y)}{H^2}\right] f(t), \tag{42}$$

where $f(t)$ is a gradual amplification function of time, introduced to prevent instabilities in the initial transient phase, defined as:

$$\begin{aligned} f(t) &= (1-\cos(\pi t/2))/2 \quad \text{for } t \leq 2 \text{ s}, \\ f(t) &= 1 \quad \text{for } t > 2 \text{ s}. \end{aligned} \tag{43}$$

The imposed inflow condition makes the clamped beam oscillate progressively in time. It can be noticed that the region in which the structure can move is relatively small compared to the whole computational domain. Therefore, the problem is solved through the hybrid Lagrangian–Eulerian approach in which only the region close to the cylinder and to the elastic beam is treated as Lagrangian. This choice leads to a smaller remeshing zone, reducing significantly the overall mesh regeneration time with respect to a full Lagrangian solution (with Eulerian inlet and outlet boundaries).

Since the beam fluctuates in a confined portion of the channel, the remeshing region is kept fixed by setting a sufficiently large extension through enlarging factors ($\beta_x^+ = 12, \beta_x^- = 25, \beta_y^+ = 18, \beta_y^- = 18$), so that the beam does not reach

TABLE 1 Input data Turek–Hron benchmark.

Geometry and material data	
L (m)	1.5[a]
$2H$ (m)	0.41
l (m)	0.35
h (m)	0.02
R (m)	0.05
$\rho_{f,0}$ (kg/m^3)	1000
μ_f (Pa s)	1
ρ_s (kg/m^3)	10 000
E_s (MPa)	1.4
ν_s (−)	0.4
$\overline{V}$ (m/s)	1
Cylinder center coordinates (m)	(0.2;0.2)
Point A coordinates (m)	(0.6;0.2)

[a] A reduced domain of length $L = 1.5\ m$ has been considered with respect to the one in Reference 51.

TABLE 2 Turek–Hron benchmark.

	Lagrangian	**Present approach**
Computational time	38 h	23 h
% solution	69.23	94.17
% remeshing	30.70	5.77
% output	0.07	0.06
Remeshing time/number of time steps	0.0017	0.0002
total time/number of time steps	0.0056	0.0039
	Simulated time: 15 s	

Note: Comparison of computing time between Lagrangian and proposed approaches.

the borders during the analysis (see Figure 14). Moreover, it is reasonable to assume that the region close to the outlet section does not affect the beam motion. Therefore, the length of the channel has been reduced to $L = 1.5$ m. The fluid domain has been discretized with elements having average size of 0.004 m, leading to approximately 39k nodes and 77k elements. The computational costs in the two approaches are reported in Table 2, showing a significant decrease of the remeshing effort in the present method. It can be observed that under the same analysis condition, both remeshing time and total computational time are lower in the hybrid Lagrangian–Eulerian approach.

Considering that the two approaches can have different time evolution of the meshes, the number of time steps can be slightly different (stable time step size depends on the smallest element in the mesh). To guarantee a fair comparison between the two approaches, also the remeshing time over the number of time steps and the total time over the number of time steps are evaluated. The first index measures how the remeshing procedure costs for a single time step and the second evaluates the time cost of a single time step. In the present test, both indexes shows the effectiveness of the proposed approach.

Fluid velocity contour plots at different time steps are shown in Figure 15, while the vertical displacement of the beam at point A is plotted as a function of time in Figure 16. It can be observed that the vertical displacement increases in the initial transient phase until the steady-state condition is reached, characterized by a constant oscillating amplitude ΔU_y and frequency f. The latter are listed in Table 3 and compared to those reported in References 51 and 26.

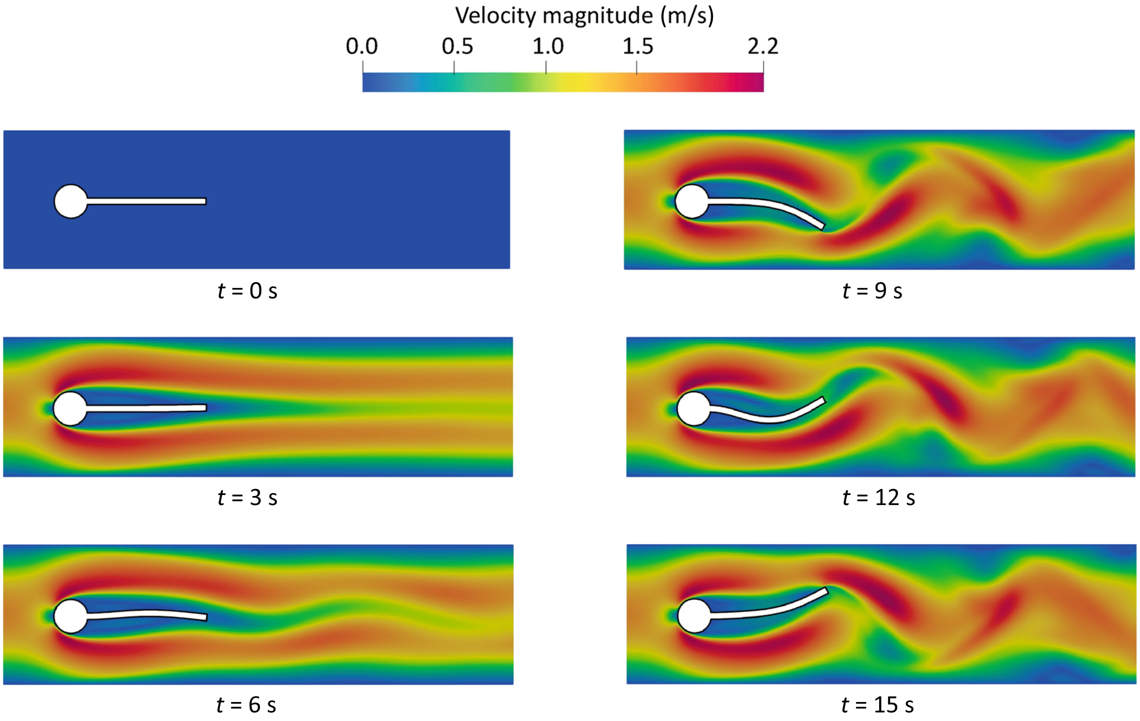


FIGURE 15 Turek-Hron benchmark. Hybrid approach. Snapshots of fluid velocity magnitude at different times.

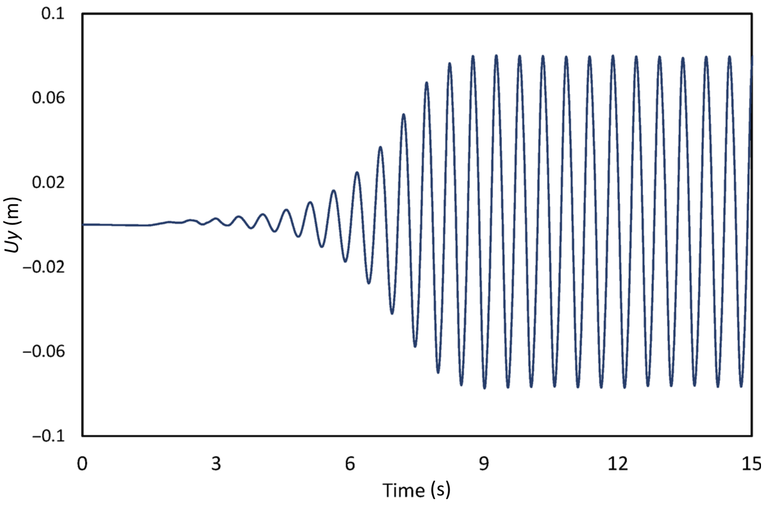


FIGURE 16 Turek–Hron benchmark. Hybrid approach. Time history of beam vertical displacement at point A.

