



# Enhanced semi-explicit particle finite element method via a modified Strang splitting operator for incompressible flows

Julio Marti<sup>1,2</sup> · Eugenio Oñate<sup>2</sup>

Received: 8 June 2022 / Revised: 30 August 2022 / Accepted: 15 September 2022 / Published online: 22 March 2023  
© The Author(s) under exclusive licence to OWZ 2023

## Abstract

This work presents an enhanced version of the semi-explicit particle finite element method for incompressible flow problems. This goal is achieved by improving the methodology that results from applying the Strang splitting operator by adding an acceleration term. The advective step is evaluated on the mesh considering the new term leading to a more efficient algorithm. Two test cases are solved for validating the methodology and estimating its accuracy. The numerical results demonstrate that the proposed scheme improves the accuracy and the computational efficiency of the semi-explicit PFEM scheme.

**Keywords** Incompressible Navier–Stokes · PFEM · Lagrangian · Strang splitting

## 1 Introduction

Traditionally the numerical simulations of fluid flows have been carried out with the Finite Element Method (FEM) using Eulerian formulations of the Navier–Stokes equations [1]. However, the continuous growth of computing power has led to more research into Lagrangian finite element models due to their intrinsic ability for tracking the domain deformations in fluid problems [2–6].

A particular class of Lagrangian methods developed by the authors is the so-called particle finite element method (PFEM) [6,7]. It has been successfully applied to the simulation of free-surface hydrodynamics [6,8], fluid–structure interaction [7,9–16], immiscible two-fluid flows [17–19], thermo-mechanical forming processes [20], among others. There also exist Lagrangian models similar to PFEM that have been applied to modeling material forming problems [4,21,22].

Although the traditional PFEM approach has been improved in recent years, one of its main drawbacks is that an element can be inverted during the iterative solution process

leading to solver failure. For this reason, all PFEM versions are equipped with a critical time estimator to avoid this situation. Although this alleviates considerably the restrictions due to the possibility of element inversion, the method remains computationally expensive, limiting its applicability for solving large practical 3D problems.

To overcome the mentioned shortcoming, in [23–25] the authors have developed semi-explicit PFEM approaches using the Lie–Trotter splitting operator [26–28]. In this kind of approaches, the Navier–Stokes equations are split in two parts, the advective one ( $\frac{d\mathbf{v}}{dt} = 0$ ) and the diffusive one (Stokes equation). The advective part is solved together with the additional equation  $\mathbf{v} = \frac{d\mathbf{x}}{dt}$ , considering that the computational mesh has a virtual (massless) particle associated to it, motion of which is obtained explicitly using a sub-stepping-based streamline integration. Once the particle advection is performed, the Stokes equations are solved implicitly on a new mesh. This approach has many advantages such as making it possible to use large time steps without incurring into stability restrictions. The main reason for this is that mesh nodes move during the explicit step only and elements do not deform during the implicit solution. This ensures convergent solutions without severe time step restrictions. However, the poor accuracy of the temporal approximations of the first-order splitting [26] limits the quality of the results. To mitigate this problem, in [29] Marti and Ryzhakov applied the second-order Strang operator [30,31] in such a way that the advective part of the equations is solved first. Then the diffusive part is solved, followed by a new solution of the

✉ Julio Marti  
julio.marti@cimne.upc.edu

<sup>1</sup> MECAMAT Group, Department of Engineering, Faculty of Science and Technology, Universitat de Vic-Universitat Central de Catalunya (UVic-UCC), 08500 Vic, Spain

<sup>2</sup> Centre Internacional de Mètodes Numèrics en Enginyeria (CIMNE), Gran Capitán s/n, 08034 Barcelona, Spain

advective equation. The two advective steps are solved using a Runge–Kutta-based streamline integration. Although the Strang-splitting-based PFEM approach [29] improves the accuracy and robustness of the method, the solution of the advective steps requires a constant particle velocity, while the particles are convected following the streamlines. Recently, the authors [32] have improved this approach allowing the update of the velocity of the particles as they are convected for cases when the inertial forces are high, as it occurs for a fluid contained in a spinning cylinder. The authors also have solved the advective steps using a SPH cubic spline kernel-based approach. The numerical results have confirmed that the enhanced PFEM scheme improves the accuracy of previous semi-explicit versions, but the necessity to check for the particle neighbors several times leads to a more expensive solver from the computational point of view. In addition, the kernel used is not able to correctly represent a linear velocity field.

This work aims to improving the PFEM Strang-based strategy developed in [32] by removing the drawbacks mentioned in the above lines. The key feature of the new method consist in evaluating the inertial (acceleration) term which appears in both the convective and diffusive steps, using a mesh-based procedure.

The organization of the paper is as follows. Section 2 presents the governing Navier–Stokes equations. Also, the new Strang splitting methodology is introduced. Section 3 briefly explains the discrete form of the split equations. Section 4 presents the test cases that have been used to confirm the desired convergence. Finally, Sect. 5 presents the conclusions of this work.

## 2 Governing equations at a continuum level

Let  $\Omega^t$  denote a domain containing a viscous incompressible fluid with a fixed boundary  $\Gamma_D$  and a free surface  $\Gamma_N$ . The evolution of the velocity vector  $\mathbf{v} = \mathbf{v}(\mathbf{x}, t)$  and the pressure  $p = p(\mathbf{x}, t)$  is governed by the Navier–Stokes equations given by

– Momentum

$$\rho \frac{\partial \mathbf{v}}{\partial t} + \rho \mathbf{v} \cdot \nabla \mathbf{v} = -\nabla p + \nabla \cdot (2\mu \nabla^s \mathbf{v}) + \rho \mathbf{b} \quad (1)$$

– Incompressibility

$$\nabla \cdot \mathbf{v} = 0 \quad (2)$$

where  $\mu$  is the dynamic viscosity,  $\rho$  is the density,  $\mathbf{b}$  is the body force vector,  $\nabla$  is the gradient operator, and  $\nabla^s$  its symmetric part. Note that the term  $\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v}$  corresponds to the material derivative of the velocity  $\frac{D\mathbf{v}}{Dt}$ . Equations 1 and 2 are

completed with the standard Neumann and Dirichlet boundary conditions at the boundaries  $\Gamma_N$  and  $\Gamma_D$ , respectively,

$$-\rho \mathbf{n} + 2\mu \nabla^s \mathbf{v} \cdot \mathbf{n} = \bar{\mathbf{t}} \quad \text{on } \Gamma_N \quad (3)$$

$$\mathbf{v} = \bar{\mathbf{v}} \quad \text{on } \Gamma_D \quad (4)$$

## 3 Strang operator splitting for the PFEM

Adding and subtracting to the momentum equation a new variable acceleration  $\mathbf{a}$ , which approximation is defined below, and applying the Strang splitting operator [29,33], the unknown solution  $\mathbf{v}$  at  $t^n + \delta t$  of Eq. 1 is obtained in three steps as follows:

– Step 1 (advection 1)

$$\begin{aligned} \rho \frac{\partial \mathbf{v}^*}{\partial t} + \rho \mathbf{v}^* \cdot \nabla \mathbf{v}^* &= \rho \mathbf{a} \quad \text{with} \\ \mathbf{v}^*(t^n, \mathbf{x}) &= \mathbf{v}(t^n, \mathbf{x}) \quad t \in [t^n, t^n + \delta t/2] \end{aligned} \quad (5)$$

– Step 2 (diffusion)

$$\begin{aligned} \rho \frac{\partial \mathbf{v}}{\partial t} &= -\nabla p + \nabla \cdot (2\mu \nabla^s \mathbf{v}) + \rho \mathbf{b} - \rho \mathbf{a} \quad \text{with} \\ \mathbf{v}(t^n, \mathbf{x}) &= \mathbf{v}^*(t^n + \delta t/2, \mathbf{x}) \quad t \in [t^n, t^n + \delta t] \end{aligned} \quad (6)$$

– Step 3 (advection 2)

$$\begin{aligned} \rho \frac{\partial \hat{\mathbf{v}}}{\partial t} + \rho \hat{\mathbf{v}} \cdot \nabla \hat{\mathbf{v}} &= \rho \mathbf{a} \quad \text{with} \\ \hat{\mathbf{v}}(t^n + \delta t/2, \mathbf{x}) &= \mathbf{v}(t^n + \delta t, \mathbf{x}) \quad t \in [t^n + \delta t/2, t^n + \delta t] \end{aligned} \quad (7)$$

Equation 5 is solved for the fractional velocity  $\mathbf{v}^*$  for half of the time increment  $\delta t/2$ . The solution of Eq. 5 is used as the initial condition for solving Eq. 6 for the velocities  $\mathbf{v}$  and the pressure  $p$  for a full time increment  $\delta t$ , as explained in Sect. 4.2. The velocity field obtained from Eq. 6 is the initial condition for solving Eq. 7 for the final velocity field for a half time increment,  $\delta t/2$ .

