



# A parallel multi-resolution Smoothed Particle Hydrodynamics model with local time stepping

Guixun Zhu<sup>a</sup>, Yongdong Cui<sup>a</sup>, Boo Cheong Khoo<sup>a</sup>, Siming Zheng<sup>b,c,\*</sup>, Zongbing Yu<sup>d,e</sup>, Yelin Gao<sup>d</sup>

<sup>a</sup> Department of Mechanical Engineering, National University of Singapore, 9 Engineering Drive 1, 117575, Singapore

<sup>b</sup> Ocean College, Zhejiang University, Zhoushan, Zhejiang 316021, China

<sup>c</sup> School of Engineering, Computing and Mathematics, University of Plymouth, Drake Circus, Plymouth PL4 8AA, UK

<sup>d</sup> School of Naval Architecture, State Key Laboratory of Structural Analysis for Industrial Equipment, Dalian University of Technology, Dalian 116024, PR China

<sup>e</sup> State Key Laboratory of Hydraulic Engineering Simulation and Safety, Tianjin University, Tianjin, 300072, PR China



## ARTICLE INFO

### Keywords:

Smoothed Particle Hydrodynamics

Multi-resolution

Local time step

Parallel computing

## ABSTRACT

Smoothed Particle Hydrodynamics (SPH) model has exhibited remarkable efficacy in addressing strongly nonlinear large deformation problems. However, the expensive computational cost could limit its application. To address this, we propose a novel parallel multi-resolution SPH model with a local time step strategy. This model integrates the multi-resolution model with a Message Passing Interface-based parallelization framework, wherein subdomains discretize the computational domain at varying resolutions. Meanwhile, a local fourth-order Runge-Kutta time integration scheme is introduced, enabling different subdomains to use different time steps. Subdomains of varying resolutions are devoid of overlapping regions, and changes in particle resolution at interfaces are achieved through a dynamic strategy involving particle merging and splitting. Besides that, we conducted a comparative analysis of different SPH discretization formats concerning their impact on the accuracy of multi-resolution particle discretization. Finally, several numerical tests were conducted to validate the accuracy and efficiency of the present multi-resolution SPH model.

## 1. Introduction

Smoothed Particle Hydrodynamics (SPH) serves as a computational technique extensively employed for simulating fluid dynamics as well as various physical phenomena. Originally introduced by Gingold and Monaghan [19], and Lucy [42] in the 1970s, the SPH model involves discretizing the fluid domain into particles, each endowed with mass, position, velocity, and other pertinent properties. Due to its Lagrangian characteristics, SPH offers distinct advantages compared to conventional Eulerian methods that rely on mesh dependency, and one salient advantage resides in its adaptability to simulate problems characterized by significant non-linearity and large deformations [66,61,43,68,38].

However, the SPH method presently suffers from some drawbacks, e.g., more expensive computational cost compared with classical mesh-based methods. The SPH model needs complex mathematical calculation among particles and their numerous neighboring

\* Corresponding author at: Ocean College, Zhejiang University, Zhoushan, Zhejiang 316021, China.  
E-mail address: [siming.zheng@zju.edu.cn](mailto:siming.zheng@zju.edu.cn) (S. Zheng).

particles, often involving interpolation with tens or even hundreds of them. Moreover, there is either a need for small time steps (for the weakly compressible SPH (WCSPH) model) or pressure Poisson equation solutions (for the Incompressible SPH (ISPH) model) [11,9,80]. High-resolution demands in some conditions mean simulating vast numbers of particles to accurately depict intricate fluid behaviors, leading to heightened computational expenses. These challenges amplify when extended to three-dimensional (3D) problems. Many strategies are used to improve computational efficiency in SPH, including coupling SPH with more efficient models [63,2,51,79] and computer hardware acceleration [12,53,21,80,76].

Another feasible approach to improve the computational efficiency is the development of multi-resolution models. Multi-resolution SPH models achieve local resolution adaptivity by using different numbers of particles at different particle densities, thus using more particles where high resolution is needed and fewer particles in relatively stable areas. This strategy can significantly reduce the total number of particles, lower computational complexity, and provide more efficient simulations in complex fluid dynamics phenomena. Feldman and Bonet [16] introduced a particle-splitting approach in SPH that conserves both mass and energy, marking a significant advancement towards adaptive particle refinement. It has been applied to various problems, including viscous flows and shallow waters, and later refined by Reyes López et al. [54] to conserve density rate changes. However, these methods lack a derefinement procedure for eliminating or clustering particles when they become unnecessary, which could enhance algorithmic efficiency. In two-dimensional (2D) simulations, Barcarolo et al. [4], Sun et al. [58], Chiron et al. [8] introduced a technique where each coarse (parent) particle is subdivided into four smaller (child) particles while retaining the original parent particle. This process effectively minimizes the number of refined particles and simplifies derefinement by enabling the reactivation of parent particles and the removal of child particles. This technique has been successfully applied in multi-resolution  $\delta+$ -SPH simulations, and an enhanced level of control over the smoothing length was reported [58]. Further refinement of this methodology has been made by Chiron et al. [8], who incorporated principles of adaptive mesh refinement to establish an APR approach. Greater precision in interactions and property interpolation between parent and child particles in transitional zones was achieved through this modification. Hu et al. [22] introduced a multi-resolution SPH approach that directly couples subdomains of varying resolutions without requiring an overlap region. To ensure methodological accuracy, they employed a second-order consistent discretization of spatial differential operators. Dynamic particle refinement and coarsening are integrated via splitting and merging mechanisms. An alternative approach to particle splitting for multi-phase fluid simulations was proposed by Yang and Kong [71], Yang et al. [72]. Efficient handling of interface resolution through a single parameter was demonstrated in multi-phase cases, although limitations exist in terms of its accuracy and applicability to 3D scenarios. Zhang et al. [74,75] presented an SPH model designed specifically for fluid-structure interaction and addressing multiphase challenges through the utilization of multi-resolution techniques. Distinct smoothing lengths and time intervals were utilized to offer varying degrees of spatial and temporal resolution, specifically for the multi-phase fluids and structures involved. Khayyer et al. [30] developed an Enhanced Multi-resolution ISPH-SPH solver for simulating fluid-structure interaction, incorporating adaptive resolution schemes without numerical stabilizing terms. Their work demonstrated accurate reproduction of various benchmark tests, showing reduced computational costs while maintaining stability and consistency compared to other particle-based FSI solvers. Lüthi et al. [44] introduced a fully adaptive SPH framework for efficient 3D melt pool simulations in Additive Manufacturing, achieving a five fold speed increase through dynamic refinement-coarsening and a novel sorting algorithm. Their findings underscored the importance of careful particle refinement criteria, dynamic particle loading for memory management, and the significant impact of free-surface and wetting forces on simulation reliability and stability. Gao et al. [17,18] developed a method called Block-based Adaptive Particle Refinement (BAPR), enabling adaptive and focused local refinement. The ‘activation status’ of block arrays determines refinement zones, and a regularization technique ensures the isotropic distribution of new particles in activated blocks. Recently, Ricci et al. [55] developed a multi-resolution SPH algorithm based on a domain decomposition. Buffer layers are applied to the interface of different resolution subdomains. The conversion of particle resolutions is realized through the mass change of the buffer layers. In addition, there are some other multi-resolution SPH models and their applications [39,23,46,70,77].

Some of the above models incorporate overlapping regions to connect subdomains of varying resolutions, each equipped with its distinct smoothing length, resulting in significant computational cost reduction. Nevertheless, this approach poses challenges when it comes to dynamically refining subdomains as the continuous identification of overlapping zones becomes imperative. Non-overlapping multi-resolution models offer a means to circumvent these challenges; however, they do introduce numerical discretization errors in the vicinity of interfaces [22]. This is due to the requirement of accommodating particles of different resolutions within the support domain of an SPH particle. It is essential to do a comprehensive analysis and discussion of the SPH discrete format at the multi-resolution interface. Moreover, the published SPH models described in the previous paragraph employ a global time step. In numerical simulations, choosing an appropriate time step is a critical decision that directly affects both accuracy and efficiency. This choice encompasses the satisfaction for stability criteria aligned with numerical methods, adherence to Courant-Friedrichs-Lowy (CFL) conditions, factoring in physical phenomena and temporal scales, attaining requisite accuracy levels, and mindful consideration of computational resources [64,65]. The global time step within the multi-resolution model is established by the smallest time step within the subdomain, ensuring holistic stability across the computational domain. Numerical simulations encounter significant constraints with global time stepping, particularly in the context of multi-scale or intricate systems. Its one-size-fits-all nature hampers computational efficiency, leading to wasted resources in slower-changing regions. It lacks the flexibility to adapt to varying requirements across different areas, making it less effective for capturing local dynamics. Additionally, in parallel computing, efficiency is constrained as all tasks need to synchronize with the slowest one. These issues collectively make global time stepping a low efficient choice for complex simulations. In contrast, local time stepping offers several advantages, particularly in multi-scale and complex simulations [25,13,35,36,27]. By enabling various regions to function at their respective optimal time scales, it improves computational efficiency and minimizes overhead in slow dynamic areas. Furthermore, local time stepping can improve parallel

computing performance, as tasks can progress independently based on their own time constraints rather than waiting for the slowest task.

In this study, we introduce a new multi-resolution SPH modeling approach based on the Message Passing Interface (MPI)-based parallel SPH framework. Regarding MPI-based parallel frameworks for the SPH method, several advancements have been made. Ihmsen et al. [24] proposed efficient parallel neighborhood inquiries and explored solutions to optimize time steps. Cherfils et al. [7] developed JOSEPHINE, implementing dynamic loading balancing for unstable free-surface flows. [12] introduced DualSPHysics, leveraging GPU parallel processing for significant acceleration [14]. Oger et al. [53] discussed parallelizing particle methods on distributed memory, achieving good efficiencies in massive particle discretization. Guo et al. [21] proposed a massively parallel scheme for free-surface flow simulation using ISPH with a Hilbert space-filling curve. Ji et al. [26] presented a multi-resolution parallel framework for large-scale SPH simulations, integrating dynamic load balancing with CVP partitioning. Additionally, Zhu et al. [80] developed an MPI-based parallel SPH framework with a dynamic load balance strategy based on computational particle numbers and running time, effectively handling complex flows. More recently, Liu et al. [34] contributed by developing an MPI-based parallelization framework for SPH, enhancing scalability and efficiency in simulating extreme mechanics problems through optimized memory management, communication strategies, and portability testing, paving the way for future advancements in load balancing and heterogeneous parallelism.

The present model is developed using the MPI-based parallel SPH framework proposed by Zhu et al. [80]. Computational domain is divided into multiple subdomains, each processed on different cores. Our innovation lies in integrating this MPI-based strategy with a multi-resolution concept, necessitating different resolutions across varying MPI subdomains. This integration offers several benefits. MPI effectively manages communication between areas with different resolutions, simplifying the complexity of multi-resolution model development. Programmatic independence is granted to each subdomain, enabling customized model configurations tailored to specific resolution sectors through localized model adaptation within particular resolution subdomains. Moreover, the absence of overlapping buffer zones between resolution areas reduces MPI communication overhead. Particle resolution at the interfaces of the subdomains is achieved through dynamic merging and splitting strategies. We also explore the effects of different discretization techniques on the error distribution in multi-resolution models. Additionally, we introduce a local time-stepping strategy where different resolution sub-domains utilize varying time intervals. Large time steps are employed in low-resolution subdomains. This approach, when coupled with dynamic load balancing in the MPI-based parallel framework, helps to reduce overall computational costs.

The remainder of this paper is organized as follows. The governing equations are described in Section 2. The multi-resolution SPH model is present in Section 3. Validations of the present model are described in Section 4. Finally, conclusions are drawn in Section 5.

## 2. Governing equations

We follow the classical SPH approach of weak compressibility usually adopted in SPH numerical schemes (e.g., see Colagrossi and Landrini [10]), and mass and momentum conservation can be written in Lagrangian form as:

$$\begin{cases} \frac{dp}{dt} = -\rho \nabla \cdot \mathbf{u}, \\ \frac{d\mathbf{u}}{dt} = -\frac{\nabla p}{\rho} + \nu \nabla^2 \mathbf{u} + \mathbf{f}_{body}, \end{cases} \quad (1)$$

where  $\mathbf{u}$ ,  $t$ ,  $\rho$ ,  $p$  and  $\nu$  denote velocity vector, time, density, pressure and kinematic viscosity, respectively.  $\mathbf{f}_{body}$  is body force per unit mass acting on the fluid.

The pressure  $p$  is calculated directly from the density using a linear equation of state

$$p = (\rho - \rho_0)c^2, \quad (2)$$

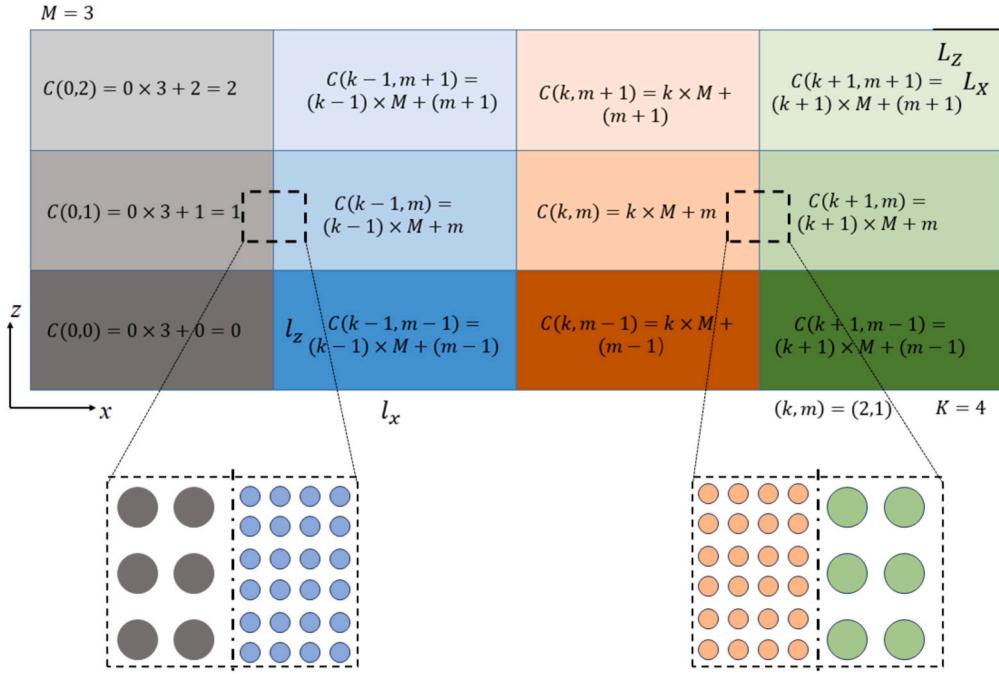
where  $c$  and  $\rho_0$  are the numerical speed of sound and initial fluid density, respectively.

## 3. Multi-resolution SPH model

The current multi-resolution SPH model is based on the MPI-based parallel framework proposed by Zhu et al. [80] as shown in Fig. 1. It is solved in different subdomains using different resolutions. The index sort method is used to get grid index  $C(k, m)$  for subdomains. The location of the subdomains and adjacent subdomains can be determined using the index. The fine and coarse subdomains can therefore be well determined. Furthermore, varying time intervals are employed for distinct resolution subdomains. In this section, we will introduce the current SPH model in terms of spatial discretization, particle shifting algorithm, splitting and merging of SPH particles, and local time stepping scheme.

### 3.1. Spatial discretization

The classical SPH method is based on gradient approximations of a function  $f(\mathbf{r})$  which are achieved by solving the variables with a kernel function  $W$



**Fig. 1.** Sketch for MPI-based parallel multi-resolution SPH model.

$$\nabla f(\mathbf{r}) = \int_{\Omega} f(\mathbf{r}') \nabla W(\mathbf{r} - \mathbf{r}') d\mathbf{r}', \quad (3)$$

where  $\nabla W(\mathbf{r} - \mathbf{r}')$  is the gradient of the kernel function  $W$ . The Gaussian kernel is used as

$$W_G = \begin{cases} \frac{1}{(\pi h)^{dim}} e^{-q^2}, & 0 \leq q \leq 3, \\ 0, & \text{otherwise,} \end{cases} \quad (4)$$

where  $h = 1.2(dx_0)$  ( $dx_0$  denotes initial coarse particle spacing),  $dim$  is 2 and 3 for 2D problems and 3D problems, respectively.  $q = \frac{|r_{ij}|}{h}$ ,  $|r_{ij}|$  denotes the distance between particle  $i$  and  $j$ . The Gaussian kernel does not have a compact support. Therefore, the Gaussian kernel with  $3h$  as the cut-off limit has been used for the cases studied.

