Computers and Fluids 313 (2026) 107066

Contents lists available at ScienceDirect

Computers and Fluids

journal homepage: www.elsevier.com/locate/compfluid

# ALE–SPH formulation with 3D parallel localized implicit iterative particle shifting: From particle regularity to industrial applications

Mohamad Amin Ghazi [a], Renato Vacondio [a,*], Jean-Christophe Marongiu [b], Eduardo Di Costanzo [b,c]

[a] Department of Civil Engineering and Architecture, University of Parma, Parma, Italy
[b] R&D Department, Andritz Hydro, Vevey, Switzerland
[c] Hamburg University of Technology, Institute for Fluid Dynamics and Ship Theory, Hamburg, Germany

ARTICLE INFO

*Keywords:*
ALE-SPH
Localized implicit iterative particle shifting
Advection correction step
Parallel computing
Industrial application
Pelton turbine

ABSTRACT

This paper presents a three-dimensional formulation of the Localized Novel Implicit Iterative Particle Shifting (L-NIIPS) method within the Arbitrary Lagrangian–Eulerian Smoothed Particle Hydrodynamics (ALE-SPH) framework to minimize particle concentration gradients, thereby improving the regularity of the particle distribution. A multi-node parallelization approach is introduced for the L-NIIPS approach to efficiently handle complex 3D cases. The methodology also incorporates wall contribution terms into the Advection Correction Step (ACS) formulation to ensure consistent transport of physical quantities through the boundary integral method, thereby eliminating pressure oscillations near solid boundaries and yielding noise-free pressure fields. Through numerical evaluation, optimal L-NIIPS parameters are identified that strike a balance among accuracy, computational cost, and particle distribution regularity, proving robust across different flow configurations and eliminating the need for case-specific tuning. Validation is performed using established SPH benchmarks, including the Taylor–Green Vortex (TGV), moving square box, and 2D/3D impinging jets on a flat plate and further assessed through a demanding 3D Pelton bucket simulation. Results confirm that the proposed methodology significantly enhances accuracy and smoothness of the physical fields while maintaining a manageable computational overhead, making it suitable for industrial applications.

## 1. Introduction

Smoothed Particle Hydrodynamics (SPH) is a particle-based meshless method, originally formulated for astrophysical problems by Gingold et al. [1], and Lucy [2]. In practice, standard SPH schemes suffer from a lack of zeroth-order consistency and various numerical instabilities, such as particle clustering, tensile instability, and difficulties in handling shock waves. To mitigate some of these limitations, the Arbitrary Lagrangian–Eulerian SPH (ALE-SPH) framework was proposed by Vila [3] and later extended to the Weakly Compressible SPH (WCSPH) context by Oger et al. [4]. This approach offers advantages over purely Lagrangian SPH, particularly in cases involving strong deformations and complex interface dynamics. More recently, Michel et al. [5] introduced the Renormalized High-Order Diffusive SPH (RHOD-SPH) scheme, which combines a non-conservative pressure gradient, Riemann solvers, and advanced Particle Shifting Techniques (PSTs) to achieve high-order accuracy with reduced dissipation and suppressed tensile instabilities [6].

Furthermore, the integration of Riemann solvers into SPH was first introduced by Monaghan [7]. Later, the Godunov SPH (GSPH) method was proposed by Inutsuka [8] to enhance shock-capturing capabilities and ensure exact conservation across particle interfaces, and was further improved by Parshikov and Medin [9]. Cha and Whitworth [10], as well as Puri and Ramachandran [11], incorporated approximate Riemann solvers to strike a balance between accuracy and computational efficiency. The main advantage of this formulation is that stability is achieved without introducing any artificial viscosity parameter. Nevertheless, Riemann-solver SPH schemes tend to be excessively diffusive, even when employing Godunov's flux based on the exact Riemann solver [3].

The integration of high-order reconstruction techniques into ALE-SPH frameworks marked a significant advancement toward achieving higher-order spatial accuracy. Cleary and Monaghan [12], using the Monotonic Upstream-Centered Schemes for Conservation Laws

* Corresponding author.
*E-mail address:* renato.vacondio@unipr.it (R. Vacondio).

https://doi.org/10.1016/j.compfluid.2026.107066
Received 22 December 2025; Received in revised form 13 March 2026; Accepted 1 April 2026
Available online 2 April 2026

(MUSCL) reconstruction commonly employed in finite difference methods (FDM) [13], demonstrated that such approaches elevate convergence from first to second order in smooth regions. Subsequent extensions by Avesani et al. [14,15] and Michel et al. [16] introduced WENO techniques for enhanced accuracy [17,18]. Nogueira et al. [19] presented a novel SPH scheme based on Moving Least Squares (MLS) reconstructions [14,20], combined with a posteriori Multi-dimensional Optimal Order Detection (MOOD) [21,22], resulting in an efficient, stable, and low-dissipation SPH formulation. However, despite these efforts to reduce numerical diffusion in the ALE-SPH framework, global error analysis reveals that the convergence rate remains below second order and exhibits an early saturation effect.

Additionally, another challenge in SPH arises from anisotropic particle distributions due to its Lagrangian nature [4,23–25]. Particle Shifting Techniques (PSTs) address this issue by relocating particles to restore a more uniform distribution, either through velocity corrections [4,23,26–28] or position modifications [24,25,29–34]. Recently, Michel et al. [28] proposed a PST formulation based on relative velocities to overcome limitations of Fick's law [25,35]. Despite these advances, achieving consistently uniform particle distributions remains a challenge for the explicit shifting methods as they cannot guarantee global minimization of particle concentration gradients. Consequently, Rastelli et al. [34] introduced the Implicit Iterative Particle Shifting (IIPS) method, which minimizes particle concentration gradients globally but faces challenges related to computational cost and stability. Recently, Ghazi et al. [33] proposed the Novel Implicit Iterative Particle Shifting (NIIPS) approach, which reformulated the IIPS using neighbor interactions to address the stability in particle distribution, while their localized version (L-NIIPS) reduces computational cost and constrained the shifting magnitude.

The correction of the physical quantities after particle shifting varies across methodologies. Xu et al. [24] and Suchde et al. [36] maintain strict conservation of mass and momentum by avoiding any post-shifting treatments. Taylor-series expansion methods, which linearly approximate physical quantities after shifting using local gradients, are commonly employed [25,31–33]. However, these methods often suffer from a lack of conservation. ALE-based approaches, such as those proposed by Sun et al. [26] and Antuono et al. [27], apply explicit correction terms to account for the convective transport. Although implementation details may vary across different studies, this structure is inherently limited by two main disadvantages: the introduction of a one-step delay in the coupled update of particle positions and field variables, and the requirement for a careful integration of the post-shifting terms directly within the SPH solver. To tackle these problems, Rastelli et al. [37,38] introduced Advection Correction Steps (ACS) and Moving Least Squares (MLS) re-interpolation strategies; however, both approaches introduced pressure field oscillations, and the MLS method proved to be computationally expensive.

Building on these observations, this work extends the L-NIIPS formulation [33] to three dimensions. Furthermore, to enable efficient 3D simulations, a parallelization strategy is proposed based on multi-node execution. The Advection Correction Step (ACS) [38] is employed to update the physical quantities as the post-shifting treatment. However, the ACS formulation is modified in this study to ensure accuracy also near the boundaries discretized with the boundary integral method, which is integrated into the ALE-SPH framework implemented in ASPHODEL [39,40], an in-house solver developed by the Andritz Hydro company for simulating Pelton turbines. Although SPH has been widely applied in various industrial contexts [41–50], this work places particular emphasis on Pelton turbine simulations, which hold particular relevance in hydraulic machinery [40,41,51–58]. Moreover, a numerical investigation is conducted on the predefined threshold for particle uniformity and the limitation on shifting magnitude. This investigation facilitates a balanced compromise between numerical accuracy, computational cost, and particle uniformity within the domain, thereby allowing users to employ the proposed method without the need for extensive parameter tuning across different cases for the L-NIIPS.

This paper is organized as follows. First, the governing equations of the ALE-SPH framework are introduced, formulated in terms of a Riemann problem. Next, a detailed derivation of the three-dimensional L-NIIPS method is presented. The subsequent section describes the parallelization strategy for solving the linear system arising from L-NIIPS, which is employed to compute the shifting vectors. Furthermore, the Advection Correction Step (ACS) is discussed, serving as a post-shifting treatment of the physical quantities in the presence of boundary integral method. Finally, the results of two-dimensional and three-dimensional simulations are reported in the last two sections, respectively.

## 2. Governing equations

The Arbitrary Lagrangian–Eulerian (ALE) approach was first introduced into the WCSPH framework by Vila [3] to discretize the Euler equations for fluid flow simulation. In ALE, the conservation laws are expressed for a control volume that moves with its own velocity, $\mathbf{v}_0$, called the transport velocity. The ALE-SPH formulation presented in this work is implemented in ASPHODEL [39,40], an in-house tool developed by Andritz Hydro company for simulating Pelton turbine flows [41,42,53,59,60]. The formulation is based on midpoint integration of the transport velocity $\mathbf{v}_0(\mathbf{r}_{ij})$, obtained by solving a local Riemann problem at the particle interfaces:

$$\begin{cases} \dfrac{d\mathbf{r}_i}{dt} = \mathbf{v}_{0i}, \\ \dfrac{d\omega_i}{dt} = \omega_i \displaystyle\sum_{j=1}^{J} 2\left(\mathbf{v}_0(\mathbf{r}_{ij}) - \mathbf{v}_{0i}\right)\cdot\nabla W_{ij}\,\omega_j, \\ \dfrac{d(\omega_i\rho_i)}{dt} = -\omega_i \displaystyle\sum_{j=1}^{J} 2\,\rho_{ij}^E\left(\mathbf{v}_{ij}^E - \mathbf{v}_0(\mathbf{r}_{ij})\right)\cdot\nabla W_{ij}\,\omega_j, \\ \dfrac{d(\omega_i\rho_i\mathbf{v}_i)}{dt} = -\omega_i \displaystyle\sum_{j=1}^{J}\left[2\,\rho_{ij}^E\,\mathbf{v}_{ij}^E \otimes \left(\mathbf{v}_{ij}^E - \mathbf{v}_0(\mathbf{r}_{ij})\right) + p_{ij}^E\,\mathbf{I}\right]\cdot\nabla W_{ij}\,\omega_j + \omega_i\rho_i\,\mathbf{g}. \end{cases} \tag{1}$$

In the presented formulation, $\mathbf{r}_i$, $\omega_i$, $\rho_i$ and $\mathbf{v}_i$ denote the position, volume, density and velocity of particle $i$, while $p_{ij}^E$, $\rho_{ij}^E$, and $\mathbf{v}_{ij}^E$, denoted with the superscript $E$, represent the pressure, density, and velocity at the interface between particles $i$ and $j$. In three dimensions, the velocity vector is given by $\mathbf{v} = \{u, v, w\}$ and the index $j$ denotes a particle in the neighborhood of particle $i$. For simplicity, in the above system of equations the boundary terms are excluded. Moreover, the gravitational acceleration is represented by the vector $\mathbf{g}$. In this work, the approximate Primitive Variable Riemann Solver (PVRS) is employed [61,62] which is a linearized solver where the Jacobian matrix is taken as constant and corresponds to a certain mean state between the left and right states, associated with particles $i$ and $j$. Moreover the Wendland $C6$ kernel [63] is employed, as recommended in [33,34] for its stability and smoothness properties. To improve spatial accuracy, the Monotone Upstream-centered Scheme for Conservation Laws (MUSCL), originally proposed by Van Leer [13], is adopted using the minmod slope limiter [64].

The governing equations in Eq. (1) are integrated in time with the third-order Runge–Kutta (RK3) scheme [65]. The fluid flow is assumed to be weakly compressible, with density variations constrained to around 1% [66]. As a result, the mass and momentum conservation equations are closed using an equation of state, typically the barotropic Tait equation:

$$p = \frac{c_0^2\rho_0}{\gamma}\left[\left(\frac{\rho}{\rho_0}\right)^{\gamma} - 1\right]. \tag{2}$$

Here, $\gamma = 7$ represents the polytropic exponent for water, while $c_0$, $\rho_0$, and $p_0$ correspond to the reference speed of sound, density, and pressure, respectively. To ensure weak compressibility, the reference speed

of sound is chosen as $c_0 \approx 10\|\mathbf{v}\|$, where $\|\mathbf{v}\|$ is the reference velocity. In the present study, it is defined as the prescribed inlet velocity for cases involving a jet, while for all remaining cases it is explicitly stated. This choice results in a Mach number $Ma = \|\mathbf{v}\|/c_0$ of approximately 0.1, aligning with the assumptions of a weakly compressible regime. Such a formulation imposes a restriction on the time step, determined by the Courant–Friedrichs–Lewy (CFL) condition. Therefore, the simulation time step $\Delta t$ is constrained by the CFL condition as follows:

$$\Delta t = CFL \cdot \min_i \left( \frac{h_i}{c_0 + \|\mathbf{v}_i\|} \right). \tag{3}$$

Here, the parameters are defined as follows: the smoothing length of particle $i$ is $h_i = 2\Delta$, where $\Delta$ is the particle spacing, and the CFL number is prescribed a value of 0.2. Additionally, solid boundaries are treated using boundary integral method [54], combined with interface-based Riemann solvers [40], which ensure robust performance for complex geometries, such as Pelton buckets, where the surfaces exhibit high curvature and thin structural features.

## 3. Localized novel implicit iterative particle shifting (L-NIIPS)

In this work, the Localized Novel Implicit Iterative Particle Shifting (L-NIIPS) method, originally proposed by Ghazi et al. [33] for two-dimensional simulations, is extended to three dimensions. To the best of our knowledge, this is the first derivation of an implicit iterative particle shifting approach to fully three-dimensional formulation. The L-NIIPS approach allows users to specify a maximum allowed shifting magnitude and relies on a localization procedure that identifies (flags) particles eligible for shifting when their value of $h \cdot |\nabla C|$ exceeds the prescribed threshold $L_\infty(\nabla C)_{\text{thr}}$, or when they are neighbors of such particles [33]. The shifting procedure is triggered when the global indicator $L_\infty(\nabla C)$, defined as the maximum value of $h \cdot |\nabla C_i|$, exceeds the pre-defined threshold $L_\infty(\nabla C)_{\text{thr}}$. The process continues until this condition is no longer met or the prescribed maximum shifting magnitude is reached.

The gradient of particle concentration in 3D is represented as a generic function of particle coordinates:

$$\nabla C(\mathbf{X}, \mathbf{Y}, \mathbf{Z}),$$

where,

$$\mathbf{X} = (x_1, \dots, x_i, \dots, x_n), \quad \mathbf{Y} = (y_1, \dots, y_i, \dots, y_n), \quad \mathbf{Z} = (z_1, \dots, z_i, \dots, z_n),$$

are the coordinate arrays of all fluid particles in the $x$-, $y$-, and $z$-directions, respectively. Each triplet $(x_i, y_i, z_i)$ denotes the position of particle $i$, and $n$ is the total number of chosen (flagged) particles in the system by the localization procedure. The aim of the implicit iterative particle shifting technique is to compute a new set of particle coordinates

$$\bar{\mathbf{X}} = (\bar{x}_1, \dots, \bar{x}_i, \dots, \bar{x}_n), \quad \bar{\mathbf{Y}} = (\bar{y}_1, \dots, \bar{y}_i, \dots, \bar{y}_n), \quad \bar{\mathbf{Z}} = (\bar{z}_1, \dots, \bar{z}_i, \dots, \bar{z}_n),$$

such that the concentration gradient $\nabla C$ is minimized for each particle. Since the implicit shifting formulation is purely geometric, aiming to determine particle positions such that $\nabla C \approx \nabla 1 = 0$ [33,34], it is inherently independent of the flow dynamics and physical parameters of the test case. This condition promotes a well-distributed particle arrangement throughout the simulation regardless of the flow complexity. Following this, the main objective of the shifting method is to:

$$\nabla C_i(\bar{\mathbf{X}}, \bar{\mathbf{Y}}, \bar{\mathbf{Z}}) = \left( \frac{\partial C_i}{\partial x}, \frac{\partial C_i}{\partial y}, \frac{\partial C_i}{\partial z} \right)(\bar{\mathbf{X}}, \bar{\mathbf{Y}}, \bar{\mathbf{Z}}) = 0, \quad i = 1, \dots, n. \tag{4}$$

Moreover, the gradient of particle concentration $\nabla C_i(\bar{\mathbf{X}}, \bar{\mathbf{Y}}, \bar{\mathbf{Z}})$ using the first-order Taylor-series expansion for particle $i$ is expressed as:

$$\begin{aligned} \nabla C_i(\bar{\mathbf{X}}, \bar{\mathbf{Y}}, \bar{\mathbf{Z}}) \approx \nabla C_i(\mathbf{X}, \mathbf{Y}, \mathbf{Z}) &+ \sum_{j=1}^{J} \frac{\partial}{\partial x_j} \left[ \nabla C_i(\mathbf{X}, \mathbf{Y}, \mathbf{Z}) \right] \delta x_j \\ &+ \sum_{j=1}^{J} \frac{\partial}{\partial y_j} \left[ \nabla C_i(\mathbf{X}, \mathbf{Y}, \mathbf{Z}) \right] \delta y_j \\ &+ \sum_{j=1}^{J} \frac{\partial}{\partial z_j} \left[ \nabla C_i(\mathbf{X}, \mathbf{Y}, \mathbf{Z}) \right] \delta z_j. \end{aligned} \tag{5}$$

Where $\delta x_j = \bar{x}_j - x_j$, $\delta y_j = \bar{y}_j - y_j$, and $\delta z_j = \bar{z}_j - z_j$ represent the components of the shifting vector for particle $j$. Using this notation, Eq. (5) can be decomposed into its directional components. For example, along the $x$-axis, the corresponding contribution reads:

$$\begin{aligned} \nabla_x C_i(\bar{\mathbf{X}}, \bar{\mathbf{Y}}, \bar{\mathbf{Z}}) \approx \nabla_x C_i(\mathbf{X}, \mathbf{Y}, \mathbf{Z}) &+ \sum_{j=1}^{J} \underbrace{\frac{\partial}{\partial x_j} \left[ \nabla_x C_i(\mathbf{X}, \mathbf{Y}, \mathbf{Z}) \right]}_{a_{ij}} \delta x_j \\ &+ \sum_{j=1}^{J} \underbrace{\frac{\partial}{\partial y_j} \left[ \nabla_x C_i(\mathbf{X}, \mathbf{Y}, \mathbf{Z}) \right]}_{b_{ij}} \delta y_j \\ &+ \sum_{j=1}^{J} \underbrace{\frac{\partial}{\partial z_j} \left[ \nabla_x C_i(\mathbf{X}, \mathbf{Y}, \mathbf{Z}) \right]}_{e_{ij}} \delta z_j. \end{aligned} \tag{6}$$

