Engineering with Computers
https://doi.org/10.1007/s00366-024-01956-6

ORIGINAL ARTICLE

Cut-PFEM: a Particle Finite Element Method using unfitted boundary meshes

Rubén Zorrilla[1,2] · **Alessandro Franci**[1,2]

Received: 31 October 2023 / Accepted: 8 February 2024

Abstract
In this work, we present a novel unfitted mesh boundary strategy in the context of the Particle Finite Flement Method (PFEM) aiming to improve endemic limitations of the PFEM relative to boundary conditions treatment and mass conservation. In this new methodology, which we called Cut-PFEM, the fluid–wall interaction is not performed by adding interface elements, as is done in the standard PFEM boundaries. Instead, we use an implicit representation of (all or some of) the boundaries by introducing the use of a level set function. Such distance function detects the elements trespassing the (virtual) contours of the domain to equip them with opportunely boundary conditions, which are variationally enforced using Nitsche's method. The proposed Cut-PFEM circumvents important issues associated with the standard PFEM contact detection algorithm, such as the artificial addition of mass to the computational domain and the anticipation of contact time. Furthermore, the Cut-PFEM represents a natural ground for the imposition of alternative wall boundary conditions (*e.g.*, pure slip) which pose significant difficulties in a standard PFEM framework. Several numerical examples, featuring both no-slip and slip boundary conditions, are presented to prove the accuracy and robustness of the method in two-dimensional and three-dimensional scenarios.

Keywords PFEM · CutFEM · Cut-PFEM · Nitsche's method · Level set methods · Mass conservation · Slip conditions

1 Introduction

The Particle Finite Element Method (PFEM) is a Lagrangian mesh-based strategy suitable for large-deformation problems. The method uses the standard Finite Element Method (FEM) for the solution of the governing equations and a fast remeshing procedure to guarantee the good quality of the FEM mesh at each computation step. Whenever a limit distortion is reached, a new discretization of the computational domain is created by combining the Delaunay triangulation (DT) [1] with a boundary-recognition strategy, the Alpha-Shape (AS) method [2]. Since the very first works, the PFEM has shown its high potential in simulating complex free-surface fluid flows and fluid–structure interaction (FSI) phenomena [3–5]. The suitability of PFEM for large-deformation problems was also proven in the context of nonlinear solid mechanics analysis, as shown, for instance, in [6, 7].

The remeshing operations carried out in PFEM, together with its Lagrangian description, permit tracking accurately the deforming domain while avoiding elements distortion. However, the PFEM remeshing is also responsible for perturbing the equilibrium configuration reached at the previous computational step and may affect mass conservation [8]. In fact, remeshing operations can change the internal elemental connectivities and modify the topology of the computational domain. Remarkably, in standard PFEM, this latter mechanism occurs whenever the computational domain comes into contact with an external, rigid or deformable, body. This is because the new discretization is built over a cloud of points, that include not only the nodes from the previous mesh but also those used to discretize the external contours. As a consequence, the new discretization may include external contact elements that were not present in the previous mesh. This fact has important consequences for the PFEM numerical solution. On the one hand, these contact elements are beneficial for the solution algorithm, because they allow

✉ Alessandro Franci
falessandro@cimne.upc.edu

Rubén Zorrilla
rzorrilla@cimne.upc.edu

[1] Departament d'Enginyeria Civil i Ambiental, Universitat Politècnica de Catalunya, 08034 Barcelona, Spain

[2] International Center for Numerical Methods in Engineering (CIMNE), 08034 Barcelona, Spain

Published online: 12 April 2024

for modeling easily the interaction with the boundaries via a conforming-mesh approach. On the other hand, these elements are also responsible for introducing artificially new mass to the system and anticipating the actual contact time between the computational body and the external contours. In this respect, we remark that, in the standard PFEM, the entity of both drawbacks can be partially mitigated, though not completely eliminated, *e.g.*, by adjusting the mesh size and by using ad–hoc definition of the alpha parameter in the AS method [8]. Besides, it is important to note that these issues are more pronounced in unsteady fluid dynamics problems than in solid mechanics, where custom strategies can be more easily employed to mitigate them [9]. Due to this reason, in this work, we have decided to focus on computational fluid dynamics (CFD) problems.

Only very recently, the issues associated with the standard DT-AS remeshing procedure have been addressed in PFEM formulations for CFD analysis. For example, in [10], the authors proposed to use a level set method to minimize volume variation caused by PFEM remeshing. In [11], an automatic mesh adaptation algorithm was proposed to reduce the error in terms of mass conservation and contact detection.

The pure Lagrangian nature of PFEM and its body-fitted domain representation also have implications on the boundary conditions (BCs) modeling. In fact, if, on the one hand, such a Lagrangian conforming-mesh framework allows for an easy representation of no-slip conditions, both in static and dynamic regimes (*e.g.*, piston motion [12]), on the other hand, it is less natural for applying slip conditions. These cases require special treatment on the boundary nodes to avoid topological inconveniences, such as the artificial leakage of fluid. This explains why only a few applications of PFEM to slip conditions modeling can be found in the literature.

A first attempt at representing free-slip BCs was done in [13, 14]. In these works, slip conditions were modeled by letting the nodes of solid walls move only until a prefixed distance. Once this threshold separation was reached, the nodes were reallocated to their original positions. The referenced works showed the application of this approach only to simple 2D geometries. In [15], two different free-slip PFEM models were proposed. Both methods used the standard DT for boundary detection, but the contact elements were not treated as regular fluid elements, as is done in the standard PFEM. In the first approach, these contact elements were assigned a null viscosity to mimic slip conditions, whereas, in the second one, these elements were used to impose free-slip conditions by modifying the momentum equations accordingly. Both approaches were proven to be suitable for 2D free-surface fluid analysis, but they do not solve the described issues on contact detection. Very recently, in [16] a hybrid Lagrangian–Eulerian PFEM was proposed to enforce non-homogeneous BCs. The Eulerian description was used for boundary treatment, while a Lagrangian mesh was used for the rest of the domain. This PFEM strategy method was shown to succeed in applying slip and inlet conditions in problems with confined domains but was not tested for free-surface flows.

In a deeper analysis, all the presented drawbacks of PFEM analysis, *i.e.*, violation of mass conservation, anticipation of contact timing, and the unnatural representation of slip BCs, arise from the Lagrangian body-fitted nature of the method. For this reason, we decided to explore the use of an unfitted body approach in the context of the PFEM.

The most characteristic feature that distinguishes unfitted mesh methods from the traditional body-fitted approaches is the implicit representation of some (or all) of the boundaries of the computational domain. The implicit representation of the boundaries has been proven to be an extremely efficient technique for dealing with arbitrary large boundary displacements and rotations, and complex, potentially ill-conditioned, input geometries. This latter aspect turns into a competitive advantage, especially in large-scale real-life scenarios, such as the modeling of natural hazards, as it permits bypassing any manual geometry repairing operation.

The advantages of using an unfitted description of the boundaries come at the price of a more complex BCs imposition. This can be approached in several manners, thus resulting in different methods among which we highlight the Immersed Boundary Method (IBM) [17–20], the eXtended-Finite Element Method (X-FEM) [21, 22], the Shifted Boundary Method (SBM) [23–26] and the Cut-Finite Element Method (CutFEM), also known as Embedded Boundary Method (EBM) [27]. In this regard, the CutFEM will be our method of choice owing to its proven capability to keep the accuracy and convergence rates of the formulation without the need to introduce neither blending elements nor extra degrees of freedom (DOFs).

The CutFEM is based on identifying the elements intersected by unfitted boundaries. Accordingly, the elements of the mesh can be classified into two types: those entirely filled with fluid, requiring no special treatment, and those partially filled (*i.e.*, intersected by unfitted boundaries). In short, the CutFEM works by enhancing such partially filled (or intersected) elements with a subintegration procedure and a variational (weak) imposition of the BCs. The purpose of the subintegration procedure is to reallocate the elements' integration points in accordance to the unfitted boundaries, making possible to consider only the portion of the computational mesh that is occupied by the fluid. To what concerns the BCs imposition, we shall note that in the CutFEM case, the use of standard (*i.e.*, strong) techniques is not sufficient as some of (or all) the boundaries of the problem may lie inside the elements owing to the unfitted nature of the mesh. In this context, variational BCs imposition methods are required to keep the enforcement of the problem constraints

on the real boundaries, something that results in optimal convergence and accuracy properties.

Hence, our idea is to combine the conventional PFEM with the CutFEM approach, giving rise to the so-called Cut-PFEM. In this new approach, the interaction of the fluid with the boundaries can be modeled in an unfitted mesh fashion, *i.e.*, the BCs are imposed variationally over the current mesh and not geometrically by creating contact elements. The boundary terms are imposed after the evaluation of a distance function that becomes negative whenever a fluid node reaches the (virtual) rigid contours.

This approach allows us not only to avoid using contact elements and all the associated drawbacks (increase of volume and anticipation of contact time) but also to easily model slip conditions. All these advantages of the Cut-PFEM versus the standard PFEM will be discussed and proved extensively in this work.

We also remark that the Cut-PFEM approach is compatible with the other methods proposed in the literature to improve PFEM remeshing [10, 11]. Furthermore, as will be shown in this work, it can be used together with the standard PFEM body-fitted approach which can be still convenient for the imposition of some BCs, such as wave-maker devices.

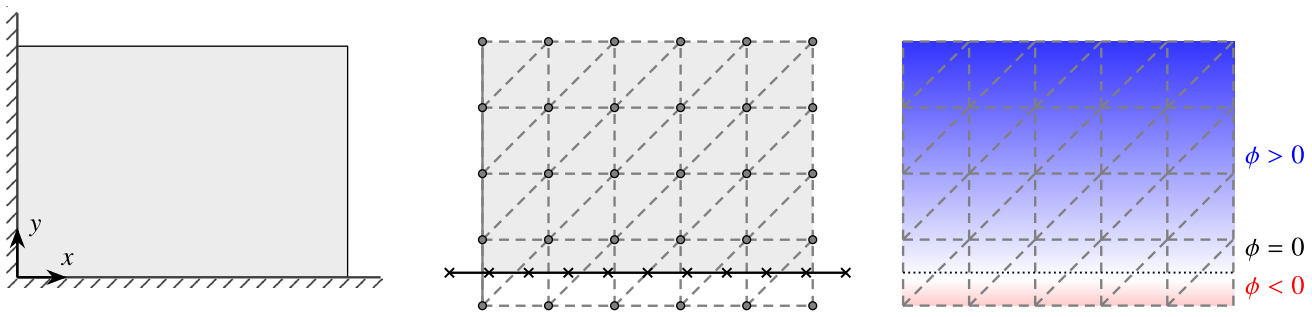
The remainder of the paper is structured as follows. In Sect. 2, we introduce in a simple and schematic way the concept and representation of unfitted boundaries. In Sect. 3, we present the governing equations of the problem. In Sect. 4, we introduce the variational form making special emphasis on the new boundary terms. In Sect. 5, we give the fully discretized and linearized form of the governing equations. In Sect. 6, we describe the scheme of subintegration used for the intersected elements. In Sect. 7, we recall the main concepts and basic algorithm of PFEM remeshing procedure and we highlight the particularities of this scheme when it is employed in an unfitted domain representation. In Sect. 8, we summarize the Cut-PFEM solution strategy, also providing pseudocoded algorithms. In Sect. 9, we validate the proposed Cut-PFEM approach through the solution of three benchmark problems. Key aspects of the method, such as mass preservation skills, slip conditions modeling, and robustness of the algorithm, are deeply discussed. Finally, in Sect. 10, we provide some concluding remarks and we discuss potential future developments in the framework of Cut-PFEM.

2 Unfitted boundaries' representation

The representation of unfitted boundaries is one of the core features of the proposed Cut-PFEM methodology. In this section, we describe the basics for the implicit representation of unfitted boundaries as well as how this approach can be applied within a PFEM framework.

First, it is important to note that the implicit representation of a surface boundary is commonly achieved by the use of a discrete level set function [28]. This function is built by calculating a signed distance from each node of the computational mesh to the unfitted boundaries, thus meaning that the zero-distance isosurface becomes the implicit representation of the boundary in question. Hence, points that lie in the interior/exterior of the computational domain feature a positive/negative distance value. A direct consequence of this approach is that the elements that are intersected by any unfitted boundary will necessarily have both positive and negative distance values at their nodes.

To clarify these concepts, let us consider the simple 2D geometry plotted in Fig. 1, assimilable to a dam-break scenario, as an example for the discussion at hand. Our objective is to solve this problem using the unfitted representation of some of its boundaries. For the sake of generality,



(a) Problem description. (b) Computational domain. (c) Distance field.

Fig. 1 Graphical representation of an unfitted 2D mesh. **a** Description of the problem, being the grey solid pattern the fluid mass and the dashed one the vertical and horizontal walls. **b** Hypothetical Cut-FEM discretization of the problem. The grey dashed lines and round markers represent the computational background mesh. The solid grey vertical line represents a standard body-fitted boundary. The black lines and cross markers depict the unfitted boundary auxiliary mesh from which the level set is to be computed. **c** Distance field that implicitly represents the unfitted horizontal wall. The thin dotted line highlights the zero isosurface of the level set

we model the bottom wall as an unfitted boundary while we treat the vertical one as a standard body-fitted PFEM wall. Nonetheless, we highlight that there is no technical limitation in this regard, so all the boundaries could be perfectly solved using an unfitted representation. Figure 1 shows a possible discretization for the problem according to this new representation. As it can be observed, the discrete version of the problem involves two separate meshes: a volume mesh (grey lines in Fig. 1), that represents the fluid domain, and an auxiliary surface mesh (black lines in Fig. 1), which serves only to describe the unfitted boundaries. The volume mesh, commonly referred to as the background mesh, is where the governing equations are solved. The boundaries of this mesh align with those of the computational domain for the standard body-conforming boundaries (left vertical wall) but do not so for the unfitted ones (bottom horizontal wall). To what concerns the auxiliary surface mesh, sometimes denoted as the skin mesh, it is important to clearly state that it has neither computational nor topological requirements, meaning that any triangulation (*e.g.*, *stl* meshes featuring entity duplications, gaps, and overlaps) can be perfectly used.

By focusing on the unfitted contour, one can observe that the elements at the bottom of the volume are cut by the auxiliary mesh discretizing the horizontal wall. Hence, the idea is to implicitly describe such intersections in terms of the background mesh using a level set (distance) function. In this simple example, in which the unfitted boundary can be analytically described by the equation $y = 0$, the distance field ϕ can be analytically computed at each i-node of the background mesh as $\phi_i = y_i$. This expression results in the contour field plotted in Fig. 1. Considering that blue and red colors denote the positive and negative distance nodes, respectively, the white color band necessarily contains the zero isosurface of the distance (highlighted by a dotted line in Fig. 1) that implicitly represents the unfitted horizontal wall.