The discrete SPH scheme uses a set of interpolation particles carrying physical properties to discretize the fluid domain, as follows [37,40]:

$$\nabla f_i = \sum_j f_j \nabla_i W_{ij} V_j, \quad (5)$$

where  $f_i$  and  $f_j$  denote the mathematical abbreviations of  $f(\mathbf{r}_i)$  and  $f(\mathbf{r}_j)$ , respectively.  $\nabla_i W_{ij}$  is the mathematical abbreviations of  $\nabla_i W(\mathbf{r}_i - \mathbf{r}_j)$ .  $V_j$  is the volume of particle  $j$ .

In the original form of SPH, the gradient of the function  $f$  was discretized as [37,40]:

$$\nabla f_i = \sum_j (f_j + f_i) \nabla_i W_{ij} V_j, \quad (6)$$

$$\nabla f_i = \sum_j (f_j - f_i) \nabla_i W_{ij} V_j. \quad (7)$$

A correction matrix  $\mathbf{L}$  could also be introduced to guarantee the exact gradient of function [6,52,40] as

$$\nabla f_i = \sum_j (f_j + f_i) \mathbf{L}_i \nabla_j W_{ij} V_j, \quad (8)$$

$$\nabla f_i = \sum_j (f_j - f_i) \mathbf{L}_i \nabla_j W_{ij} V_j, \quad (9)$$

where

$$\mathbf{L}_i = \left[ \sum_j (\mathbf{r}_{ij}) \otimes \nabla_i W_{ij} V_j \right]^{-1}. \quad (10)$$

Eqs. (6), (7), (9) and (10) are used in many multi-resolution SPH models [16,54,4,62,22].

For free surface flow (or a truncated boundary), using Eq. (7) and Eq. (9) will cause the particles to experience unphysical forces toward the outside of the boundary. This is due to the discretization error caused by truncation. Using Eq. (6) or Eq. (9) will generate a ‘force’ that offsets it. For the particles near the free surface, i.e., those whose neighbor particles include free surface particles, Eq. (6) or Eq. (9) is used to calculate the gradient. Free surface particles are the outermost particles of the boundary, determined using Wang et al. [67]’ method. In subsequent sections, we discuss the accuracy of these discrete formats for the present multi-resolution SPH model.

The Laplacian of the function  $f$  in the SPH method is discretized as

$$\nabla^2 f_i = 2 \sum_j \left( \frac{f_i - f_j}{r_{ij}} \right) e_{ij} \nabla_i W_{ij} V_j, \quad (11)$$

where  $e_{ij}$  denotes unit vector between particle  $i$  and  $j$ . The above format is proven to have only 0th order accuracy [15,60]. Considering the correction matrix  $\mathbf{L}_i$ , we can get

$$\nabla^2 f_i = 2 \sum_j \left( \frac{f_i - f_j}{r_{ij}} \right) e_{ij} \mathbf{L}_i \nabla_i W_{ij} V_j. \quad (12)$$

To improve the accuracy of the Laplacian of the function  $f$ , a third-order Taylor expansion is introduced

$$f_j = f_i + r_{ij} f'_i + \frac{1}{2} r_{ij}^2 f''_i + O(h^3). \quad (13)$$

Multiplying both sides by a  $\nabla(\nabla_i W_{ij})$ , and integrating over the domain yields

$$\sum_j (f_j - f_i) \nabla(\nabla_i W_{ij}) V_j \approx \sum_j (r_j - r_i) \nabla f_i \nabla(\nabla_i W_{ij}) V_j + \sum_j \frac{1}{2} (r_j - r_i)^2 \nabla^2 f_i \nabla(\nabla_i W_{ij}) V'_j. \quad (14)$$

According to the above equations, the Laplacian of the function  $f$  in the SPH method can be calculated as

$$\nabla^2 f_i = \frac{\sum_j (f_j - f_i) \nabla(\nabla_i W_{ij}) V_j - \sum_j (r_j - r_i) \nabla f_i \nabla(\nabla_i W_{ij}) V_j}{\sum_j \frac{1}{2} (r_j - r_i)^2 \nabla(\nabla_i W_{ij}) V_j}. \quad (15)$$

A sufficiently smooth kernel function in the case of regularly distributed particles [40], one can get

$$\sum_j \frac{1}{2} (r_j - r_i)^2 \nabla(\nabla_i W_{ij}) V_j \approx 1. \quad (16)$$

By substituting Eq. (16) into Eq. (15), we can ignore the denominator phase of Eq. (15). After introducing

$$\nabla(\nabla_i W_{ij}) = -2 \frac{(e_{ij}) \nabla_i W_{ij}}{r_{ij}}, \quad (17)$$

one can get

$$\nabla^2 f_i = 2 \sum_j \frac{f_i - f_j - r_{ij} \nabla f_i}{|r_{ij}|^2} r_{ij} \nabla_i W_{ij} V_j, \quad (18)$$

where  $\nabla f_i$  can be solved by using Eq. (7). For free surface flow (or a truncated boundary), Eq. (6) is used. If we consider the correction matrix, we can have

$$\nabla^2 f_i = 2 \sum_j \frac{f_i - f_j - r_{ij} \nabla f_i}{|r_{ij}|^2} r_{ij} \mathbf{L}_i \nabla_i W_{ij} V_j, \quad (19)$$

where  $\nabla f_i$  can be solved by using Eq. (9) since the correction matrix  $\mathbf{L}_i$  is used for  $\nabla^2 f$ . For free surface paritces, Eq. (8) is applied.

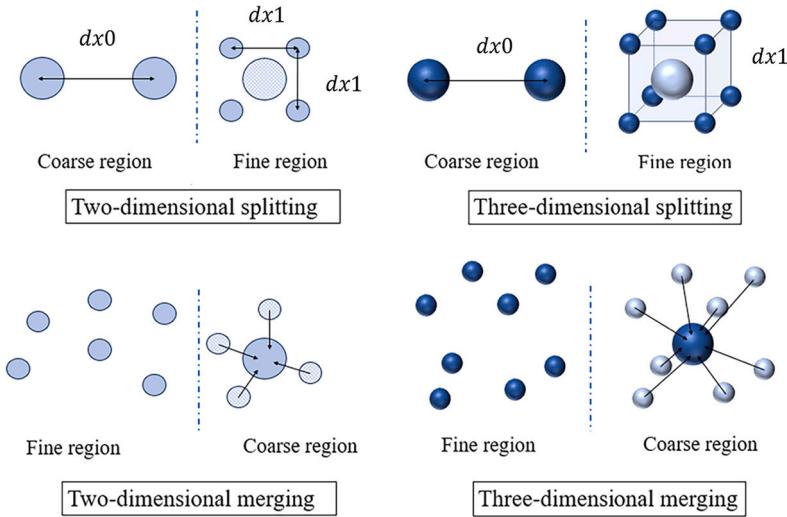
To ensure numerical stability, a numerical dissipation  $\delta$ -term and an artificial viscosity term are added to the discrete governing equations. The  $\delta$  term has been proven to reduce pressure oscillation problems in a large number of applications [49,73,59,45,23]. The numerical dissipation  $\delta$ -term and the artificial viscosity can be written as:

$$\delta h c \sum_j \Psi_{ij} \mathbf{L}_i \nabla_i W_{ij} V_j, \quad (20)$$

$$\sum_j \alpha h c \Pi_{ij} \mathbf{L}_i \nabla_i W_{ij} V_j, \quad (21)$$

$$\begin{cases} \Psi_{ij} = 2(\rho_j - \rho_i) \frac{\mathbf{r}_j - \mathbf{r}_i}{|\mathbf{r}_j - \mathbf{r}_i|^2} - (\langle \nabla \rho \rangle_i^L + \langle \nabla \rho \rangle_j^L), \\ \langle \nabla \rho \rangle_i^L = \sum_j (\rho_j - \rho_i) \mathbf{L}_i \nabla_i W_{ij} V_j, \end{cases} \quad (22)$$

$$\Pi_{ij} = \frac{(\mathbf{u}_j - \mathbf{u}_i) \cdot (\mathbf{r}_j - \mathbf{r}_i)}{|\mathbf{r}_j - \mathbf{r}_i|^2}, \quad (23)$$



**Fig. 2.** 2D and 3D splitting and merging process. ( $dx_0$  and  $dx_1$  denote coarse particle spacing and fine particle spacing, respectively.)

where  $\delta$  and  $\alpha$  are 0.1 and 0.01, respectively. Finally, the governing equations can be discretized as

$$\begin{cases} \frac{dp}{dt} = -\rho \nabla \cdot \mathbf{u} + \delta hc \sum_j \Psi_{ij} \mathbf{L}_i \nabla_i W_{ij} V_j, \\ \frac{d\mathbf{u}}{dt} = -\frac{\nabla p}{\rho} + \nu \nabla^2 \mathbf{u} + \sum_j \alpha hc \Pi_{ij} \mathbf{L}_i \nabla_i W_{ij} V_j + \mathbf{f}_{body}. \end{cases} \quad (24)$$

Ghost particle method [47] is used as boundary conditions. For the present model, ghost particles in the refined region also need to be refined.

### 3.2. Particle shifting

In the SPH method, the movement of particles can lead to uneven particle distribution, resulting in particle overlapping or gaps. This irregular distribution might affect the method's precision and steadiness. One way to address this is by slightly adjusting the particles to ensure a more uniform distribution. This particle adjustment technique was discussed in [69], and subsequent research [33,57] validated its effectiveness in maintaining consistent particle arrangement and numerical stability. Moreover, this approach has been improved for free surface flows [28,58]. We follow the improved approach by Wang et al. [67], in which the shifting vector  $\delta \mathbf{r}_i$  is determined as

$$\delta \mathbf{r}_i = \begin{cases} 0 & \text{if } i \in F_p \\ k_{cf} \cdot Ma \cdot 2h^2 \sum_j V_j \chi_{ij} \mathbf{L}_i \nabla_i W_{ij} & i \in |\mathbf{r}_{ij}| < l_a \text{ if } i \in V_p \\ k_{cf} \cdot Ma \cdot 2h^2 \sum_j V_j (1 + \chi_{ij}) \mathbf{L}_i \nabla_i W_{ij} & i \in |\mathbf{r}_{ij}| < kh \text{ if } i \in I_p \\ \chi_{ij} = 0.2 \left( \frac{W_{ij}}{W(\Delta x_f)} \right)^4 & k_{cf} = \frac{c}{h} \Delta t \end{cases} \quad (25)$$

where  $F_p$  denotes free surface particles,  $V_p$  denotes the free-surface vicinity region and  $I_p$  indicates the interior fluid domain,  $Ma$  is Mach number. To distinguish different particles, please refer to Wang et al. [67].  $l_a$  represents the distance between the particle and its closest free-surface particle.  $\Delta t$  is the time step. Our tests indicate that  $\Delta x_f$  should be the initial coarse particle spacing for all resolution particles (even fine particles).

### 3.3. Particle splitting and merging

As the system evolves over time, particles originating from regions with low resolution may exit the system and migrate into areas with high resolution. In this context, “coarse regions” refer to areas with low levels of detail, while “fine regions” denote zones with fine details and high resolution. Particles situated in coarse regions have large initial spaces between them, whereas particles in fine regions exhibit comparatively small initial spacing. Particles in coarse regions are called “coarse particles”, while those in fine regions are referred to as “fine particles.”

As coarse particles migrate from coarse resolution regions to fine resolution regions, a single coarse particle needs to be split. This entails the removal of the initial coarse particle, concomitant with the introduction of four and eight new fine particles, respectively, for 2D and 3D problems as shown in Fig. 2. The spatial disposition of the fine particles with respect to the original coarse particle is depicted in the diagram. The mass of the coarse particle is evenly distributed among the fine particles. Interpolating at the location

of the original coarse particle provides access to pertinent physical information, encompassing velocity and density attributes. The method of splitting coarse particles into fine particles is shown as Algorithm 1. The splitting process is executed every time step.

---

**Algorithm 1** Procedure for splitting SPH particles.
 

---

- 1: Determine whether the region of the particle is a refined region
  - 2: For every coarse particle in the fine region, add four and eight particles for 2D and 3D problems, respectively, and the relative positions of the particles are shown in Fig. 2. The coarse particle is located at the center of the four and eight particles for 2D and 3D problems. For example, the four new fine particles positions  $\mathbf{r}_{f1}(x_{f1}, y_{f1}) = \mathbf{r}_c(x_c + 0.5 * dx0, y_c + 0.5 * dx0)$ ,  $\mathbf{r}_{f2}(x_{f2}, y_{f2}) = \mathbf{r}_c(x_c + 0.5 * dx0, y_c - 0.5 * dx0)$ ,  $\mathbf{r}_{f3}(x_{f3}, y_{f3}) = \mathbf{r}_c(x_c - 0.5 * dx0, y_c + 0.5 * dx0)$ , and  $\mathbf{r}_{f4}(x_{f4}, y_{f4}) = \mathbf{r}_c(x_c - 0.5 * dx0, y_c - 0.5 * dx0)$ , where  $\mathbf{r}_f$  is the position of fine particle,  $\mathbf{r}_c$  is the position of coarse particle.
  - 3: Velocity and density can be obtained by Taylor expansion interpolation. The gradient of velocity and density of the coarse particles can be solved by Eq. (9). Then the velocity and density of the new fine particles can be calculated based on the distance from coarse particles.
  - 4: Remove old coarse particles.
- 

When fine particles transition from high-resolution regions to coarse regions, they need to coalesce into coarse particles. The mass of a fine particle accounts for one-fourth and one-eighth of the mass of a coarse particle, respectively, for 2D and 3D problems. As a result, the fine particles are required for the amalgamation into a single coarse particle as indicated in Fig. 2. All the fine particles must be present within the coarse region for this merging process. The specifics of the merging procedure are elucidated in Algorithm 2. The merging process is executed every time step.

---

**Algorithm 2** Procedure for merging SPH particles.
 

---

- 1: Determines whether the area is a coarse region.
  - 2: Find four and eight fine particles in the coarse region, respectively, for 2D and 3D problems that have the largest distance less than  $1.5dx1$  ( $dx1$  is the initial fine particle spacing). Those particles are determined to be the particles that can be merged. For example, if there are three fine particles among its neighboring particles for a fine particle in a coarse area, and the distance between these three fine particles is less than  $1.5dx1$ , then we determine that these four particles merge into one coarse particle for a two-dimensional problem.
  - 3: Add one new coarse particle, and the position is the coordinate mean of the position of the old fine particle.
  - 4: The mass of the new coarse particle is four and eight times, respectively, the mass of the old fine particle for 2D and 3D problems. Velocity and density can be obtained by Taylor expansion interpolation. The gradient of velocity and density of the old fine particles can be solved by Eq. (9). Four or eight interpolated velocities and densities at the new coarse position can be obtained. The average of these four or eight interpolated properties is regarded as the physical properties of the new coarse particles.
  - 5: Remove the old fine particles.
- 

An exceptional situation arises when certain fine particles cannot amalgamate due to a scarcity of particles within the group, specifically fewer than four particles for 2D problems and eight for 3D problems. Unmergeable fine particles can be deleted directly in the coarse area, such as the dam break cases in Section 4. But this will lead to non-conservation of system mass. Unmergeable fine particles can also be retained in the coarse region, as in the remaining examples in Section 4. The results show that this approach does not cause significant errors in the results. Since the number of particles that are not merged is generally very small, we recommend keeping those unmergeable fine particles.

### 3.4. Local time stepping scheme

The main idea of the local time stepping approach is to use different time steps in different subdomains (parallel models) to reduce computational expenditure. In the context of the SPH model, adherence to the CFL condition for numerical stability governs the selection of time steps. Introducing various particle sizes (potentially influenced by factors like heightened velocities within localized regions) introduces the potential for divergent time step choices across different subdomains. Nevertheless, implementing this approach necessitates specialized measures to ensure the overall computational domain stability, even when operating with disparate time steps.