Note that the original form of the Strang operator splitting for the PFEM developed in [29] is recovered by making  $\mathbf{a} = \mathbf{0}$ .

## 4 Numerical solution

In the semi-explicit versions of the PFEM [23–25,29,34], the mesh remains fixed and the advective step is solved considering virtual particles. The particle positions  $\mathbf{x}_p$  only coincide with the nodes of the mesh  $\mathbf{x}_m$  at the beginning of the advective step and are moved following the streamlines of the flow using the last known velocity. Thus, there is no possibility

that an element inverts due the fact that the mesh movement does not depend on the unknown velocity. Once the particles are moved to the new position, a new mesh is generated and is used for solving the implicit step (Sect. 4.2). In the following, the solution of the convective and diffusive parts of the algorithm is described.

### 4.1 Solution of the advective parts

As mentioned above, the convective part is solved in two steps. The first one goes from  $t^n$  to  $t^{n+1/2}$ , and the second one goes from  $t^{n+1/2}$  to  $t^{n+1}$ . The methodology used to solve the first of them is presented below.

The velocity and the position at time  $t^{n+1/2}$  for a particle  $p$  located at  $\mathbf{x}_p$  is obtained by integrating Eq. 5 and the additional equation  $\frac{D\mathbf{x}}{Dt} = \mathbf{v}^*$  from  $t^n$  to  $t^{n+1/2}$  as

$$\mathbf{v}^{*,n+1/2}(\mathbf{x}_p^{n+1/2}) = \mathbf{v}_p^n + \int_n^{n+1/2} \mathbf{a}^t(\mathbf{x}_p^t) dt \tag{8}$$

along with

$$\mathbf{x}_p^{n+1/2} = \mathbf{x}_p^n + \int_n^{n+1/2} \mathbf{v}^{*,t}(\mathbf{x}_p^t) dt \tag{9}$$

where  $\mathbf{x}_p^t$  is the function describing the movement of the particle from its position at  $t^n$  to that at  $t^{n+1/2}$  ( $t : n < t < n + 1/2$ ). Note that Eqs. 8 and 9 are not only strongly coupled, but also highly nonlinear, as the computation of the position depends on the current velocity. In previous versions of the PFEM, this system is solved iterative limiting the stability of the scheme, due to the possibility of element inversion.

In order to avoid the iterative solutions of Eqs. 8 and 9, the X-IVS technique was proposed by Idelsohn et al. [25]. It is based on the idea of moving the particles (nodes) following the streamlines of the flow using the last known velocity field. As the particles are moved explicitly, the problem becomes a linear one. The new configuration is used to solve the momentum equation. In this work, this idea is also used to approximate the position and velocity field at  $t^{n+1/2}$ . Applying the X-IVS technique to Eqs. 8 and 9, the terms  $\mathbf{a}^t$  and  $\mathbf{v}^{*,t}$  are replaced by  $\mathbf{a}^n$  and  $\mathbf{v}^n$ . According to this, the final set of equations read:

$$\mathbf{x}_p^{n+1/2} = \mathbf{x}_p^n + \int_n^{n+1/2} \mathbf{v}^n(\mathbf{x}_p^t) dt \tag{10}$$

$$\mathbf{v}_p^{*,n+1/2} = \mathbf{v}_p^n + \int_n^{n+1/2} \mathbf{a}^n(\mathbf{x}_p^t) dt \tag{11}$$

Equations 10 and 11 are explicit. Note that Eq. 11 modifies the particle velocity while it moves. Thus the time integration

of the velocities and the position are improved, as we will show in Example 1.

Both the position (Eq. 10) and the velocities (Eq. 11) of the particles are computed by applying the following 4th order Runge–Kutta (RK4) scheme:

$$\mathbf{x}_p^{n+1/2} = \mathbf{x}_p^n + \frac{(\delta t/2)}{6} (\mathbf{v}_1 + 2\mathbf{v}_2 + 2\mathbf{v}_3 + \mathbf{v}_4) \tag{12}$$

$$\mathbf{v}_p^{*,n+1/2} = \mathbf{v}_p^n + \frac{(\delta t/2)}{6} (\mathbf{a}_1 + 2\mathbf{a}_2 + 2\mathbf{a}_3 + \mathbf{a}_4) \tag{13}$$

where  $\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3$  and  $\mathbf{v}_4$  are the intermediate velocities and  $\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3$  and  $\mathbf{a}_4$  are the intermediate accelerations. The algorithm to evaluate  $\mathbf{x}_p^{n+1/2}$  and  $\mathbf{v}_p^{n+1/2}$  is presented in Algorithm 1.

---

**Algorithm 1** Evaluation of the particle position  $\mathbf{x}_p^{n+1/2}$  and velocity  $\mathbf{v}^{n+1/2}$ .

---

**for** each virtual particle  
 At the beginning  $\mathbf{x}_{p1} (= \mathbf{x}_p^n) = \mathbf{x}_m$  and velocity  $\mathbf{v}_1$ ;  
 Move nodes according to  $\mathbf{x}_{p2} = \mathbf{x}_p^n + \frac{\delta t}{4} \mathbf{v}_1$ ;  
 Evaluate the velocity as  $\mathbf{v}_2 = \sum_{j=1}^3 \mathbf{N}_j \mathbf{v}_j^n$  and the acceleration as  $\mathbf{a}_2 = \sum_{j=1}^3 \mathbf{N}_j \mathbf{a}_j^n$ ;  
 Save  $\mathbf{v}_2$  and  $\mathbf{a}_2$ ;  
 Set back to its original location  $\mathbf{x}_p = \mathbf{x}_m$ ;  
 Move nodes according to  $\mathbf{x}_{p3} = \mathbf{x}_p^n + \frac{\delta t}{4} \mathbf{v}_2$ ;  
 Evaluate the velocity as  $\mathbf{v}_3 = \sum_{j=1}^3 \mathbf{N}_j \mathbf{v}_j^n$  and the acceleration as  $\mathbf{a}_3 = \sum_{j=1}^3 \mathbf{N}_j \mathbf{a}_j^n$ ;  
 Save  $\mathbf{v}_3$  and  $\mathbf{a}_3$ ;  
 Set back to its original location  $\mathbf{x}_p = \mathbf{x}_m$ ;  
 Move nodes according to  $\mathbf{x}_{p4} = \mathbf{x}_p^n + \frac{\delta t}{2} \mathbf{v}_3$ ;  
 Evaluate the velocity as  $\mathbf{v}_4 = \sum_{j=1}^3 \mathbf{N}_j \mathbf{v}_j^n$  and the acceleration as  $\mathbf{a}_4 = \sum_{j=1}^3 \mathbf{N}_j \mathbf{a}_j^n$ ;  
 Save  $\mathbf{v}_4$  and  $\mathbf{a}_4$ ;  
 Set back to its original location  $\mathbf{x}_p = \mathbf{x}_m$ ;  
 Move the mesh nodes according to Eq. 12 and with a velocity defined by Eq. 13;  
 where  $\mathbf{N}_j$  are the shape functions of the element where the particle lays.

---

Once the particle advection is performed, the particles are connected by a new mesh generated with the Delaunay method [35] or the extended Delaunay tessellation procedure [36], equipped with the alpha-shape technique [37] for identifying the external boundaries. The mesh regeneration step is standard for all PFEM schemes and is not detailed here. The reader is referred to [7] for the standard implementation.

Similarly to other semi-explicit PFEM procedures, the proposed method is free of the advection stability issue. Namely, by performing an explicit advective step and then solving the Stokes equations on the fixed mesh, the stability restrictions induced by the CFL-number ( $CFL < 1$ ) are removed. Nevertheless, in comparison with the previous version of the PFEM, the current strategy improves the accuracy

by considering the acceleration in the advective motion, as it will be shown in the examples.

Once the Stokes problem is solved, the explicit mesh motion step is repeated following the procedure described in Algorithm 1, but now applied from  $t^{n+1/2}$  to  $t^{n+1}$ . Hence, the upper indices  $n$  and  $n+1/2$  are replaced by  $n+1/2$  and  $n$ , respectively. Note that this second movement of the nodes could cause that some element of the mesh is inverted. Element inversion results in a negative Jacobian [38] of the corresponding element [39]. This leads to the failure of the simulation. In order to guarantee the mesh quality and diminish the amount of highly distorted elements, the Jacobian of the corresponding elements is evaluated, and if any of them is negative, a second remeshing is performed (See Algorithm 2). This considerably improves mesh quality, to the expense in a increase in the computational cost. In any case, the overall solver is still cheaper than the strategy presented in [32]. For the simulated examples in this work, it was not necessary to carry out a second remeshing.