Similarly, the expression for the $y$-axis can be written as:

$$\begin{aligned} \nabla_y C_i(\bar{\mathbf{X}}, \bar{\mathbf{Y}}, \bar{\mathbf{Z}}) \approx \nabla_y C_i(\mathbf{X}, \mathbf{Y}, \mathbf{Z}) &+ \sum_{j=1}^{J} \underbrace{\frac{\partial}{\partial x_j} \left[ \nabla_y C_i(\mathbf{X}, \mathbf{Y}, \mathbf{Z}) \right]}_{b_{ij}} \delta x_j \\ &+ \sum_{j=1}^{J} \underbrace{\frac{\partial}{\partial y_j} \left[ \nabla_y C_i(\mathbf{X}, \mathbf{Y}, \mathbf{Z}) \right]}_{d_{ij}} \delta y_j \\ &+ \sum_{j=1}^{J} \underbrace{\frac{\partial}{\partial z_j} \left[ \nabla_y C_i(\mathbf{X}, \mathbf{Y}, \mathbf{Z}) \right]}_{g_{ij}} \delta z_j. \end{aligned} \tag{7}$$

And for the $z$-axis, the corresponding expression becomes:

$$\begin{aligned} \nabla_z C_i(\bar{\mathbf{X}}, \bar{\mathbf{Y}}, \bar{\mathbf{Z}}) \approx \nabla_z C_i(\mathbf{X}, \mathbf{Y}, \mathbf{Z}) &+ \sum_{j=1}^{J} \underbrace{\frac{\partial}{\partial x_j} \left[ \nabla_z C_i(\mathbf{X}, \mathbf{Y}, \mathbf{Z}) \right]}_{e_{ij}} \delta x_j \\ &+ \sum_{j=1}^{J} \underbrace{\frac{\partial}{\partial y_j} \left[ \nabla_z C_i(\mathbf{X}, \mathbf{Y}, \mathbf{Z}) \right]}_{g_{ij}} \delta y_j \\ &+ \sum_{j=1}^{J} \underbrace{\frac{\partial}{\partial z_j} \left[ \nabla_z C_i(\mathbf{X}, \mathbf{Y}, \mathbf{Z}) \right]}_{f_{ij}} \delta z_j. \end{aligned} \tag{8}$$

It is important to highlight that in Eqs. (6) and (7), the terms $a_{ij}$, $b_{ij}$, and $d_{ij}$ have already been defined along the $x$- and $y$-axes in [33] for the two-dimensional formulation of the L-NIIPS. In contrast, the final terms in both equations ($e_{ij}$ and $g_{ij}$), as well as the entire Eq. (8), result from the extension of the method to three dimensions. Considering that the $\nabla C$ components in three dimensions can be approximated using SPH spatial interpolation, the expression along the $z$-axis is given by:

$$\nabla_z C_i(\mathbf{X}, \mathbf{Y}, \mathbf{Z}) = \sum_{k=1}^{K} \frac{\partial W(x_{ik}, y_{ik}, z_{ik})}{\partial z_k} \omega_k. \tag{9}$$

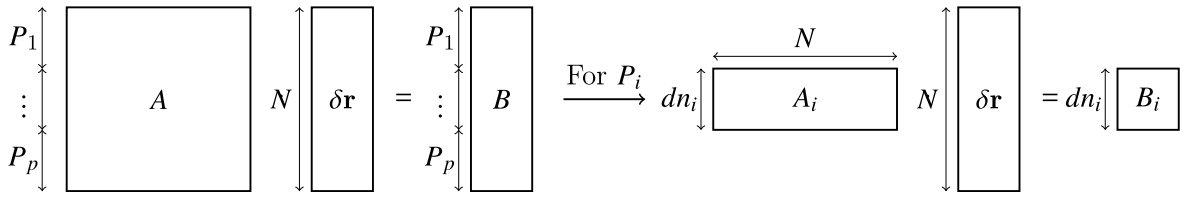



**Fig. 1.** Parallelization procedure. Matrix and vector distribution for a spatially parallel linear system $\mathbf{A}\delta\mathbf{r} = \mathbf{B}$ solved with BiCGSTAB across processes $P_i$ with $i \in [1, p]$.

$$
\underbrace{\begin{bmatrix}
\nabla_x^2 W_{11}\omega_1 - \phi_{1xx} & \nabla_{xy}^2 W_{11}\omega_1 - \phi_{1xy} & \nabla_{xz}^2 W_{11}\omega_1 - \phi_{1xz} & \cdots & \nabla_x^2 W_{1n}\omega_n & \nabla_{xy}^2 W_{1n}\omega_n & \nabla_{xz}^2 W_{1n}\omega_n \\
\nabla_{yx}^2 W_{11}\omega_1 - \phi_{1yx} & \nabla_y^2 W_{11}\omega_1 - \phi_{1yy} & \nabla_{yz}^2 W_{11}\omega_1 - \phi_{1yz} & \cdots & \nabla_{yx}^2 W_{1n}\omega_n & \nabla_y^2 W_{1n}\omega_n & \nabla_{yz}^2 W_{1n}\omega_n \\
\nabla_{zx}^2 W_{11}\omega_1 - \phi_{1zx} & \nabla_{zy}^2 W_{11}\omega_1 - \phi_{1zy} & \nabla_z^2 W_{11}\omega_1 - \phi_{1zz} & \cdots & \nabla_{zx}^2 W_{1n}\omega_n & \nabla_{zy}^2 W_{1n}\omega_n & \nabla_z^2 W_{1n}\omega_n \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
\nabla_x^2 W_{n1}\omega_1 & \nabla_{xy}^2 W_{n1}\omega_1 & \nabla_{xz}^2 W_{n1}\omega_1 & \cdots & \nabla_x^2 W_{nn}\omega_n - \phi_{nxx} & \nabla_{xy}^2 W_{nn}\omega_n - \phi_{nxy} & \nabla_{xz}^2 W_{nn}\omega_n - \phi_{nxz} \\
\nabla_{yx}^2 W_{n1}\omega_1 & \nabla_y^2 W_{n1}\omega_1 & \nabla_{yz}^2 W_{n1}\omega_1 & \cdots & \nabla_{yx}^2 W_{nn}\omega_n - \phi_{nyx} & \nabla_y^2 W_{nn}\omega_n - \phi_{nyy} & \nabla_{yz}^2 W_{nn}\omega_n - \phi_{nyz} \\
\nabla_{zx}^2 W_{n1}\omega_1 & \nabla_{zy}^2 W_{n1}\omega_1 & \nabla_z^2 W_{n1}\omega_1 & \cdots & \nabla_{zx}^2 W_{nn}\omega_n - \phi_{nzx} & \nabla_{zy}^2 W_{nn}\omega_n - \phi_{nzy} & \nabla_z^2 W_{nn}\omega_n - \phi_{nzz}
\end{bmatrix}}_{\mathbf{A}}
\underbrace{\begin{bmatrix} \delta x_1 \\ \delta y_1 \\ \delta z_1 \\ \vdots \\ \delta x_n \\ \delta y_n \\ \delta z_n \end{bmatrix}}_{\delta\mathbf{r}}
\approx
\underbrace{\begin{bmatrix} \nabla_x C_1 \\ \nabla_y C_1 \\ \nabla_z C_1 \\ \vdots \\ \nabla_x C_n \\ \nabla_y C_n \\ \nabla_z C_n \end{bmatrix}}_{\mathbf{B}}
\tag{14}
$$

**Box I.**

Here, $K$ denotes the number of neighboring particles within the kernel support of particle $i$, with $x_{ik} = x_i - x_k$, $y_{ik} = y_i - y_k$ and $z_{ik} = z_i - z_k$. By systematically expanding the terms on the right-hand side of Eq. (8) and applying in Eq. (9), the first term becomes:

$$
e_{ij} = \begin{cases} \dfrac{\partial^2 W(0,0,0)}{\partial x_j \partial z_j}\omega_j - \displaystyle\sum_{\substack{k=1 \\ k\neq i}}^{K} \frac{\partial^2 W(x_{ik}, y_{ik}, z_{ik})}{\partial x_j \partial z_j}\omega_k, & j = i \\ \dfrac{\partial^2 W(x_{ij}, y_{ij}, z_{ij})}{\partial x_j \partial z_j}\omega_j, & j \neq i \end{cases} \tag{10}
$$

the second term:

$$
g_{ij} = \begin{cases} \dfrac{\partial^2 W(0,0,0)}{\partial y_j \partial z_j}\omega_j - \displaystyle\sum_{\substack{k=1 \\ k\neq i}}^{K} \frac{\partial^2 W(x_{ik}, y_{ik}, z_{ik})}{\partial y_j \partial z_j}\omega_k, & j = i \\ \dfrac{\partial^2 W(x_{ij}, y_{ij}, z_{ij})}{\partial y_j \partial z_j}\omega_j, & j \neq i \end{cases} \tag{11}
$$

and finally for the third term:

$$
f_{ij} = \begin{cases} \dfrac{\partial^2 W(0,0,0)}{\partial z_j^2}\omega_j - \displaystyle\sum_{\substack{k=1 \\ k\neq i}}^{K} \frac{\partial^2 W(x_{ik}, y_{ik}, z_{ik})}{\partial z_j^2}\omega_k, & j = i \\ \dfrac{\partial^2 W(x_{ij}, y_{ij}, z_{ij})}{\partial z_j^2}\omega_j, & j \neq i \end{cases} \tag{12}
$$

Substituting these identities into Eq. (5) and applying the condition defined in Eq. (4) yields:

$$
\begin{cases}
\nabla_x C_i = \displaystyle\sum_j a_{ij}\,\delta x_j + \sum_j b_{ij}\,\delta y_j + \sum_j e_{ij}\,\delta z_j & x\text{-direction} \\
\nabla_y C_i = \displaystyle\sum_j b_{ij}\,\delta x_j + \sum_j d_{ij}\,\delta y_j + \sum_j g_{ij}\,\delta z_j & y\text{-direction} \\
\nabla_z C_i = \displaystyle\sum_j e_{ij}\,\delta x_j + \sum_j g_{ij}\,\delta y_j + \sum_j f_{ij}\,\delta z_j & z\text{-direction}
\end{cases} \tag{13}
$$

Eq. (13) is a system of linear equations of size $dn \times dn$, that can be written in the form of Eq. (14), where $dn$ is equal to the number of flagged particles ($n$) multiplied by the number of spatial dimensions ($d$, i.e. 2 or 3) in the L-NIIPS approach (see the equation in Box I).

The system presented in Eq. (14) represents a system of linear equations of the form $\mathbf{A}\,\delta\mathbf{r} = \mathbf{B}$, where $\mathbf{A}$ is a sparse matrix containing the second derivatives of the kernel function, $\delta\mathbf{r}$ is the unknown shifting vector, and $\mathbf{B}$ is the right-hand side vector containing the gradient of particle concentration. The system is solved iteratively inside the Newton–Raphson (NR) iterations using a BiCGSTAB solver to obtain $\delta\mathbf{r}$. The $\phi$ terms in Eq. (14), originally introduced by Ghazi et al. [33] for the $x$- and $y$-directions in two dimensions, are here extended to the $z$-direction as follows (see the equation in Box II).

## 4. Parallelization procedure of the L-NIIPS

In SPH simulations with a large number of particles, parallel computation is crucial for efficiency, especially in three-dimensional cases. Implementing the implicit iterative shifting techniques, which requires solving a linear system of equations, demands special considerations to ensure compatibility with the parallel structure of the solver.

When solving the linear system $\mathbf{A}\delta\mathbf{r} = \mathbf{B}$ in a spatially parallel environment, the sparse system matrix $\mathbf{A}$, the shifting vector $\delta\mathbf{r}$ and the right-hand side vector $\mathbf{B}$ are distributed among the available CPU processes. Each processor ($P_i$) handles a local matrix $A_i \in \mathbb{R}^{dn_i \times N}$, which corresponds to a block of $dn_i$ rows of the global matrix $A \in \mathbb{R}^{N \times N}$. Here, $dn_i$ denotes the number of flagged particles multiplied by the spatial dimensions ($d$, i.e. 2 or 3) in the sub-domain assigned to process $P_i$, and $N = \sum_{i=1}^{p} dn_i$. The right-hand-side vector follows the same distribution logic, while the solution vector is needed in full in each process. This row-wise distribution is illustrated in Fig. 1.

Communication is required at each iteration of the BiCGSTAB solver to exchange data corresponding to the matrix–vector products, which can represent a bottleneck (see Algorithm 1). In this work, all parallel operations are performed with the Message Passing Interface (MPI) protocol (*AllGather*() and *AllReduceSum*() in Algorithm 1) to maintain the same parallelization strategy as in the in-house Andritz solver, ASPHODEL. Local matrix and vector operations utilize the *Eigen* library [67].

$$\begin{cases} \phi_{1xz} = \sum_{\substack{k=1 \\ k\neq n}}^{K} \dfrac{\partial^2 W(x_{1k}, y_{1k}, z_{1k})}{\partial x_1 \partial z_1}\omega_k, \quad \phi_{1yz} = \sum_{\substack{k=1 \\ k\neq n}}^{K} \dfrac{\partial^2 W(x_{1k}, y_{1k}, z_{1k})}{\partial y_1 \partial z_1}\omega_k, \quad \phi_{1zz} = \sum_{\substack{k=1 \\ k\neq n}}^{K} \dfrac{\partial^2 W(x_{1k}, y_{1k}, z_{1k})}{\partial z_1^2}\omega_k, \\ \phi_{nxz} = \sum_{\substack{k=1 \\ k\neq n}}^{K} \dfrac{\partial^2 W(x_{nk}, y_{nk}, z_{nk})}{\partial x_n \partial z_n}\omega_k, \quad \phi_{nyz} = \sum_{\substack{k=1 \\ k\neq n}}^{K} \dfrac{\partial^2 W(x_{nk}, y_{nk}, z_{nk})}{\partial y_n \partial z_n}\omega_k, \quad \phi_{nzz} = \sum_{\substack{k=1 \\ k\neq n}}^{K} \dfrac{\partial^2 W(x_{nk}, y_{nk}, z_{nk})}{\partial z_n^2}\omega_k. \end{cases} \tag{15}$$

**Box II.**

**Algorithm 1** Parallel BiCGSTAB Solver

**Require:** Sparse matrix $\mathbf{A}^{local}$, right-hand side $\mathbf{B}^{local}$, tolerance $\texttt{tol} = 0.01$, maximum iterations $\texttt{maxIter} = 100$

**Ensure:** Approximate solution $\mathbf{x}$ (representing $\delta\mathbf{r}$, not to be confused with the solver residual)

Initialize $\mathbf{x} = \mathbf{0}$
Compute initial residual: $\mathbf{r}_0 = B^{local} - \mathbf{A}^{local} \cdot \mathbf{x}$, $\hat{\mathbf{r}} = \mathbf{r}_0$
Initialize search direction $\mathbf{p}_0 = \mathbf{r}_0$, auxiliary vector $\mathbf{v}_0 = \mathbf{0}$
Initialize scalars: $\rho_{\text{old}} = 1$, $\alpha = 1$, $\omega = 1$
**for** $i = 0$ to $\texttt{maxIter}$ **do**
  $\rho = \texttt{AllReduceSum}(\hat{\mathbf{r}}^\top \mathbf{r}_i)$ ▷ Global dot product across all processes
  **if** $i > 0$ **then**
    $\beta = \frac{\rho}{\rho_{\text{old}}} \frac{\alpha}{\omega}$ ▷ Compute scalar for updating $\mathbf{p}$
    $\mathbf{p}_i = \mathbf{r}_i + \beta(\mathbf{p}_{i-1} - \omega \mathbf{v}_{i-1})$ ▷ Update search direction
  **end if**
  $\mathbf{p}_i^{global} = \texttt{AllGather}(\mathbf{p}_i)$ ▷ Gather full vector $\mathbf{p}$ from all processes
  $\mathbf{v}_i = \mathbf{A}^{local} \cdot \mathbf{p}_i^{global}$ ▷ Matrix-vector product with distributed matrix
  $\alpha = \rho / \texttt{AllReduceSum}(\hat{\mathbf{r}}^\top \mathbf{v}_i)$ ▷ Compute step size scalar $\alpha$
  $\mathbf{s}_i = \mathbf{r}_i - \alpha \mathbf{v}_i$ ▷ Compute intermediate residual
  **if** $\sqrt{\texttt{AllReduceSum}(\mathbf{s}_i^\top \mathbf{s}_i)} < \texttt{tol}$ **then** ▷ Check convergence with residual norm
    $\mathbf{x} = \mathbf{x} + \alpha \cdot \mathbf{p}_i^{global}$ ▷ Update solution
    **return** $\mathbf{x}$
  **end if**
  $\mathbf{s}_i^{global} = \texttt{AllGather}(\mathbf{s}_i)$ ▷ Gather full vector $\mathbf{s}$
  $\mathbf{t}_i = \mathbf{A}^{local} \cdot \mathbf{s}_i^{global}$ ▷ Matrix-vector product for correction
  $\omega = \frac{\texttt{AllReduceSum}(\mathbf{t}_i^\top \mathbf{s}_i)}{\texttt{AllReduceSum}(\mathbf{t}_i^\top \mathbf{t}_i)}$ ▷ Compute scalar $\omega$
  $\mathbf{x} = \mathbf{x} + \alpha \cdot \mathbf{p}_i^{global} + \omega \cdot \mathbf{s}_i^{global}$ ▷ Update solution vector
  $\mathbf{r}_{i+1} = \mathbf{s}_i - \omega \mathbf{t}_i$ ▷ Update residual vector
  **if** $\sqrt{\texttt{AllReduceSum}(\mathbf{r}_{i+1}^\top \mathbf{r}_{i+1})} < \texttt{tol}$ **then** ▷ Check convergence
    **return** $\mathbf{x}$
  **end if**
  $\rho_{\text{old}} = \rho$ ▷ Update $\rho_{\text{old}}$ for next iteration
**end for**

## 5. Preliminary tests

In order to evaluate the effectiveness of the proposed three-dimensional shifting formulation together with the parallelization procedure for constructing the linear system of Eq. (14), distributing it across processors (see Fig. 1), and solving it with the BiCGSTAB solver described in Algorithm 1, a confined cubic domain of size $1 \times 1 \times 1$ m$^3$ ($L = 1$) is considered. One million fluid particles are initialized inside the box, which is bounded by solid wall particles, corresponding to a spatial resolution of $\Delta/L = 0.01$. The initial particle positions are randomly perturbed in all three directions by 20% of $\Delta$. Subsequently, the three-dimensional L-NIIPS algorithm is activated to assess its capability of recovering a uniform particle distribution. Tests were conducted on the University of Parma cluster, where each node is equipped with a dual 16-core AMD EPYC 7282 processor running at 2.8 GHz. Each test involved 250 Newton–Raphson iterations. In this preliminary assessment no limitation is imposed on the shifting magnitude, as the Navier–Stokes equations are not solved.

As shown in Fig. 2, a strong scaling comparison is carried out using 1, 2, 4, 8, and 16 processors to assess the efficiency of the parallelization strategy. The results are based on the average CPU time recorded over the 250 Newton–Raphson iterations. Increasing the number of processors yields a speedup close to the ideal trend. In particular, the simulation with 8 processors achieves a speedup of 6.81, while that with 16 processors reaches 12.73, both relative to the serial run. These results confirm the good scalability of the proposed parallelization procedure.

Furthermore, Fig. 3 illustrates the effectiveness of the proposed three-dimensional formulation together with the accuracy of the parallelization procedure. The value of $L_\infty(\nabla C)$ is measured at each Newton–Raphson iteration using 16 processors. Starting from a highly perturbed initial particle distribution, where $L_\infty(\nabla C) = 0.186$, the proposed shifting method reduces this value to $L_\infty(\nabla C) = 0.00406$ after 250 iterations, thereby demonstrating the capability of the three-dimensional L-NIIPS formulation to recover a uniform particle distribution. Moreover, the red triangles represent the same quantity obtained from the serial simulation, confirming the accuracy of the parallelization procedure, as no differences are observed between the results obtained in serial and with 16 processors.