TABLE 3 Turek–Hron benchmark.

	ΔU_y (m)	f (Hz)
FSI2[51]	0.161	2.00
Lagrangian	0.158	1.93
Hybrid	0.158	1.92

Note: Comparison of steady-state amplitude and frequency of beam oscillation.

4.4 | Swimming of a 2D jellyfish

The simplified swimming of a jellyfish, proposed in Reference 52, is considered. It consists of a jellyfish swimming in an 8 m × 8 m fluid domain containing a Newtonian fluid characterized by an initial density $\rho_{f,0} = 1$ kg/m^3 and dynamic viscosity $\mu_f = 0.01$ Pa s. The initial shape of the jellyfish is described through the following parametric equation:

$$\boldsymbol{x}(s) = (\gamma s, \beta - \alpha(\gamma s)^2 - \eta(\gamma s)^4) \quad \text{for } s \in [-1, 1]. \tag{44}$$

$$\begin{aligned} \alpha = 0.03, \quad \beta = 1.0, \quad \gamma = 2.6, \quad \eta = 0.05 \quad & \text{for the top curve} \\ \alpha = 0.02, \quad \beta = 0.2006, \quad \gamma = 2.45, \quad \eta = 0.045 \quad & \text{for the bottom curve.} \end{aligned}$$

The geometry of this FSI problem is depicted in Figure 17. The top and bottom profiles are joined through tangent circular arches at the ends. To simulate the upward motion of the immersed jellyfish, horizontal (x-direction) body forces $\boldsymbol{F}_i$ characterized by a period $T = 50$ s are applied in its lower part. The subscript $i = 1, 2$ indicates quantities related to the leg 1 and 2. The body forces have the same magnitude but opposite directions on the two legs and they achieve their maximum value at the jellyfish tips. Within one time period, $0 \leq t \leq T$, the magnitude of the applied horizontal loads $\boldsymbol{F}_i$ is given by the following expression:

$$F_1(r_1, t) = \begin{cases} 10(2 - r_1)(1 - t/3) & \text{for } r_1 \leq 2 \text{ m and } t \leq 3 \text{ s} \\ 0 & \text{otherwise} \end{cases}, \tag{45}$$

$$F_2(r_2, t) = -F_1(r_1, t), \tag{46}$$

where r_i is a polar coordinate defined as:

$$r_i = |\boldsymbol{X} - \boldsymbol{X}_i| \qquad \text{for } r_i \leq 2 \text{ m}, \tag{47}$$

where $\boldsymbol{X}$ is the vector containing the nodal coordinates x and y and $\boldsymbol{X}_i$ are the coordinates of the two tips (see Figure 17).

The jellyfish behavior is described by a Neo-Hookean hyperelastic material model, with strain energy potential U defined as[53]:

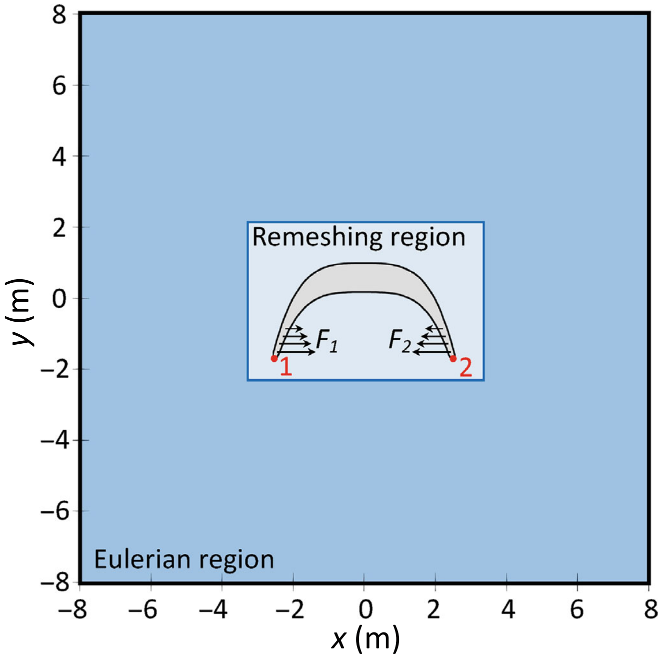


FIGURE 17 Swimming of two-dimensional jellyfish. Problem geometry and applied body forces.

$$U = C_{10}(I_1 - 3) + \frac{1}{D_1}(J^{\text{el}} - 1)^2, \tag{48}$$

where I_1 and J^{el} are the first deviatoric strain invariant and the elastic volume ratio, respectively. The material parameters C_{10} and D_1 can be defined through the solid initial shear modulus $\mu_0 = 2C_{10}$ and bulk modulus $K_0 = 2/D_1$. In the present example, the material parameters have been set to $C_{10} = 2.5$ Pa and $D_1 = 0$ (incompressible material). The structural density is set equal to the fluid one $\rho_s = \rho_{f,0} = 1$ kg/m^3.

The computational domain is subdivided into three subdomains: the structural subdomain, consisting of the jellyfish modeled with Abaqus/Explicit; the Lagrangian subdomain, that is, the remeshing region defined by a moving box containing the Lagrangian elements surrounding the jellyfish, modeled with the Lagrangian PFEM; the Eulerian subdomain, defined over the remainder of the computational domain, modeled with the Eulerian PFEM. The total mesh consists of 122,679 nodes and approximately 243k elements, whose average size is 0.05 m. As introduced in Section 3.2, a rectangular remeshing box enclosing the jellyfish is here defined (see Figure 17) with enlarging factors $\beta_x^+ = 15, \beta_x^- = 15, \beta_y^+ = 25, \beta_y^- = 15$. A higher value of β_y^+ has been used to account for the solid body upward motion: this choice enlarges the Lagrangian remeshing region in the positive y-direction, allowing for a reduction of the number of straightening operations. The size and position of the remeshing region will be updated when the jellyfish reaches its borders. Fluid nodes inside the remeshing region are treated as Lagrangian since they are close to the FSI interface, whereas all the other nodes, including those on the interface between the Eulerian and Lagrangian subdomains, are set to be Eulerian. The runtime triangulation procedure occurs only in the limited area of the remeshing box.

Snapshots of the jellyfish configuration at various time steps are shown in Figure 18, where the results are compared to those in Reference 52. It can be observed that good agreement is obtained even though a small discrepancy is found in the last configuration at $t = 24.22$ s. This difference could be justified by a different constitutive law adopted for the solid part: a viscoelastic incompressible model is used in Reference 52, while Abaqus/Explicit hyperelastic Neo-Hookean model without viscosity is assumed here for the solid part[‡]. The remeshing region containing the jellyfish during its motion in the fluid is depicted in Figure 19. The computational effort in both the fully Lagrangian and the proposed approach is reported in Table 4. The computational burden related to the mesh generation is reduced in the hybrid Lagrangian–Eulerian formulation. In addition, the remeshing and total time per time step, obtained under the same analysis conditions, are drastically reduced.