### 4.2 Solution of the diffusive part

The Stokes problem (Eq. 6) is solved on the domain  $\Omega^{n+1/2}$  defined by the position of the particles  $\mathbf{x}_p^{n+1/2}$  obtained by solving the first advective step (Eq. 12). The particles velocity  $\mathbf{v}_p^{*,n+1/2}$  from Eq. 13 is used as the initial condition  $\mathbf{v}^n = \mathbf{v}^{*,n+1/2}$  for solving Eq. 6.

In order to obtain the velocities  $\mathbf{v}$  and the pressure  $p$  for a full time increment  $\delta t$ , the weak form of Eq. 6, following the standard FEM, the Crank–Nicolson time integration scheme and the fractional step method [40–42] can be written as

– Fractional velocity

$$\begin{aligned} & \sum_{\text{elem}} \left[ \int_{\Omega_e} \rho \mathbf{N} \cdot \tilde{\mathbf{v}}^{n+1} d\Omega + \delta t \int_{\Omega_e} \nabla \mathbf{N} : \mu \nabla^s \tilde{\mathbf{v}}^{n+1} d\Omega \right. \\ & = \int_{\Omega_e} \rho \mathbf{N} \cdot \mathbf{v}^n d\Omega + \delta t \int_{\Omega_e} \nabla \mathbf{N} : p^n d\Omega \\ & + \delta t \int_{\Omega_e} \nabla \mathbf{N} : \mu \nabla^s \mathbf{v}^n d\Omega + \delta t \int_{\Omega_e} \rho \mathbf{N} \cdot \mathbf{b} d\Omega \\ & \left. - \delta t \int_{\Omega_e} \rho \mathbf{N} \cdot \mathbf{a}^n d\Omega + \int_{\Gamma_{Ne}} \mathbf{N} \cdot \bar{\mathbf{t}} d\Gamma \right] \end{aligned} \quad (14)$$

– Computation of the pressure

$$\begin{aligned} & \sum_{\text{elem}} \left[ \frac{\delta t}{2} \int_{\Omega_e} \nabla N \cdot \nabla p^{n+1} d\Omega \right. \\ & = \left. \frac{\delta t}{2} \int_{\Omega_e} \nabla N \cdot \nabla p^n d\Omega + \int_{\Omega_e} \rho N \nabla \cdot \tilde{\mathbf{v}} d\Omega \right] \end{aligned} \quad (15)$$

– Velocity update

$$\begin{aligned} & \sum_{\text{elem}} \left[ \int_{\Omega_e} \rho \mathbf{N} \cdot \mathbf{v}^{n+1} d\Omega = \int_{\Omega_e} \rho \mathbf{N} \cdot \tilde{\mathbf{v}} d\Omega \right. \\ & \left. + \frac{\delta t}{2} \int_{\Omega_e} \nabla \mathbf{N} : (p^{n+1} - p^n) d\Omega \right] \end{aligned} \quad (16)$$

where  $\tilde{\mathbf{v}}$  is the intermediate velocity vector.

The velocities, the accelerations and the pressure are approximated over each element with  $n$  nodes using standard linear elements as

$$\mathbf{v}(\mathbf{x}) = \sum_{l=1}^n \mathbf{N}_l(\mathbf{x}, t) \mathbf{v}_l = \mathbf{N}^T(\mathbf{x}) \begin{pmatrix} v_{1x} \\ v_{1y} \\ v_{2x} \\ v_{2y} \\ v_{3x} \\ v_{3y} \end{pmatrix} \quad (17)$$

$$\mathbf{a}(\mathbf{x}) = \sum_{l=1}^n \mathbf{N}_l(\mathbf{x}) \mathbf{a}_l = \mathbf{N}^T(\mathbf{x}) \begin{pmatrix} a_{1x} \\ a_{1y} \\ a_{2x} \\ a_{2y} \\ a_{3x} \\ a_{3y} \end{pmatrix} \quad (18)$$

$$p(\mathbf{x}) = \sum_{l=1}^n N_l(\mathbf{x}) p_l = \mathbf{N}_p^T(\mathbf{x}) \begin{pmatrix} p_1 \\ p_2 \\ p_3 \end{pmatrix} \quad (19)$$

where

$$\mathbf{N} = \begin{pmatrix} N_1 & 0 & N_2 & 0 & N_3 & 0 \\ 0 & N_1 & 0 & N_2 & 0 & N_3 \end{pmatrix} \quad (20)$$

$$\mathbf{N}_p = (N_1 \ N_2 \ N_3) \quad (21)$$

$N_l$  are the standard linear FE shape functions and  $l$  stands for the nodal index.

Substituting the finite element approximations into Eqs. 14–16 leads to the following discrete form of Eqs. 14–16.

$$(\mathbf{M} + \delta t \mathbf{K}) \tilde{\mathbf{v}} = (\mathbf{M} + \delta t \mathbf{K}) \mathbf{v}^n + \delta t \mathbf{G} p^n + \delta t \mathbf{F} \mathbf{b} - \delta t \mathbf{M} \mathbf{a}^n \quad (22)$$

$$\frac{\delta t}{2} \mathbf{L} p^{n+1} = \frac{\delta t}{2} \mathbf{L} p^n - \mathbf{D} \tilde{\mathbf{v}} \quad (23)$$

$$\mathbf{v}^{n+1} = \tilde{\mathbf{v}} + \frac{\delta t}{2} \mathbf{M}^{-1} \mathbf{G} (p^{n+1} - p^n) \quad (24)$$

Equation 23 is stabilized using the ASGS stabilization technique for the pressure components [43]. For the sake of simplicity, the stabilization terms are omitted here. They can be consulted in [44]. Other stabilization techniques for the pressure equation can be used [45,46].

*Evaluation of the acceleration term a.*

In the semi-explicit PFEM versions, the advective part ( $\frac{d\mathbf{v}}{dt} = \frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v} = \mathbf{0}$ ) of the Navier–Stokes equations is solved together with the additional equation  $\mathbf{v} = \frac{d\mathbf{x}}{dt}$  using the streamlines. The convection of the particles depends on the velocity at every moment, while their velocities remain unchanged. This assumption is valid only when the inertial forces are negligible.

In order to generalize the formulation to problems where the inertial forces are not negligible, i.e.  $\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v} \neq 0$  an acceleration term is added consistently in the advective terms (see Eqs. 5 and 7) and the Stokes equation (Eq. 6). With this new term in the advective part, the particles change their positions following the streamlines Eq. 12, while the acceleration streamlines are used to modify the velocity Eq. 13. Next it is explained how the acceleration term is evaluated.

The acceleration  $\mathbf{a}$  ( $\mathbf{a} = \frac{\partial \mathbf{v}}{\partial t} + \mathbf{c} \cdot \nabla \mathbf{v}$ ) is computed by the following finite element discrete form

$$\mathbf{M}\mathbf{a} = \mathbf{M} \frac{\partial \mathbf{v}}{\partial t} + \mathbf{C}\mathbf{v} \tag{25}$$

where  $\mathbf{M}$  is the mass matrix and  $\mathbf{C}$  is given by Eq. 26.

$$\mathbf{C} = \rho \sum_{\text{elem}} \int_{\Omega_e} \mathbf{N} \cdot \mathbf{c} \cdot \nabla \mathbf{N}^T d\Omega \tag{26}$$

$\mathbf{c}$  is the convective velocity evaluated at the Gauss point as  $\mathbf{N}^T(\mathbf{x})\mathbf{v}$ .

According to Sect. 4.1, the acceleration term  $\mathbf{a}$  (Eq. 25) is evaluated at time  $t^n$  and it is assumed to be constant during the time step (see Algorithm 2). Therefore, Eq. 25 can be written as

$$\mathbf{M}\mathbf{a}^n = \mathbf{C}\mathbf{v}^n \tag{27}$$

The acceleration vector  $\mathbf{a}$  is obtained by using a lumped form of the mass matrix  $\mathbf{M}$ . Thus, the solution of Eq. 27 has a negligible computational cost. Equation 27 is also stabilized using the ASGS stabilization technique [43].

The explicit mesh motion step is repeated following the procedure described in Algorithm 1, but now applied from  $t^{n+1/2}$  to  $t^{n+1}$ . Hence, the upper indices  $n$  and  $n+1/2$  are replaced by  $n+1/2$  and  $n$ , respectively.