## 6. Advection correction step (ACS)

To preserve the stability of the numerical scheme, the shifting vector obtained by solving the linear system in Eq. (14) is used to update the physical quantities through an Advection Correction Step (ACS). This correction procedure, originally proposed by Rastelli et al. [38], is derived from a fictitious time step applied to the ALE formulation in Eq. (1). A detailed derivation of the ACS formulation is provided in [38]. The advantage of this method over other post-shifting treatments lies in its non-intrusive nature, as it can be applied independently of the specific approximation adopted for the governing equations and SPH solver (e.g. Eq. (1)). Instead, it can immediately account for the change of the physical quantities due to the advection fluxes using the following equations:

$$\begin{cases} \delta\mathbf{r}_i = \bar{\mathbf{r}}_i - \mathbf{r}_i, & \text{(Position variation)} \\ \bar{\omega}_i - \omega_i = \omega_i \sum_{j\in F} (\delta\mathbf{r}_j - \delta\mathbf{r}_i)\cdot\nabla W_{ij}\,\omega_j, & \text{(Volume flux)} \\ \bar{\omega}_i\bar{\rho}_i - \omega_i\rho_i = -\omega_i \sum_{j\in F} \left(\rho_i\,\delta\mathbf{r}_i + \rho_j\,\delta\mathbf{r}_j\right)\cdot\nabla W_{ij}\,\omega_j, & \text{(Mass flux)} \\ \bar{\omega}_i\bar{\rho}_i\bar{\mathbf{v}}_i - \omega_i\rho_i\mathbf{v}_i = -\omega_i \sum_{j\in F} \left(\rho_i\,\mathbf{v}_i \otimes \delta\mathbf{r}_i + \rho_j\,\mathbf{v}_j \otimes \delta\mathbf{r}_j\right)\cdot\nabla W_{ij}\,\omega_j. & \text{(Momentum flux)} \end{cases} \tag{16}$$

In this formulation, indices i and $j \in \mathcal{F}$ refer to fluid particles, $\bar{\omega}_i$, $\bar{\rho}_i$, and $\bar{\mathbf{v}}_i$ represent the updated volume, density, and velocity of particle $i$ after the shifting procedure. However, the implementation of this post-shifting treatment by Rastelli et al. [38] resulted in pressure oscillations near solid walls when using the boundary integral method. Therefore, this formulation requires modification through the introduction of additional terms to accurately incorporate the influence of wall boundaries into the ACS update.

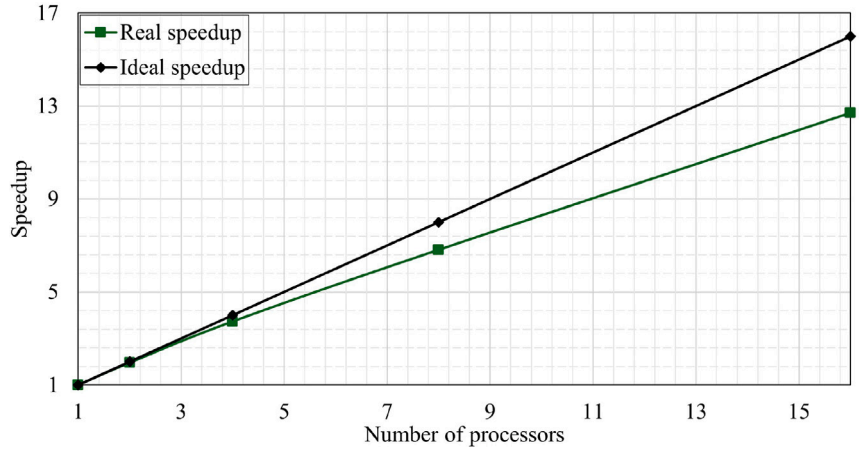



**Fig. 2.** Preliminary test. Strong scaling comparison of the parallelization procedure.

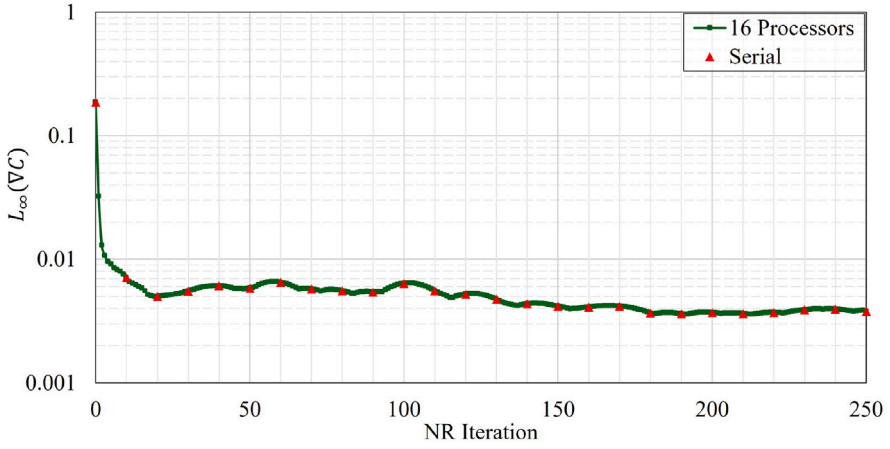



**Fig. 3.** Preliminary test. $L_\infty(\nabla C)$ comparison for the simulation with 16 processors (green squared line) and in serial (red triangles).

Consider the ALE–SPH formulation of Eq. (1), where the Riemann solution at the particle interface, $\boldsymbol{\Phi}_{ij}^{E} = (p_{ij}^{E}, \rho_{ij}^{E}, \mathbf{v}_{ij}^{E})$, is replaced by the particle-centered quantities $\rho_i$, $\rho_j$, $p_i$, $p_j$, $\mathbf{v}_i$, and $\mathbf{v}_j$. Incorporating the boundary integral terms and omitting gravitational and pressure forces in the momentum equation, Eq. (1) becomes:

$$
\begin{cases}
\dfrac{d\mathbf{r}_i}{dt} = \mathbf{v}_{0i}, \\
\dfrac{d\omega_i}{dt} = \omega_i \displaystyle\sum_{j\in\mathcal{F}} (\mathbf{v}_{0j} - \mathbf{v}_{0i}) \cdot \nabla W_{ij}\, \omega_j + \omega_i \sum_{j\in\mathcal{W}} (\mathbf{v}_{0j} - \mathbf{v}_{0i}) \cdot \mathbf{n}_{ij}\, W_{ij}\, \omega_j', \\
\dfrac{d(\omega_i \rho_i)}{dt} = -\omega_i \displaystyle\sum_{j\in\mathcal{F}} \big(\rho_i(\mathbf{v}_i - \mathbf{v}_{0i}) + \rho_j(\mathbf{v}_j - \mathbf{v}_{0j})\big) \cdot \nabla W_{ij}\, \omega_j \\
\qquad\qquad - \omega_i \displaystyle\sum_{j\in\mathcal{W}} \big(\rho_i(\mathbf{v}_i - \mathbf{v}_{0i}) + \rho_j(\mathbf{v}_j - \mathbf{v}_{0j})\big) \cdot \mathbf{n}_{ij}\, W_{ij}\, \omega_j', \\
\dfrac{d(\omega_i \rho_i \mathbf{v}_i)}{dt} = -\omega_i \displaystyle\sum_{j\in\mathcal{F}} \big[\rho_i \mathbf{v}_i \otimes (\mathbf{v}_i - \mathbf{v}_{0i}) + \rho_j \mathbf{v}_j \otimes (\mathbf{v}_j - \mathbf{v}_{0j})\big] \cdot \nabla W_{ij}\, \omega_j \\
\qquad\qquad - \omega_i \displaystyle\sum_{j\in\mathcal{W}} \big[\rho_i \mathbf{v}_i \otimes (\mathbf{v}_i - \mathbf{v}_{0i}) + \rho_j \mathbf{v}_j \otimes (\mathbf{v}_j - \mathbf{v}_{0j})\big] \cdot \mathbf{n}_{ij}\, W_{ij}\, \omega_j'.
\end{cases}
\tag{17}
$$

In this equation, indices $j \in \mathcal{W}$ denote neighboring wall particles, and $\omega_j'$ represents the area of the corresponding surface element (segment) when the boundary integral method is used. The vector $\mathbf{n}_{ij}$ indicates the wall normal, oriented from wall particle $j$ toward fluid particle $i$. Let us now consider this formulation at a fictitious time step $\Delta t^*$, during which the physical time of the simulation is frozen [38], and the only movement occurs through the shifting vectors. Consequently, we can replace $\mathbf{v}_{0i}$ and $\mathbf{v}_{0j}$ with $\frac{\delta \mathbf{r}_i}{\Delta t^*}$ and $\frac{\delta \mathbf{r}_j}{\Delta t^*}$, respectively, as follows:

$$
\begin{cases}
\dfrac{d\omega_i}{\Delta t^*} = \omega_i \displaystyle\sum_{j\in\mathcal{F}} \left(\frac{\delta \mathbf{r}_i}{\Delta t^*} - \frac{\delta \mathbf{r}_j}{\Delta t^*}\right) \cdot \nabla W_{ij}\, \omega_j + \omega_i \sum_{j\in\mathcal{W}} \left(\frac{\delta \mathbf{r}_i}{\Delta t^*} - \frac{\delta \mathbf{r}_j}{\Delta t^*}\right) \cdot n_{ij}\, W_{ij}\, \omega_j', \\
\dfrac{d(\omega_i \rho_i)}{\Delta t^*} = -\omega_i \displaystyle\sum_{j\in\mathcal{F}} \left(\rho_i \frac{\delta \mathbf{r}_i}{\Delta t^*} + \rho_j \frac{\delta \mathbf{r}_j}{\Delta t^*}\right) \cdot \nabla W_{ij}\, \omega_j \\
\qquad\qquad - \omega_i \displaystyle\sum_{j\in\mathcal{W}} \left(\rho_i \frac{\delta \mathbf{r}_i}{\Delta t^*} + \rho_j \frac{\delta \mathbf{r}_j}{\Delta t^*}\right) \cdot n_{ij}\, W_{ij}\, \omega_j', \\
\dfrac{d(\omega_i \rho_i \mathbf{v}_i)}{\Delta t^*} = -\omega_i \displaystyle\sum_{j\in\mathcal{F}} \left(\rho_i \mathbf{v}_i \otimes \frac{\delta \mathbf{r}_i}{\Delta t^*} + \rho_j \mathbf{v}_j \otimes \frac{\delta \mathbf{r}_j}{\Delta t^*}\right) \cdot \nabla W_{ij}\, \omega_j \\
\qquad\qquad - \omega_i \displaystyle\sum_{j\in\mathcal{W}} \left(\rho_i \mathbf{v}_i \otimes \frac{\delta \mathbf{r}_i}{\Delta t^*} + \rho_j \mathbf{v}_j \otimes \frac{\delta \mathbf{r}_j}{\Delta t^*}\right) \cdot n_{ij}\, W_{ij}\, \omega_j'.
\end{cases}
\tag{18}
$$

By canceling the fictitious time step $\Delta t^*$ from both sides of the equations, the fluid–fluid interaction becomes identical to that obtained by Rastelli et al. [38]. Considering only the fluid–wall interactions, the wall contribution terms in the induced advection by the shifting procedure can be expressed as follows:

$$
\begin{cases}
\bar{\omega}_i - \omega_i = \omega_i \displaystyle\sum_{j\in\mathcal{W}} (\delta \mathbf{r}_j - \delta \mathbf{r}_i) \cdot \mathbf{n}_{ij}\, W_{ij}\, \omega_j', \\
\bar{\omega}_i\, \bar{\rho}_i - \omega_i\, \rho_i = -\omega_i \displaystyle\sum_{j\in\mathcal{W}} \big(\rho_i\, \delta \mathbf{r}_i + \rho_j\, \delta \mathbf{r}_j\big) \cdot \mathbf{n}_{ij}\, W_{ij}\, \omega_j', \\
\bar{\omega}_i\, \bar{\rho}_i\, \bar{\mathbf{v}}_i - \omega_i\, \rho_i\, \mathbf{v}_i = -\omega_i \displaystyle\sum_{j\in\mathcal{W}} \big(\rho_i\, \mathbf{v}_i \otimes \delta \mathbf{r}_i + \rho_j\, \mathbf{v}_j \otimes \delta \mathbf{r}_j\big) \cdot \mathbf{n}_{ij}\, W_{ij}\, \omega_j'.
\end{cases}
\tag{19}
$$

It is important to note that in the above formulation, $\delta\mathbf{r}_j$ refers to the shifting vector associated with wall particles. Since wall particles are not shifted in the L-NIIPS procedure, their displacement vector is zero, i.e., $\delta\mathbf{r}_j = \mathbf{0}$. Although these particles are not shifted, they do influence the system through the vector $\mathbf{B}$ and the $\boldsymbol{\phi}$ terms located on the diagonal of the $\mathbf{A}$ matrix in Eq. (14). As a result, the contributions from wall particles in Eq. (19) can be combined with those in Eq. (16), leading to the following formulation for ACS including the boundary integral treatment. Given that $\delta\mathbf{r}_j = \mathbf{0}$ for wall particles, the resulting expression simplifies accordingly and yields the following complete form of equation for ACS:

$$\begin{cases} \bar{\omega}_i - \omega_i = \omega_i \sum_{j\in\mathcal{F}} (\delta\mathbf{r}_j - \delta\mathbf{r}_i)\cdot\nabla W_{ij}\,\omega_j + \omega_i \sum_{j\in\mathcal{W}} (-\delta\mathbf{r}_i)\cdot\mathbf{n}_{ij} W_{ij}\,\omega'_j, \\ \bar{\omega}_i\bar{\rho}_i - \omega_i\rho_i = -\omega_i \sum_{j\in\mathcal{F}} \left(\rho_i\delta\mathbf{r}_i + \rho_j\delta\mathbf{r}_j\right)\cdot\nabla W_{ij}\,\omega_j - \omega_i \sum_{j\in\mathcal{W}} \rho_i\delta\mathbf{r}_i\cdot\mathbf{n}_{ij} W_{ij}\,\omega'_j, \\ \bar{\omega}_i\bar{\rho}_i\bar{\mathbf{v}}_i - \omega_i\rho_i\mathbf{v}_i = \underbrace{-\omega_i \sum_{j\in\mathcal{F}} \left(\rho_i\mathbf{v}_i\otimes\delta\mathbf{r}_i + \rho_j\mathbf{v}_j\otimes\delta\mathbf{r}_j\right)\cdot\nabla W_{ij}\,\omega_j}_{\text{Fluid contribution}} \\ \qquad \underbrace{- \omega_i \sum_{j\in\mathcal{W}} \rho_i\mathbf{v}_i\otimes\delta\mathbf{r}_i\cdot\mathbf{n}_{ij} W_{ij}\,\omega'_j}_{\text{Wall contribution}}. \end{cases} \tag{20}$$

In this equation, the wall contribution terms account for the interaction between fluid and wall particles during the ACS update. These wall contribution terms ensure that the influence of wall particles is accurately incorporated into the ACS update, even though the wall particles themselves remain fixed and are not subject to shifting. The benefits of incorporating wall contributions will be demonstrated in the results section. It is also noted that the L-NIIPS approach is consistently applied with the complete formulation of ACS in Eq. (20), unless stated otherwise.

## 7. 2D results

In this paper, three test cases are evaluated in 2D: the Taylor–Green Vortex (TGV), the moving square box [68], and an impinging jet. The TGV case is employed to assess the performance of the L-NIIPS approach in conjunction with the ACS in scenarios involving highly distorted particle distributions. To investigate the influence of a moving object and wall boundaries, the moving square box scenario serves as an appropriate benchmark. Additionally, an impinging jet simulation is conducted to evaluate the robustness of the method under high-impact conditions on a flat plate in the presence of a free-surface condition.

### 7.1. 2D Taylor–Green vortex (TGV)

In fluid dynamics, the TGV represents a decaying vortex flow that admits an exact solution to the incompressible Navier–Stokes equations. This test case provides a suitable benchmark for assessing particle shifting methods, as it continuously generates strong vortical structures that induce significant particle disorder. Consequently, the shifting procedure must remain actively enabled throughout the simulation to maintain particle distribution regularity. However, in this work, the reference solution of weakly compressible approximation recently derived by Antuono et al. [69] is employed. The computational domain is periodic with side length $L = 1$ m. The reference velocity $U$ is set to 1 m/s, and the reference density $\rho$ is 1 kg/m$^3$. The weakly compressible approximation of the analytical solution for TGV [69] is given as follows:

$$\begin{cases} u = u_0 + \varepsilon u_1 + \mathcal{O}(\varepsilon^2), \\ v = v_0 + \varepsilon v_1 + \mathcal{O}(\varepsilon^2), \\ p = p_0 + \varepsilon p_1 + \mathcal{O}(\varepsilon^2). \end{cases} \tag{21}$$

Here, $\varepsilon = \mathrm{Ma}^2 \ll 1$. Moreover, $u_0$, $v_0$, and $p_0$ represent the incompressible analytical solutions for velocity and pressure, while $u_1$, $v_1$, and $p_1$ denote the weakly compressible correction terms [69]. The incompressible analytical solution, assuming unit length, unit reference velocity, and unit density, is given as follows:

$$\begin{cases} u_0 = F\sin(2\pi x)\cos(2\pi y), \\ v_0 = -F\cos(2\pi x)\sin(2\pi y), \\ p_0 = \dfrac{F^2}{4}\left[\cos(4\pi x) + \cos(4\pi y)\right]. \end{cases} \tag{22}$$

Where $F = \exp(-8\pi^2\nu t)$, $x$ and $y$ denote the spatial coordinates, and $\nu$ represents the kinematic viscosity. Additionally, the weakly compressible correction terms in Eq. (21) are defined as follows [69]:

$$\begin{cases} u_1 = \nu\pi F^2\sin(4\pi x) - \dfrac{F^3}{16}\sin(2\pi x)\cos(2\pi y)\left[\cos(4\pi x) - 5\cos(4\pi y) + 5\right] + \mathcal{O}(\nu), \\ v_1 = \nu\pi F^2\sin(4\pi y) + \dfrac{F^3}{16}\cos(2\pi x)\sin(2\pi y)\left[-5\cos(4\pi x) + \cos(4\pi y) + 5\right] + \mathcal{O}(\nu), \\ p_1 = \dfrac{F^4}{64}\Big[-5\cos(4\pi x) - 5\cos(4\pi y) + \cos(8\pi x) + \cos(8\pi y) + 12\cos(4\pi x)\cos(4\pi y) \\ \quad - \cos(8\pi x)\cos(4\pi y) - \cos(4\pi x)\cos(8\pi y)\Big] + \mathcal{O}(\nu). \end{cases} \tag{23}$$

Using this reference solution enables an accurate comparison between the results obtained from the weakly compressible solver and the analytical solution proposed in Eq. (21).