In very simple scenarios, like the one previously described, the distance function can be easily described with an analytical function. However, this is unlikely the case when dealing with real-life problems as these potentially entail more complex geometrical patterns that require the use of a level set calculation algorithm. Although this sort of algorithm is more related to the computer graphics community, in the following, we briefly describe the steps of a feasible implementation for our target application. First, it is needed to select the elements of the background mesh that are candidate to be intersected by the skin ones. This procedure can be efficiently achieved by building an octree-based search structure on top of the skin auxiliary mesh. Once it is found that a background mesh element is intersected, the intersection plane can be built from the intersections of the element's edges with the boundary entities. Then, the level set values are computed as the minimum distance from each node to the reconstructed intersection plane.

Last but important, we shall remark that in a PFEM simulation, the mesh nodes move according to the Lagrangian solution of the governing equations. Consequently, the distance function requires recurrent computation. In the case of implicit schemes applied to large-deformation problems, such as the ones analyzed in this work, the distance function must be updated at each time step.

3 Governing equations

The fluid dynamics problem is governed by the balance of linear momentum and mass-conservation equations. As it is customary in the PFEM, the problem is solved in the Updated Lagrangian framework [3]. After denoting the computational domain as Ω and the total time as T, the problem to be solved can be expressed as

$$\rho\frac{\partial \mathbf{v}}{\partial t} - \nabla\cdot\boldsymbol{\sigma} - \rho\mathbf{g} = \mathbf{0} \quad \text{in } \Omega\times(0,T]\,, \tag{1a}$$

$$-\frac{1}{\kappa}\frac{\partial p}{\partial t} + \nabla\cdot\mathbf{v} = 0 \quad \text{in } \Omega\times(0,T]. \tag{1b}$$

Here, ρ and κ are the fluid density and bulk modulus, $\mathbf{v}$ and p are the velocity and pressure fields, $\mathbf{g}$ is the volume acceleration (*i.e.*, gravity), $\nabla\cdot$ is the divergence operator in current configuration and $\boldsymbol{\sigma}$ is the Cauchy stress tensor, which is defined as

$$\boldsymbol{\sigma} = 2\mu\mathbf{d}'(\mathbf{v}) - p\mathbf{I}\,. \tag{2}$$

μ is the effective (*i.e.*, apparent) viscosity of the fluid and $\mathbf{d}'$ is the deviatoric part of the deformation rate tensor $\mathbf{d}$, which is computed from the velocity field as

$$\mathbf{d}'(\mathbf{v}) = \frac{1}{2}\left(\nabla\mathbf{v} + \nabla\mathbf{v}^T\right) - \frac{1}{3}(\nabla\cdot\mathbf{v})\mathbf{I}\,, \tag{3}$$

being ∇ the gradient operator in the current configuration. We note that Eq. 1b is solved including a small compressibility to allow the explicit computation of pressure [29, 30]. Such compressibility is controlled by κ, implying that the standard incompressible divergence-free form is recovered as $\kappa\to\infty$.

The governing equations are completed by the corresponding BCs, which are applied on the computational domain boundary $\partial\Omega = \Gamma$. More specifically, Γ is defined as the union of the two disjoint sets Γ_D (the Dirichlet boundary) and Γ_N (the Neumann boundary), such that $\Gamma_D\cup\Gamma_N = \Gamma$ and $\Gamma_D\cap\Gamma_N = \emptyset$. Hence, the BCs for the Eq. 1 are

$$\mathcal{P}(\mathbf{v} - \bar{\mathbf{v}}) = \mathbf{0} \quad \text{on } \Gamma_D\times(0,T]\,, \tag{4a}$$

$$\boldsymbol{\sigma}\cdot\mathbf{n} - \bar{\mathbf{t}} = \mathbf{0} \quad \text{on } \Gamma_N\times(0,T]\,, \tag{4b}$$

being $\bar{\mathbf{v}}$ and $\bar{\mathbf{t}}$ the prescribed wall velocity and traction to be imposed on Γ_D and Γ_N, respectively. As usual, $\mathbf{n}$ denotes the outward unit normal vector of the boundary. $\mathcal{P}$ is a matrix projection operator that allows us to conveniently define the two different Dirichlet BCs that we are considering in this work. The former is the standard pure stick (also known as no-slip) condition and can be straightforwardly obtained using the identity matrix as projection operator, that is $\mathcal{P} = \mathbf{I}$. The latter is the pure slip (*i.e.*, no penetration) boundary condition, which only constrains the velocity in the direction perpendicular to the wall. This is achieved by setting the projection operator to the normal projection matrix, such that $\mathcal{P} = \mathbf{P}_n = \mathbf{n} \otimes \mathbf{n}$.

4 Variational form

Let us first define the notation

$$(a, b)_\Omega = \int_\Omega ab \quad \text{and} \quad (\mathbf{a}, \mathbf{b})_\Omega = \int_\Omega \mathbf{ab}$$

for the scalar and vector $L^2(\Omega)$-inner products on the interior of Ω and

$$\langle a, b\rangle_\Gamma = \int_\Gamma ab \quad \text{and} \quad \langle \mathbf{a}, \mathbf{b}\rangle_\Gamma = \int_\Gamma \mathbf{ab}$$

for the boundary ones on Γ. Complementary, we also extend the previously described notation for the boundaries to distinguish the Dirichlet body-fitted boundaries, henceforth denoted as $\Gamma_{D'}$, from the unfitted ones, which we denote as Γ_ϕ. Hence, from now on, we will consider that $\Gamma_{D'} \cup \Gamma_\phi = \Gamma_D$ and $\Gamma_{D'} \cap \Gamma_\phi = \emptyset$. We also introduce the functional spaces $\boldsymbol{V} := \boldsymbol{H}^1_0(\Omega)$ (*i.e.*, the space of functions in $\boldsymbol{H}^1(\Omega)$ vanishing on $\Gamma_{D'}$) and $Q := L^2(\Omega)/\mathbb{R}$ (*i.e.*, the space of square-integrable functions in $\mathbb{R}$) for the velocity and pressure approximations.

Before getting into the details of the variational form, it is important to recall that the conventional BCs imposition technique employed in the standard body-fitted PFEM is no longer suitable for the unfitted boundaries. This is due to the fact that there are no DOFs located over Γ_ϕ, something that precludes the use of a strong Dirichlet imposition at the algebraic level. Hence, the conditions over Γ_ϕ necessarily need to be enforced by using variational (weak) imposition techniques. Among the different approaches to weakly impose the Dirichlet constraints, in this work, we opt for Nitsche's method [31]. A complete discussion, including consistency and stability analyses as well as error estimates, of the application of Nitche's method to the imposition of general BCs in the Poisson model problem can be found in [32]. We also refer the reader to [33, 34] for a detailed analysis of the stability and convergence properties of Nitsche's method applied to Stokes and Navier–Stokes (Oseen) problems. A similar discussion but for the slip and Navier-slip BCs can be found in [35, 36]. Complementary, the modified Nitsche approaches presented in [37, 38] also deserve a mention.

The previously defined functional spaces allow us to define the variational form as follows. Find $\mathbf{u} \in \boldsymbol{V}$ and $p \in Q$, such that

$$\begin{aligned}
&\left(\mathbf{w}, \rho\frac{\partial \mathbf{v}}{\partial t}\right)_\Omega + \left(\nabla\mathbf{w}, 2\mu\mathbf{d}'\right)_\Omega - (\nabla\cdot\mathbf{w}, p)_\Omega - (\mathbf{w}, \rho\mathbf{g})_\Omega \\
&- \left(q, \frac{1}{\kappa}\frac{\partial p}{\partial t}\right)_\Omega + (q, \nabla\cdot\mathbf{v})_\Omega - \langle \mathbf{w}, \bar{\mathbf{t}}\rangle_{\Gamma_N} \\
&- \langle \mathbf{w}, (2\mu\mathbf{d}'(\mathbf{v}) - p\mathbf{I})\cdot\mathbf{n}\rangle_{\Gamma_\phi} + \beta\langle \mathbf{w}, \mathcal{P}(\mathbf{v} - \bar{\mathbf{v}})\rangle_{\Gamma_\phi} \\
&- \langle \left(2\mu\mathbf{d}'(\mathbf{w}) - q\mathbf{I}\right)\cdot\mathbf{n}, \mathcal{P}(\mathbf{v} - \bar{\mathbf{v}})\rangle_{\Gamma_\phi} = 0
\end{aligned} \tag{5}$$

for all $\mathbf{w} \in \boldsymbol{V}$ and $q \in Q$.

At this point, it is important to stress that in addition to the standard Navier–Stokes volumetric terms, the variational problem that we solve includes a set of boundary terms that are crucial for the Cut-PFEM. Taking this into account and with the aim of simplifying the forthcoming discussion, let us start by identifying each of the contributions in Eq. 5. Hence, we define the boundary fluxes

$$\begin{aligned}
f_{\Gamma_N} &= -\langle \mathbf{w}, \bar{\mathbf{t}}\rangle_{\Gamma_N}\,, \\
f_{\Gamma_\phi} &= -\langle \mathbf{w}, (2\mu\mathbf{d}'(\mathbf{v}) - p\mathbf{I})\cdot\mathbf{n}\rangle_{\Gamma_\phi}\,, \\
f^{BC}_{\Gamma_\phi} &= \beta\langle \mathbf{w}, \mathcal{P}(\mathbf{v} - \bar{\mathbf{v}})\rangle_{\Gamma_\phi} - \langle \left(2\mu\mathbf{d}'(\mathbf{w}) - q\mathbf{I}\right)\cdot\mathbf{n}, \mathcal{P}(\mathbf{v} - \bar{\mathbf{v}})\rangle_{\Gamma_\phi}\,,
\end{aligned}$$

as well as the volume contribution

$$\begin{aligned}
f_\Omega = &\left(\mathbf{w}, \rho\frac{\partial \mathbf{v}}{\partial t}\right)_\Omega + \left(\nabla\mathbf{w}, 2\mu\mathbf{d}'\right)_\Omega - (\nabla\cdot\mathbf{w}, p)_\Omega \\
&- (\mathbf{w}, \rho\mathbf{g})_\Omega - \left(q, \frac{1}{\kappa}\frac{\partial p}{\partial t}\right)_\Omega + (q, \nabla\cdot\mathbf{v})_\Omega\,,
\end{aligned}$$

which altogether allow us to write the variational form above (Eq. 5) in a more compact form as

$$f_\Omega + f_{\Gamma_N} + f_{\Gamma_\phi} + f^{BC}_{\Gamma_\phi} = 0\,.$$

The f_{Γ_N} and f_{Γ_ϕ} boundary contributions come from the integration by parts of the stress term, while $f^{BC}_{\Gamma_\phi}$ collects the Nitsche terms required for the weak imposition of Dirichlet BCs on Γ_ϕ. In the following subsections, we discuss the nature of these boundary terms as well as their role in the Cut-PFEM.

4.1 Boundary traction

The boundary traction term coming from the integration by parts of the stress divergence is customarily employed

for the imposition of Neumann BCs on Γ_N. As it can be observed in Eq. 5, this can be equivalently done in the Cut-PFEM method, either by setting an external traction $\bar{\mathbf{t}}$ on f_{Γ_N} or by simply neglecting such contribution (*i.e.*, $\bar{\mathbf{t}} = \mathbf{0}$) to effectively enforce a traction-free behavior.

Furthermore, it is important to bear in mind that the interpolation of $\mathbf{w}$ is not zero on Γ_ϕ or, in other words, that $\mathbf{w}$ vanishes on Γ_D but is no longer zero on Γ_ϕ (*i.e.*, on the interface cuts). Therefore, it is required to add the boundary traction contribution f_{Γ_ϕ} on Γ_ϕ. We highlight this as a difference with respect to the conventional PFEM, in which the boundary traction is omitted in all the Dirichlet boundaries.

4.2 Weak Dirichlet BCs imposition

As it can be observed in Eq. 5, the Nitsche-based Dirichlet BCs imposition $f^{BC}_{\Gamma_\phi}$ is composed of two terms. The first one reading $\beta\langle \mathbf{w}, \mathcal{P}(\mathbf{v}-\bar{\mathbf{v}})\rangle_{\Gamma_\phi}$ is nothing but a penalty term scaled by the penalization parameter β, which we define as

$$\beta = \gamma\left(\frac{\mu}{h} + \rho\|\mathbf{v}\| + \frac{\rho h}{\Delta t}\right), \tag{7}$$

being h a characteristic element size and Δt the time increment. We note that β needs to be dimensionally consistent with the formulation to be solved, in our case the Navier–Stokes equations. Besides this, it is designed to scale according to different flow magnitudes to be suitable for a wide range of flow scenarios and thus reduce the formulation dependency on the user-defined constant γ.

The second Nitsche contribution in $f^{BC}_{\Gamma_\phi}$, namely $\langle \left(2\mu\mathbf{d}'(\mathbf{w}) - q\mathbf{I}\right)\cdot\mathbf{n}, \mathcal{P}(\mathbf{v}-\bar{\mathbf{v}})\rangle_{\Gamma_\phi}$, is commonly referred to as the symmetric counterpart of the boundary flux (traction). Its role is to enhance the stability of the formulation and to reduce the dependency on γ.

Remark 1 In this work, we have assumed that only Dirichlet BCs are imposed on the unfitted boundaries Γ_ϕ. However, we shall remark that there is no technical limitation in this regard. Hence, the presented methodology could perfectly be extended to more complex wall models, such as Robin-type BCs (*e.g.*, Navier-slip [36]).

5 Discrete form and solution strategy

The discrete domain is obtained from the partition of Ω into a set of non-overlapping elements. It is important to mention that only simplicial elements with linear $\mathbf{v}$ and p approximations are used in this work (linear triangles in 2D and linear tetrahedra in 3D). It is known that this interpolation does not satisfy the well-known inf-sup or LBB condition and thus requires the use of stabilization techniques. Among the many successful approaches that can be found in the literature, such as the streamline upwind Petrov–Galerkin (SUPG) [39], the Galerkin/least-squares (GLS) [40] or the variational multiscales (VMS) family of techniques, featuring the algebraic subgrid scales (ASGS) [41–43] and orthogongal subscales (OSS) [44–46], in this work, we chose the finite increment calculus (FIC) [47–50]. In this regard, we note that owing to the segregated resolution strategy that we use, we only require the continuity equation to be stabilized [47]. As these extra terms are not of special interest for the focus of this work, and considering that any alternative to the FIC can be equivalently used, we decided to omit the stabilization terms from the discussion at hand. Nevertheless, we refer the reader to [51] for a detailed description of the derivation and implementation of the FIC stabilization used in this work.