### 5 Overall solution strategy

The solution procedure can be summarized as: Given the nodal positions, the velocity and the pressure at time  $t^n$  find these variables at  $t^{n+1}$ . The solution steps are outlined in Algorithm 2.

---

### Algorithm 2

Solution algorithm for the simulation of incompressible flows.

---

```

for  $t = t^{n+1}$ 
  Calculate the acceleration  $\mathbf{a}^n$  by solving Eq. 27.
  Evaluation of the particle position  $\mathbf{x}_p^{n+1/2}$  and velocity  $\mathbf{v}^{n+1/2}$  following Algorithm 1
  Re-mesh the fluid domain
  Solve the Stokes problem (Eq. 6) uncoupling the velocity and pressure via the fractional step technique:
    – Find the intermediate velocity  $\tilde{\mathbf{v}}$  by solving Eq. 22 ;
    – Solve the Poisson’s equation for the pressure  $p^{n+1}$  (Eq. 23);
    – Correct the velocity by solving Eq. 24 to obtain a divergence-free solution with  $\mathbf{v}^{n+1} = \tilde{\mathbf{v}}$  on  $\Gamma_D$ ;
  Evaluation of the particle position  $\mathbf{x}_p^{n+1}$  and velocity  $\mathbf{v}^{n+1}$  following Algorithm 1
  if the mesh is too distorted Re-mesh the fluid domain
  
```

---

## 6 Numerical examples

The new PFEM scheme has been implemented in the Kratos MultiPhysics code, an academic Open Source software [47]. In the following section, two numerical examples are presented that show the advantage of the new model. The first one is a 2D benchmark of a flow in a spinning cylindrical container. The second example is the well-known flow around a circular cylinder.

### 6.1 Viscous flow in a spinning circular container

The test presented in [32] is chosen here to show how the additional acceleration term improves the numerical solution, and secondly to evaluate the convergence of the modified approach. In the test, a cylinder of radius  $Re = 0.5$  [m] is filled up with a fluid rotates about it axis with a constant angular velocity  $w$  in the absence of gravity forces. As a result of the centrifugal forces, the fluid presents a circular internal free surface of radius  $Ri = 0.21$  [m]. The geometry, dimensions and boundary conditions are shown in Fig. 1.

The analytical solution (polar coordinates) of this problem is given by

$$\mathbf{v}_\theta = w\mathbf{r} \tag{28}$$

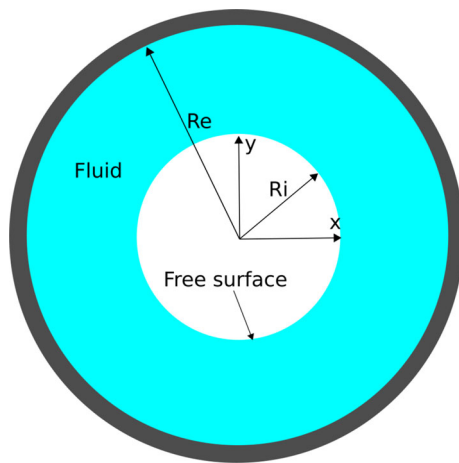
$$p = \frac{\rho w^2}{2}(r^2 - Ri^2) \tag{29}$$

The fluid has a density of 1000 [kg/m<sup>3</sup>] and a viscosity of 0.001 [kg/ms].

To simulate numerically this test, the velocity is prescribed to  $\mathbf{v}_\theta = Re$  at  $Re = 0.5$  [m] and the pressure is fixed to zero at  $Ri = 0.21$  [m].

In this work the non-homogeneous boundary condition is imposed by employing a mixed Lagrangian–Eulerian (LE) technique presented in [48]. This technique is characterized by treating only the solid nodes as Eulerian (i.e., fixed). In





**Fig. 1** Geometry

other words, solid nodes are not changing their position along time although its velocity can be different from zero without altering the geometric definition of the boundary due to a Lagrangian motion. As a consequence of this, a convective velocity appears in the element containing these nodes and its discrete form is given by  $\mathbf{c}(\mathbf{x}) = \mathbf{N}^T(\mathbf{x})(\mathbf{v} - \mathbf{v}^L)$  where  $\mathbf{v}$  and  $\mathbf{v}^L$  are the Eulerian and Lagrangian velocities. For the fluid particles  $\mathbf{v} = \mathbf{v}^L$ , while for the solid ones  $\mathbf{v}^L = 0$ . Its effect is introduced by adding an additional matrix  $\mathbf{C}_{LE} = \rho \sum_{\text{elem}} \int_{\Omega_e} \mathbf{N} \cdot \mathbf{c} \cdot \nabla \mathbf{N}^T d\Omega$  to the left-hand side of Eq. 22. Note that while the matrix  $\mathbf{C}$  given by Eq. 26 is evaluated in the entire domain in order to obtain the acceleration  $\mathbf{a}$  and appear on the right-hand side of Eq. 22, the matrix resulting from the LE technique appears only on elements with Eulerian nodes [48], representing a small portion of the overall domain. In the case that the solid nodes have zero velocity, this matrix does not intervene in the formulation.

To show the effect of the acceleration term in the evaluation of the particle position and its velocity in the new method, the domain was discretized with a non-structured mesh ( $h = 0.01$  [m]). The example was solved over a time span of 0.5 [s], and the time increment was set to 0.01 [s]. The results are compared with the ones obtained using the numerical formulation presented in [29], which from now on will be distinguished in the paper, as “without acceleration”. Figure 2 shows the trajectory of two particles. The first particle is located near the wall ( $x = 0.0$  [m] and  $y = -0.355863$  [m]), while the second one in the middle of the domain ( $x = 0.$  [m] and  $y = -0.485586$  [m]). The reference solution is obtained by integrating the velocity (Eq. 28) with the Runge–Kutta scheme (Fig. 2).

Figure 2b and d shows the trajectory of the two particles selected. It can be observed how the particle position without considering the acceleration term [29] deviate from the reference solution. However, as will be seen later, this error decreases over time.

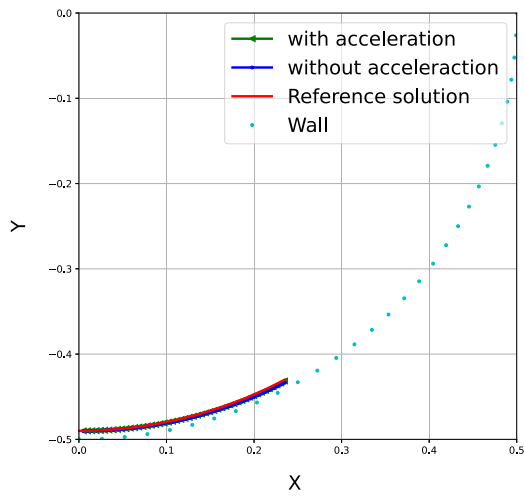
The modulus of the velocity field at time  $t = 0.5$  [s] is shown in Fig. 3. Note that flattened elements appear close to the wall in the approach without including the acceleration term (Fig. 3). This effect is noticeable by inspecting the right part of Fig. 3. We note that in this example no particles were seed or removed during the simulation to improve the mesh quality.

The time accuracy of the new approach was obtained by running the example for the following time steps: 0.01, 0.005, 0.0025, 0.0008 and 0.0005 [s]. The error measure was estimated by the area enclosed between the obtained and the reference radial velocity distribution along the horizontal cut plane at  $y = 0$  [m] from  $x = R_i$  to  $x = R_e$  at  $t = 0.5$  [s] for the given time step. The influence of the spatial discretization error was excluded by considering as a reference solution the one obtained using  $dt = 0.0005$  [s]. The convergence data (error versus time step size in logarithmic scale) is shown in Fig. 4a. One can observe that the approach without the acceleration term presents the same slope but a larger error than the version considering the acceleration. Also the strategy presented here shows the same spatial convergence than the strategy presented in [32], where the convective steps are solved using a SPH kernel.