We introduce a Local Time Stepping scheme approach within the framework of the present SPH multi-resolution model. Let  $\Delta t_f$  denote the time step for particles within the fine subdomains, while particles in coarse subdomains progress with a time step of  $\Delta t_c = 2\Delta t_f$ . Uniformly applied throughout the computational domain, the fourth-order Runge-Kutta (RK4) integration method governs the time integration process. In cases where one time step advances within the coarse regions, two consecutive time steps are undertaken within the fine regions to ensure synchronization in terms of physical time, as illustrated in Fig. 3.

At the initial moment, as shown in Fig. 3(a), both coarse particles and fine particles are at the same physical time. Coarse particles and fine particles undergo simultaneous time integration, as illustrated in Fig. 3(b). However, due to different time step sizes, after one time integration step, the coarse particles are now at time  $t + \Delta t_c$ , while the fine particles are at time  $t + \Delta t_f$ . Consequently, the fine particles require an additional time integration step, as depicted in Fig. 3(c). The local time step strategy only needs to complete three Runge-Kutta time integrals. It differs from the global time step strategy, which requires two integrals in the coarse and fine regions.

A particle near the subdomain boundary may require particle information from adjacent subdomains. If the time steps of two subdomains are different, how to obtain the physical properties of the corresponding time steps (including intermediate time level) to solve the governing equation is the main problem.

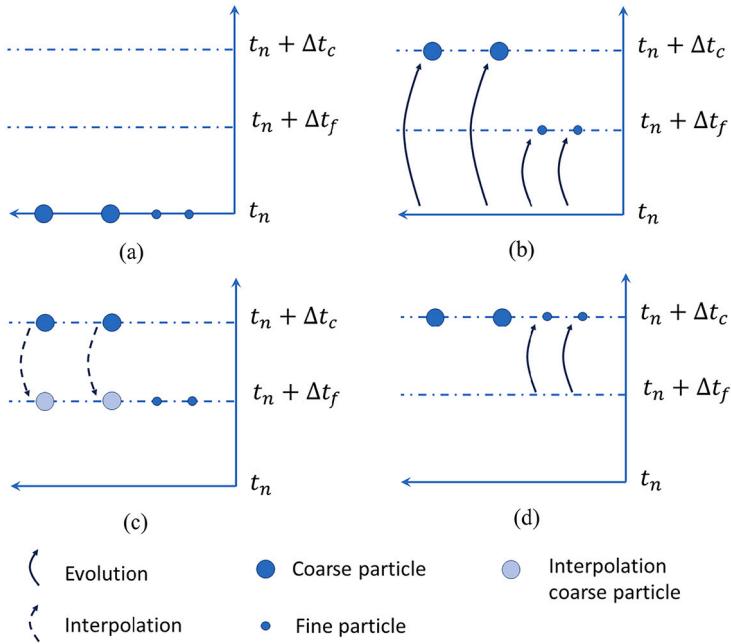


Fig. 3. Scheme of local time stepping.

The time-integral scheme of the RK4 is used. The general explicit RK4 time integration scheme with steps can be written as

$$m_1 = F(\mathbf{U}^n), \quad (26)$$

$$m_2 = F\left(\mathbf{U}^n + \frac{1}{2}\Delta t m_1\right), \quad (27)$$

$$m_3 = F\left(\mathbf{U}^n + \frac{1}{2}\Delta t m_2\right), \quad (28)$$

$$m_4 = F\left(\mathbf{U}^n + \Delta t m_3\right), \quad (29)$$

$$\mathbf{U}^{n+1} = \mathbf{U}^n + \frac{1}{6}\Delta t(m_1 + 2m_2 + 2m_3 + m_4), \quad (30)$$

where  $\mathbf{U}^n$  denotes the physical properties (density, velocity and position in the present SPH model) at  $n^{th}$  time step.  $F(\mathbf{U}^n)$  is the physical properties of time  $t$ , which can be written as

$$\frac{\partial \mathbf{U}^n}{\partial t} = F(\mathbf{U}^n). \quad (31)$$

Taking the above equation into Eqs. (26)-(29), we get

$$m_1 = \frac{\partial \mathbf{U}^n}{\partial t}, \quad (32)$$

$$m_2 = \frac{\partial \mathbf{U}^n}{\partial t} + \frac{1}{2}\Delta t \frac{\partial^2 \mathbf{U}^n}{\partial t^2}, \quad (33)$$

$$m_3 = \frac{\partial \mathbf{U}^n}{\partial t} + \frac{1}{2}\Delta t \frac{\partial^2 \mathbf{U}^n}{\partial t^2} + \frac{1}{4}\Delta t^2 \frac{\partial^3 \mathbf{U}^n}{\partial t^3}, \quad (34)$$

$$m_4 = \frac{\partial \mathbf{U}^n}{\partial t} + \Delta t \frac{\partial^2 \mathbf{U}^n}{\partial t^2} + \frac{1}{2}\Delta t^2 \frac{\partial^3 \mathbf{U}^n}{\partial t^3} + \frac{1}{4}\Delta t^3 \frac{\partial^4 \mathbf{U}^n}{\partial t^4}. \quad (35)$$

In Fig. 3(a), the coarse and fine particles are at the same time. At the interface of regions with different resolutions, the derivatives of the particles' physical quantities with respect to time are consistent, even though the time steps of the coarse and fine particle regions are different. Thus, the following results can be obtained [35]:

$$\left\{ \frac{\partial \mathbf{U}_c}{\partial t}, \frac{\partial^2 \mathbf{U}_c}{\partial t^2}, \frac{\partial^3 \mathbf{U}_c}{\partial t^3}, \frac{\partial^4 \mathbf{U}_c}{\partial t^4}, \dots \right\}^n = \left\{ \frac{\partial \mathbf{U}_f}{\partial t}, \frac{\partial^2 \mathbf{U}_f}{\partial t^2}, \frac{\partial^3 \mathbf{U}_f}{\partial t^3}, \frac{\partial^4 \mathbf{U}_f}{\partial t^4}, \dots \right\}^n, \quad (36)$$

where labels  $c$  and  $f$  represent the coarse area and fine area, respectively. Substituting Eq. (36) into Eqs. (32)-(35), the relationship of intermediate  $m$  values under different time step regions is obtained. Firstly, the intermediate  $m$  value at time step  $\Delta t_f$  in the

refined region is transformed into the intermediate  $m$  value at time step  $\Delta t_c$  in the coarse region. The intermediate time level in the refined region for use in time advancing in the coarse region is:

$$m_1^{f \rightarrow c} = m_1^f, \quad (37)$$

$$m_2^{f \rightarrow c} = \left(1 - \frac{\Delta t_c}{\Delta t_f}\right)m_1^f + \frac{\Delta t_c}{\Delta t_f}m_2^f, \quad (38)$$

$$m_3^{f \rightarrow c} = \left(1 - \frac{\Delta t_c}{\Delta t_f}\right)m_1^f + \left(\frac{\Delta t_c}{\Delta t_f} - \frac{\Delta t_c^2}{\Delta t_f^2}\right)m_2^f + \frac{\Delta t_c^2}{\Delta t_f^2}m_3^f, \quad (39)$$

$$m_4^{f \rightarrow c} = \left(1 - 2\frac{\Delta t_c}{\Delta t_f} + \frac{\Delta t_c^3}{\Delta t_f^3}\right)m_1^f + 2\left(\frac{\Delta t_c}{\Delta t_f} - \frac{\Delta t_c^2}{\Delta t_f^2}\right)m_2^f + 2\left(\frac{\Delta t_c^2}{\Delta t_f^2} - \frac{\Delta t_c^3}{\Delta t_f^3}\right)m_3^f + \frac{\Delta t_c^3}{\Delta t_f^3}m_4^f, \quad (40)$$

where  $f \rightarrow c$  denotes intermediate  $m$  values in fine regions transferred to intermediate  $m$  values in coarse regions. Conversely, the intermediate time level in the coarse region for use in time advancing in the refined region is:

$$m_1^{c \rightarrow f} = m_1^c, \quad (41)$$

$$m_2^{c \rightarrow f} = \left(1 - \frac{\Delta t_f}{\Delta t_c}\right)m_1^c + \frac{\Delta t_f}{\Delta t_c}m_2^c, \quad (42)$$

$$m_3^{c \rightarrow f} = \left(1 - \frac{\Delta t_f}{\Delta t_c}\right)m_1^c + \left(\frac{\Delta t_f}{\Delta t_c} - \frac{\Delta t_f^2}{\Delta t_c^2}\right)m_2^c + \frac{\Delta t_f^2}{\Delta t_c^2}m_3^c, \quad (43)$$

$$m_4^{c \rightarrow f} = \left(1 - 2\frac{\Delta t_f}{\Delta t_c} + \frac{\Delta t_f^3}{\Delta t_c^3}\right)m_1^c + 2\left(\frac{\Delta t_f}{\Delta t_c} - \frac{\Delta t_f^2}{\Delta t_c^2}\right)m_2^c + 2\left(\frac{\Delta t_f^2}{\Delta t_c^2} - \frac{\Delta t_f^3}{\Delta t_c^3}\right)m_3^c + \frac{\Delta t_f^3}{\Delta t_c^3}m_4^c, \quad (44)$$

where  $c \rightarrow f$  denotes intermediate  $m$  values in coarse region transferred to intermediate  $m$  values in fine regions. The above formula constructs the relationship between the intermediate  $m$  values of different time step regions. For the region with different time steps, the RK4 time integration of the coarse regions requires information from the fine regions to complete the SPH discretization (solution of the governing equation) at intermediate time levels. For example, the particle information in the fine particle region at intermediate time level  $m_2$  needs to be updated by the transferred intermediate value  $m_2^{c \rightarrow f}$  before information exchange among subdomains.

After completing the first RK4 integration (Fig. 3(b)), the coarse particle region reaches  $\Delta t_n + \Delta t_c$ , and the fine particle region reaches  $\Delta t_n + \Delta t_f$ . Therefore, the fine particle needs to perform another RK4 time integral from  $\Delta t_n + \Delta t_f$  to  $\Delta t_n + \Delta t_c$ . At this point, we need the value of the coarse grain region in  $\Delta t_n + \Delta t_f$  as the initial value in the second RK4 integration. The physical properties of coarse particles at  $\Delta t_n + \Delta t_f$  (as shown in Fig. 3(c)) can be written as

$$U_c^{n+0.5} = U_c^n + \frac{1}{6}\Delta t_f \left(m_1^{c \rightarrow f} + 2m_2^{c \rightarrow f} + 2m_3^{c \rightarrow f} + m_4^{c \rightarrow f}\right), \quad (45)$$

where  $U_c^{n+0.5}$  denotes the interpolated physical properties of coarse region at the  $\Delta t_c + \Delta t_f$ .

After that, the fine regions need to perform an RK4 time integration from  $t_n + t_f$  to  $t_n + t_c$ . The integral solution for fine particles requires information from coarse particles at the intermediate time level. However, the coarse-particle region has not been numerically calculated at this moment. In this work, intermediate time information of coarse particles at  $t_n + t_f$  is obtained by utilizing a linear interpolation of Eqs. (26)-(29) at the intermediate time level. Specifically, we have

$$m_1^{f+0.5} = m_1^f + \Delta t_f \frac{\partial m_1^f}{\partial t}, \quad (46)$$

$$m_2^{f+0.5} = m_2^f + \Delta t_f \frac{\partial m_2^f}{\partial t}, \quad (47)$$

$$m_3^{f+0.5} = m_3^f + \Delta t_f \frac{\partial m_3^f}{\partial t}, \quad (48)$$

$$m_4^{f+0.5} = m_4^f + \Delta t_f \frac{\partial m_4^f}{\partial t}, \quad (49)$$

where label  $f + 0.5$  denotes intermediate  $m$  values in coarse region from  $t_n + t_f$  to  $t_n + t_c$ .

Considering Eqs. (32)-(35), Eqs. (46)-(49) can be rewritten as

$$m_1^{f+0.5} = \frac{\partial \mathbf{U}^f}{\partial t} + \Delta t_f \frac{\partial^2 \mathbf{U}^f}{\partial t^2}, \quad (50)$$

$$m_2^{f+0.5} = \frac{\partial \mathbf{U}^f}{\partial t} + \frac{3}{2}\Delta t_f \frac{\partial^2 \mathbf{U}^f}{\partial t^2} + \frac{1}{2}\Delta t_f^2 \frac{\partial^3 \mathbf{U}^f}{\partial t^3}, \quad (51)$$

$$m_3^{f+0.5} = \frac{\partial \mathbf{U}^f}{\partial t} + \frac{3}{2} \Delta t_f \frac{\partial^2 \mathbf{U}^f}{\partial t^2} + \frac{3}{4} \Delta t_f^2 \frac{\partial^3 \mathbf{U}^f}{\partial t^3} + \frac{1}{4} \Delta t_f^3 \frac{\partial^4 \mathbf{U}^f}{\partial t^4}, \quad (52)$$

$$m_4^{f+0.5} = \frac{\partial \mathbf{U}^f}{\partial t} + 2 \Delta t_f \frac{\partial^2 \mathbf{U}^f}{\partial t^2} + \frac{3}{2} \Delta t_f^2 \frac{\partial^3 \mathbf{U}^f}{\partial t^3} + \frac{1}{4} \Delta t_f^3 \frac{\partial^4 \mathbf{U}^f}{\partial t^4}. \quad (53)$$

According to Eqs. (32)-(35), we may have

$$\frac{\partial \mathbf{U}^f}{\partial t} = \frac{\partial \mathbf{U}^c}{\partial t} = m_1^c, \quad (54)$$

$$\frac{\partial^2 \mathbf{U}^f}{\partial t^2} = \frac{\partial^2 \mathbf{U}^c}{\partial t^2} = m_1^c \frac{-2}{\Delta t_c} + m_2^c \frac{2}{\Delta t_c}, \quad (55)$$

$$\frac{\partial^3 \mathbf{U}^f}{\partial t^3} = \frac{\partial^3 \mathbf{U}^c}{\partial t^3} = m_2^c \frac{-4}{\Delta t_c^2} + m_3^c \frac{4}{\Delta t_c^2}, \quad (56)$$

$$\frac{\partial^4 \mathbf{U}^f}{\partial t^4} = \frac{\partial^4 \mathbf{U}^c}{\partial t^4} = m_1^c \frac{4}{\Delta t_c^3} + m_3^c \frac{-8}{\Delta t_c^3} + m_4^c \frac{4}{\Delta t_c^3}. \quad (57)$$

The reason to consider Eqs. (54)-(57) is to avoid solving for the time gradient, and to obtain the interpolation through  $m$  value. Combining Eqs. (50)-(53) and Eqs. (54)-(57), we could have

$$m_1^{c+0.5 \rightarrow f+0.5} = (1 - 2 \frac{t_f}{t_c}) m_1^c + 2 \frac{t_f}{t_c} m_2^c, \quad (58)$$

$$m_2^{c+0.5 \rightarrow f+0.5} = (1 - 3 \frac{t_f}{t_c}) m_1^c + (3 \frac{t_f}{t_c} - 2 \frac{t_f^2}{t_c^2}) m_2^c + 2 \frac{t_f^3}{t_c^3} m_3^c, \quad (59)$$

$$m_3^{c+0.5 \rightarrow f+0.5} = (1 - 3 \frac{t_f}{t_c} + \frac{t_f^3}{t_c^3}) m_1^c + (3 \frac{t_f}{t_c} - 3 \frac{t_f^2}{t_c^2}) m_2^c + (3 \frac{t_f^2}{t_c^2} - 2 \frac{t_f^3}{t_c^3}) m_3^c + \frac{t_f^3}{t_c^3} m_4^c, \quad (60)$$

$$m_4^{c+0.5 \rightarrow f+0.5} = (1 - 4 \frac{t_f}{t_c} + \frac{t_f^3}{t_c^3}) m_1^c + (4 \frac{t_f}{t_c} - 6 \frac{t_f^2}{t_c^2}) m_2^c + (6 \frac{t_f^2}{t_c^2} - 2 \frac{t_f^3}{t_c^3}) m_3^c + \frac{t_f^3}{t_c^3} m_4^c, \quad (61)$$

where label  $c + 0.5 \rightarrow f + 0.5$  intermediate  $m$  values in coarse region transferred to intermediate  $m$  values in fine regions at  $t_n + t_f$ . From  $t_n + t_f$  to  $t_n + t_c$ , the coarse particles do not participate in the solution of the governing equation, and the intermediate  $m$  value is obtained by numerical interpolation. Since the time step is always constant, the above template is fixed throughout the calculation process.