Following the approach presented in [47], we use an iterative two-step segregated strategy for the resolution of the governing equations in Eq. 1. Hence, at each iteration k of the strategy, a first linearized momentum equation is solved to obtain the velocity increment $\Delta\tilde{\mathbf{v}}^{k+1}$. Then, this correction is applied to the nodal velocities and the FIC-stabilized continuity equation is solved to obtain $\tilde{p}^{k+1}$ and update the material response for the next iteration. Assuming a generic time discretization with a time step increment Δt together with the previously described finite element discretization, results in solving the algebraic problem

$$\left(\mathbf{K} + \mathbf{K}^\phi\right)\Delta\tilde{\mathbf{v}}^{k+1} = \mathbf{r} + \mathbf{r}^\phi \tag{8}$$

with the left-hand side (LHS) ij-nodal contributions being computed at each iteration k as

$$\begin{aligned}\mathbf{K}_{ij} &= \int_\Omega \frac{2\rho}{\Delta t}\mathbf{N}_i\mathbf{N}_j d\Omega + \int_\Omega \mathbf{B}_i^T\mathbf{C}\mathbf{B}_j d\Omega \quad \text{and} \\ \mathbf{K}^\phi_{ij} &= \int_{\Gamma_\phi}\left(\beta\mathbf{N}_i - \mathbf{B}_i^T\mathbf{C}\mathbf{P}^v_n\right)\mathbf{N}_j\mathbf{P}_\phi d\Gamma\,,\end{aligned} \tag{9}$$

as well as the residual right-hand side (RHS) i-node contributions

$$\mathbf{r}_i = \int_\Omega \rho\mathbf{N}_i\mathbf{N}_j\dot{\mathbf{v}}_j d\Omega + \int_\Omega \mathbf{B}_i^T\boldsymbol{\sigma} d\Omega - \int_\Omega \rho\mathbf{N}_i\mathbf{N}_j\mathbf{g}_j d\Omega - \int_{\Gamma_N}\mathbf{N}_i\mathbf{N}_j\bar{\mathbf{t}}_j d\Gamma \tag{10}$$

and

$$\mathbf{r}^\phi_i = -\int_{\Gamma_\phi}\mathbf{N}_i\mathbf{P}^v_n\boldsymbol{\sigma} d\Gamma - \int_{\Gamma_\phi}\left(\beta\mathbf{N}_i - \mathbf{B}_i^T\mathbf{C}\mathbf{P}^v_n\right)\mathbf{N}_j\mathbf{P}_\phi\left(\mathbf{v}_j - \bar{\mathbf{v}}_j\right)d\Gamma\,, \tag{11}$$

in which $\dot{\mathbf{v}}_j$ and $\boldsymbol{\sigma}$ are the j-node acceleration vector and the Voigt representation of the stress tensor, respectively, computed at the previous iteration k. As usual, $\mathbf{N}_i$ and $\mathbf{B}_i$ denote

the i-node shape function and strain matrices, which for the simplified 2D case are

$$\mathbf{N}_i = \begin{bmatrix} N_i & 0 \\ 0 & N_i \end{bmatrix} \quad \text{and} \quad \mathbf{B}_i = \begin{bmatrix} \frac{\partial N_i}{\partial x} & 0 \\ 0 & \frac{\partial N_i}{\partial y} \\ \frac{\partial N_i}{\partial y} & \frac{\partial N_i}{\partial x} \end{bmatrix} .$$

The tensor $\mathbf{C}$ is the constitutive matrix in Voigt notation, computed as

$$\mathbf{C} = \begin{bmatrix} \kappa\Delta t + \frac{4\mu}{3} & \kappa\Delta t - \frac{2\mu}{3} & 0 \\ \kappa\Delta t - \frac{2\mu}{3} & \kappa\Delta t + \frac{4\mu}{3} & 0 \\ 0 & 0 & \mu \end{bmatrix}$$

in the simplified 2D case. $\mathbf{P}_n^v$ is an auxiliary matrix to do the normal projection of $\boldsymbol{\sigma}$ in Voigt notation. For the 2D case, the components of $\mathbf{P}_n^v$ are

$$\mathbf{P}_n^v = \begin{bmatrix} n_x & 0 & n_y \\ 0 & n_y & n_x \end{bmatrix} .$$

Similarly, $\mathbf{P}_\phi$ denotes the discrete version of $\mathcal{P}$. Hence, for the 2D pure stick case

$$\mathbf{P}_\phi = \mathbf{I} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix},$$

while for the pure slip one

$$\mathbf{P}_\phi = \mathbf{n} \otimes \mathbf{n} = \begin{bmatrix} n_x n_x & n_x n_y \\ n_y n_x & n_y n_y \end{bmatrix} .$$

Once Eq. 8 is solved, the kinematics are updated according to the $\Delta\tilde{\mathbf{v}}^{k+1}$. Then, the current iteration pressure is obtained by solving the FIC-stabilized discrete continuity equation

$$\left(\mathbf{M} + \mathbf{S}^\tau\right)\tilde{\mathbf{p}}^{k+1} = \mathbf{M}^n\tilde{\mathbf{p}}^n - \mathbf{Q}^T\tilde{\mathbf{v}}^{k+1} + \mathbf{f}^{\tau,k+1} , \tag{12}$$

with

$$\mathbf{M}_{ij} = \int_\Omega \frac{1}{\kappa\Delta t} N_i N_j d\Omega \quad \text{and} \quad \mathbf{Q}_{ij} = \int_\Omega \mathbf{B}_i^T \mathbf{m} \mathbf{N}_j d\Omega . \tag{13}$$

The matrix $\mathbf{M}^n$ is the equivalent to $\mathbf{M}_{ij}$ but considering the previous step time increment Δt^n. Similarly, $\mathbf{p}^n$ corresponds to the previous step pressure solution. Vector $\mathbf{m}$ is an auxiliary array for the calculation of the divergence in Voigt notation, and $\mathbf{S}^\tau$ and $\mathbf{f}^{\tau,k+1}$ are stabilizing terms arising from the FIC methodology described in [51].

Last but important, the solution of iteration $k+1$ must be completed by updating the material response, that is to say $\boldsymbol{\sigma}$, $\mathbf{C}$ and μ (in the non-Newtonian case), with the recently obtained values $\tilde{\mathbf{v}}^{k+1}$ and $\tilde{p}_{k+1}$.

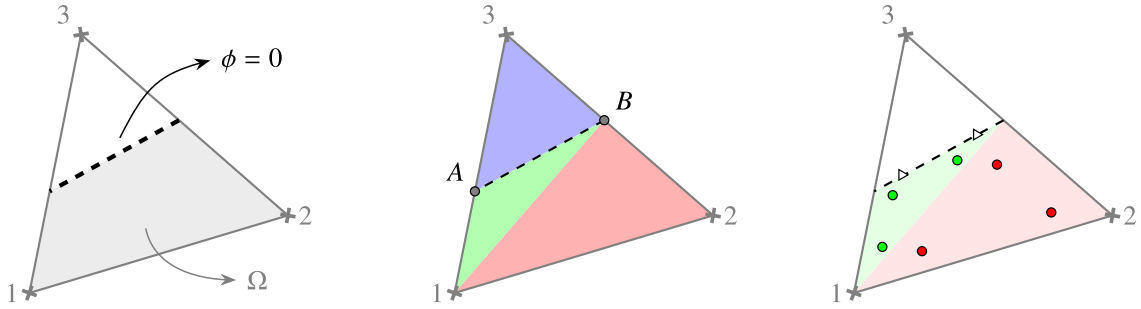
Remark 2 For a standard Newtonian case, $\mathbf{C}$ is constant and can be computed once from the material parameters μ and κ. Conversely, in the non-Newtonian case, $\mathbf{C}$ is obtained from the previous iteration material response (*i.e.*, effective viscosity), meaning that $\mathbf{C} = \mathbf{C}(\mu^k, \kappa)$. Please refer to [52] for more details on non-Newtonian fluid modeling with a similar PFEM formulation.

Remark 3 We remark that the standard Nitsche imposition presented in Eq. 5 has been slightly modified after the introduction of the segregated solution strategy. More specifically, in the mass-conservation problem (Eq. 12), we have neglected the contribution of the Nitsche stability term, *i.e.*, $\langle q\mathbf{n}, \mathcal{P}(\mathbf{v} - \bar{\mathbf{v}})\rangle_{\Gamma_\phi}$. In this respect, we note that the impact of this term over the overall solution is expected to be negligible, since the difference $\mathbf{v} - \bar{\mathbf{v}}$ tends to zero (or is of the order of the residual tolerance) after the convergence of the velocity step. This means that our Nistche imposition terms effectively turn into $\beta\langle \mathbf{w}, \mathcal{P}(\mathbf{v} - \bar{\mathbf{v}})\rangle_{\Gamma_\phi} - \langle 2\mu\mathbf{d}'(\mathbf{w}) \cdot \mathbf{n}, \mathcal{P}(\mathbf{v} - \bar{\mathbf{v}})\rangle_{\Gamma_\phi}$.

6 Subintegration

After having established the role of the distance function in characterizing the domain boundaries (as discussed in Sect. 2), it becomes evident that certain elements will extend into both the interior and exterior of the computational domain. To ensure accurate computations, it is crucial to distinguish between these two regions and integrate the previous variational terms only inside the computational domain. This is achieved through a procedure called subintegration, which adjusts the position of the integration points in the intersected elements according to the position of the level set. Skipping such subintegration procedure would not only imply the addition (or subtraction) of some mass to the system but also a deterioration in the formulation accuracy rates.

Figure 2 describes the subintegration procedure for a sample triangle intersected by an unfitted boundary. Figure 2a presents the configuration of the example. As it can be observed, the element conformed by the nodes 1, 2, and 3 is intersected by the zero isosurface of the level set (black dashed line), resulting in a portion of the element outside the computational domain (white region) and a "wet" portion belonging to Ω (grey region) where the governing equations are to be evaluated. The first step of the subintegration is to virtually split the intersected triangle (tetrahedron in 3D) into sub-triangles (sub-tetrahedra in 3D) in accordance to the position of the level set. It is important to stress that this is a virtual subdivision, meaning that no modification in the mesh connectivity is done. Figure 2b depicts the splitting for the present example. As can be noted, the splitting starts



(a) Intersected triangle configuration. (b) Triangulation. (c) Integration points reallocation.

Fig. 2 Steps of the subintegration for an intersected element. **a** Configuration of a sample intersected element. The black dashed line represents the unfitted boundary, while the grey shadowed region denotes the "wet" region occupied by the fluid. **b** Virtual splitting of the intersected elements. **c** Reallocation of Gauss points in the "wet" portion of the element

by finding which are the intersection points between the element edges and the level set, something that can be easily done from the nodal values of the distance function. Specifically, for the current triangle, we find intersection points A and B for edges 1–3 and 2–3, respectively. This results in a splitting pattern conformed by the sub-triangles 1AB, 12B, and 3AB (green, red, and blue regions in Fig. 2b). Once the splitting is completed, the position of the "new" integration points (round markers in Fig. 2c) can be computed as usual using a Jacobian transformation from the standard isoparametric triangle to the sub-triangles in the splitting. In this regard, we note that only the integration points belonging to sub-triangles lying on Ω (green and red sub-triangles in the example) should be considered as the other ones (blue sub-triangle in the example) are outside the computational domain. Similarly, the integration points over the interface can be also obtained (triangle markers in Fig. 2c) by doing the Jacobian transformation from the standard isoparametric line to the intersection AB.

For the sake of completeness, in Fig. 3, we also retake the toy problem presented in Sect. 2 to describe how the subintegration procedure is applied to two elements with a different splitting pattern. Finally, it is worth mentioning that even though, in Figs. 2 and 3, we show the procedure for the standard 2nd order Gauss integration rule, the very same technique can be equally applied to any other integration order/rule.

7 Remeshing algorithm

The PFEM is a Lagrangian finite element method that employs an efficient remeshing strategy to address mesh distortion when solving problems involving large deformations. The mesh reconstruction algorithm of PFEM consists of three main steps:

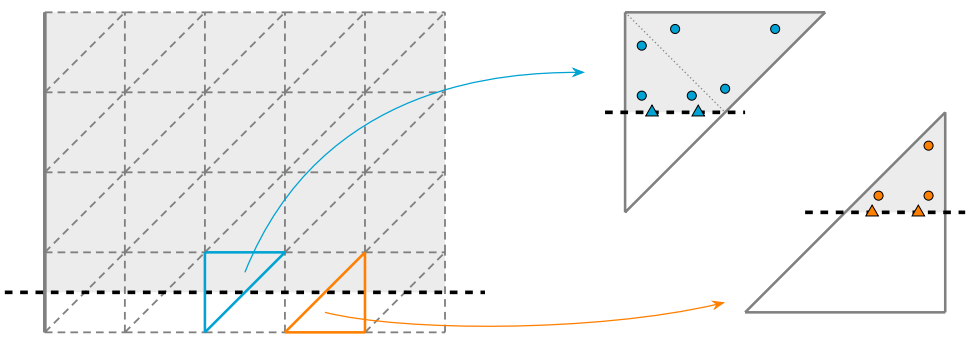


Fig. 3 Graphical representation of the subintegration strategy. The grey dashed lines in the left figure depict the mesh, while the grey solid one does so for the body-fitted boundary. The dashed black line represents the unfitted boundary. The two right figures depict the location of the integration points for the elements highlighted in cyan and orange. The circle dots show the integration points for the area (volume) integrals and triangle ones do so for the boundary ones. Complementary, the grey dotted line denotes the auxiliary splitting edges

1. *Elimination of elements*: Removing elements from the previous distorted mesh and maintaining the nodes.
2. *Delaunay triangulation*: Performing Delaunay triangulation [1] over the nodes of the previous mesh.
3. *Boundary recognition*: Identifying the actual boundaries using the Alpha-Shape method [2].

