Computer Methods in Applied Mechanics and Engineering 448 (2026) 118428

Contents lists available at ScienceDirect

Comput. Methods Appl. Mech. Engrg.

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

Stabilized explicit material point method for fluid flow and fluid-structure interaction simulations using dual high-order B-spline volume averaging

Zhang Cheng, Shiwei Zhao *, Hao Chen, Jidong Zhao *

Department of Civil and Environmental Engineering, The Hong Kong University of Science and Technology, Clearwater Bay, Kowloon, Hong Kong SAR, China

ARTICLE INFO

Keywords:
Explicit MPM
Weakly compressible fluid
Fluid-structure interaction
Stabilized methods

ABSTRACT

Traditional explicit Material Point Methods (MPM) for weakly compressible fluids often suffer from volumetric locking, cell-crossing instability, and excessive energy dissipation, particularly in fluid-structure interaction (FSI) scenarios. This study presents a stabilized explicit MPM framework that integrates dual high-order B-spline volume averaging to address these challenges. The proposed dual averaging technique simultaneously smooths deformation gradients and pressure fields using cubic B-spline basis functions to eliminate cell-crossing errors and reduce volumetric locking. A blended APIC/FLIP mapping scheme is developed to enhance energy conservation and stability at coarse grid resolutions. The framework is further enhanced by seamlessly integrating various complementary techniques such as δ-correction, pressure smoothing, and specialized boundary handling for more robust and effective modeling of free-surface and FSI problems. The framework is rigorously validated through benchmark cases, including 1D elastic wave propagation, Poiseuille flow, lid-driven cavity flow, water sloshing, dam break, and water impact on elastic obstacles. The simulation results demonstrate a remarkable reduction in pressure oscillations and improved particle distribution uniformity compared to prior MPM approaches. The proposed method establishes a robust and efficient tool for large-deformation FSI problems and bridges gaps in accuracy and stability for industrial-scale applications.

1. Introduction

The study of fluid flow represents a critical interdisciplinary field that transcends fundamental physics and extends its influence into a wide range of engineering disciplines and practical applications, including aerospace engineering, the automotive industry, biomedical engineering, civil and environmental engineering [1–5]. In recent years, the Computational Fluid Dynamics (CFD) has experienced significant advancements and is utilized to address various complex fluid flow problems [6,7], which can be primarily categorized into mesh-based and mesh-free approaches. Mesh-based CFD techniques discretize the fluid domain into finite cells, with numerical solvers applied to each cell to determine flow characteristics. Prominent mesh-based methods include the Finite Difference Method (FDM) [8], the Eulerian-based Finite Element Method (FEM) [9–11], and the Finite Volume Method (FVM) [12]. Each of these methods offers distinct advantages regarding accuracy, stability, and computational efficiency, depending on the flow regime and specific application. In contrast, mesh-free methods offer a more flexible numerical approach by discretizing the computational

* Corresponding authors.
E-mail addresses: ceswzhao@ust.hk (S. Zhao), jzhao@ust.hk (J. Zhao).

https://doi.org/10.1016/j.cma.2025.118428
Received 21 April 2025; Received in revised form 16 September 2025; Accepted 19 September 2025
0045-7825/© 2025 Elsevier B.V. All rights are reserved, including those for text and data mining, AI training, and similar technologies.

domain into particles, with fluid behavior modeled through the interactions between these particles. Mesh-free methods, such as the Smoothed Particle Hydrodynamics (SPH) [13–15], the Material Point Method (MPM) [16–20], the Peridynamics (PD) [21–24], discontinuous Galerkin (DG) method [25], and the Moving Particle Semi-Implicit (MPS) method [26], have garnered increasing attention due to their capacity to efficiently simulate highly dynamic and complex fluid phenomena [27–29]. With their mesh-free and Lagrangian nature, these methods can effectively solve challenging physical problems involving moving boundaries and extensively deformed interfaces, which pose significant difficulties for mesh-based approaches [28]. Meanwhile, their particle-based nature allows highly parallel computing on modern high-performance computing systems such as Graphics Processing Units (GPUs) [30,31].

In recent years, MPM has been significantly advanced for simulations of either compressible or incompressible fluids [32–35]. Evolving from the Particle-in-Cell (PIC) method [36], MPM integrates the advantages of both Eulerian and Lagrangian approaches, functioning as a hybrid technique that effectively employs both grid and particle representations [37,38]. In particular, boundary conditions can be easily implemented through nodes in MPM with no need of additional boundary particles [39]. For compressible flows, the weakly compressible equation of state (EOS), which establishes the relationship between the fluid pressure and bulk modulus, is often incorporated into explicit MPM to model various phenomena, such as free-surface flows [40–42] and multiphase flows [43]. However, the EOS often leads to significant pressure oscillations, which can cause issues with numerical stability and accuracy [44]. By contrast, implicit MPM with consideration of fluid incompressibility demonstrates enhanced performance in mitigating pressure oscillations, e.g., the fractional-step method [45–47] and the monolithic approach [29,48]. Nevertheless, its convergence issue and complex implementation may hinder the application in challenging problems such as fluid-structure interactions (FSI) involving large deformation [44]. Hence, further advancements in explicit MPM deserve greater efforts, especially in the following challenges: (i) volumetric locking due to the incompressibility constraint of the fluid [42,49,50]; (ii) instability caused by cell-crossing [51]; (iii) excessive dissipation of momentum and energy resulting from the information exchange between particles and their surrounding nodes [52–54]; (iv) the accumulation of quadrature errors during particle-node mapping [55,56].

Volumetric locking is a common issue in MPM, particularly when nearly incompressible materials are assigned to material points. In such cases, the volume of material points does not change significantly under stress, which restricts the available deformation modes and results in the “locking” of the simulated material [50,56,57]. To address this locking phenomenon in weakly compressible fluid flows, several smoothing techniques have been developed. Researchers have employed multi-field MPM formulations, wherein stress, strain, and velocity (or displacement) are treated as independent interpolation variables. Notable examples of such approaches include the three-field formulation [42] and the two-field formulation with stabilization [44]. Additionally, deformation-related smoothing methods, such as the $\overline{F}$ method [58] and the $\overline{B}$ method [59], have been widely applied in MPM to mitigate this issue. Building on these methodologies, Zhao et al. [60] combined the $\overline{F}$ method with the particle-grid mapping scheme to achieve the volume-averaged deformation gradient for stress updates. This approach facilitates improved simulations for nearly incompressible materials, such as solids with a Poisson’s ratio around 0.5 or water. Telikicherla and Moutsanidis [61] introduced a projection technique to smooth the volumetric components of strain and stress, incorporating lower-order basis functions for this process. This method has since been refined and successfully applied to fluid analysis [62]. Sun et al. [63] employed a B-spline density smoothing technique to simulate FSI problems using a contact-based approach. He et al. [64] proposed the use of an additional sparse projection grid, performing particle-grid mapping on this grid to smooth volumetric strains and enhance computational efficiency. Although existing methods effectively simulate incompressible materials, the performance of circumventing volumetric locking in fluid simulations still requires further improvement.

When material points move between adjacent background cells, discontinuities in strain and the corresponding stress may occur, leading to cell-crossing instability [62,65]. To address this issue, some methods have been proposed to smooth the gradient of interpolation function, including the generalized interpolation material point (GIMP) [51], the convected particle domain interpolation (CPDI) [66,67], the dual domain material point (DDMP) method [68]. In addition, B-spline basis functions have been integrated into MPM to achieve more accurate discretization of the background domain and to reduce cell-crossing errors [69–71]. It has been demonstrated that the accuracy of numerical results obtained using B-spline functions surpasses that of the aforementioned shape functions, such as GIMP and CPDI [70]. For large deformations, the quadratic B-spline shape function will yield poorer performance in terms of accuracy and convergence compared to cubic and quartic B-spline functions [70]. Therefore, despite a potential increase in computational cost, it is advantageous to adopt higher-order B-spline shape functions. These techniques have been effective in mitigating cell-crossing errors in solid mechanics, but their validity remains unclear for fluid flows.

The mapping process between material points and the background grid in the MPM framework presents significant challenges, as it can result in accuracy loss and potential instability. The traditional Particle-In-Cell (PIC) method [36] can lead to considerable energy dissipation, while the Fluid-Implicit-Particle (FLIP) method [72] may introduce numerical noise and instability. Although the two methods are often combined, this blending does not fully mitigate the aforementioned issues [52]. The newly developed affine PIC (APIC) method has shown effectiveness in both energy conservation and numerical stability for fluid dynamics problems. However, it may experience accuracy loss at lower grid and particle resolutions [53]. Similar challenges are observed in the Taylor PIC (TPIC) method [54]. As an extension of the APIC approach, the polynomial PIC (PolyPIC) method [53] addresses some of the limitations of APIC by incorporating additional polynomial modes, but this also increases computational cost. Therefore, the development of a more efficient mapping scheme remains a critical concern for fluid flows in MPM.

The standard MPM is known to be prone to discretization and quadrature errors, primarily due to the particle clumping in regions with significant shear [29,56]. Some studies have been conducted to quantify these errors [73,74], and several methods for mitigating this issue have been proposed [51,66]. Kularathna and Soga [46] employ Gauss points for evaluating volume integrals in internal cells, while Zhang et al. [75] introduce a particle shifting technique (PST) used in SPH to maintain uniform particle distribution.

However, PST employs a nearest-neighbor search for particles, which incurs considerable computational overhead. Recently, a novel particle shifting technique, termed δ-correction, was proposed by Baumgarten and Kamrin [56]. This PST method leverages the grid to minimize nodal volume errors and reduce computational cost. Their comparative study demonstrated that the δ-correction approach effectively mitigates the quadrature errors induced by particle clusters.

This work aims to present a stabilized explicit MPM framework for weakly compressible fluids, effectively addressing the aforementioned challenges. A dual volume averaging technique based on particle-grid mapping and cubic B-spline basis functions is proposed to smooth the assumed deformation gradient $\overline{F}$, in conjunction with a π projection technique to smooth pressure. The innovative combination significantly reduces volumetric locking and cell-crossing errors, leading to a more stabilized and accurate simulation for fluid flow simulations in explicit MPM. Moreover, a blended APIC/FLIP scheme is proposed to minimize particle-grid mapping errors and ensure energy conservation, even at relatively coarse grid resolutions. By seamlessly integrating complementary techniques, we propose a unified framework that ensures high-order information exchange between material particles and background grids. This approach yields robust computational performance while maintaining efficiency. To mitigate particle clustering, the δ-correction technique is employed to reduce quadrature errors and promote a more uniform particle distribution. For simulating FSI problems, a DEM-enriched contact algorithm is incorporated. Additionally, free surface tracking and pressure smoothing techniques are implemented to optimize the pressure field effectively.

This rest of the paper is organized as follows. Section 2 introduces the governing equations of fluid flow, the weakly compressible equation of state, and the MPM solution strategy. Section 3 presents the proposed stabilized framework designed to address volumetric locking, quadrature errors, and errors resulting from cell-crossing and the particle-grid mapping process. A computational algorithm, including detailed implementation aspects, is also summarized in this section. In Section 4, numerical examples are conducted and analyzed to evaluate the performance of the proposed explicit MPM in simulating various fluid flow problems. Finally, Section 5 provides the conclusions.

2. Explicit MPM formulation for weakly-compressible fluids

2.1. Governing equations

A dynamic problem can be controlled by the momentum balance equation in an updated Lagrangian form:

$$\rho\dot{\boldsymbol{v}} = \nabla\cdot\boldsymbol{\sigma} + \rho\boldsymbol{b} \tag{1}$$

where $\boldsymbol{v}$ is the velocity, $\boldsymbol{\sigma}$ is the Cauchy stress tensor, ρ is the mass density and $\boldsymbol{b}$ is an acceleration term induced by body force imposed on materials.

The following general boundary conditions are typically considered for a mechanical problem:

$$\boldsymbol{u} = \bar{\boldsymbol{u}} \quad \text{on} \quad \Gamma_u \tag{2a}$$

$$\boldsymbol{\sigma}\cdot\boldsymbol{n} = \bar{\boldsymbol{t}} \quad \text{on} \quad \Gamma_t \tag{2b}$$

where $\boldsymbol{n}$ is the unit normal vector pointing outward in the present domain Ω; $\bar{\boldsymbol{u}}$ and $\bar{\boldsymbol{t}}$ are the prescribed material displacement on boundary Γ_u and traction on boundary Γ_t, respectively; Eqs. (2a) and (2b) are Dirichlet and Neumann boundary conditions, respectively.

By introducing the virtual displacement vector $\delta\boldsymbol{u}$ and performing an integration of Eq. (1) over the solution domain Ω, the weak form is derived as:

$$\int_\Omega \rho\dot{\boldsymbol{v}}\cdot\delta\boldsymbol{u}dV = \int_\Omega \rho\boldsymbol{b}\cdot\delta\boldsymbol{u}dV + \int_{\partial\Omega}\bar{\boldsymbol{t}}\cdot\delta\boldsymbol{u}dS - \int_\Omega \boldsymbol{\sigma}\cdot\nabla\delta\boldsymbol{u}dV \tag{3}$$

2.2. Weakly compressible fluid model

The fluid stress can be split into two components, namely:

$$\boldsymbol{\sigma} = \boldsymbol{\tau} - p\mathbf{I} \tag{4}$$

where $\boldsymbol{\tau} = 2\mu\dot{\boldsymbol{\epsilon}}'$ is the deviatoric stress tensor, and p represents the volumetric stress tensor, i.e., the fluid pressure (with compression being positive), and μ is the dynamic viscosity. $\mathbf{I}$ is the identity tensor, and $\dot{\boldsymbol{\epsilon}}'$ is the deviatoric part of strain rate tensor, with the strain rate $\dot{\boldsymbol{\epsilon}}$ defined as follows:

$$\dot{\boldsymbol{\epsilon}} = \frac{1}{2}\left(\nabla\boldsymbol{v} + (\nabla\boldsymbol{v})^T\right) \tag{5}$$

The fluid pressure can be explicitly given by the equation of state (EOS), i.e.,

$$\frac{Dp}{Dt} = -K\frac{D\epsilon_v}{Dt} \tag{6}$$

where K is the bulk modulus, $\epsilon_v = \mathrm{tr}(\dot{\boldsymbol{\epsilon}})$ is the volumetric strain. The selection of K for water should satisfy the requirement that density fluctuations remain below 3 %, as specified by Chen et al. [44]. Under this condition, water can be treated as weakly compressible fluid. In our numerical examples, we adhere to this criterion when choosing K.

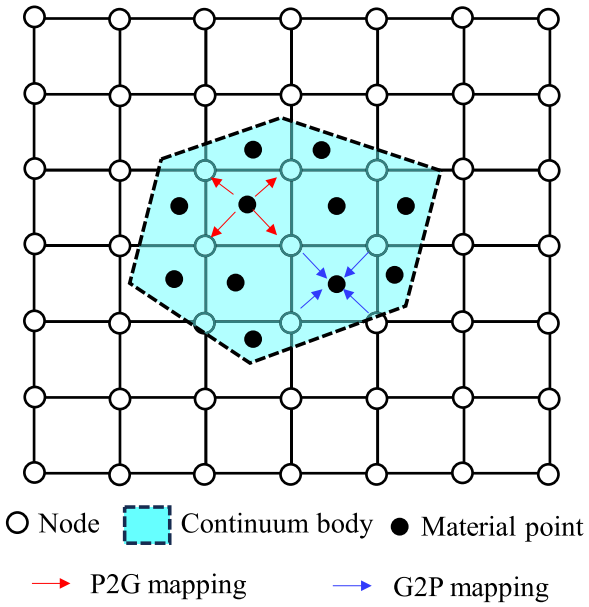

Fig. 1. Illustration of information mapping between particles and the background nodes in MPM.

2.3. General MPM implementation

As illustrated in Fig. 1, the continuum body in MPM is discretized into a collection of Lagrangian material points that carry physical properties such as mass and momentum. The background grid, composed of structural elements, is used for the mapping process, which includes both particle-to-grid (P2G) and grid-to-particle (G2P) mappings. The P2G mapping refers to the transfer of physical information from material points to grid nodes, while the G2P mapping involves the transfer of data from grid nodes back to the material points. Additionally, a shape function facilitates the communication of information between the material points and the background grid. During the P2G mapping, physical data from the material points are mapped to the grid nodes, enabling the solution of the momentum equilibrium equations. Subsequently, the results from the nodes are mapped back to the material points, updating their state variables. The detailed MPM procedure is summarized below.

(1) P2G mapping. The mass and momentum carried by material points are projected into nodes through the use of the shape function:

$$m_i^n = \sum_p m_p w_{ip}^n \tag{7}$$

$$m_i^n \boldsymbol{v}_i^n = \sum_p m_p \boldsymbol{v}_p^n w_{ip}^n \tag{8}$$

where a superscript n represents the physical quantity at time step n, m_i^n represents the nodal mass, which is derived from the particle mass m_p^n, and $\boldsymbol{v}_i^n$ represents the nodal velocity, mapped from the particle velocity $\boldsymbol{v}_p^n$. The function $w_{ip}^n = w_i^n(\boldsymbol{x}_p)$ is the interpolation basis function that maps values from node i to the position of material point p, and $\sum_p$ denotes the summation over all material points that are affected by the basis function associated with node i.

(2) Grid solution. The weak form in Eq. (3) is discretized in both spatial and temporal domains, and then using the explicit Euler method to update the nodal velocity (momentum) as follows:

$$m_i^n \boldsymbol{v}_i^{n+1} = m_i^n \boldsymbol{v}_i^n + \left(\boldsymbol{f}_{i,ext}^n + \boldsymbol{f}_{i,int}^n \right) \Delta t \tag{9}$$

where,

$$\boldsymbol{f}_{i,int}^n = -\sum_p V_p^n \boldsymbol{\sigma}_p^n \cdot \nabla w_{ip}^n \tag{10}$$

$$\boldsymbol{f}_{i,ext}^n = \sum_p m_p \boldsymbol{b}_p w_{ip}^n + \int_{\partial\Omega_\tau} \bar{\boldsymbol{t}} w_{ip} dS \tag{11}$$

(3) G2P mapping. The velocities of material points can be updated in terms of nodal velocities through G2P mapping. Two common ways are used to update the velocity: (i) the FLIP method, which updates the velocity using the nodal acceleration, and (ii) the PIC method, which directly maps the updated nodal velocity to the material point. These two schemes can be expressed as:

$$\left(\boldsymbol{v}_p^{n+1} \right)_{\text{FLIP}} = \boldsymbol{v}_p^n + \sum_i w_{ip} \boldsymbol{a}_i^{n+1} \Delta t \tag{12}$$

$$\left(\boldsymbol{v}_p^{n+1}\right)_{\mathrm{PIC}} = \sum_i w_{ip}\boldsymbol{v}_i^{n+1} \tag{13}$$

where a superscript n or $n+1$ represents the physical quantity at time step n or $n+1$; $\boldsymbol{v}_i$ and $\boldsymbol{a}_i$ are the velocity and acceleration at nodes, respectively; w_{ip} denotes the basis function for interpolation. In general, the FLIP and PIC schemes can be blended as:

$$\boldsymbol{v}_p^{n+1} = (1-\beta)\left(\boldsymbol{v}_p^{n+1}\right)_{\mathrm{FLIP}} + \beta(\boldsymbol{v}_p^{n+1})_{\mathrm{PIC}} \tag{14}$$

$$\boldsymbol{x}_p^{n+1} = \boldsymbol{x}_p^n + \sum_i w_{ip}\boldsymbol{v}_i^{n+1}\Delta t \tag{15}$$

where $\boldsymbol{x}_p$ is the position of the material point, and β is the blending coefficient, which ranges from 0 to 1. The total PIC scheme is applied when $\beta = 1$, while the total FLIP scheme is used when $\beta = 0$. The PIC method enhances numerical stability, though it can introduce excessive damping, potentially leading to inaccuracies in dynamic simulations. On the other hand, FLIP better preserves kinetic energy but may result in oscillations, especially in stress. The combined kinematics update strikes a balance, mitigating numerical noise while reducing artificial damping.