Finally, we summarize the local time step strategy calculation process from  $t_n + t_c$  as follows:

1. Performing the first RK4 time integration, coarse particles and fine particles simultaneously. The first transferred intermediate  $m^{c \rightarrow f}$  and  $m^{f \rightarrow c}$  values relative to the coarse particle and the fine particle are calculated by Eqs. (37)-(40) and Eqs. (41)-(44), respectively. Based on the intermediate values transferred, we can get the particle information that needs to be transferred to different subdomains (with different time steps).

2. The value of coarse particle region at  $t_n + t_f$  is calculated according to Eq. (45).

3. Performing the second RK4 time integration in the refined region. The particle information in the coarse region that needs to be transferred is obtained by the intermediate  $m^{c+0.5 \rightarrow f+0.5}$  using Eqs. (58)-(61).

In the local time step strategy, the time step of the coarse regions is twice that of the fine regions. This results in coarse regions requiring fewer discrete solution steps of the governing equations. Since the time steps of the coarse refined regions are certain, the solution format of the intermediate time  $m$  value of the transfer is also certain. The method of solving the transfer intermediate time  $m$  values does not need to involve complex interpolation. Therefore, the additional cost of local time steps is minimal.

Ultimately, the flow chart illustrating the parallel multi-resolution SPH model with local time stepping is presented (see Fig. 4). Within it, ‘initial’ denotes the initial dynamic load balancing strategy; ‘loop’ signifies the program’s iterative calculation process; and the yellow square symbolizes the multi-resolution SPH model introduced in this article, incorporating local time stepping.

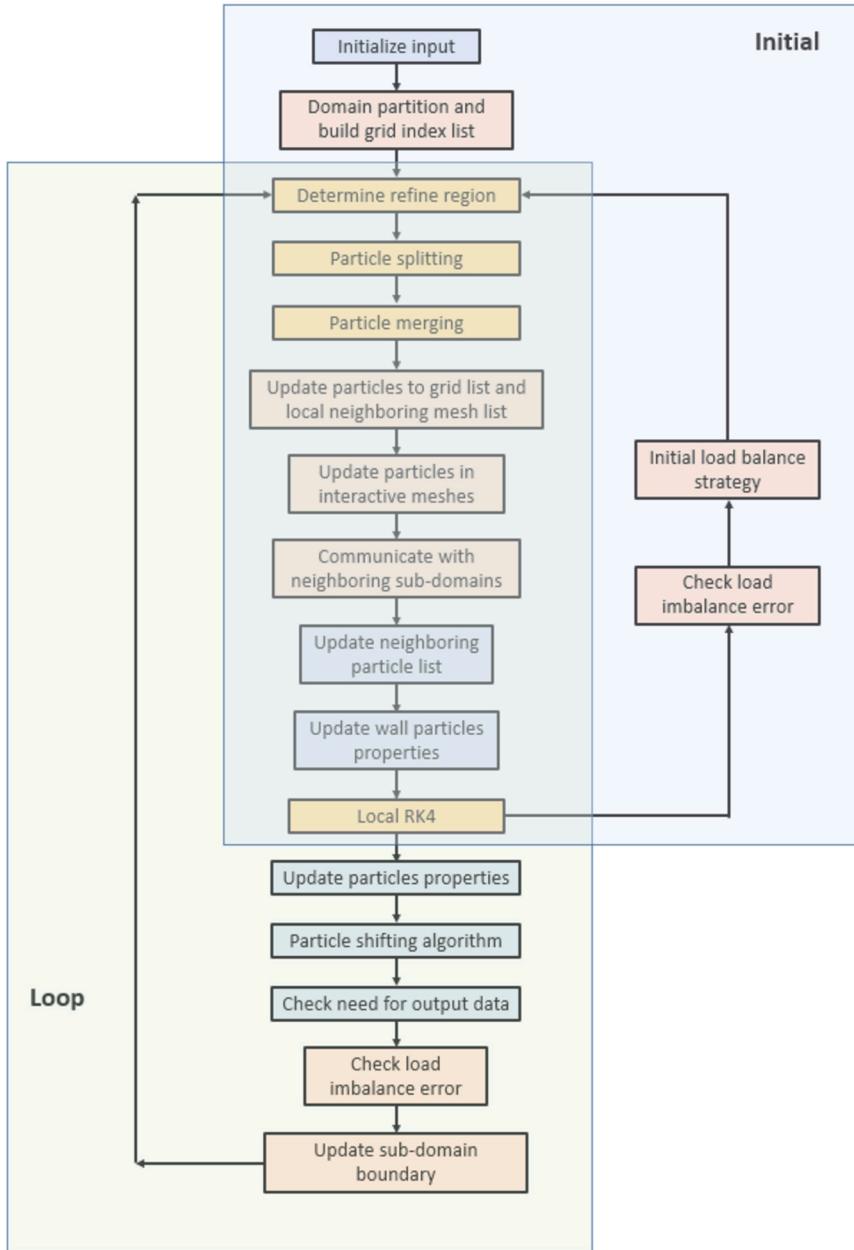
## 4. Numerical validation

### 4.1. Verification of spatial discretization

The validation of spatial discretization accuracy for multi-resolution interfaces is tested. The computation domain is illustrated in Fig. 5, with smaller particles positioned in the refined region. The initial particle spacings of the fine and coarse particles are 0.01 m and 0.02 m, respectively. The following function is calculated by SPH interpolation:

$$f_v = x^2 + y^2 + \sin(xy). \quad (62)$$

The gradient and Laplacian from the SPH model are compared with the analytical values,  $\frac{\partial f_v}{\partial x}$  and  $\frac{\partial^2 f_v}{\partial x^2} + \frac{\partial^2 f_v}{\partial y^2}$ . We define the accuracy as  $\frac{1}{N_f} \sum_{N_f} (f_c - f_a)/f_a$ , where  $f_c$  is the SPH results,  $f_a$  is the analytical results,  $N_f$  is the number of fine particle.



**Fig. 4.** Flow chart of the parallel multi-resolution SPH model. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

Fig. 6 shows the results of gradient component  $f_x$  using different discretization schemes. The gradient component  $f_x$  obtained from Eq. (6) and Eq. (7) deviates from the analytical solution at the interface of different resolutions. If Eq. (8) and Eq. (9) are used, the gradient component  $f_x$  at the interface of different resolutions is more consistent with the analytical result plotted in Fig. 6(a). Moreover, the error is smaller if a correction matrix  $\mathbf{L}$  is used as listed in Table 1. Fig. 7 presents the  $\nabla^2 f$  from different discretization schemes. The  $\nabla^2 f$  obtained using equations Eq. (11) and Eq. (12) has a large error near the interface. The results obtained using Eq. (18) and Eq. (19) are smooth and more consistent with analytical results (see Fig. 7(a)). Using a modified matrix can result in a reduction of errors of  $\nabla^2 f$  as shown in Table 1.

Figs. 8 and 9 show the results of  $f_x$  and  $\nabla^2 f$  under initial irregular distribution. The numerical results of Eq. (6), Eq. (7), and Eq. (8) deviate far from the analytical solution of  $f_x$ . The numerical results of Eq. (11), Eq. (12), and Eq. (18) deviate far from the analytical solution of  $\nabla^2 f$ . Numerical errors can be reduced by using Eq. (9) and Eq. (19) for  $f_x$  and  $\nabla^2 f$ , respectively. The error results listed in Table 1 quantitatively prove that using Eq. (9) and Eq. (19) can reduce the discrete error of the multi-resolution SPH model in the case of irregular particle distribution.

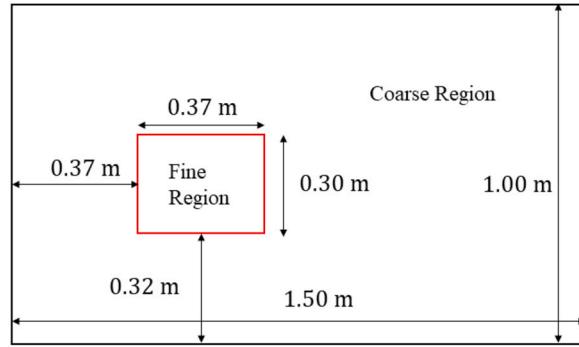
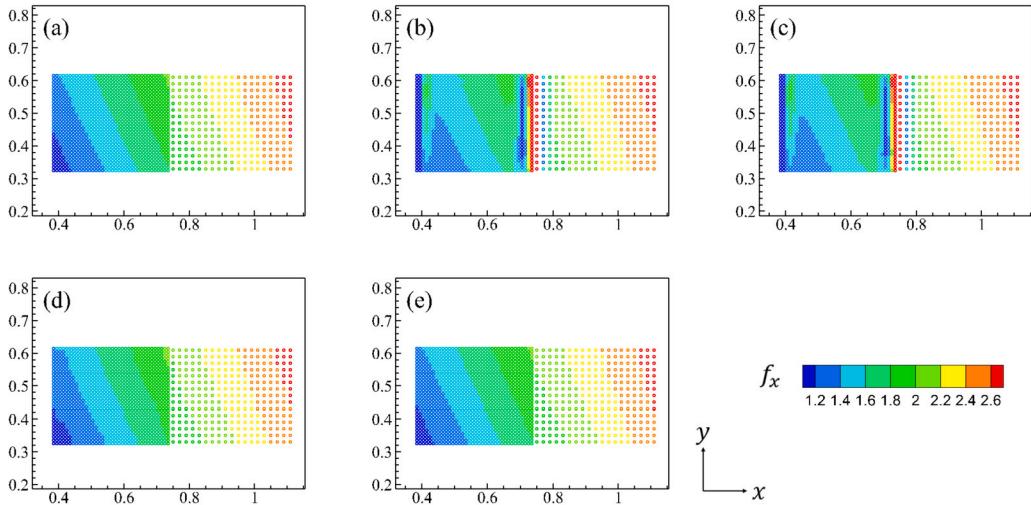
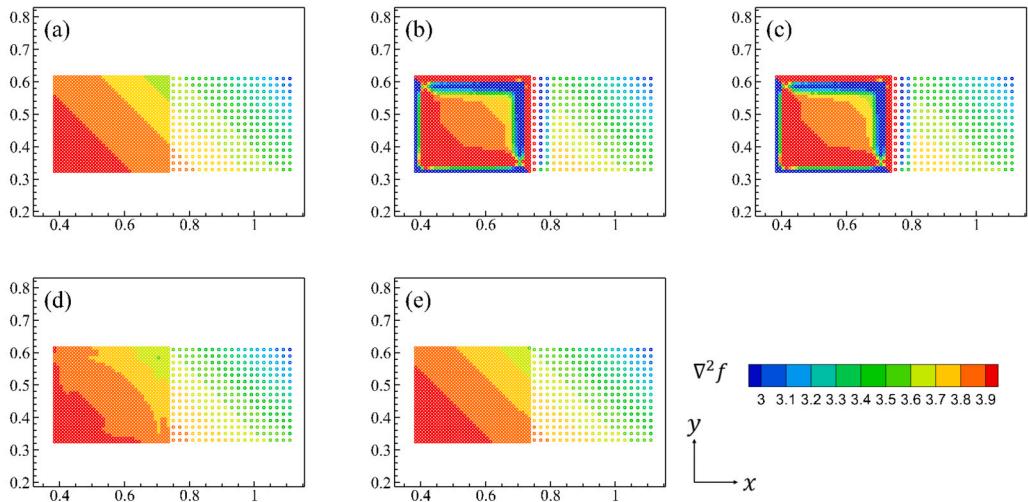
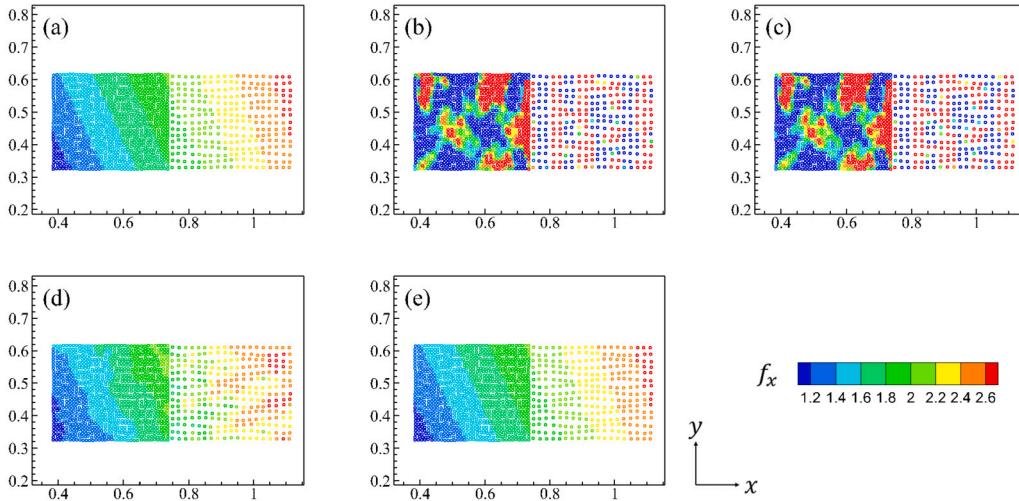
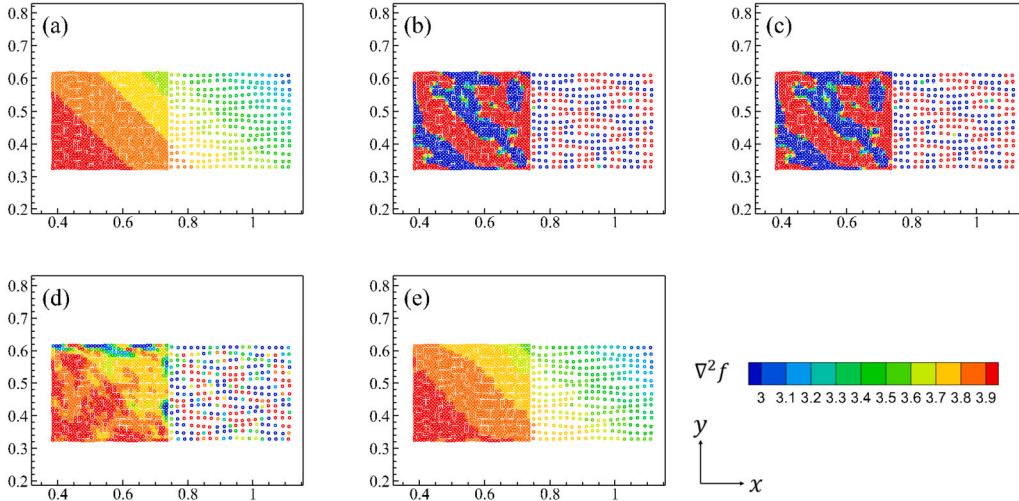


Fig. 5. Schematic of the multi-resolution particle arrangement.

Fig. 6. The gradient component  $f_x$  for different discretization schemes with regular particle distribution. (a): analytical results; (b): Eq. (6); (c): Eq. (7); (d): Eq. (8); (e): Eq. (9).Fig. 7. The Laplacian  $\nabla^2 f$  for different discretization schemes with regular particle distribution. (a): analytical results; (b): Eq. (11); (c): Eq. (12); (d): Eq. (18); (e): Eq. (19).



**Fig. 8.** The gradient component  $f_x$  for different discretization schemes with irregular particle distribution. (a): analytical results; (b): Eq. (6); (c): Eq. (7); (d): Eq. (8); (e): Eq. (9).



**Fig. 9.** The Laplacian  $\nabla^2 f$  for different discretization schemes with irregular particle distribution. (a): analytical results; (b): Eq. (11); (c): Eq. (12); (d): Eq. (18); (e): Eq. (19).

**Table 1**  
Errors of the gradient component  $f_x$  and the Laplacian  $\nabla^2 f$  cases using different discretization schemes.

	Distribution	Eq. (6)	Eq. (7)	Eq. (9)	Eq. (10)
$f_x$	Regular	0.10381	0.10278	0.00497	0.00012
	Irregular	0.71573	0.70624	0.01957	0.00052
	Distribution	Eq. (11)	Eq. (12)	Eq. (18)	Eq. (19)
$\nabla^2 f$	Regular	0.15777	0.15725	0.00381	0.00023
	Irregular	0.81893	0.80970	0.03112	0.00249

In summary, Eq. (9) and Eq. (19) are used to solve the gradient and Laplacian of the governing equations. However, using Eq. (9) to discrete the governing equation to simulate the free surface flow problems would lead to interface error [52]. Therefore, we only use the discretization scheme Eq. (9) for internal particles, and the discrete governing equation Eq. (8) for particles near the free surface. The method for determination of internal particles is the same as Wang et al. [67].