In order to evaluate the performance of the L-NIIPS method with ACS formulation, the impacts of the shifting magnitude limitation, and the threshold value $L_\infty(\nabla C)_{\text{thr}}$ are analyzed. The maximum shifting magnitude obtained by the L-NIIPS method is important as it is directly used in Eq. (20) to update the physical quantities with the ACS procedure. Moreover, the selection of an appropriate threshold remains an open question, as no general guideline has been established by existing implicit iterative shifting methods in the literature [33,34,38]. In all cases reported in Table 1, the percentage of mid-point pressure error at $(0.5L, 0.5L)$, together with the $L_2$-norm of pressure, $L_2(p)$, and the $L_2$-norm of the velocity in the $x$-direction, $L_2(u)$, are presented at the non-dimensional time $tU/L = 1.0$. The $L_2$-norms are computed as follows:

$$L_2(u) = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left\|\frac{u_i^{\text{SPH}} - u_i^{\text{analyt}}}{u_{\max}^{\text{analyt}}}\right\|^2}, \qquad L_2(p) = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\frac{p_i^{\text{SPH}} - p_i^{\text{analyt}}}{p_{\max}^{\text{analyt}}}\right)^2},$$

where $u_{\max}^{\text{analyt}}$ and $p_{\max}^{\text{analyt}}$ denote the maximum analytical values of the velocity and pressure, respectively, and $n$ is the total number of fluid particles. In all cases, the particle spacing is set to $\Delta/L = 0.01$, and the flow is assumed to be inviscid. To evaluate the influence of the shifting parameters, three different limitations on the shifting magnitudes, 2%, 5%, and 10% of $\Delta$ are considered. Regarding the threshold value, the analysis begins at 0.001 and increases by a factor of 4. This choice is motivated by the observation that the L-NIIPS typically provides a $L_\infty(\nabla C)$ 2–3 times bigger than the threshold due to its localization mechanism and the imposed limit on the shifting magnitude [33]. Among the listed cases in Table 1, Cases 1 to 4 exhibit the lowest errors, while Case 6 shows the highest deviation from the analytical pressure and velocity. Furthermore, Cases 1 and 4 perform similarly, indicating that increasing the threshold from 0.001 to 0.004 does not significantly affect the error in the physical quantities. This indicates that threshold values within the investigated range provide satisfactory numerical accuracy. In contrast, larger thresholds, such as those employed in Cases 5 and 6, lead to a noticeable degradation of the solution accuracy. Regarding the maximum shifting magnitude, smaller values are preferable, as they improve accuracy when coupled with the ACS formulation, which explicitly depends on the shifting magnitude (see Eq. (20)). Conversely, excessively large shifting magnitudes may induce spurious perturbations and consequently affect the accuracy of the computed physical quantities.

**Table 1**
Inviscid TGV. Comparison of different shifting setup based on errors in physical quantities at $tU/L = 1.0$ for $\Delta/L = 0.01$.

| Case | Threshold | Shifting Limit (%$\Delta$) | Pressure Error (%) | $L_2(p)$ | $L_2(u)$ |
|---|---|---|---|---|---|
| Case 1 | 0.004 | 2 | 2.7 | 0.0298 | 0.0037 |
| Case 2 | 0.004 | 5 | 5.2 | 0.0341 | 0.0039 |
| Case 3 | 0.004 | 10 | 6.6 | 0.0365 | 0.0038 |
| Case 4 | 0.001 | 2 | 3.1 | 0.0303 | 0.0033 |
| Case 5 | 0.016 | 2 | 10.2 | 0.0539 | 0.0052 |
| Case 6 | 0.064 | 2 | 15.1 | 0.1638 | 0.0107 |

Moreover, to confirm the low pressure error observed in Case 1, measured as only 2.7% at the midpoint, the pressure field for all cases at the non-dimensional time $tU/L = 1.0$ is shown in Fig. 4. As illustrated, Case 1 and 4 continue to yield the most accurate pressure field all over the domain, effectively capturing both high and low pressure regions.

In Fig. 5, a comparison is made between Cases 1–4 in terms of the computational cost, as they all provide acceptable pressure errors relative to the analytical solution. The CPU time for each case is normalized by the CPU time of the simulation without shifting (No-Shifting). This is intentional, as different explicit shifting procedures reported in the literature [26,28,30] are computed within the SPH particle–particle interaction loops, and their additional computational cost is either negligible or typically limited to about 5%–10% [33] compared to the No-Shifting case. As observed, Case 4 with 0.001 threshold significantly increases the computational cost up to 61%, as the L-NIIPS method requires a higher number of Newton–Raphson (NR) iterations per time step and also the localization procedure identifies (flags) a larger number of particles for shifting in the domain due to the strict value of threshold. Consequently, the resulting system of linear equations in Eq. (14) becomes larger. Among the tested configurations, Case 1, which applies a lower shifting magnitude, exhibits the lowest computational cost, with only a 42% overhead relative to the simulation without shifting. However, it is important to note that, as a general remark, the CPU time and overall computational cost can vary significantly depending on the structure of the source code (in our case, ASPHODEL); the implementation details of neighbor searching; loop structures; and the construction and solution of the linear system in Eq. (14).

Additionally, in Fig. 6, the value of $L_\infty(\nabla C)$ is shown for Cases 1–4, demonstrating that the L-NIIPS method is capable of maintaining a well-distributed particle arrangement. Moreover, it is observed that using lower shifting magnitude (like in Case 1) does not significantly affect particle uniformity or increase the value of $L_\infty(\nabla C)$.

Furthermore, the pressure error can be defined as:

$$\epsilon_p = \frac{\left\| p^{\text{SPH}} - p^{\text{analyt}} \right\|}{p^{\text{analyt}}}, \tag{24}$$

and is evaluated in Fig. 7 at the midpoint of the domain considering four spatial resolutions ($\Delta/L = 0.02, 0.01, 0.005, 0.0025$) at the non-dimensional time $tU/L = 1.0$, for Case 1 and Case 4. As shown, the observed order of convergence for both cases is close to 2, which is the theoretical order of convergence of the SPH operators at continuous level. It should be noted that at coarse resolutions, both Case 1 and Case 4 yield comparable performance. However, having a lower threshold, as in Case 4, results in a slight improvement in error values at higher resolutions compared to Case 1 (with a threshold set as 0.004). This negligible improvement, although, comes at a significant overhead in computational cost (see Fig. 5).

As a general remark, the calibrated pair, including a threshold of 0.004 and a maximum shifting magnitude equal to 2% of $\Delta$, is adopted for all subsequent simulations. Although these values yield satisfactory results, they could be slightly adjusted according to the desired balance between accuracy and computational efficiency. Nevertheless, it is strongly recommended to remain within the investigated range of Cases 1–5 in Table 1.

The TGV test case has also been simulated with Reynolds number Re = 100, specifically to assess the effectiveness of L-NIIPS with ACS in improving the accuracy of the viscous operator. Fig. 8 shows the obtained velocity magnitude field (Fig. 8(a)) and pressure field (Fig. 8(b)) obtained from the simulation, which demonstrates good agreement with the analytical solution.

Moreover, to enable a direct comparison with existing studies in the literature, Fig. 9 presents the volume error $E_v$ for the viscous case at a spatial resolution of $\Delta/L = 0.0025$, computed as:

$$E_v(\%) = \left| \frac{\sum_{i=1}^{n} \omega_i}{L^2} - 1.0 \right| \times 100. \tag{25}$$

Although the ALE formulation in Eq. (1) does not inherently conserve volume, the combination of L-NIIPS and ACS achieves the lowest volume error in the domain compared to previously published results [26,30,70,71].

### 7.2. 2D moving square box

In this section, a moving square box test case is considered. The test involves a square box of side length $L = 1$ m placed inside a rectangular domain (10 m × 5 m). The box starts from rest and accelerates for 1 s to reach a constant velocity $U = 1$ m/s, after which it continues to move at this velocity. This test case is particularly important as it allows investigation of the L-NIIPS and ACS formulation in the context of confined domains, specifically, the interaction between the moving box and the surrounding wall boundaries. The simulation is performed using a spatial resolution of $\Delta/L = 0.02$, Reynolds number Re = 100, and reference density $\rho = 1$ kg/m$^3$. For the L-NIIPS parameters, the threshold is set to 0.004 and the maximum shifting magnitude is limited to 2% of $\Delta$.

To assess the performance of the L-NIIPS setup, specifically, the effects of the threshold and maximum shifting magnitude, in the presence of wall boundaries, Fig. 10(a) shows the value of $L_\infty(\nabla C)$ throughout the simulation and Fig. 10(b) focuses on the box at the non-dimensional time $tU/L = 5.0$. Owing to the presence of sharp edges, the figure demonstrates that the L-NIIPS method provides a well-distributed particle arrangement around the moving box. The value of $h \cdot |\nabla C|$ for all particles near the box remain within the predefined threshold range, with no gaps or clustering near the walls.

Fig. 11 presents the reference results provided by SPHERIC (SPH rEsearch and engineeRing International Community) [68] for the non-dimensional velocity and pressure fields at the non-dimensional time $tU/L = 5.0$ with an incompressible assumption. These reference results serve as a reliable benchmark for evaluating the performance of the proposed methods introduced in this study.

Consequently, Fig. 12 compares the velocity magnitude and pressure field obtained with different formulations. The first row presents the IIPS with ACS results obtained by Rastelli et al. [37], which exhibit significant pressure noise. This behavior is mainly attributed to the instability of the IIPS formulation in particle distribution [33], the large shifting magnitudes, and the adoption of ACS formulation excluding the wall contribution terms in Eq. (20). The second row shows the simulation of L-NIIPS with ACS, again without considering the wall contributions terms in Eq. (20). While this treatment ensures a smooth velocity field, it produces noticeable pressure oscillations near the walls and leads to a loss of symmetry in the solution. Finally, the third row (L-NIIPS + ACS) by adopting the complete formulation of ACS demonstrates the most accurate results compared to the benchmark data from the SPHERIC validation tests (Fig. 11). To the best of the authors' knowledge, this solution provides the closest match to the

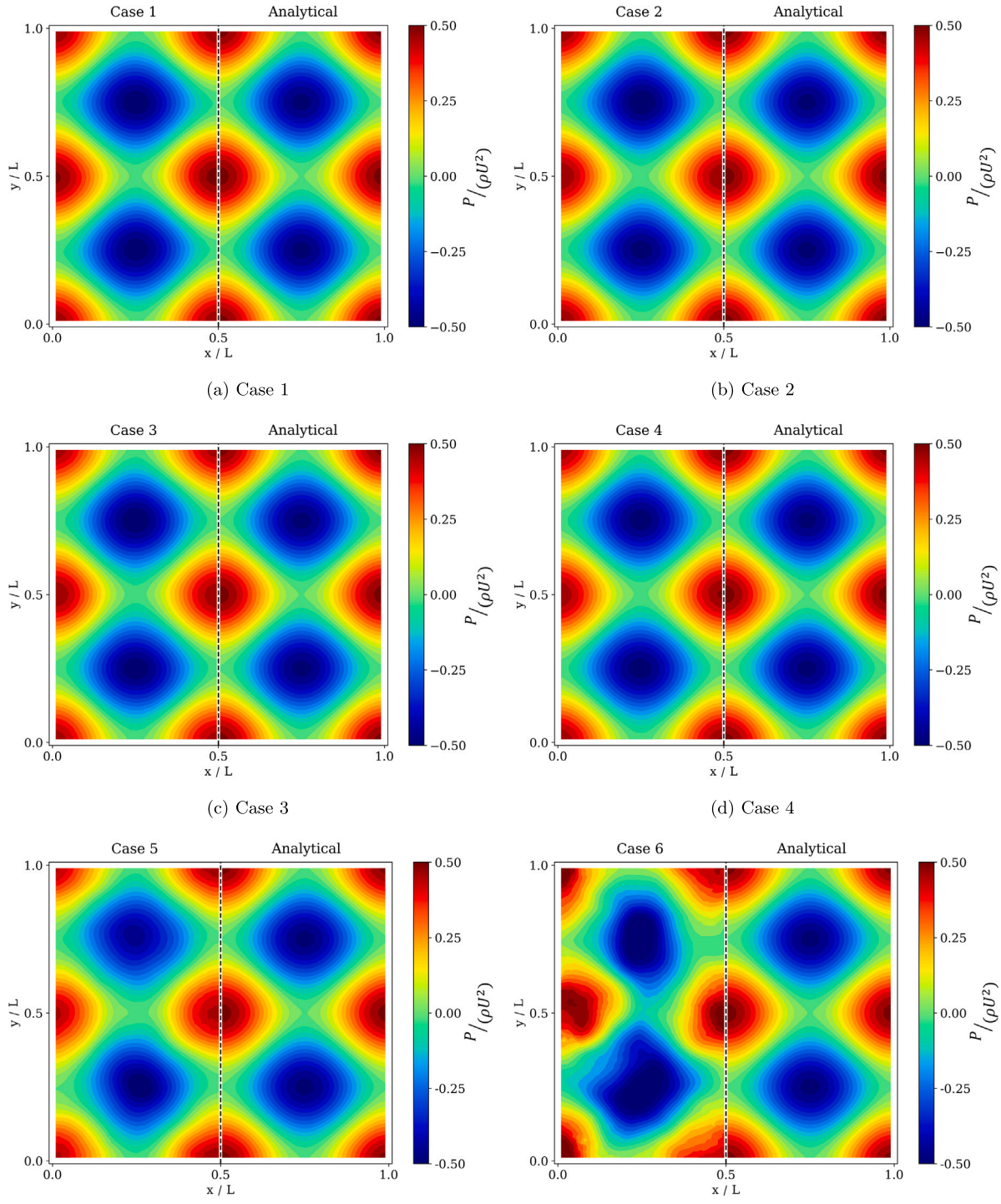



**Fig. 4.** Inviscid TGV. Snapshots of the pressure field at the non-dimensional time $tU/L = 1.0$ for $\Delta/L = 0.01$.

SPHERIC results [68] in terms of both velocity and pressure fields, with a nearly symmetric pressure distribution.

Furthermore, after confirming the smooth and noise-free pressure field achieved by the L-NIIPS scheme with ACS (Fig. 12(f)), we evaluate the mass and volume conservation properties within the domain, since the introduction of wall contribution terms in Eq. (20) through the boundary integral approach may not be strictly conservative. Since this is a confined test case involving continuous particle shifting around the moving box, it provides an appropriate framework to monitor the impact of the wall contribution terms in Eq. (20) on mass and volume conservation. The mass error is computed using an expression analogous to Eq. (25) as follows:

$$E_m(\%) = \left| \frac{\sum_{i=1}^{n} m_i}{m_0} - 1.0 \right| \times 100, \tag{26}$$

where $m_i$ is the mass of particle $i$, and $m_0$ is the total initial mass of fluid particles. As illustrated in Fig. 13(a), when implementing the complete formulation of ACS, the total mass increases slightly over the

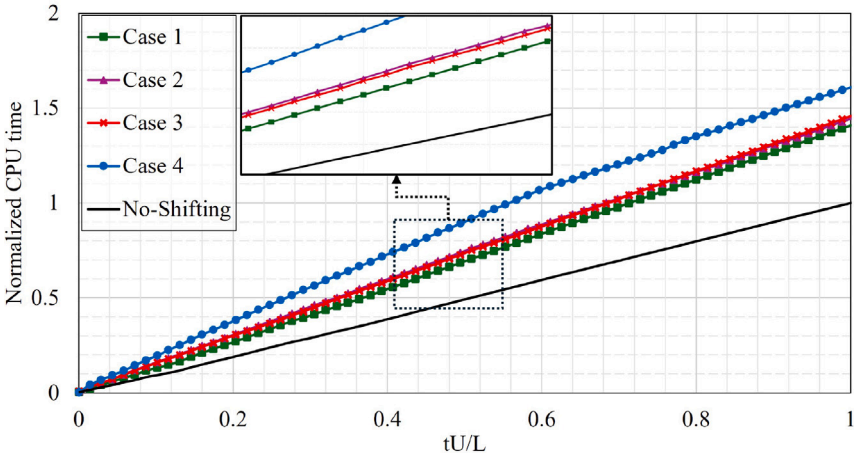



**Fig. 5.** Inviscid TGV. Relative CPU time comparison for Cases 1–4.

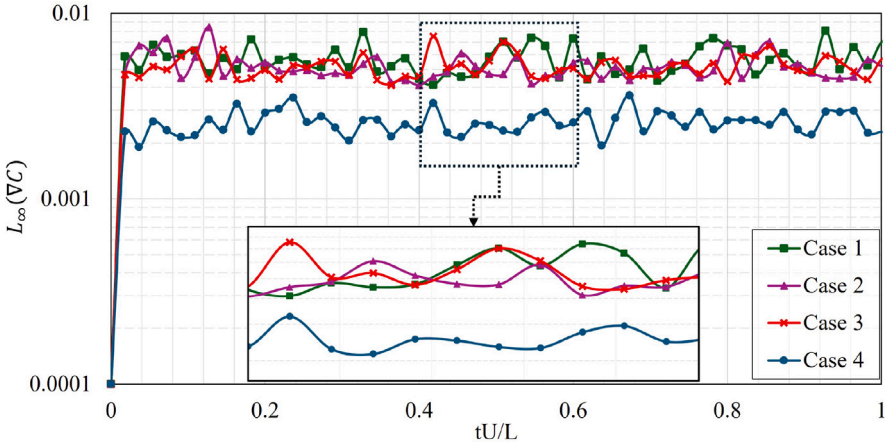



**Fig. 6.** Inviscid TGV. $L_\infty(\nabla C)$ comparison for Cases 1–4.

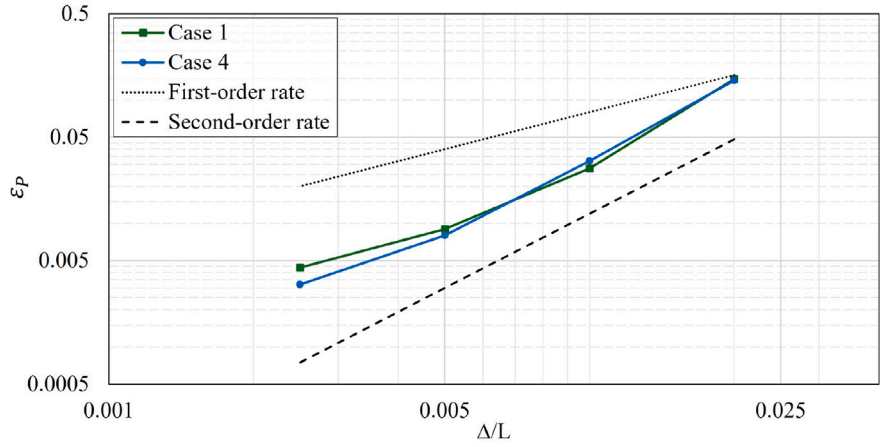
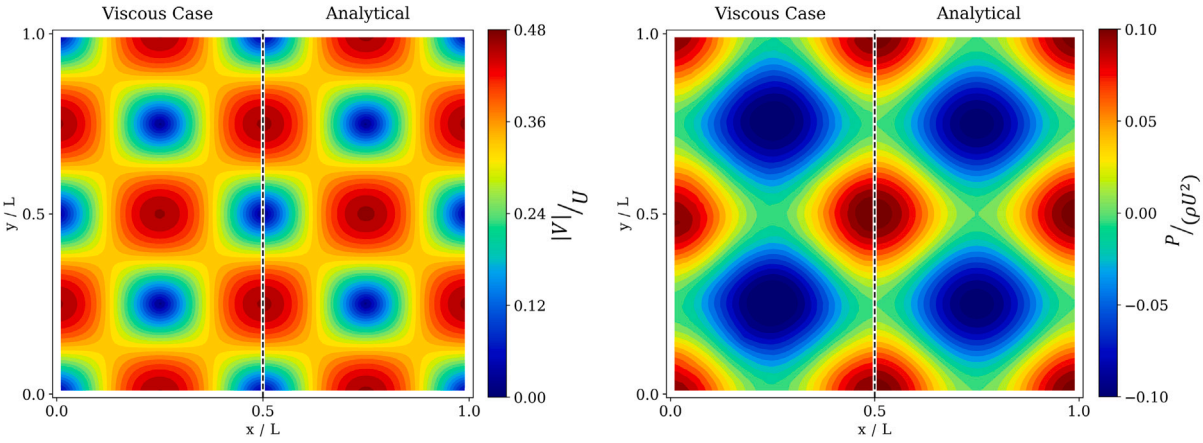



**Fig. 7.** Inviscid TGV. Convergence analysis of the mid-point pressure error at the non-dimensional time $tU/L = 1.0$ for Case 1 and 4.

course of the simulation. As discussed in relation to Eq. (20), the wall contribution terms introduced in the ACS are essential for producing a smooth pressure field. However, they lead to a minor increase in the total mass, which reaches approximately 0.0062% by the end of the simulation at the non-dimensional time $tU/L = 5.0$. This deviation remains negligible and does not introduce any effect on the physical behavior or numerical accuracy of the scheme.