(4) Grid reset. Once the acceleration, velocity, and position of the material point are updated, all the physical information carried by the background grid is reset. The simulation then advances to the next step to repeat the above procedure.

2.4. Advanced particle-grid mapping scheme

Conventional PIC and FLIP mapping schemes often fail to simultaneously meet the requirements for energy conservation and numerical stability [36,72]. Despite the use of the blended PIC/FLIP approach, significant numerical noise remains unavoidable during simulations. The APIC mapping scheme mitigates the energy dissipation issue inherent in the standard PIC method by replacing the locally constant velocity description with a locally affine for each material point [52,76]. This enhancement allows the APIC method to more effectively preserve both linear and angular momentum, while also significantly reducing numerical noise. Unlike the traditional PIC scheme, the APIC method additionally introduces changes at both P2G and G2P procedures.

At the P2G stage, the APIC scheme introduces a matrix $\boldsymbol{C}_p^n$, where the local velocity of a particle at the grid node $\boldsymbol{x}_i$ is expressed as $\boldsymbol{v}_p^n + \boldsymbol{C}_p^n(\boldsymbol{x}_i - \boldsymbol{x}_p^n)$. This formulation defines the transfer from particles to the grid, i.e.,

$$m_i^n\boldsymbol{v}_i^n = \sum_p w_{ip}^n m_p\left(\boldsymbol{v}_p^n + \boldsymbol{C}_p^n(\boldsymbol{x}_i - \boldsymbol{x}_p^n)\right) \tag{16}$$

where $\boldsymbol{C}_p^n = \boldsymbol{B}_p^n\left(\boldsymbol{D}_p^n\right)^{-1}$, $\boldsymbol{D}_p^n$ is analogous to an inertia matrix for affine motion, given by

$$\boldsymbol{D}_p^n = \sum_i w_{ip}\left(\boldsymbol{x}_i - \boldsymbol{x}_p^n\right)\left(\boldsymbol{x}_i - \boldsymbol{x}_p^n\right)^{\mathrm{T}} \tag{17}$$

$\boldsymbol{B}_p^n$ contains the angular momentum information and is modified during the process of mapping data from the grid to material points (G2P). The corresponding transfer from the grid to material points is expressed as follows:

$$\boldsymbol{B}_p^{n+1} = \sum_i w_{ip}\boldsymbol{v}_i^{n+1}\left(\boldsymbol{x}_i - \boldsymbol{x}_p^n\right)^{\mathrm{T}} \tag{18}$$

The APIC mapping scheme effectively mitigates the dissipation inherent in the traditional PIC method, while being less sensitive to the numerical noise characteristic of the FLIP method. However, it may suffer from a loss of accuracy when applied at lower grid and particle resolutions [53]. Motivated by the blending PIC/FLIP, we propose a hybrid FLIP and APIC scheme to overcome this issue. The only distinction between the blended APIC/FLIP method and the original APIC lies in the velocity update, i.e.,

$$\boldsymbol{v}_p^{n+1} = (1-\beta)\left(\boldsymbol{v}_p^n + \sum_i w_{ip}\boldsymbol{a}_i^{(n+1)}\Delta t\right) + \beta\left(\sum_i w_{ip}\boldsymbol{v}_i^{(n+1)}\right), \quad \text{APIC/FLIP} \tag{19a}$$

$$\boldsymbol{v}_p^{n+1} = \sum_i w_{ip}\boldsymbol{v}_i^{(n+1)}, \quad \text{APIC} \tag{19b}$$

where the blending coefficient β ranges from 0 to 1. Specifically, in our simualtions, $\beta = 1$ is chosen if not explicitly specified. The blended APIC/FLIP scheme offers a good balance between energy conservation and accuracy (see the dam break case in Section 4.5), making it particularly suitable for dynamic problems at relatively coarse grid resolutions.

2.5. Stability condition

The explicit MPM uses the central difference method for time integration, which is conditionally stable. For stability, the time step Δt must be smaller than a critical time step Δt_c. This condition is commonly known as the Courant-Friedrichs-Lewy (CFL) condition [77–79], which is given by:

$$\Delta t_c = \min\frac{h^e}{c_p} \tag{20}$$

where h^e is the characteristic length of the element and c_p is the sound speed. For the weakly compressible fluid, c_p can be calculated using the formula $\sqrt{K_f/\rho_f}$.

However, in large-deformation problems where the velocity of material points is comparable to the sound speed, the particle velocity must be considered when determining the critical time step. Therefore, the CFL condition is modified to [78]:

$$\Delta t_c = \frac{h^e}{\max(c_p + |\boldsymbol{v}_p|)} \tag{21}$$

where $\boldsymbol{v}_p$ is the particle velocity.

3. Stabilized high-order MPM framework

The cubic B-spine shape function is employed to stabilize the explicit MPM in conjunction with a dual volume averaging technique. The stabilized MPM solver is further enhanced by correction and smoothing, and specific boundary handling techniques. The full computation procedure is summarized in Algorithm 1.

Algorithm 1: Pseudocode of MPM with the proposed explicit stabilized algorithm.

Input: Simulation time-steps;
foreach *timestep* **do**
  Nodes reset and mapping:
    Reset background nodes dynamics.
    Compute basis functions at material points.
    Map material points' mass and momentum to nodes.
  Boundary Handling:
    Detect free-surface nodes.
    Perform DEM-enriched contact analysis.
  Dual volume averaging:
    Calculate volume-averaged deformation gradient.
    Compute strain and stress for material points.
    Calculate volume-averaged pressure.
  Solution of Variables:
    Map internal and external forces from material points to nodes.
    Update the nodal velocity and acceleration.
    Map information from nodes to material points.
  Correction and smoothing:
    Apply δ-correction and relocate material points.
    Apply pressure smoothing for visualization.

3.1. Cubic B-spline shape function

The B-spline basis functions can be defined through a knot vector, which is a set of ordered, non-decreasing values in R known as knots. A knot vector is defined by

$$\Xi = \{\xi_1, \xi_2, \xi_i, \dots, \xi_{\mathrm{n+p+1}}\} \tag{22}$$

where ξ_i is the i^{th} knot, i is the knot index, $i = 1, 2, \dots, n + p + 1$.

Let p denote the polynomial degree and n the total number of B-spline basis functions. The knot vector subdivides the parametric domain into different segments or elements. The B-spline basis functions for a given knot vector are constructed step by step using the Cox-de Boor recursion formula [80],

$$N_{i,p}(\xi) = \frac{\xi - \xi_i}{\xi_{i+p} - \xi_i} N_{i,p-1}(\xi) + \frac{\xi_{i+p+1} - \xi}{\xi_{i+p+1} - \xi_{i+1}} N_{i+1,p-1}(\xi), \quad \text{for } p > 0 \tag{23}$$

with

$$N_{i,0}(\xi) = \begin{cases} 1 & \text{if } \xi_i \leqslant \xi < \xi_{i+1}, \\ 0 & \text{otherwise.} \end{cases} \tag{24}$$

While quadratic B-spline basis functions have been widely used in MPM [29,62,69,70,81], few studies explored the application of cubic (or higher-order) B-spline basis functions, particularly in the context of fluid flow analysis. In general, there are four types

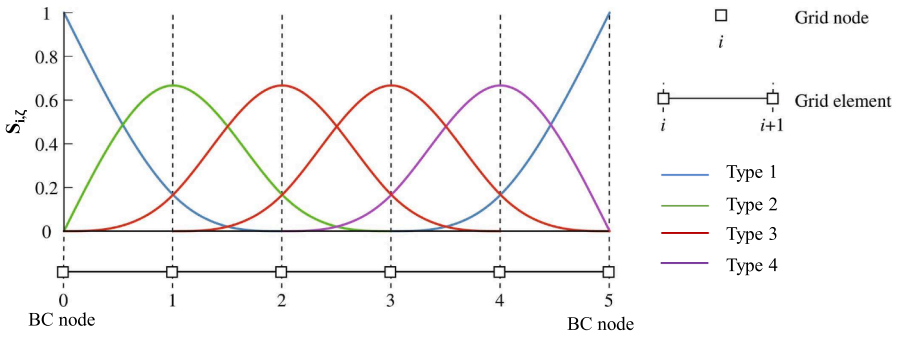

Fig. 2. Cubic B-spline shape functions for a series of five elements in 1D.

of cubic B-spline basis functions [71] (see Fig. 2 for the 1D case). The *type 1* basis function is defined at boundary nodes $\zeta_i = \zeta_B$, i.e. $i = 0$ or 5, given by:

$$S^1_{i,\zeta}(r) = \begin{cases} \frac{1}{6}r^3 + r^2 + 2r + \frac{4}{3}, & -2 \le r \le -1 \\ -\frac{1}{6}r^3 + r + 1, & -1 \le r \le 0 \\ \frac{1}{6}r^3 - r + 1, & 0 \le r \le 1 \\ -\frac{1}{6}r^3 + r^2 - 2r + \frac{4}{3}, & 1 \le r \le 2 \end{cases} \tag{25}$$

where $r = (\zeta_p - \zeta_i)/h$, ζ_p is the position of particles, ζ_i is nodal position, h is element size.

The *type 2* basis function corresponds to the node which is located at one cell away from left boundary $\zeta_i = \zeta_B + h$, i.e. $i = 1$, given by:

$$S^2_{i,\zeta}(r) = \begin{cases} -\frac{1}{3}r^3 - r^2 + \frac{2}{3}, & -1 \le r \le 0 \\ \frac{1}{2}r^3 - r^2 + \frac{2}{3}, & 0 \le r \le 1 \\ -\frac{1}{6}r^3 + r^2 - 2r + \frac{4}{3}, & 1 \le r \le 2 \end{cases} \tag{26}$$

The *type 3* basis function corresponds to the node where the index i is located at least two cells away from any boundary $\zeta_i \geqslant \zeta_B + 2h$, i.e. $i = 2$ or 3, given by:

$$S^3_{i,\zeta}(r) = \begin{cases} \frac{1}{6}r^3 + r^2 + 2r + \frac{4}{3}, & -2 \le r \le -1 \\ -\frac{1}{2}r^3 - r^2 + \frac{2}{3}, & -1 \le r \le 0 \\ \frac{1}{2}r^3 - r^2 + \frac{2}{3}, & 0 \le r \le 1 \\ -\frac{1}{6}r^3 + r^2 - 2r + \frac{4}{3}, & 1 \le r \le 2 \end{cases} \tag{27}$$

The *type 4* basis function is related to the node which is located at one cell away from right boundary $\zeta_i = \zeta_B - h$, i.e. $i = 4$, given by:

$$S^4_{i,\zeta}(r) = \begin{cases} \frac{1}{6}r^3 + r^2 + 2r + \frac{4}{3}, & -2 \le r \le -1 \\ -\frac{1}{2}r^3 - r^2 + \frac{2}{3}, & -1 \le r \le 0 \\ \frac{1}{3}r^3 - r^2 + \frac{2}{3}, & 0 \le r \le 1 \end{cases} \tag{28}$$

By using C^2-continuous cubic B-spline basis functions, we can maintain a smooth representation of the strain rate as material points move across element boundaries, effectively eliminating cell-crossing errors.

3.2. Dual volume averaging

We introduce a dual volume averaging technique that extends the volume-averaged deformation gradient [60] to further consider the volume-averaged pressure, i.e., volume averaging both deformation and pressure. It is important to highlight that the efficacy of the volume-averaged deformation gradient [60] has not been clearly articulated in previous studies. In this work, we provide a rigorous explanation of how our proposed dual volume averaging technique effectively addresses the issue of volumetric locking in explicit MPM.

3.2.1. Implementation

As shown in Fig. 3, the support region of the cubic B-spline functions at each node influences 16 elements in 2D. In the first volume averaging process, the Jacobians J_p, which represent the determinant of the deformation gradient at the particle location, are mapped to the nodes to obtain $\bar{J}_I$. Subsequently, $\bar{J}_I$ is mapped back to the particles to obtain the volume-averaged $\bar{J}_p$. The original relative deformation gradient $\Delta \boldsymbol{F}_p$ is then replaced by the volume-averaged $\Delta \bar{\boldsymbol{F}}_p$, based on $\bar{J}_p$, for pressure calculations. In the second

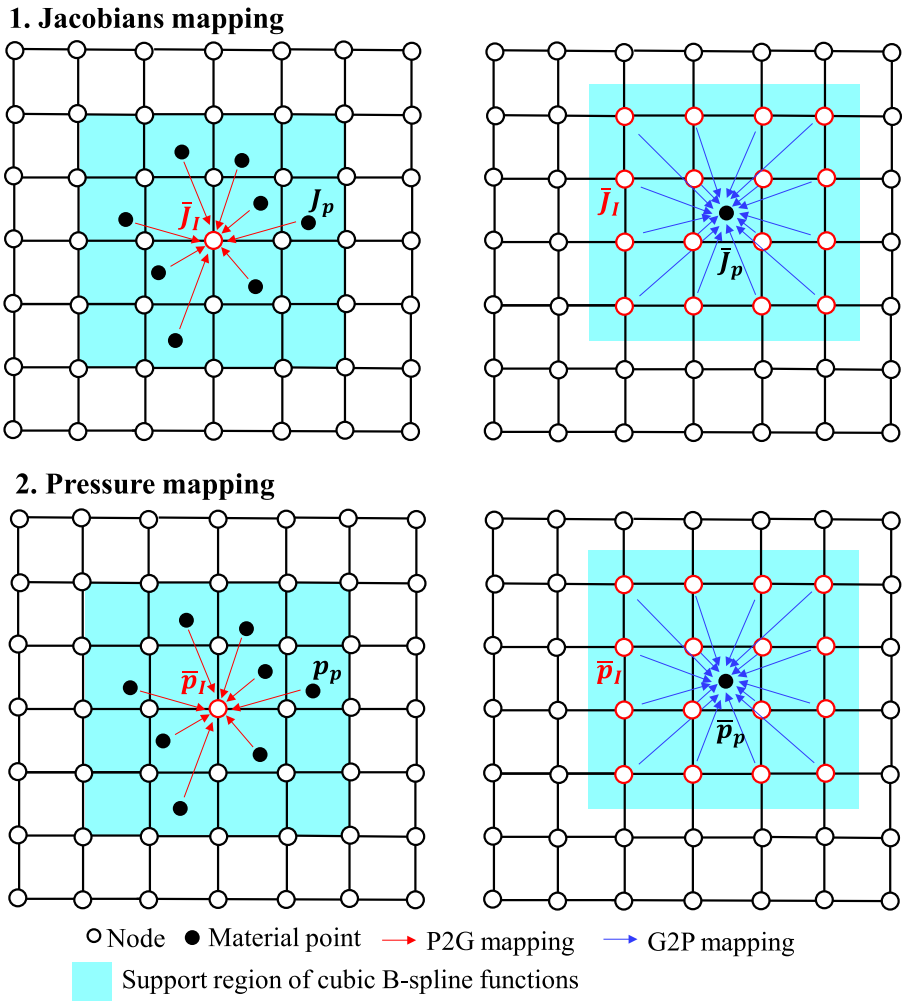

Fig. 3. Illustration of dual volume averaging based on cubic B-spline shape functions.

volume averaging step, a similar mapping procedure is applied to the particle pressure p to obtain the volume-averaged pressure $\bar{p}_p$. The detailed implementation is depicted as follows. With the original relative deformation gradient at t^{n+1}, i.e.,

$$\Delta \boldsymbol{F}_p^{n+1} = \boldsymbol{I} + \Delta t \sum_i \boldsymbol{v}_i^{n+1} \otimes \nabla w_{ip} \tag{29}$$

the deformation gradient can be updated as

$$\boldsymbol{F}_p^{n+1} = \Delta \boldsymbol{F}_p^{n+1} \cdot \boldsymbol{F}_p^n \tag{30}$$

The volume of material points can be updated as:

$$V_p^{n+1} = J_p^{n+1} \cdot V_p^0 \tag{31}$$

where $J_p^{n+1} = \det \boldsymbol{F}_p^{n+1}$ is the determinant of deformation gradient at t^{n+1}.

The nodal volume-averaged Jacobians is given by

$$\bar{J}_i^{n+1} = \sum_p w_{ip} V_p^n (\bar{J}_p^n \Delta J_p^{n+1}) / V_i^n \tag{32}$$

with

$$V_i^n = \sum_p w_{ip} V_p^n \tag{33a}$$

$$\Delta J_p^{n+1} = \det(\Delta \boldsymbol{F}_p^{n+1}) \tag{33b}$$

$$\bar{J}_p^n = \det(\bar{\boldsymbol{F}}_p^n) \tag{33c}$$

The relative assumed deformation gradient $\Delta\bar{\boldsymbol{F}}_p^{n+1}$ can be obtained by

$$\Delta\bar{\boldsymbol{F}}_p^{n+1} = \left(\frac{\sum_i w_{ip}\bar{J}_i^{n+1}}{\bar{J}_p^n \Delta J_p^{n+1}} \right)^{1/\dim} \Delta\boldsymbol{F}_p^{n+1} \tag{34}$$

where dim is simulation dimension, for 2D and 3D cases, dim is set to 2 and 3, respectively. w_{ip} is cubic B-spline basis function for interpolation from node i to the material point p.

Therefore, the fluid pressure can be computed using the EOS presented in Eq. (6), based on the relative assumed deformation gradient $\Delta\bar{\boldsymbol{F}}_p^{n+1}$. Additionally, the shear stress, as described in Eq. (4), can also be calculated. Next, update for deformation gradient $\bar{\boldsymbol{F}}_p^{n+1} = \Delta\bar{\boldsymbol{F}}_p^{n+1} \cdot \bar{\boldsymbol{F}}_p^n$, as the nodal volume-averaged Jacobians given by Eq. (32) for the subsequent step must be calculated using the determinant of $\bar{\boldsymbol{F}}_p^{n+1}$.

The π projection technique is then applied [61] to the fluid pressure, i.e.,

$$\pi(\bar{p})_i^{n+1} = \sum_p w_{ip} V_p^n p^{n+1} / V_i^n \tag{35a}$$

$$V_i^n = \sum_p w_{ip} V_p^n \tag{35b}$$

so that the volume-averaged pressure reads

$$\bar{p}_p^{n+1} = \sum_i w_{ip} \pi(\bar{p})_i^{n+1} \tag{36}$$

Note that p_p^{n+1} represents the fluid pressure computed using the EOS as given in Eq. (6), while $\bar{p}_p^{n+1}$ denotes the volume-averaged pressure after applying the π-projection for p_p^{n+1}. This averaged pressure will be used in Eq. (9) to solve the momentum equation and update the nodal velocity. By sequentially applying volume averaging for both deformation gradient and pressure, volumetric locking can be significantly alleviated.

3.2.2. Why it works

Although the constraint counting method [82], which originated from the FEM, has been utilized in MPM to provide meaningful insights [62,64], it primarily serves as a qualitative tool for assessing the propensity for locking. Hence, we elaborate a detailed discussion as follows. Volumetric locking occurs when an excessive number of incompressibility constraints are imposed on the deformation of elements, leading to an overly stiff response [50,57]. To clarify this phenomenon, we reformulate the explicit kinematic equation in matrix form for theoretical analysis, as follows:

$$\boldsymbol{M}\ddot{\boldsymbol{U}} + (\boldsymbol{K}_{\text{dev}} + \boldsymbol{K}_{\text{vol}})\boldsymbol{U} = \boldsymbol{f}^{\text{ext}} \tag{37}$$

where $\boldsymbol{M} = \text{diag}(\boldsymbol{M}_1, \boldsymbol{M}_2, \ldots) \in \mathbb{R}^{n_u \times n_u}$ represents the diagonal mass matrix. $n_u = d \cdot n$ denotes nodal degree of freedom, which is the product of the number of the dimension d and the number of active nodes n. $\boldsymbol{U} = [\boldsymbol{u}_1^T, \boldsymbol{u}_2^T, \ldots, \boldsymbol{u}_n^T]^T \in \mathbb{R}^{n_u}$ is displacement vector. $\boldsymbol{f}_{ext} \in \mathbb{R}^{n_u}$ denotes external force vector.