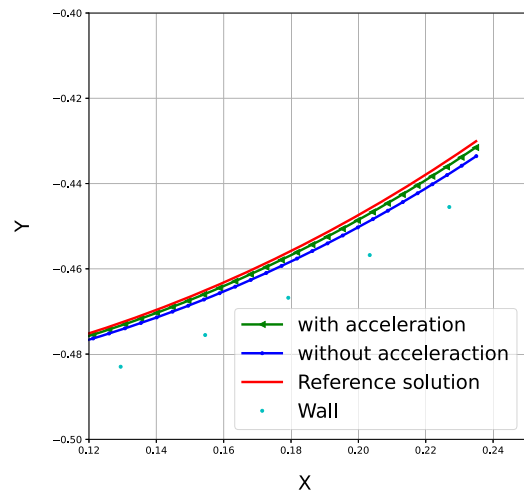
To check the spatial accuracy of the new PFEM technique, the example was solved with a small time step ( $dt = 0.0005$  [s]) and different mesh sizes of  $h = 0.05$  [m],  $h = 0.015$  [m] and  $h = 0.008$  [m]. The analytical solution was taken as the reference solution. Figure 4b shows the error versus the mesh size. The method exhibits second-order convergence with respect to the mesh size. Also the error with the SPH kernel methodology presented in [32] is displayed. Marti and Oñate [32] presented an enhanced version of the semi-explicit PFEM which was obtained by applying the Strang operator splitting to the Navier–Stokes equations and solving the advective steps using a SPH kernel.

Table 1 shows the average computational times of the element-based and kernel-based [32] solutions corresponding to the two advective steps on a given mesh. The time required to generate the mesh was 0.06139655 [s], which is the step that cannot be easily parallelized. One can see that the element-based implementation is approximately five times “cheaper” than the SPH kernel one. Parallel implementation leads to a reduction in the computational times, which is, however, lower than the ideal are. On the other hand, the new PFEM strategy presented has second-order accuracy in time and in space, with a slightly larger error than the one of [32] even though this uses a kernel that not correctly represent a linear field, however the new PFEM strategy is computationally more efficient and, thus, best suited for practical applications.

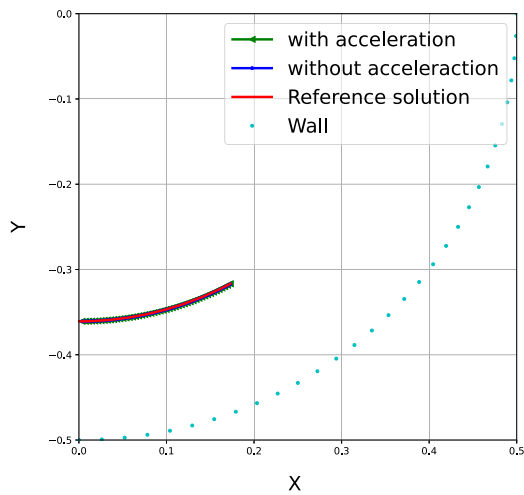
The same example was run for 2.5 [s] to show the importance of the acceleration term.



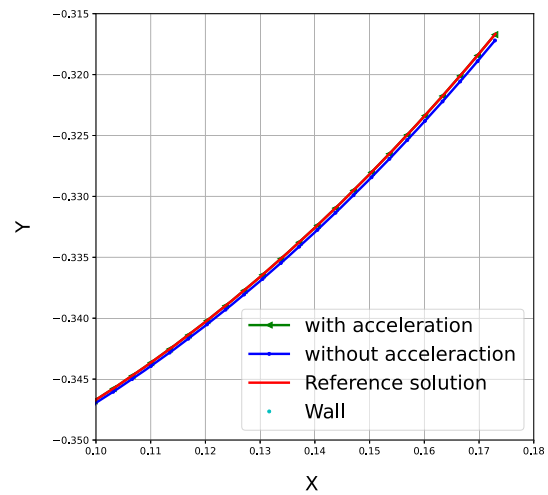
(a) First particle: trajectory



(b) First particle: zoom of the last positions



(c) Second particle: trajectory



(d) Second particle: zoom of the last positions

Fig. 2 Particle trajectories

**Fig. 3** Velocity field at  $t = 0.5$  [s]

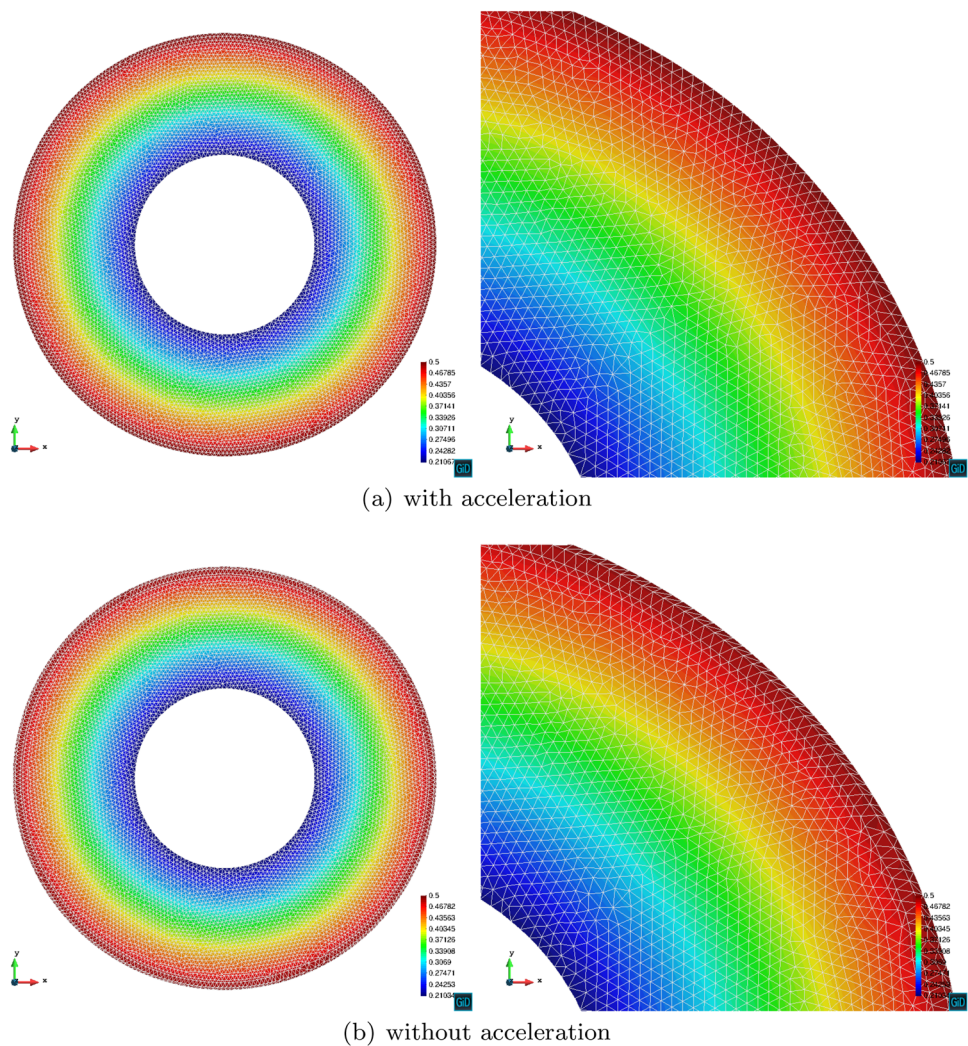


Figure 5 shows how the particle position without considering the acceleration term deviate from the reference solution, leading to failure of the solver. The former is presented in more detail in Figs. 6 and 7.

## 6.2 Flow around a circular cylinder.

This test has been chosen to show the capability of the new PFEM scheme for solving complex problems. According to Fig. 8, a circle with a diameter of 1 [m] has been placed inside a rectangular domain of 21 [m] width and 11 [m] height. The velocity has been set to  $v_x = 1.0$  [m/s] in the inlet and the bottom and upper walls are slip. The Dirichlet boundary condition has been imposed by employing the LE technique presented before. The pressure is fixed to zero in the outlet. The density and viscosity are  $\rho = 1.0$  [kg/m<sup>3</sup>] and  $\mu = 0.001$  [kg/(ms)], respectively. The geometry and the material properties are taken from [25].

The domain is discretized with an unstructured mesh of 3-nodes triangles where the minimum and the maximum mesh

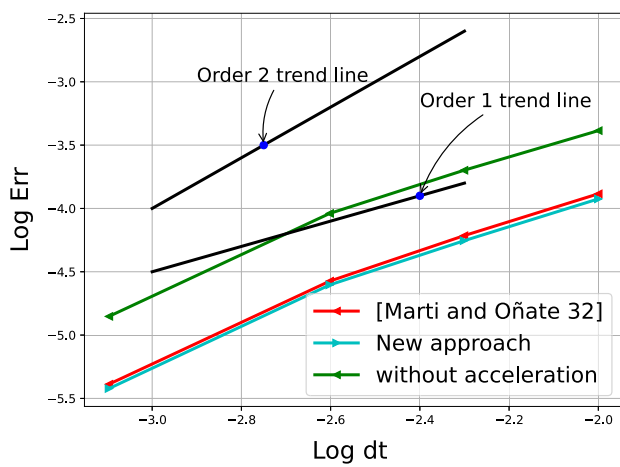
sizes were:  $h_{\min} = 0.0073$  [m] and  $h_{\max} = 0.2$  [m], respectively. In this, test particles were seed or removed during the simulation to further improve the mesh quality.

