Improving accuracy of the moving grid Particle Finite Element Method via a scheme based on Strang splitting.

J. Marti[a,b], P.Ryzhakov[a,b]

[a] *Centre Internacional de Mètodes Numèrics en Enginyeria (CIMNE)*
Gran Capitán s/n, 08034 Barcelona, Spain
[b] *Department of Civil and Environmental Engineering, Universitat Politècnica de Catalunya (UPC), 08034 Barcelona, Spain*

Abstract

Particle finite element method (PFEM) is a computational tool suitable for simulating fluid dynamics problems characterized by presence of moving boundaries. In this paper a new version of the method for incompressible flow problems is proposed aiming at accuracy improvement. This goal is achieved essentially by applying Strang operator splitting to Navier-Stokes equations and selecting adequate integration schemes for the resulting advective and Stokes sub-problems. For achieving efficient implementation, the pressure and the velocity in the Stokes part are decoupled via the fractional step technique as in the classical PFEM. However, at the first fractional step an explicit pressure prediction procedure for alleviating mass losses is introduced. Three test cases are solved, validating the methodology and estimating its accuracy. The numerical evidence proves that the proposed scheme improves the accuracy of the PFEM.

Keywords:
incompressible Navier-Stokes, Free-surface flows, PFEM, Lagrangian, Strang splitting

1. INTRODUCTION

Lagrangian finite element models for incompressible fluid flow problems have emerged since more than two decades. They were found advantageous

Email address: julio.marti@cimne.upc.edu (J. Marti)

for applications characterized by large shape changes of the computational domain shape and/or presence of moving boundaries [1, 2, 3, 4, 5]. The main strength of Lagrangian fluid models stems from their intrinsic ability of tracking the domain deformations: the domain boundaries are represented by computational mesh that deforms together with the corresponding media. Moreover, in Lagrangian models convective term of the momentum equation vanishes, resulting in a symmetric generalized tangent matrix in implicit versions of the method, which is advantageous in the scope of using iterative linear solvers. While these features of Lagrangian models are attractive for solving various problems, such models are characterized by an important limitation being prone to mesh degradation. Unlike another moving-grid class of methods, namely the Arbitrary Lagrangian/Eulerian (ALE) approach [6], purely Lagrangian methods do not permit applying mesh smoothing procedures as the mesh position is uniquely defined by the solution of the physical problem. Thus, mesh degradation cannot be tackled by adjusting nodal positions (done in ALE) and calls for obligatory re-meshing. Lagrangian finite element models for the fluid flows equipped with the re-meshing and boundary recognition schemes received the name of the particle finite element method (PFEM) [5]. In order to keep the re-meshing cost minimum, the procedure used in the PFEM consists in reconnecting the existing nodes using Delaunay method [7]. The unknowns of the problem are stored at nodes (which have the connotation of immaterial particles), since mesh elements are not preserved from one time step to another. For further details regarding basic ”ingredients” of PFEM the reader is referred to [8].

Various versions of PFEM have been applied to the simulation of free-surface hydrodynamics [5, 9], fluid-structure interaction [10, 8, 11, 12, 13, 14, 15, 16], immiscible two-fluid flows [17, 18, 19, 20] and thermo-mechanical forming processes [21, 22] and bottle manufacturing [23, 24]. There also exist Lagrangian models similar to PFEM, that have been applied to modeling of material forming [3, 25, 26].

Even though the basic philosophy of the PFEM has not changed since its emergence back in 2004 [5], active research has been devoted to improving the approach with respect to robustness and accuracy. These can be defined more specifically in terms of the following features:

- Alleviation of the mesh degradation impact

- Improvement of time accuracy of the method

- Improvement of mass conservation of the method

Mesh degradation faced when applying the method to problems characterized by large distortions in the domain shape was already mentioned above. In classical implicit PFEM simulations critical situations emerge when a mesh element fully degrades (attaining zero area) or even inverts within one time step, i.e. during the non-linear iteration process. For such cases re-meshing procedure provides no remedy as it is performed only at the end of each time step, once the convergence was achieved. For implicit models such situation leads to immediate failure of the solver. In order to prevent such failures several remedies were proposed. First one consisted in reducing time step ensuring that no element is inverted. It was shown that the critical time step can be estimated e.g. from Courant number of the problem [27]. Adaptive time stepping is used in the vast majority of the existing PFEM solvers. The disadvantage of this approach originates from the fact that the critical time step may become excessively small, leading to a strong bottleneck in the model from the computational efficiency point of view. Additional remedy consists in adaptive mesh refinement/de-refinement that allows improving element quality locally.

Qualitatively different remedy was established recently. Methods splitting the solution into an explicit and an implicit step were developed. These rely on resolving nodal advection explicitly, while treating the rest of the problem implicitly (it was introduced for Lagrangian PFEM in [28, 29] and for Lagrangian-Eulerian PFEM in [30]). Since in these methods mesh nodes move during the explicit step only and elements do not deform during the implicit step, one ensures convergent solutions without severe time step restrictions. Moreover, in its original form the explicit-implicit scheme [28] treats the implicit step as linear. A version of the method that introduced an additional step for correcting nodal positions was proposed in [27].

While the above-mentioned remedy alleviated time step limitations and led to solvers with improved computational efficiency, these newest versions of the PFEM turned out to exhibit low time accuracy. This originated from the fact that the solution of the governing equations was performed using an operator splitting equivalent to that of Lie-Trotter method, which is known to reduce the time accuracy [31, 32, 33].

Yet another crucial issue of the PFEM models affecting their accuracy has been the mass conservation quality. Vast majority of PFEM models use popular fractional step approach [34, 35, 36] in its algebraic version (see e.g.

[37]) for decoupling the velocity and the pressure, leading to strong computational speed-up of the model. However, as shown in [38] fractional step-based solvers typically lead to considerable mass conservation deficiencies, particularely evident when applied to the simulation of free surface flow problems. This deficiency is associated with the artificial zero-pressure boundary condition used for solving the Poisson's equation arising in the second step of the method. To palliate this problem, several authors proposed approaches based on introducing artificial compressibility which enables pressure condensation and does not require solving Poisson's equation for the pressure [11, 12, 39]. This strategy led to an additional volumetric stiffness contribution (proportional to the artificial compressibility constant) to the generalized tangent matrix of the problem. It became evident that these methods are efficient only when moderate values of the artificial compressibility constant (bulk modulus) are used and thus can be applied to a limited number of problems. Large value of the bulk modulus led to ill conditioning of the discrete system of equations.

Present work focuses on improving the accuracy and robustness of the method focusing on the above-described three features. We strive to develop a second order time-accurate version of PFEM with good mass conservation features and suitable for modeling truly incompressible flows. The key feature of the method is the application of a second-order-accurate operator splitting procedure to the Navier-Stokes equations. The overall scheme proposed consists of three steps: two of them in which the mesh is moving and one where the Stokes equation is solved on a "frozen" mesh. Due to the implicit nature of the pressure in incompressible flows, pressure is integrated implicitly. However, for relieving the mass losses an explicit pressure prediction step is introduced within the framework of the fractional step method. Being explicit, this step introduces no modification in the tangent matrix.

The paper is structured as follows: first, the system of equations governing an incompressible Newtonian fluid flow is presented. Next, the main ingredients of the improved PFEM version are introduced, particularly the Strang advection-diffusion splitting which ensures improved accuracy. Next, temporal and spatial discretization of the governing equations is shown. The overall solution algorithm is outlined. The paper concludes with three numerical examples. In the first two the method is examined with respect to its time accuracy. The third one focuses on the mass conservation quality of the method.

2. NUMERICAL MODEL

2.1. Governing equations at a continuum level

Let Ω^t denote a domain containing a viscous incompressible fluid with a fixed boundary Γ_D and a free surface Γ_N. The evolution of the velocity $\mathbf{v} = \mathbf{v}(\mathbf{x}, t)$ and the pressure $p = p(\mathbf{x}, t)$ are governed by the Navier-Stokes equations given by

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

$$\nabla \cdot \mathbf{v} = 0 \tag{2}$$

where μ is the dynamic viscosity, ρ is the density, $\mathbf{b}$ is the body force, ∇ is the gradient operator and ∇^s its symmetric part.

Eqs.1 and 2 are completed with the standard Neumann (Eq.3) and Dirichlet (Eq.4) boundary conditions at Γ_N and Γ_D, respectively.

$$-p\mathbf{n} + 2\mu \nabla^s \mathbf{v} \cdot \mathbf{n} = \bar{\mathbf{t}} \quad on \quad \Gamma_N \tag{3}$$

$$\mathbf{v} = \bar{\mathbf{v}} \quad on \quad \Gamma_D \tag{4}$$

Next subsection introduces splitting strategies that decouple the momentum equation into advective and diffusive parts emphasising their impact in the context of the PFEM.