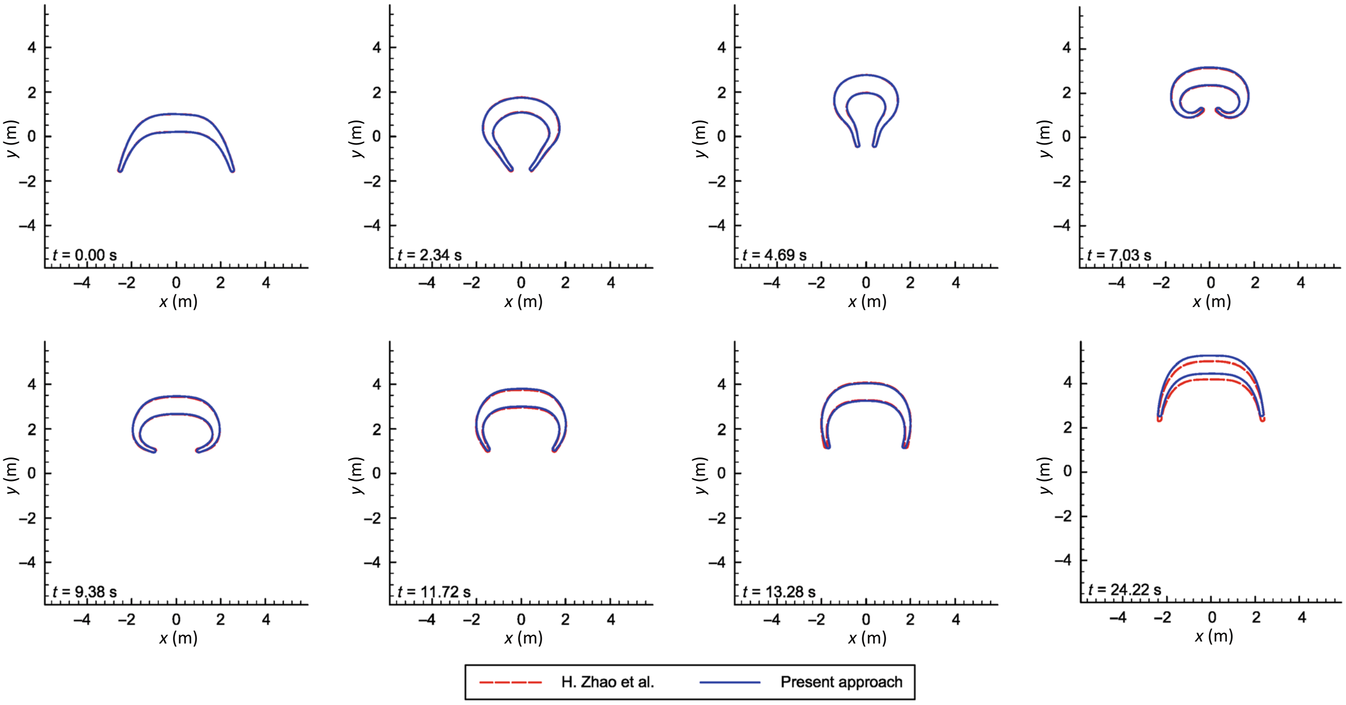


FIGURE 18 Swimming of two-dimensional jellyfish. Comparison of configurations at different times.

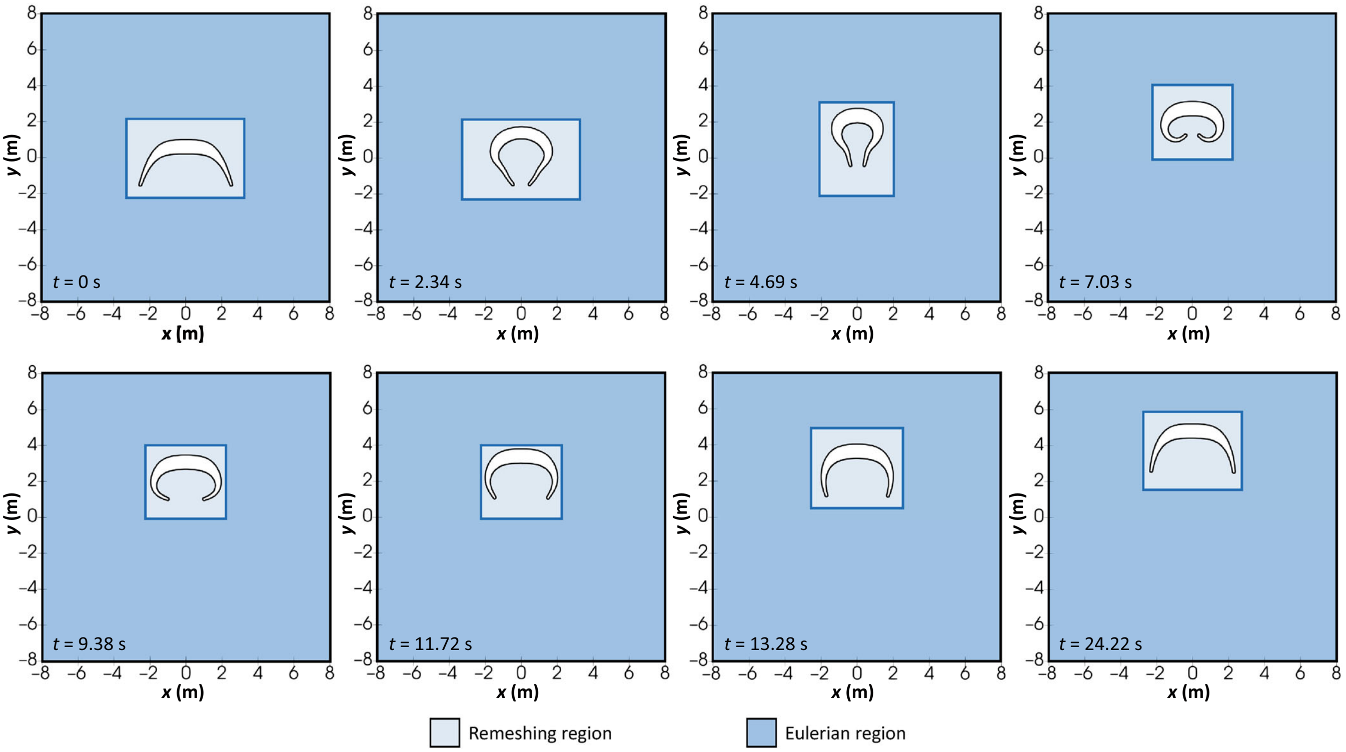


FIGURE 19 Swimming of two-dimensional jellyfish: moving and evolving remeshing region following jellyfish movement and deformation.

TABLE 4 Swimming of two-dimensional jellyfish.