The deviatoric stiffness matrix $\boldsymbol{K}_{\text{dev}}$ and volumetric stiffness matrix $\boldsymbol{K}_{\text{vol}}$ can be written as follows:

$$\boldsymbol{K}_{\text{dev}} = \sum_p \boldsymbol{B}_p^T \boldsymbol{D}_{\text{dev},p} \boldsymbol{B}_p V_p \tag{38a}$$

$$\boldsymbol{K}_{\text{vol}} = K \sum_p V_p \boldsymbol{G}_p \boldsymbol{G}_p^T \tag{38b}$$

where $\boldsymbol{B}_p$ represents strain-displacement matrix, $\boldsymbol{D}_{\text{dev},p}$ is deviatoric material matrix, K denotes bulk modulus, $\boldsymbol{G}_p = [\nabla w_1(\boldsymbol{x}_p)^T, \nabla w_2(\boldsymbol{x}_p)^T, \ldots, \nabla w_n(\boldsymbol{x}_p)^T] \in \mathbb{R}^{n_u}$ is basis function gradients.

Due to the characteristics of a weakly compressible fluid, each material point is assigned a volumetric constraint, expressed as follows:

$$\boldsymbol{G}_p^T \boldsymbol{U} \approx 0 \qquad p = 1, 2, \ldots, n_p \tag{39}$$

We have the system of kinematic equations presented in Eq. (37) and a set of constraint equations outlined in Eq. (39). In MPM, the number of constraint equations n_p exceeds the dimension n_u of the volumetric stiffness matrix $\boldsymbol{K}_{vol}$. This results in a statically indeterminate motion matrix equation, making it impossible to obtain a unique exact solution. Instead, only an approximate solution can be achieved. The resulting error may propagate into the strain field and be amplified by the bulk modulus, ultimately leading to pressure oscillations.

In this study, the higher-order volume-averaging technique employs cubic B-spline basis functions, which have a larger support region and activate more nodes, thereby increasing the dimension of the stiffness matrix n_u. Additionally, the volume-averaged deformation gradient establishes relationships between material points through the use of nodes, which reduces the number of constraint equations n_p to the number of active nodes n. Consequently, the kinematic matrix system can be solved to yield a unique and accurate solution, thereby avoiding errors associated with an over-determined system.

The second volume averaging for pressure can be reformulated in continuous form as follows:

$$\bar{p}(\boldsymbol{x}) = \sum_i w_i(\boldsymbol{x}) \frac{\sum_p w_{ip} V_p p(\boldsymbol{x}_p)}{\sum_p w_{ip} V_p} = \sum_i w_i(\boldsymbol{x}) \frac{\int w_i(\boldsymbol{y}) p(\boldsymbol{y})\, d\boldsymbol{y}}{\int w_i(\boldsymbol{z})\, d\boldsymbol{z}} \tag{40}$$

This equation can be viewed as a filtering process that reconstructs the original pressure field $p(\boldsymbol{y})$ into a new pressure field $\bar{p}(\boldsymbol{x})$. To investigate its filtering characteristics, we assume a pressure field defined by $\cos(k\boldsymbol{x} + b)$, where w and b represent the wave number (proportional to spatial frequency) and the phase of the pressure field, respectively.

For simplicity, we consider a one-dimensional element with two nodes, located at positions x_i and x_{i+1}. Linear basis functions are employed, expressed as follows:

$$w_i(x) = \begin{cases} \frac{x - x_i}{h} + 1 & \text{if } x \in [x_i - h, x_i], \\ \frac{x_i - x}{h} + 1 & \text{if } x \in [x_i, x_i + h], \\ 0 & \text{otherwise.} \end{cases} \tag{41}$$

where $h = x_{i+1} - x_i$ denotes element size. Substituting Eq. (41) into Eq. (40), and noting that

$$\int_{x_i}^{x_{i+1}} w_i(z)\, dz = h/2 \tag{42}$$

We can get the volume-averaged pressure field $\bar{p}(x)$,

$$\bar{p}(x) = \frac{2}{h}\left(\frac{x_{i+1} - x}{h}\right) \int_{x_i}^{x_{i+1}} \left(\frac{x_{i+1} - x}{h}\right) \cos(kx + b)\, dx + \frac{2}{h}\left(\frac{x - x_i}{h}\right) \int_{x_i}^{x_{i+1}} \left(\frac{x - x_i}{h}\right) \cos(kx + b)\, dx \tag{43}$$

Furthermore, the integral result can be written as:

$$\bar{p}(x) = \frac{x_i + x_{i+1} - 2x}{2h^3k^2}(\cos\alpha - \cos\beta) + \frac{1}{2h^2k}\left[-(x_{i+1} - x)\sin\alpha + (x - x_i)\sin\beta\right] \tag{44}$$

where $\alpha = kx_i + b$, $\beta = kx_{i+1} + b$.

For simplicity, we assume $b = 0$, $x_i = 0$ and $x_{i+1} = h$, $x = (x_i + x_{i+1})/2$, the Eq. (44) will degenerate into :

$$\bar{p}\left(\frac{h}{2}\right) = \frac{\sin(kh)}{hk} \tag{45}$$

The original pressure function $\cos(kx + b)$ takes the value $\cos(kh/2)$ at the same position. When k is small (indicating low spatial frequency), we have $\sin(kh) \approx kh$, resulting in $\bar{p} \approx \frac{kh}{hk} = 1$. In this case, the original value $\cos(kh/2)$ is approximately equal to 1, indicating minimal attenuation at low frequencies. Conversely, when k is large (indicating high spatial frequency), we find that $\bar{p} \sim \frac{1}{k}$, thus the fluctuating pressure approaches 0 as k increases toward infinity, signifying significant attenuation at high frequencies. Therefore, by volume-averaging the pressure field, high-frequency pressure oscillations are filtered out within the interval $[x_i, x_{i+1}]$, resulting in a smooth pressure field that functions as a low-pass filter for noise reduction. The actual pressure can be considered as the sum of low-frequency physical pressure and high-frequency fluctuating pressure. Due to the low-pass filtering characteristics of pressure projection, pressure oscillations are effectively removed, allowing the stable physical pressure component to be retained. The volume-averaged pressure is subsequently used to solve the equilibrium equations at the nodes, thereby breaking the self-cycle of pressure oscillations.

To sum up, the first volume averaging aims to reduce excessive volumetric constraints and is utilized to compute the volume-averaged deformation gradient, which is subsequently input into the constitutive model for stress calculations. In contrast, the second volume averaging acts as a low-pass filtering and is applied to obtain the volume-averaged pressure , which is used to solve the equilibrium equations at the nodes. These two processes operate in sequence and collaborate to mitigate the issue of volumetric locking in MPM.

Remark 1. In FEM community, the $\boldsymbol{u}/p$ mixed element has been successfully applied in the solution of the Stokes problems, where stability is governed by the Ladyzenskaya-Babuska-Brezzi (LBB) condition [57]. Our approach targets pressure-velocity coupling directly and the stabilization method used in this study is inherited from the $\bar{F}$ method in FEM. As discussed in previous study, the $\bar{F}$ method is not to satisfy the well-known LBB stability condition, but the degree of dissatisfaction is 'slight', so it is not altogether mathematically abhorrent [83].

3.3. Correction and smoothing

3.3.1. δ-correction

It is acknowledged that particle clustering in MPM will degrade numerical integration with substantial quadrature errors, see e.g., [56]. Such clustering errors are directly related to the particle length scale, disrupting the conservation of mass and momentum [29]. The quadrature error can be assessed through the integrals of the basis functions and numerical integration over the material points. When material point clusters occur, the following condition must be satisfied:

$$\sum_p V_p w_{ip} > \int w_i(\boldsymbol{x}) \mathrm{d}\Omega \tag{46}$$

Where V_p is the volume of the material point, w_{ip} is the basis function at the location of the material point, $w_i(\boldsymbol{x})$ is the spatial basis function. Eq. (46) indicates an over-estimation in the numerical integration, leading to integration errors. A set of nodal errors, denoted as E_i, can be defined to quantify this over-estimation error.

$$E_i = \max\left(0, -V_i + \sum_p V_p w_{ip}\right), \quad \text{where} \quad V_i = \int w_i(\boldsymbol{x})\mathrm{d}\Omega \tag{47}$$

Note that E_i quantifies the excess volume at node i resulting from the mapping of material points using the basis functions. When material points near node i cluster together, the nodal numerical integrals are prone to quadrature errors. If E_i exceeds a threshold E_c, then the δ-correction is applied.

The global volume error for all active nodes can be measured by

$$\|E\|^2 = \sum_i E_i^2 \tag{48}$$

and its gradient with respect to each material point is given by

$$\left(\nabla\|E\|^2\right)_p = 2V_p \sum_i E_i \nabla w_{ip} \tag{49}$$

A shifting vector $\delta\boldsymbol{x}_p^{n+1}$ is then introduced by

$$\delta\boldsymbol{x}_p^{n+1} = -b_0\left(\nabla\|E\|^2\right)_p, \tag{50}$$

with

$$b_0 = \frac{\|E\|^2}{\sum_{p=1}^{N_p}\left(\left(\nabla\|E\|^2\right)_p \cdot \left(\nabla\|E\|^2\right)_p\right)} \tag{51}$$

The position $\boldsymbol{x}_p^{n+1}$ of the material can be updated as

$$\boldsymbol{x}_p'^{n+1} = \boldsymbol{x}_p^{n+1} + \delta\boldsymbol{x}_p^{n+1} \tag{52}$$

where $\boldsymbol{x}_p'^{n+1}$ is the updated position. The Newton method can be used to find the optimized shifting vectors that minimizes $\|E\|^2$. With the updated position $\boldsymbol{x}_p^{n+1}$, we can recompute the basis function w_{ip} and its gradient ∇w_{ip}. The nodal error $\|E\|^2$ can then be re-evaluated. If it still exceeds the threshold, the procedure outlined in Eq. (49)–(52) will be repeated until $\|E\|^2$ satisfies the required criteria.

3.3.2. Pressure smoothing

A straightforward pressure smoothing approach, based on the weighted average method, was incorporated into the MPM formulation for visualization [84]. Initially, the pressure values from material points are projected into the neighboring nodes, as described by Eq. (53). At the nodes, the average pressure is calculated by weighting material points by their mass. Subsequently, nodal pressure will be interpolated back to material points using nodal basis functions, as outlined in Eq. (54).

$$\hat{p}_i = \frac{\sum_p w_i(\boldsymbol{x}_p) m_p p_p}{m_i} \tag{53}$$

$$\hat{p}_p = \sum_i w_i(\boldsymbol{x}_p)\hat{p}_i \tag{54}$$

where $\hat{p}_p$ denotes the smooth particle pressure and $\hat{p}_i$ represents the nodal pressure, which is interpolated from the particle pressure p_p by using the interpolation basis function $w_i(\boldsymbol{x}_p)$. Note that the pressure smoothing based on the weighted average method is employed only for visualization purpose and the smoothed pressure is not involved in next iteration.

3.4. Specific boundary handling

3.4.1. Free surface tracking

Neglecting surface tension, the pressure at the free surface of a single fluid flow is assumed to be equal to the atmospheric pressure [84]. Accordingly, the zero Dirichlet pressure boundary condition at the free surface for a single fluid flow problem is expressed as:

$$p = 0 \quad \text{on} \quad \Gamma_D \tag{55}$$

The free surface tracking method proposed by [84] is employed. As shown in Fig. 4 the volume fraction F of the material point in each cell is computed using Eq. (56).

$$F = \frac{\sum_{p=1}^{N_p^e} V_p}{V_e} \tag{56}$$

where V_p is the volume of the material point, V_e is the cell volume and N_p^e is the number of particle within the cell. Nodes associated with at least one cell having a volume fraction below a specified threshold, F_c, are identified as free surface nodes. At these nodes, the Dirichlet pressure boundary condition is applied.

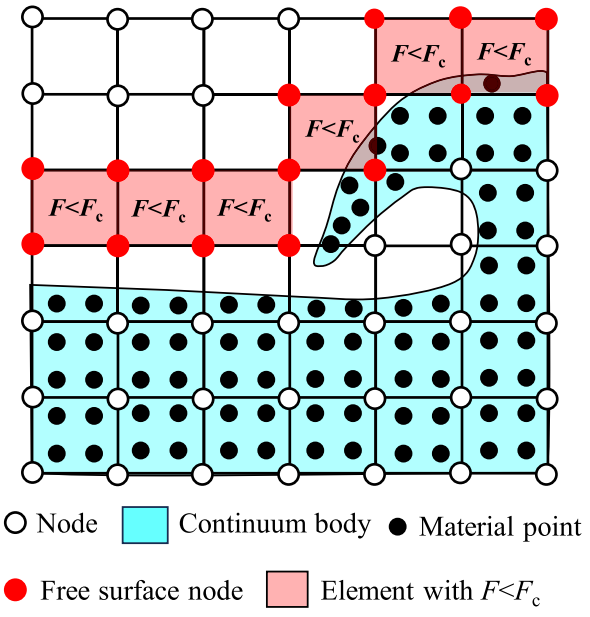

Fig. 4. Free surface capturing in MPM by using volume fractions.

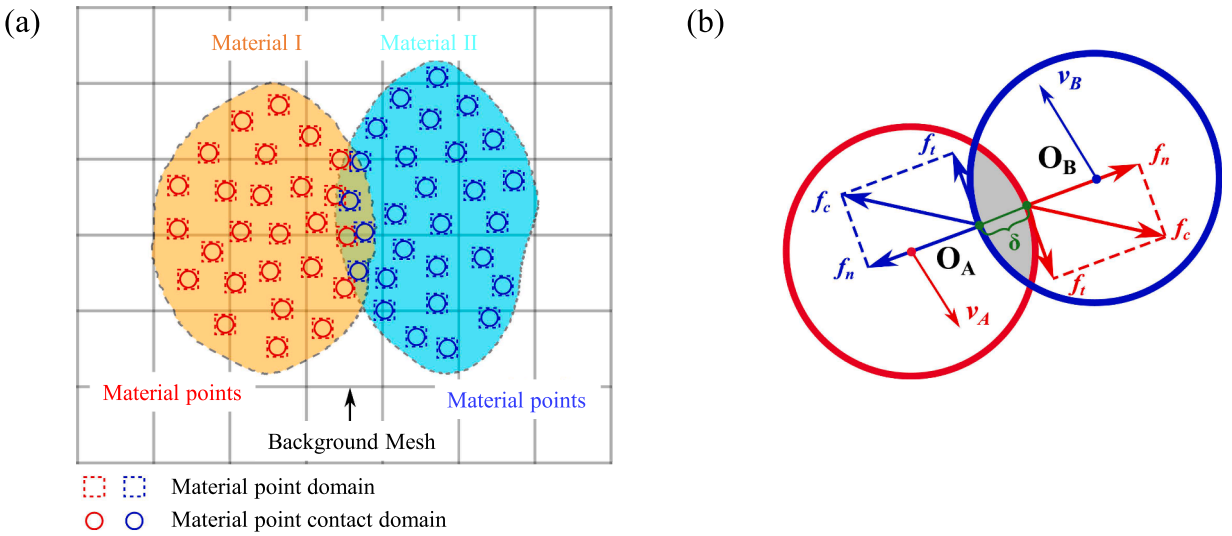

Fig. 5. DEM-enriched contact model [85]:(a) schematic representation of the material point pair domain; (b) illustration of the contact law implementation between two material points, each modeled as a disk.

3.4.2. DEM-enriched contact

The traditional MPM contact algorithm facilitates material collisions by mapping data from material points to the background grid. However, advanced interpolation techniques, such as B-spline and GIMP, may lead to premature contact between material points, even if their positions are separated by an additional cell, as long as their shape functions overlap at the same node. To address this issue, Chen et al. [85] introduced a DEM-enriched contact approach within the MP-DEM framework. This method combines the strengths of both MPM and DEM, providing a unified and flexible approach to boundary-particle and particle-particle interactions in multi-material systems. Traditional DEM simulations are often time-consuming due to the need for neighbor searching among particles. However, by coupling DEM with the MPM, we can leverage the background mesh to streamline this process. Material points that contribute to the same node are considered potential contact pairs, and the penetration depth is calculated based on their relative positions.

Fig. 5 (a) illustrates the contact configuration between two materials. The contact law is applied to material points, which are assumed to be disk-shaped, to simplify contact detection and improve computational efficiency [85]. As shown in Fig. 5 (b), the

resulting contact force acting on the material point is determined by both the penetration depth δ and the relative velocity between points O_A and O_B, as outlined in the following force-displacement relationship:

$$\boldsymbol{f}_n^{n+1} = k_n \delta \cdot \boldsymbol{n} \tag{57a}$$

$$\boldsymbol{f}_t^{n+1} = \min(\boldsymbol{f}_t^n + \Delta \boldsymbol{f}_t, \mu \| \boldsymbol{f}_n^{n+1} \|) \tag{57b}$$

$$\Delta \boldsymbol{f}_t = k_t \Delta \boldsymbol{u} \tag{57c}$$

$$\Delta \boldsymbol{u} = (\boldsymbol{v}_A - \boldsymbol{v}_B)\Delta t - [(\boldsymbol{v}_A - \boldsymbol{v}_B) \cdot \boldsymbol{n}]\boldsymbol{n}\Delta t \tag{57d}$$

where superscript n or $n+1$ denotes physical quantity at the time step of n or $n+1$. $\boldsymbol{f}_n$ and $\boldsymbol{f}_t$ denotes the contact force in normal and tangential direction, respectively; δ denotes the penetration depth; $\boldsymbol{n}$ denotes the contact normal vector; $\Delta \boldsymbol{f}_t$ is the increment of tangential force; $\Delta \boldsymbol{u}$ is the relative displacement in tangential direction; $\boldsymbol{v}_A$ and $\boldsymbol{v}_B$ denotes the velocity at point O_A and O_B, respectively. Δt denotes time step increment; k_n and k_t are normal and tangential contact stiffness, respectively; μ is the friction coefficient.

After calculating the contact forces at the material points, we can incorporate these forces as body forces in the governing equation (i.e., Eq. (1)). This allows them to participate in the MPM calculations in the same manner as other body forces.

Explicit MPM formulations are preferred over implicit schemes due to their simplicity in implementation and high-performance parallelism on GPUs [86]. In contrast, the convergence of implicit MPM depends on the nonlinear behavior of local contact or material responses. As a result, careful management of displacement increments is essential to ensure convergence, but this can compromise the robustness of implicit solvers [87]. While recent studies have employed implicit MPM formulations coupled with FEM for contact scenarios, these methods are limited to small and finite deformations [87–90]. The applicability of these approaches to large deformation (e.g., in FSI) problems remains limited. In this study, we introduce a DEM-enriched contact approach within the proposed explicit stabilized MPM framework, specifically designed to simulate FSI problems and effectively mitigate pressure oscillations.

4. Numerical examples

4.1. 1D elastic wave propagation

The 1D elastic wave propagation is used to validate the accuracy of pressure calculations [44]. Fig. 6 shows the geometry and boundary conditions for this example. A flap divides the column of $L = 1$ m into two halves: the left half is assigned a uniform pressure field of $P_1 = 1000$ Pa, while the right half is assigned a pressure of $P_2 = 100$ Pa. The top and bottom boundaries are fixed in the y-direction, while the two sides are fixed in the x-direction. The water has a density of $\rho = 1000\,\mathrm{kg/m^3}$, a dynamic viscosity of $\mu = 0.001$ Pa · s, and a bulk modulus of $K = 2.5$ MPa. Using the formula $c = \sqrt{K/\rho}$, the sound speed c is calculated to be 50 m/s. Once the flap is removed, the elastic wave will propagate through the water column. For $t < 10$ ms, the analytical solution is given as

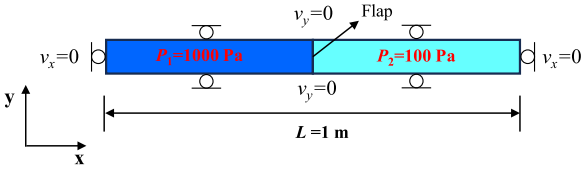

Fig. 6. Elastic wave propagation: the geometry and boundary conditions.