2.2. Operator splitting for the PFEM

Classical implicit PFEM In the traditional versions of PFEM [5, 9, 10, 8, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24] the strategy for solving the Navier-Stokes equations on the moving grid consisted in solving momentum and continuity (Eqs. 1, 2) for velocity and pressure and, consequently updating the mesh position integrating the following equation $\frac{d\mathbf{x}}{dt} = \mathbf{v}$. This procedure was carried out at each iteration step of the non-linear implicit solver. As already mentioned, the deficiency of this approach consisted in the fact that the elements could be inverted at any non-linear iteration step.

Splitting in modern explicit-implicit PFEM Recent versions of PFEM rely on explicit-implicit schemes resulting from applying operator splitting methodology [28]. Generally, operator splitting methods[40, 41, 42] are widely used for solving convection-diffusion equations. Momentum conservation Eq.1 belongs to this class of equations.

Splitting methods are based on the assumption that the problem can be separated into two [43] i.e. advective one and diffusion (Stokes) one. As a consequence of this split, simpler equations are obtained which can be solved at a much lower computational cost than the original system. The solution of the original problem is obtained knowing the solutions of the sub-problems [44]. The sub-problems are often solved sequentially, i.e. the output of the first sub-problem is the input to the next sub-problem within a time step.

In the context of PFEM [28, 27] the advective problem serves as the basis for the movement of the mesh nodes (immaterial particles of the PFEM). Once the advection problem is solved in a single explicit step, the end-of-step domain configuration is obtained and this configuration is used for solving the Stokes problem. Application of this split to the momentum conservation Eq.1 can be written as:

$$\rho\frac{\partial \mathbf{v}^*}{\partial t} + \rho\mathbf{v}^* \cdot \nabla\mathbf{v}^* = \mathbf{0} \quad \text{with} \quad \mathbf{v}^*(\mathrm{t}, \mathbf{x}) = \mathbf{v}(\mathrm{t}, \mathbf{x}) \quad \mathrm{t}\epsilon[\mathrm{t}, \mathrm{t} + \delta t] \tag{5}$$

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

This type of splitting is known in literature as "Lie-Trotter method" [42]. Although Lie-Trotter split has been commonly used due to simplicity of its implementation, the main drawback is that it leads to reduction of time accuracy of the overall method to first order independent of the accuracy of the time integration schemes used for the resulting sub-problems [45, 33].

Strang operator splitting for PFEM In the present work we propose to employ a different strategy for decoupling the advection and the Stokes steps in order to improve the time accuracy of the method. The idea consists in applying the Strang splitting method, which is expected to improve the time accuracy of the resulting scheme to second order according to the demonstration of [45] made for scalar convection-diffusion equation.

As a result of applying the Strang operator splitting, Eq.1 can be divided and the unknown solution at $t + \delta t$ is obtained in three following steps:

$$\rho\frac{\partial \mathbf{v}^*}{\partial t} + \rho\mathbf{v}^* \cdot \nabla\mathbf{v}^* = \mathbf{0} \quad \text{with} \quad \mathbf{v}^*(\mathrm{t}, \mathbf{x}) = \mathbf{v}(\mathrm{t}, \mathbf{x}) \quad \mathrm{t}\epsilon[\mathrm{t}, \mathrm{t} + \delta t/2] \tag{7}$$

$$\begin{aligned} \rho\frac{\partial \mathbf{v}}{\partial t} &= -\nabla p + \nabla \cdot (2\mu\nabla^s\mathbf{v}) + \rho\mathbf{b} \quad \text{with} \\ \mathbf{v}(\mathrm{t}, \mathbf{x}) &= \mathbf{v}^*(\mathrm{t} + \delta t/2, \mathbf{x}) \quad \mathrm{t}\epsilon[\mathrm{t}, \mathrm{t} + \delta\ \mathrm{t}] \end{aligned} \tag{8}$$

$$\rho\frac{\partial \hat{\mathbf{v}}}{\partial t} + \rho\hat{\mathbf{v}} \cdot \nabla\hat{\mathbf{v}} = \mathbf{0} \quad \text{with} \quad \hat{\mathbf{v}}(\mathrm{t}+\delta t/2, \mathbf{x}) = \mathbf{v}(\mathrm{t}+\delta t, \mathbf{x}) \quad \mathrm{t}\epsilon[\mathrm{t}+\delta t/2, \mathrm{t}+\delta t] \tag{9}$$

Eq.7 must be solved for half of the time increment $\delta t/2$. The solution of Eq.7 is used as the initial condition of Eq.8 which is solved for a full time increment δt. The obtained result of Eq.8 is the initial condition for Eq.9. Lastly, Eq.9 is solved for a half time increment, $\delta t/2$.

As Strang splitting is expected to provide second order accuracy [46], at least a second-order scheme for time discretization should be employed for the advection and diffusion sub-problems in order to obtain a globally second-order method. Spatial discretization and time integration of Eq.8 and Eqs.7,9 is explained in the following sub-section.

2.3. Crank-Nicolson scheme and FEM for the solution of the diffusive part.

Let us concentrate first of the solution of Stokes equation (Eq. 8) on a "frozen" mesh (this expression is used in order to emphasize that the mesh position does not change during the corresponding step). The integration of nodal motion (Eqs.7 and 9) is described in Section 2.4. At this point we consider that the first mesh motion step is performed and the configuration for solving Stokes problem is obtained.

For the momentum conservation (Eq.8) we use a second order scheme, namely, the Crank-Nicolson one. In principle, any second-order time integration scheme can be used. Using time integration scheme of higher order is not justified as Strang splitting would annihilate its benefit reducing the accuracy to second order. Crank-Nicholson scheme applied for time discretization of the momentum equation leads to:

$$\rho\frac{\mathbf{v}^{n+1}-\mathbf{v}^{n}}{\Delta t}=\frac{1}{2}\{-(\nabla p^{n+1}+\nabla p^{n})+ \\ \nabla\cdot(2\mu\nabla^{s}\mathbf{v}^{n+1})+\nabla\cdot(2\mu\nabla^{s}\mathbf{v}^{n})\}+\rho\mathbf{b} \tag{10}$$

From now on, to reduce the clutter, the super-index $n+1$ will be neglected.

In the present work the fractional step method [34, 36, 35] is applied to the momentum equation in order to obtain an efficient implementation. Fractional step methods decouple the pressure from the velocity, making the solution procedure much less computationally expensive[37]. This method is used in many Lagrangian fluid models, such as e.g. [47, 5, 48, 49]. However, as we shall see further, the fractional step used here will have a certain modification with respect to its standard version, so as to ensure good mass conservation features of the overall scheme.

According to the fractional step approach Eq.10 can be written as

$$\frac{\rho}{\Delta t}\tilde{\mathbf{v}}-\nabla\cdot(\mu\nabla^{s}\tilde{\mathbf{v}})=\frac{\rho}{\Delta t}\mathbf{v}^{n}-\nabla p^{\star}+\nabla\cdot(\mu\nabla^{s}\mathbf{v}^{n})+\rho\mathbf{b} \tag{11}$$

$$\mathbf{v}=\tilde{\mathbf{v}}-\frac{\Delta t}{2\rho}(\nabla p-\nabla p^{\star}) \tag{12}$$

where $\tilde{\mathbf{v}}$ is the intermediate velocity. Here a new variable, p^{*} has been introduced. This variable will represent an approximation to the end-of-step pressure p^{n+1}. The approximation we propose will be defined below. Note that considering $p^{*}=p^{n}$, one obtains the standard fractional step, where the fractional momentum equation is solved using the historical value of the pressure.