The example was run using a time step of 0.0025 [s].

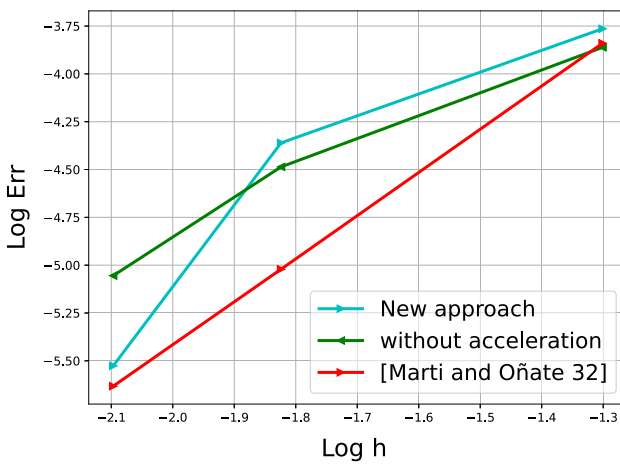
Figure 9 shows the velocity and the pressure fields at 80 [s]. Figures 10 and 11 compares the drag and the lift with the ones using the first-order version of the PFEM-2 approach [25]. Note that in [25] the authors only reported the periodic part of the solution without mentioning when it started. Also the drag and the lift obtained by Mittal [49] are introduced.

We can observed that Figs. 10 and 11 show a good agreement in the oscillation frequency and amplitude with the results from the semi-explicit PFEM approach with fixed mesh [25]. From the graph, the drag mean value is 8% above the reference mean value and the lift force oscillates with an amplitude is 13% above the reference value. These values are slightly smaller than the ones presented in [25]. However, the approach presented here obtain practically the same results (lift and drag) than [25] without the need to solve the convective term with an excessive number of particles.





(a) Error vs time step size.

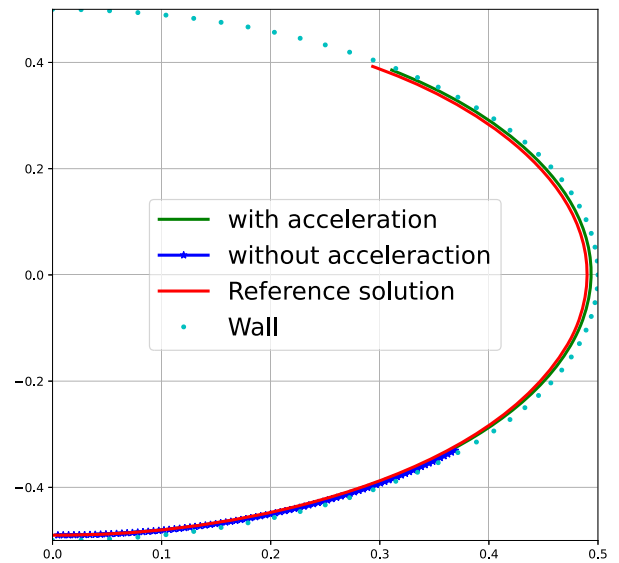


(b) Error vs different mesh sizes (h).

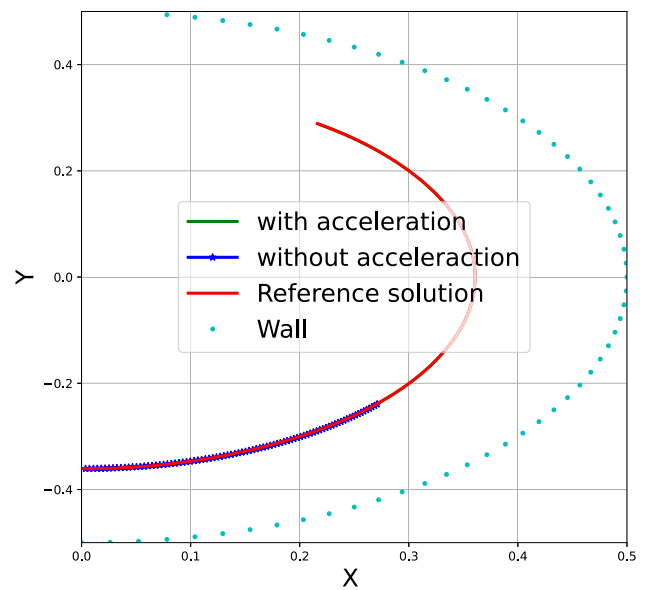
**Fig. 4** Flow in spinning circular container

**Table 1** Computational cost of different solution steps of the proposed PFEM strategy

Advection step	Sequential (s)	4 Cores (s)
SPH kernel	0.217965	0.1130016
Mesh based	0.03907245	0.0265653



(a) First particle.



(b) Second particle.