	Lagrangian	Present approach
Computational time	12 h	9 h
% solution	73.48	94.03
% remeshing	26.22	5.68
% output	0.30	0.29
Remeshing time/number of time step	0.0070	0.0010
Total time/number of time step	0.0268	0.0183
	Simulated time: 24.5 s	

Note: Comparison of computational burden between the Lagrangian and proposed approach.

4.5 | Disk in a lid-driven cavity flow

In this application, a soft disk is immersed in a 1 m × 1 m squared fluid domain with an imposed unitary velocity at the top edge. Due to the imposed velocity at the upper boundary, the structure is pushed progressively toward the top by the fluid flow. The fluid initial density is set to $\rho_{f,0} = 1\ \mathrm{kg/m^3}$, whereas the dynamic viscosity is $\mu_f = 0.01$ Pa s. The disk, initially centered at $C(0.6\ \mathrm{m}, 0.5\ \mathrm{m})$, has a radius $r = 0.2$ m and it is modeled in Abaqus/Explicit as a hyperelastic Neo-Hookean solid with $c_{10} = 0.05$ Pa and $D_1 = 0$. The geometry of the problem is shown in Figure 20.

The computational domain is discretized with approximately 77k elements of average size 0.005 m. As shown in Figure 20, the fluid domain is divided into an Eulerian region and a moving Lagrangian remeshing one containing the solid during the motion, defined through enlarging factors $\beta_x^+ = 20$, $\beta_x^- = 20$, $\beta_y^+ = 20$, $\beta_y^- = 20$.

Results obtained with the present approach are compared in Figure 21 with the reference solutions in Reference 52, while the evolution of the remeshing region at different times can be seen in Figure 22. It can be observed that good agreements are achieved in the initial phase of the analysis. However, when the disk approaches the top edge, more evident differences are encountered. It should be pointed out that a viscoelastic incompressible model is used in Reference 52, whereas the viscosity of the solid is not considered in the present work[§]. Furthermore, slight differences in the numerical solutions can be expected, in view of the complexity of the FSI analysis, involving a highly deformable structure, and of the high velocity imposed in a relatively small domain. The comparison of the remeshing cost between the pure Lagrangian formulation with top Eulerian boundary and the present approach is reported in Table 5, confirming that significant computational savings are guaranteed by the proposed approach in terms of remeshing and total time per time step.

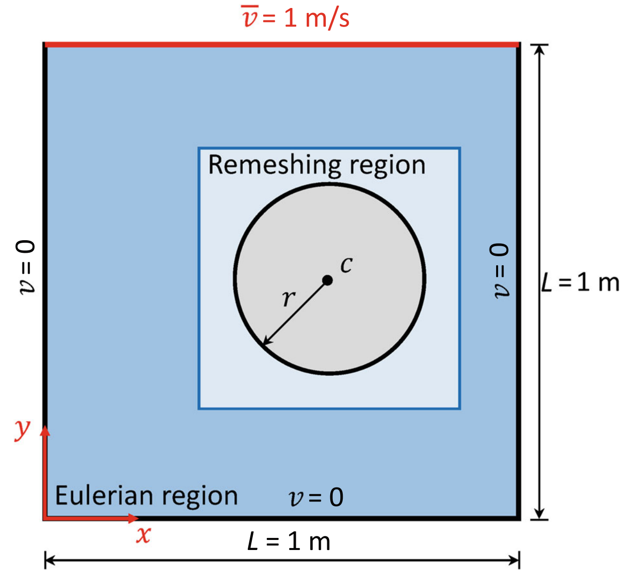


FIGURE 20 Disk in lid-driven cavity. Problem geometry.

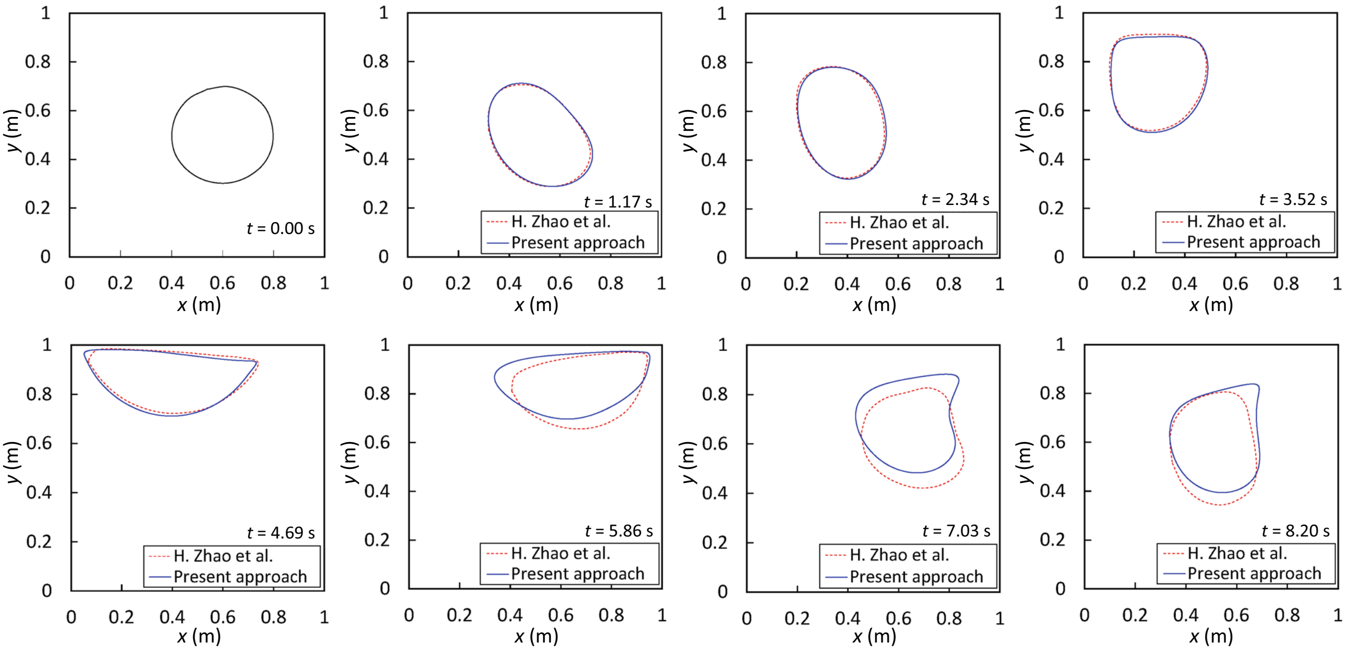


FIGURE 21 Disk in lid-driven cavity: comparison of configurations at different times.