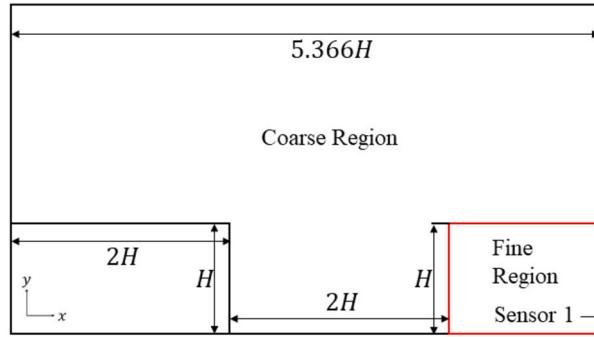
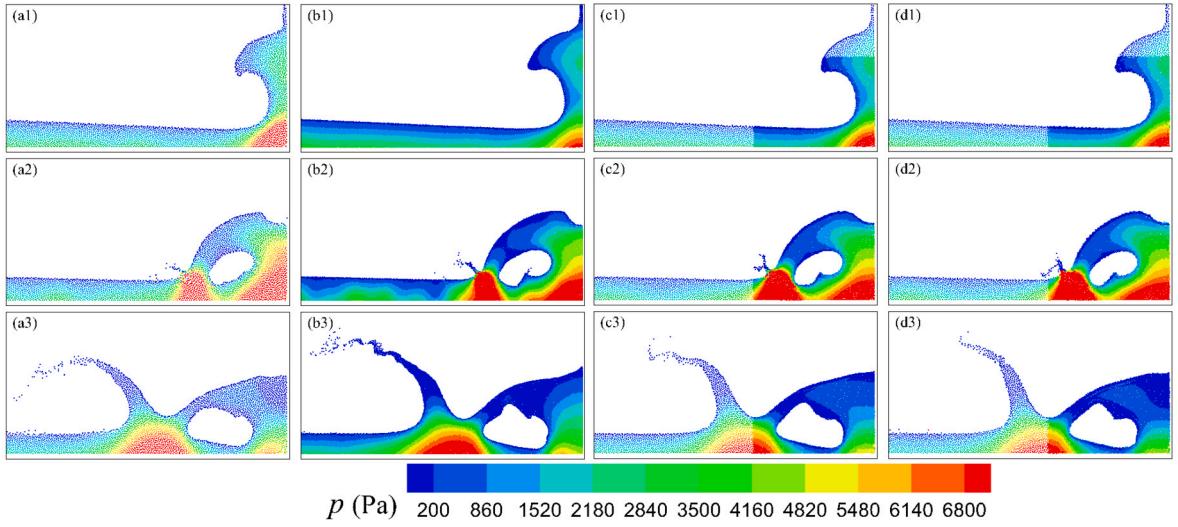


Fig. 10. Schematic of 2D dam-breaking case.

Fig. 11. Pressure field of the dam breaking case (a), case(b), case(c) and case(d) at  $t(g/H)^{0.5} = 4.12$  (1), 4.85 (2) and 5.58 (3).

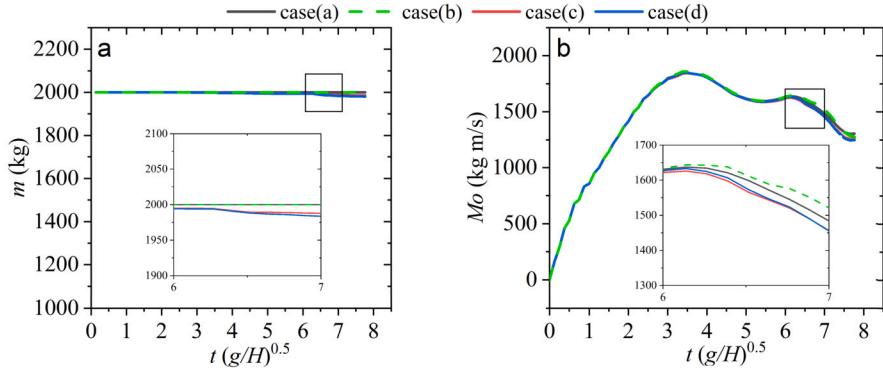
#### 4.2. Dam breaking

Dam breaking is a well-studied case in SPH literature, showcasing the model's proficiency in addressing large deformation challenges. Fig. 10 shows a sketch of the initial setup, where the initial water height is  $H = 0.6$  m, length of the tank is  $5.366H$ . The numerical sound speed is  $c_0 = 10\sqrt{gH}$ . A refinement zone is defined in the bottom right-hand corner, and a pressure sensor is placed at the height of 0.16 m at the right wall. Ghost particles at the refined region are refined. Twelve cores are used, eight of which are used to refine the region. To prove the multi-resolution model in this paper, four simulations are considered:

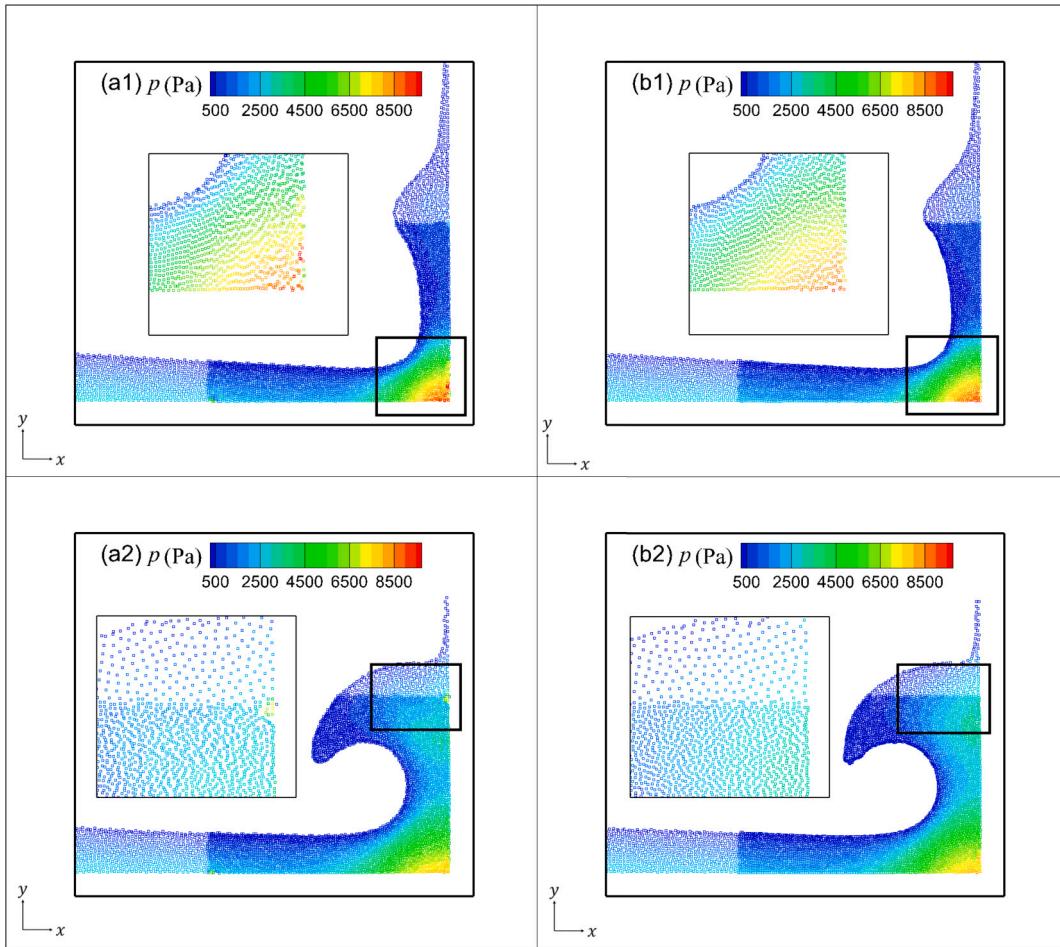
- a) Uniform coarse simulation:  $dx = H/30$ ,  $\Delta t = 0.00002$  s;
- b) Uniform fine simulation:  $dx = H/60$ ,  $\Delta t = 0.00001$  s;
- c) multi-resolution and global time step simulation:  $dx0 = H/30$ ,  $\Delta t_0 = 0.00001$  s,  $dx1 = H/60$ ,  $\Delta t_1 = 0.00001$  s;
- d) multi-resolution and local time step simulation:  $dx0 = H/30$ ,  $\Delta t_0 = 0.00002$  s,  $dx1 = H/60$ ,  $\Delta t_1 = 0.00001$  s.

Fig. 11 shows the pressure distribution diagrams of the four cases at three different moments. The four cases exhibit similar pressure fractions and free surface shapes, although there are some minor differences. Moreover, the multi-resolution model also exhibits smooth continuous pressure distributions at the interfaces of different resolutions. The same results can be found in models with the multi-resolution local time stepping.

Fig. 12 shows the time series of mass and momentum. The mass in cases (a) and (b) is conserved. However, the system mass of cases (c) and (d) decreases over time. This occurs when certain particles are unable to combine due to having fewer nearby fine particles than necessary for merging. Then they are deleted directly as the calculation proceeds in this case. Until about  $t(g/H)^{0.5} = 3.0$ , the system momentum is the same for the four examples. As time progresses, the system momentum of cases (c) and (d) is also smaller than that of cases (a) and (b). Note that the momentum of Case (b) surpasses that of Case (a). This could be primarily attributed to the larger particle distance in Case (a), which consequently yields a higher discrete error. Consequently, there is a faster dissipation of mechanical energy from the system.



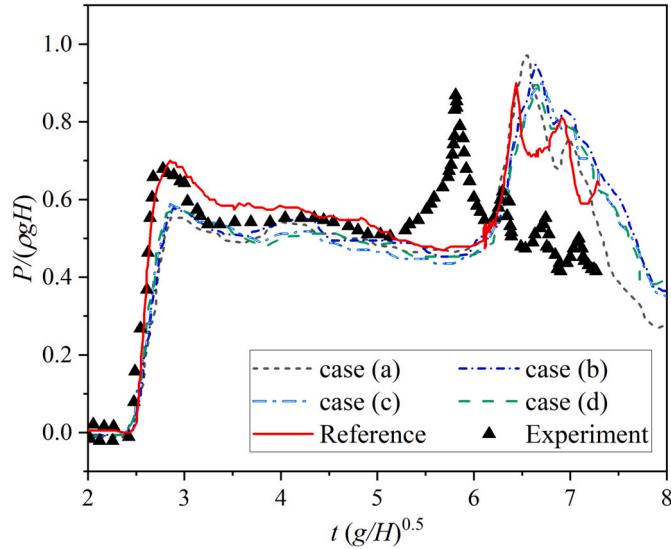
**Fig. 12.** Time series of system mass  $m$  and momentum  $Mo$  for the dam breaking cases.



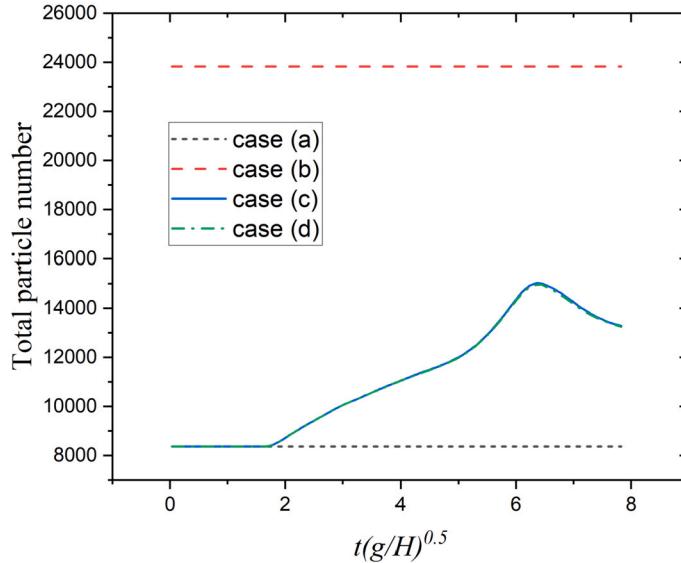
**Fig. 13.** The pressure field and local particle distribution with different  $\Delta x_f$ . (a1)  $\Delta x_f = H/30$  for coarse particle and  $\Delta f = H/60$  for fine particle at  $t(g/H)^{0.5} = 4.70$ ; (b1)  $\Delta x_f = H/30$  for fine and coarse particles at  $t(g/H)^{0.5} = 4.70$ ; (a2)  $\Delta x_f = H/30$  for coarse particle and  $\Delta f = H/60$  for fine particle at  $t(g/H)^{0.5} = 5.64$ ; (b2)  $\Delta x_f = H/30$  for fine and coarse particles at  $t(g/H)^{0.5} = 5.64$ .

Fig. 13 shows the pressure field and local particle distribution for the different  $\Delta x_f$  cases at two moments. This indicates that setting  $\Delta x_f$  with a larger particle distance can enhance the anisotropic distribution of particles. The primary rationale behind this observation is that the support domain length of fine particles matches that of coarse particles.

Fig. 14 compares the pressure time series at the sensor 1. The model of this article captures the first impact of water to avoid the first impact  $t(g/H)^{0.5} = 3.0$ , although the peak value is less than the experimental value and reference results. The second pressure peak lags behind the experimental results around  $t(g/H)^{0.5} = 6.5$ . This difference was also discovered in the reference resolution.



**Fig. 14.** Time series of pressure at sensor 1, compared with between the experimental results [5] and reference results [47].



**Fig. 15.** Time series of total particle numbers.

**Table 2**  
Cost of the 2D dam breaking cases.

case	a	b	c	d
Cost (s)	1518.75	22104.73	3579.35	1789.46

More importantly, the current multi-resolution model and the results of the single-solution analysis model have not shown a large difference. The local time stepping strategy has not led to significant changes in the results. Table 2 presents the time cost of the four cases. It can be seen that the parallel multi-resolution SPH model with the local time stepping can significantly reduce computational time. Fig. 15 shows the time series of total particle numbers. It can be found the present model can reduce the particle numbers. The local time step method has a limited effect on particle number.