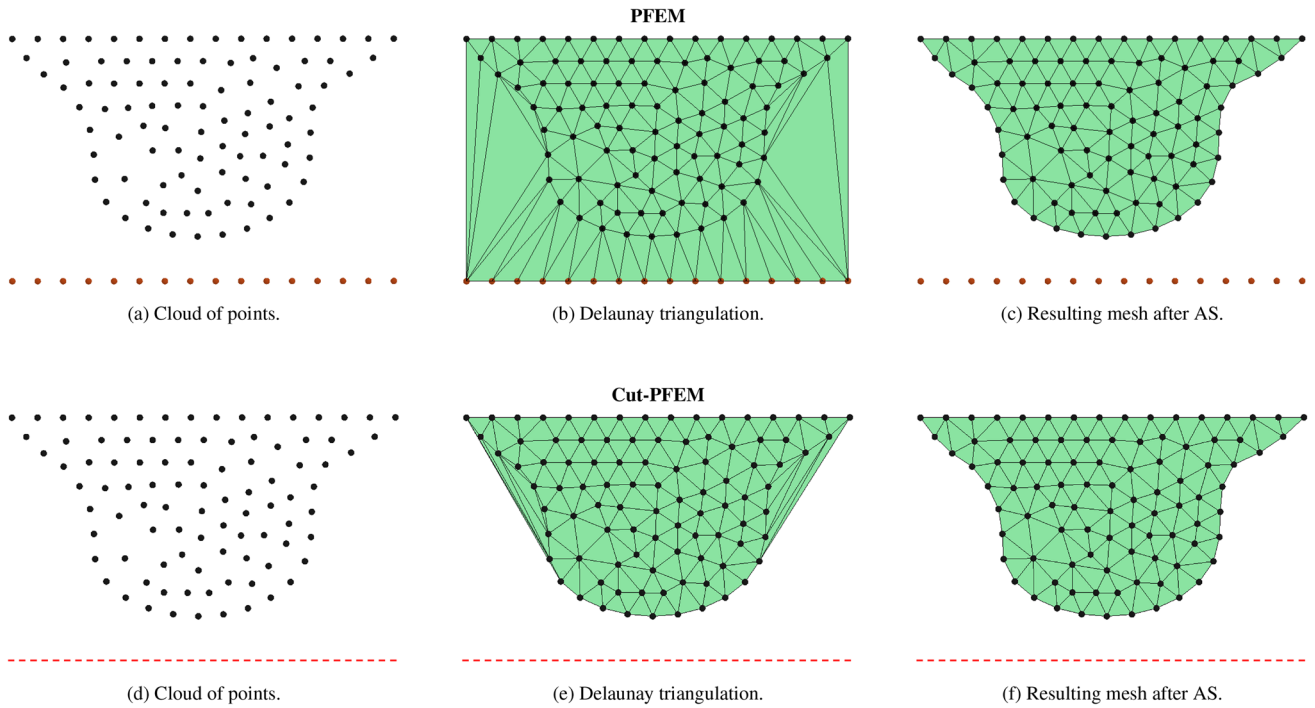
For more in-depth information about the PFEM remeshing algorithm, refer to [53]. The same remeshing algorithm described above is also utilized in Cut-PFEM. However, as we will explain later, the proposed method allows for the mitigation of certain drawbacks associated with the standard PFEM.

In [8], it was shown that the remeshing strategy employed in PFEM can introduce artificial topological changes and variations in volume within the computational domains. These undesired effects primarily arise from the combination of Delaunay triangulation and the Alpha-Shape method. These processes not only have the potential to alter the connectivity within the internal domain but can also add or remove elements at the external boundaries of the computational domain. Notably, the addition of elements that connect the fluid free surface with the rigid boundaries is the mechanism through which PFEM detects contact with the external boundaries.

Figures 4a–c and 5a–c provide a graphical representation of how contact detection is performed within the PFEM framework. Delaunay triangulation is carried out over a point cloud that encompasses not only the nodes of the previous fluid mesh but also nodes representing the discretization of the rigid boundaries. In Fig. 4a–c, the fluid remains separated from the rigid boundaries, as none of the contact elements within the convex hull formed by the Delaunay triangulation meet the Alpha-Shape criterion. In contrast, in the scenario depicted in Fig. 5a–c, the fluid has approached the rigid boundaries closely enough that some of the elements connecting the fluid and the rigid nodes remain, as they fulfill the Alpha-Shape control.

These illustrations clearly show that this contact detection algorithm has two main drawbacks. First, it results in an anticipation of the timing of physical contact. Second, it artificially introduces new elements into the computational domain, violating mass conservation. The relevance of both these inconveniences depends on the mesh size, as discussed in [8].

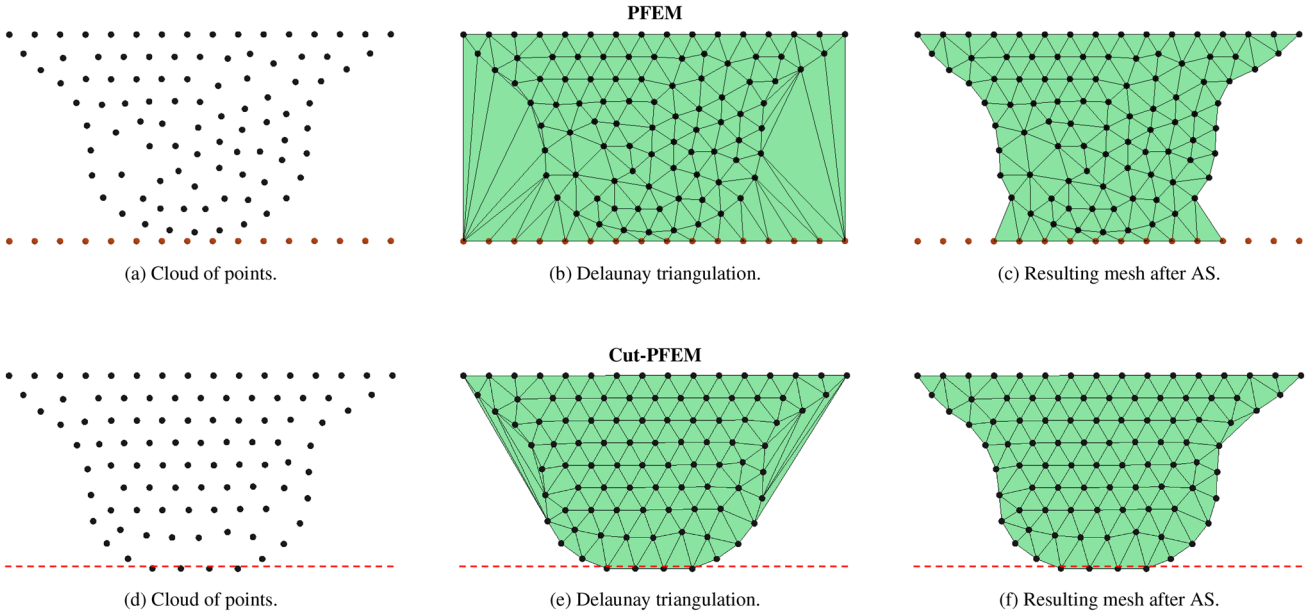
In Cut-PFEM, these two inherent drawbacks of PFEM remeshing are alleviated by the fact that the rigid contours are solely virtual, and therefore, they are not considered during the remeshing step. This is visually demonstrated in Figs. 4d–f and 5d–f, where we represent the virtual boundary with a dashed line. The Delaunay triangulation is conducted solely over the fluid nodes, resulting in a smaller convex hull compared to the one created by the standard PFEM strategy. This approach helps to prevent the



(a) Cloud of points. (b) Delaunay triangulation. (c) Resulting mesh after AS.

(d) Cloud of points. (e) Delaunay triangulation. (f) Resulting mesh after AS.

Fig. 4 Remeshing steps in standard PFEM and Cut-PFEM for a non-contact situation



(a) Cloud of points. (b) Delaunay triangulation. (c) Resulting mesh after AS.

(d) Cloud of points. (e) Delaunay triangulation. (f) Resulting mesh after AS.

Fig. 5 Remeshing steps in standard PFEM and Cut-PFEM for a contact situation

generation of contact elements that underlie the topological issues described earlier. Unlike the standard PFEM, the contact between the fluid and rigid boundaries is not guaranteed through geometry but rather by the imposition, in a weak form, of the Nitsche boundary terms, as presented in the previous sections.

Another significant advantage of Cut-PFEM mesh handling is that the boundary nodes are treated like any other nodes within the mesh. This means that they can be safely engaged in standard procedures for adding or removing nodes to enhance mesh quality, as discussed in [53]. This is in contrast to the standard PFEM, where boundary nodes require special treatment to prevent numerical issues such as artificial fluid leakage [53]. This improved behavior of Cut-PFEM over the conventional PFEM is even more clear in the presence of local mesh refinement or when slip BCs are considered [13, 15].

Remark 4 In a Cut-PFEM framework, fluid–solid detachment is modeled as in the standard PFEM, *i.e.*, the contact elements are removed from the mesh when they do not fulfill the AS check anymore [8]. This procedure induces a mass loss whose magnitude is, therefore, analogous for PFEM and Cut-PFEM solution schemes. Nevertheless, we also remark that the use of slip boundary conditions has shown to be helpful in reducing this mass-conservation issue [10, 13], and these conditions are treated more easily in Cut-PFEM than in PFEM.

Remark 5 The Cut-PFEM only deals with elements and thus requires an *ad-hoc* procedure for the treatment of the isolated nodes that may be created during the PFEM remeshing [53]. A possibility is to fix these nodes in a strong form, something that might result in boundaries represented by a collection of fixed nodes, as happens in the standard PFEM. Alternatively, these nodes can be removed, together with their kinematics, from the cloud of points. However, we highlight that such removal does not affect all the isolated nodes, but only the (typically) small portion of them reaching virtual walls. In this work, we opted to remove the isolated nodes as these have a minimal effect on the examples we are solving.

8 Solution algorithm

Aiming at further clarifying the discussion at hand, in this section, we present two algorithms. In the first one (Alg. 1), we describe the steps of the Cut-PFEM simulation strategy. For each time step, these can be summarized in remeshing, distance function calculation, and non-linear solution strategy loop. The second algorithm (Alg. 2) details one of the steps in Alg. 1, the assembly of the linear momentum problem (Eq. 8). In this regard, we note that we omitted the continuity problem algorithm as it would be almost identical to Alg. 2.

Finally, we shall mention that a complete open-source implementation of the Cut-PFEM presented in this work can be found in Kratos Multiphysics [54, 55]. For both pre- and post-processing, we use the GiD simulation software [56, 57].

Algorithm 1: Cut-PFEM solution algorithm.

Data:
k_{max} : Max. iterations
tol : Residual tolerance
T : Simulation end time
Δt : Time step increment
$\mathcal{M}$: Computational mesh
while $t \leq T$ **do**
  // Execute remeshing algorithm
  if $\mathcal{M}$ *is distorted* **then**
    RemoveElements($\mathcal{M}$);
    DelaunayTriangulation($\mathcal{M}$);
    BoundaryRecognition($\mathcal{M}$);
  end

  // Calculate distance in current mesh
  CalculateDistanceFunction($\mathcal{M}$) $\rightarrow \phi$;

  // Non-linear solution strategy loop
  while $k \leq k_{max}$ **do**
    // Calculate velocity update
    AssembleMomentum($\mathcal{M}$) ; // Assemble Eq. 8 (see Alg. 2)
    SolveMomentum() $\rightarrow \Delta\tilde{\mathbf{v}}^{k+1}, r_M^C$; // Solve Eq. 8
    // Update nodal positions
    UpdateTopology($\mathcal{M}, \Delta\tilde{\mathbf{v}}^{k+1}$) $\rightarrow \mathbf{x}^{k+1}$;
    // Calculate pressure update
    AssembleContinuity($\mathcal{M}$) ; // Assemble Eq. 12
    SolveContinuity() $\rightarrow \tilde{p}^{k+1}, r_C$; // Solve Eq. 12
    // Calculate new material response
    UpdateMaterialResponse($\tilde{\mathbf{v}}^{k+1}, \tilde{p}^{k+1}$) $\rightarrow \boldsymbol{\sigma}^{k+1}$;
    // Update time derivatives (acceleration)
    UpdateTimeDerivatives($\tilde{\mathbf{v}}^{k+1}$) $\rightarrow \dot{\mathbf{v}}^{k+1}$;
    // Check convergence
    if $r_M \leq tol$ *and* $r_C \leq tol$ **then**
      break;
    else
      k += 1
    end
  end

  // Advance in time
  t += Δt;
end

Algorithm 2: Cut-PFEM momentum assembly.

Data:
$\mathcal{M}$: Computational mesh
Output:
$\mathbf{r}$: Global momentum RHS vector
$\mathbf{K}$: Global momentum LHS matrix

// Initialize RHS and LHS
$\mathbf{r} \rightarrow \mathbf{0}$;
$\mathbf{K} \rightarrow \mathbf{0}$;
foreach *elem in* $\mathcal{M}$ **do**
  // Check if the element is cut by the unfitted boundaries
  $n_{pos} = 0$; // No. positive distance nodes
  $n_{neg} = 0$; // No. negative distance nodes
  is_cut = False;
  foreach *node in elem* **do**
    if $node(\phi) < 0$ **then**
      n_{neg} += 1;
    else
      n_{pos} += 1;
    end
  end
  if $n_{pos} \neq 0$ *and* $n_{neg} \neq 0$ **then** is_cut = True ;
  // Calculate shape functions values
  if is_cut **then**
    CalculateSubintegration($elem$);
    CalculateCutShapeFunctionsValues($elem$);
    CalculateInterfaceShapeFunctionsValues($elem$);
  else
    CalculateStandardShapeFunctionsValues($elem$);
  end
  // Assemble element contributions
  $\mathbf{r}$ += $\mathbf{r}(elem)$; // RHS contribution (Eq. 10)
  $\mathbf{K}$ += $\mathbf{K}(elem)$; // LHS contribution (Eq. 9)
  if is_cut **then**
    $\mathbf{r}$ += $\mathbf{r}^{\phi}(elem)$; // RHS interface contribution (Eq. 11)
    $\mathbf{K}$ += $\mathbf{K}^{\phi}(elem)$; // LHS interface contribution (Eq. 9)
  end
end

return $\mathbf{r}$, $\mathbf{K}$

9 Numerical examples

9.1 Falling cylinder in a viscous fluid

In this test, we consider the gravity-induced motion of a cylinder in a channel filled with a viscous fluid. The solid body, at rest at the beginning, is expected to undergo an accelerated motion until it reaches an asymptotic terminal velocity. Due to the narrowness of the channel and the high viscosity of the fluid, the problem dynamics are greatly influenced by the interaction of the fluid with the lateral walls. The central importance of the BCs and the availability of an analytical solution for the terminal velocity of the solid motivate the resolution of this test in the proposed Cut-PFEM validation. It is worth noting that this test also involves fluid–structure interaction (FSI) phenomena between the viscous fluid and the cylinder. These aspects, which we highlight are not the focus of this work, are handled using the well-assessed PFEM–FEM monolithic FSI strategy presented in [14]. The interaction between the solid circle and the fluid is performed using the standard PFEM body-fitted approach. However, our focus is on the lateral boundaries of the channel which are treated in an unfitted manner in the Cut-PFEM spirit.