In contrast to mass, volume is not inherently conserved in the ALE-SPH formulation and is not directly affected by the shifting procedure. Nevertheless, as shown in Fig. 13(b), the shifting process helps maintain low volume errors, as also demonstrated in the TGV test case (see Fig. 9). For the moving box case, the volume error reaches a maximum of 0.0022% during the simulation and reduces to 0.0004% by the final time step at the non-dimensional time $tU/L = 5.0$.




(a) Velocity field (b) Pressure field

**Fig. 8.** Viscous TGV. Snapshot of the velocity field (left) and pressure field (right) at the non-dimensional time $tU/L = 1.0$ for $Re = 100$, $\Delta/L = 0.01$ and $h/\Delta = 2.0$.

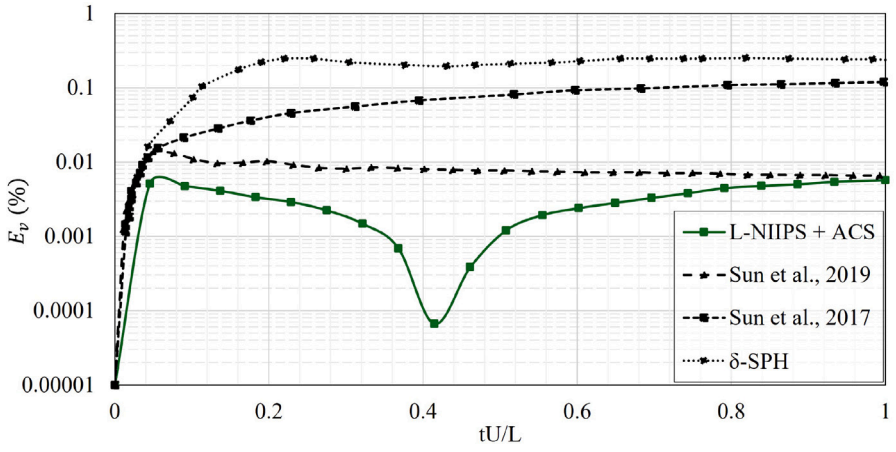
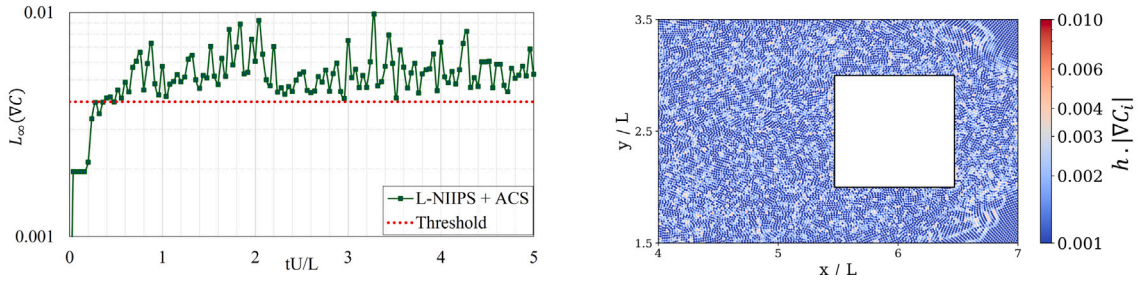



**Fig. 9.** Viscous TGV. Volume error comparison of the proposed method with respect to the results obtained by Sun et al. [26,30] and $\delta$-SPH [70,71] for $Re = 100$, $\Delta/L = 0.0025$ and $h/\Delta = 2.0$.



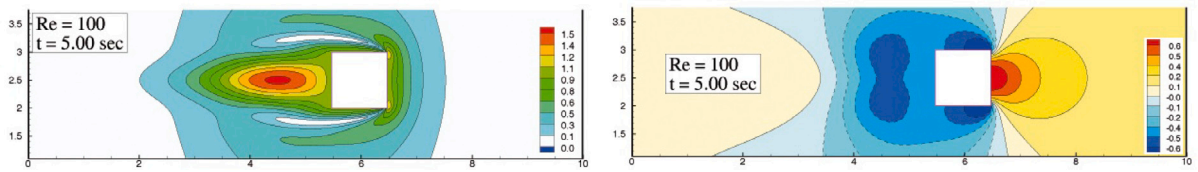

(a) $L_\infty(\nabla C)$ (b) $h \cdot |\nabla C_i|$ at $tU/L = 5.0$

**Fig. 10.** Moving Square Box. Particle distribution evaluation.

Finally, in terms of computational cost, Fig. 14 shows that the simulation incorporating the shifting method is approximately 29% more expensive compared to the simulation performed without any shifting (No-Shifting).

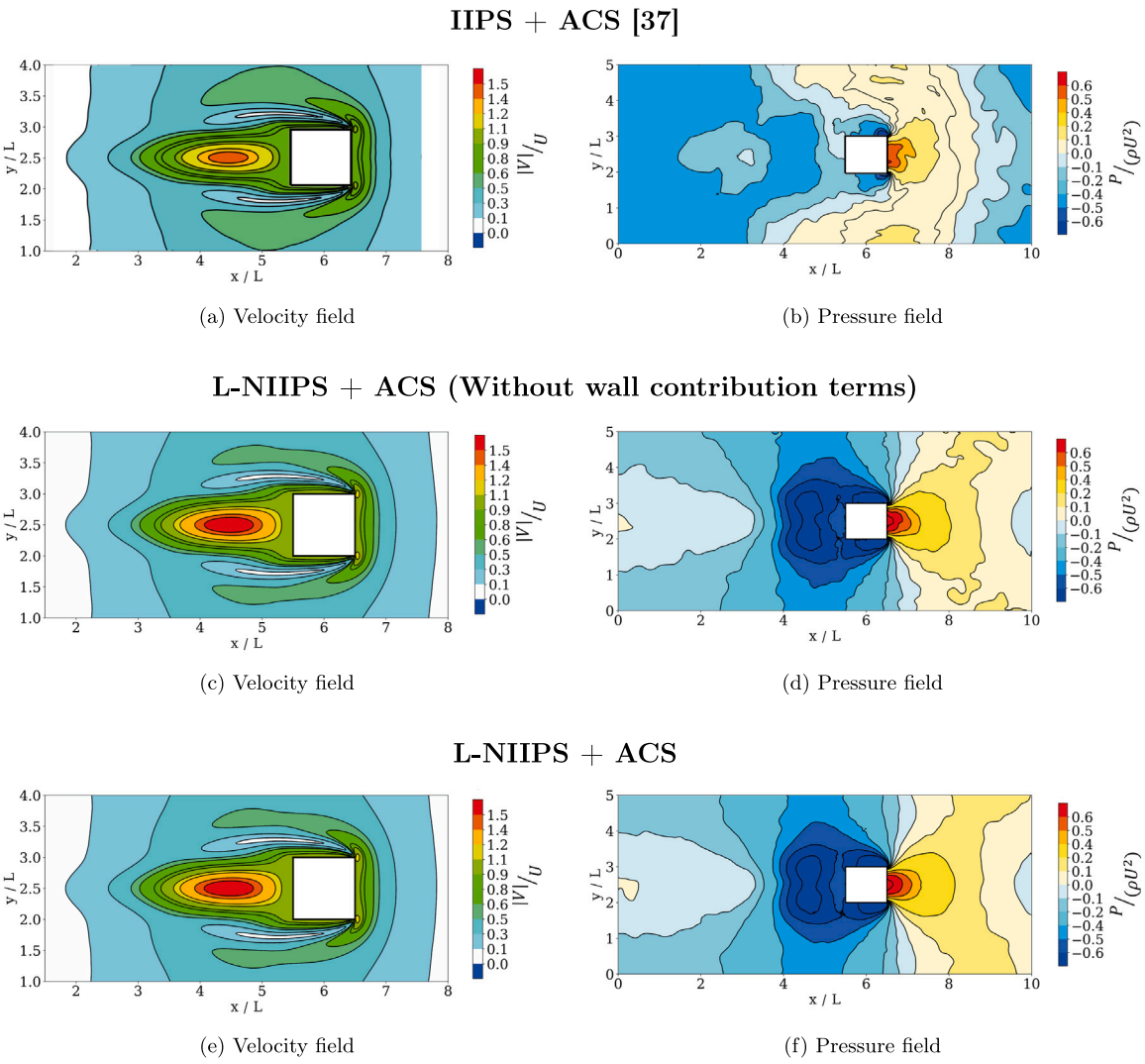
### 7.3. 2D impinging jet

The application of a jet perpendicularly impinging on a flat plate is investigated in this section. This case is associated with an implicit analytical solution derived by Taylor [72] for the pressure and velocity distribution at the flat plate. This free-surface problem serves as a foundational test case for simulating flows interacting with rigid structures, such as those found in various types of hydraulic turbines. Fig. 15 illustrates the domain configuration and geometric parameters. The colormap represents the pressure coefficient obtained by the L-NIIPS and ACS formulation with a maximum shifting magnitude limited to 2%




(a) Non-dimensional velocity field $V/U$ (b) Non-dimensional pressure field $P/(\rho U^2)$

**Fig. 11.** Moving Square Box. Snapshots of the velocity (left) and pressure (right) fields using a Navier–Stokes finite difference solver provided by SPHERIC [68] at the non-dimensional time $tU/L = 5.0$ for incompressible flow.




(a) Velocity field (b) Pressure field

(c) Velocity field (d) Pressure field

(e) Velocity field (f) Pressure field

**Fig. 12.** Moving Square Box. Snapshots of the velocity (left column) and pressure (right column) fields at the non-dimensional time $tU/L = 5.0$ using different approaches.

of $\Delta$, and a threshold set to 0.004 for a spatial resolution of $H/\Delta = 40$ at the converged state in time ($tU/H = 75.0$). For the cases involving a jet, the pressure coefficient is defined as:

$$C_p = \frac{p}{0.5\rho U^2}, \tag{27}$$

where $p$ denotes the pressure, $\rho$ is the reference density, and $U$ is the jet velocity. In this study, $\rho = 1000$ kg/m$^3$, $U = 100$ m/s, and the jet diameter is $H = 0.04$ m.

Furthermore, Fig. 16 illustrates the particle distribution and the value of $h\cdot|\nabla C|$ for each particle shortly after impact (Fig. 16(a)) at the non-dimensional time $tU/H = 4.0$ and at converged state in time (Fig. 16(b)) at $tU/H = 75.0$ for a resolution of $H/\Delta = 40$. The results demonstrate the method's strong capability in maintaining particle uniformity with most particles adhering closely to the predefined threshold.

Additionally, Fig. 17 presents the pressure distribution at the non-dimensional time $tU/H = 75.0$ for $H/\Delta = 40$, where the ACS formulation is applied according to the L-NIIPS procedure, excluding the wall contribution terms in Eq. (20). The resulting pressure field clearly demonstrates the significant influence of these wall terms in comparison to Fig. 15 where the complete ACS formulation is used. Their omission leads to noticeable pressure noise, highlighting the

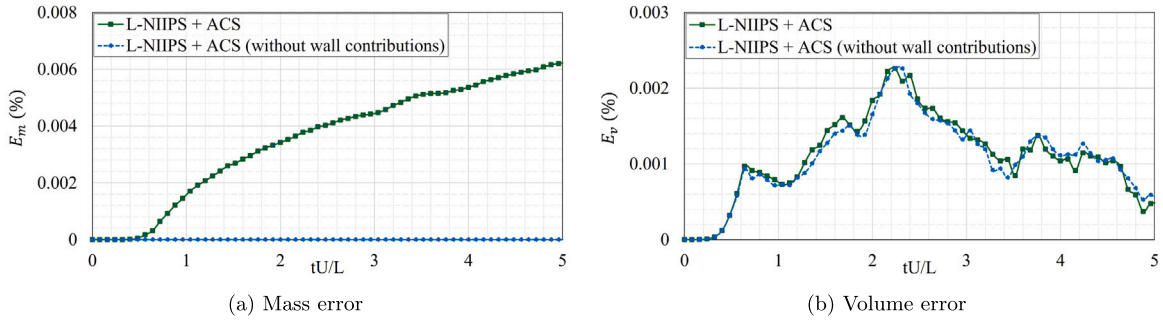



**Fig. 13.** Moving Square Box. Mass (left) and volume (right) error monitoring.

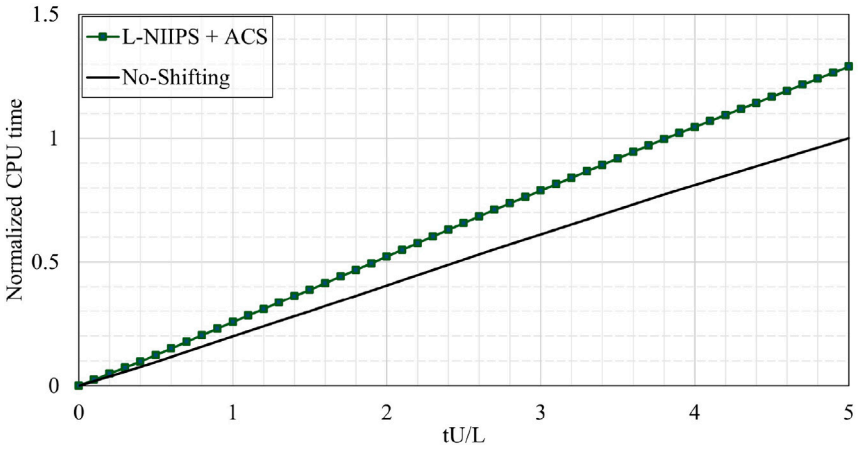



**Fig. 14.** Moving Square Box. Relative CPU time comparison.

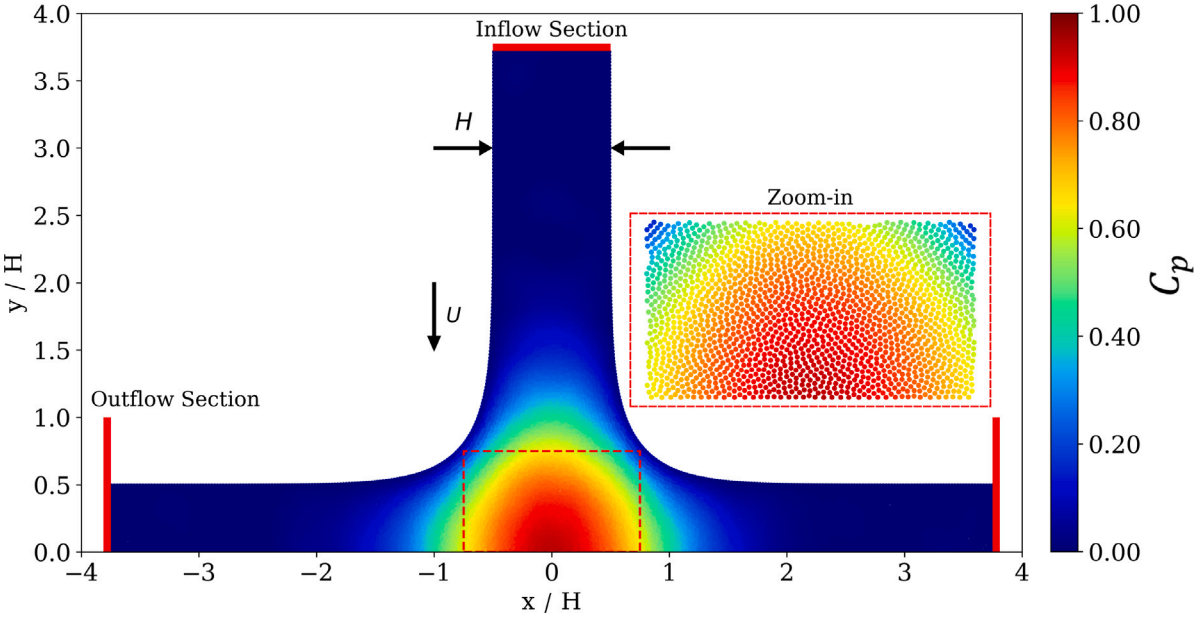
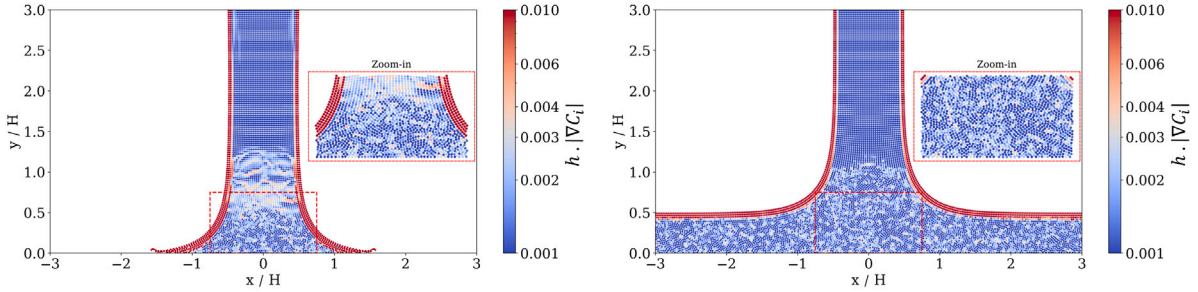



**Fig. 15.** 2D Impinging Jet. Setup and geometric parameters obtained by L-NIIPS + ACS at the converged state in time $tU/H = 75.0$ for $H/\Delta = 40$.

necessity of including wall contribution terms in the ACS for obtaining physically consistent pressure distributions.

Moreover, at the stagnation point $(x/H = 0,\ y/H = 0)$, the measured pressure is shown in Fig. 18 for three different resolutions: $H/\Delta = 20, 40, 80$, over the non-dimensional time interval $tU/H \in [25.0, 100.0]$ using L-NIIPS with ACS. The results demonstrate that the pressure signal progressively converges toward the analytical solution as the resolution increases. Notably, for the finest resolution $(H/\Delta = 80)$,




(a) at $tU/H = 4.0$ (b) at $tU/H = 75.0$

**Fig. 16.** 2D Impinging Jet. The value of $h \cdot |\nabla C|$ for each particle after impact (left) and at the converged state in time (right) for $H/\Delta = 40$.