#### 4.3. Oscillating drop under a central force field

A 2D drop evolving under the action of a central conservative force field is simulated [50,3,67,31]. It is a simple free-surface flow test case, and a comparison between the present numerical results and the exact analytical ones enables us to reveal the accuracy,

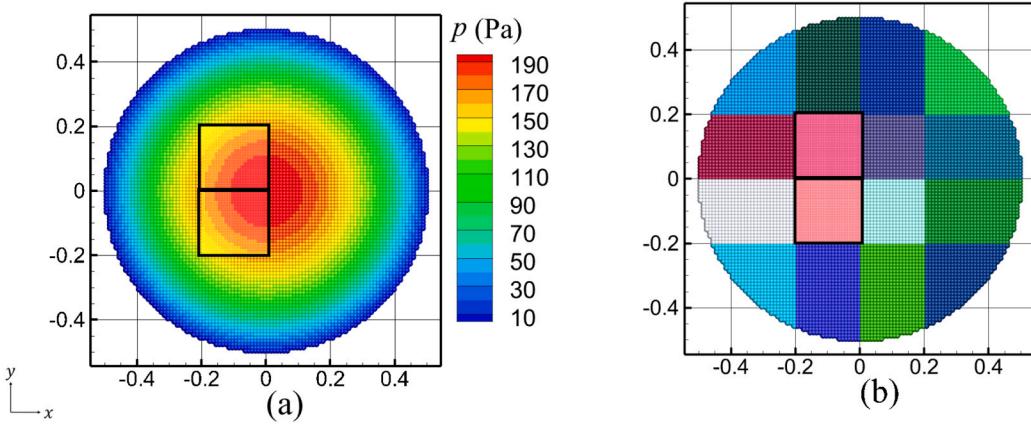


Fig. 16. Schematic of the oscillating drop. (a) Pressure field; (b) Subdomain distributions (Two black blocks are refined subdomains.)

energy conservation and convergence features of the new multi-resolution scheme in long-term simulations. The fluid for this case is assumed to be inviscid and incompressible. The drop is exposed to a potential field of  $0.5\Omega^2 R^2$  ( $\Omega = 1.2$  is a dimensional parameter,  $R = 0.5$  m is the radius of the initial circular drop), and accordingly is subjected to a conservative force field being equal to

$$f_x = -\Omega^2 x, \quad (63)$$

$$f_y = -\Omega^2 y, \quad (64)$$

where  $f_x$  and  $f_y$  are forces along  $x$  and  $y$  directions, respectively. The initial velocities and pressure fields are

$$u(x, y) = \Gamma x, \quad (65)$$

$$v(x, y) = -\Gamma y, \quad (66)$$

$$p = 0.5\rho_0(\sigma(0)^2 + \Omega^2)R^2 + (x^2 + y^2), \quad (67)$$

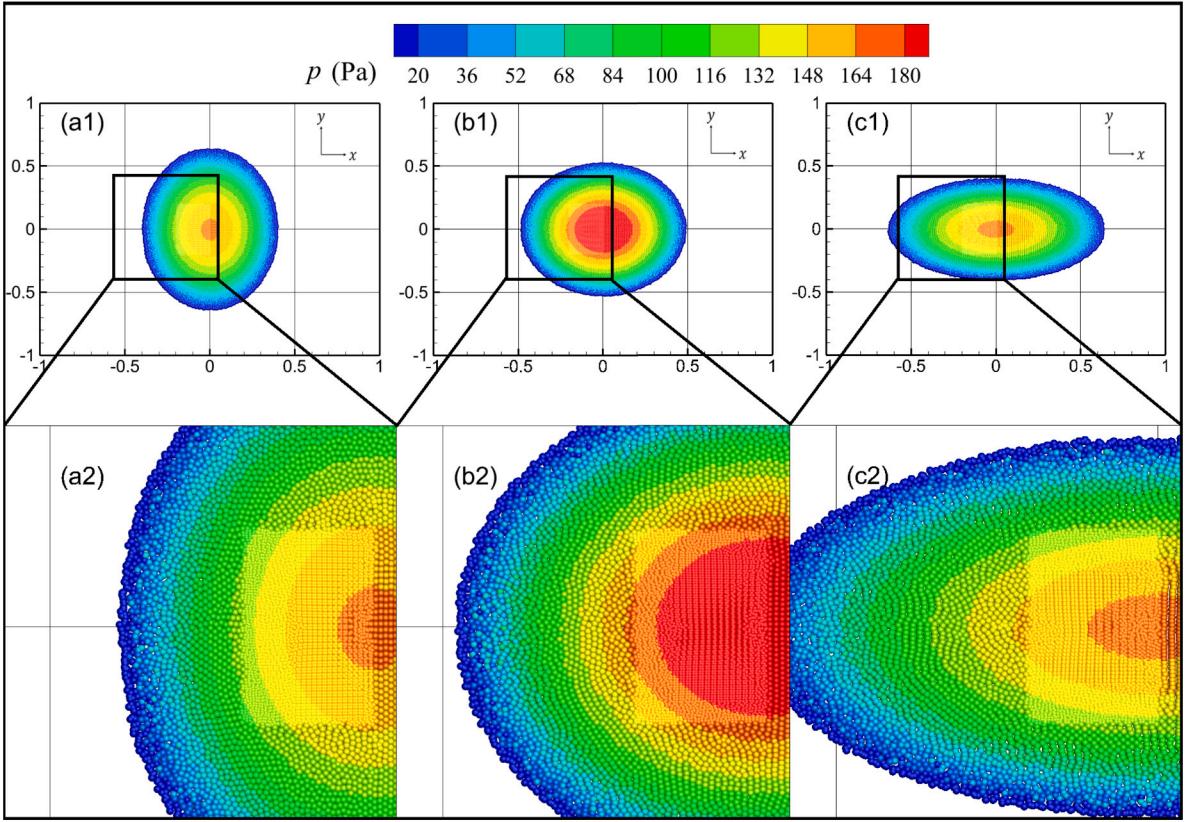
where  $\Gamma$  is set to 0.4 in this numerical test. The initial pressure and subdomains are displayed in Fig. 16. Two subdomains are set to refined regions as plotted in Fig. 16 (b).

Fig. 17 depicts the pressure field obtained by the proposed model. It can be found that at different calculation moments, the pressure field remains stable and the model does not cause instability in the pressure field. Meanwhile, the local enlargement shows that the model ensures smooth and stable interfaces with different particle resolutions.

Fig. 18 shows the time evolution of the vertical semi-major axis obtained by using three different coarse particle spacing,  $R/dx0 = 50$ ,  $R/dx0 = 100$ , and  $R/dx0 = 200$ . Since the ratio of coarse particles to fine particles remains 2:1, the initial particle spacing of fine particles also changes accordingly. The present SPH results show a good convergence towards the analytical solutions Root Mean Square Errors (RMSE) are 0.01128, 0.01148, 0.00759, 0.00602 for  $R/dx0 = 50$  global time step,  $R/dx0 = 50$  local time step  $R/dx0 = 100$  local time step, and  $R/dx0 = 200$  local time step. The local time step strategy does not lead to significant differences in the results from those using the global time step.

Fig. 19 (a) shows that the pressure field at 5 seconds is smooth and stable. However, its normalized kernel summation-based density has noise at the multi-resolution interface, as shown in Fig. 19 (b). This indicates an error in normalization near the interface of different resolutions. In addition, kernel summation-based density produces noise near the free surface. On the one hand, this is due to kernel discretization errors caused by truncation boundaries. On the other hand, the gradient is calculated using Eq. (7) near the free surface, which will lead to an error in the velocity gradient, as shown in Fig. 19 (c). Fig. 19 (d) shows the particles near the free surface, the gradient of which is calculated using Eq. (7). This overlaps with the velocity gradient noise region as plotted in Fig. 19 (c). Moreover, after calculating the gradient using Eq. (9) inside the fluid, the multi-resolution interface does not produce large numerical noise.

Fig. 20 illustrates a quantitative comparison of the mechanical energy time series using global time step and local time step methods. The outcomes yielded by the local time step method closely resemble those of the global time step method. The RMSE values for kinematic energy are 0.27203 and 0.26509 for the global time step and local time step methods, respectively. For potential energy, the RMSE values are 0.20129 and 0.22075 for the global time step and local time step methods, respectively. The total mechanical energy undergoes oscillatory fluctuations over time. The total energy of the system remains relatively stable in comparison to the initial total mechanical energy, yet there is a mild downward trend over time. This could stem from the particle shifting technique introduced by Wang et al. [67], which may not ensure the conservation of system energy and could potentially result in a non-conservation of system volume, as discussed by Lyu and Sun [45].



**Fig. 17.** Pressure the oscillating drop. (a1) Pressure field at 7.5 s; (a2) Local pressure field at 7.5 s; (b1) Pressure field at 8.4 s; (b2) Local pressure field at 8.4 s; (c1) Pressure field at 9.3 s; (c2) Local pressure field at 9.3 s.

#### 4.4. Taylor-Green flow

The new multi-resolution scheme including the newly incorporated Laplacian model for modeling Laplacian of velocity for viscous stresses can be validated through consideration of a Taylor-Green flow. The Taylor-Green vortex flow provides a time-dependent, spatially periodic analytical solution of the Navier-Stokes equations [33,1,29]. The flow encourages particle stretching and bunching, making it a demanding test of the algorithm's effectiveness at modeling internal flows. The analytical velocity field is

$$u(x, y, t) = -U e^{bt} \cos(2\pi x) \sin(2\pi y), \quad (68)$$

$$v(x, y, t) = -U e^{bt} \sin(2\pi x) \cos(2\pi y), \quad (69)$$

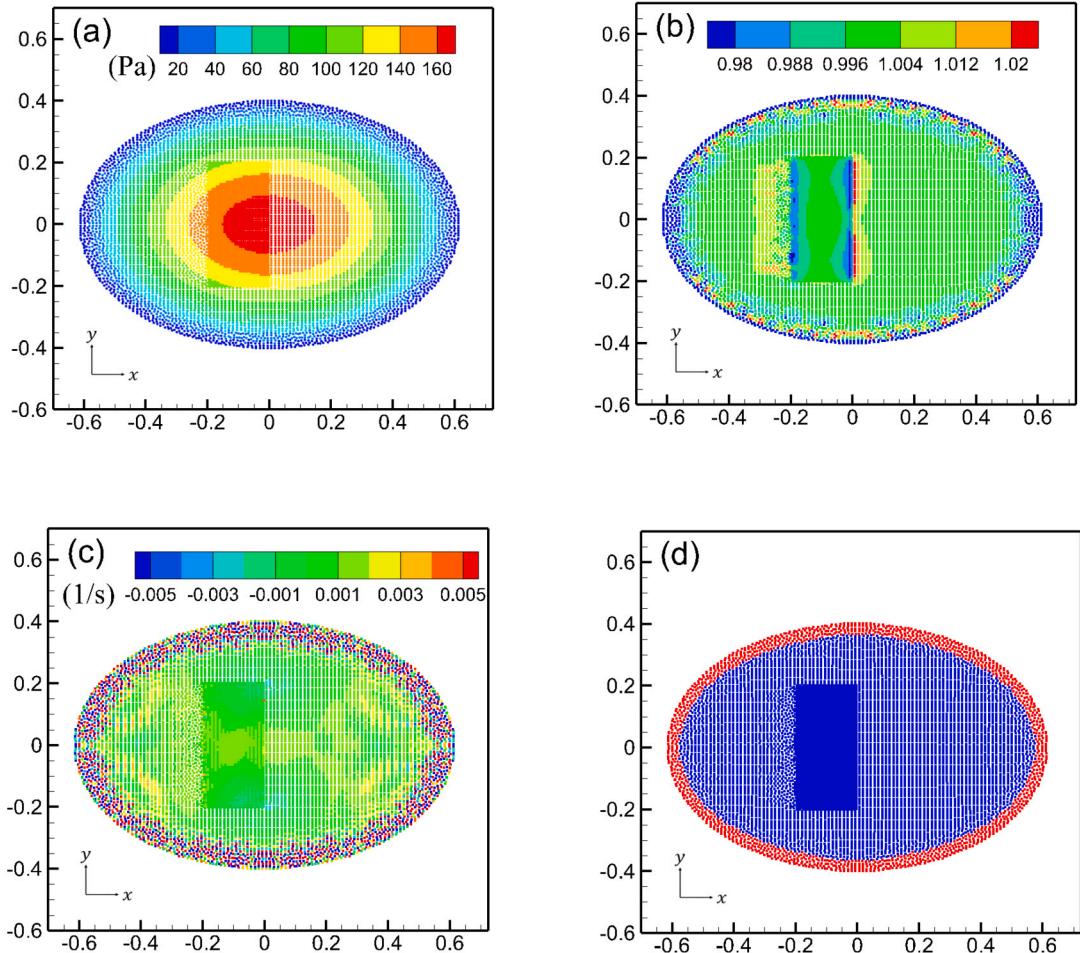
where  $u$  and  $v$  are horizontal and vertical velocities, respectively.  $U = 1.0$  m/s is the velocity scale and  $b = -8\pi^2/Re$  is the velocity decay rate (Reynolds number,  $Re = UL/v$ ,  $L = 1.0$  m is the length of the square domain). Periodic boundary conditions are applied in both coordinate directions. In this case,  $Re = 1000$  is considered. The initial velocity field and subdomain distribution are shown in Fig. 21. Four subdomains are selected as the refined regions as shown in Fig. 21 (c). Initial particle spacing is 0.01 m with time step 0.00005 s. The refined particle spacing is 0.005 m with time step 0.000025 s.

Fig. 22 shows the velocity and pressure fields predicted by the numerical simulation. The utilization of the multi-resolution model alongside a local time-stepping strategy does not yield notable numerical noise. The local enlargement of the image shows that the interfaces at different resolutions are smooth, indicating the successful implementation of the dynamic particle merging and splitting strategy.

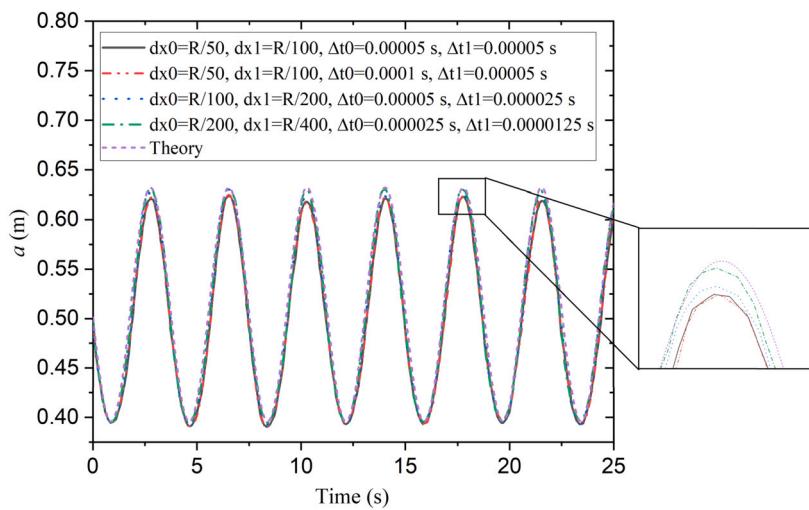
Fig. 23 demonstrates the accuracy of the SPH scheme by comparing the pressure distribution along  $x = 0$ , the horizontal velocity distribution along  $x = 0$ , and the vertical velocity distribution along  $y = 0$  with the theoretical results. The numerical results predicted by using the current model are in good agreement with the theoretical ones.

#### 4.5. Flow past a cylinder

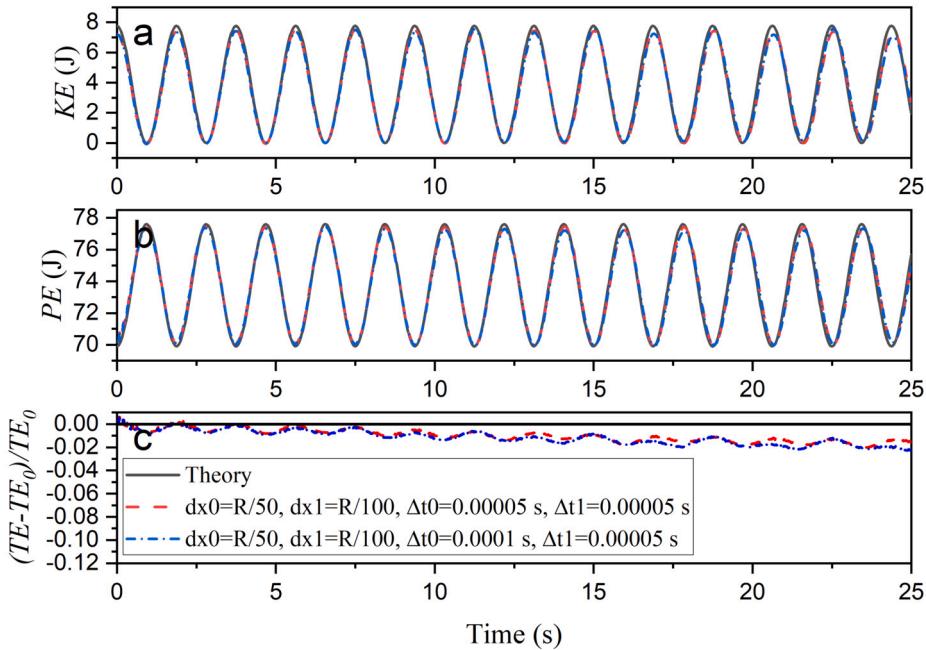
The setup of the flow past a cylinder case is shown in Fig. 24, where the cylinder is inclined with the diameter of  $D = 0.2$  m. The center of the cylinder is placed at  $(4D, 4D)$ , and the coarse particle resolution is set as  $dx_0 = D/20$ . The density of the fluid is  $\rho = 1000$  kg/m<sup>3</sup>. Reynolds number is  $Re = UD/v$ , where  $v = 0.001$  m<sup>2</sup>/s for all the cases. Different inflow velocities are to be selected,



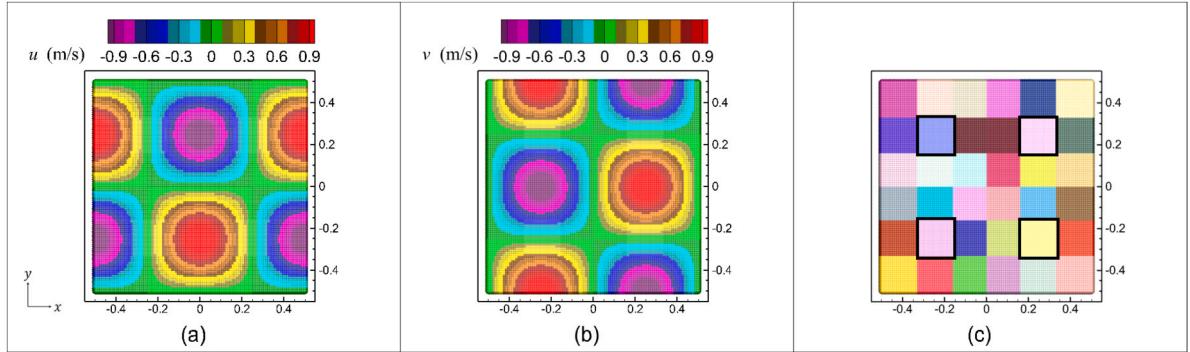
**Fig. 18.** (a) Pressure field of the oscillating drop at 5.0 s with the local time step strategy. (b) Normalized kernel summation-based density field at 5.0 s; (c) Velocity divergence field at 5.0 s; (d) Particles (Red dots) near the free surface at 5.0 s.