Applying the incompressibility condition Eq.2 to Eq.12 leads to the Poisson's equation for the pressure

$$\nabla\cdot\tilde{\mathbf{v}}-\frac{\Delta t}{2\rho}\nabla\cdot(\nabla p-\nabla p^{\star})=0 \tag{13}$$

Explicit pressure prediction for improving mass conservation As already mentioned, in a standard second order fractional step approach, $p^{\star}=p^{n}$ and thus the pressure gradient term inside Eq.11 represents the gradient of the historical pressure. This term remains constant within the first fractional step (solution for intermediate velocity $\tilde{\mathbf{v}}$). While the time step is small enough, approximating the current pressure by the pressure of the previous step might be acceptable. However, for large time steps, the quality of such an approximation deteriorates. Following the idea presented in [50], an improved prediction of p^* can be computed. To accomplish this, the predicted pressure is obtained using the assumption of slight compressibility[51]. Thus, the density changes are related to the pressure changes by the following relationship:

$$dp = \frac{\kappa}{\rho} d\rho \tag{14}$$

where κ is the elastic bulk modulus of the fluid. Then applying the material time derivative to the equation above and using the equation of conservation of mass we get:

$$\frac{Dp}{Dt} = \frac{\kappa}{\rho}\frac{D\rho}{Dt} = -\kappa\nabla\cdot\mathbf{v} \tag{15}$$

The time-discretized version of the equation can be written as

$$\frac{p^{n+1} - p^n}{\Delta t} = -\kappa\nabla\cdot\mathbf{v}^{n+1} \tag{16}$$

Renaming p^{n+1} as p^* (emphasizing that this is only a prediction for the end-of-step pressure), we propose to evaluate p^* using the $\mathbf{v}^n$ instead of $\mathbf{v}^{n+1}$. Thus the predicted pressure can be evaluated explicitly as

$$p^* = p^n - \kappa\nabla\cdot\mathbf{v}^n\Delta t \tag{17}$$

Its space-discrete version can be written as

$$\mathbf{M}^p p^* = \mathbf{M}^p p^n - \kappa\mathbf{B}\ \mathbf{v}^n\Delta t \tag{18}$$

where $\mathbf{M}^p$ is the mass matrix and $\mathbf{B}$ is the strain rate matrix.

The pressure obtained from Eq. 18 provides an initial guess for the pressure and is used in Eq.11 to evaluate the intermediate velocity.

It should be noted that unlike similar pressure prediction schemes presented in [38, 49] present approach is explicit. It is characterized by a negligible computational cost in case of using lumped mass matrix in the term $\mathbf{M}^p p^*$.

The values of the pressure obtained using Eq.17 can be used as a boundary condition for the Eq.13 instead of the traditional zero pressure at the free surface for inviscid or low viscosity flows known to introduce severe mass losses [38].

Using higher order explicit tie integration schemes for obtaining pressure prediction appears unfeasible as it would require evaluating the unknown velocity at the intermediate steps, strongly increasing the computational cost. Although the scheme employed here (Eq. 18) is first order accurate, it is used only for obtaining the initial guess for the pressure. Afterwards the pressure is corrected via solving Poisson's equation.

Weak form of the problem: spatial discretization

The weak form of the governing equations is obtained using standard Finite Element procedure. Eqs. 11-13 and the natural boundary condition are multiplied by a vector of linear finite element trial functions **N** and are integrated over the domain Ω.

Multiplying Eqs. 11-13 and the natural boundary condition by a vector of linear finite element trial functions **N** and integrating over the domain Ω gives:

$$\begin{aligned} \int_\Omega \rho \mathbf{N} \cdot \tilde{\mathbf{v}} d\Omega - \Delta t \int_\Omega \mathbf{N} \cdot \nabla \cdot (\mu \nabla^s \tilde{\mathbf{v}}) d\Omega = \int_\Omega \rho \mathbf{N} \cdot \mathbf{v}^n d\Omega - \\ \Delta t \int_\Omega \mathbf{N} \cdot \nabla p^\star d\Omega + \Delta t \int_\Omega \mathbf{N} \cdot \nabla \cdot (\mu \nabla^s \mathbf{v}^n) d\Omega + \Delta t \int_\Omega \rho \mathbf{N} \cdot \mathbf{b} d\Omega + \\ \Delta t \int_{\Gamma_N} \mathbf{N} \cdot \left(\bar{\mathbf{t}} + p^\star \mathbf{n} - \mu \nabla^s \tilde{\mathbf{v}} \cdot \mathbf{n} - \mu \nabla^s \mathbf{v}^n \cdot \mathbf{n} \right) d\Gamma \end{aligned} \tag{19}$$

$$\frac{\Delta t}{2} \int_\Omega N \nabla \cdot \nabla p d\Omega = \frac{\Delta t}{2} \int_\Omega N \nabla \cdot \nabla p^\star d\Omega + \int_\Omega \rho N \nabla \cdot \tilde{\mathbf{v}} d\Omega \tag{20}$$

$$\begin{aligned} \int_\Omega \rho \mathbf{N} \cdot \mathbf{v} d\Omega = \int_\Omega \rho \mathbf{N} \cdot \tilde{\mathbf{v}} d\Omega - \frac{\Delta t}{2} \int_\Omega \mathbf{N} \cdot (\nabla p - \nabla p^\star) d\Omega + \\ \frac{\Delta t}{2} \int_{\Gamma_N} \mathbf{N} \cdot (p - p^\star) \, \mathbf{n} d\Gamma \end{aligned} \tag{21}$$

Next, subdividing the domain Ω into non-overlapping elements, Ω_e, applying the divergence theorem to the diffusion terms and splitting the integral in the weak formulation into a sum of integrals over each element allows rewriting the Eqs.19-21 as

$$\sum_{elem}[\int_{\Omega_e}\rho\mathbf{N}\cdot\tilde{\mathbf{v}}d\Omega+\Delta t\int_{\Omega_e}\nabla\mathbf{N}:\mu\nabla^s\tilde{\mathbf{v}}d\Omega=$$

$$\int_{\Omega_e}\rho\mathbf{N}\cdot\mathbf{v}^n d\Omega+\Delta t\int_{\Omega_e}\nabla\mathbf{N}:p^\star d\Omega+$$

$$\Delta t\int_{\Omega_e}\nabla\mathbf{N}:\mu\nabla^s\mathbf{v}^n d\Omega+\Delta t\int_{\Omega_e}\rho\mathbf{N}\cdot\mathbf{b}d\Omega+\int_{\Gamma_{N_e}}\mathbf{N}\cdot\bar{\mathbf{t}}d\Gamma] \tag{22}$$

$$\sum_{elem}[\frac{\Delta t}{2}\int_{\Omega_e}\nabla N\cdot\nabla p d\Omega=\frac{\Delta t}{2}\int_{\Omega_e}\nabla N\cdot\nabla p^\star d\Omega+\int_{\Omega_e}\rho N\nabla\cdot\tilde{\mathbf{v}}d\Omega] \tag{23}$$

$$\sum_{elem}[\int_{\Omega_e}\rho\mathbf{N}\cdot\mathbf{v}d\Omega=\int_{\Omega_e}\rho\mathbf{N}\cdot\tilde{\mathbf{v}}d\Omega+\frac{\Delta t}{2}\int_{\Omega_e}\nabla\mathbf{N}:(p-p^\star)d\Omega] \tag{24}$$

Considering linear finite element element approximations for the velocity and the pressure unknowns $\mathrm{v}_i(\mathbf{x})=\mathbf{N}^T(\mathbf{x})\,\mathrm{v}_i$ and $p(\mathbf{x})=\mathbf{N}^T(\mathbf{x})\,p$ where v_i and p are the nodal values, leads to the following discrete form of the governing equations

$$(\mathbf{M}+\Delta t\mathbf{K})\,\tilde{\mathbf{v}}=(\mathbf{M}+\Delta t\mathbf{K})\,\mathbf{v}^n+\Delta t\mathbf{G}p^\star+\Delta t\mathbf{F}\mathbf{b} \tag{25}$$

$$\frac{\Delta t}{2}\mathbf{L}p^{n+1}=\frac{\Delta t}{2}\mathbf{L}p^\star-\mathbf{D}\tilde{\mathbf{v}} \tag{26}$$

$$\mathbf{v}^{n+1}=\tilde{\mathbf{v}}+\frac{\Delta t}{2}\mathbf{M}^{-1}\mathbf{G}(p^{n+1}-p^\star) \tag{27}$$

The matrices and vectors presented above are defined as

$$\mathbf{K}=\sum_{elem}\int_{\Omega_e}\mu\nabla\mathbf{N}\nabla^s\mathbf{N}d\Omega \tag{28}$$

$$\mathbf{M}=\sum_{elem}\int_{\Omega_e}\rho\mathbf{N}\mathbf{N}d\Omega \tag{29}$$

$$\mathbf{F}=\sum_{elem}\int_{\Omega_e}\rho\mathbf{N}d\Omega \tag{30}$$

$$\mathbf{G} = \sum_{elem} \int_{\Omega_e} \nabla \mathbf{N} \mathbf{N} d\Omega \tag{31}$$

$$\mathbf{D} = \sum_{elem} \int_{\Omega_e} \rho \mathbf{N} \nabla \mathbf{N} d\Omega \tag{32}$$

$$\mathbf{L} = \sum_{elem} \int_{\Omega_e} \nabla \mathbf{N} \nabla \mathbf{N} d\Omega \tag{33}$$

In order to obtain stable solutions, the Eq.26 must be stabilized since equal order velocity-pressure interpolations were used. In this work ASGS stabilization technique has been employed [52]. For the sake of simplicity, stabilization terms are omitted here. They can be consulted in [53] where the stabilization used here is described in detail.