**Fig. 5** Particle trajectories

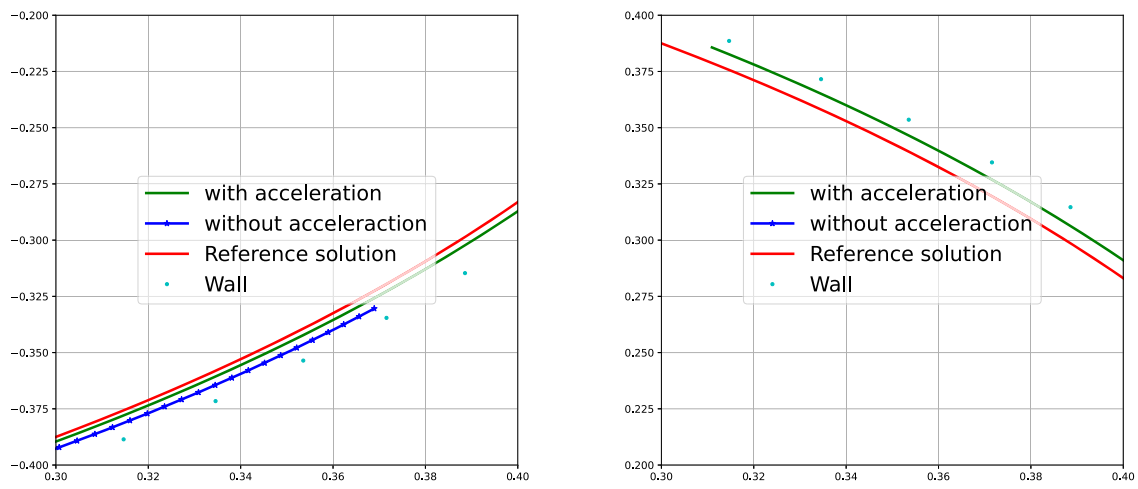


Fig. 6 Zoom of the last positions for first particle

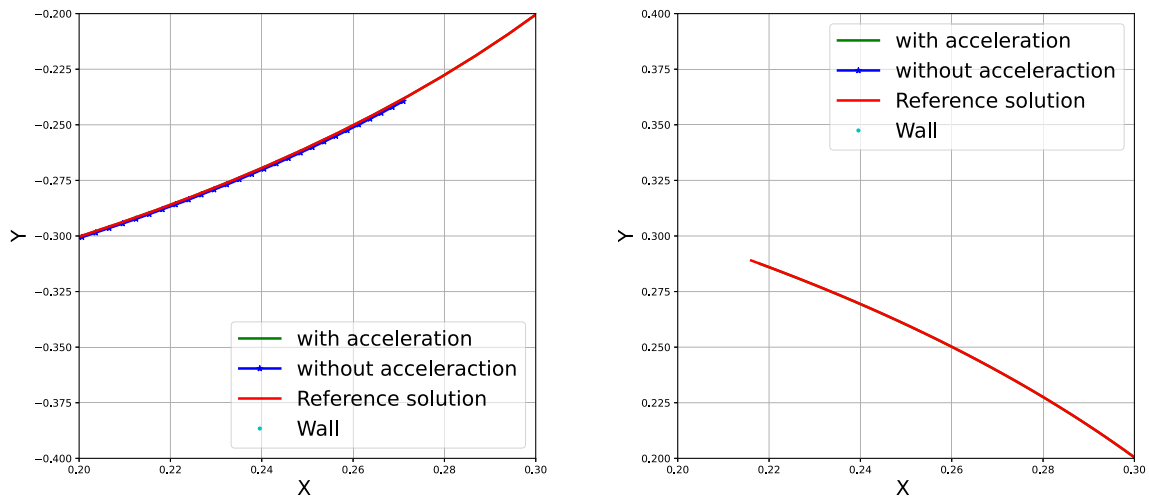


Fig. 7 Zoom of the last positions for second particle

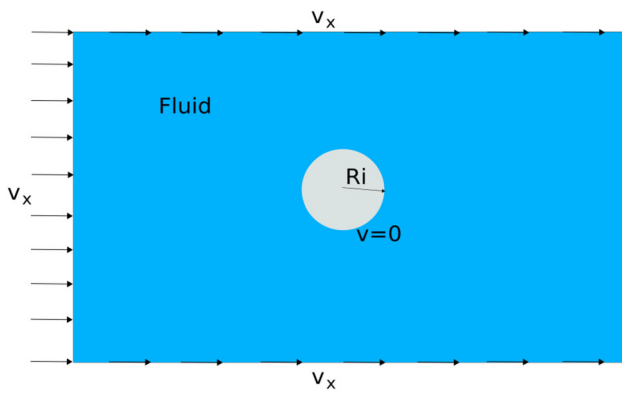
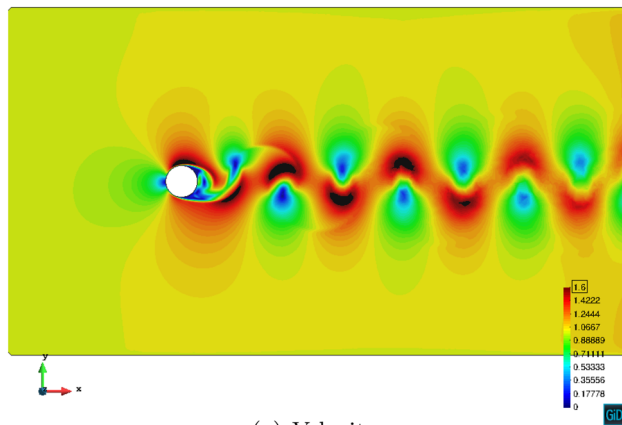
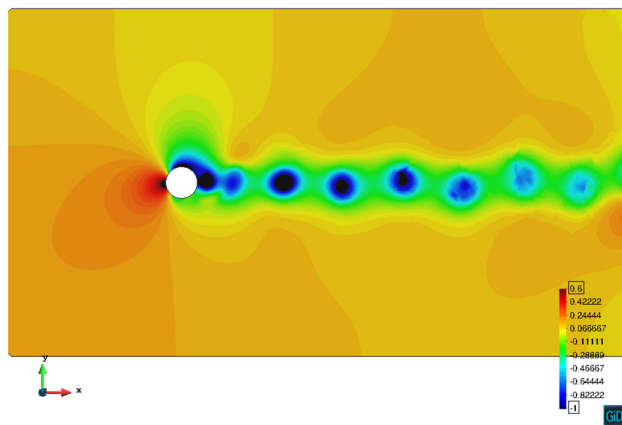


Fig. 8 Geometry



(a) Velocity.

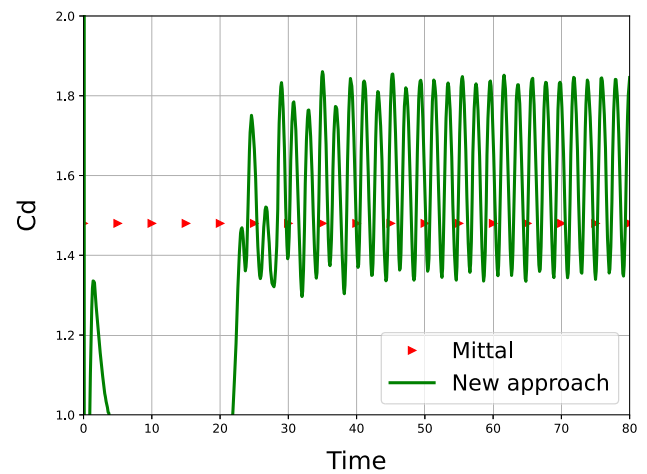


(b) Pressure.

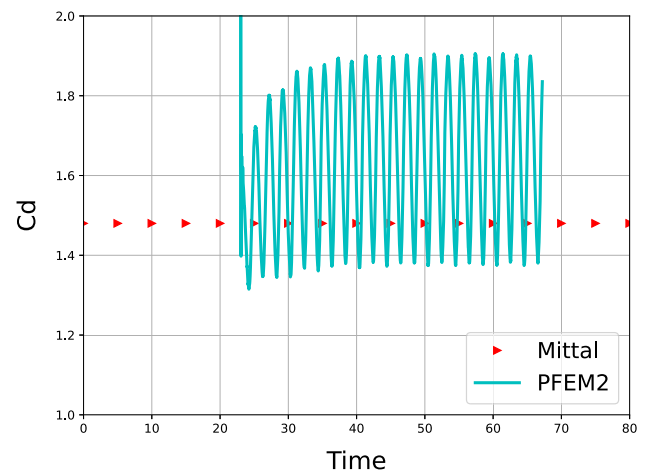
Fig. 9 Flow around a circular cylinder. Velocity and pressure fields at 80 [s]

### 7 Summary and conclusions

A modified new semi-explicit Strang-based PFEM technique has been proposed for modeling incompressible flows. The improvement of the PFEM strategy has consisted in calculating the convective step using an auxiliary acceleration term. The integration of both the position and the velocities, which



(a) New approach versus Mittal.

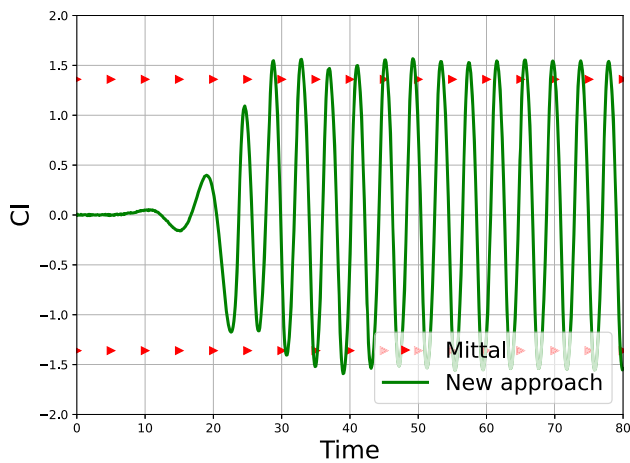


(b) PFEM2 versus Mittal.

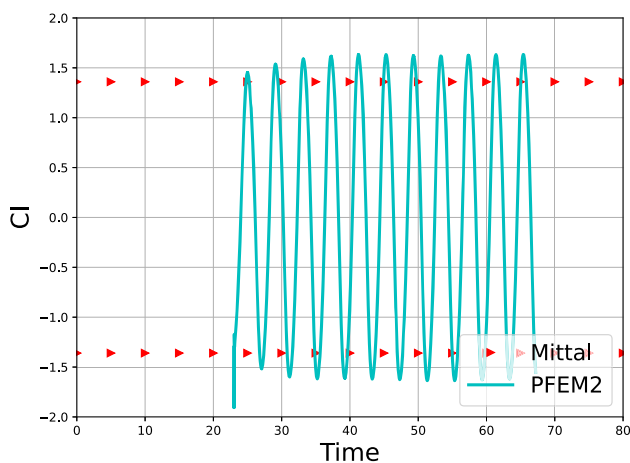
Fig. 10 Flow around a circular cylinder. Drag coefficient

emanate from the convective step, have been performed using a fourth Runge–Kutta scheme and interpolating from the mesh the intermediate velocities and accelerations. As a consequence, the new PFEM approach leads to a more efficient computational solver.

The enhanced PFEM scheme has been compared with results for 2D test cases from the literature. The first test has shown the importance of taking into account the acceleration term proposed in [32] and reveals second-order convergence versus mesh size and time. The second example has shown the good features of the new PFEM scheme for solving a more complicated setting. The numerical results have confirmed that the new PFEM scheme improves the accuracy and speed of previous semi-explicit versions.



(a) New approach versus Mittal.



(b) PFEM2 versus Mittal.

**Fig. 11** Flow around a circular cylinder. Lift coefficient

**Acknowledgements** The authors acknowledge financial support from the CERCA programme of the Generalitat de Catalunya, the Ministerio de Ciencia, Innovación e Universidades of Spain via the Severo Ochoa Programme for Centres of Excellence in RD (referece: CEX2018-000797-S) and the project PARAFUIDS (PID2019-104528RB-100) of the National Research Plan of the Spanish Government.