For practical purposes as well as to ease the comparison to the reference work [58], we consider the gravity acceleration $g = 9.81$ m/s^2 acting in the horizontal direction, as shown in Fig. 6. In [58], the same problem was solved considering slip conditions using an Immersed Finite Element Method (IFEM). The cylinder has radius $r = 2.5$ mm and is located at the central point of the channel, whose width is $L = 40$ mm. A mesh of mean size $h = 0.50$ mm has been used in the zone around the solid object and $h = 0.75$ mm in the rest of the domain.

The fluid has viscosity $\mu = 0.1\, Pa \cdot s$ and density $\rho_f = 1000$ kg/m^3. The solid cylinder has Young modulus $E = 10^7$ Pa, Poisson ratio $\nu = 0.35$, and density $\rho_s = 1200$ kg/m^3. Under these conditions, the deformations experienced by the solid during its motion can be neglected, thus making it possible to estimate the terminal velocity of the solid (v_{end}) using the analytical solution for a free-falling rigid cylinder proposed in [59]. According to this work, the terminal velocity of a cylinder within a channel with pure slip walls can be computed as

$$v_{\text{end}} = \frac{(\rho_s - \rho_f) g r^2}{4\mu} \left(ln\left(\frac{L}{r}\right) - 0.9157 + 1.7244\left(\frac{r}{L}\right)^2 - 1.7302\left(\frac{r}{L}\right)^4 \right). \quad (14)$$

For the sake of a more complete validation and comparison, the same test has been solved considering both slip and no-slip BCs on the lateral rigid walls. For the no-slip case, our reference solution is the one obtained in [14] using the same FSI approach used here but with a standard PFEM body-fitted formulation.

The time evolution of the cylinder vertical velocity obtained with slip and no-slip BCs is shown in Fig. 7a and b, respectively. As depicted in Fig. 7, the results show a significant difference between the slip and no-slip boundary models, highlighting the crucial role of the BCs in this problem. Additionally, for both the no-slip and slip cases, we observe that the Cut-PFEM results practically overlap with the reference solutions, proving the accuracy of the proposed methodology for the modeling of the BCs. This strong agreement with the reference solution is further substantiated by the results depicted in Figs. 8 and 9, which illustrate the velocity field solution at the final time instant, $t = 1.5\ s$, for the no-slip and slip cases, respectively.

This example is also employed to assess the sensitivity of the Cut-PFEM solution to the penalty constant γ (Eq. 7). For this purpose, we solved the same problem while considering a wide range of penalty constants, spanning five orders of magnitude from the smallest to the largest values. For this analysis, we focus on the no-slip solution only.

We initially examine the effect of γ choice on the linear system's conditioning by monitoring the number of iterations required by the iterative linear solver to converge, both with and without a preconditioner. In this study, we utilized a bi-conjugate gradient (BiCG) iterative solver and an incomplete LU factorization (ILU0) as preconditioner. Figure 10a illustrates the number of iterations of the linear solver obtained for various penalty coefficient values. The graphs demonstrate that when the preconditioner is used, the number of iterations remains limited across the entire range of the analyzed penalty coefficients. Instead, without a preconditioner, the number of iterations significantly increases for $\gamma = 10^4$, although it remains nearly constant for all the other tested values of the penalty coefficient, remarkably ranging from $\gamma = 10^0$ to $\gamma = 10^4$. To complete the penalty parameter analysis, we also monitor its impact on the solution accuracy. Figure 10b displays the final cylinder velocity (at $t = 1.5$ s) for the same range of penalty coefficients. The graph reveals that the solution is practically identical for the tested values of γ (the percentage difference between the maximum and the minimum velocity values is smaller than the 0.1%).

We shall remark that this study evidences the robustness of the method, as it enables the use of a wide range of penalty coefficient values without impacting the solving scheme, both in terms of the quality of the linear system and solution accuracy.

9.2 Non-Newtonian flow

In this second test, we model the flow of a mass of liquefied sand, which is modeled as a non-Newtonian fluid. The study of this type of free-surface flow is pertinent to various real-world situations and engineering challenges. These include predicting the runout of geophysical flows like mudflows or debris flows, as well as assessing the slump of fresh concrete or cement pastes to determine their workability. Despite the difference in scale between the two mentioned problems, in both scenarios, the interaction between the mobilized material and the underlying surface plays a crucial role in the problem's dynamics. Additionally, it is essential to minimize mass variations induced by the numerical model, which could significantly affect the results. As explained in previous sections, this latter aspect presents a notable challenge within a standard PFEM framework, which requires the creation of new elements to detect contact with the external boundaries. Consequently, this test is particularly valuable for evaluating the accuracy and relevance of the proposed Cut-PFEM strategy.

The problem is based on the experimental test presented in [60]. At the beginning, the material is retained by a vertical wall and has initial height $H = 26.7$ cm and length $L = 32$ cm, as shown in Fig. 11. The material starts moving after the removal of the wall, which is assumed to occur instantaneously. Following [61], the saturated sand is modeled as a Bingham fluid with the following material properties: density $\rho = 1600$ kg/m^3, dynamic viscosity $\mu = 300$ Pa s and yield shear $\tau_0 = 50$ Pa. To address the well-known numerical challenges associated with this non-Newtonian model, we employ the same regularized model as the one used in [52]. The regularization parameter used for the Bingham model is $m = 1000$ s. In the reference work [61], no-slip BCs were applied to both the horizontal and vertical walls using a standard PFEM approach. In this study, we maintain the same choice of BCs but employing the Cut-PFEM for their modeling. The penalty constant γ of Eq. 7 has been set equal to 10.

A mesh convergence study has been also performed by running the test with five different refinement levels, corresponding to an average element size of 0.04 m, 0.02 m, 0.01 m, 0.005 m, and 0.0025 m.

Figure 12 shows the Cut-PFEM results obtained with the finest mesh ($h = 0.0025$ m) at six time instants. The velocity contours are plotted over the deforming configuration of the non-Newtonian fluid. The pictures show that the mobilized material undergoes a sudden acceleration after the instantaneous wall removal. After 1 s, the material slows down sensibly and the velocity progressively reduces. The pictures also show that, due to the use of no-slip boundary conditions, part of the material keeps stuck to the vertical wall until the end of the simulation.

Figure 13 depicts the time evolution of the front position x-coordinate of the moving material. The Cut-PFEM results are plotted together with the experimental observations of

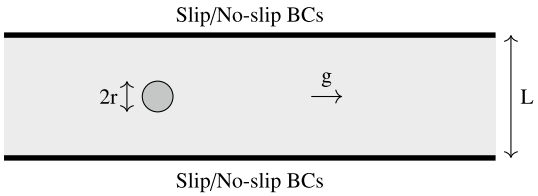
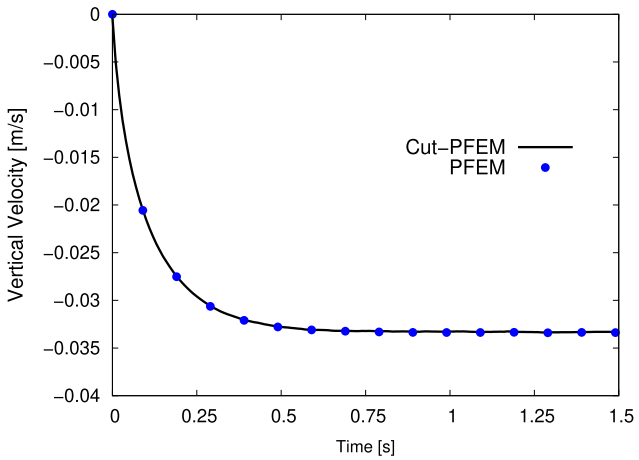
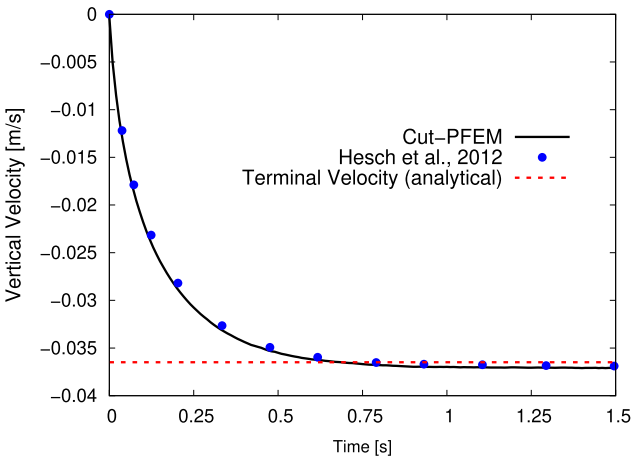


Fig. 6 Falling cylinder in a viscous fluid. Problem description



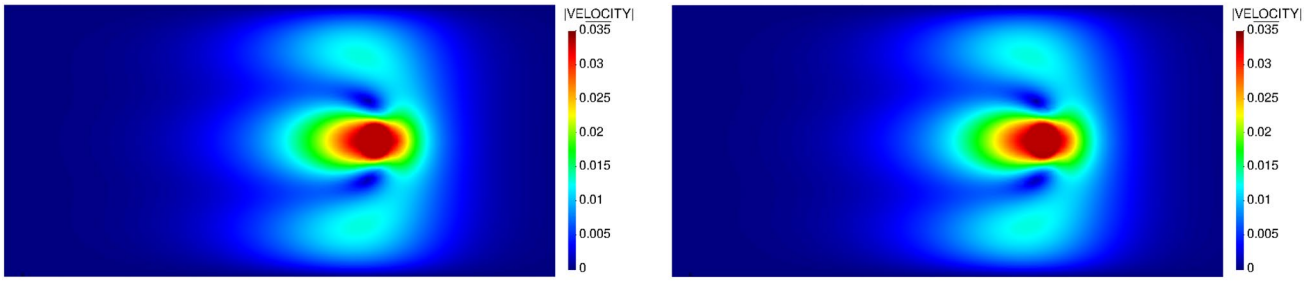
(a) No-Slip conditions.



(b) Slip conditions.

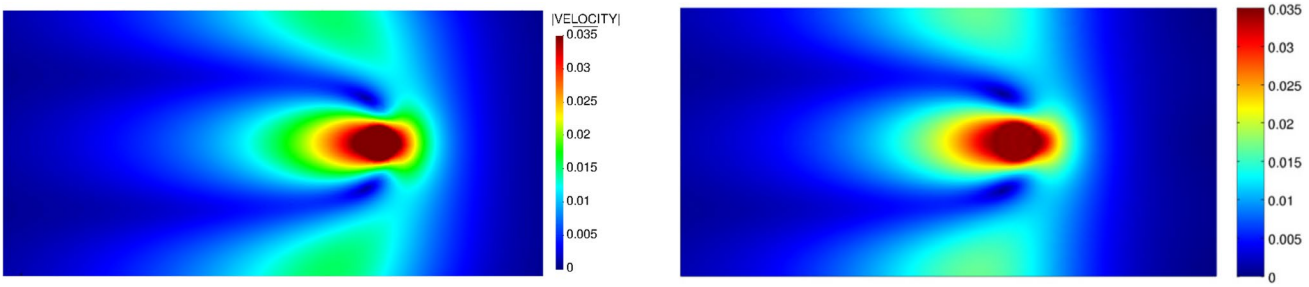
Fig. 7 Falling cylinder in a viscous fluid. Time evolution of the vertical velocity of the cylinder. **a** No-slip solutions obtained with the proposed Cut-PFEM and the standard body-fitted PFEM. **b** Slip solutions obtained with Cut-PFEM and IFEM from [58], and terminal velocity computed analytically using Eq. 14 according to [59]

[60], the numerical results obtained in [61], and the results of the standard PFEM. We note that the PFEM results have been obtained using the same two-step FIC-stabilized formulation as the Cut-PFEM but with the customary PFEM no-slip body-fitted BCs. Both Cut-PFEM and PFEM results plotted in the graph have been obtained considering a mean mesh size $h = 0.0025$ m. A good agreement is found among the three distinct numerical formulations, while the experimental observations exhibit a slight divergence, particularly during the initial phase of the test. This discrepancy between the numerical and experimental results may arise from the assumption in the numerical tests that the retaining vertical



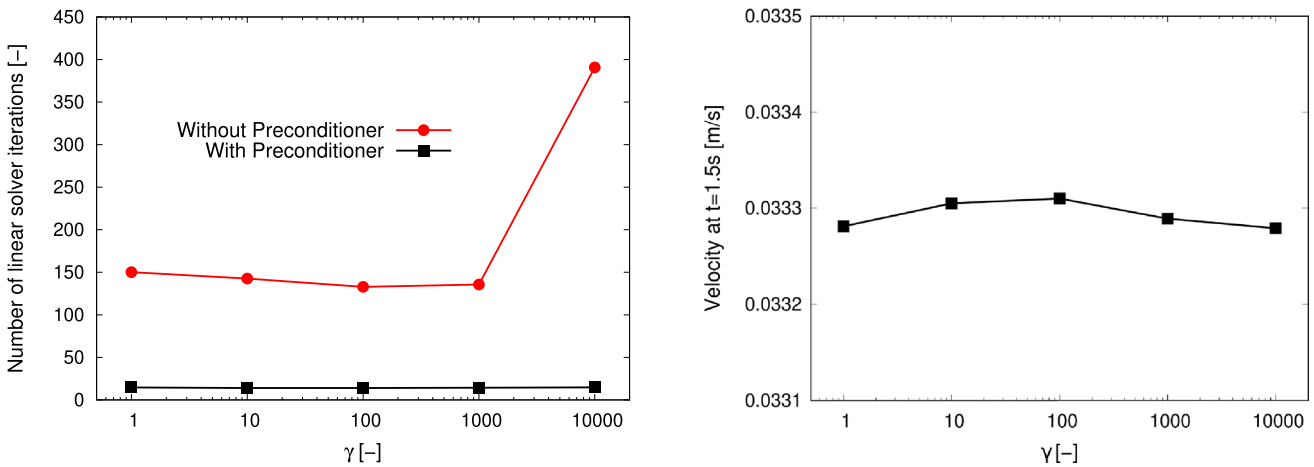
Cut-PFEM. PFEM.

Fig. 8 Falling cylinder with no-slip BCs on the lateral walls. Velocity field at $t = 1.5$ s obtained with Cut-PFEM and PFEM



Cut-PFEM. IFEM from [58].

Fig. 9 Falling cylinder with slip BCs on the lateral walls. Velocity field at $t = 1.5$ s obtained with Cut-PFEM and IFEM from [58]



(a) Iterations of BiCG linear solver. (b) Cylinder velocity at $t = 1.5$ s.

Fig. 10 Falling cylinder with no-slip BCs on the lateral walls. Sensitivity analysis of the penalty constant γ defined in Eq. 7. **a** Mean number of iterations of the linear solver with and without preconditioner for different values of γ. **b** Final velocity for different values of γ

wall is removed instantaneously, or it could be attributed to the inadequacy of the Bingham model and relative parameters in representing the real material [61]. Nevertheless, these results confirm the accuracy of the proposed numerical strategy, also in a non-Newtonian flow scenario.