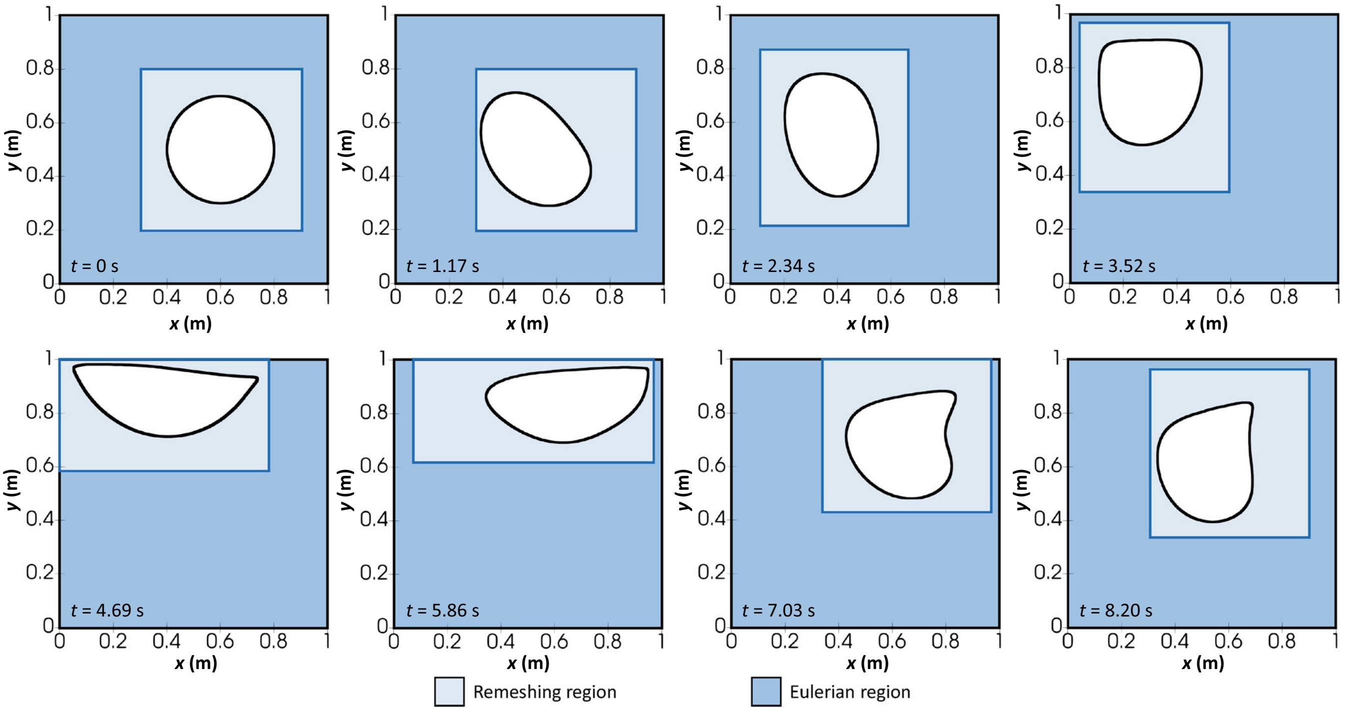


FIGURE 22 Disk in lid-driven cavity. Remeshing region and deforming disk positions at different times. The deformed disk is always entirely contained inside the remeshing region.

TABLE 5 Disk in lid-driven cavity.

	Lagrangian	Present approach
Computational time	9 h	6 h
% solution	72.60	96.58
% remeshing	27.26	3.28
% output	0.14	0.14
Remeshing time/number of time step	0.0015	0.0001
Total time/number of time step	0.0054	0.0041
	Simulated time: 8.25 s	

Note: Comparison of computational burden between fully Lagrangian and proposed approaches.

4.6 | Falling 3D solid in water

In this example, a 3D solid falling into a tank is considered. The tank of dimensions 150 mm × 140 mm × 140 mm is filled with water of density $\rho_f = 996.51$ kg/m^3 up to the height of 131 mm. The test consists in releasing a rigid solid having dimensions 20 mm × 20 mm × 21 mm and density $\rho_s = 1410$ kg/m^3 into water at rest.[54] At the beginning of the test, the body was partially immersed (see Figure 23) to reduce water splashes.

This problem has been analyzed with both the standard Lagrangian PFEM and the proposed method. In the latter, a Lagrangian region is defined in the portion of the domain close to the free surface and in the central part of the tank (Figure 24), in order to automatically detect the FSI interface. Fluid nodes are set to be Eulerian in the remaining region. The computational domain has been discretized with 104k nodes, defining the vertices of approximately 550k elements. Snapshots of the free surface evolution and the motion of the solid in the water are shown in Figure 24. The vertical displacement of the falling solid obtained by both approaches is compared in Figure 25 to both the numerical and experimental solutions provided in Reference 54.

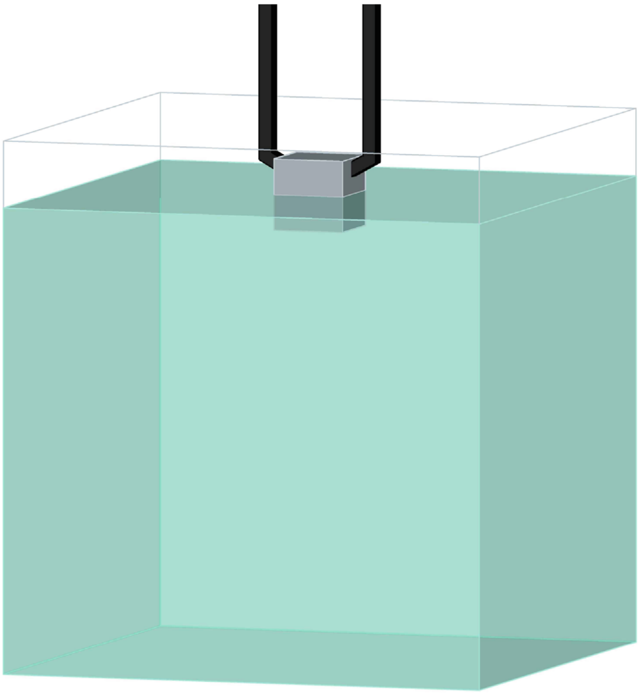


FIGURE 23 Falling cube in water tank. Experimental setup.