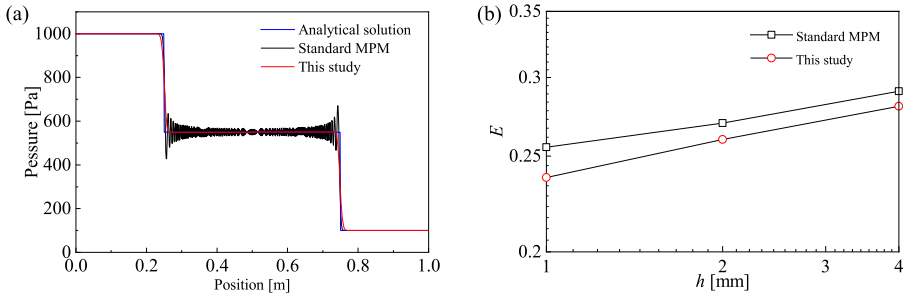

Fig. 7. Elastic wave propagation: (a) pressure distribution for an element size of 1 mm at $t = 5$ ms; (b) logarithmic relationship between normalized pressure errors and element sizes.

follows [44]:

$$\widetilde{p}(x,t)=\begin{cases}1000 & \text{if } x<0.5-ct,\\ 100 & \text{if } x>0.5+ct,\\ 550 & \text{otherwise.}\end{cases} \tag{58}$$

The grid used for the simulation is a square mesh with an element size of $h=1$ mm, containing one particle per cell, resulting in a total of 1000 material points. The time step is set to $\Delta t=1\times10^{-5}$ s. The pressure distribution along the x-axis for the standard MPM and the enhanced method proposed in this study, which uses dual volume-averaged smoothing, are shown in Fig. 7 (a). To better investigate the influence of the smoothing technique on pressure calculations, linear basis functions are employed for the numerical simulations. Compared with the analytical solution, the numerical results enhanced by the proposed method exhibit smoother pressure distributions, while the standard MPM introduces significant pressure oscillations in the transition region. To quantify the error, the normalized pressure error norm is defined as follows:

$$E=\frac{\sum_p|\widetilde{p}(x_p)-p(x_p)|V_p}{\int_\Omega\widetilde{p}(x)\mathrm{d}\Omega} \tag{59}$$

where $\widetilde{p}(x_p)$ is the analytical value at $x=x_p$, $p(x_p)$ is the numerical value, V_p is the particle's volume.

Fig. 7 (b) presents the logarithmic relationship between normalized pressure errors E and different element sizes of $h=1,2,4$ mm. As the element size decreases, the errors for both the standard MPM and the proposed method decrease almost linearly, indicating mesh convergence for both approaches. Additionally, pressure oscillations caused by volumetric locking lead to larger errors in the standard MPM, while the proposed smoothing method reduces these errors and improves pressure accuracy. Thus, the dual volume-averaging method not only predicts a much smoother pressure field but also effectively reduces errors, maintaining higher accuracy.

4.2. Poiseuille flow

Poiseuille flow is a well-established benchmark used to validate the accuracy of CFD models [91], and the corresponding setup is illustrated in Fig. 8. In this scenario, a fluid initially at rest between two fixed plates begins to flow under the influence of a body force F, which is identical with the constant pressure gradient [91,92], eventually reaching a steady state.

This example examines the dynamic behavior of a weakly compressible fluid, characterized by a density of $\rho=1000\,\mathrm{kg/m^3}$ and a dynamic viscosity of $\mu=0.001$ Pa · s. The initial geometry, shown in Fig. 8 (a), consists of a two-dimensional square computational domain with $L=1$ mm and $H=1$ mm. The left and right boundaries are fixed, with velocity conditions of $v_x=0$ m/s and $v_y=0$ m/s, while periodic boundary conditions are applied to the top and bottom surfaces. A uniform body force $F=8\,\mathrm{mm/s^2}$ is imposed over the entire domain. The grid used for the simulation is a square mesh with an element size of $h=0.05$ mm, containing 4 particles per cell, resulting in a total of 1600 material points. The initial particle configuration is depicted in Fig. 8 (b). The time step is set to $\Delta t=2\times10^{-6}$ s. To capture the fluid's compressibility, a bulk modulus $K=0.1$ MPa is selected, which is significantly higher than the expected maximum pressure, $p_{\max}\approx2$ Pa. This choice ensures that the fluid's volumetric strain remains minimal, and density variations have a negligible effect on the overall solution.

The analytical solution for the transient behavior of the flow is given by [91]:

$$v(y,t)=\frac{F}{2\mu_0}y(y-H)+\sum_{n=0}^{\infty}\frac{4FH^2}{\mu_0\pi^3(2n+1)^3}\times\sin\left[\frac{\pi y}{H}(2n+1)\right]\exp\left[-\frac{(2n+1)^2\pi^2\mu_0}{H^2}t\right] \tag{60}$$

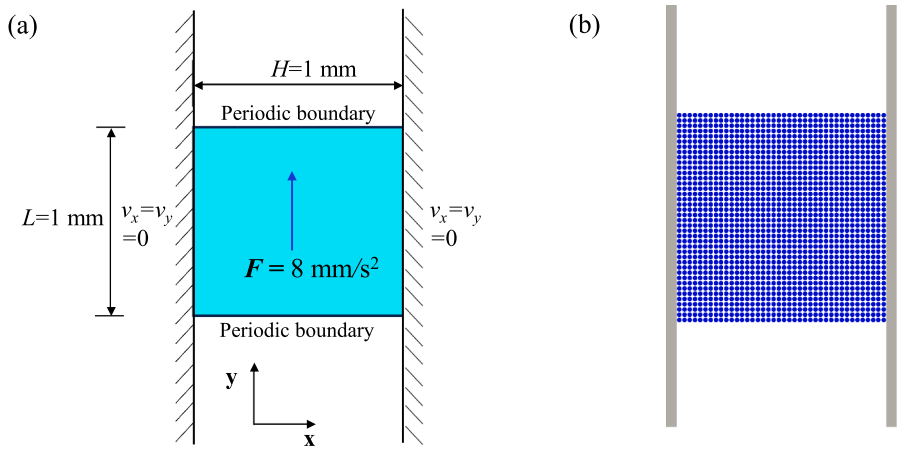

Fig. 8. Poiseuille flow: (a) initial configuration and boundary conditions; (b) initial configuration of material points.

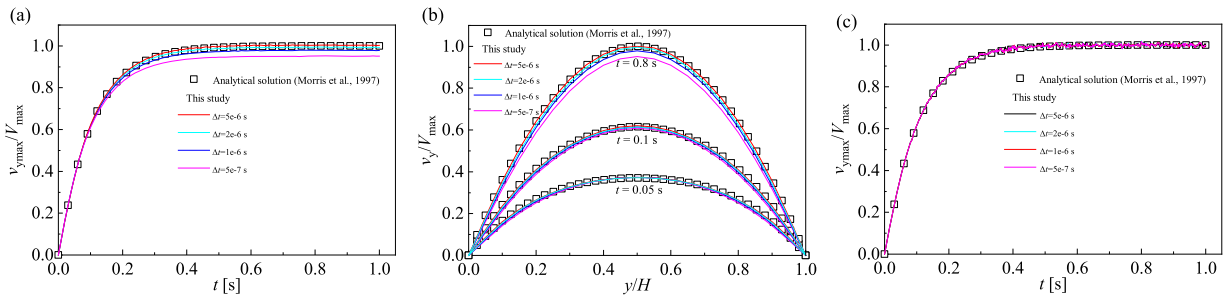

Fig. 9. Poiseuille flow: (a) variation of maximum velocity with time; (b) velocity profile at different time instants; (c) variation of maximum velocity with time without the proposed stabilization method.

where H is the distance between the plates, and $\mu_0 = 10^{-6}\,\mathrm{m}^2 \cdot \mathrm{s}^{-1}$ is the fluid's viscosity. The maximum velocity at steady state, $V_{\max}$, is given by $\frac{F}{8H^2\mu_0}$, which serves to normalize the velocity field.

Fig. 9 (a) illustrates the variation of the maximum velocity with time, showing an initial increase followed by stabilization after $t = 0.5$ s. The numerical results are in excellent agreement with the analytical solution. In Fig. 9 (b), the velocity profile along the horizontal section is extracted. The expected parabolic shape, as predicted by the analytical solution, is successfully reproduced in the numerical simulation, thereby demonstrating the accuracy and robustness of the proposed stabilization method.

Fig. 9 (a) and (b) also present transient velocity curves and velocity profiles ($t = 0.8$ s), respectively, for other different time steps, i.e. $\Delta t = 5 \times 10^{-7}, 1 \times 10^{-6}, 5 \times 10^{-6}$ s. The results indicate that a smaller time step leads to underestimation compared to the analytical solution. We also present results without stabilization techniques in Fig. 9 (c) for comparison. These results demonstrate that while varying Δt (within a small range) does not significantly affect the overall accuracy of the numerical solution, some fluctuations are still observable. The proposed stabilized method acts as a smoothing process to mitigate these velocity fluctuations, though it inherently introduces numerical dissipation and errors at each time step. At the initial stage of the simulation, for instance at $t = 0.05$ s, the discrepancy between the analytical and numerical results with varying time steps Δt is minimal. However, as time progresses, this error accumulates, eventually leading to an underestimation of the system's behavior at the steady-state. A smaller time step Δt leads to more total steps and greater numerical error accumulation, ultimately reducing the quality of the solution.

However, as shown in Fig. 9 (a)-(b), increasing Δt helps to reduce the error, and the numerical results progressively converge toward the exact solution. This phenomenon reflects excessive energy dissipation can be controlled by enlarging time steps. As Δt increases, the dissipation decreases and becomes negligible beyond $\Delta t = 2 \times 10^{-6}$ s. Therefore, the dissipation is controllable, and selecting a relatively larger time step can maintain acceptable error levels.

4.3. Lid driven cavity

The lid-driven cavity flow is a widely recognized benchmark case for verifying numerical methods in CFD. It is particularly useful for evaluating the accuracy of schemes applied to fluid dynamics problems in confined spaces, especially those with non-uniform velocity boundary conditions. However, this problem presents significant challenges for traditional MPM implementations. The issue

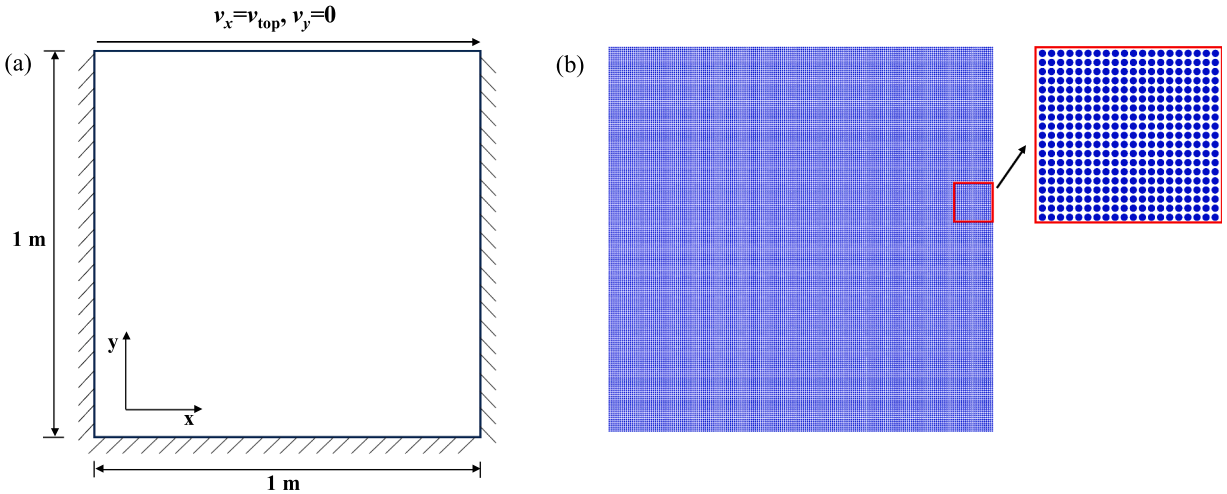

Fig. 10. Cavity: (a) initial configuration and boundary conditions; (b) initial configuration of material points.

arises because material points are driven by the nodal velocity field, which can lead to numerical instability, particularly at the corners where corner eddies often form [93]. To address this issue, Kularathna and Soga [93] opted not to update particle positions during the simulation, thereby avoiding instability. Despite this approach, they found that simulations could not be conducted with Reynolds numbers exceeding 10. More recently, Baumgarten and Kamrin [56] demonstrated that the application of δ-correction can help stabilize the simulation and yield better results compared to other particle-shifting methods, such as avoid-a-void MPM [94] and SPH-like correction in MPM [56,95]. While Baumgarten and Kamrin [56] employed a cell-based anti-locking technique for strain smoothing, their results for the velocity field remain insufficiently smooth and require further improvement.

The present study investigates the dynamic behavior of a weakly compressible fluid with a density of $\rho = 1000\,\mathrm{kg/m^3}$. The initial geometry is shown in Fig. 10 (a), where the computational domain is a two-dimensional square with a side length of $L = 1\,\mathrm{m}$. The top boundary is defined by the condition $v_y = 0\,\mathrm{m/s}$, while the remaining three boundaries are treated as fixed walls, imposing the conditions $v_x = v_y = 0\,\mathrm{m/s}$. The scenario under consideration involves setting the velocity of the top boundary, v_{top}, to 1 m/s, which corresponds to a Reynolds number of $Re = \rho v_{top} L/\mu = 1000$. The dynamic viscosity is chosen to be $\mu = 1\,\mathrm{Pa{\cdot}s}$, the same value used by Baumgarten and Kamrin [56].

A square grid with an element size of $h = 0.01\,\mathrm{m}$ and 4 particles per cell is used, yielding 40,000 material points in total. The initial particle configuration is shown in Fig. 10 (b). The time step Δt is set to be 0.0001 s, and a quasi-steady state is reached at $t = 40$ s. To model the fluid's compressibility, a bulk modulus of $K = 1.5\,\mathrm{MPa}$ is selected, which is considerably larger than the expected maximum pressure, $p_{\max} \approx 500$ Pa [56]. This choice ensures that the fluid's volumetric strain remains small, and any density variations are negligible, thus having minimal impact on the overall solution.

Fig. 11 shows the magnitude of the velocity field, $||\vec{\boldsymbol{v}}||$, for the lid-driven cavity flow at various simulation times. At the initial time, $t = 2.1$ s, a small vortex forms in the top-right corner due to the horizontal velocity imposed on the nodes of the top surface. As the simulation progresses, the influence of this vortex expands, eventually reaching a steady-state condition by $t = 40$ s. Fig. 12 (a)-(d) presents comparison of velocity contours at the steady state with findings from previous studies where GIMP basis functions were used in MPM simulations. Throughout the simulation, the particle velocity field demonstrates a relatively smoother solution relative to earlier methods, including the Avoid-a-void MPM [56,94], SPH-like correction in MPM [56,95], and δ-correction MPM [56]. Additionally, the implementation of δ-correction significantly reduces particle clustering, resulting in a more uniform particle

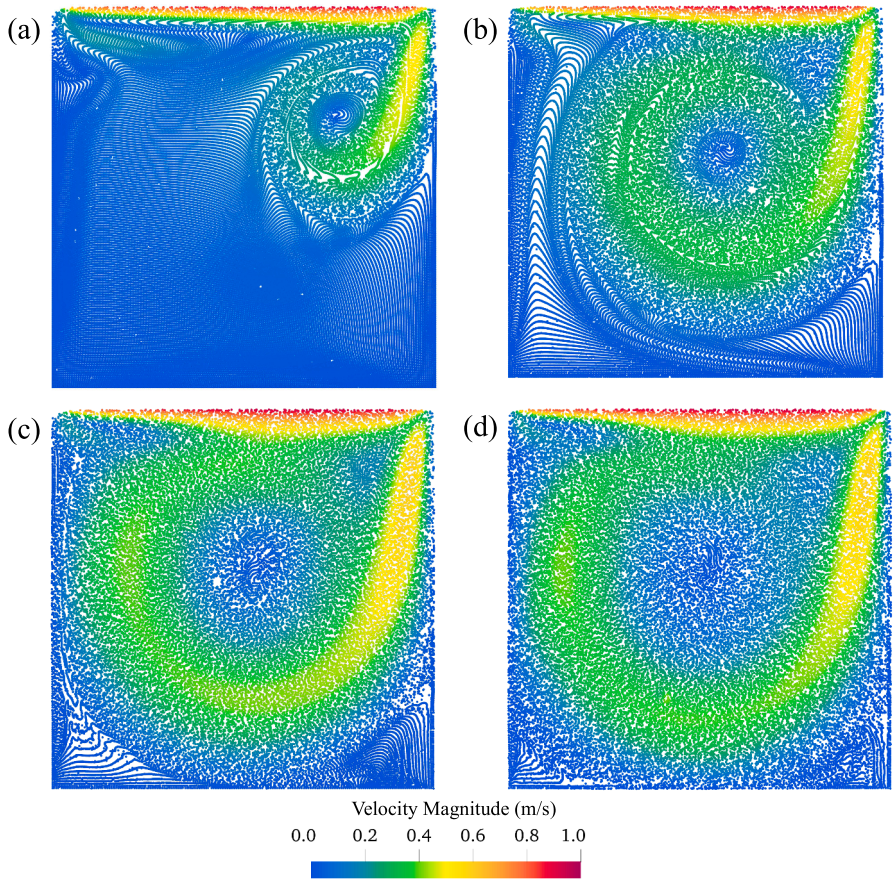

Fig. 11. Particle velocity magnitude at different moments: (a) $t = 2.1$ s; (b) $t = 10$ s; (c) $t = 20$ s; (d) $t = 40$ s.

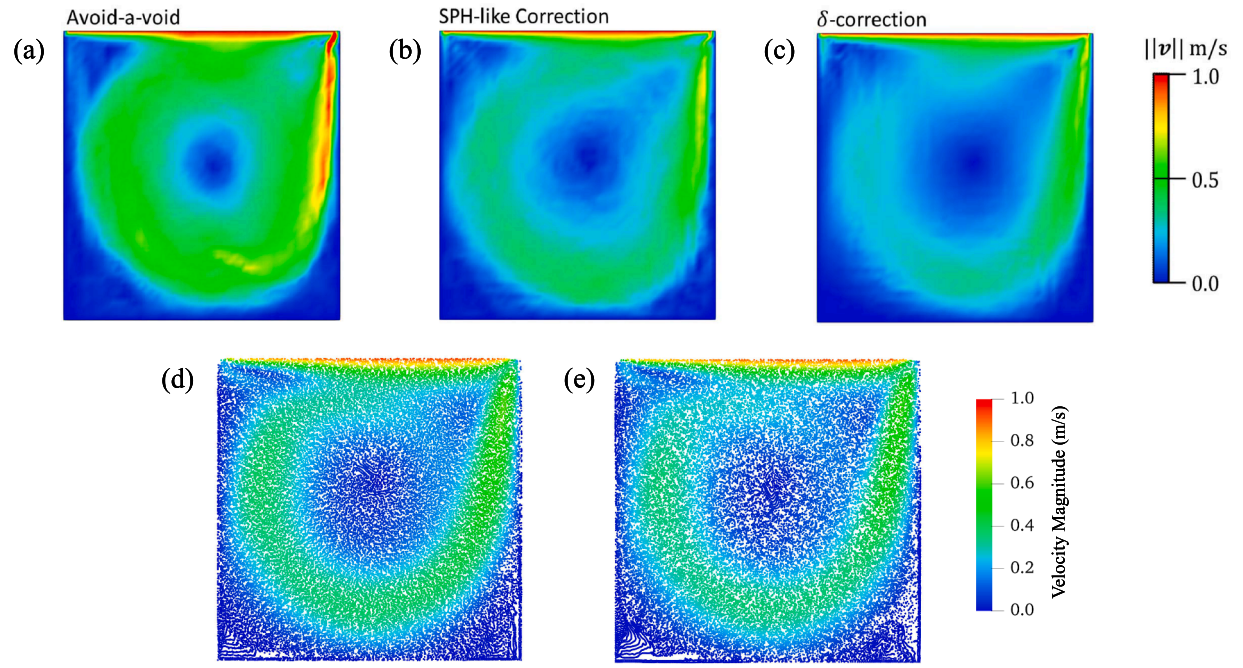

Fig. 12. Comparison of steady-state velocity contours with previous studies: (a) Avoid-a-void [56,94]; (b) SPH-like correction [56,95]; (c) δ-correction [56]; (d) this study; (e) this study without δ-correction.