It is particularly interesting to note that, despite the global results of PFEM and Cut-PFEM are very similar for a fine mesh, the front-advancing mechanism provided by the two approaches is quite different. To emphasize this point, we display in Fig. 14 the details of the contact zone between the fluid and the basal surface for both the standard PFEM and the Cut-PFEM approaches. As explained in the previous sections, the standard PFEM necessarily creates new elements that connect the front of the fluid with the fixed nodes representing the basal surface. In contrast, in the Cut-PFEM, the walls are virtual, meaning that the fluid nodes move freely until their distance value becomes negative, at which point the no-slip BCs are weakly applied, as described earlier.

The primary consequence of this distinct treatment of no-slip BCs pertains to the mass-conservation properties of the two schemes. In this regard, the Cut-PFEM consistently outperforms the standard PFEM, because it does not require the construction of new elements to model fluid–solid contact interactions. This is corroborated by the graphs in Fig. 15a, which illustrate the time evolution of accumulated volume variations resulting from remeshing for three different meshes obtained using the Cut-PFEM and the standard PFEM. In terms of absolute value, the volume variation due to remeshing in Cut-PFEM is always smaller than in PFEM. Remarkably, even the results obtained with the Cut-PFEM coarsest mesh ($h = 0.01$ m) outperform those obtained with the PFEM finest mesh ($h = 0.0025$ m).

Figure 15a also reveals distinct convergence behaviors in the two approaches. In the PFEM, convergence occurs from above, meaning that with finer meshes, the increment of volume due to remeshing decreases. Conversely, in Cut-PFEM, convergence happens from below, as finer meshes result in a smaller loss of volume due to remeshing. This differing behavior can be attributed to the distinct mechanisms of volume alteration that occur during PFEM remeshing, as explained in detail in [8]. In this specific test, two main sources contribute to the volume variation: the elimination of excessively stretched elements on the free surface of the fluid, and the creation of new elements connecting the fluid front and the solid base. The former contribution, which leads to a volume loss, affects both the PFEM and the Cut-PFEM, instead, the latter one, which induces volume gain, only occurs in the standard PFEM. By comparing the volume conservation curves plotted in Fig. 15a, it appears clear that the volume-gain mechanism of creating new elements in the fluid front is predominant in this test. All this also explains why Cut-PFEM exhibits an overall volume loss, while PFEM results in a positive volume variation.

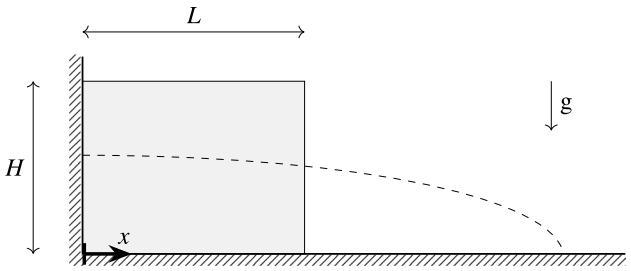


Fig. 11 Two-dimensional flow of a non-Newtonian material. Problem description

We remark that the volume-loss mechanism associated with fluid–solid detachment [8] also affects the mass conservation, but, in this test, its magnitude is relatively low versus the two mechanisms described before. Furthermore, as already explained in Remark 4, the fluid–solid detachment mechanism affects the mass conservation of PFEM and CUT-PFEM in the same way. For these reasons, we can neglect its contribution in the comparison of the two PFEM solution schemes.

Figure 15b shows the final runout x-coordinate obtained with Cut-PFEM and PFEM with five different mesh sizes. Combining these results with those of Fig. 15a, we note the close correlation between the lack of mass conservation and the overall numerical solution. In the standard PFEM, the additional mass in the system introduced by the remeshing operations makes the fluid front advance artificially. Instead, in Cut-PFEM, the opposite situation occurs. In both cases, the effects of remeshing reduces progressively with mesh refinement. We also note that, considering the same mesh size, the error of Cut-PFEM results versus the expected solution (*i.e.*, the one obtained with the finest mesh) is smaller than the one obtained with PFEM.

The advantages of using Cut-PFEM extend beyond improved mass conservation performance. As discussed in the previous section, the Cut-PFEM results in a much simpler application of slip BCs. In the preceding example, these conditions were examined in the context of confined flow. However, within a PFEM framework, it is valuable to investigate their functionality in flows involving free surfaces, such as the one in this test. To achieve this, we analyze the same problem in a three-dimensional geometry while applying slip BCs to the lateral walls. Furthermore, to demonstrate the flexibility of the Cut-PFEM strategy that allows combining standard body-fitted and unfitted boundaries, we model the lateral walls with slip BCs using Cut-PFEM and the basal surface with no-slip boundaries using the standard PFEM technique. The width of the channel equals the initial height, that is 26.7 cm.

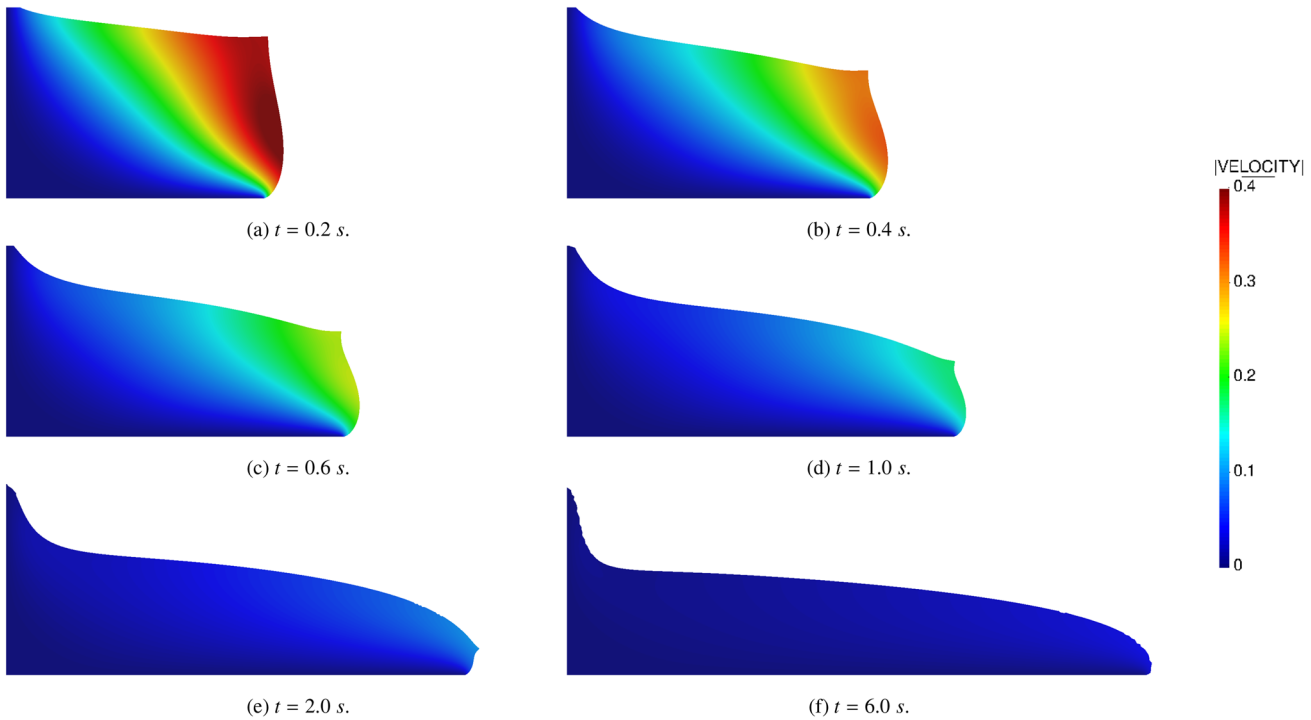


Fig. 12 Two-dimensional flow of a non-Newtonian material. Velocity field at six different time instants obtained with Cut-PFEM. No-slip BCs are applied on the contours

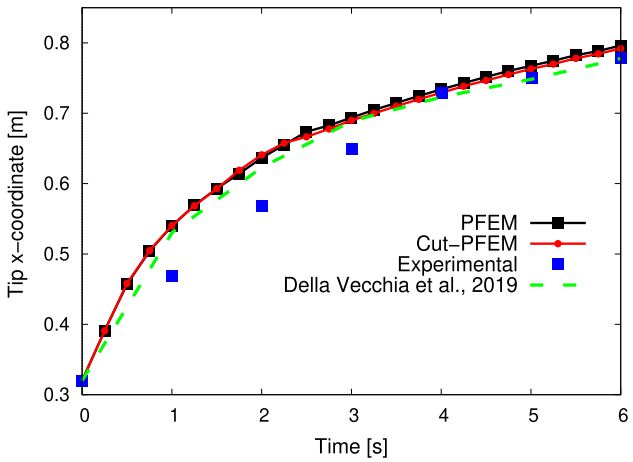


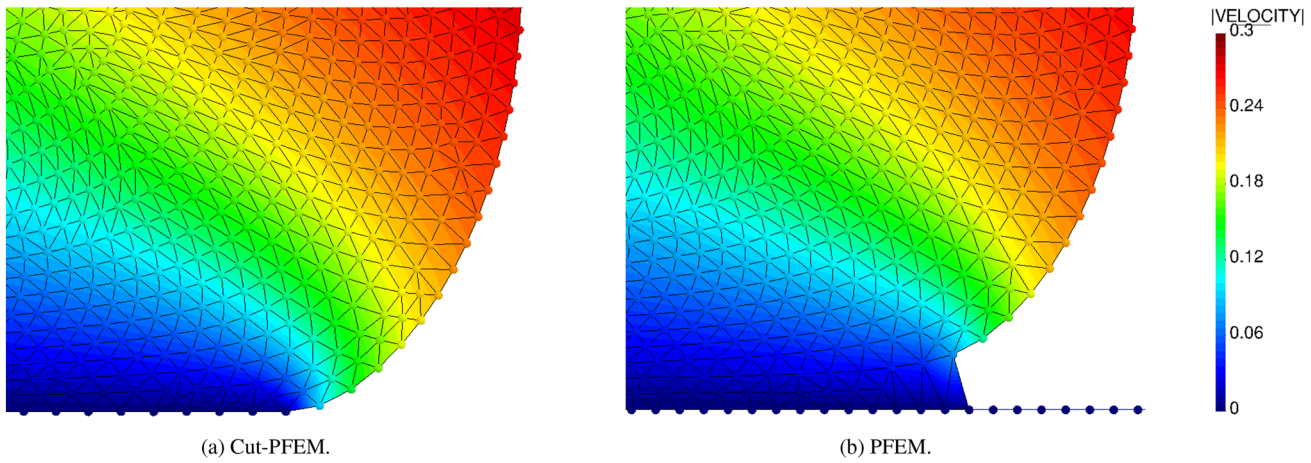
Fig. 13 Two-dimensional flow of a non-Newtonian material. Time evolution of the x-coordinate of the front of the mobilized material. The graph shows the experimental observations from [60], numerical results from [61], together with the results of a standard PFEM and the proposed Cut-PFEM approach

Figure 16 presents four images of the 3D analysis. Qualitatively, we can say that the velocity contour field proves a satisfactory modeling of the lateral walls slip BCs.

To quantitatively assess the accuracy of the slip BCs, we compare in Fig. 17 the 2D results obtained solely with PFEM with the 3D ones obtained using the Cut-PFEM to model the slip BCs on the lateral walls. More specifically, we plot the position of the free surface obtained for the 2D and the 3D case at the same time instants considered in Fig. 17. For the 3D results, we represent the position of the free surface of the central section. The graph reveals a strong agreement between the 2D and 3D results, meaning that there is no wall shear effect in the solution or, in other words, that the slip BC effectively turns into a symmetry condition as expected. This confirms the accuracy of the Cut-PFEM in modeling pure slip BCs and demonstrates its effective performance in 3D problem-solving.

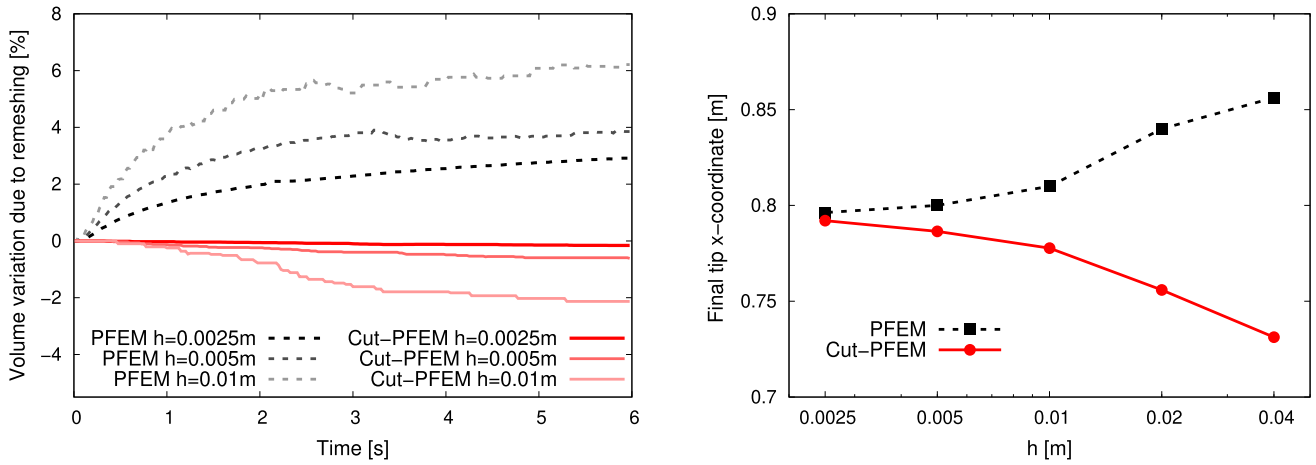
9.3 Water dam break

The objective of this final test is to prove that the proposed Cut-PFEM can be reliably employed in scenarios involving strong impacts. To this end, we examine a water dam-break test and its subsequent impact on a vertical wall. In particular, we take as reference the work presented in [62], a well-established benchmark for validating free-surface fluid solvers. Figure 18 illustrates the initial geometry of the 2D analysis. At the beginning, the water column has height $H = 29.2$ cm and width $L = 14.6$ cm. We apply no-slip BCs to all rigid contours. An average mesh size $h = 0.25$ cm and a time step duration $\Delta t = 10^{-4}$ s have been used. For the no-slip boundaries, a penalty constant $\gamma = 10$ (Eq. 7) has been used.



(a) Cut-PFEM. (b) PFEM.