Having presented the discrete version of Stokes problem to solve, the next ingredient to be defined is the strategy employed for moving the mesh.

2.4. The Runge-Kutta scheme for the integration of the convective part

The convective part of the Navier-Stokes equations (Eqs.7-9) can be converted into two ordinary differential equations taking into account the definition of the material derivative and the velocity of the particles as

$$\frac{D\mathbf{v}^*}{Dt} = \mathbf{0} \tag{34}$$

$$\frac{D\mathbf{x}}{Dt} = \mathbf{v}^* \tag{35}$$

As already mentioned, in PFEM the mesh nodes are treated as immaterial particles. Let us consider a particle p. Its position and velocity will be designated as $\mathbf{x}_p$ and $\mathbf{v}^*(\mathbf{x}_p)$, respectively. Then, integration of Eq.34 and Eq.35 for such particle at the first advection step (i.e. from t^n till $t^{n+1/2}$) yields:

$$\mathbf{v}^{*,n+1/2}(\mathbf{x}_p) = \mathbf{v}^{*,n}(\mathbf{x}_p) \tag{36}$$

along with

$$\mathbf{x}_p^{n+1/2} = \mathbf{x}_p^n + \frac{\Delta t}{2} \mathbf{v}^{*,n} \tag{37}$$

As Eq.36 states that the two velocities ($\mathbf{v}^{*,n+1/2}(\mathbf{x}_p)$ and $\mathbf{v}^{*,n}(\mathbf{x}_p)$) are equal. Thus, according to Eq.37 the position of the particle depends on the velocity at time t^n, while the velocity Eq.36 does not depend on the position of the particle. Therefore, the mesh movement does not depend on the unknown velocity. In this work particle position (Eq.35) will be computed by applying 4th order Runge-Kutta scheme. This explicit scheme is known for providing very accurate trajectory integration at low cost and is widely used in numerical particle tracing [54, 55]. According to this integration scheme, the particle position can be computed as:

$$\mathbf{x}_p^{n+1/2} = \mathbf{x}_p^n + \frac{(\Delta t/2)}{6}(\mathbf{v}_1 + 2\mathbf{v}_2 + 2\mathbf{v}_3 + \mathbf{v}_4) \tag{38}$$

where $\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3$ and $\mathbf{v}_4$ are the intermediate velocities defined as:

$$\begin{aligned} \mathbf{v}_1 &= \mathbf{v}^{*,n}(\mathbf{x}_{p1} = \mathbf{x}_p^n) \\ \mathbf{v}_2 &= \mathbf{v}^{*,n}(\mathbf{x}_{p2} = \mathbf{x}_p^n + \frac{\Delta t}{4}\mathbf{v}_1) \\ \mathbf{v}_3 &= \mathbf{v}^{*,n}(\mathbf{x}_{p3} = \mathbf{x}_p^n + \frac{\Delta t}{4}\mathbf{v}_2) \\ \mathbf{v}_4 &= \mathbf{v}^{*,n}(\mathbf{x}_{p4} = \mathbf{x}_p^n + \frac{\Delta t}{2}\mathbf{v}_3) \end{aligned} \tag{39}$$

Considering the above, the movement of the particles following Eq.38 is implemented. Fig. 1 illustrates the corresponding steps. Let us consider that each node of the computational mesh has an immaterial (virtual) particle associated to it. On Fig. 1 mesh nodes are shown as squares, while virtual particles are indicated as circles. The position of the two will be denoted as $\mathbf{x}_m$ (position of the mesh nodes) and $\mathbf{x}_p$ (position of the virtual particles). At the beginning of the mesh update step the positions of the nodes and the virtual particles coincide: $\mathbf{x}_{p1} = \mathbf{x}_m$. The velocity $\mathbf{v_1}$ equals nodal velocity. During the second step of Runge-Kutta scheme the virtual particle p moves according to $\mathbf{x}_{p2} = \mathbf{x}_p^n + \frac{\Delta t}{4}\mathbf{v}_1$ (see Fig. 1-a). The velocity at the virtual particle is being computed from the nodal velocities of the mesh element containing the particle as $\mathbf{v}_2 = \sum_{i=1}^{3} N_i(\mathbf{x}_{p2})\mathbf{v}_i^{*,n}(\mathbf{x}_m)$. Then the virtual particle is set back to its original location (see dashed line in the figure). During the third step, the virtual particle moves according to $\mathbf{x}_{p3} = \mathbf{x}_p^n + \frac{\Delta t}{4}\mathbf{v}_2$ (see Fig. 1-b). This gives the second approximation of the particle position at the intermediate time $t + \Delta t/4$. Then velocity is evaluated as $\mathbf{v}_3 = \sum_{i=1}^{3} N_i(\mathbf{x}_{p3})\mathbf{v}_i^{*,n}(\mathbf{x}_m)$ (see Fig. 1-c). Then the virtual

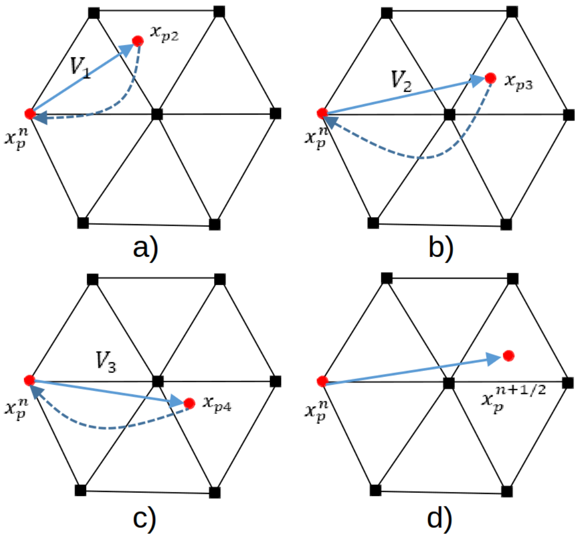


Figure 1: Schematic representation of particle position evaluation during the first advective step.

particle is brought back to its origin again. In forth step the particle moves according to $\mathbf{x}_{p4} = \mathbf{x}_p^n + \frac{\Delta t}{2}\mathbf{v}_3$ and the end-of-step velocity $\mathbf{v}_4$ is evaluated: $\mathbf{v}_4 = \sum_{i=1}^{3} N_i(\mathbf{x}_{p4})\mathbf{v}_i^{*,n}(\mathbf{x}_m)$ (see Fig. 1-d). After that the mesh nodes are moved according to Eq. 38 using the 4 intermediate values of the position of the corresponding particle. This gives the mesh configuration corresponding to $t^{n+1/2}$. We note that the described advective step involves motion of individual (unconnected) particles. Thus no element can be inverted independent of the time step used.

Once the particle advection is complete, they must be connected by a new mesh. The mesh is generated using the Delaunay method [7] or (EDT)[56] equipped with the alpha-shape technique [57] used for identifying the external boundaries. Being standard for all the PFEM methods, the re-meshing strategy is not described in detail here and reader is referred to [8] for standard implementation.