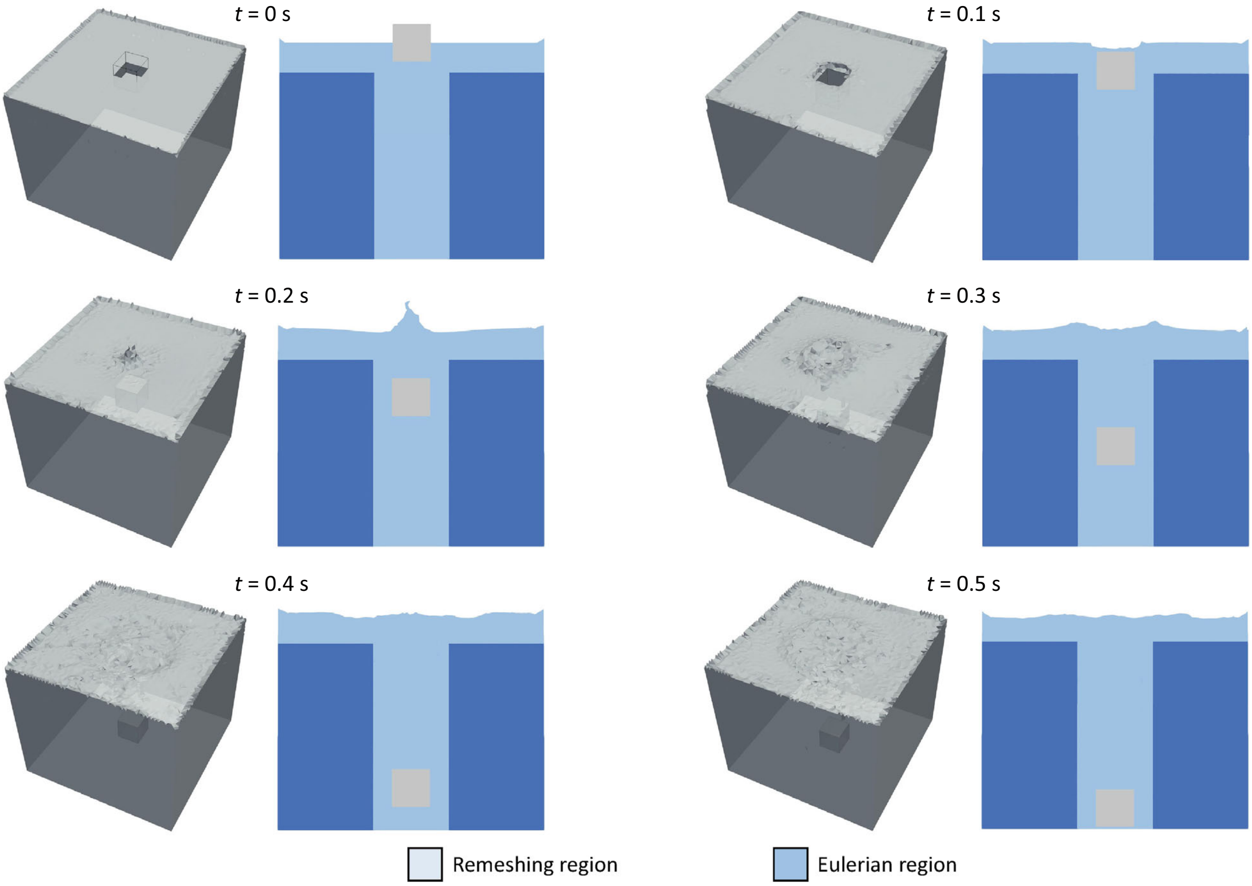


FIGURE 24 Falling cube in water tank. Evolution of free surface showing Lagrangian and Eulerian regions.

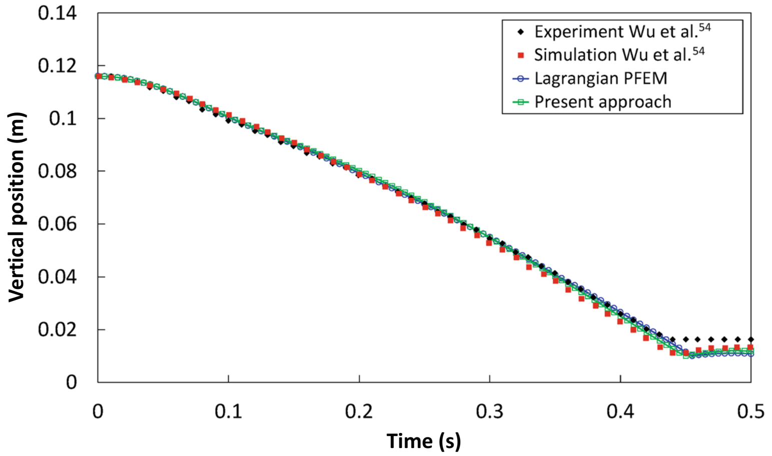


FIGURE 25 Falling cube in water tank. Comparison of vertical displacements of the cube obtained with different approaches.

TABLE 6 Falling cube in water tank.

	Lagrangian	Present approach
Computational time	28 h	22 h
% solution	79.05	95.64
% remeshing	19.98	3.41
% output	0.97	0.95
Remeshing time/number of time step	0.0138	0.0018
Total time/number of time step	0.0692	0.0535
	Simulated time: 0.5 s	

Note: Comparison of computational costs between the Lagrangian and proposed approaches.

The computational cost of remeshing is drastically reduced in the hybrid Lagrangian–Eulerian approach when compared to that of the standard Lagrangian PFEM (see Table 6). Furthermore, the total time over the number of time steps is lower in the proposed method, confirming the reduced computational cost. It is worth noting that in this example the bigger is the fluid tank, the larger would be the computational gain.

5 | CONCLUSIONS

In this work, a hybrid Lagrangian-Eulerian formulation for PFEM has been presented. This approach aims to exploit the advantages of the two kinematic descriptions for the treatment of fluid and FSI problems in explicit PFEM. In the hybrid approach proposed here, the Lagrangian description is adopted in regions where large changes in domain topology are expected, such as zones close to free surfaces and FSI interfaces. By contrast, in regions where no significant domain evolution is expected, or where complex boundary conditions are imposed, an Eulerian framework is preferred.

The proposed technique allows for the exploitation of the advantages of both Eulerian and Lagrangian formulations, using each approach where it is really suited. With respect to the standard PFEM, the time spent in the mesh regeneration is here reduced. The remeshing process is performed only in the small Lagrangian portion of the domain, since mesh distortion does not occur in the Eulerian region, where the computational mesh is kept fixed. With respect to the standard Eulerian approach, no additional equations are needed to describe the position of the free-surface or the interfaces between fluid and solid in FSI problems. Compared with the ALE approach, of which the proposed technique can be seen as representing a peculiar case, there is no need for a definition of the mesh motion.

When FSI problems are considered, a rectangular remeshing region enclosing the structure is introduced in the fluid domain. In order to properly contain the solid throughout the analysis, the position and space extension of the remeshing region are updated when the body approaches its borders. Inside this zone, nodes are treated as Lagrangian to facilitate the definition of the FSI interface, while the remaining nodes are kept Eulerian. Through this technique, the Lagrangian region is restricted to a small portion of the domain in correspondence of the structure. Consequently, only elements belonging to this region may undergo large distortions and should be remeshed.

To validate the proposed hybrid Lagrangian–Eulerian approach, several numerical examples have been analyzed. The comparison of the obtained results with those available in the literature shows good agreement, with a considerable reduction of the computational cost with respect to the purely Lagrangian PFEM.

This work represents a starting point for future possible developments, such as the adoption of different time steps in the Lagrangian and Eulerian regions. Furthermore, the possibility of solving the Eulerian region with different techniques (e.g., the finite difference method) will be also studied in order to further reduce the computational cost.

ACKNOWLEDGMENTS

This work has been partially supported by the Italian Ministry of University and Research (MUR) through the PRIN project XFAST-SIMS (No. 20173C478N). Cheng Fu gratefully acknowledges the support by Tetra Pak Packaging Solutions.

CONFLICT OF INTEREST STATEMENT

The authors declare no potential conflict of interest.

DATA AVAILABILITY STATEMENT

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

ENDNOTES

[*] The PFEM is characterized by the continuous application of the Delaunay triangulation, which changes the nodal connectivities. To avoid data interpolations from the previous distorted mesh to the current one, needed in the case of higher-order interpolations, with additional nodes placed along the element edges, linear shape functions are used in the PFEM.[4]

[†] No step index will be used to denote these matrices, assuming that they will remain constant over a certain number of steps

[‡] A viscoelastic model, exactly equivalent to what was proposed in Reference 52, is not available in Abaqus/Explicit. A constitutive model as close as possible to the one in Reference 52 has been used.