Fig. 14 Two-dimensional flow of a non-Newtonian material. Detail of the contact zone of the PFEM and the Cut-PFEM analyses (results at $t = 0.5$ s)



(a) (b)

Fig. 15 Two-dimensional flow of a non-Newtonian material. **a** PFEM and Cut-PFEM time evolution of accumulated volume variations due to remeshing for three different mean mesh sizes h. **b** Final front position obtained with CUT-PFEM and PFEM for five different meshes

In Fig. 19, we compare our Cut-PFEM numerical solution alongside experimental observations at four different time intervals. The images show a notable agreement between the experimental data and our numerical simulations, affirming the suitability of Cut-PFEM for analyzing complex free surface problems, including violent impacts.

Additionally, we present in Fig. 20 the time evolution of the x-coordinate of the wavefront position prior to impact with the vertical wall. The plot includes the Cut-PFEM results together with the solution obtained using the standard PFEM, as well as the numerical and experimental results documented in [62]. Notably, all the numerical results closely align with each other but consistently show higher values compared to the experimental solution. This overestimation is a common characteristic of various numerical methods applied to this test and is generally attributed to the representation of retaining wall removal, which is instantaneous in numerical simulations.

Finally, it is also noteworthy that in the PFEM, the wavefront moves slightly faster than in the Cut-PFEM. This minor difference can be attributed to the PFEM mechanism of creating new elements at the front, which is completely avoided in the Cut-PFEM. Nevertheless, the solutions of PFEM and Cut-PFEM exhibit a high degree of similarity, as evinced by Fig. 21. Such resemblance underscores a significant advantage of the proposed Cut-PFEM approach, given that the standard PFEM is a well-established approach for simulating complex fluid–structure interaction mechanisms and violent impacts [53].

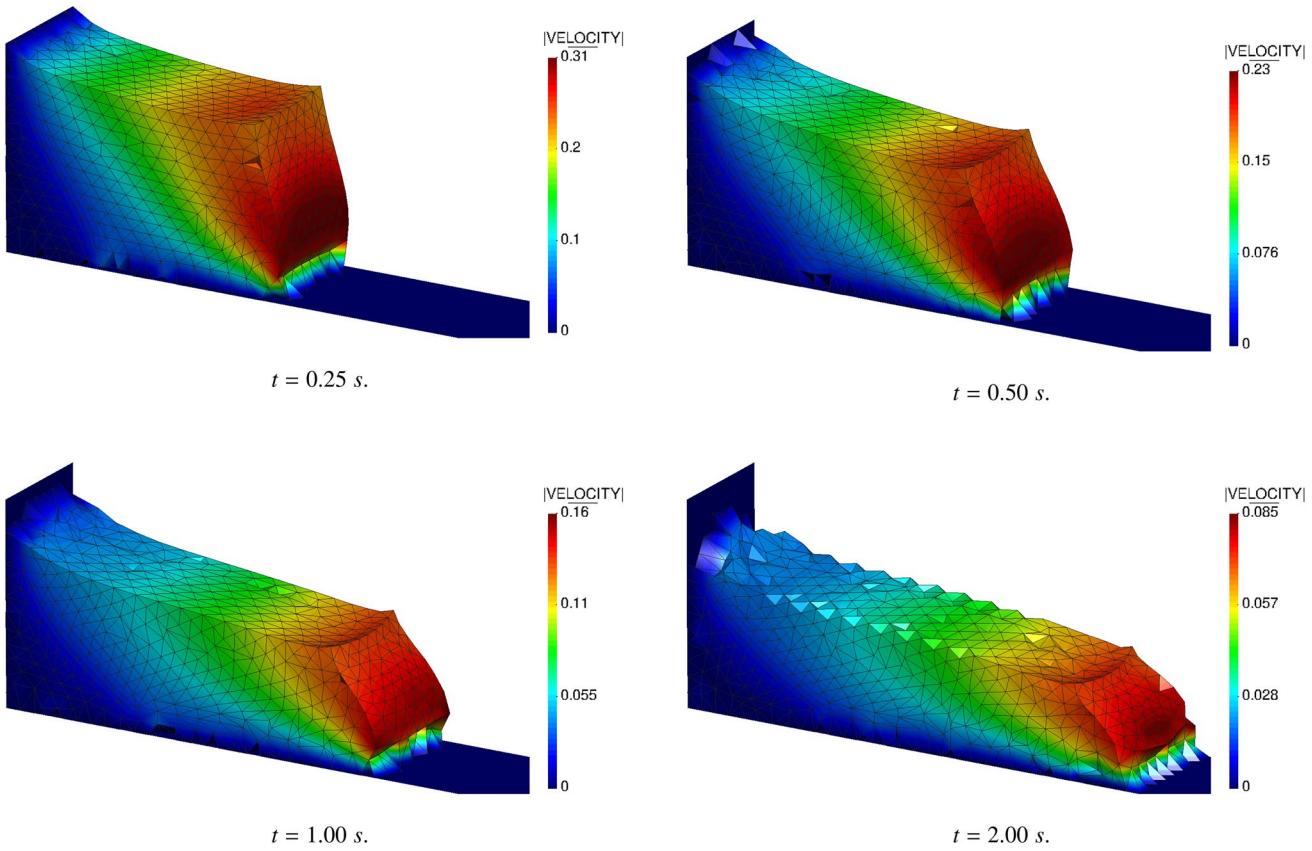


Fig. 16 Three-dimensional flow of a non-Newtonian material. Velocity field at four different time instants obtained with Cut-PFEM considering slip BC on lateral walls and standard no-slip BCs on the rest of boundaries

10 Conclusions

In this work, we have presented a new PFEM approach using unfitted boundary meshes. The method combines the fully Lagrangian description of PFEM with an implicit representation of boundaries, proper of the well-known CutFEM. In this unfitted solution algorithm, the boundary elements are recognized via the computation of a distance function, and the necessary boundary conditions are applied using Nitsche's method. The resulting unfitted PFEM strategy, which we have named Cut-PFEM, has proven to be a reliable tool for solving complex free-surface fluid problems and to improve intrinsic limitations of the standard body-fitted PFEM. In particular, thanks to its virtual representations of the boundaries, the Cut-PFEM circumvents the artificial creation of contact elements performed in the standard body-fitted PFEM and the consequent violation

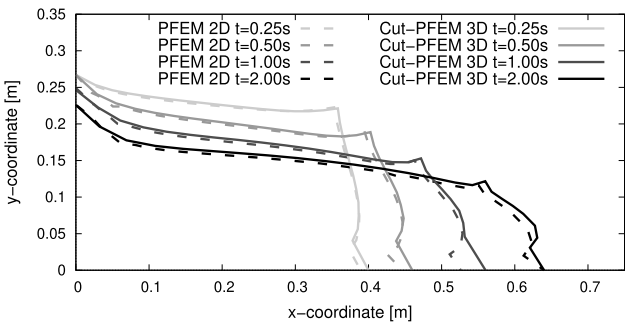


Fig. 17 Two-dimensional flow of a non-Newtonian material. 2D PFEM and midplane 3D Cut-PFEM comparison of the free-surface position at four time instants. 3D Cut-PFEM results have been obtained with pure slip (symmetry) BCs in the lateral walls

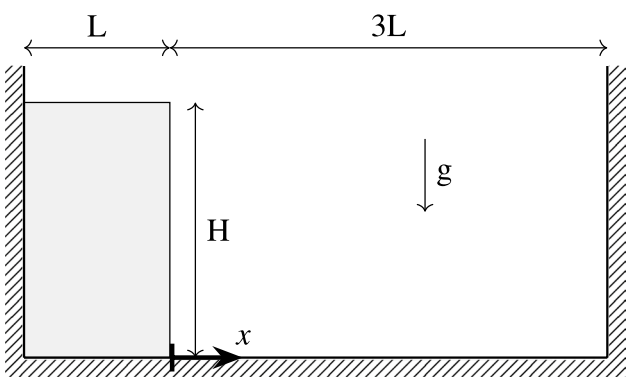
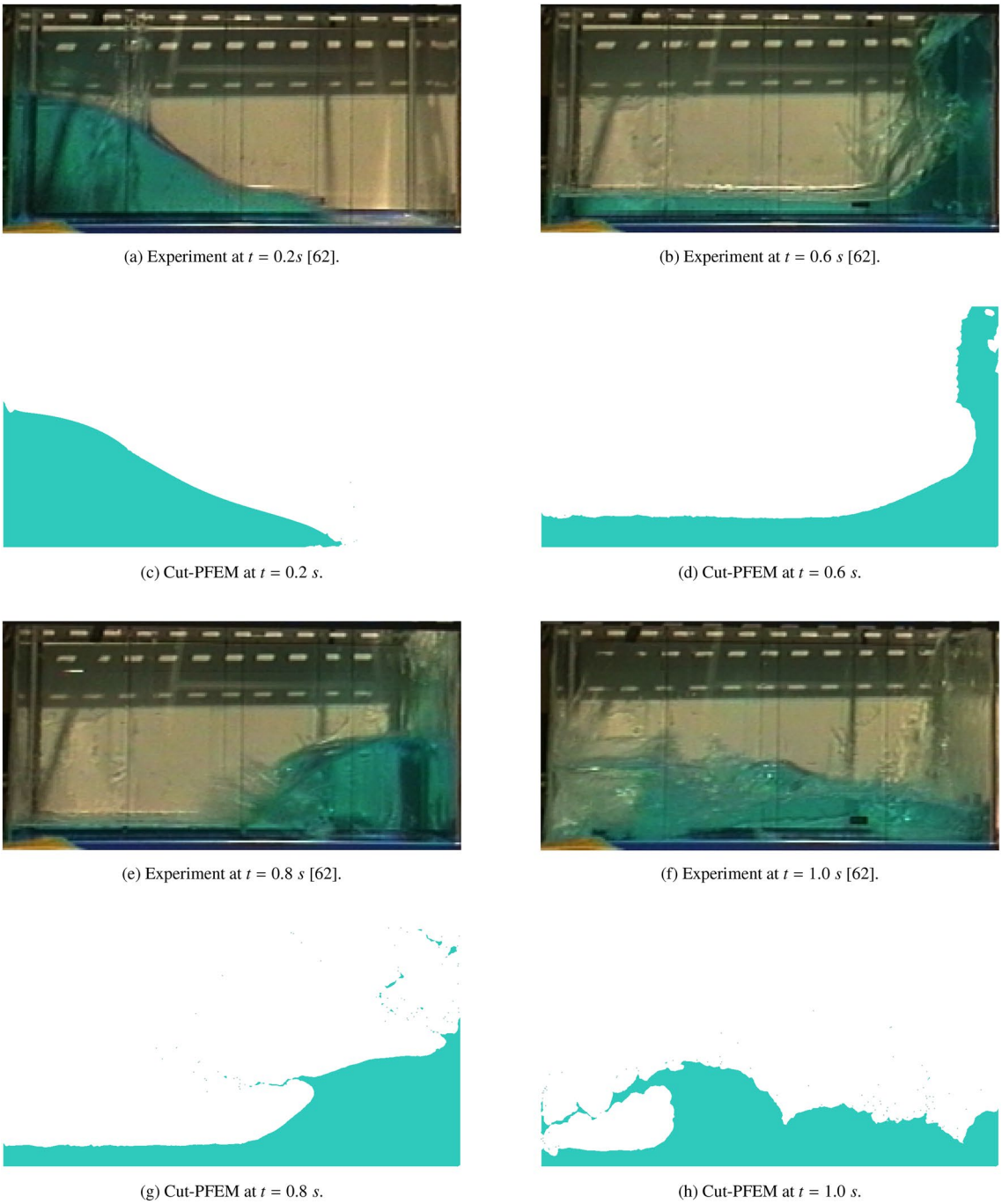


Fig. 18 Collapse of a water column. Problem description



(a) Experiment at $t = 0.2s$ [62].

(b) Experiment at $t = 0.6\ s$ [62].

(c) Cut-PFEM at $t = 0.2\ s$.

(d) Cut-PFEM at $t = 0.6\ s$.

(e) Experiment at $t = 0.8\ s$ [62].

(f) Experiment at $t = 1.0\ s$ [62].

(g) Cut-PFEM at $t = 0.8\ s$.

(h) Cut-PFEM at $t = 1.0\ s$.

Fig. 19 Water dam break. Results obtained with Cut-PFEM versus experimental observations from [62]

of mass conservation. Furthermore, Cut-PFEM allows for a natural application of slip boundary conditions without requiring any extra ad-hoc treatment as it is typically done in body-fitted PFEM. The mentioned advantages of Cut-PFEM versus standard PFEM are thoroughly demonstrated via the solution of three benchmark problems, involving complex situations and phenomena, such as free-surface flows, violent impacts, and non-Newtonian fluids. A sensibility analysis of the Nitsche's penalty parameter used in the imposition of the boundary conditions has been also presented.