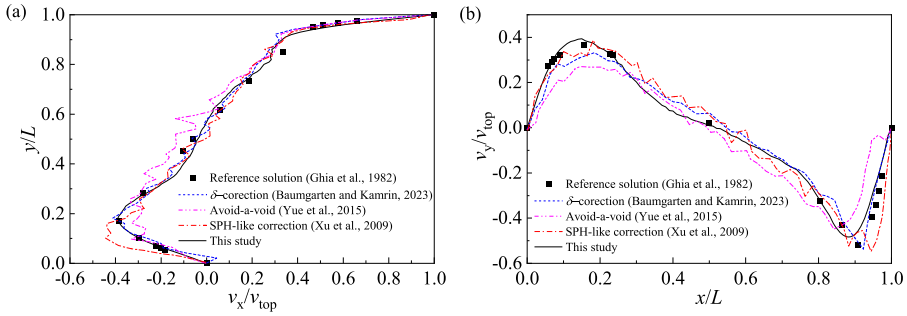

Fig. 13. Central velocity of cavity at $t = 40$ s: (a) horizontal velocity; (b) vertical velocity.

distribution at steady state. As shown in Fig. 12 (e), the absence of δ-correction leads to sparse particle distribution in certain areas near the boundary, which may adversely affect the accuracy of the simulation results [29]. This issue is mitigated following the application of δ-correction, as evidenced in Fig. 12 (d).

To further evaluate the accuracy of the simulation, nodal velocity profiles along the center of the cavity, specifically $v_x(0.5, y)$ and $v_y(x, 0.5)$, are extracted and presented in Fig. 13. It is important to note that the previous numerical results utilizing cubic B-spline basis functions are chosen for comparison. These profiles are also compared with the results of Ghia et al. [96], obtained using a multigrid method based on the FDM to solve the Navier-Stokes equations for incompressible flow. As shown in Fig. 13, both the Avoid-a-void method and SPH-like correction exhibit larger deviations from the reference solution. While the δ-correction provides better predictions compared to these two methods, some fluctuations are still evident in the velocity profiles. In contrast, the proposed method delivers smoother and more accurate results. Overall, the simulation results exhibit good agreement with the reference solution, thereby further validating the accuracy of the proposed method.

4.4. Water sloshing

Two cases of 2D water sloshing with varying excitation frequencies are examined to analyze the flow patterns and pressure distribution along the tank's walls. As shown in Fig. 14, the tank has dimensions of 1 m in height (H) and 1 m in width (B), with an initial water depth (H_w) of 0.3 m. The pitch motion is governed by $w = A\sin(\omega_p t)$, where the rolling amplitude, A, is set to 5° for both cases. The excitation period T_p and frequency ω_p are provided in Table 1. The physical properties of the water, including density ($\rho = 1000\,\mathrm{kg/m^3}$), bulk modulus ($K = 10\,\mathrm{MPa}$), dynamic viscosity ($\mu_w = 0.001$ Pa·s), and gravity ($g = 9.81\,\mathrm{m/s^2}$), are specified. The

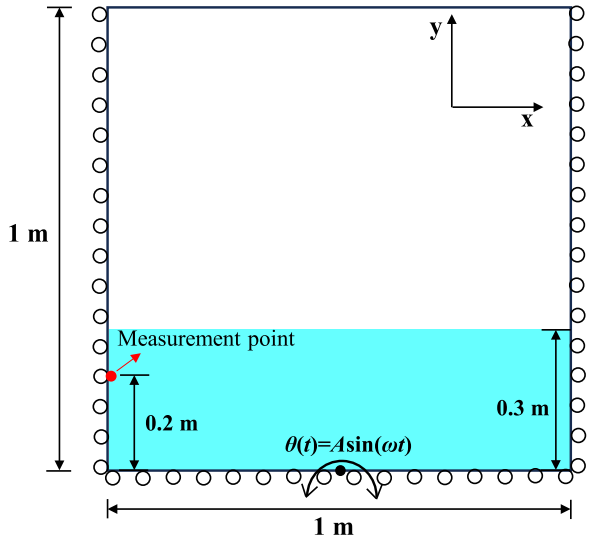

Fig. 14. Water sloshing: initial configuration and boundary conditions.

Table 1
Pitch motion parameters for two cases: slow and fast sloshing.

Case	ω_p(rad/s)	ω_0(rad/s)	ω_p/ω_0	T_p (s)
1	0.95	4.76	0.20	6.61
2	5.47	4.76	1.15	1.15

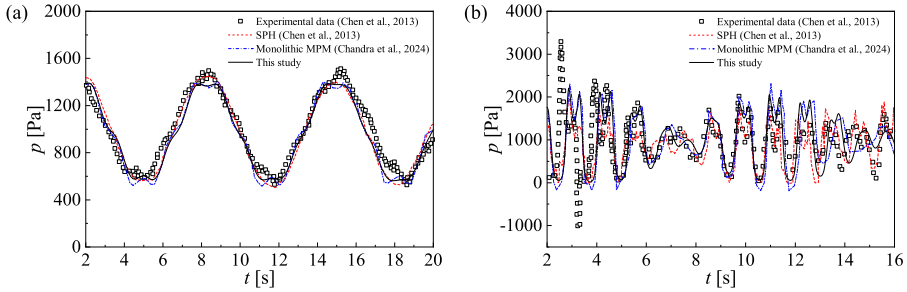

Fig. 15. Pressure variation with time at the measurement point: (a) slow sloshing; (b) fast sloshing.

water's natural angular frequency is $\omega_0 = 4.76$ rad/s, with Case 1 representing slow sloshing and Case 2 representing fast sloshing. A sensor is positioned at a height of $H_p = 0.2$ m on the left wall to measure real-time pressure variation, as depicted in Fig. 14.

The computational mesh is discretized into a square grid with an element size of $h = 0.02$ m, and each cell contains 4×4 material points, yielding 12,000 material points in total. The time step Δt is set to be $0.0001s$. To address the challenges associated with non-uniform pitching motion imposed by moving boundary conditions [97], a periodic body force field is applied to material points, while the boundary conditions on the surrounding walls remain unchanged. As shown in Fig. 14, the surrounding walls are modeled as slip-roller boundaries. The body force $\vec{b}$ is then adjusted to be a function of the pitching angle $\theta(t)$, i.e.,

$$b_x(t) = -g\sin(\theta t) \tag{61a}$$

$$b_y(t) = -g\sin(\theta t) \tag{61b}$$

For Case 1, the ratio of ω_p/ω_0 is set to 0.20, which theoretically results in a near-flat free surface during the sloshing motion. The pressure evolution recorded by the sensor at the measurement point in the experiment [98] follows a pattern closely resembling a harmonic sinusoidal function, as illustrated in Fig. 15 (a). The same figure also presents the numerical pressure variation for Case 1. It is evident that the numerical results obtained using the proposed approach closely match both the experimental data [98] and results from other computational methods [29,98]. Notably, both the experimental and numerical pressure curves exhibit two distinct peaks corresponding to the wave trough and crest. This behavior is also observed in simulations using the monolithic MPM method. However, it is not observed in the results from SPH, which are considered to exhibit higher numerical dissipation due to the strong artificial viscosity inherent in the method [99].

For Case 2, the ratio of ω_p/ω_0 is set to 1.15. As shown in Fig. 15 (b), violent liquid sloshing leads to large pressure fluctuations, and the experimental pressure evolution at the measurement point exhibits significant irregularities over time. The experimentally measured pressure shows a negative peak, primarily due to the high velocity of air at the interface, which results in suction on the tank wall [100]. However, this phenomenon cannot be simulated in the present study, as the model does not consider air. Notably, two distinct peaks appear at each wave crest, which can be attributed to the asynchronization between the water sloshing and the tank motion, causing a secondary impact on the tank wall. The results obtained using the SPH method, as shown in Fig. 15 (b), tend to underpredict the pressure during the first 6 seconds. This underestimation is likely due to the increased energy dissipation induced by greater numerical damping in SPH.

Both the monolithic MPM and the proposed approach produce smoother pressure-time histories compared to those obtained using the SPH method. However, phase discrepancies are observed at certain time intervals in the simulation, particularly between

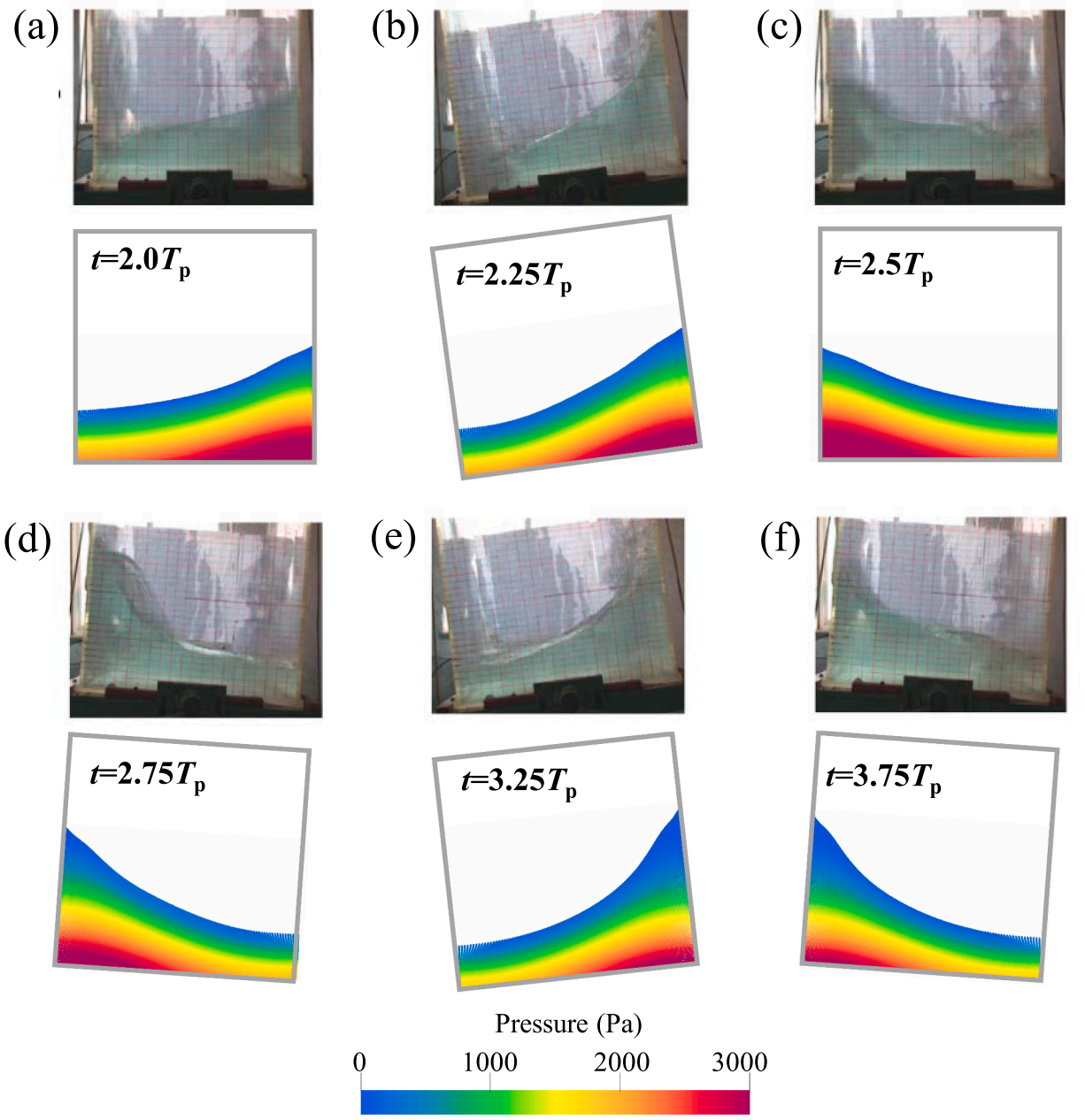

Fig. 16. Comparison between numerical and experimental results[98] at different time in Case 2: (a) $t = 2.0T_p$; (b) $t = 2.25T_p$; (c) $t = 2.5T_p$; (d) $t = 2.75T_p$; (e) $t = 3.25T_p$; (f) $t = 3.75T_p$.

6–8 s and 14–16 s in the monolithic MPM results. Apart from the pressure analysis at the measurement point, Fig. 16 presents a comparison of snapshots at various moments (t/T_p = 2.0, 2.25, 2.5, 2.75, 3.25, and 3.75) between the numerical results and the experimental configuration from [98] for Case 2. Overall, the numerical simulations demonstrate good agreement with the experimental observations, with the sloshing behavior at each time step closely matching the experimental findings.

4.5. Dam break

The proposed approach is used for simulation of a 2D dam break problem, a well-established benchmark for fluid dynamics that has been extensively studied in the context of the MPM [42,44–46,62,99]. The setup follows the configuration outlined in [42,101]. As illustrated in Fig. 17, a water column is initially defined with a height h_0 and length l_0, with a temporary gate to facilitate water consolidation. The surrounding boundaries are modeled as slip walls. Once the water has settled under the influence of gravity and reaches a state of equilibrium, the gate will be removed, allowing the water column to flow freely due to gravity. During the simulation, both the run-out length L and run-out height H are monitored and recorded throughout the process.

The simulation employs the following parameters: density $\rho = 1000\,\mathrm{kg/m^3}$, bulk modulus $K = 20\,\mathrm{MPa}$, dynamic viscosity $\mu = 0.001\,\mathrm{Pa\cdot s}$, and gravity acceleration $g = 9.81\,\mathrm{m/s^2}$. Two initial aspect ratios for the water column are considered: $h_0/l_0 = 1.0$ and $h_0/l_0 = 2.0$. For the aspect ratio of 1.0, the dimension is $h_0 = l_0 = 0.057\,\mathrm{m}$. For the aspect ratio of 2.0, the height is $h_0 = 0.114\,\mathrm{m}$, while the length remains $l_0 = 0.057\,\mathrm{m}$.

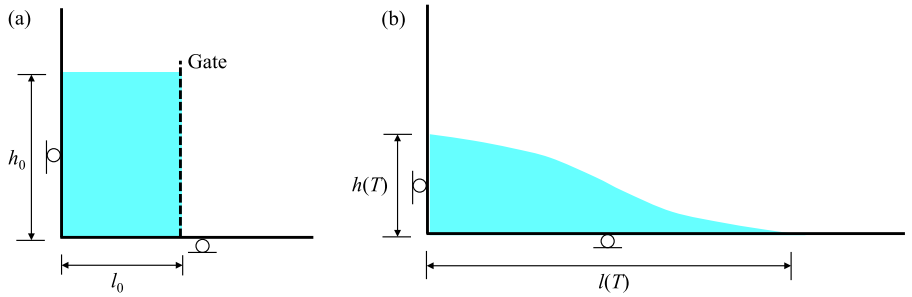

Fig. 17. Dam break simulation: (a) initial geometry; (b) geometry at normalized time T.

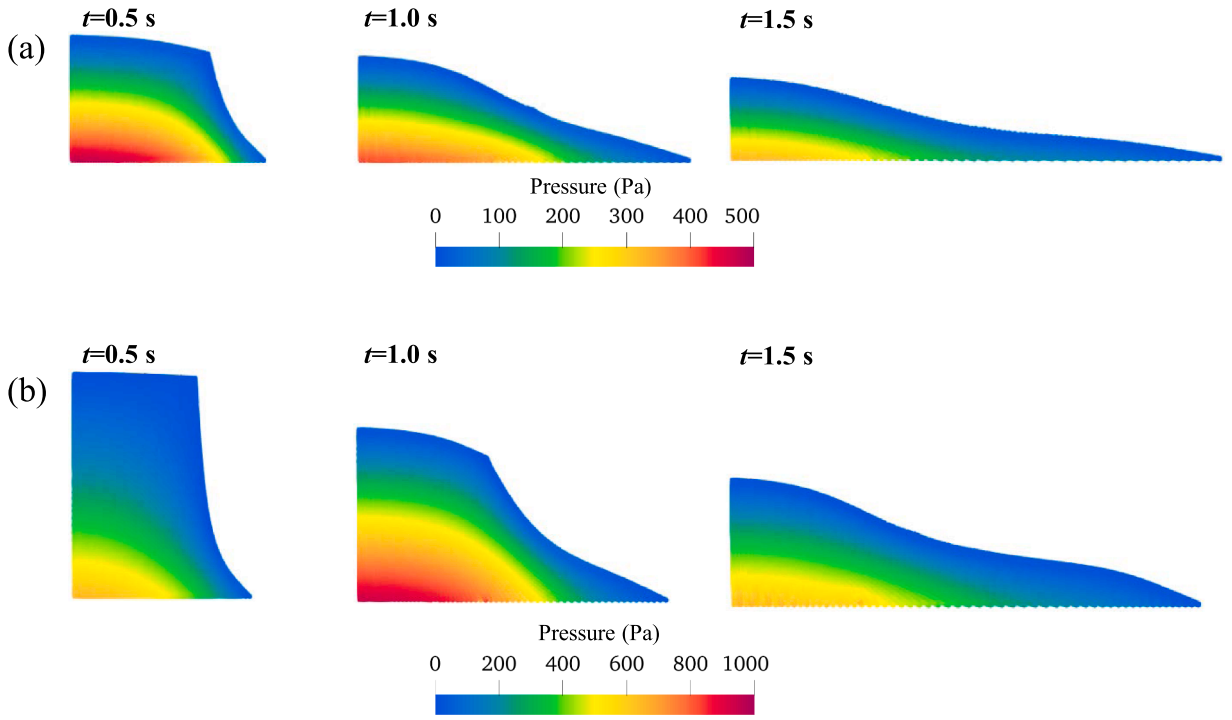

Fig. 18. Pressure contours throughout the computation: (a) aspect ratio of 1.0; (b) aspect ratio of 2.0.

The simulation is conducted by using a square grid with an element size of $h = 0.0019$ m. The fluid cells are initially arranged in a 2×2 particle configuration, resulting in a discrete fluid domain consisting of 3200 material points for an aspect ratio of 1.0 and 6400 material points for an aspect ratio of 2.0. The time step for the simulations is set to $\Delta t = 0.0001$ s.

Fig. 18 shows a series of snapshots illustrating simulation results at selected time intervals: $t = 0.5$, 1.0, and 1.5 s, for the two cases under consideration. As shown, the proposed methodology produces smooth pressure distribution contours and effectively mitigates issues related to volumetric locking, such as checkerboarding, thereby demonstrating its robustness in capturing fluid dynamics.