The obtained mesh configuration is used to solve the diffusion part (Stokes Eq.) Eqs. 25-27 . After that, explicit mesh motion step is repeated following the procedure described above, but applied from $t^{n+1/2}$ till t^{n+1}. The formulae are as follows:

$$\mathbf{x}_p^{n+1} = \mathbf{x}_p^{n+1/2} + \frac{(\Delta t/2)}{6}(\mathbf{v}_1 + 2\mathbf{v}_2 + 2\mathbf{v}_3 + \mathbf{v}_4) \tag{40}$$

where $\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3$ and $\mathbf{v}_4$ are the intermediate velocities defined as:

$$\begin{aligned}
\mathbf{v}_1 &= \mathbf{v}^{*,n+1/2}(\mathbf{x}_{p1} = \mathbf{x}_p^{n+1/2}) \\
\mathbf{v}_2 &= \mathbf{v}^{*,n+1/2}(\mathbf{x}_{p2} = \mathbf{x}_p^{n+1/2} + \frac{\Delta t}{4}\mathbf{v}_1) \\
\mathbf{v}_3 &= \mathbf{v}^{*,n+1/2}(\mathbf{x}_{p3} = \mathbf{x}_p^{n+1/2} + \frac{\Delta t}{4}\mathbf{v}_2) \\
\mathbf{v}_4 &= \mathbf{v}^{*,n+1/2}(\mathbf{x}_{p4} = \mathbf{x}_p^{n+1/2} + \frac{\Delta t}{2}\mathbf{v}_3)
\end{aligned} \tag{41}$$

It should be noted that the splitting of the Navier-Stokes Eqs. into an advective and diffusive parts and performing the advective step via explicit particle motion removes the stability restrictions defined by the CFL-number as shown in [30]. Thus, the method proposed here, similarly to [30] and [28], does not require reducing time step in order to avoid element inversions as it was done in the former PFEM versions, where the implicit step was also involving the mesh motion (such as [8]), [58], [12]). This does not mean however, that any time step can be used in practice, as in the case of using very large time steps the solution becomes unacceptably inaccurate.

Summarizing, the main difference between the approach proposed here and the recently proposed semi-explicit PFEM approches ([30], [29], [28]) stems from the fact that it relies on a second-order accurate operator split leading to two advective steps, that are integrated by a more accurate scheme.

In this work we shall concentrate on the class of Lagrangian methods known as Particle Finite Element Methods (PFEM), which are very similar to the classical Lagrangian Finite Element method used in solids.

2.5. OVERALL SOLUTION STRATEGY

To this end, all the ingredients of the PFEM model based on the Strang split are specified. The problem to be solved can be formulated as: ”given the nodal positions, the velocity and the pressure at time t^n, find these variables at t^{n+1}”. The overall solution strategy is summarized in Algorithm 1.

3. EXAMPLES

The present formulation is implemented in Kratos Multiphysics code, an in-house Open Source software [59], developed at CIMNE [60]. For the

Algorithm 1: Solution algorithm for the simulation of incompressible flows.

```
for $t = t^{n+1}$ do
    Move the particles to the new position at $\mathbf{x}^{n+1/2}$(Eq.38);
    if the mesh is too distorted then
        Re-mesh the fluid domain;
    end
    Compute prediction for the pressure $p^{\star}$ (Eq.17);
    Solve the Stokes problem on the "frozen" mesh:
        • Find intermediate velocity $\tilde{\mathbf{v}}$ solving Eq.25 on $\Omega$ with $\tilde{\mathbf{v}} = \bar{\mathbf{v}}$ on $\Gamma_D$ ;
        • Solve the Poisson's equation for the pressure (Eq.26) on $\Omega$ with
          $p = p^*$ on $\Gamma_N$. Result: $p^{n+1}$;
        • Correct the velocity to obtain a divergence-free solution (Eq.27) on $\Omega$
          with $\mathbf{v}^{n+1} = \bar{\mathbf{v}}$ on $\Gamma_D$ . Result: $\mathbf{v}^{n+1}$;
    Move the particles to the new position $\mathbf{x}^{n+1}$(Eq.40);
    if the mesh is too distorted then
        Re-mesh the fluid domain;
    end
end
```

solution of the implicit step of the velocity as well as the pressure Poisson's equation, conjugate gradient (CG) solver equipped with a diagonal pre-conditioner is used. The convergence tolerance of the linear solver is set to 10^{-6} and 10^{-4} for the velocity and the pressure, respectively.

3.1. Test with an analytic solution. Flow in a cavity.

The test proposed in [37] is used here to evaluate the temporal and the spatial accuracy of the proposed method. It models a unit cavity filled with an incompressible Newtonian fluid subjected to an external force corresponding to the following analytic velocity and pressure fields:

$$\mathrm{v}_x(\mathrm{x},\mathrm{y},t) = f(\mathrm{x})f'(\mathrm{y})\mathrm{g}(t) \tag{42}$$

$$\mathrm{v}_y(\mathrm{x},\mathrm{y},t) = -f'(\mathrm{x})f(\mathrm{y})g(t) \tag{43}$$

$$p = 100\mathrm{x}^2 \tag{44}$$

where

$$\begin{aligned} f(\mathrm{x}) &= 100\mathrm{x}^2(1-\mathrm{x})^2 \\ g(t) &= cos(4\pi t)e^{-t} \end{aligned} \tag{45}$$

Pressure was initialized according to the Eq. 44 in the entire domain and fixed to zero at left upper corner (x,y)=(0,1). Velocity at all the walls was fixed to zero. The viscosity μ and the density ρ were 0.001 kg/ms and 1 kg/m^3, respectively.

External force to be prescribed in order to obtain the manufactured solution defined by Eqs. 42-44 can be obtained by introducing Eqs. 42-45 inside the Navier-Stokes equations. Note that the external force vector in the present example depends both on time and space and therefore must be recomputed whenever particle movement is performed.

The example was solved using different meshes and time steps. The meshes used were as follows: 20 × 20 (h=0.05 m), 40 × 40 (h=0.025 m), 100 × 100 (h=0.01 m) and 200 × 200 (h=0.005 m) triangular elements.

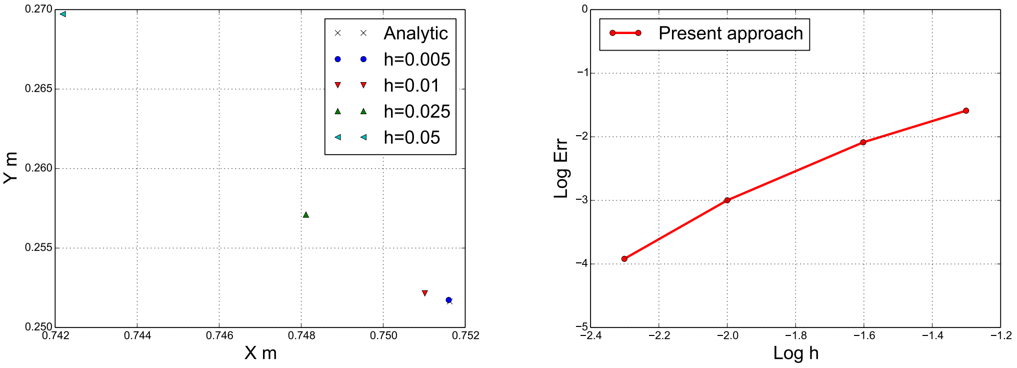
In order to evaluate the spatial and temporal accuracy, the idea presented in [53] was followed. It consists in tracking a particle which is initially located at the position (x,y)=(0.75, 0.25) during the entire simulation.

The final position of the selected particle at time 0.5 s is plotted on the Fig.2(a). One can see that the particle position approaches the analytical solution as the mesh size decreases. To check the spatial accuracy of the methodology, the example was simulated using a small time step (dt=0.0005 s) and the error in the final position of the inspected node was calculated as

$$\text{Error} = \sqrt{(\mathrm{x}_r - \mathrm{x}_n)^2 + (\mathrm{y}_r - \mathrm{y}_n)^2)} \tag{46}$$

where x_r and x_n are the analytic and the numerical result, respectively.

Fig.2(b) presents the error calculated using the Eq.46 versus mesh size. The method exhibits second order of convergence with respect to the mesh size.



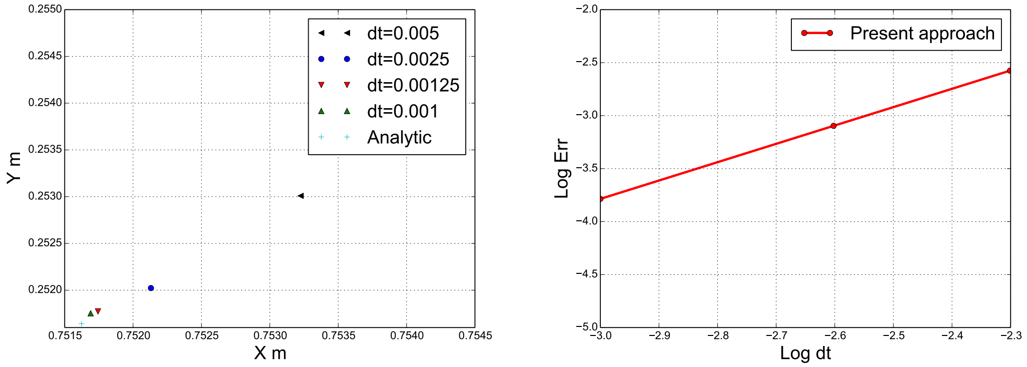
(a) Final position.