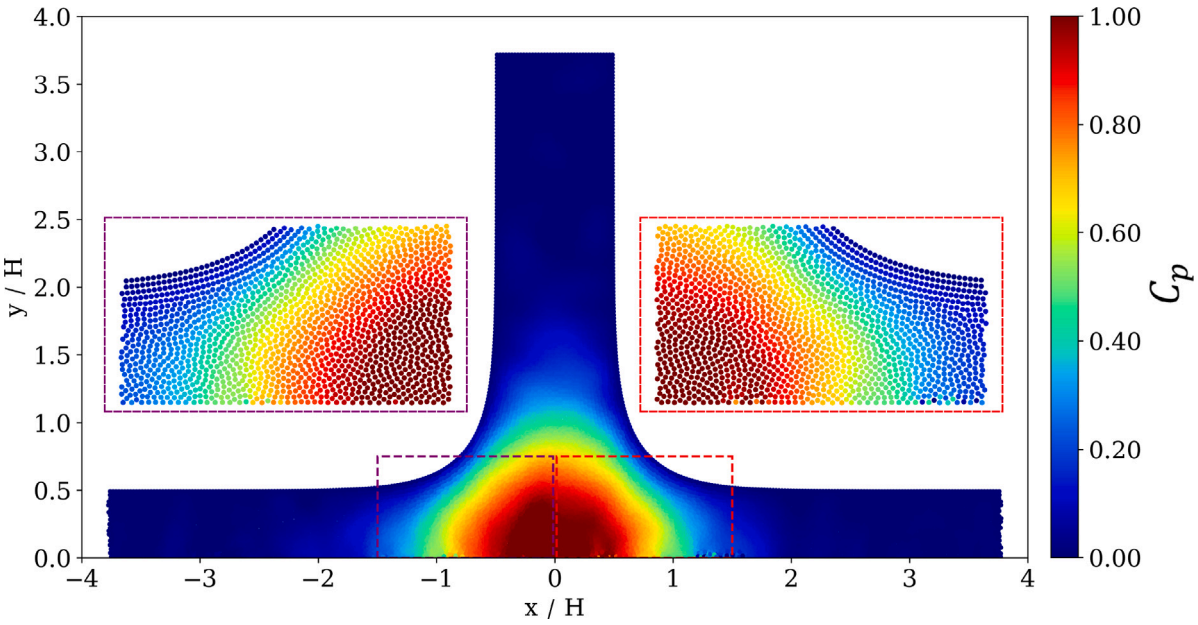



**Fig. 17.** 2D Impinging Jet. Pressure field obtained by L-NIIPS + ACS (Without wall contribution terms) at the converged state in time $tU/H = 75.0$ for $H/\Delta = 40$.

convergence occurs earlier than in the coarser cases, reducing spurious pressure oscillations even from the non-dimensional time $tU/H = 25.0$. In particular, the simulation with $H/\Delta = 80$ demonstrates close agreement with the analytical solution, yielding an error of 3.1% at $tU/H = 100.0$, with minimal oscillations and improved accuracy compared to the results reported in [38], however, the remaining oscillations originate from the acoustic waves arising from the weakly compressible nature of the solver.

Finally, Fig. 19 presents the pressure distribution along the flat plate ($x/H \in [-2.0, 2.0]$). The numerical results, obtained by time-averaging the pressure field over the non-dimensional interval $tU/H \in [25.0, 100.0]$, show excellent agreement with the analytical solution [72, 73], particularly at the highest resolution ($H/\Delta = 80$). Remarkably, even at the coarsest resolution ($H/\Delta = 20$), the results retain a good level of agreement with the analytical profile, highlighting the robustness of the proposed numerical approach. Furthermore, the root mean square error (RMSE), evaluated in terms of the pressure coefficient $C_p$ at the pressure probes shown in the figure with respect to the analytical solution, is defined as

$$\text{RMSE} = \sqrt{\frac{1}{M}\sum_{i=1}^{M}\left(C_{p,i}^{\text{num}} - C_{p,i}^{\text{analyt}}\right)^2},$$

where $M = 13$ represents the total number of pressure probes. The corresponding RMSE values are 0.0368, 0.0197, and 0.0116 for the resolutions $H/\Delta = 20$, 40, and 80, respectively. These results further confirm the progressive convergence of the solution and the high accuracy achieved while refining the resolution.

## 8. 3D results

To evaluate the effectiveness of the proposed methods in three dimensions, we consider two benchmark problems: a 3D impinging jet and a 3D static Pelton bucket. In both cases, particle shifting is performed using the L-NIIPS method, while physical quantities are updated with the complete ACS formulation. The 3D impinging jet serves as a validation of accuracy, after which the focus shifts to the Pelton bucket to further assess the method's performance in a more complex geometry. All shifting parameters are kept consistent with the 2D test cases, namely a threshold of 0.004, and a maximum shifting magnitude of $2\%\Delta$.

### 8.1. 3D impinging jet

In this section, a 3D impinging jet on a flat plate is simulated. The jet, with a velocity $U = 20$ m/s directed along the $z$-axis, impacts a horizontal plate located in the $x$–$y$ plane. The jet diameter is set to $H = 0.03$ m, the spatial resolution is chosen as $H/\Delta = 40$ and the reference density $\rho$ is 1000 kg/m$^3$. Fig. 20 demonstrates the effectiveness of the proposed 3D formulation of L-NIIPS with ACS in maintaining particle uniformity close to the threshold (Fig. 20(a)) and producing a smooth

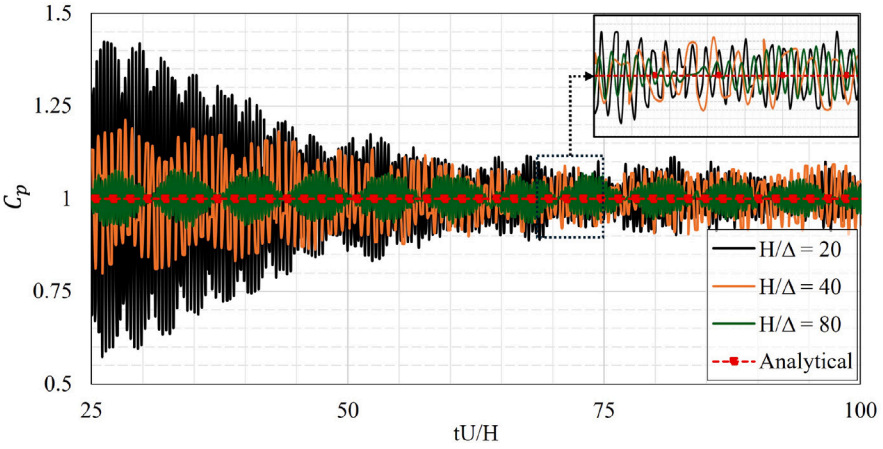



**Fig. 18.** 2D Impinging Jet. Comparison of the pressure signal at the stagnation point for three different resolutions $H/\Delta = 20, 40, 80$.

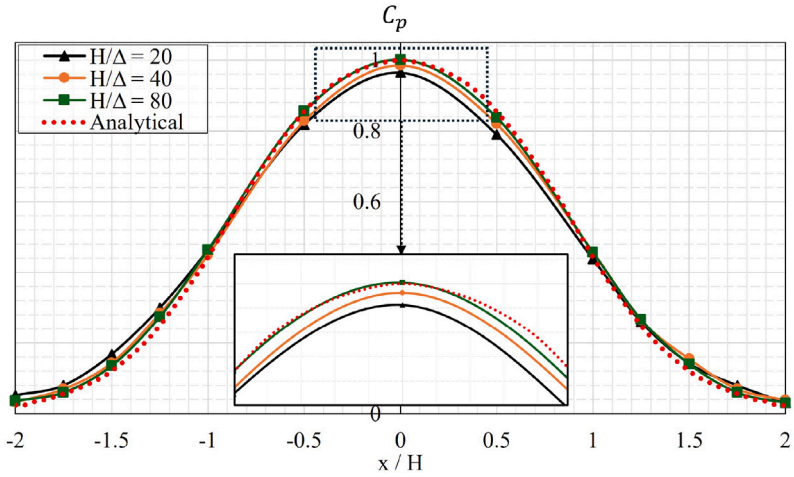
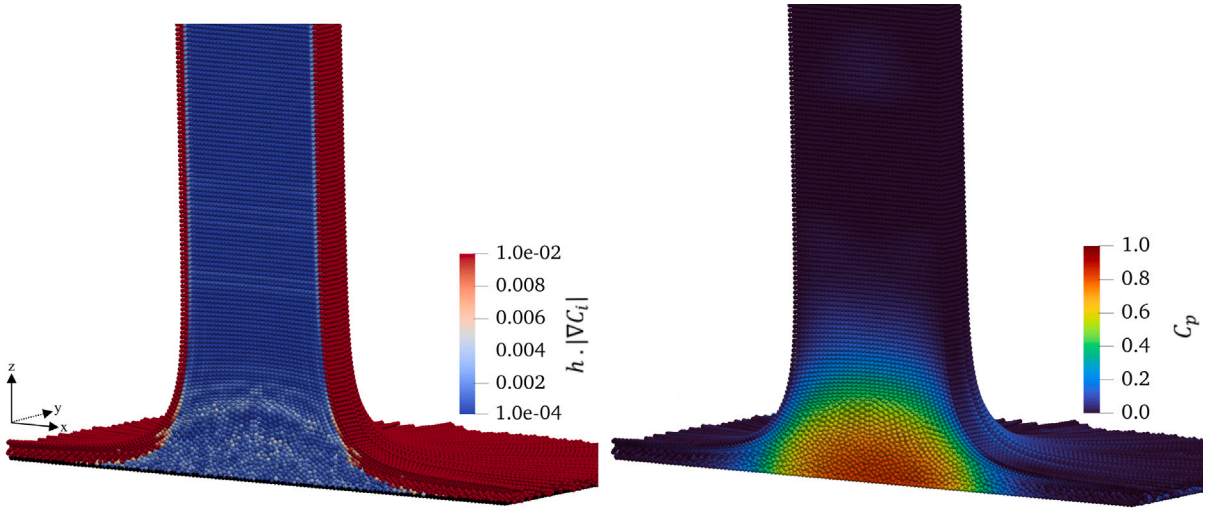



**Fig. 19.** 2D Impinging Jet. Pressure distribution along the flat plate for three different resolutions $H/\Delta = 20, 40, 80$ over the time interval $tU/H \in [25.0, 100.0]$.




(a) Particle distribution (b) Pressure field

**Fig. 20.** 3D Impinging Jet. The value of $h \cdot |\nabla C|$ for each particle (left) and pressure field (right) at the converged state in time $tU/H = 50.0$ for $H/\Delta = 40$.

and stable pressure field (Fig. 20(b)) at a converged state in time $tU/H = 50.0$.

Moreover, Fig. 21 shows the pressure distribution along the flat plate at the given resolution, obtained using the L-NIIPS with ACS approach. A direct comparison is provided with the experimental results of Kvicinsky [51,52] and with the SPH simulations of Michel et al. [28], who employed a multi-invariant shifting technique, at the non-dimensional time $tU/H = 50.0$. This comparison is carried

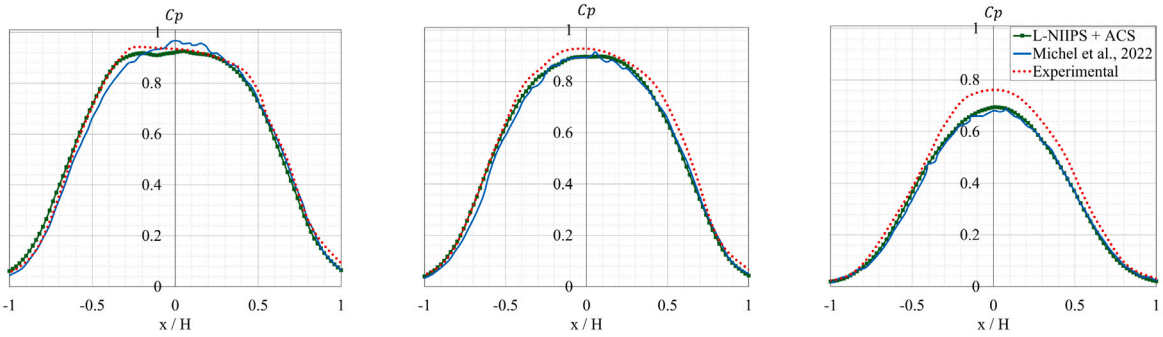



**Fig. 21.** 3D Impinging Jet. Comparison of the pressure distribution along the flat plate for three different cross-sections at the non-dimensional time $tU/H = 50.0$ with L-NIIPS + ACS (green squared line), experimental data [51] (red dots), and Michel et al. [28] results (blue line).

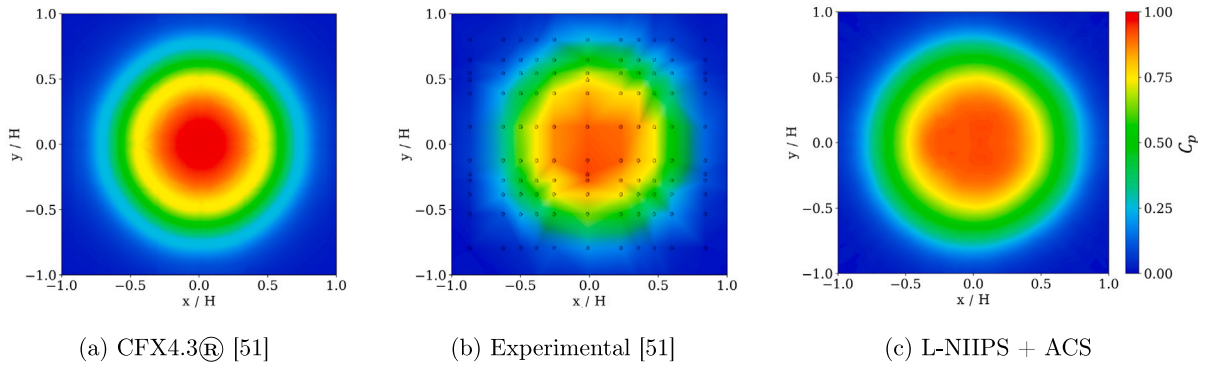



**Fig. 22.** 3D Impinging Jet. Comparison of the pressure field at the converged state in time $tU/H = 50.0$ with experimental and CFX4.3® results [51].

out at three different cross-sections along the flat plate: $y/H = 0$, $y/H = 0.27$, and $y/H = 0.53$ (from left to right) with 100 equally spaced pressure probes at each section. The results demonstrate good agreement with the experimental data, particularly in capturing the peak pressure region, and show reduced oscillations compared to the simulation results reported in [28].

Finally, Fig. 22 presents the pressure field on the flat plate at the non-dimensional time $tU/H = 50.0$, compared with the experimental and CFX4.3® results reported by Kvicinsky [51]. The comparison confirms the smoothness of the predicted pressure distribution and shows excellent agreement in both high and low pressure regions with the experimental data.

### 8.2. 3D static pelton bucket

To further assess the proposed method, we simulate a static Pelton bucket impacted by a water jet of velocity $U = 19.61$ m/s directed along the $z$-axis. The jet diameter is $H = 0.03$ m, with a spatial resolution of $H/\Delta = 60$, and the reference density is set to $\rho = 1000$ kg/m$^3$. The bucket surface, provided by Andritz Hydro company, is discretized with 106,021 wall particles at the same resolution. The simulation proceeds until a converged state is achieved, after approximately 2 ms of physical time, yielding around 515,320 fluid particles. Although Pelton turbine casings are typically asymmetric, the casing is excluded in this study to isolate and analyze the flow behavior near the bucket. To reduce the computational cost, a symmetry plane is applied to halve the domain, with appropriate boundary conditions enforced using fictitious fluxes, following the ghost particle approach introduced by Colagrossi and Landrini [74].

Fig. 23 illustrates the particle-based representation of the bucket at a spatial resolution of $H/\Delta = 60$, where the non-dimensional mass of the solid-wall particles ($\omega'/\Delta^2$) is shown via the colormap. As evident from the figure, the particle distribution is locally refined in regions with sharp edges to better capture the bucket curvatures. Moreover, to enable a detailed comparison of pressure distribution on the bucket surface, three vertical and three horizontal lines equipped with pressure probes are placed, following the experimental configuration proposed by Kvicinsky [51].

For comparison, an additional simulation is carried out within a purely Lagrangian framework, without employing any particle shifting procedure (No-Shifting case). Apart from this, all aspects of the simulation setup are kept identical to those in the shifting case. Fig. 24 displays the value of $h \cdot |\nabla C|$ for each fluid particle in a clipped cross-section at $x/H = -1.32$, comparing the No-shifting case (left column) and the L-NIIPS with ACS (right column) at three key stages: immediately after jet impact (Fig. 24(a)), after the jet exits the bucket (Fig. 24(b)), and at the converged state in time (Fig. 24(c)).

This comparison highlights the effectiveness of the 3D formulation of the L-NIIPS in maintaining a uniform particle distribution throughout the simulation. The values of $h \cdot |\nabla C|$ remain close to the defined threshold, indicating well-controlled particle regularity, and significantly outperforming the No-Shifting case. It is worth noting that in Fig. 24(c), some splashed particles appear to rejoin the water sheet near the trailing edge of the bucket. These particles originate from geometric

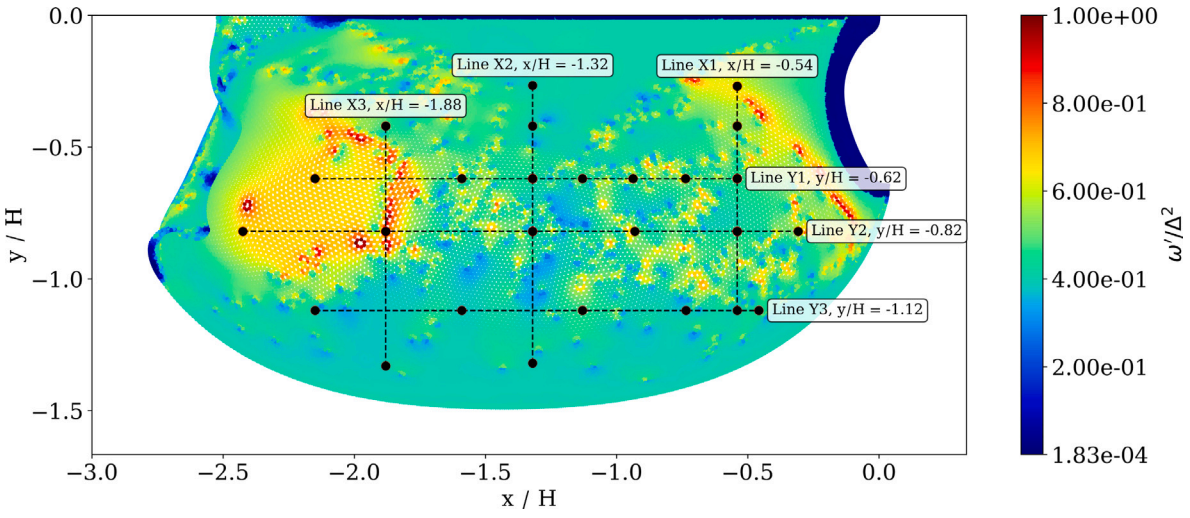



**Fig. 23.** 3D Static Pelton Bucket. Non-dimensional mass of the solid-wall particles and location of the pressure probes for $H/\Delta = 60$.

features and do not affect the pressure field or overall flow behavior, as will be further confirmed in the following.