To further assess the accuracy and reliability of the fluid solver, several quantitative evaluations of simulation results are provided. Fig. 19 displays the temporal evolution of the water configuration for aspect ratios of 1.0 and 2.0, with comparisons made to experimental data [101], the standard MPM and the displacement-based MPM [62]. Quadratic B-spline basis functions were utilized in the displacement-based MPM, while GIMP basis functions were employed in the standard MPM. The time step was set to $\Delta t = 2.5 \times 10^{-7}$ s for both numerical simulations.

The following normalized parameters are used to characterize the variation of the free surface during the run-out process:

$$T = t\sqrt{\frac{h_0 g}{l_0^2}}, L = \frac{l}{l_0}, H = \frac{h}{h_0} \tag{62}$$

The simulation results demonstrate generally good agreement with experimental findings. However, a slight discrepancy is observed, with the predicted flow exhibiting marginally faster motion than the experimental observations. This deviation can be attributed to two key simplifications in the modeling process. First, the simulation does not fully capture the detailed geometry and motion of the gate. Previous studies [102] have indicated that the geometry of the gate should be considered in the modeling process, as it can influence flow dynamics by generating irregular flow structures, wave patterns, and initial vorticity. Second, the bottom boundary is modeled as frictionless, whereas, in reality, the wall possesses some roughness. This difference may lead to the onset of turbulence at the fluid interface, as well as boundary layer effects that can influence the horizontal velocity [29].

We further examine the effect of the blending coefficient β on simulation results and energy conservation. For comparison, we employ a dam break model with an aspect ratio of 1 and a coarser element size of $h = 0.0038$ m. The only modification is the blending coefficient β, with all other model parameters held constant. As shown in Fig. 20 (a), the pressure contours at $t = 1.0$ s for $\beta = 0.0$ (representing the FLIP scheme) exhibit significant pressure oscillations, compared to those for $\beta = 0.05$ and $\beta = 1.0$. However, the water surface geometry for $\beta = 1.0$ (representing the APIC scheme) shows noticeable irregularities. Furthermore, the energy dynamics of the dam break process are shown in Fig. 20 (b). While the potential energy remains similar across all three values of β, the kinetic energy for $\beta = 1.0$ is lower than that for the other two cases. As a result, the total energy for $\beta = 1.0$ exhibits a decreasing trend, whereas the total energy for $\beta = 0.0$ and $\beta = 0.05$ remains nearly constant. This indicates that the APIC scheme results in significant energy dissipation at low grid resolutions [53]. In contrast, the results for $\beta = 0.05$ demonstrate strong performance in both pressure stability and energy conservation, indicating an improvement over the original APIC scheme at low grid resolutions.

Previous studies have shown that quadratic B-spline shape functions exhibit poorer accuracy and convergence compared to cubic B-spline functions in large deformation scenarios [70]. This subsection presents a comparative analysis to investigate the necessity of

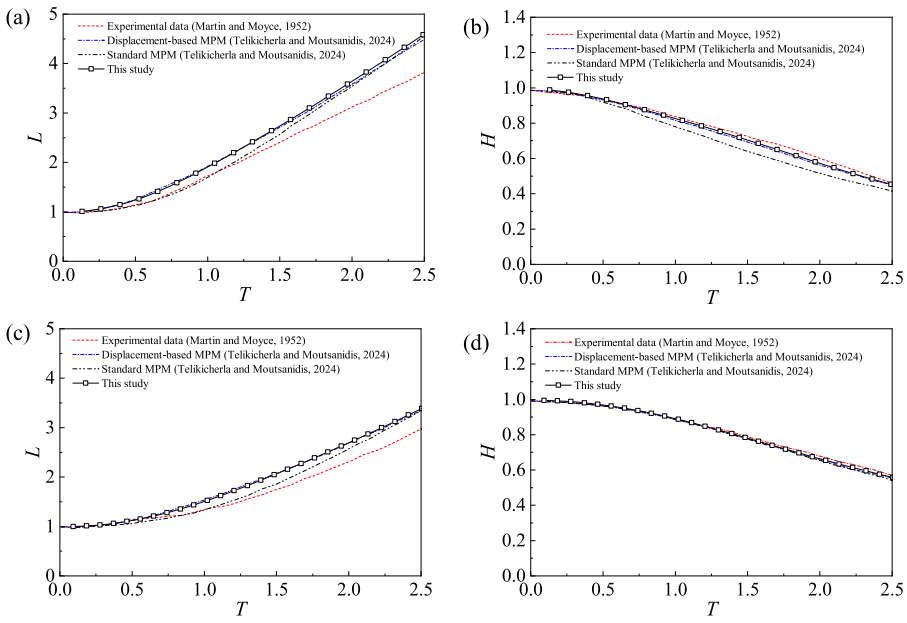

Fig. 19. Comparison with previous work: (a) normalized run out length of aspect ratio of 1.0; (b) normalized run out height of aspect ratio of 1.0; (c) normalized run out length of aspect ratio of 2.0; (d) normalized run out height of aspect ratio of 2.0.

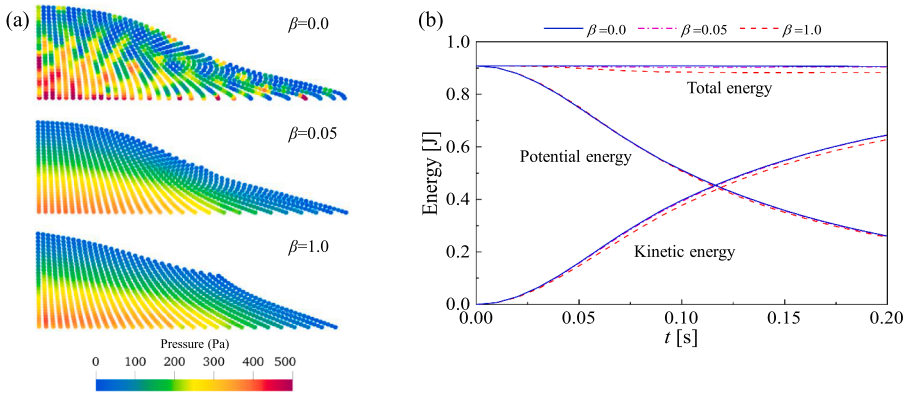

Fig. 20. Comparison of (a) configuration at $t = 1.0$ s and (b) energy under different blending coefficient β.

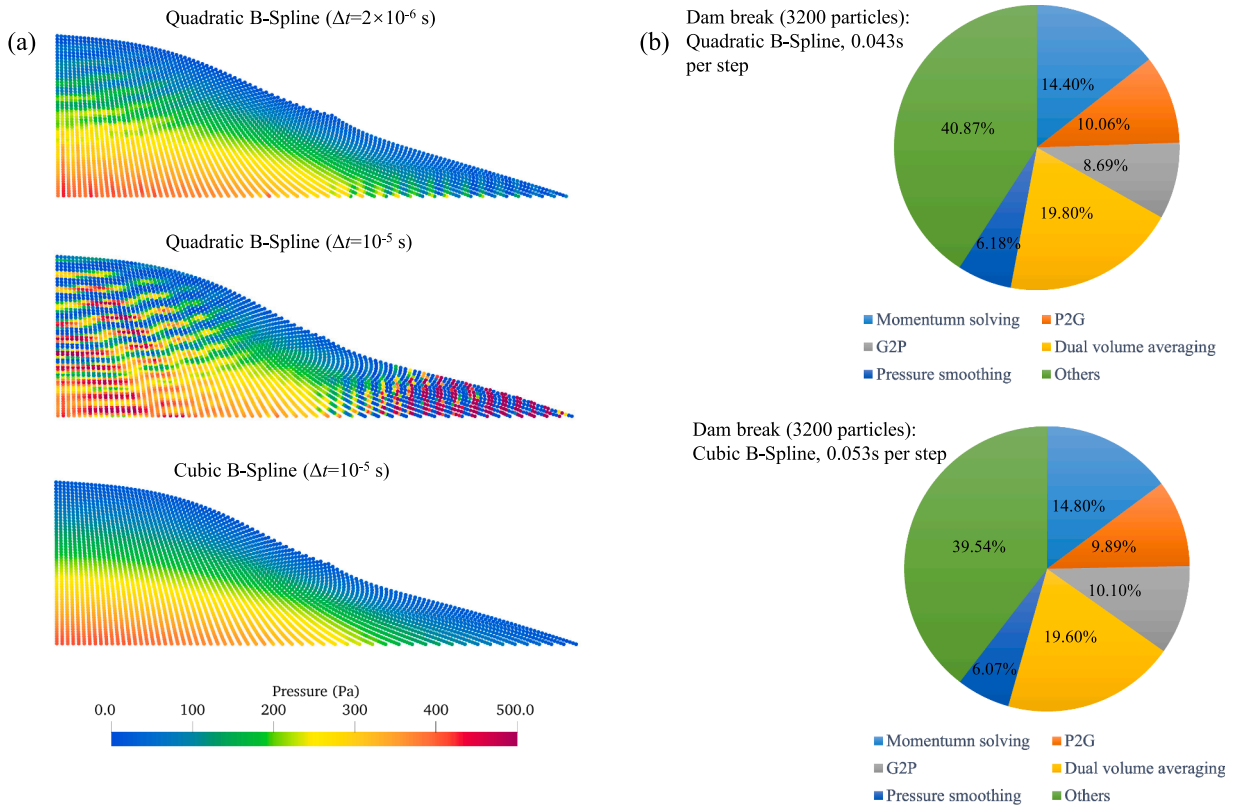

Fig. 21. Comparison between quadratic and cubic B-spline functions: (a) configuration at $t = 0.1$ s with different time steps; (b) running-time profile.

higher-order B-spline functions in fluid flow simulations. Fig. 21 compares the performance of quadratic and cubic B-spline functions in terms of both the pressure field and running time profiles. As illustrated in Fig. 21 (a), pressure oscillations are observed when quadratic B-spline functions are used. Furthermore, the time step for quadratic B-splines must be smaller to prevent significant numerical instability. For instance, when the time step is increased from $\Delta t = 2 \times 10^{-6}$ s to $\Delta t = 10^{-5}$ s, substantial pressure oscillations and numerical instability occur, even leading to a breakdown in computation at $t = 0.2$ s. In contrast, cubic B-spline functions allow for a larger time step (e.g., $\Delta t = 10^{-5}$ s), resulting in a smoother pressure field. The running time profiles are also compared in Fig. 21 (b), where the time per step increases from 0.043 s to 0.053 s when using cubic B-spline functions, compared to quadratic ones. Despite some per-step time increment (23.2 %), the time step can be enlarged (400 %). This makes the adoption of cubic B-spline functions essential. Additionally, Fig. 21 (b) reveals approximately a 20 % increase in runtime due to the dual volume averaging technique, which is deemed acceptable given the overall improvements in fluid flow simulations.

4.6. Dam break with a rigid wall

The proposed stabilized MPM is used to simulate the impact of a water column on a rigid wall. This scenario presents greater challenges compared to cases without a wall, as the water pressure tends to become unstable and induces significant oscillations upon impact. As shown in Fig. 22 (a) and (b), two distinct 2D cases with different geometries ($L/l_0 = 1.5$ and $L/l_0 = 2.7$) are examined. Both cases have been previously studied in the literature [44,60,62], ensuring a robust comparison across various methods. A larger L/l_0 ratio corresponds to a higher conversion of potential energy into kinetic energy, resulting in a more intense impact with the rigid wall. Moreover, in cases with a longer bottom, the rebound of water exhibits a more complex configuration. Therefore, simulating Case 2 ($L/l_0 = 2.7$) presents greater challenges than simulating Case 1 ($L/l_0 = 1.5$). The existing numerical results serve as a reliable benchmark for comparison, facilitating the evaluation of accuracy and performance of the proposed approach.

4.6.1. Case 1: $L/l_0 = 1.5$

Fig. 22 (a) illustrates the geometry and boundary conditions of the problem setup. The model consists of a water column with a length of 4 m and a height of 2 m, initially contained by a gate, which is subsequently released to allow the water to flow. A rigid wall is positioned at the right boundary to facilitate water rebound. In accordance with the approach of Zhao et al. [60], the water is modeled as a weakly compressible Newtonian fluid with a bulk modulus $K = 20$ MPa, dynamic viscosity $\mu = 0.001$ Pa · s, density $\rho = 1000\,\mathrm{kg/m^3}$ and gravity acceleration $g = 9.81\,\mathrm{m/s^2}$. The background grid is discretized into quadrilateral elements, each with a size of 0.1 m, and each element is initialized with 4 material points, yielding a total of 3200 material points, consistent with the configuration presented by Zhao et al. [60]. The blending coefficient β, representing a fraction of APIC, is set to 0.05 due to the low grid resolution, and the time step Δt is fixed at 0.0001 s.

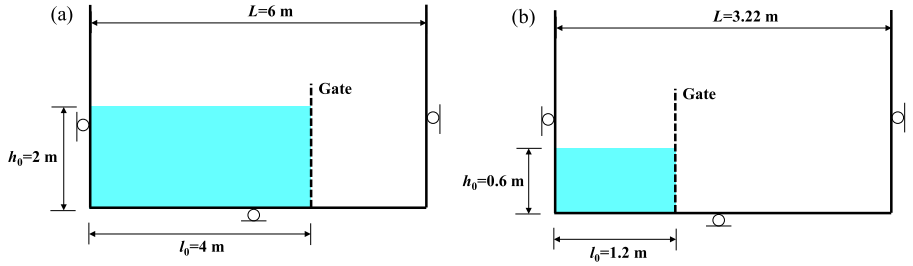

Fig. 22. Initial configuration of dam break with a rigid wall simulation: (a) $L/l_0 = 1.5$; (b) $L/l_0 = 2.7$.

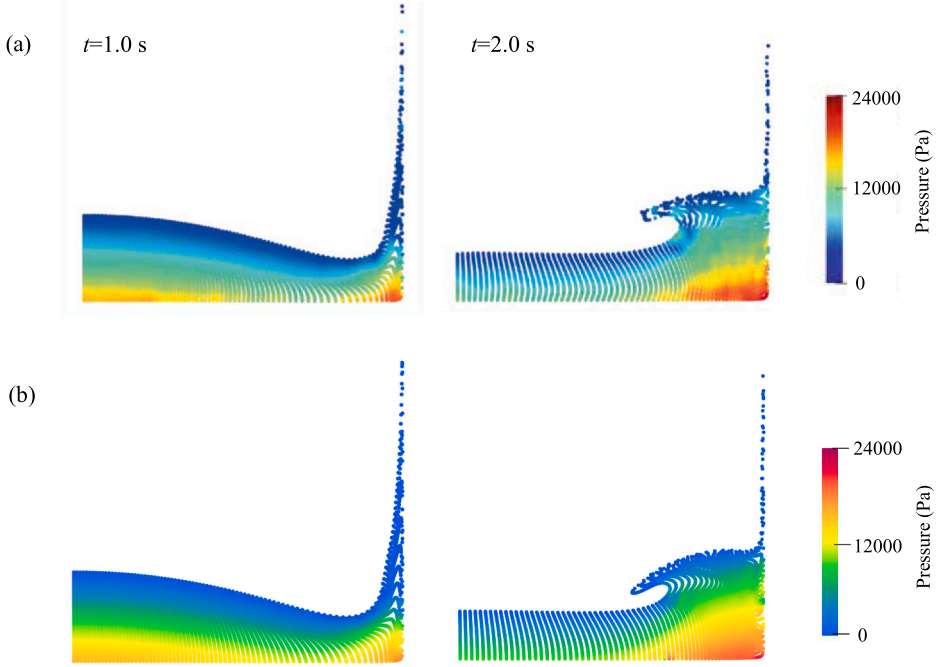

Fig. 23. Comparison between pressure field simulated by (a) Zhao et al. 2023 [60] and (b) this study.

Fig. 23 presents flow snapshots simulated using the proposed stabilized MPM method, compared with reference solutions from Zhao et al. [60]. Pressure smoothing is excluded to ensure a more equitable comparison. Both simulations utilize a weakly compressible fluid model, and the total number of discrete material points for the fluid domain is same, with values of 3200. Quadratic B-spline basis functions were used in previous study and the time step was set to $\Delta t = 10^{-5}$ s. As shown, while Zhao et al. [60] effectively mitigates volumetric locking to a large extent through volume averaging of the deformation gradient, some pressure oscillations remain, particularly near the boundary. In contrast, the proposed improved method demonstrates a similar free surface configuration but further alleviates the volumetric locking issue, resulting in a more optimized pressure distribution. A noteworthy difference is observed in the pressure near the free surface. By employing the free-surface detection algorithm [29,84], the proposed method sets the pressure to zero near the free surface. However, in the results from Zhao et al. [60], non-zero pressure values are observed in certain regions near the free surface, especially at $t = 2$ s.

In subsequent simulations, the proposed method continues to face challenges in mitigating pressure oscillations, primarily due to repetitive water impacts with the wall and the associated disturbances. As shown in Fig. 24 (a), at $t = 6$ s, although the water tends to stabilize, pressure fluctuations persist in the left part of the tank. This phenomenon is attributed to the local particle clustering. Specifically, the particles in the right part of the tank are more uniformly distributed, leading to a more stable pressure field in that region. As previously explained, particle clustering results in increased spatial integration errors in the MPM, which can exacerbate volumetric locking and pressure instability [56]. The application of the δ-correction technique has been shown to improve this issue. Fig. 24 (b) illustrates the pressure field at $t = 6$ s after the application of the δ-correction technique. As observed, the particle distribution becomes more uniform, and the pressure field shows significant improvement compared to Fig. 24 (a).

4.6.2. Case 2: $L/l_0 = 2.7$

The following case, with a different configuration, is simulated using the proposed MPM framework. This problem has been explored in previous studies employing various numerical approaches, making it a benchmark for comparison [29,44,45,62,103].

Fig. 22 (b) shows the initial geometry and boundary conditions for the setup. The water column has an initial length of $l_0 = 1.2$ m and height $h_0 = 0.6$ m, confined behind a virtual gate within a container measuring 3.22 m by 2.5 m. All walls are considered frictionless boundaries. The fluid has a mass density of $\rho = 1000\,\mathrm{kg/m^3}$, dynamic viscosity $\mu = 0.001\,\mathrm{Pa \cdot s}$, and bulk modulus $K = 20$ MPa. At $t = 0$ s, after the fluid has consolidated under gravity, the gate is suddenly released, allowing the water to flow freely under the influence of gravity ($g = 9.81\,\mathrm{m/s^2}$).

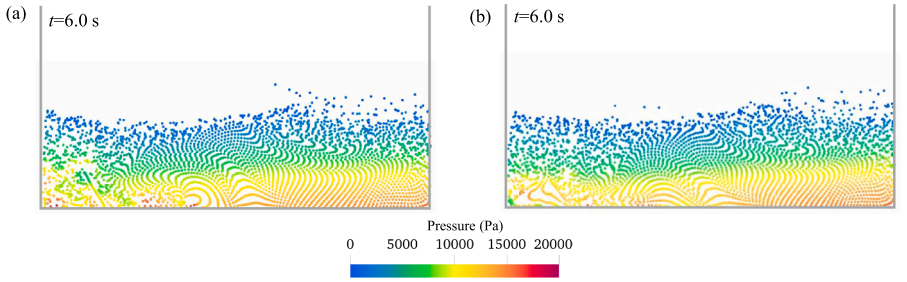

Fig. 24. Pressure contours of simulation results at $t = 6$ s performed (a) without and (b) with δ-correction.