(b) Relative error

Figure 2: Flow in cavity. Final position at 0.5s using different mesh resolutions.

To minimize the spatial error, the temporal accuracy is evaluated using a fine structured mesh with 200 × 200 triangular elements (h=0.005 m). The example was simulated for 0.5 s using time steps varying from 0.001 to 0.005 s. Fig.3(a) shows the final position of the tracked particle at time 0.5 s for different time steps. We note that the spatial discretization error remains constant, which explains why the solution converges to a value slightly different from the analytic one. Therefore, to estimate the time accuracy the solution obtained using a very small time step was taken as the reference instead of the analytic one.

Error in the final position versus time step size is presented in the Fig.3(b) in a logarithmic scale. Figure shows nearly second-order time accuracy in terms of the particle position.



(a) Final position.

(b) Error in position. Comparison with semi-explicit reference [53]

Figure 3: Flow in cavity. Final position at 0.5s using different time step sizes.

Next, error in terms of velocity in the considered particle was measured. We compare the obtained results with those of [29], where the method of [30] was implemented. The method of [30, 29] is a semi-explicit PFEM that uses sub-stepping-based streamline integration of the particle positions in a single explicit advection step. Comparison is shown in Fig. 4.

One can see that while [29] is first order accurate, present approach results in a second order accuracy. This indicates that the methodology proposed here based on the application of Strang splitting improves the time accuracy of the semi-explicit PFEM. Fig. 4 shows also that for achieving an error of e.g. 10^{-4} in terms of velocity, the approach of [29] would require using the time step approximately 3 times smaller in comparison with that of the present method.

Additionally, we compared our results with the ones obtained using a semi-explicit PFEM strategy with no advection-diffusion decoupling [53]. The strategy used in the reference provides similar rate of convergence as our method, but exhibits a slightly smaller error for a given time step size. We note, however, that the approach of the reference [53] relies on applying Runge-Kutta scheme to the entire momentum equation and thus involves evaluation of the momentum equation residual 4 times, making it much more computationally expensive.

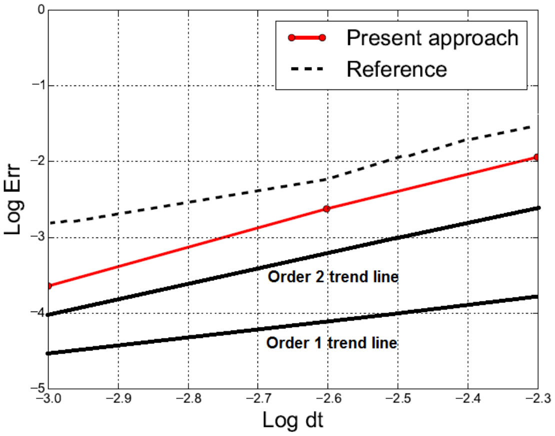


Figure 4: Flow in cavity. Convergence data (error in velocity at t=0.5 s vs time step). Comparison of the present approach with the semi-explicit PFEM based on streamline integration of particle positions [29].

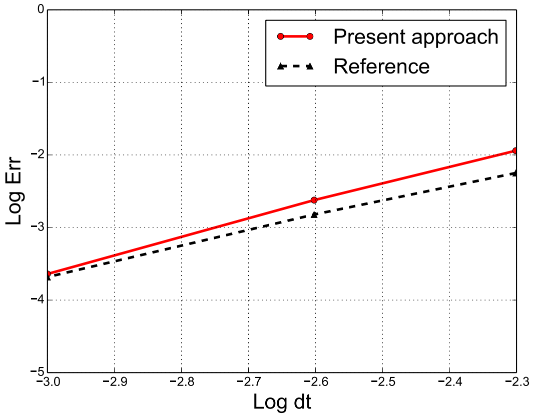


Figure 5: Flow in cavity. Convergence data (error in velocity at t=0.5 s vs time step). Comparison of the present approach with the semi-explicit approach based on applying Runge-Kutta integration to the entire momentum equation [53]

3.2. Second test with an analytic solution. Flow between parallel plates

To further test the accuracy of the approach, an example modeling a steady laminar flow between two parallel plates was solved. The settings used here are taken from [61]. The fluid is moving in horizontal direction between two parallel plates due to external uniform pressure of 160000 Pa prescribed at the inlet of the horizontal channel. The channel length is L=10 m and its horizontal walls (plates) are separated by D=1 m. No-slip boundary condition is prescribed at both plates. The fluid properties are: density ρ=1000 kg/m^3, dynamic viscosity μ=10^4 $Pa \cdot s$.

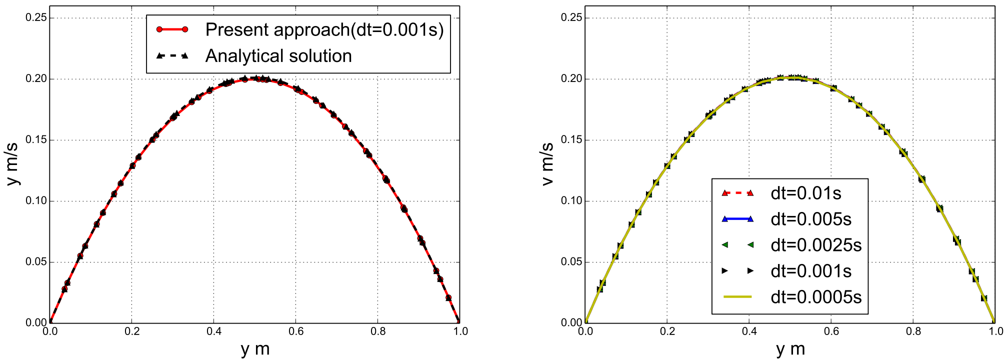
The exact solution for the horizontal velocity component along y-axis is:

$$\mathrm{v}_x(y) = \frac{1}{2\mu}\frac{\partial p}{\partial x}\left(y^2 - \frac{D^2}{4}\right) \tag{47}$$

The pressure gradient in our example is $\frac{\partial p}{\partial x} = \frac{P}{L} = 16000\ Pa/m$.

The problem is discretized with an unstructured and nearly uniform mesh of size h=0.05 m. Total simulation time is set to 1 s.

Fig. 6(a) shows velocity distribution along the vertical cut at x=5 m. Numerical simulation result obtained using dt=0.001 s is compared with the exact solution. The solutions show excellent agreement.



(a) Comparison with the analytic solution.

(b) Results obtained for different time step sizes.

Figure 6: Flow between parallel plates. Velocity profile along the vertical coordinate at x = 5 m.

Next, results obtained using different time steps are displayed (see Fig. 6(b)). In order to estimate the time accuracy, the area between the curves corresponding to the solution obtained for the given time step and the reference solution was used as an error measure. In order to exclude the influence of the spatial discretization error, the solution obtained using dt=0.00001 s

was taken as a reference solution. The convergence data (error versus time step size in logarithmic scale) is shown in Fig.7. One can observe that the convergence line slope corresponds to second order accuracy in time.

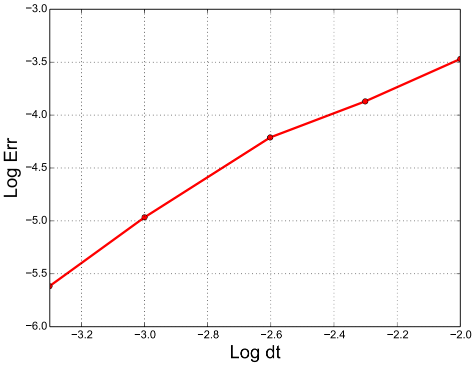


Figure 7: Flow between parallel plates. Error in horizontal velocity along the vertical cut at x=5 m vs time step size.