[§] See the comment to the previous example.

ORCID

Cheng Fu https://orcid.org/0000-0002-1326-6348
Massimiliano Cremonesi https://orcid.org/0000-0001-7885-029X
Umberto Perego https://orcid.org/0000-0003-1454-5629

REFERENCES

1. Donea J, Huerta A. *Finite Element Methods for Flow Problems*. John Wiley & Sons; 2003.
2. Osher S, Fedkiw R. Level set methods: an overview and some recent results. *J Comput Phys*. 2001;169(2):463-502.
3. Hirt CW, Nichols B. Volume of fluid (VOF) method for the dynamics of free boundaries. *J Comput Phys*. 1981;39(1):201-225.
4. Cremonesi M, Franci A, Idelsohn S, Oñate E. A state of the art review of the particle finite element method (PFEM). *Arch Comput Methods Eng*. 2020;27(5):1709-1735.
5. Oñate E, Idelsohn S, Del Pin F, Aubry R. The particle finite element method—an overview. *Int J Comput Methods*. 2004;1(2):267-307.
6. Idelsohn S, Oñate E, Del Pin F. The particle finite element method: a powerful tool to solve incompressible flows with free-surfaces and breaking waves. *Int J Numer Methods Eng*. 2004;61(7):964-989.
7. Idelsohn S, Oñate E, Del Pin F, Calvo N. Fluid–structure interaction using the particle finite element method. *Comput Methods Appl Mech Eng*. 2006;195(17-18):2100-2123.
8. Idelsohn S, Marti J, Limache A, Oñate E. Unified Lagrangian formulation for elastic solids and incompressible fluids: application to fluid–structure interaction problems via the PFEM. *Comput Methods Appl Mech Eng*. 2008;197(19):1762-1776. doi:10.1016/j.cma.2007.06.004
9. Cremonesi M, Frangi A, Perego U. A Lagrangian finite element approach for the analysis of fluid-structure interaction problems. *Int J Numer Methods Eng*. 2010;84(5):610-630.
10. Zhu M, Scott MH. Improved fractional step method for simulating fluid-structure interaction using the PFEM. *Int J Numer Methods Eng*. 2014;99(12):925-944. doi:10.1002/nme.4727
11. Meduri S, Cremonesi M, Perego U, Bettinotti O, Kurkchubasche A, Oancea V. A partitioned fully explicit Lagrangian finite element method for highly nonlinear fluid-structure interaction problems. *Int J Numer Methods Eng*. 2018;113(1):43-64.

12. Cerquaglia M, Thomas D, Boman R, Terrapon V, Ponthot JP. A fully partitioned Lagrangian framework for FSI problems characterized by free surfaces, large solid deformations and displacements, and strong added-mass effects. *Comput Methods Appl Mech Eng*. 2019;348:409-442. doi:10.1016/j.cma.2019.01.021
13. Cremonesi M, Ferri F, Perego U. A basal slip model for Lagrangian finite element simulations of 3D landslides. *Int J Numer Anal Methods Geomech*. 2017;41(1):30-53.
14. Zhang X, Krabbenhoft K, Pedroso D, et al. Particle finite element analysis of large deformation and granular flow problems. *Comput Geotech*. 2013;54:133-142.
15. Idelsohn S, Mier-Torrecilla M, Oñate E. Multi-fluid flows with the particle finite element method. *Comput Methods Appl Mech Eng*. 2009;198(33-36):2750-2767.
16. Becker P, Idelsohn S, Oñate E. A unified monolithic approach for multi-fluid flows and fluid–structure interaction using the particle finite element method with fixed mesh. *Comput Mech*. 2015;55:1091-1104. doi:10.1007/s00466-014-1107-0
17. Oñate E, Franci A, Carbonell J. A particle finite element method for analysis of industrial forming processes. *Comput Mech*. 2014;54(1):85-107.
18. Ryzhakov P. An axisymmetric PFEM formulation for bottle forming simulation. *Comput Mech*. 2017;4:3-12. doi:10.1007/s40571-016-0114-7
19. Aubry R, Idelsohn S, Oñate E. Particle finite element method in fluid-mechanics including thermal convection-diffusion. *Comput Struct*. 2005;83(17-18):1459-1475.
20. Oñate E, Franci A, Carbonell J. A particle finite element method (PFEM) for coupled thermal analysis of quasi and fully incompressible flows and fluid-structure interaction problems. In: Idelsohn, S. (eds) *Numerical Simulations of Coupled Problems in Engineering*. vol 33. Springer; 2014:129-156.
21. Donea J, Huerta A, Ponthot J-P, Rodriguez-Ferran A. Arbitrary Lagrangian-Eulerian methods. In: Stein E, Borst R, Hughes TJR, eds. *Encyclopedia of Computational Mechanics*. 2nd ed. Wiley; 2004. https://doi.org/10.1002/9781119176817.ecm2009
22. Harlow FH. The particle-in-cell method for numerical solution of problems in fluid dynamics. Technical Report. LADC-5288. Los Alamos National Lab.(LANL). 1962.
23. Więckowski Z. The material point method in large strain engineering problems. *Comput Methods Appl Mech Eng*. 2004;193(39):4417-4438. doi:10.1016/j.cma.2004.01.035
24. Idelsohn S, Nigro N, Limache A, Oñate E. Large time-step explicit integration method for solving problems with dominant convection. *Comput Methods Appl Mech Eng*. 2012;217-220:168-185. doi:10.1016/j.cma.2011.12.008
25. Ryzhakov P, Marti J, Dialami N. A unified arbitrary Lagrangian–Eulerian model for fluid–structure interaction problems involving flows in flexible channels. *J Sci Comput*. 2022;90(3):1-22.
26. Cremonesi M, Meduri S, Perego U. Lagrangian-Eulerian enforcement of non-homogeneous boundary conditions in the particle finite element method. *Comput Particle Mech*. 2020;7(1):41-56.
27. Masó M, Franci A, Pouplana DI, Cornejo A, Oñate E. A Lagrangian-Eulerian procedure for the coupled solution of the Navier-Stokes and shallow water equations for landslide-generated waves. *Adv Model Simul Eng Sci*. 2022;9(1):15. doi:10.1186/s40323-022-00225-9
28. Franci A, Oñate E, Carbonell J. Unified Lagrangian formulation for solid and fluid mechanics and FSI problems. *Comput Methods Appl Mech Eng*. 2016;298:520-547.
29. Oñate E, Cornejo A, Zárate F, Kashiyama K, Franci A. Combination of the finite element method and particle-based methods for predicting the failure of reinforced concrete structures under extreme water forces. *Eng Struct*. 2022;251:113510. doi:10.1016/j.engstruct.2021.113510
30. Meduri S, Cremonesi M, Perego U. An efficient runtime mesh smoothing technique for 3D explicit Lagrangian free-surface fluid flow simulations. *Int J Numer Methods Eng*. 2019;117(4):430-452.
31. Cheng S, Dey T, Shewchuk J, Sahni S. *Delaunay Mesh Generation*. CRC Press; 2013.
32. Edelsbrunner H, Mücke E. Three-dimensional alpha shapes. *ACM Trans Graph*. 1994;13(1):43-72.
33. Franci A, Cremonesi M. On the effect of standard PFEM remeshing on volume conservation in free-surface fluid flow problems. *Comput Particle Mech*. 2017;4(3):331-343.
34. Franci A. *Unified Lagrangian Formulation for Fluid and Solid Mechanics, Fluid-Structure Interaction and Coupled Thermal Problems Using the PFEM*. Springer Theses. Springer International Publishing; 2016.
35. Farhat C, Rallu A, Shankaran S. A higher-order generalized ghost fluid method for the poor for the three-dimensional two-phase flow computation of underwater implosions. *J Comput Phys*. 2008;227(16):7674-7700. doi:10.1016/j.jcp.2008.04.032
36. Bernard-Champmartin A, De Vuyst F. A low diffusive Lagrange-remap scheme for the simulation of violent air–water free-surface flows. *J Comput Phys*. 2014;274:19-49.
37. Cremonesi M, Meduri S, Perego U, Frangi A. An explicit Lagrangian finite element method for free-surface weakly compressible flows. *Comput Particle Mech*. 2017;4(3):357-369.
38. Fortin M, Brezzi F. *Mixed and Hybrid Finite Element Methods*. Vol 3. Springer-Verlag; 1991.
39. Bochev P, Dohrmann C, Gunzburger M. Stabilization of low-order mixed finite elements for the stokes equations. *SIAM J Num Anal*. 2006;44(1):82-101.
40. Dohrmann C, Bochev P. A stabilized finite element method for the stokes problem based on polynomial pressure projections. *Inte J Num Methods Fluids*. 2004;46(2):183-201.
41. Brooks AN, Hughes TJR. Streamline upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-stokes equations. *Comput Methods Appl Mech Eng*. 1982;32(1):199-259. doi:10.1016/0045-7825(82)90071-8
42. Courant R, Friedrichs K, Lewy H. On the partial difference equations of mathematical physics. *IBM J Res Dev*. 1967;11(2):215-234.