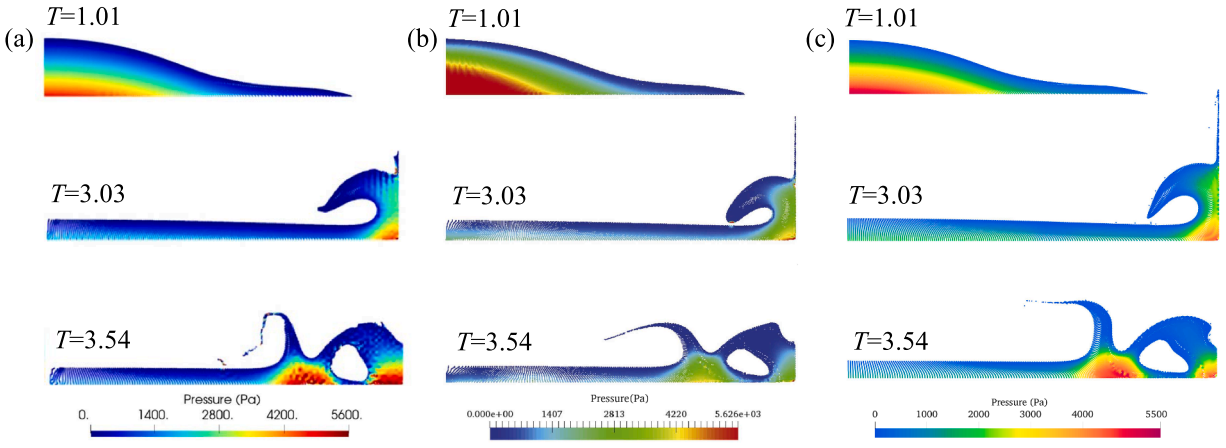

Fig. 25. Comparison of pressure contours with previous study: (a) Displacement-based MPM [62]; (b) vp-WCMPM [44]; (c) this study.

A structured quadrilateral element is used for grid discretization in the MPM, with a elemsize size of $h = 0.01$ m, where each grid cell contains four material points. This results in a total of 28,800 material points for the fluid domain. The time step is set to $\Delta t = 5 \times 10^{-5}$ s for the simulation. Fig. 25 shows a series of snapshots depicting simulation results at different normalized times: $T = 1.01$, $T = 3.03$, and $T = 3.54$, where T is the normalized time, as defined in Eq. (62).

Fig. 25 also presents a comparison of snapshots predicted by the vp-WCMPM method [44] and the displacement-based MPM [62]. Pressure smoothing is not applied to facilitate a more balanced comparison. Both methods utilize a weakly compressible fluid model, and the total number of discrete material points for the fluid domain is close, with values of 28,800 and 25,088, respectively. Quadratic B-spline basis functions were used in the displacement-based MPM and the time step was set to $\Delta t = 2 \times 10^{-6}$ s, while GIMP basis functions were employed for the vp-WCMPM method. Although the free surface profiles generated by both methods are generally comparable, some differences are observed in the pressure field. At $T = 1.01$, slight oscillations are present in the pressure contours near the bottom of the domain in the vp-WCMPM simulation. At $T = 3.03$ and $T = 3.54$, the displacement-based

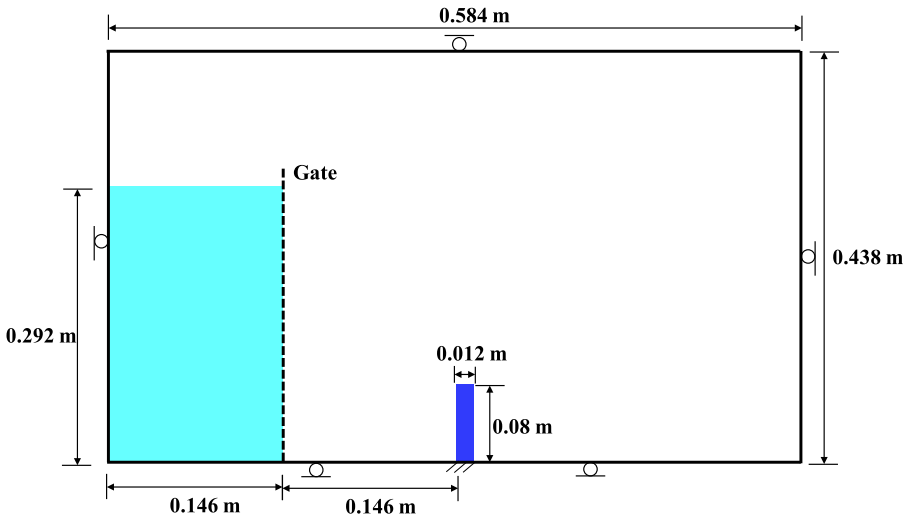

Fig. 26. Problem setup.

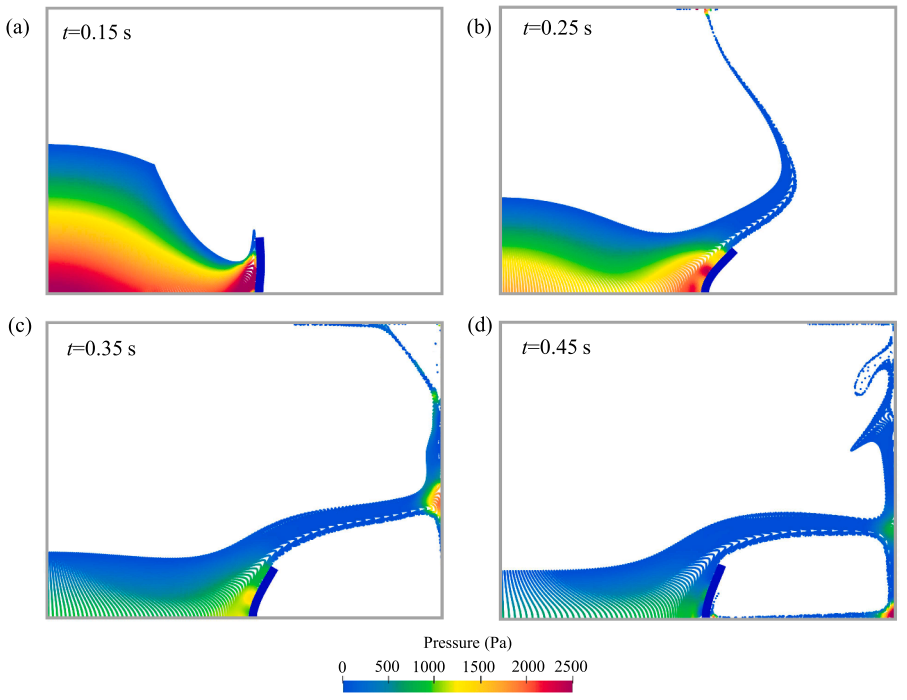

Fig. 27. Simulation snapshots at different time: (a) $t = 0.15$ s; (b) $t = 0.25$ s; (c) $t = 0.35$ s; (d) $t = 0.45$ s.

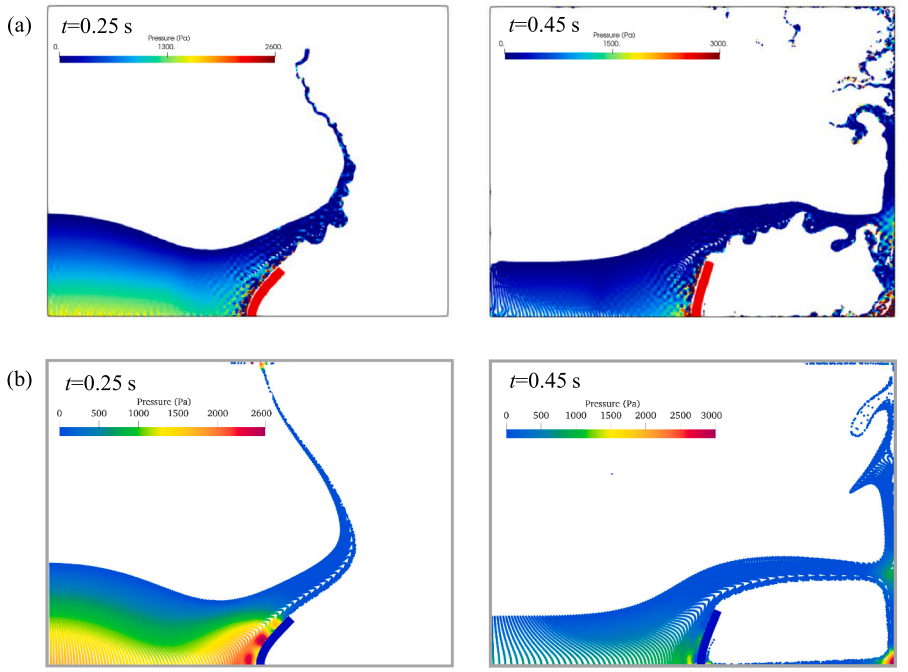

Fig. 28. Comparison with previous study: (a) Displacement-based MPM [62]; (b) this study.

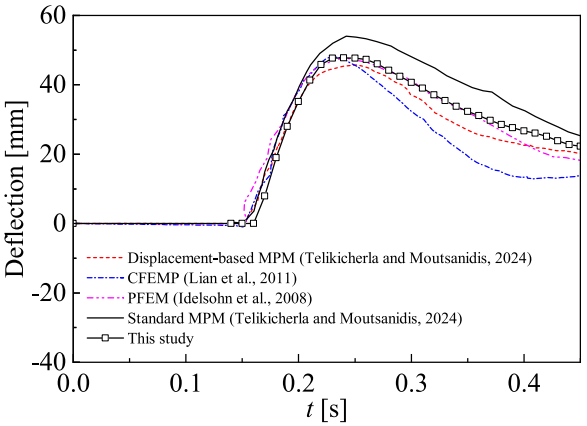

Fig. 29. Comparison of horizontal displacement of the obstacle with different methods.

MPM exhibits pressure fluctuations in regions of high pressure concentration. In contrast, the pressure distribution from the proposed method remains relatively smooth throughout the simulation, with free surface profiles demonstrating consistent and reasonable behavior. These observations suggest that the proposed method effectively mitigates volumetric locking and pressure instability, thereby offering improved performance for weakly compressible flow problems.

4.7. Water impact with an elastic obstacle

In this section, a dynamic problem involving the impact of a water column on an elastic obstacle is investigated, which serves as a classic case for validating FSI simulations, as illustrated in Fig. 26. The initial dimensions of the water column are $h_0 = 0.292\,\text{m}$ in height and $l_0 = 0.146\,\text{m}$ in length. The elastic obstacle has a height of $h = 0.08\,\text{m}$ and a length of $l = 0.012\,\text{m}$. The water properties are set as follows: density $\rho_w = 1000\,\text{kg/m}^3$, bulk modulus $K = 10\,\text{MPa}$, dynamic viscosity $\mu = 0.001\,\text{Pa}\cdot\text{s}$, and gravity acceleration $g = 9.81\,\text{m/s}^2$. The material of the obstacle is elastic with a density $\rho_s = 2500\,\text{kg/m}^3$, modulus $E = 1\,\text{MPa}$, and Poisson's ratio $\nu = 0.0$. The grid size is $0.00584\,\text{m}$, with each cell containing 16 material points. The model contains 20,000 material points for the water domain and 434 material points for the obstacle. A blending coefficient of $\beta = 0.05$ is used due to the relatively low grid resolution, and the time step Δt is set by $\Delta t = 1 \times 10^{-5}\,\text{s}$. This FSI problem has been examined using various numerical methods, including the

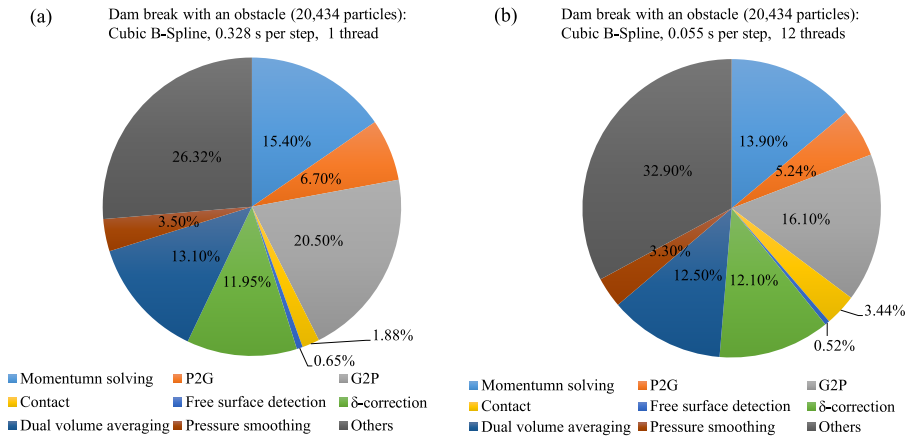

Fig. 30. Breakdown of time cost per step for the simulation of dam break with an obstacle: (a) single-thread; (b) multi-thread.

vp-WCMPM [44], displacement-based MPM [62], particle finite element method (PFEM) [104], coupled finite element-material point method (CFEMP) [105], and the improved coupled finite element-material point method (ICFEMP) [106].

The deformation of the elastic obstacle and the evolution of the free surface at four distinct time instances ($t = 0.15\,\text{s}, 0.25\,\text{s}, 0.35\,\text{s}, 0.45\,\text{s}$) are shown in Fig. 27. As observed, the proposed method produces a stable and smooth pressure field, which aligns closely with the results documented in the literature [44,62,104,105]. However, some pressure oscillations are noted in the contact region. Specifically, the pressure near the obstacle is less smooth compared to other regions. This is likely attributed to numerical oscillations stemming from the DEM-enriched contact algorithm. During the water impact on the obstacle, significant deformation occurs. The high contact stiffness causes the contact force to change dramatically over a short time during the collision process, which may induce instability in the contact force calculations. These fluctuating contact forces are applied to the material points as body forces, leading to potential numerical oscillations for pressure. Fig. 28 presents a comparison of the simulation results between the Displacement-based MPM [62] and our method. To ensure a more equitable comparison, pressure smoothing is not applied here. Quadratic B-spline basis functions were used in the displacement-based MPM and the time step was set to $\Delta t = 5 \times 10^{-7}$ s. The evolution of the water exhibits a similar configuration at different time steps. However, pressure oscillations are significantly mitigated by our method, suggesting the advantages and stability of our method in dealing with FSI problems. Furthermore, the temporal variation of the horizontal deflection at the top-left corner of the obstacle, shown in Fig. 29, is well benchmarked with results from other studies [44,62,104,105].

Fig. 30 illustrates the breakdown of the average computation time per step. For a total of 20,434 particles, the average computation time per step is 0.328 seconds when using single-thread computing. In contrast, this running time decreases to 0.055 seconds with the implementation of multi-thread acceleration (12 threads). The cumulative running time proportions for various components, including dual volume averaging, δ-correction, pressure smoothing, and contact and free surface detection, are 31 % for single-thread computing and 36 % for multi-thread computing. These values are considered acceptable, given the improvements in numerical stability and accuracy. To further enhance the speed of numerical simulations, the integration of GPU computing is anticipated in the future [31].

5. Conclusions

This study presents a stabilized explicit MPM framework for simulating weakly compressible fluid flows and their interaction with structures. To mitigate the cell-crossing instability commonly observed in traditional methods, cubic B-spline basis functions are employed, replacing the conventional linear or second-order basis functions. Volumetric locking is effectively addressed through dual volume averaging applied to both deformation gradient and pressure, which are based on the cubic B-spline particle-grid mapping scheme. To counteract the accuracy loss at low grid resolutions inherent in the APIC mapping scheme, a blended APIC/FLIP strategy is introduced. This scheme enhances the stability of the grid-to-particle mapping while ensuring energy conservation throughout the simulation. Additionally, to reduce quadrature errors caused by particle clusters, the δ-correction method is implemented, further stabilizing the numerical solution. A simple yet effective free surface tracking technique, combined with pressure smoothing, is also employed to enhance overall simulation performance. Furthermore, the proposed framework is extended to FSI problems by incorporating an advanced DEM enriched contact algorithm, which effectively mitigates pressure oscillations and enhances the stability of FSI simulations.

A series of benchmark tests, including 1D elastic wave propagation, Poiseuille flow, lid-driven cavity flow, water sloshing, dam break, and the impact of water on obstacles, are conducted to verify and validate the proposed framework. The results demonstrate good agreement with experimental data as well as numerical results from other studies. Notably, the proposed stabilized explicit MPM scheme maintains numerical stability and pressure smoothing, even for challenging dynamic problems. This approach offers a

promising avenue for large-deformation simulations involving fluid flow and fluid-solid contact, thereby advancing the application of MPM in complex FSI scenarios.

The benchmark tests are limited to 2D problems, but the proposed framework can be readily extended to 3D ones to address challenges such as volumetric locking, cell-crossing instability, and excessive energy dissipation, as the spatial dimension is already incorporated in the formulation. Compared with the vanilla explicit MPM, the computational cost of the proposed framework may moderately increase, which can be, however, mitigated by parallel computing, e.g., OpenMP used in this study or GPU computing as implemented in our recent work [31]. As for the stabilized technique, it is worth pointing out that pressure oscillations tend to increase over time as particle clumping becomes more pronounced. The δ-correction has limited effectiveness in addressing this issue, highlighting the need for a more effective PST technique for MPM. Moreover, the DEM-enriched contact method may exacerbate pressure fluctuations in FSI problems due to numerical instability in large deformations. These limitations are expected to be addressed in our further studies. It is important to acknowledge that the present work offers a robust, practical framework for engineering applications rather than a mathematically rigorous solution. We view this as a necessary and pragmatic compromise to address immediate stability challenges. A truly fundamental and universal mathematical formulation for MPM remains a valuable and meaningful objective for future studies.

CRediT authorship contribution statement

Zhang Cheng: Writing – original draft, Methodology, Investigation; **Shiwei Zhao:** Writing – review & editing, Supervision, Resources, Funding acquisition; **Hao Chen:** Writing – review & editing, Methodology; **Jidong Zhao:** Writing – review & editing, Supervision, Resources, Funding acquisition.

Data availability

The simulation scripts will be available on request.

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

This work was financially supported by National Natural Science Foundation of China (by Project No. 52439001) and Research Grants Council of Hong Kong (by GRF Projects No. 16206322 and No. 16211221, CRF Project No. C7082-22G, and TRS Projects No. T22-606/23-R and No. T22-603/24N). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the financial bodies.

References

[1] F.Z. Wang, I. Animasaun, T. Muhammad, S. Okoya, Recent advancements in fluid dynamics: drag reduction, lift generation, computational fluid dynamics, turbulence modelling, and multiphase flow, Arab. J. Sci. Eng. 49 (8) (2024) 10237–10249.
[2] S. Sharma, D. Goswami, M. Goswami, A. Deb, B. Padhan, S. Chattopadhyay, Computational fluid dynamics modeling of multicomponent elastomeric complex profile while flowing through extrusion die, Chem. Eng. J. 451 (2023) 138756.
[3] Y. Li, Z. Guo, L. Wang, H. Yang, A coupled bio-chemo-hydro-wave model and multi-stages for micp in the seabed, Ocean Eng. 280 (2023) 114667.
[4] H. Yang, Z. Guo, Y. Li, H. Wang, Y. Dou, Experimental study on microbubble drag reduction on soil-steel interface, Appl. Ocean Res. 116 (2021) 102891.
[5] B. He, P. Yue, Y. Zhu, H. Liu, Z. Guo, Y. Li, Experimental study on hole erosion behaviors of marine soil solidified by an innovative organic composite material, Appl. Ocean Res. 163 (2025) 104755.
[6] B. Després, C. Mazeran, Lagrangian gas dynamics in two dimensions and lagrangian systems, Arch. Ration. Mech. Anal. 178 (3) (2005) 327–372.
[7] C. Michler, S. Hulshoff, E.V. Brummelen, R.D. Borst, A monolithic approach to fluid-structure interaction, Comput. Fluids 33 (5–6) (2004) 839–848.
[8] J.W. Thomas, Numerical Partial Differential Equations: Finite Difference Methods, 22, Springer Science & Business Media, 2013.
[9] J.N. Reddy, D.K. Gartling, The finite element method in heat transfer and fluid dynamics, CRC Press, 2010.
[10] Y. Li, L. Wang, C. Dong, G. Feng, X. Sun, Z. Guo, A coupled mathematical model of microbial grouting reinforced seabed considering the response of wave-induced pore pressure and its application, Comput. Geotech. 184 (2025) 107235.
[11] X. Wang, Z. Yin, W. Wu, H. Zhu, Differentiable finite element method with Galerkin discretization for fast and accurate inverse analysis of multidimensional heterogeneous engineering structures, Comput. Methods Appl. Mech. Eng. 437 (2025) 117755.
[12] N. Ashgriz, J. Mostaghimi, An introduction to computational fluid dynamics, Fluid Flow Handbook 2002. 1–49.
[13] R.A. Gingold, J.J. Monaghan, Smoothed particle hydrodynamics: theory and application to non-spherical stars, Mon. Not. R. Astron. Soc. 181 (3) (1977) 375–389.
[14] R. Feng, G. Fourtakas, B.D. Rogers, D. Lombardi, A general smoothed particle hydrodynamics (sph) formulation for coupled liquid flow and solid deformation in porous media, Comput. Methods Appl. Mech. Eng. 419 (2024) 116581.
[15] R. Feng, G. Fourtakas, B.D. Rogers, D. Lombardi, Modelling internal erosion using 2d smoothed particle hydrodynamics (sph), J. Hydrol. 639 (2024) 131558.
[16] D. Sulsky, Z. Chen, H.L. Schreyer, A Particle Method for History-Dependent Materials, Technical Report, Sandia National Lab.(SNL-NM), Albuquerque, NM (United States), 1993.
[17] J. Yu, J. Zhao, W. Liang, S. Zhao, A semi-implicit material point method for coupled thermo-hydro-mechanical simulation of saturated porous media in large deformation, Comput. Methods Appl. Mech. Eng. 418 (2024) 116462.
[18] W. Liang, K.-Y. He, Y.-F. Jin, Z.-Y. Yin, A gradient-smoothed material point method for reducing cell crossing noise in large deformation problems, Comput. Geotech. 169 (2024) 106169.
[19] W. Liang, J. Zhao, Multiscale modeling of large deformation in geomechanics, Int. J. Numer. Anal. Methods Geomech. 43 (5) (2019) 1080–1114.