## Declarations

**Conflict of interest** The authors declare no conflict of interest.

## References

- Girault V, Raviart P-A (2011) Finite element methods for Navier–Stokes equations: theory and algorithms, 1st edn. Springer Publishing Company Incorporated, Berlin
- Ramaswamy B, Kawahara M (1987) Lagrangian finite element analysis applied to viscous free surface fluid flow. *Int J Numer Methods Fluids* 7(9):953–984
- Radovitzky R, Ortiz M (1998) Lagrangian finite element analysis of Newtonian fluid flows. *Int J Numer Methods Eng* 43(4):607–619
- Muttin F, Coupez T, Bellet M, Chenot J-L (1993) Lagrangian finite-element analysis of time-dependent viscous free-surface flow using an automatic remeshing technique: application to metal casting flow. *Int J Numer Methods Eng* 36(12):2001–2015
- Bennett A (2006) Lagrangian fluid dynamics. Cambridge University Press, Cambridge
- Idelsohn SR, 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
- Oñate E, Idelsohn SR, Del Pin F, Aubry R (2004) The particle finite element method: an overview. *Int J Comput Methods* 1:267–307
- Idelsohn SR, Marti J, Souto-Iglesias A, Oñate E (2008) Interaction between an elastic structure and free-surface flows: experimental versus numerical comparisons using the PFEM. *Comput Mech* 43(1):125–132
- Marti J, Idelsohn S, Limache A, Calvo J, D’Elía N (2006) A Fully coupled particle method for quasi-incompressible fluid hypoelastic structure interactions. In Cardona A, Nigro N, Sonzogni V, Storti M (eds) *Mecánica computacional*, vol XXV, pp 809–827
- Idelsohn SR, Marti J, Limache A, Oñate E (2008) Unified Lagrangian formulation for elastic solids and incompressible fluids. Application to fluid-structure interaction problems via the PFEM. *Comput Methods Appl Mech Eng* 197:1762–1776
- Cremonesi M, Frangi A, Perego U (2010) A Lagrangian finite element approach for the analysis of fluid-structure interaction problems. *Int J Numer Methods Eng* 84(5):610–630
- Cremonesi M, Meduri S, Perego U, Frangi A (2017) An explicit Lagrangian finite element method for free-surface weakly compressible flows. *Comput Part Mech* 4(3):357–369
- 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
- Cerquaglia ML, Thomas D, Boman R, Terrapon V, Ponthot J-P (2019) 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* 348:409–442
- Zhu M, Scott MH (2014) Improved fractional step method for simulating fluid-structure interaction using the PFEM. *Int J Numer Methods Eng* 99(12):925–944
- Zhu M, Scott MH (2017) Unified fractional step method for Lagrangian analysis of quasi-incompressible fluid and nonlinear structure interaction using the PFEM. *Int J Numer Meth Eng* 109(9):1219–1236
- Idelsohn SR, Mier-Torrecilla M, Oñate E (2009) Multi-fluid flows with the particle finite element method. *Comput Methods Appl Mech Eng* 198(33–36):2750–2767
- Idelsohn SR, Mier-Torrecilla M, Nigro N, Oñate E (2010) On the analysis of heterogeneous fluids with jumps in the viscosity using a discontinuous pressure field. *Comput Mech* 46(1):115–124
- Idelsohn SR, Mier-Torrecilla M, Marti J, Oñate E (2011) The particle finite element method for multi-fluid flows. In: Oñate E, Owen R (eds) *Particle-Based Methods: Fundamentals and Applications*. Springer, Netherlands, pp 135–158
- Oñate E, Rojek J, Chiumenti M, Idelsohn SR, Del Pin F, Aubry R (2006) Advances in stabilized finite element and particle methods for bulk forming processes. *Comput Methods Appl Mech Eng* 195(48–49):6750–6777
- Hyre M (2002) Numerical simulation of glass forming and conditioning. *J Am Ceram Soc* 85(5):1047–1056

22. Feulvarch E, Moulin N, Saillard P, Lornage T, Bergheau J-M (2005) 3D simulation of glass forming process. *J Mater Process Technol* 164:1197–1203
23. Idelsohn SR, Marti J, Becker P, Oñate E (2014) Analysis of multifluid flows with large time steps using the particle finite element method. *Int J Numer Methods Fluids* 75(9):621–644
24. Becker P (2015) *An enhanced particle finite element method with special emphasis on landslides and debris flows*. Ph.D. Thesis, Universitat Politècnica de Catalunya
25. Idelsohn SR, Nigro N, Gimenez J, Rossi R, Marti J (2013) A fast and accurate method to solve the incompressible Navier–Stokes equations. *Eng Comput* 30(2):197–222
26. Yazici Y (2010) Operator splitting methods for differential equations. Ph.D. Thesis, Izmir Institute of Technology
27. Semenov YuA (1977) On the lie-trotter theorems in  $l(p)$  spaces. *Lett Math Phys* 1:379–385
28. Omer R, Bashier E, Arbab AI (2017) Numerical solutions of a system of odes based on lie-trotter and Strang operator-splitting methods. *Univers J Comput Math* 5(2):20–24
29. Marti J, Ryzhakov P (2020) Improving accuracy of the moving grid particle finite element method via a scheme based on Strang splitting. *Comput Methods Appl Mech Eng* 369:113212
30. Strang G (1968) On the construction and comparison of difference schemes. *SIAM J Numer Anal* 5(3):506–517
31. MacNamara S, Strang G (2016) Operator splitting. In: *Splitting methods in communication, imaging, science, and engineering*. Springer, pp 95–114
32. Marti J, Oñate E (2022) An enhanced semi-explicit particle finite element method for incompressible flows. *Comput Mech* 70(3):607–620
33. Gimenez J, Aguerre H, Idelsohn SR, Nigro N (2019) A second-order in time and space particle-based method to solve flow problems on arbitrary meshes. *J Comput Phys* 380:295–310
34. Ryzhakov P, Marti J, Idelsohn SR, Oñate E (2017) Fast fluid-structure interaction simulations using a displacement-based finite element model equipped with an explicit streamline integration prediction. *Comput Methods Appl Mech Eng* 315:1080–1097
35. Delaunay B et al (1934) Sur la sphere vide. *Izv Akad Nauk SSSR, Otdelenie Matematicheskii i Estestvennyka Nauk* 7(793–800):1–2
36. Edelsbrunner H, Tan TS (1993) An upper bound for conforming Delaunay triangulations. *Discrete Comput Geom* 10(2):197–213
37. Edelsbrunner H, Mücke E (1994) Three-dimensional alpha shapes. *ACM Trans Graph* 13(1):43–72
38. Holzapfel Gerhard A (2002) *Nonlinear solid mechanics: a continuum approach for engineering science*. *Meccanica* 37(4):489–490
39. Aubry R, Oñate E, Idelsohn SR (2006) Fractional step like schemes for free surface problems with thermal coupling using the Lagrangian PFEM. *Int J Comput Mech* 38:294–309
40. Chorin AJ (1967) A numerical method for solving incompressible viscous problems. *J Comput Phys* 2:12–26
41. Yanenko, N.N.: The method of fractional steps. In: *The solution of problems of mathematical physics in several variables*. Springer edition. Translated from Russian by T. Cheron (1971)
42. Temam R (1969) Sur l’approximation de la solution des equations de Navier-Stokes par la methode des pase fractionaires. *Arch Ration Mech Anal* 32:135–153
43. Codina R (2001) A stabilized finite element method for generalized stationary incompressible flows. *Comput Methods Appl Mech Eng* 190(20–21):2681–2706
44. Marti J, Ryzhakov P (2019) An explicit-implicit finite element model for the numerical solution of incompressible Navier–Stokes equations on moving grids. *Comput Methods Appl Mech Eng* 350:750–765
45. Oñate E (2016) Finite increment calculus (FIC): a framework for deriving enhanced computational methods in mechanics. *Adv Model Simul Eng Sci* 3:12
46. 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
47. Kratos Multiphysics at GitHub. <https://github.com/KratosMultiphysics/Kratos>. Accessed 03 Jan 2020
48. Cremonesi M, Meduri S, Perego U (2019) Lagrangian–Eulerian enforcement of non-homogeneous boundary conditions in the particle finite element method. *Comput Part Mech* 7:05
49. Mittal S, Kumar V (2001) Flow-induced vibrations of a light circular cylinder at Reynolds numbers 103 to 104. *J Sound Vib* 245:923–946, 08

**Publisher’s Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.