**Fig. 19.** Time evolutions of the horizontal semi-major axis. ( $dx_0$  and  $dx_1$  denote initial the particle spacing of coarse and fine particles, respectively;  $\Delta t_0$  and  $\Delta t_1$  denote time steps for initial coarse and fine particles, respectively.)



**Fig. 20.** Time histories of mechanical energy. (a) Potential energy  $PE$ ; (b) Kinetic energy  $KE$  (c) The change of total mechanical energy  $TE$ . ( $TE_0$  is the initial total mechanical energy.)



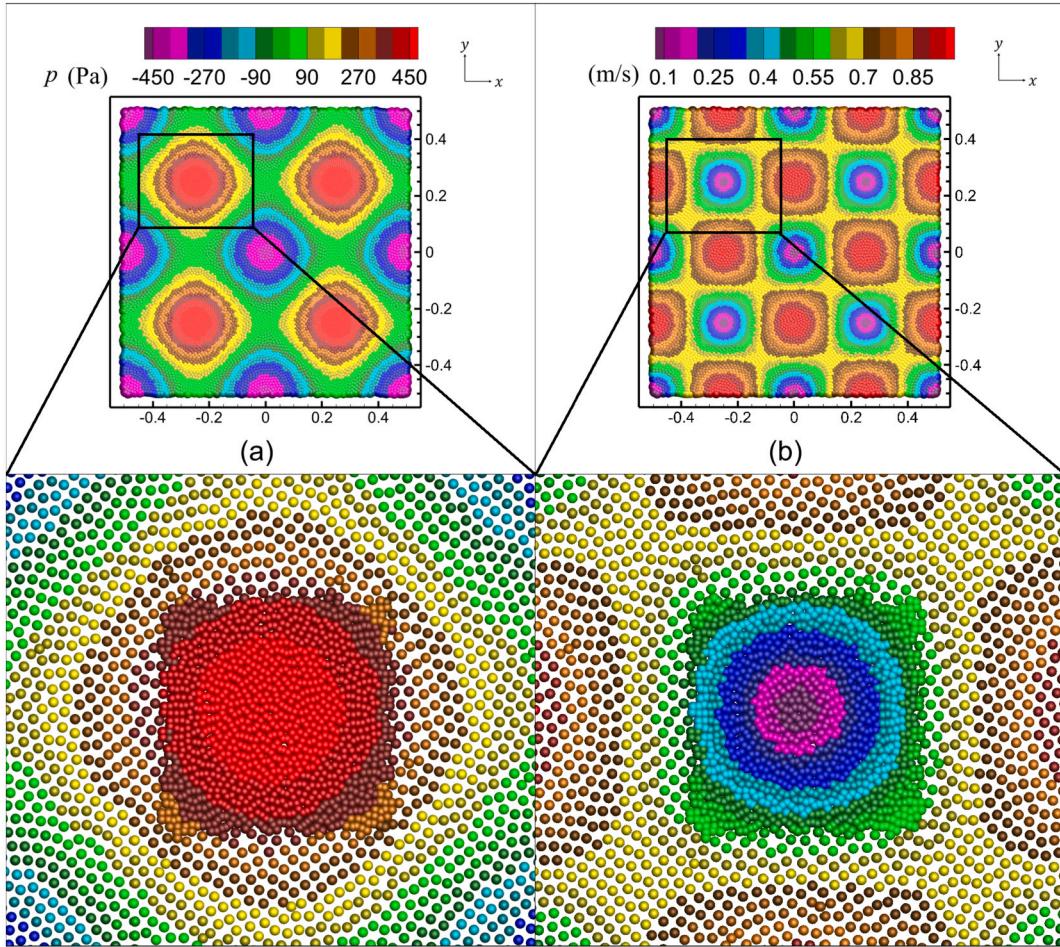
**Fig. 21.** Schematic of the Taylor-Green Flow. (a) Horizontal velocity field; (b) Vertical velocity field; (c) Subdomain distributions (Four black blocks are refined subdomains.)

resulting in various Reynolds numbers. Inlet pressure is determined through Shepard interpolation. Conversely, the outlet pressure remains fixed at zero. A buffer zone with a thickness of  $3dx_0$  is defined at both the inlet and outlet. Slip boundary conditions are imposed on the upper and lower sides, while a non-slip boundary condition is enforced on the cylinder's surface. For the cylinder, the ghost particles need to be refined. Subdomains that include inlet and outlet boundary particles are not refined.

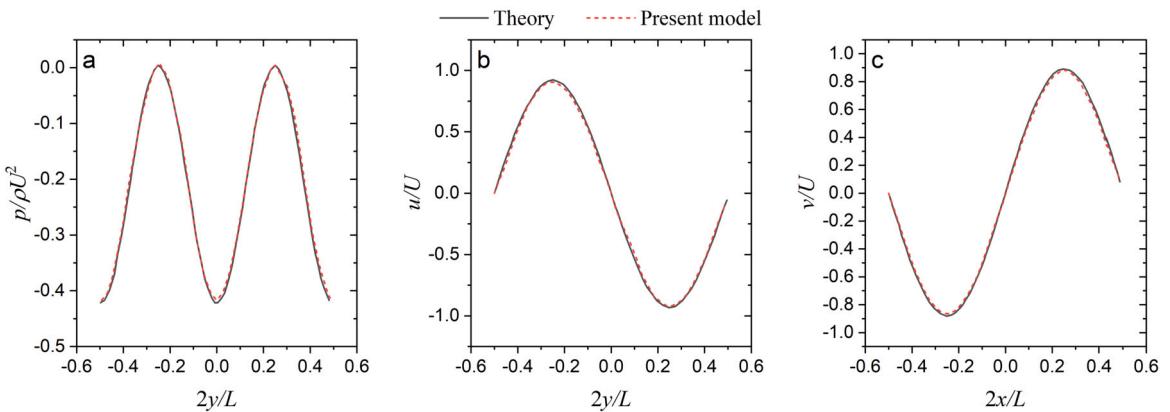
#### 4.5.1. $Re = 200$

For the  $Re = 200$  case, the fluid is moving at a velocity of  $U_f = 1.0$  m/s in the rightward direction. A refinement region is defined beyond the cylinder, and any subdomains containing particles within 0.05 meters of the cylinder's exterior are recognized as part of this refinement region. At the initial moment, the computing domain is evenly divided by 24 cores, as shown in Fig. 25(a). After the dynamic loading balancing strategy, the refinement area was reduced (as shown in Fig. 25(b)) but still wrapped around the cylinder because the refinement rule was within 0.05 m outside the cylinder.

Fig. 26 shows the local vorticity field from  $(tU_f)/D = 0.5$  to 3.0. As time progresses, the vorticity on the right side of the cylinder gradually increases. Vorticity is smooth across interfaces of different resolutions. Fig. 27 shows the horizontal and vertical velocity fields at three moments. As time progresses, the leftward horizontal velocity area on the right side of the cylinder gradually increases. The vertical velocity is symmetrical up and down with the center of the cylinder as the axis. Over time, the velocity remains stable and smooth at the multi-resolution interface, as shown in the local velocity field.



**Fig. 22.** The numerical results from the present model at  $tU/L = 1.0$ . (a) Pressure field; (b) Speed field.



**Fig. 23.** Comparison of theoretical and present model predictions of hydrodynamical variables at  $tU/L = 1.0$  with  $Re = 1000$ . (a) Pressure profiles along  $x = 0$ ; (b) Horizontal velocity profiles along  $x = 0$ ; (c) Vertical velocity profiles along  $y = 0$ .

The drag force in the SPH model can be calculated as [48]. The drag coefficient  $C_d$  can be calculated as

$$C_d = \frac{F_d}{0.5\rho U_f^2 D}. \quad (70)$$

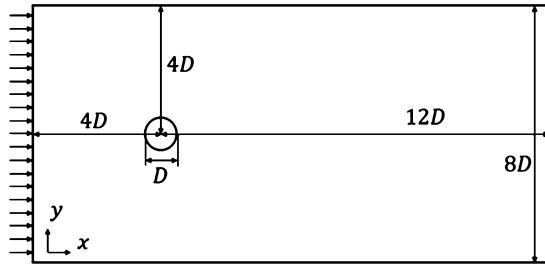


Fig. 24. Schematic of the flow over an cylinder.

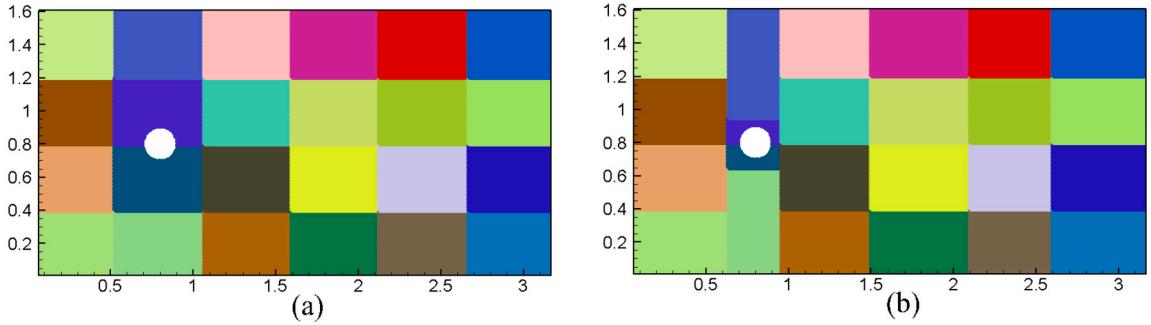
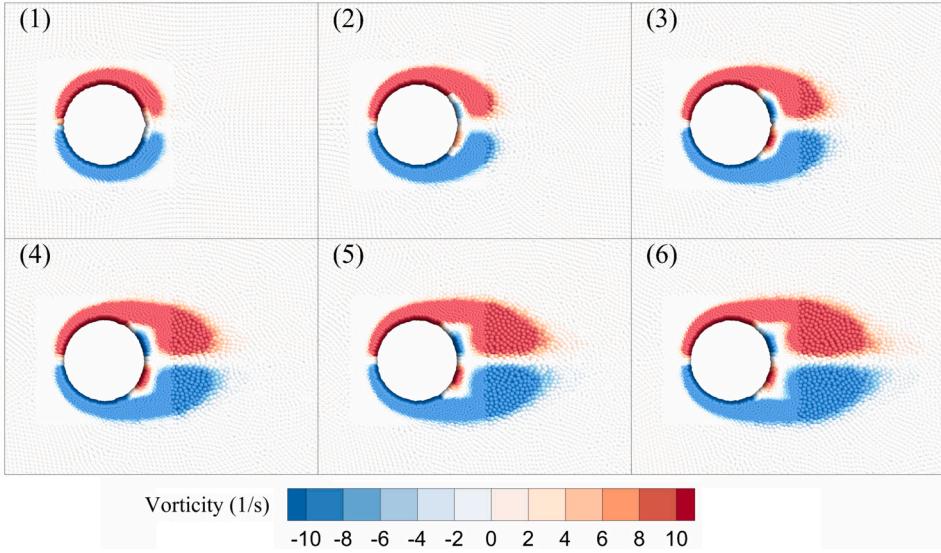


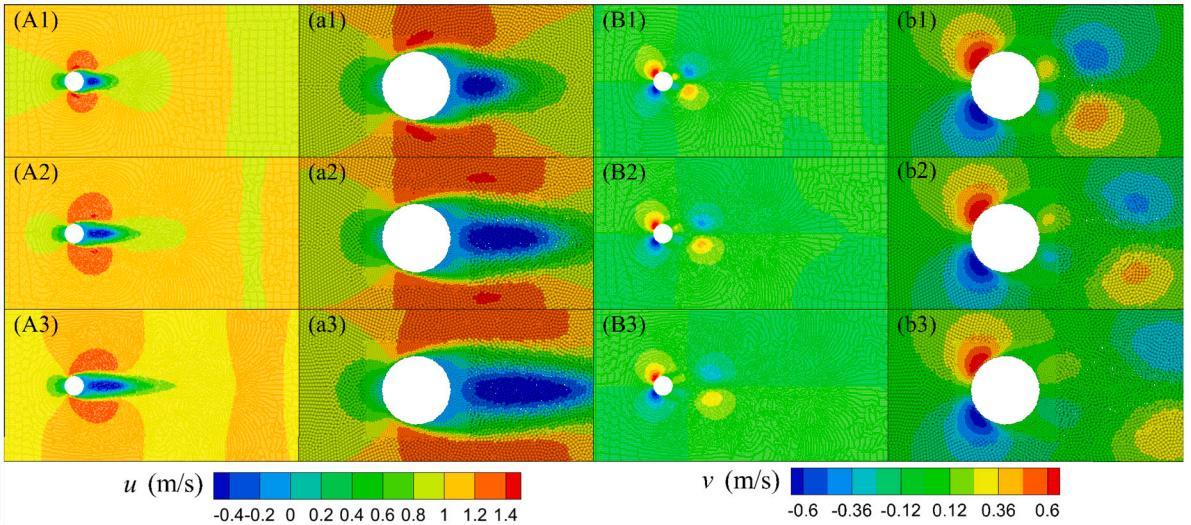
Fig. 25. Initial subdomains distribution (a) and subdomains distribution after initial load balance strategy [80] (b).

Fig. 26. Local vorticity field of  $Re = 200$  case near cylinder at  $(tU_f)/D = 0.5$  (1),  $1.0$  (2),  $1.5$  (3),  $2.0$  (4),  $2.5$  (5), and  $3.0$  (6).

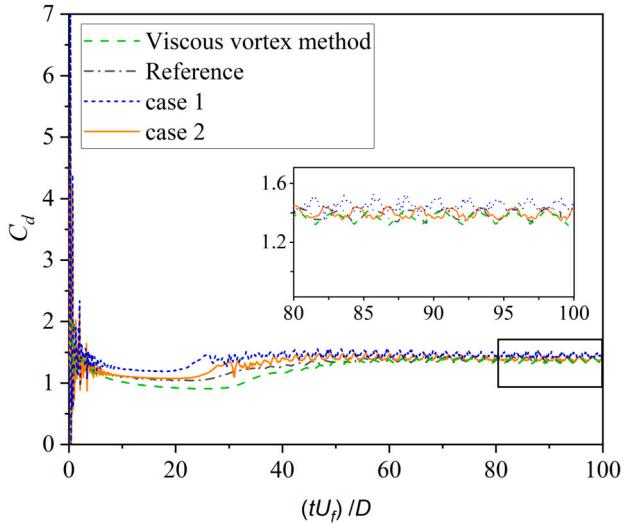
The time series of drag coefficient  $C_d$  are shown in Fig. 28. At the beginning, the drag coefficient experiences some oscillations. After that, the drag coefficient slightly drops down and then increases fast. The drag coefficient predicted by the present model enters the stable oscillating region at  $(tU_f)/D = 40$ , which is slightly earlier than the results obtained by the reference solution and the viscous vortex method (around  $(tU_f)/D = 50$ ). After  $(tU_f)/D = 95$ , the results from case 2 agree well with reference results [48] and those from viscous vortex methods [20,48]. For case 1 which has smaller particle resolution than case 2, the predicted drag coefficient is larger than the one from case 2, reference resolution and results obtained by viscous vortex methods. Despite the presence of minor discrepancies in the results, the results affirm the credibility of the present model's predictions.