Furthermore, Fig. 25 compares the performance of the proposed method for L-NIIPS with ACS at different resolutions ($H/\Delta = 60, 30$), as well as for the No-Shifting case, all evaluated against experimental data [51,52] obtained from pressure probes described in Fig. 23 at the non-dimensional time $tU/H = 13.05$. For the numerical simulations, 25 equally spaced pressure probes are inserted between each pair of experimental probes (allowing for a more detailed assessment of the smoothness and average behavior of the pressure distribution). As illustrated in Figs. 25(a), 25(c), 25(d), and 25(e), the L-NIIPS method with ACS shows good agreement with the experimental measurements. In contrast, Figs. 25(b) and 25(f) reveal a slight overestimation of the pressure, although the overall trend remains consistent with the experimental results. It is worth noting that, according to the experimental study reported in [51], one of the pressure sensors was identified as malfunctioning, as highlighted in Fig. 25(d). Moreover, increasing the resolution from $H/\Delta = 30$ to $H/\Delta = 60$ improves the computed pressure coefficient and eliminates oscillations. In contrast, the No-Shifting case exhibits significant oscillations in the pressure coefficient, while the L-NIIPS with ACS ($H/\Delta = 60$) maintains a smooth and consistent pressure coefficient profile across all defined probe lines.

Consequently, to validate the previously discussed results, Fig. 26 depicts the pressure field at the non-dimensional time $tU/H = 13.05$. The first row shows the pressure field within the fluid domain, while the second row presents the distribution on the bucket. Results are reported for the No-Shifting case (left column) and the L-NIIPS with ACS approach (right column), both evaluated at a resolution of $H/\Delta = 60$. The L-NIIPS with ACS approach clearly yields a smoother pressure field, with well-distributed gradients and reduced numerical oscillations, especially at the bucket surface. In contrast, the No-Shifting case exhibits visible artifacts and irregularities, particularly near the central region of the bucket. These results reinforce the performance of L-NIIPS with ACS in capturing the pressure distribution.

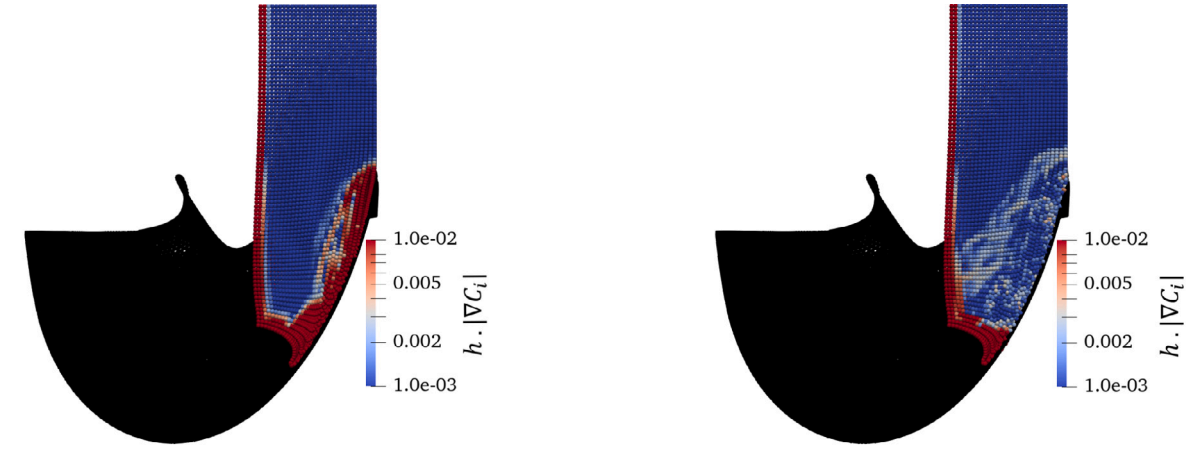
Finally, Fig. 27 presents the computational cost for the resolution $H/\Delta = 60$, showing that the simulation using L-NIIPS with ACS is 52% more expensive compared to the No-Shifting case. It should be noted that the flat line in the plot at the beginning of the simulation is due to the use of an accelerated phase of the simulation before the fluid impacts the bucket.

## 9. Conclusion and future works

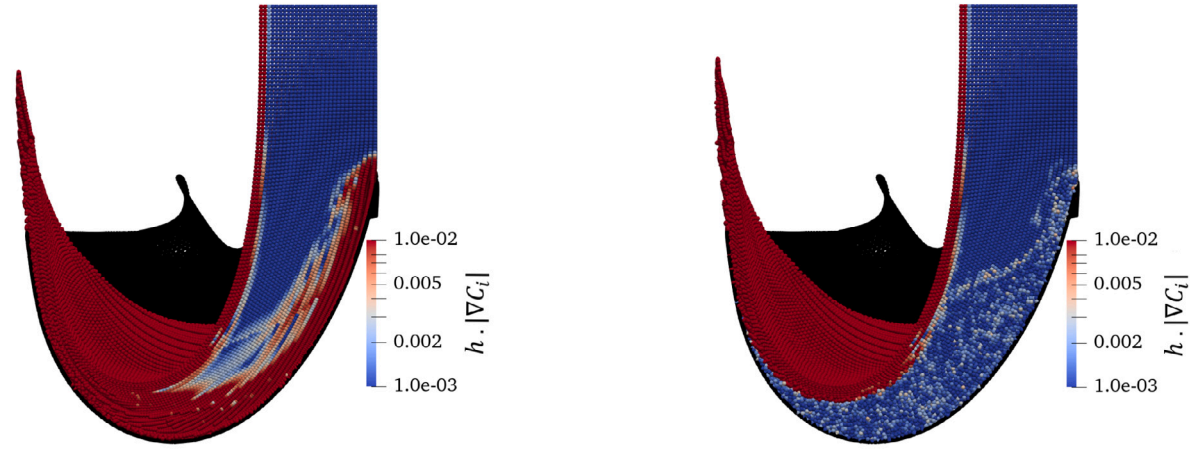
This work introduced the three-dimensional formulation of the Localized Novel Implicit Iterative Particle Shifting (L-NIIPS) algorithm as a robust particle regularization technique within the Arbitrary Lagrangian–Eulerian Smoothed Particle Hydrodynamics (ALE-SPH) framework. The method effectively addresses key limitations of both explicit and existing implicit shifting schemes by solving a localized system of linear equations to maintain a uniform particle distribution throughout the domain. For the first time, an implicit iterative shifting technique is applied in three dimensions, and combined with a proposed multi-node parallelization strategy enabling efficient large-scale 3D simulations. The Advection Correction Step (ACS) incorporating wall contribution with boundary integral method was also introduced yielding significant improvements in pressure field smoothness. Moreover, a numerical investigation of the L-NIIPS method coupled with ACS was conducted to determine suitable shifting parameters, including the threshold value and the maximum shifting magnitude, in order to balance computational cost, numerical accuracy, and stability. The results indicate that low threshold values (e.g., 0.001) decrease computational efficiency, whereas higher values (above 0.01) may induce pressure oscillations. Regarding the maximum shifting magnitude, although relatively large values (e.g., 10% of $\Delta$) can still yield acceptable results, smaller values are generally preferred to ensure higher accuracy in the prediction of physical quantities.

The combined L-NIIPS and Advection Correction Step (ACS) approach was validated against a range of canonical benchmarks. In 2D flows, including the Taylor–Green Vortex, moving square box, and impinging jet the method produced minimal numerical error and noise-free pressure fields. In 3D, the impinging jet further confirmed the robustness of the methodology, demonstrating the capability of the 3D L-NIIPS formulation with ACS in providing a uniform particle distribution and accurate pressure distribution. Finally, the 3D static Pelton Bucket simulation showcased the method's industrial applicability, successfully handling complex geometry and free-surface flow while incurring only moderate computational overhead.

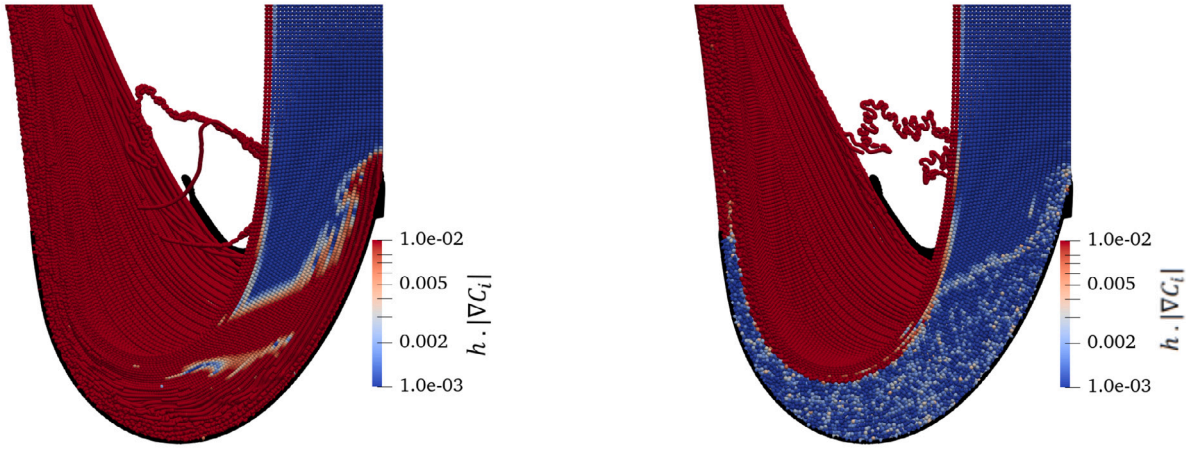
Future work may focus on assessing the performance of L-NIIPS with different kernel functions, in multiphase flow applications and variable resolution simulations. In addition, extending the linear solver to




(a) at $tU/H = 1.9$

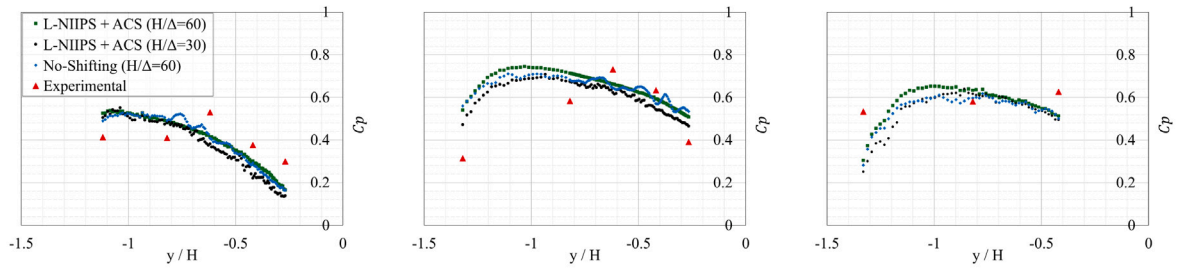



(b) at $tU/H = 3.56$

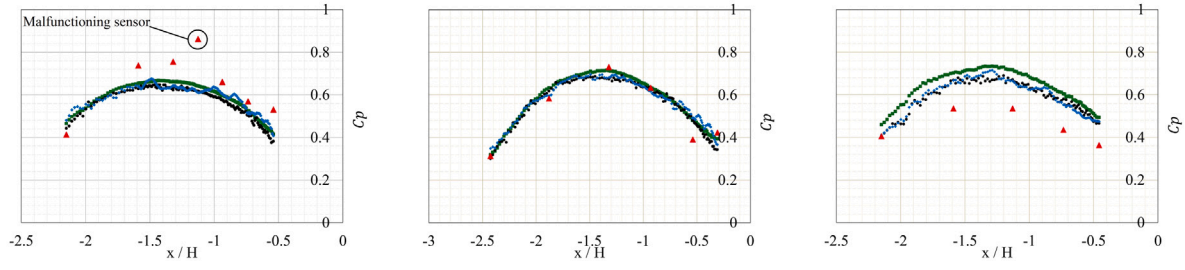



(c) at $tU/H = 13.05$

**Fig. 24.** 3D Static Pelton Bucket. The value of $h \cdot |\nabla C|$ for each particle for the No-Shifting case (left column) and L-NIIPS + ACS (right column) at different time instants for $H/\Delta = 60$.

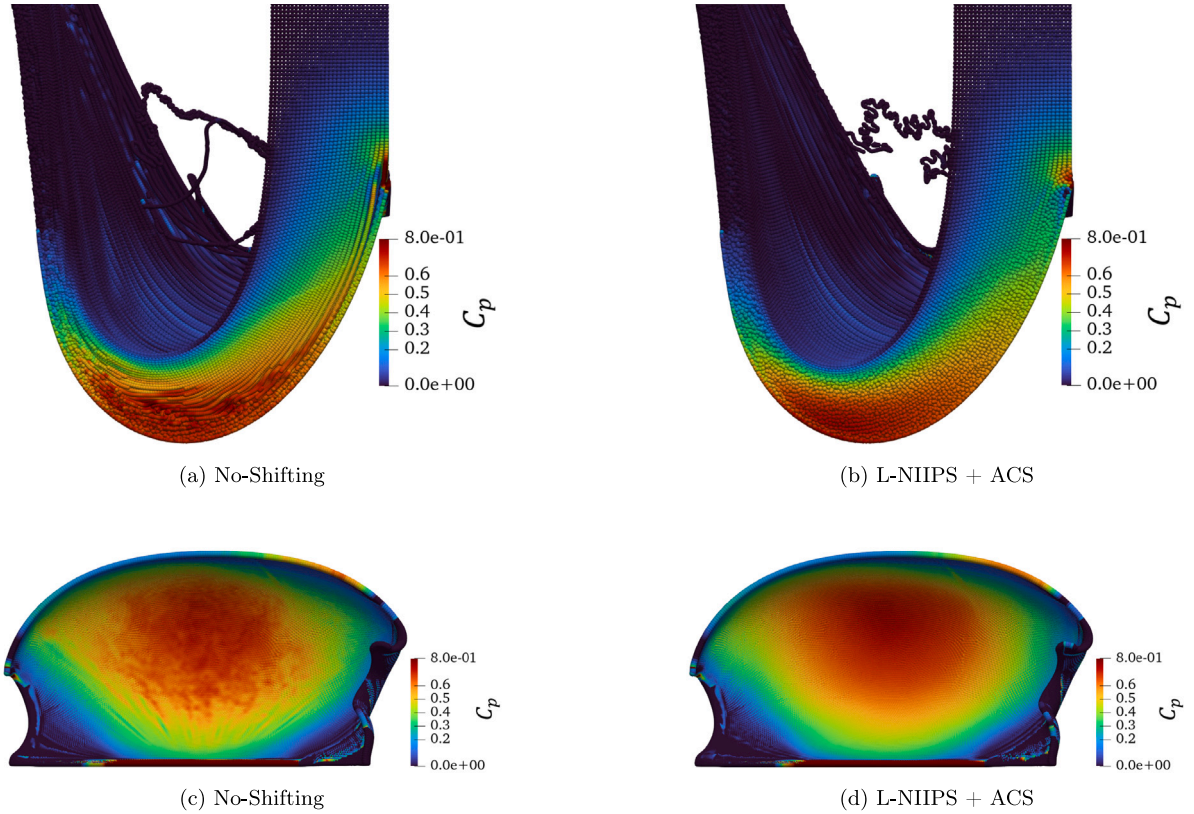



(a) Line X1, $x/H = -1.88$ (b) Line X2, $x/H = -1.32$ (c) Line X3, $x/H = -0.54$




(d) Line Y1, $y/H = -0.62$ (e) Line Y2, $y/H = -0.82$ (f) Line Y3, $y/H = -1.12$

**Fig. 25.** 3D Static Pelton Bucket. Comparison of the pressure distribution along the bucket at the non-dimensional time $tU/H = 13.05$ with L-NIIPS + ACS, No-Shifting case and experimental data [51,52].




(a) No-Shifting (b) L-NIIPS + ACS

(c) No-Shifting (d) L-NIIPS + ACS

**Fig. 26.** 3D Static Pelton Bucket. Comparison of the pressure field at the converged state in time $tU/H = 13.05$ in fluid (first row) and in the bucket (second row) for L-NIIPS with ACS and No-Shifting case for $H/\Delta = 60$.

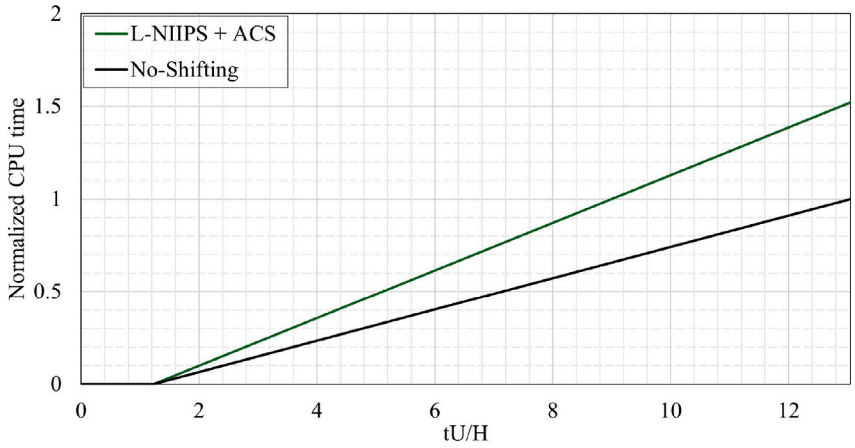



**Fig. 27.** 3D Static Pelton Bucket. Relative CPU time comparison.

GPU architectures represents a promising direction to further enhance computational efficiency.

## CRediT authorship contribution statement

**Mohamad Amin Ghazi:** Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis. **Renato Vacondio:** Writing – review & editing, Supervision, Methodology, Funding acquisition, Formal analysis, Conceptualization. **Jean-Christophe Marongiu:** Writing – review & editing, Validation, Supervision, Software, Methodology, Investigation, Funding acquisition, Conceptualization. **Eduardo Di Costanzo:** Writing – review & editing, Software, Conceptualization.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Acknowledgments

The fourth author was supported by the State Secretariat for Education, Research and Innovation (SERI) of Switzerland and Co-Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or European Research Executive Agency (REA). Neither the European Union nor the granting authority can be held responsible for them.

## Data availability

No data was used for the research described in the article.

## References

[1] Gingold RA, Monaghan JJ. Smoothed particle hydrodynamics: theory and application to non-spherical stars. Mon Not R Astron Soc 1977;181(3):375–89. http://dx.doi.org/10.1093/mnras/181.3.375.

[2] Lucy LB. A numerical approach to the testing of the fission hypothesis. Astron J 1977;82:1013. http://dx.doi.org/10.1086/112164.

[3] Vila JP. On particle weighted methods and smooth particle hydrodynamics. Math Models Methods Appl Sci 1999;09(02):161–209. http://dx.doi.org/10.1142/s0218202599000117.

[4] Oger G, Marrone S, Le Touzé D, de Leffe M. SPH accuracy improvement through the combination of a quasi-Lagrangian shifting transport velocity and consistent ALE formalisms. J Comput Phys 2016;313:76–98. http://dx.doi.org/10.1016/j.jcp.2016.02.039.