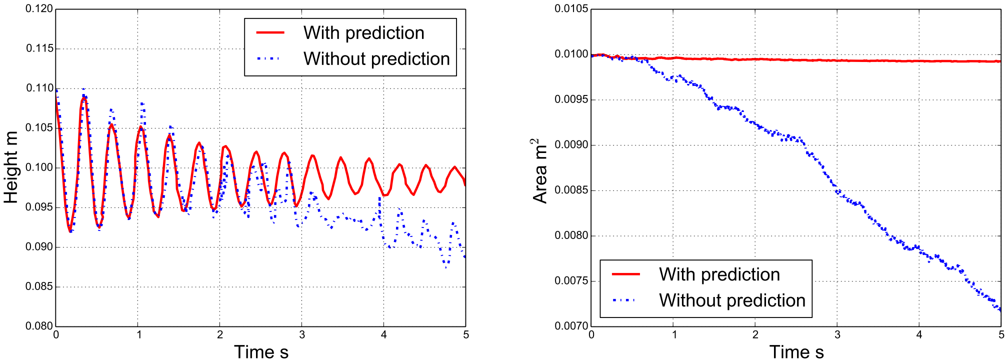
3.3. Sloshing

The last test aims at assessing the ability of the method to preserve mass in the simulation of flows with large free surface motions. The domain is essential a square of 0.1 m x 0.1 m, however its upper horizontal boundary is perturbed, being given a sinusoidal shape: $y_{up} = 0.1 + 0.01cos(\frac{\pi x}{0.1})$

The free surface shape has a sinusoidal [2] form. The properties of the fluid are: viscosity μ=0.1 kg/ms, density ρ=1000 kg/m^3 and gravity g_y=-10 m/s^2. For the prediction of the pressure the bulk modulus was set to be κ=10 kg/ms^2. The test was simulated for 5 s with dt=0.005 s. The domain was discretized using a triangular mesh with a 0.002 m mesh size.

Fig.8(a) shows the evolution of the free surface at the left wall of the container with and without the pressure estimation algorithm. Both methodologies present a good agreement until 2 s, where from that time on the version without pressure estimation starts to have a change at the amplitude and the frequency of oscillation of the free surface. This is due to the significant mass loss that sets on in the formulation without the pressure prediction(see Fig.8(b)). One can see that at the end of the simulation (t=5 s), the non-improved version of the model led to a loss of nearly thirty percent of its overall mass while the version with the prediction leads to a loss of about one percent only.

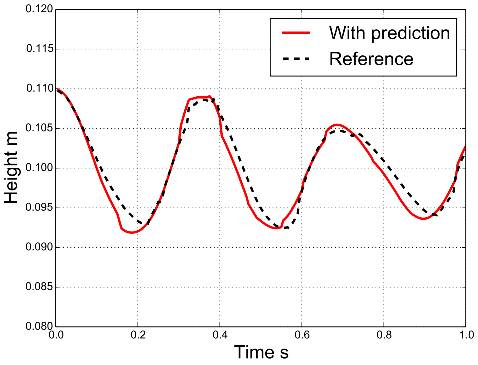
Fig.9 compares the results obtained using the pressure prediction algorithm with ones from [53]. A quick inspection of the figure reveals a good agreement with [53].



(a) Oscillation free surface.

(b) Variation of the domain area.

Figure 8: Fluid sloshing. Comparison between of the results obtained with and without pressure prediction.



(a) Variation of the surface.

Figure 9: Fluid sloshing. Comparison with reference [53].

4. SUMMARY AND CONCLUSIONS

In this paper, a new version of the explicit-implicit particle finite element method was introduced. The model was developed aiming at achieving second-order accuracy in time, an attractive feature distinguishing it from the former versions of the explicit-implicit PFEM. This was achieved by employing Strang operator splitting to Navier-Stokes equations, dividing them into two parts, namely the convective and the diffusive ones. While the diffusive part was solved on a "frozen" mesh using the Crank-Nicolson scheme, the convective part (coinciding in PFEM with the nodal movement) was integrated using the Runge-Kutta scheme. To solve the Stokes equation, the fractional step technique was employed. In addition, a completely explicit pressure prediction procedure was introduced in the first fractional step to improve mass conservation features of the method.

The results of numerical tests confirmed that, the developed strategy offers a second-order time accuracy and provides control over mass losses. In addition, a possible element inversion has been prevented due to the fact that the advective and the diffusive parts were treated separately.

Even though the proposed method constitutes an attractive version of the PFEM, it should be considered that it still relies on re-construction of the entire mesh. This step is generally non-parallelizable and remains the efficiency bottleneck of the model. Developing a method that optimizes the mesh re-construction (for example, by considering local re-meshing or re-connection, rather than re-constructing the entire mesh) defines the next step that must be considered when establishing a new generation of efficient Lagrangian fluid solvers.

The results of this work also reveal that when devising methods that include advection step solved in a Lagrangian fashion, applying high order operator splitting is essential for ensuring overall accuracy of the method. This may be also useful for those hybrid Lagrangian-Eulerian methods that resolve diffusion part of the problem on a fixed-grid or for other classes of Lagrangian numerical methods.

Acknowledgements

The authors acknowledges financial support from the *Ministerio de Ciencia, Innovacion e Universidades* of Spain via the Severo Ochoa Programme for Centres of Excellence in RD (referece: CEX2018-000797-S) as well as via AMADEUS project grant (reference: PGC2018-101655-B-I00).

Conflict of interest

The authors declare that they have no conflict of interest.

[1] B. Ramaswamy and M. Kawahara. Lagrangian finite element analysis applied to viscous free surface fluid flow. International Journal for Numerical Methods in Fluids, 7(9):953–984, 1987.

[2] R. Radovitzky and M. Ortiz. Lagrangian finite element analysis of newtonian fluid flows. International Journal for Numerical Methods in Engineering, 43(4):607–619, 1998.

[3] F. Muttin, T. Coupez, M. Bellet, and J.-L. Chenot. Lagrangian finite-element analysis of time-dependent viscous free-surface flow using an automatic remeshing technique: Application to metal casting flow. International Journal for Numerical Methods in Engineering, 36(12):2001–2015, 1993.

[4] A. Bennett. Lagrangian fluid dynamics. Cambridge University Press, 2006.

[5] S. R. Idelsohn, E. Oñate, and F. Del Pin. The particle finite element method: a powerful tool to solve incompressible flows with free-surfaces and breaking waves. International journal for numerical methods in engineering, 61(7):964–989, 2004.

[6] J. Donea, A. Huerta, J.P. Ponthot, and A. Rodríguez-Ferran. Arbitrary lagrangian–eulerian methods. Encyclopedia of Computational Mechanics Second Edition, pages 1–23, 2017.

[7] B. Delaunay et al. Sur la sphere vide. Izv. Akad. Nauk SSSR, Otdelenie Matematicheskii i Estestvennyka Nauk, 7(793-800):1–2, 1934.

[8] E. Oñate, S.R. Idelsohn, F. Del Pin, and R. Aubry. The particle finite element method: an overview. International Journal of Computational Methods, 1:267–307, 2004.

[9] S.R. Idelsohn, J. Marti, A. Souto-Iglesias, and E. Oñate. Interaction between an elastic structure and free-surface flows: experimental versus numerical comparisons using the PFEM. Computational Mechanics, 43(1):125–132, 2008.

[10] J. Marti, S. R. Idelsohn, A. Limache, N. Calvo, and J. D'elia. A fully coupled particle method for quasi incompressible fluid-hypoelastic structure interactions. 2006.

[11] S.R Idelsohn, J. Marti, A. Limache, and E. Oñate. Unified Lagrangian formulation for elastic solids and incompressible fluids. application to fluid-structure interaction problems via the PFEM. Computer Methods in Applied Mechanics and Engineering, 197:17621776, 2008.

[12] P. Ryzhakov, R. Rossi, S.R. Idelsohn, and E. Oñate. A monolithic lagrangian approach for fluid–structure interaction problems. Computational mechanics, 46(6):883–899, 2010.

[13] M. Cremonesi, A. Frangi, and U. Perego. A Lagrangian finite element approach for the analysis of fluid–structure interaction problems. International Journal for Numerical Methods in Engineering, 84(5):610–630, 2010.

[14] M. Cremonesi, S. Meduri, U. Perego, and A. Frangi. An explicit Lagrangian finite element method for free-surface weakly compressible flows. Computational Particle Mechanics, 4(3):357–369, Jul 2017.

[15] M.L. Cerquaglia, G. Deliége, R. Boman, V. Terrapon, and J-P Ponthot. Free-slip boundary conditions for simulating free-surface incompressible flows through the particle finite element method. International Journal for Numerical Methods in Engineering, 110(10):921–946, 2017.

[16] M.L. Cerquaglia, D. Thomas, R. Boman, V. Terrapon, and J-P Ponthot. A fully partitioned lagrangian framework for fsi problems characterized by free surfaces, large solid deformations and displacements, and strong added-mass effects. Computer Methods in Applied Mechanics and Engineering, 348:409–442, 2019.

[17] S.R. Idelsohn, M. Mier-Torrecilla, and E. Oñate. Multi-fluid flows with the particle finite element method. Computer methods in applied mechanics and engineering, 198(33-36):2750–2767, 2009.