[20] M. Molinos, P. Navas, M. Pastor, M.M. Stickle, On the dynamic assessment of the local-maximum entropy material point method through an explicit predictor-corrector scheme, Comput. Methods Appl. Mech. Eng. 374 (2021) 113512.
[21] S.A. Silling, M.L. Parks, J.R. Kamm, O. Weckner, M. Rassaian, Modeling shockwaves and impact phenomena with eulerian peridynamics, Int. J. Impact Eng. 107 (2017) 47–57.
[22] C. Yang, F. Zhu, J. Zhao, Coupled total-and semi-lagrangian peridynamics for modelling fluid-driven fracturing in solids, Comput. Methods Appl. Mech. Eng. 419 (2024) 116580.
[23] C. Yang, F. Zhu, J. Zhao, A multi-horizon fully coupled thermo-mechanical peridynamics, J. Mech. Phys. Solids 191 (2024) 105758.
[24] C. Yang, J. Zhao, F. Zhu, R. Feng, A multi-horizon peridynamics for coupled fluid flow and heat transfer, J. Fluid Mech. 1010 (2025) 66.
[25] F. Vilar, P.-H. Maire, R. Abgrall, A discontinuous galerkin discretization for solving the two-dimensional gas dynamics equations written under total lagrangian formulation on general unstructured grids, J. Comput. Phys. 276 (2014) 188–234.
[26] S. Koshizuka, Y. Oka, Moving-particle semi-implicit method for fragmentation of incompressible fluid, Nucl. Sci. Eng. 123 (3) (1996) 421–434.
[27] D. Violeau, B.D. Rogers, Smoothed particle hydrodynamics (sph) for free-surface flows: past, present and future, J. Hydraul. Res. 54 (1) (2016) 1–26.
[28] H. Gotoh, A. Khayyer, On the state-of-the-art of particle methods for coastal and ocean engineering, Coast. Eng. J. 60 (1) (2018) 79–103.
[29] B. Chandra, R. Hashimoto, S. Matsumi, K. Kamrin, K. Soga, Stabilized mixed material point method for incompressible fluid flow analysis, Comput. Methods Appl. Mech. Eng. 419 (2024) 116644.
[30] S. Zhao, Z. Lai, J. Zhao, Leveraging ray tracing cores for particle-based simulations on gpus, Int. J. Numer. Methods Eng. 124 (3) (2023) 696–713.
[31] H. Chen, S. Zhao, J. Zhao, A sparse-memory-encoding GPU-MPM framework for large-scale simulations of granular flows, Comput. Geotech. 180 (2025) 107113.
[32] P.J. Baioni, T. Benacchio, L. Capone, C.D. Falco, Portable, massively parallel implementation of a material point method for compressible flows, Comput. Part. Mech. 2024, pp. 1–23.
[33] M. Molinos, B. Chandra, M.M. Stickle, K. Soga, On the derivation of a component-free scheme for lagrangian fluid-structure interaction problems, Acta Mech. 234 (5) (2023) 1777–1809.
[34] J. Yu, W. Liang, J. Zhao, Enhancing dynamic modeling of porous media with compressible fluid: A THM material point method with improved fractional step formulation, Comput. Methods Appl. Mech. Eng. 444 (2025) 118100.
[35] J. Yu, J. Zhao, K. Soga, S. Zhao, W. Liang, A fully coupled THMC-MPM framework for modeling phase transition and large deformation in methane hydrate-bearing sediment, J. Mech. Phys. Solids. 206 (2026) 106368.
[36] F.H. Harlow, The particle-in-cell computing method for fluid dynamics, Methods Comput. Phys. 3 (1964) 319–343.
[37] A. Vaucorbeil, V.P. Nguyen, C.R. Hutchinson, A total-lagrangian material point method for solid mechanics problems involving large deformations, Comput. Methods Appl. Mech. Eng. 360 (2020) 112783.
[38] W.M. Coombs, Ghost stabilisation of the material point method for stable quasi-static and dynamic analysis of large deformation problems, Int. J. Numer. Methods Eng. 124 (21) (2023) 4841–4875.
[39] Z. Sun, H. Li, Y. Gan, H. Liu, Z. Huang, L. He, Material point method and smoothed particle hydrodynamics simulations of fluid flow problems: a comparative study, Progress Comput. Fluid Dyn. Int. J. 18 (1) (2018) 1–18.
[40] J. Li, Y. Hamamoto, Y. Liu, X. Zhang, Sloshing impact simulation with material point method and its experimental validations, Comput. Fluids 103 (2014) 86–99.
[41] X. Zhao, D. Liang, M. Martinelli, Mpm simulations of dam-break floods, J. Hydrodyn. 29 (3) (2017) 397–404.
[42] C. Mast, P. Mackenzie-Helnwein, P. Arduino, G.R. Miller, W. Shin, Mitigating kinematic locking in the material point method, J. Comput. Phys. 231 (16) (2012) 5351–5373.
[43] A.S. Baumgarten, K. Kamrin, A general fluid-sediment mixture model and constitutive theory validated in many flow regimes, J. Fluid Mech. 861 (2019) 721–764.
[44] Z.-P. Chen, X. Zhang, K.Y. Sze, L. Kan, X.-M. Qiu, Vp material point method for weakly compressible problems, Comput. Fluids, 176 (2018) 170–181.
[45] F. Zhang, X. Zhang, K.Y. Sze, Y. Lian, Y. Liu, Incompressible material point method for free surface flow, J. Comput. Phys. 330 (2017) 92–110.
[46] S. Kularathna, K. Soga, Implicit formulation of material point method for analysis of incompressible materials, Comput. Methods Appl. Mech. Eng. 313 (2017) 673–686.
[47] W. Liang, J. Zhao, H. Wu, K. Soga, Multiscale, multiphysics modeling of saturated granular materials in large deformation, Comput. Methods Appl. Mech. Eng. 405 (2023) 115871.
[48] L. Moreno, R. Wuechner, A. Larese, A mixed stabilized mpm formulation for incompressible hyperelastic materials using variational subgrid-scales, Comput. Methods Appl. Mech. Eng. 435 (2025) 117621.
[49] J. Dolbow, T. Belytschko, Volumetric locking in the element free galerkin method, Int. J. Numer. Methods Eng. 46 (6) (1999) 925–942.
[50] W.M. Coombs, T.J. Charlton, M. Cortis, C.E. Augarde, Overcoming volumetric locking in material point methods, Comput. Methods Appl. Mech. Eng. 333 (2018) 1–21.
[51] S.G. Bardenhagen, E.M. Kober, et al., The generalized interpolation material point method, Comput. Model. Eng. Sci. 5 (6) (2004) 477–496.
[52] C. Jiang, C. Schroeder, A. Selle, J. Teran, A. Stomakhin, The affine particle-in-cell method, ACM Trans. Graph. 34 (4) (2015) 1–10.
[53] C. Fu, Q. Guo, T. Gast, C. Jiang, J. Teran, A polynomial particle-in-cell method, ACM Trans. Graph. 36 (6) (2017) 1–12.
[54] K. Nakamura, S. Matsumura, T. Mizutani, Taylor particle-in-cell transfer and kernel correction for material point method, Comput. Methods Appl. Mech. Eng. 403 (2023) 115720.
[55] M. Steffen, R.M. Kirby, M. Berzins, Analysis and reduction of quadrature errors in the material point method (mpm), Int. J. Numer. Methods Eng. 76 (6) (2008) 922–948.
[56] A.S. Baumgarten, K. Kamrin, Analysis and mitigation of spatial integration errors for the material point method, Int. J. Numer. Methods Eng. 124 (11) (2023) 2449–2497.
[57] E.A. D.S. Neto, D. Peric, D.R. Owen, Computational Methods for Plasticity: Theory and Applications, John Wiley & Sons, 2008.
[58] E.D.S. Neto, D. Perić, M. Dutko, D. Owen, Design of simple low order finite elements for large strain analysis of nearly incompressible solids, Int. J. Solids Struct. 33 (20–22) (1996) 3277–3296.
[59] J. Simo, R.L. Taylor, K. Pister, Variational and projection methods for the volume constraint in finite deformation elasto-plasticity, Comput. Methods Appl. Mech. Eng. 51 (1–3) (1985) 177–208.
[60] Y. Zhao, C. Jiang, J. Choo, Circumventing volumetric locking in explicit material point methods: a simple, efficient, and general approach, Int. J. Numer. Methods Eng. 124 (23) (2023) 5334–5355.
[61] R.M. Telikicherla, G. Moutsanidis, Treatment of near-incompressibility and volumetric locking in higher order material point methods, Comput. Methods Appl. Mech. Eng. 395 (2022) 114985.
[62] R.M. Telikicherla, G. Moutsanidis, A Displacement-based Material Point Method for Weakly Compressible Free-surface Flows, Comput. Mech. 2024, pp. 1–17.
[63] Z. Sun, Y. Hua, Y. Xu, X. Zhou, Simulation of fluid-structure interaction using the density smoothing b-spline material point method with a contact approach, Comput. Math. Appl. 176 (2024) 525–544.
[64] K.-Y. He, Y.-F. Jin, X.-W. Zhou, Z.-Y. Yin, X. Chen, A generalized projection algorithm for overcoming volumetric locking in explicit material point method, Comput. Geotech. 186 (2025) 107391.
[65] W.M. Coombs, R.E. Bird, G. Pretti, The aggregated material point method (agmpm), Comput. Methods Appl. Mech. Eng. 442 (2025) 118012.
[66] A. Sadeghirad, R.M. Brannon, J. Burghardt, A convected particle domain interpolation technique to extend applicability of the material point method for problems involving massive deformations, Int. J. Numer. Methods Eng. 86 (12) (2011) 1435–1456.
[67] A. Sadeghirad, R.M. Brannon, J. Guilkey, Second-order convected particle domain interpolation (cpdi2) with enrichment for weak discontinuities at material interfaces, Int. J. Numer. Methods Eng. 95 (11) (2013) 928–952.
[68] D.Z. Zhang, X. Ma, P.T. Giguere, Material point method enhanced by modified gradient of shape function, J. Comput. Phys. 230 (16) (2011) 6379–6398.

[69] R. Tielen, E. Wobbes, M. Möller, L. Beuth, A high order material point method, Procedia Eng. 175 (2017) 265–272.
[70] Y. Gan, Z. Sun, Z. Chen, X. Zhang, Y. Liu, Enhancement of the material point method using b-spline basis functions, Int. J. Numer. Methods Eng. 113 (3) (2018) 411–431.
[71] A.D. Vaucorbeil, V.P. Nguyen, S. Sinaie, J.Y. Wu, Material point method after 25 years: theory, implementation, and applications, Adv. Appl. Mech. 53 (2020) 185–398.
[72] J.U. Brackbill, H.M. Ruppel, Flip: a method for adaptively zoned, particle-in-cell calculations of fluid flows in two dimensions, J. Comput. Phys. 65 (2) (1986) 314–343.
[73] S. Bardenhagen, Energy conservation error in the material point method for solid mechanics, J. Comput. Phys. 180 (1) (2002) 383–403.
[74] M. Steffen, P. Wallstedt, J. Guilkey, R. Kirby, M. Berzins, Examination and analysis of implementation choices within the material point method (mpm), Comput. Model. Eng. Sci. 31 (2) (2008) 107–127.
[75] F. Zhang, X. Zhang, K.Y. Sze, Y. Liang, Y. Liu, Improved incompressible material point method based on particle density correction, Int. J. Comput. Methods 15 (07) (2018) 1850061.
[76] C. Jiang, C. Schroeder, J. Teran, An angular momentum conserving affine-particle-in-cell method, J. Comput. Phys. 338 (2017) 137–164.
[77] R. Courant, K. Friedrichs, H. Lewy, On the partial difference equations of mathematical physics, IBM J. Res. Dev. 11 (2) (1967) 215–234.
[78] X. Zhang, Z. Chen, Y. Liu, The material point method: A continuum-based particle method for extreme loading cases, Academic Press, 2016.
[79] G. Pretti, W.M. Coombs, C.E. Augarde, B. Sims, M.M. Puigvert, J.A.R. Gutiérrez, A conservation law consistent updated lagrangian material point method for dynamic analysis, J Comput Phys 485 (2023) 112075.
[80] M.G. Cox, The numerical evaluation of b-splines, IMA J. Appl. Math. 10 (2) (1972) 134–149.
[81] G. Moutsanidis, C.C. Long, Y. Bazilevs, Iga-mpm: the isogeometric material point method, Comput. Methods Appl. Mech. Eng. 372 (2020) 113346.
[82] T.J. Hughes, The Finite Element Method: Linear Static and Dynamic Finite Element Analysis, Technical Report, Courier Corporation, 2012.
[83] T. Elguedj, Y. Bazilevs, V.M. Calo, T.J. Hughes, B and f projection methods for nearly incompressible linear and non-linear elasticity and plasticity using higher-order nurbs elements, Comput. Methods Appl. Mech. Eng. 197 (33–40) (2008) 2732–2762.
[84] S. Kularathna, Splitting Solution Scheme for Material Point Method, Technical Report, University of Cambridge, 2018. Ph.D. thesis.
[85] H. Chen, S. Zhao, J. Zhao, X. Zhou, Dem-enriched contact approach for material point method, Comput. Methods Appl. Mech. Eng. 404 (2023) 115814.
[86] X.-W. Zhou, Y.-F. Jin, K.-Y. He, Z.-Y. Yin, An improved explicit mpm formulation and its coupling scheme with fem, Comput Methods Appl Mech Eng 436 (2025) 117734.
[87] X.-W. Zhou, Y.-F. Jin, K.-Y. He, Z.-Y. Yin, F.-T. Liu, A novel implicit fem-mpm coupling framework using convex cone programming for elastoplastic problems, Comput. Methods Appl. Mech. Eng. 429 (2024) 117153.
[88] C. Liu, W. Sun, Ils-mpm: an implicit level-set-based material point method for frictional particulate contact mechanics of deformable particles, Comput. Methods Appl. Mech. Eng. 369 (2020) 113168.
[89] X. Li, Y. Fang, M. Li, C. Jiang, Bfemp: interpenetration-free mpm-fem coupling with barrier contact, Comput. Methods Appl. Mech. Eng. 390 (2022) 114350.
[90] Z. Xi-Wen, J. Yin-Fu, Y. Zhen-Yu, L. Feng-Tao, C. Xiangsheng, A novel improved edge-based smoothed particle finite element method for elastoplastic contact analysis using second order cone programming, Comput. Methods Appl. Mech. Eng. 441 (2025) 118016.
[91] J.P. Morris, P.J. Fox, Y. Zhu, Modeling low reynolds number incompressible flows using sph, J. Comput. Phys. 136 (1) (1997) 214–226.
[92] L.D.G. Sigalotti, J. Klapp, E. Sira, Y. Meleán, A. Hasmy, Sph simulations of time-dependent poiseuille flow at low reynolds numbers, J. Comput. Phys. 191 (2) (2003) 622–638.
[93] S. Kularathna, K. Soga, Comparison of two projection methods for modeling incompressible flows in mpm, J. Hydrodyn. Ser. B 29 (3) (2017) 405–412.
[94] Y. Yue, B. Smith, C. Batty, C. Zheng, E. Grinspun, Continuum foam: a material point method for shear-dependent flows, ACM Trans. Graph. 34 (5) (2015) 1–20.
[95] R. Xu, P. Stansby, D. Laurence, Accuracy and stability in incompressible sph (isph) based on the projection method and a new approach, J. Comput. Phys. 228 (18) (2009) 6703–6725.
[96] U. Ghia, K.N. Ghia, C. Shin, High-re solutions for incompressible flow using the navier-stokes equations and a multigrid method, J. Comput. Phys. 48 (3) (1982) 387–411.
[97] B. Chandra, V. Singer, T. Teschemacher, R. Wüchner, A. Larese, Nonconforming dirichlet boundary conditions in implicit material point method by means of penalty augmentation, Acta Geotech. 16 (8) (2021) 2315–2335.
[98] Z. Chen, Z. Zong, H. Li, J. Li, An investigation into the pressure on solid walls in 2d sloshing using sph method, Ocean Eng. 59 (2013) 129–141.
[99] F. Zhang, X. Zhang, Y. Liu, An augmented incompressible material point method for modeling liquid sloshing problems, Int. J. Mech. Mater. Des. 14 (2018) 141–155.
[100] Y. Chen, K. Djidjeli, W. Price, Numerical simulation of liquid sloshing phenomena in partially filled containers, Comput. Fluids 38 (4) (2009) 830–842.
[101] J.C. Martin, W.J. Moyce, J. Martin, W. Moyce, W.G. Penney, A. Price, C. Thornhill, Part iv. an experimental study of the collapse of liquid columns on a rigid horizontal plane, Philos. Trans. R. Soc. London. Ser. A 244 (882) (1952) 312–324. Mathematical and Physical Sciences.
[102] M. Asai, Y. Li, B. Chandra, S. Takase, Fluid-rigid-body interaction simulations and validations using a coupled stabilized isph-dem incorporated with the energy-tracking impulse method for multiple-body contacts, Comput. Methods Appl. Mech. Eng. 377 (2021) 113681.
[103] J.P. Hughes, D.I. Graham, Comparison of incompressible and weakly-compressible sph models for free-surface water flows, J. Hydraul. Res. 48 (1) (2010) 105–117.
[104] S.R. Idelsohn, J. Marti, A. Limache, E.O. nate, Unified lagrangian formulation for elastic solids and incompressible fluids: application to fluid-structure interaction problems via the pfem, Comput. Methods Appl. Mech. Eng. 197 (19–20) (2008) 1762–1776.
[105] Y. Lian, X. Zhang, Y. Liu, Coupling of finite element method with material point method by local multi-mesh contact method, Comput. Methods Appl. Mech. Eng. 200 (47–48) (2011) 3482–3494.
[106] Z. Chen, X. Qiu, X. Zhang, Y. Lian, Improved coupling of finite element method with material point method based on a particle-to-surface contact algorithm, Comput. Methods Appl. Mech. Eng. 293 (2015) 1–19.