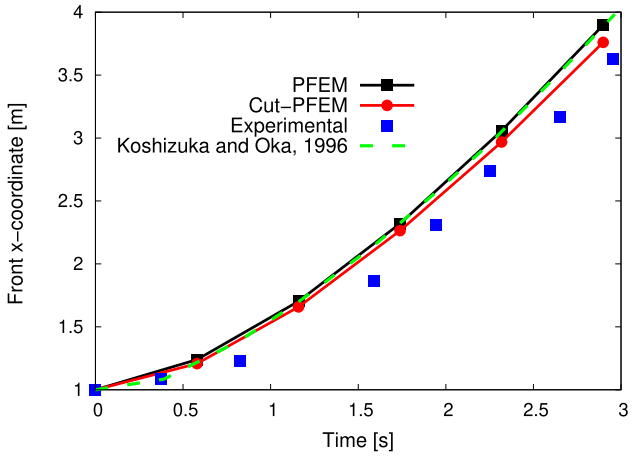
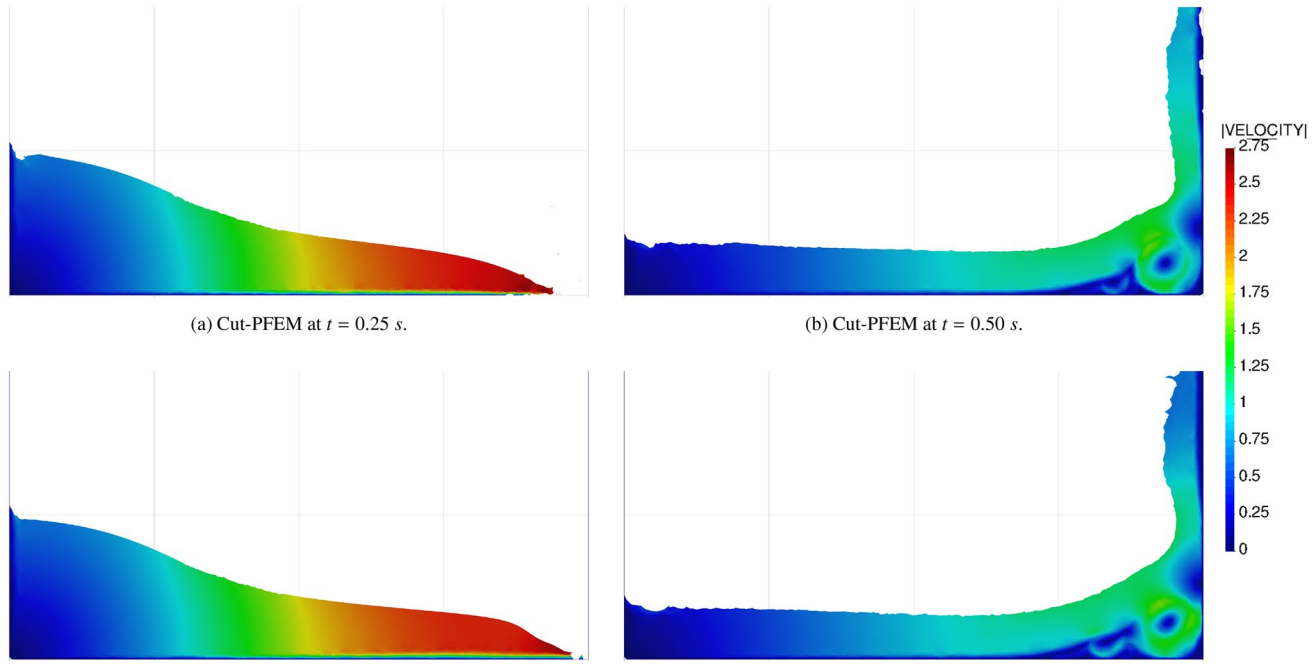


Fig. 20 Water dam break. Time evolution of wavefront position

In all situations, including 3D simulations, the accuracy and robustness of the Cut-PFEM have been proven both for slip and no-slip conditions. To conclude, we would like to remark that the main objective of this work was to verify the accuracy and robustness of the Cut-PFEM and to prove its better performance versus the standard PFEM in specific situations, such as mass conservation associated with contact detection and slip condition modeling. However, from a broader perspective, the advantages of Cut-PFEM can be much more relevant. Indeed, the method represents a fertile ground for the numerical simulation of challenging engineering problems. On the one hand, upgrading the current implementation with a level set calculation algorithm will make possible addressing problems involving complex (non-analytical) and possibly ill-conditioned boundaries (*e.g.*, *stl* meshes coming from real topographies), such as large-scale natural hazards simulations. On the other hand, one can further develop the proposed technique to be able to embed shell-like bodies, enabling thus the resolution of complex FSI phenomena involving thin structures and fluids with free surfaces. Both topics have been considered out of the scope of this paper and will be considered for future work.



(a) Cut-PFEM at $t = 0.25$ *s*.

(b) Cut-PFEM at $t = 0.50$ *s*.

(c) PFEM at $t = 0.25$ *s*.

(d) PFEM at $t = 0.50$ *s*.

Fig. 21 Water dam break. Velocity field obtained with Cut-PFEM and the standard PFEM for two time instants

Acknowledgements The authors acknowledge the financial support from the Spanish Ministry of Economy and Competitiveness, through the "Severo Ochoa Programme for Centres of Excellence in R &D" (CEX2018-000797-S). The second author acknowledges the support from MCIN/AEI/10.13039/501100011033 and FEDER *Una manera de hacer Europa* for funding his work via project PID2021-122676NB-I00.

Funding Open Access funding provided thanks to the CRUE-CSIC agreement with Springer Nature.

Data availability Data will be made available on request.

Declarations

Conflict of interest The authors have no conflicts of interest to declare that are relevant to the content of this article.

References

1. Edelsbrunner H, Tan T (1993) An upper bound for conforming Delaunay triangulations. Discrete Comput Geom 10:197–213
2. Edelsbrunner H, Mücke EP (1994) Three-dimensional alpha shapes. ACM Trans Graph 13(1):43–72
3. Idelsohn S, Oñate E, Del Pin F (2004) The particle finite element method: a powerful tool to solve incompressible flows with free-surfaces and breaking waves. Int J Numer Methods Eng 61(7):964–989
4. Oñate E, Idelsohn SR, Del Pin F, Aubry R (2004) The particle finite element method-an overview. Int J Comput Methods 1(02):267–307
5. Idelsohn SR, Oñate E, Del Pin F, Calvo N (2006) Fluid-structure interaction using the particle finite element method. Comput Methods Appl Mech Eng 195(17–18):2100–2123
6. Oliver J, Cante J, Weyler R, González C, Hernández J (2007) Particle finite element methods in solid mechanics problems. In: Computational plasticity. Springer, pp 87–103
7. Carbonell JM, Oñate E, Suárez B (2010) Modeling of ground excavation with the particle finite-element method. J Eng Mech 136(4):455–463
8. Franci A, Cremonesi M (2017) On the effect of standard PFEM remeshing on volume conservation in free-surface fluid flow problems. Comput Part Mech 4(3):331–343
9. Rodriguez JM, Larsson S, Carbonell JM, Jonsén P (2022) Implicit or explicit time integration schemes in the PFEM modeling of metal cutting processes. Comput Part Mech 9(4):709–733
10. Fernández E, Février S, Lacroix M, Boman R, Papeleux L, Ponthot J-P (2023) A particle finite element method based on level-set functions. J Comput Phys 487:112187
11. Falla R, Bobach B-J, Boman R, Ponthot J-P, Terrapon VE (2023) Mesh adaption for two-dimensional bounded and free-surface flows with the particle finite element method. Comput Part Mech 10(5):1049–76
12. Masó M, Franci A, de Pouplana I, Cornejo A, Oñate E (2022) 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 9(1):15
13. Franci A (2016) Unified Lagrangian formulation for fluid and solid mechanics, fluid-structure interaction and coupled thermal problems using the PFEM. Springer Theses, Springer International Publishing
14. Franci A, Oñate E, Carbonell JM (2016) Unified Lagrangian formulation for solid and fluid mechanics and FSI problems. Comput Methods Appl Mech Eng 298:520–547
15. Cerquaglia ML, Deliége G, Boman R, Terrapon V, Ponthot J-P (2017) Free-slip boundary conditions for simulating free-surface incompressible flows through the particle finite element method. Int J Numer Methods Eng 110(10):921–946
16. Cremonesi M, Meduri S, Perego U (2020) Lagrangian–Eulerian enforcement of non-homogeneous boundary conditions in the particle finite element method. Comput Part Mech 7:41–56
17. Peskin C (2002) The immersed boundary method. Acta Numer 112:479–517
18. Zhang L, Gerstenberger A, Wang X, Liu WK (2004) Immersed finite element method. Comput Methods Appl Mech Eng 193(21):2051–2067
19. Löhner R, Baum JD, Mestreau E, Sharov D, Charman C, Pelessone D (2004) Adaptive embedded unstructured grid methods. Int J Numer Methods Eng 60(3):641–660
20. Löhner R, Cebral J, Camelli F, Appanaboyina S, Baum J, Mestreau E, Soto O (2008) Adaptive embedded and immersed unstructured grid techniques. Comput Methods Appl Mech Eng 197(25):2173–2197
21. Foucard L, Vernerey F (2015) An X-FEM-based numerical-asymptotic expansion for simulating a Stokes flow near a sharp corner. Int J Numer Methods Eng 102(2):79–98
22. Schott B, Shahmiri S, Kruse R, Wall W (2016) A stabilized Nitsche-type extended embedding mesh approach for 3D low- and high-Reynolds-number flows. Int J Numer Methods Fluids 82(6):289–315
23. Main A, Scovazzi G (2018) The shifted boundary method for embedded domain computations. Part I: Poisson and Stokes problems. J Comput Phys 372:972–995
24. Main A, Scovazzi G (2018) The shifted boundary method for embedded domain computations. Part II: linear advection-diffusion and incompressible Navier–Stokes equations. J Comput Phys 372:996–1026
25. Colomés O, Main A, Nouveau L, Scovazzi G (2021) A weighted shifted boundary method for free surface flow problems. J Comput Phys 424:109837
26. Zorrilla R, Rossi R, Scovazzi G, Canuto C, Rodríguez-Ferran A (2024) A shifted boundary method based on extension operators. Comput Methods Appl Mech Eng 421:116782
27. Burman E, Claus S, Hansbo P, Larson MG, Massing A (2015) CutFEM: discretizing geometry and partial differential equations. Int J Numer Methods Eng 104(7):472–501
28. Osher S, Fedkiw R (2002) Level set methods and dynamic implicit surfaces. Springer, New York
29. Ryzhakov P, Oñate E, Rossi R, Idelsohn S (2012) Improving mass conservation in simulation of incompressible flows. Int J Numer Methods Eng 90(12):1435–1451
30. Franci A, Oñate E, Carbonell JM (2015) On the effect of the bulk tangent matrix in partitioned solution schemes for nearly incompressible fluids. Int J Numer Methods Eng 102(3–4):257–277

31. Nitsche J (1971) Über ein variationsprinzip zur lösung von Dirichlet-problemen bei verwendung von teilräumen, die keinen randbedingungen unterworfen sind. Abh Math Semin Univ Hambg 36(1):9–15
32. Juntunen M, Stenberg R (2009) Nitche's method for general boundary conditions. Math Comput 78(267):1353–1374
33. Massing A, Larson M, Logg A, Rognes M (2014) A stabilized Nitsche fictitious domain method for the Stokes problem. J Sci Comput 61(3):604–628
34. Massing A, Schott B, Wall W (2018) A stabilized Nitsche cut finite element method for the Oseen problem. Comput Methods Appl Mech Eng 328:262–300
35. Urquiza J, Garon A, Farinas M-I (2014) Weak imposition of the slip boundary condition on curved boundaries for Stokes flow. J Comput Phys 256:748–767
36. Winter M, Schott B, Massing A, Wall W (2018) A Nitsche cut finite element method for the Oseen problem with general Navier boundary conditions. Comput Methods Appl Mech Eng 330:220–252
37. Codina R, Baiges J (2009) Approximate imposition of boundary conditions in immersed boundary methods. Int J Numer Methods Eng 80(11):1379–1405
38. Baiges J, Codina R, Henke F, Shahmiri S, Wall WA (2012) A symmetric method for weakly imposing Dirichlet boundary conditions in embedded finite element meshes. Int J Numer Methods Eng 90(5):636–658
39. Brooks AN, Hughes TJ (1982) Streamline upwind/Petrov–Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations. Comput Methods Appl Mech Eng 32(1):199–259
40. Hughes TJ, Franca LP, Hulbert GM (1989) A new finite element formulation for computational fluid dynamics: VIII. The Galerkin/least-squares method for advective-diffusive equations. Comput Methods Appl Mech Eng 73(2):173–189
41. Hughes TJ, Feijóo GR, Mazzei L, Quincy J-B (1998) The variational multiscale method-a paradigm for computational mechanics. Comput Methods Appl Mech Eng 166(1):3–24
42. Bazilevs Y, Calo V, Cottrell J, Hughes T, Reali A, Scovazzi G (2007) Variational multiscale residual-based turbulence modeling for large eddy simulation of incompressible flows. Comput Methods Appl Mech Eng 197(1):173–201
43. Hughes TJR, Scovazzi G, Franca LP (2017) Multiscale and stabilized methods. Wiley, New York, pp 1–64
44. Codina R (2002) Stabilized finite element approximation of transient incompressible flows using orthogonal subscales. Comput Methods Appl Mech Eng 191(39):4295–4321
45. Codina R, Principe J, Guasch O, Badia S (2007) Time dependent subscales in the stabilized finite element approximation of incompressible flow problems. Comput Methods Appl Mech Eng 196(21):2413–2430
46. Codina R (2008) Analysis of a stabilized finite element approximation of the Oseen equations using orthogonal subscales. Appl Numer Math 58(3):264–283
47. Oñate E, Franci A, Carbonell JM (2014) Lagrangian formulation for finite element analysis of quasi-incompressible fluids with reduced mass losses. Int J Numer Methods Fluids 74(10):699–731
48. Oñate E (2000) A stabilized finite element method for incompressible viscous flows using a finite increment calculus formulation. Comput Methods Appl Mech Eng 182(3):355–370
49. Oñate E, Valls A, García J (2006) FIC/FEM formulation with matrix stabilizing terms for incompressible flows at low and high Reynolds numbers. Comput Mech 38(4):440–455
50. Cotela-Dalmau J, Rossi R, Oñate E (2017) A FIC-based stabilized finite element formulation for turbulent flows. Comput Methods Appl Mech Eng 315:607–631
51. Cornejo A, Franci A, Zárate F, Oñate E (2021) A fully Lagrangian formulation for fluid-structure interaction problems with free-surface flows and fracturing solids. Comput Struct 250:106532
52. Franci A, Zhang X (2018) 3D numerical simulation of free-surface Bingham fluids interacting with structures using the PFEM. J Nonnewton Fluid Mech 259:1–15
53. Cremonesi M, Franci A, Idelsohn S, Oñate E (2020) A state of the art review of the particle finite element method (PFEM). Arch Comput Methods Eng 27:1709–1735
54. Dadvand P, Rossi R, Oñate E (2010) An object-oriented environment for developing finite element codes for multi-disciplinary applications. Arch Comput Methods Eng 17(3):253–297
55. Dadvand P, Rossi R, Gil M, Martorell X, Cotela J, Juanpere E, Idelsohn S, Oñate E (2013) Migration of a generic multi-physics framework to HPC environments. Comput Fluids 80:301–309
56. Coll A, Ribó R, Pasenau M, Escolano E, Perez J, Melendo A, Monros A, Gárate J (2018) GiD v.14 Reference Manual
57. Coll A, Ribó R, Pasenau M, Escolano E, Perez J, Melendo A, Monros A, Gárate J (2018) GiD v.14 Customization Manual
58. Hesch C, Gil A, Arranz Carreño A, Bonet J (2012) On continuum immersed strategies for fluid-structure interaction. Comput Methods Appl Mech Eng 247–248:51–64
59. Happel J, Brenner H (1965) Low Reynolds number hydrodynamics. Prentice-Hall Inc., Englewood Cliffs
60. Huang Y, Zhang W, Mao W, Jin C (2011) Flow analysis of liquefied soils based on smoothed particle hydrodynamics. Nat Hazards 59:1547–1560
61. Della Vecchia G, Cremonesi M, Pisanò F (2019) On the rheological characterisation of liquefied sands through the dam-breaking test. Int J Numer Anal Methods Geomech 43(7):1410–1425
62. Koshizuka S, Oka Y (1996) Moving-particle semi-implicit method for fragmentation of incompressible fluid. Nucl Sci Eng 123(3):421–434