[18] S. R. Idelsohn, M. Mier-Torrecilla, N. Nigro, and E. Oñate. On the analysis of heterogeneous fluids with jumps in the viscosity using a discontinuous pressure field. Computational Mechanics, 46(1):115–124, 2010.

[19] S. R. Idelsohn, M. Mier-Torrecilla, J. Marti, and E. Oñate. The particle finite element method for multi-fluid flows. In Eugenio Oñate and Roger Owen, editors, Particle-Based Methods: Fundamentals and Applications, pages 135–158. Springer Netherlands, Dordrecht, 2011.

[20] P. Ryzhakov and A. Jarauta. An embedded approach for immiscible multi-fluid problems. International Journal for Numerical Methods in Fluids, 81(6):357–376, 2016.

[21] E. Oñate, J. Rojek, M. Chiumenti, S.R. Idelsohn, F. Del Pin, and R. Aubry. Advances in stabilized finite element and particle methods for bulk forming processes. Computer methods in applied mechanics and engineering, 195(48-49):6750–6777, 2006.

[22] E Oñate, J Rojek, M Chiumenti, SR Idelsohn, F Del Pin, and R Aubry. Advances in stabilized finite element and particle methods for bulk forming processes. Computer methods in applied mechanics and engineering, 195(48-49):6750–6777, 2006.

[23] P. Ryzhakov. An axisymmetric PFEM formulation for bottle forming simulation. Computational Particle Mechanics, 4(1):3–12, 2017.

[24] P. Ryzhakov, J. Garcia, and E. Oñate. Lagrangian finite element model for the 3D simulation of glass forming processes. Computers & Structures, 177:126–140, 2016.

[25] M. Hyre. Numerical simulation of glass forming and conditioning. Journal of the American Ceramic Society, 85(5):1047–1056, 2002.

[26] E. Feulvarch, N. Moulin, P. Saillard, T. Lornage, and J.-M. Bergheau. 3d simulation of glass forming process. Journal of materials processing technology, 164:1197–1203, 2005.

[27] P. Ryzhakov, J. Marti, S.R. Idelsohn, and E. Oñate. Fast fluid–structure interaction simulations using a displacement-based finite element model equipped with an explicit streamline integration prediction. Computer Methods in Applied Mechanics and Engineering, 315:1080–1097, 2017.

[28] S.R. Idelsohn, J. Marti, P. Becker, and E. Oñate. Analysis of multifluid flows with large time steps using the particle finite element method.

International Journal for Numerical Methods in Fluids, 75(9):621–644, 2014.

[29] P. Becker. An enhanced Particle Finite Element Method with special emphasis on landslides and debris flows. PhD thesis, Universitat Politecnica de Catalunya, 2015.

[30] S.R. Idelsohn, N. Nigro, J. Gimenez, R. Rossi, and J. Marti. A fast and accurate method to solve the incompressible Navier-Stokes equations. Engineering Computations, 30(2):197–222, 2013.

[31] Y. Yazici. Operator splitting methods for differential equations. PhD thesis, Izmir Institute of Technology, 2010.

[32] Yu A Semenov. On the lie-trotter theorems in l (p) spaces. Lett. Math. Phys., 1:379–385, 1977.

[33] R. Omer, E. Bashier, and A. I. Arbab. Numerical solutions of a system of odes based on lie-trotter and strang operator-splitting methods. Universal Journal of Computational Mathematics, 5(2):20–24, 2017.

[34] A.J. Chorin. A numerical method for solving incompressible viscous problems. Journal of Computational Physics, 2:12–26, 1967.

[35] N.N. Yanenko. The method of fractional steps. The solution of problems of mathematical physics in several variables. Springer edition, 1971. translated from Russian by T. Cheron.

[36] R. Temam. Sur lapproximation de la solution des equations de Navier-Stokes par la methode des pase fractionaires. Archives for Rational Mechanics and Analysis, 32:135–153, 1969.

[37] R. Codina. A stabilized finite element method for generalized stationary incompressible flows. Computer Methods in Applied Mechanics and Engineering, 190(20-21):2681 – 2706, 2001.

[38] P. Ryzhakov, E. Oñate, R. Rossi, and S. Idelsohn. Improving mass conservation in simulation of incompressible flows. International journal for numerical methods in engineering, 90(12):1435–1451, 2012.

[39] A. Franci, E. Oñate, and J.M. Carbonell. On the effect of the bulk tangent matrix in partitioned solution schemes for nearly incompressible fluids. International Journal for Numerical Methods in Engineering, 102(3-4):257–277, 2015.

[40] G Strang. Essays in Linear Algebra. 2012.

[41] A. Chertock, C. R. Doering, E. Kashdan, and A. Kurganov. A fast explicit operator splitting method for passive scalar advection. Journal of Scientific Computing, 45(1):200–214, Oct 2010.

[42] R. Omer, E. Bashier, and A. Arbab. Numerical solutions of a system of odes based on lie-trotter and strang operator-splitting methods. Universal Journal of Computational Mathematics, 5:20–, 03 2017.

[43] S. Blanes, F. Casas, and A. Murua. On the linear stability of splitting methods. Foundations of Computational Mathematics, 8(3):357–393, Jun 2008.

[44] A. Bátkai, P. Csomós, and G. Nickel. Operator splittings and spatial approximations for evolution equations. Journal of Evolution Equations, 9(3):613–636, Sep 2009.

[45] S. MacNamara and G. Strang. Operator splitting. In Splitting Methods in Communication, Imaging, Science, and Engineering, pages 95–114. Springer, 2016.

[46] G. Strang. On the construction and comparison of difference schemes. SIAM Journal on Numerical Analysis, 5(3):506–517, 1968.

[47] M. Hayashi, K. Hatanaka, and M. Kawahara. Lagrangian finite element method for free surface Navier-Stokes flow using fractional step methods. International journal for numerical methods in fluids, 13(7):805–840, 1991.

[48] M. Zhu and M. H. Scott. Improved fractional step method for simulating fluid-structure interaction using the pfem. International Journal for Numerical Methods in Engineering, 99(12):925–944, 2014.

[49] P. Ryzhakov. A modified fractional step method for fluid–structure interaction problems. Revista Internacional de Métodos Numéricos para Cálculo y Diseño en Ingeniería, 33(1-2):58–64, 2017.

[50] P. Ryzhakov, E. Oñate, R. Rossi, and S.R. Idelsohn. Improving mass conservation in simulation of incompressible flows. International Journal for Numerical Methods in Engineering, 90(12):1435–1451.

[51] P.K. Kundu and I.M. Cohen. Fluid mechanics. Academic Press, 2002.

[52] T. J. R. Hughes. Multiscale phenomena: Green's functions, the Dirichlet-to-Neumann formulation, subgrid scale models, bubbles and the origins of stabilized methods. Computer Methods in Applied Mechanics and Engineering, 73:387–401, 1995.

[53] J. Marti and P. Ryzhakov. An explicit-implicit finite element model for the numerical solution of incompressible navierstokes equations on moving grids. Computer Methods in Applied Mechanics and Engineering, 350:750 – 765, 2019.

[54] Accuracy in 3d particle tracing. In Mathematical Visualization, pages 329–341. Springer, 1998.

[55] N. Dialami, M. Chiumenti, M. Cervera, C. de Saracibar, and J.P. Ponthot. Material flow visualization in friction stir welding via particle tracing. International Journal of Material Forming, 8(2):167–181, 2015.

[56] H. Edelsbrunner and T.S. Tan. An upper bound for conforming delaunay triangulations. Discrete & Computational Geometry, 10(2):197–213, Aug 1993.

[57] H. Edelsbrunner and E. Mücke. Three-dimensional alpha shapes. ACM Trans. Graph., 13(1):43–72, January 1994.

[58] J. Marti and S.R. Idelsohn. Monolythic method for the solution of the fluid-structure interaction problem. In XIX Congreso sobre Métodos Numéricos y sus Aplicaciones (ENIEF 2007), Córdoba, Córdoba, Argentina, October 2007. AMCA (Asociación de Mecánica Computacional Argentina).

[59] P. Dadvand, R. Rossi, and E. Oñate. An object-oriented environment for developing finite element codes for multi-disciplinary applications. Archieves of Computational Methods in Engineering, 17/3:253–297, 2010.

[60] Kratos Multiphysics at GitHub. https://github.com/KratosMultiphysics/Kratos. Accessed: 2020-01-03.

[61] M. Zhu and M. H. Scott. Unified fractional step method for Lagrangian analysis of quasi-incompressible fluid and nonlinear structure interaction using the pfem. International Journal for Numerical Methods in Engineering, 109(9):1219–1236, 2017.