[5] Michel J, Colagrossi A, Antuono M, Marrone S. A regularized high-order diffusive smoothed particle hydrodynamics scheme without tensile instability. Phys Fluids 2023;35(10). http://dx.doi.org/10.1063/5.0165036.

[6] Antuono M, Marrone S, Colagrossi A. Coalescing and break-up of viscous drops with surface tension through the Smoothed Particle Hydrodynamics. Comput Methods Appl Mech Engrg 2025;442:118014. http://dx.doi.org/10.1016/j.cma.2025.118014.

[7] Monaghan JJ. SPH and Riemann solvers. J Comput Phys 1997;136(2):298–307. http://dx.doi.org/10.1006/jcph.1997.5732.

[8] Inutsuka Shu-ichiro. Reformulation of smoothed particle hydrodynamics with Riemann solver. J Comput Phys 2002;179(1):238–67. http://dx.doi.org/10.1006/jcph.2002.7053.

[9] Parshikov Anatoly N, Medin Stanislav A. Smoothed particle hydrodynamics using interparticle contact algorithms. J Comput Phys 2002;180(1):358–82. http://dx.doi.org/10.1006/jcph.2002.7099.

[10] Cha S-H, Whitworth AP. Implementations and tests of godunov-type particle hydrodynamics. Mon Not R Astron Soc 2003;340(1):73–90. http://dx.doi.org/10.1046/j.1365-8711.2003.06266.x.

[11] Puri Kunal, Ramachandran Prabhu. Approximate Riemann solvers for the godunov SPH (GSPH). J Comput Phys 2014;270:432–58. http://dx.doi.org/10.1016/j.jcp.2014.03.055.

[12] Cleary Paul W, Monaghan Joseph J. Conduction modelling using smoothed particle hydrodynamics. J Comput Phys 1999;148(1):227–64. http://dx.doi.org/10.1006/jcph.1998.6118.

[13] van Leer Bram. Towards the ultimate conservative difference scheme. V. A second-order sequel to Godunov's method. J Comput Phys 1979;32(1):101–36. http://dx.doi.org/10.1016/0021-9991(79)90145-1.

[14] Avesani Diego, Dumbser Michael, Bellin Alberto. A new class of Moving-Least-Squares WENO–SPH schemes. J Comput Phys 2014;270:278–99. http://dx.doi.org/10.1016/j.jcp.2014.03.041.

[15] Avesani Diego, Dumbser Michael, Vacondio Renato, Righetti Maurizio. An alternative SPH formulation: ADER-WENO-SPH. Comput Methods Appl Mech Engrg 2021;382:113871. http://dx.doi.org/10.1016/j.cma.2021.113871.

[16] Michel J, Antuono M, Oger G, Marrone S. Energy balance in quasi-Lagrangian Riemann-based SPH schemes. Comput Methods Appl Mech Engrg 2023;410:116015. http://dx.doi.org/10.1016/j.cma.2023.116015.

[17] Antona Rubén, Vacondio Renato, Avesani Diego, Righetti Maurizio, Renzi Massimiliano. Towards a high order convergent ALE-SPH scheme with efficient WENO spatial reconstruction. Water 2021;13(17):2432. http://dx.doi.org/10.3390/w13172432.

[18] Antona Rubén, Vacondio Renato, Avesani Diego, Righetti Maurizio, Renzi Massimiliano. A WENO SPH scheme with improved transport velocity and consistent divergence operator. Comput Part Mech 2023;11(3):1221–40. http://dx.doi.org/10.1007/s40571-023-00681-z.

[19] Nogueira Xesús, Ramírez Luis, Clain Stéphane, Loubère Raphaël, Cueto-Felgueroso Luis, Colominas Ignasi. High-accurate SPH method with Multidimensional Optimal Order Detection limiting. Comput Methods Appl Mech Engrg 2016;310:134–55. http://dx.doi.org/10.1016/j.cma.2016.06.032.

[20] Dilts Gary A. Moving-least-squares-particle hydrodynamics?I. Consistency and stability. Internat J Numer Methods Engrg 1999;44(8):1115–55. http://dx.doi.org/10.1002/(sici)1097-0207(19990320)44:8<1115::aid-nme547>3.0.co;2-l.

[21] Eirís Antonio, Ramírez Luis, Fernández-Fidalgo Javier, Couceiro Iván, Nogueira Xesús. SPH-ALE scheme for weakly compressible viscous flow with a posteriori stabilization. Water 2021;13(3):245. http://dx.doi.org/10.3390/w13030245.

[22] Clain S, Diot S, Loubère R. A high-order finite volume method for systems of conservation laws—Multi-dimensional Optimal Order Detection (MOOD). J Comput Phys 2011;230(10):4028–50. http://dx.doi.org/10.1016/j.jcp.2011.02.026.

[23] Monaghan JJ. On the problem of penetration in particle methods. J Comput Phys 1989;82(1):1–15. http://dx.doi.org/10.1016/0021-9991(89)90032-6.

[24] Xu R, Stansby P, Laurence D. Accuracy and stability in incompressible SPH (ISPH) based on the projection method and a new approach. J Comput Phys 2009;228(18):6703–25. http://dx.doi.org/10.1016/j.jcp.2009.05.032.

[25] Lind SJ, Xu R, Stansby PK, Rogers BD. Incompressible smoothed particle hydrodynamics for free-surface flows: A generalised diffusion-based algorithm for stability and validations for impulsive flows and propagating waves. J Comput Phys 2012;231(4):1499–523. http://dx.doi.org/10.1016/j.jcp.2011.10.027.

[26] Sun PN, Colagrossi A, Marrone S, Antuono M, Zhang A-M. A consistent approach to particle shifting in the $\delta$-Plus-SPH model. Comput Methods Appl Mech Engrg 2019;348:912–34. http://dx.doi.org/10.1016/j.cma.2019.01.045.

[27] Antuono M, Sun PN, Marrone S, Colagrossi A. The $\delta$-ALE-SPH model: An arbitrary Lagrangian-Eulerian framework for the $\delta$-SPH model with particle shifting technique. Comput & Fluids 2021;216:104806. http://dx.doi.org/10.1016/j.compfluid.2020.104806.

[28] Michel J, Vergnaud A, Oger G, Hermange C, Le Touzé D. On Particle Shifting Techniques (PSTs): Analysis of existing laws and proposition of a convergent and multi-invariant law. J Comput Phys 2022;459:110999. http://dx.doi.org/10.1016/j.jcp.2022.110999.

[29] Khayyer A, Gotoh H, Shimizu Y. Comparative study on accuracy and conservation properties of two particle regularization schemes and proposal of an optimized particle shifting scheme in ISPH context. J Comput Phys 2017;332:236–56. http://dx.doi.org/10.1016/j.jcp.2016.12.005.

[30] Sun PN, Colagrossi A, Marrone S, Zhang AM. The $\delta$-plus-SPH model: Simple procedures for a further improvement of the SPH scheme. Comput Methods Appl Mech Engrg 2017;315:25–49. http://dx.doi.org/10.1016/j.cma.2016.10.028, URL http://dx.doi.org/10.1016/j.cma.2016.10.028.

[31] Wang P-P, Meng Z-F, Zhang A-M, Ming F-R, Sun P-N. Improved particle shifting technology and optimized free-surface detection method for free-surface flows in smoothed particle hydrodynamics. Comput Methods Appl Mech Engrg 2019;357:112580. http://dx.doi.org/10.1016/j.cma.2019.112580.

[32] Gao Tianrun, Fu Lin. A new particle shifting technique for SPH methods based on Voronoi diagram and volume compensation. Comput Methods Appl Mech Engrg 2023;404:115788. http://dx.doi.org/10.1016/j.cma.2022.115788.

[33] Ghazi Mohamad Amin, Vacondio Renato, Marongiu Jean-Christophe. Localized implicit iterative shifting to improve particle distribution in Smoothed Particle Hydrodynamics. Comput Fluids 2025;106753. http://dx.doi.org/10.1016/j.compfluid.2025.106753.

[34] Rastelli P, Vacondio R, Marongiu JC, Fourtakas G, Rogers Benedict D. Implicit iterative particle shifting for meshless numerical schemes using kernel basis functions. Comput Methods Appl Mech Engrg 2022;393:114716. http://dx.doi.org/10.1016/j.cma.2022.114716.

[35] Xu X, Yu P. A technique to remove the tensile instability in weakly compressible SPH. Comput Mech 2018;62(5):963–90. http://dx.doi.org/10.1007/s00466-018-1542-4.

[36] Suchde Pratik, Leithäuser Christian, Kuhnert Jörg, Bordas Stéphane PA. Volume and mass conservation in Lagrangian meshfree methods. Internat J Numer Methods Engrg 2025;126(2). http://dx.doi.org/10.1002/nme.7657.

[37] Rastelli Pietro. Advances in particle distribution for SPH numerical schemes: from explicit to implicit shifting techniques [Ph.D. thesis], Università degli studi di Parma, Dipartimento di Ingegneria e architettura; 2022, URL https://hdl.handle.net/1889/4843. ©Pietro Rastelli, 2022. Attribuzione 4.0 Internazionale.

[38] Rastelli P, Vacondio R, Marongiu JC. An arbitrarily Lagrangian–Eulerian SPH scheme with implicit iterative particle shifting procedure. Comput Methods Appl Mech Engrg 2023;414:116159. http://dx.doi.org/10.1016/j.cma.2023.116159.

[39] Leduc Julien. étude physique et numérique de l'écoulement dans un système d'injection de turbine pelton [Ph.D. thesis], Écully, France: École Centrale de Lyon; 2010.

[40] Jean-Christophe Marongiu. [Ph.D. thesis], In: Lagrangian numerical method for the simulation of free surface flows: application to pelton turbines, vol. 1, Ecully, France: École Centrale de Lyon; 2007, p. 170, URL http://www.theses.fr/2007ECDL0046. Thèse de doctorat dirigée par Leboeuf, Francis. École Centrale de Lyon, 2007.

[41] Rentschler M, Neuhauser M, Marongiu JC, Parkinson E. Understanding casing flow in pelton turbines by numerical simulation. IOP Conf Ser: Earth Environ Sci 2016;49:022004. http://dx.doi.org/10.1088/1755-1315/49/2/022004.

[42] Rentschler M, Marongiu JC, Neuhauser M, Parkinson E. Overview of SPH-ALE applications for hydraulic turbines in ANDRITZ Hydro. J Hydrodyn 2018;30(1):114–21. http://dx.doi.org/10.1007/s42241-018-0012-y.

[43] Neuhauser Magdalena, Leboeuf Francis, Marongiu Jean-Christophe, Parkinson Etienne, Robb Daniel. Simulations of Rotor2013Stator interactions with SPH-ALE. In: Advances in hydroinformatics. Springer Singapore; 2013, p. 349–61. http://dx.doi.org/10.1007/978-981-4451-42-0_29.

[44] Ishiki Kenshiro, Cunningham Lee S, Rogers Benedict D. Multiphysics coupling using SPH for coastal structures subject to tsunami-driven hydrodynamic and debris impact loads. J Waterw Port, Coast Ocean Eng 2025;151(1). http://dx.doi.org/10.1061/jwped5.wweng-2157.

[45] Hanousek Nicolas, Ranabhat Bikash, English Aaron, Ahmadian Reza. A smoothed particle hydrodynamics method for vertical axis turbine design and assessment. J Hydrodyn 2024;36(5):991–1007. http://dx.doi.org/10.1007/s42241-024-0074-y.

[46] Laha Sumanta, Fourtakas Georgios, Das Prasanta Kuamr, Keshmiri Amir. Graphics processing unit accelerated modeling of thrombus formation in cardiovascular systems using smoothed particle hydrodynamics. Phys Fluids 2025;37(2). http://dx.doi.org/10.1063/5.0250640.

[47] Ma Chong, Oka Masayoshi. Investigation of the influence of different boundary conditions in SPH on ship dynamics. Ocean Eng 2024;310:118571. http://dx.doi.org/10.1016/j.oceaneng.2024.118571.

[48] Ji Zhe, Stanic Milos, Hartono Erwin Adi, Chernoray Valery. Numerical simulations of oil flow inside a gearbox by Smoothed Particle Hydrodynamics (SPH) method. Tribol Int 2018;127:47–58. http://dx.doi.org/10.1016/j.triboint.2018.05.034.

[49] Michel J, Durante D, Colagrossi A, Marrone S. Energy dissipation in violent three-dimensional sloshing flows induced by high-frequency vertical accelerations. Phys Fluids 2022;34(10). http://dx.doi.org/10.1063/5.0114635.

[50] Pilloton C, Bardazzi A, Colagrossi A, Marrone S. SPH method for long-time simulations of sloshing flows in LNG tanks. Eur J Mech B Fluids 2022;93:65–92. http://dx.doi.org/10.1016/j.euromechflu.2022.01.002.

[51] Kvicinsky Sonia. Méthode d'analyse des écoulements 3D à surface libre : application aux turbines Pelton [Ph.D. thesis], Lausanne, Switzerland: École Polytechnique Fédérale de Lausanne (EPFL); 2002, http://dx.doi.org/10.5075/EPFL-THESIS-2526, URL https://infoscience.epfl.ch/handle/20.500.14299/211495. Open access.

[52] Kvicinsky Sonia, Kueny Jean-Louis, Avellan François. Numerical and experimental analysis of free surface flow in a 3D non-rotating pelton bucket. In: Proceedings of the 9th international symposium on transport phenomena and dynamics of rotating machinery, vol. FD18/FD-125, Honolulu, Hawaii, USA: École Polytechnique Fédérale de Lausanne (EPFL); 2002, p. 1–8.

[53] Marongiu Jean-Christophe, Leboeuf Francis, Caro Joëlle, Parkinson Etienne. Free surface flows simulations in Pelton turbines using an hybrid SPH-ALE method. J Hydraul Res 2010;48:40–9. http://dx.doi.org/10.1080/00221686.2010.9641244, URL https://hal.science/hal-00566051.

[54] Marongiu JC, Leboeuf F, Parkinson E. Numerical simulation of the flow in a pelton turbine using the meshless method smoothed particle hydrodynamics: A new simple solid boundary treatment. Proc Inst Mech Eng Part A: J Power Energy 2007;221(6):849–56. http://dx.doi.org/10.1243/09576509jpe465.

[55] Vessaz C, Jahanbakhsh E, Avellan F. Flow simulation of a Pelton bucket using finite volume particle method. IOP Conf Ser: Earth Environ Sci 2014;22(1):012003. http://dx.doi.org/10.1088/1755-1315/22/1/012003.

[56] Perrig Alexandre, Avellan François, Kueny Jean-Louis, Farhat Mohamed, Parkinson Etienne. Flow in a Pelton turbine bucket: Numerical and experimental investigations. J Fluids Eng 2005;128(2):350–8. http://dx.doi.org/10.1115/1.2170120.

[57] Sun Jie, Ge XF, Zheng Yuan. SPH method used for characteristic predictions at Pelton turbine buckets: comparing with the mesh-based method. Eng Comput 2023;40(5):1245–65. http://dx.doi.org/10.1108/ec-06-2022-0394.

[58] Leguizamón Sebastián, Jahanbakhsh Ebrahim, Alimirzazadeh Siamak, Maertens Audrey, Avellan François. FVPM numerical simulation of the effect of particle shape and elasticity on impact erosion. Wear 2019;430–431:108–19. http://dx.doi.org/10.1016/j.wear.2019.04.023.

[59] Neuhauser Magdalena. Development of a coupled SPH-ALE/finite volume method for the simulation of transient flows in hydraulic machines [Ph.D. thesis], (2014ECDL0045). Ecole Centrale de Lyon; 2014, URL https://theses.hal.science/tel-01184761.

[60] Di Costanzo Eduardo, Kühl Niklas, Marongiu Jean-Christophe, Rung Thomas. Incremental model order reduction of smoothed-particle hydrodynamic simulations. Internat J Numer Methods Fluids 2025;97(12):1571–94. http://dx.doi.org/10.1002/fld.70012, URL https://onlinelibrary.wiley.com/doi/abs/10.1002/fld.70012.

[61] Toro Eleuterio. Riemann solvers and numerical methods for fluid dynamics: A practical introduction. Springer; 2009, p. 748. http://dx.doi.org/10.1007/b79761,

[62] Ivings MJ, Causon DM, Toro EF. On Riemann solvers for compressible liquids. Internat J Numer Methods Fluids 1998;28(3):395–418. http://dx.doi.org/10.1002/(SICI)1097-0363(19980915)28:3<395::AID-FLD718>3.0.CO;2-S, URL https://onlinelibrary.wiley.com/doi/abs/10.1002/%28SICI%291097-0363%2819980915%2928%3A3%3C395%3A%3AAID-FLD718%3E3.0.CO%3B2-S.

[63] Wendland Holger. Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree. Adv Comput Math 1995;4(1):389–96. http://dx.doi.org/10.1007/bf02123482.

[64] LeVeque Randall J. Numerical Methods for Conservation Laws. Birkhäuser Basel; 1992, http://dx.doi.org/10.1007/978-3-0348-8629-1.

[65] Hirsch C. Numerical computation of internal and external flows: The fundamentals of computational fluid dynamics. 2nd ed. Elsevier; 2007, http://dx.doi.org/10.1016/b978-0-7506-6594-0.x5037-1, .

[66] Monaghan JJ. Simulating free surface flows with SPH. J Comput Phys 1994;110(2):399–406. http://dx.doi.org/10.1006/jcph.1994.1034.
[67] Guennebaud Gaël, Jacob Benoît, et al. Eigen v3. 2010, http://eigen.tuxfamily.org.
[68] SPHERIC. Test 06: 2-D Incompressible Flow Around a Moving Square Inside a Rectangular Box (A. Colagrossi). 2025, SPHERIC – SPH rEsearch and engineeRing International Community. Online. [Accessed 4 July 2025].
[69] Antuono Matteo, Marrone Salvatore. Weakly compressible approximation of the Taylor–Green vortex solution. Stud Appl Math 2024;154(1). http://dx.doi.org/10.1111/sapm.12792.
[70] Molteni Diego, Colagrossi Andrea. A simple procedure to improve the pressure evaluation in hydrodynamic context using the SPH. Comput Phys Comm 2009;180(6):861–72. http://dx.doi.org/10.1016/j.cpc.2008.12.004.
[71] Marrone S, Antuono M, Colagrossi A, Colicchio G, Le Touzé D, Graziani G. $\delta$-SPH model for simulating violent impact flows. Comput Methods Appl Mech Engrg 2011;200(13–16):1526–42. http://dx.doi.org/10.1016/j.cma.2010.12.016.
[72] Taylor Geoffrey Ingram. Oblique impact of a jet on a plane surface. Philos Trans R Soc Lond Ser A Math Phys Sci 1966;260(1110):96–100. http://dx.doi.org/10.1098/rsta.1966.0034.
[73] Green Mashy D, Vacondio Renato, Peiró Joaquim. A smoothed particle hydrodynamics numerical scheme with a consistent diffusion term for the continuity equation. Comput & Fluids 2019;179:632–44. http://dx.doi.org/10.1016/j.compfluid.2018.11.020.
[74] Colagrossi A, Landrini M. Numerical simulation of interfacial flows by smoothed particle hydrodynamics. J Comput Phys 2003;191(2):448–75. http://dx.doi.org/10.1016/s0021-9991(03)00324-3.