43. Combescure A, Gravouil A. Multi-time-step explicit–implicit method for non-linear structural dynamics. *Int J Numer Methods Eng*. 2001;50(1):199-225.
44. Li Z, Leduc J, Nunez-Ramirez J, Combescure A, Marongiu J. A non-intrusive partitioned approach to couple smoothed particle hydrodynamics and finite element methods for transient fluid-structure interaction problems with large interface motion. *Comput Mech*. 2015;55(4):697-718. doi:10.1007/s00466-015-1131-8
45. Nunez-Ramirez J, Marongiu J, Brun M, Combescure A. A partitioned approach for the coupling of SPH and FE methods for transient non-linear FSI problems with incompatible time-steps. *Int J Numer Methods Eng*. 2016;109:1391-1417. doi:10.1002/nme.5331
46. Combescure A, Gravouil A. A time–space multi-scale algorithm for transient structural nonlinear problems. *Mécan Ind*. 2001;2(1):43-55.
47. Faucher V, Combescure A. A time and space mortar method for coupling linear modal subdomains and non-linear subdomains in explicit structural dynamics. *Comput Methods Appl Mech Eng*. 2003;192(5-6):509-533.
48. Carretta Y, Boman R, Bech J, Legrand N, Laugier M, Ponthot JP. Numerical modelling of microscopic lubricant flow in sheet metal forming. Application to plane strip drawing. *Int J Numer Methods Eng*. 2017;112(3):203-237.
49. Ghia U, Ghia K, Shin C. High-Re solutions for incompressible flow using the Navier-stokes equations and a multigrid method. *J Comput Phys*. 1982;48(3):387-411.
50. Wu G, Taylor R. Finite element analysis of two-dimensional non-linear transient water waves. *Appl Ocean Res*. 1994;16(6):363-372.
51. Turek S, Hron J. Proposal for numerical benchmarking of fluid-structure interaction between an elastic object and laminar incompressible flow. In: Bungartz H, Schäfer M, eds. *Fluid-Structure Interaction. Lecture Notes in Computational Science and Engineering*. Vol 53. Springer; 2006:371-385.
52. Zhao H, Freund J, Moser R. A fixed-mesh method for incompressible flow–structure systems with finite solid deformations. *J Comput Phys*. 2008;227(6):3114-3140.
53. Dassault Systèmes. Abaqus documentation. 2021 Hyperelastic behavior of rubberlike materials.
54. Wu TR, Chu CR, Huang CJ, Wang CY, Chien SY, Chen MZ. A two-way coupled simulation of moving solids in free-surface flows. *Comput Fluids*. 2014;100:347-355. doi:10.1016/j.compfluid.2014.05.010

How to cite this article: Fu C, Cremonesi M, Perego U. A hybrid Lagrangian–Eulerian particle finite element method for free-surface and fluid–structure interaction problems. *Int J Numer Methods Eng*. 2024;125(5):e7402. doi: 10.1002/nme.7402

APPENDIX A. STABILIZATION

In the present work, the Direct Pressure Stabilization originally proposed in References 39 and 40 is used. According to this method, the stabilizing effect is achieved by adding a new term in the weak form of the mass conservation, based on the application of a local L^2 polynomial projector Π of the pressure field onto a lower-order interpolation space. Equation (14) is reformulated as:

$$\int_{\Omega_f^t} q_f \left[\frac{\partial p_f}{\partial t} + (\boldsymbol{c}_f \cdot \nabla_x) p_f\right] d\Omega_f + \int_{\Omega_f^t} K_f q_f (\nabla_x \cdot \boldsymbol{v}_f) d\Omega_f + \\ - \int_{\Omega_f^t} (q_f - \Pi q_f)(p_f - \Pi p_f) d\Omega_f = 0 \quad \forall q_f \in S^p. \tag{A1}$$

The projection operator Π is defined by the following condition:

$$\int_{\Omega_f^t} \Pi q_f (p_f - \Pi p_f) d\Omega_f = 0 \quad \forall q_f \in S^p. \tag{A2}$$

Introducing the last expression into Equation (A1), the added stabilizing term can be rewritten as

$$\int_{\Omega_f^t} (q_f - \Pi q_f)(p_f - \Pi p_f) d\Omega_f = \int_{\Omega_f^t} (q_f p_f - \Pi q_f \Pi p_f) d\Omega_f. \tag{A3}$$

When 2D three-node triangular elements are considered, the projection operator applied to the linear pressure field provides a constant value in each element:

$$\Pi p_f = \tilde{\mathbf{N}}^p \mathbf{P}_f, \tag{A4}$$

where $\tilde{\mathbf{N}}^p$ is defined as $\tilde{\mathbf{N}}^p = [1/3, 1/3, 1/3]$. Applying the pressure spatial discretization, the stabilizing term in Equation (A3) is computed for each element e of area A_e through the following expression:

$$\int_{\Omega^t_{f,e}} (\mathbf{N}^{p^T}\mathbf{N}^p - \tilde{\mathbf{N}}^{p^T}\tilde{\mathbf{N}}^p) d\Omega_f \mathbf{P}_f = \mathbf{S}_{f,e}\mathbf{P}_f = \frac{A_e}{36}\begin{bmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{bmatrix}\mathbf{P}_f. \tag{A5}$$