#### 4.5.2. $Re = 1000$

For the  $Re = 1000$  case, the fluid is moving at a velocity of  $U_f = 5.0$  m/s in the rightward direction. Fig. 29 shows the simulation results of the vorticity, resolution distribution, and local vorticity field. The rules for defining the refinement region include two points: there are particles in the subdomain whose distance from the outside of the cylinder is less than 0.05 m; the subdomain in



**Fig. 27.** Horizontal field (A), local horizontal velocity field (a), vertical velocity field (B), and local vertical velocity field (b) of  $Re = 200$  case at  $(tU_f)/D = 2.5$  (1), 5.0 (2), and 7.5 (3).

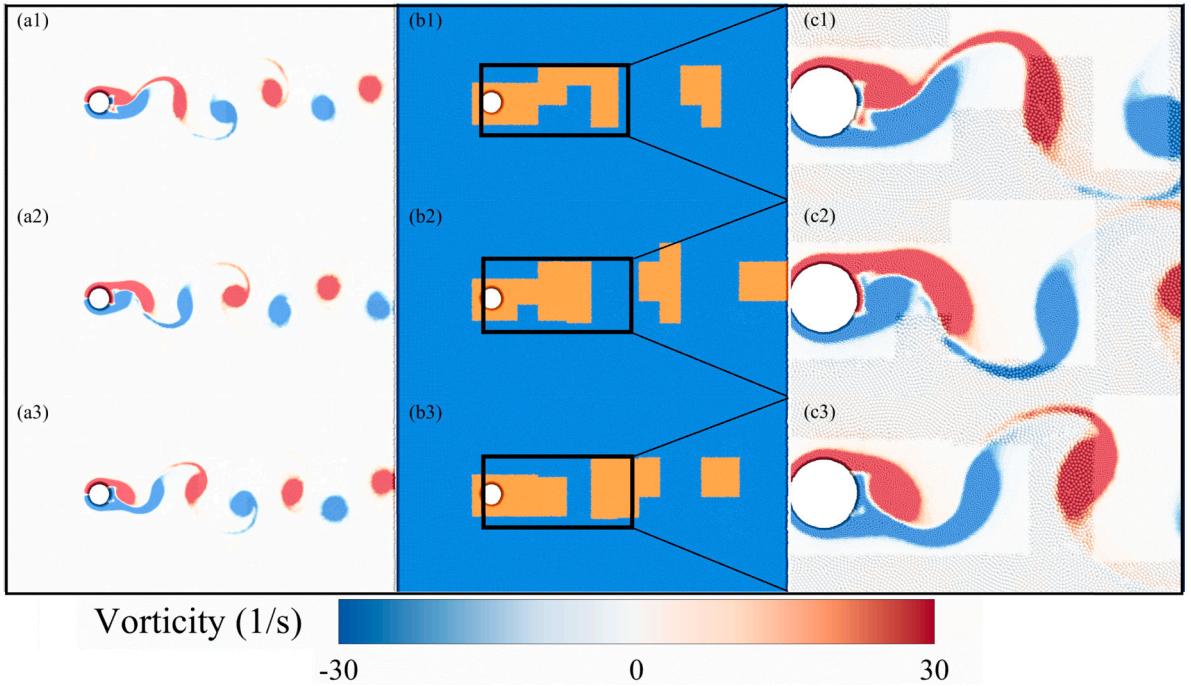


**Fig. 28.** Drag coefficient  $C_d$  for flow past a cylinder with  $Re = 200$ . (Solution obtained by Viscous vortex method [20]; Reference solution obtained by SPH from Marrone et al. [48]; case 1 obtained by the present model:  $dx_0 = 0.005$  m,  $dx_1 = 0.0025$  m; case 2 obtained by the present model:  $dx_0 = 0.01$  m,  $dx_1 = 0.005$  m.)

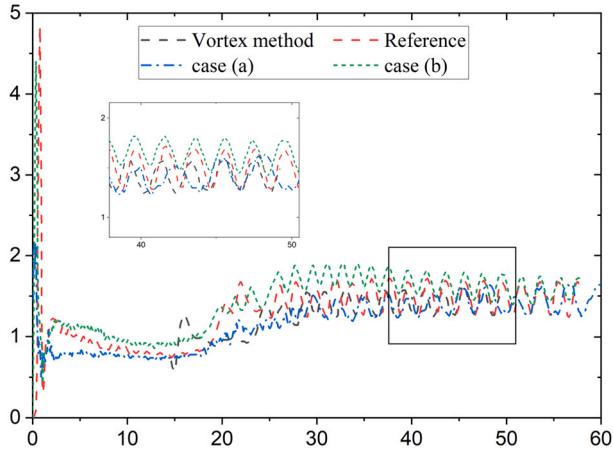
which the vorticity of particles is less than  $-30 \text{ s}^{-1}$  detecting every 0.1 s. The first rule causes the particles around the cylinder to be refined. The second rule causes the refined selection of areas to change dynamically. This rule sets the dynamic detection vortex time to validate the current model and the ability of the dynamical refinement process. The vorticity results shown in Fig. 29 demonstrate that the subdomains near the cylinder belong to the refined region, and the refined region of the flow field to the right of the cylinder is dynamic. The local vorticity map shows that the vorticity at the interface between the coarse and fine regions is also smoothly transitioned. The flow field is stable despite dynamic changes in subdomain refinement. Occasionally, the refinement area does not meet the specified requirements because the refinement area is selected only every 0.1 s. For example, the refined region in Fig. 29 is not the case where the required vorticity is less than  $-30 \text{ s}^{-1}$ . Fig. 30 shows the drag coefficient time history of the SPH multi-resolution cases. At such high Reynolds numbers, the alignment among conventional mesh-based solvers, the  $\delta$ -SPH method, and the current model persists. With increased particle density, there is a slight augmentation in drag coefficients compared to the reference solution.

#### 4.6. 3D dam breaking

To assess the effectiveness in addressing 3D challenges, a simulation of a dam breaking scenario in three dimensions was conducted in a numerical water tank. The size of the dam break calculation example is shown in Fig. 31, where the width of the tank



**Fig. 29.** Vorticity field (a), resolution distribution (b) and local vorticity field (c) of  $Re = 1000$  case at  $(tU_f)/D = 52.5$  (1), 55.0 (2) and 57.5 (3).



**Fig. 30.** Drag coefficient for flow past a cylinder with  $Re = 1000$ . (Solution obtained by the vortex method [32]; reference solution from Marrone et al. [48] using the  $\delta$ -SPH; case a:  $dx_0 = 0.01$  m,  $dx_1 = 0.005$  m; case b:  $dx_0 = 0.002$  m,  $dx_1 = 0.001$  m.)

is  $H$ . The initial particle spacing in the refinement area is  $H/100$ , and the time step is 0.00005 s. The rest of the tank has an initial particle spacing of  $H/50$  and a time step of 0.0001 s. The refinement area is set to  $x/H > 3.75$ .

Fig. 32 shows the velocity field of the 3D dam breaking case. The deformation of the fluid is similar to the 2D one as plotted in Fig. 10. At different resolution interfaces, the velocity results are stable at different moments. The waterfront time series of the present model in this paper is compared with the experimental data [41], the analytical solution [56] and reference results from a two-phase SPH model [10], as shown in Fig. 33. It can be seen that at the initial moment, the results in this paper are consistent with the experimental and reference solutions. As time goes on, the results and the reference solutions gradually deviate from the experimental data, but gradually approach the analytical solution.

#### 4.7. Performance tests

To examine the performance of the current multi-resolution SPH model, a series of 2D hydrostatic tests regarding a 1.0 m long and 1.0 deep water tank are simulated. The length of the tank is 1.0 m and the water depth is 1.0 m. Only four cores are used to simulate this problem. As shown in Fig. 34, the water tank is divided into four subdomains, where subdomain 3 is designated as the

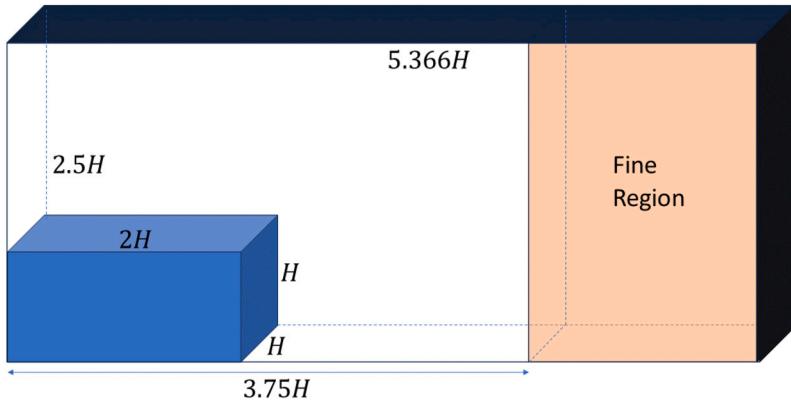


Fig. 31. Schematic of 3D dam breaking case.

**Table 3**

Time cost of different division conditions and discretization for 2D hydrostatic tests. ( $\omega$  denotes the decrease in the time of multi-resolutions relative to the time of single-resolutions,  $\omega = (t_m - t_s)/t_s$ , where  $t_m$  is the time cost of the multi-resolution model,  $t_s$  is the time cost of the single-resolution model. A plus sign indicates an increase in calculation time, and a minus sign indicates a decrease in calculation time.)

Division	d1		d2			d3		
	Particles	Time (s)	Particles	Time (s)	$\omega$	Particles	Time (s)	$\omega$
(a)	10000	123.61	4375	233.64	+0.89	4375	209.06	+0.69
(b)	10000	235.69	2800	48.04	-0.80	2800	40.16	-0.83
(c)	10000	191.04	2725	36.30	-0.80	2725	26.75	-0.86

refinement region. The initial particle spacing of coarse particles is 0.02 m, and that of fine particles is 0.01 m. Numerical simulations were tested for only 0.01 s. We adjust the subdomain size to study model performance under different partitions. Therefore, three different divisions are adopted, as shown in Fig. 34. Division (a): Divide the calculation area evenly, and the refined area accounts for one quarter. Division (b): The refined area accounts for a small proportion; Division (c) is the division obtained using the dynamic loading strategy [80]. In addition, we discuss three different discretization approaches for each division condition: (d1) use fine particles globally, the time step is 0.00001 s; (d2) use fine particles in the refined area, use coarse particles in the other areas, and the global time step is 0.00001 s; (d3) Fine particles are used in the refined area, coarse particles are used in the other areas, and the local time steps are 0.00002 s (coarse area) and 0.00001 s (refined area), respectively.

Table 3 shows the computation time for various cases in 2D hydrostatic tests. First, in the case of division (a), the multi-resolution SPH model increases the calculation time compared to the single-resolution SPH model. The computation time under the parallel model depends on the subdomain with the largest consuming time. Therefore, in the case of division (a), the computational load of subdomain (3) is not reduced. What is worse, due to the increasing size of the particle support domain (the size of the support domain in the refined region is the same as the one in the coarse region) in the multi-resolution model, the number of particle pairs increases, as shown in Fig. 35. Particle pairs are positively correlated with computation time [80,78]. Therefore, the multi-resolution models lead to an increase in calculation time for both the global time step and local time step strategy.

In contrast, in division (b) case, the multi-resolution model significantly reduces the computation time compared to the single-resolution case. The area of subdomain (2) is the largest. In the multi-resolution model, the number of particle pairs in the subdomain (2) is significantly reduced, as shown in Fig. 36, resulting in a reduction in calculation time.

The division (c) is obtained by using the dynamic load balancing strategy. The strategy is a method to balance the computing load of each MPI subdomain. The strategy is adopted to reduce the size of the refinement region and the calculation time is also reduced. The multi-resolution model with local time step strategy reduces the time cost by 86% compared with the single resolution model.

To provide more justifications on the benefits of the new scheme in terms of CPU time and memory, the hydrostatic tests are extended to 3D hydrostatic tests with a 1.0 m width of water tank. The hydrostatic test is a common and simple case for testing numerical models. Region (0-0.2 m, 0-0.2 m, 0-0.2 m) is our area of interest. Two cases are simulated here. Case (1): the whole computational domain uses single particle resolution 0.01 m and time step 0.0001 s. Case (2): the region of interest is refined using a particle resolution of 0.01 m and time step 0.0001 s; while the other regions use a particle resolution of 0.02 m and time step 0.0002 s. Case (2) adopts the present multi-resolution model with local time stepping. Numerical simulations were tested for only 0.02 s. The initial computational domain for the two cases is divided into 64 subdomains as shown in Fig. 37. For Case (1), the entire computational domain is evenly divided. Therefore the region of interest is calculated by only one kernel. Uniform partitioning of the computing domain leads to computational load balancing. As for Case (2), the refined region is simulated using eight cores. This indicates that more computing resources are used to refine the region.

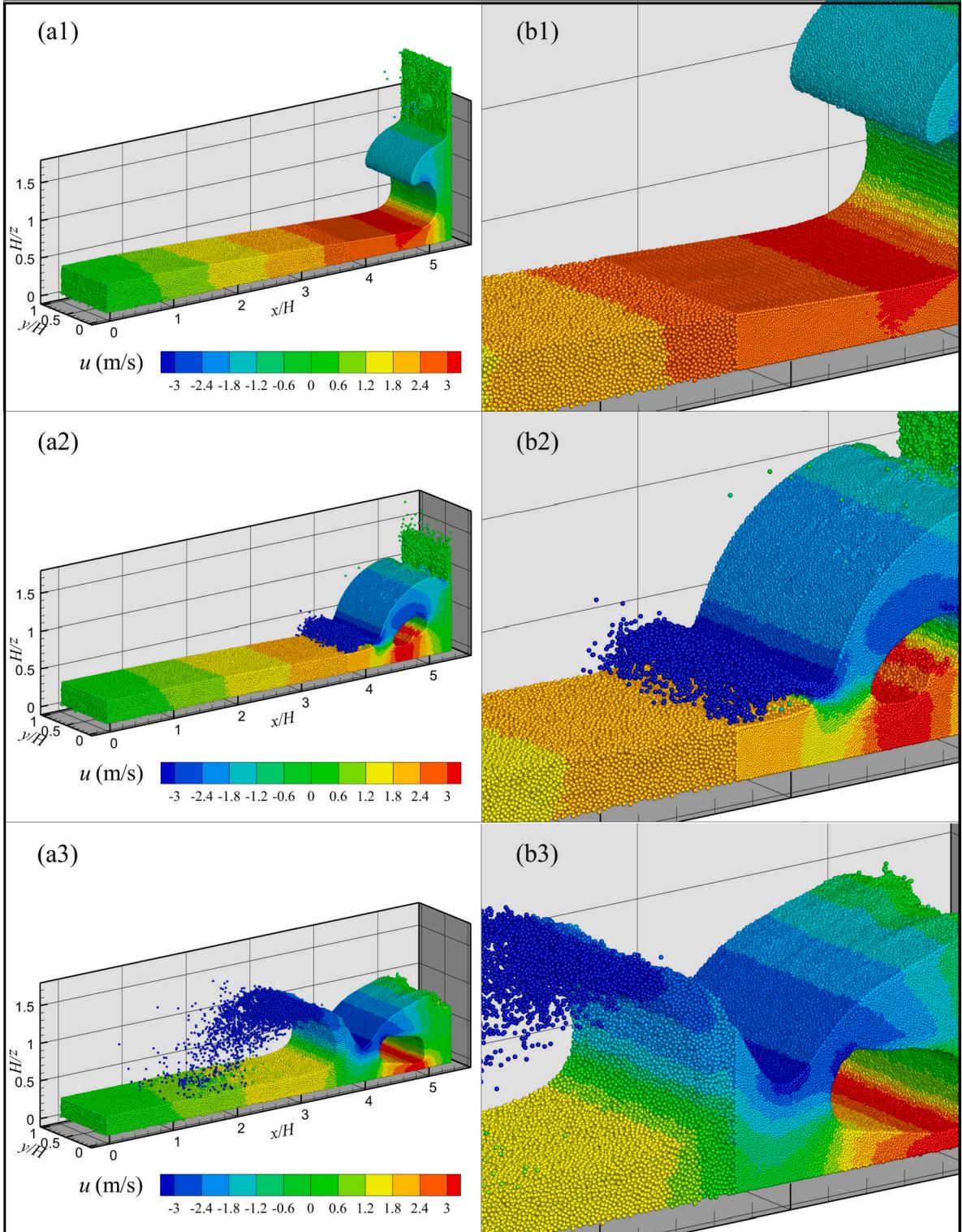


Fig. 32. Horizontal field (a) and local view (b) of the 3D dam breaking case at  $t(g/H)^{0.5} = 4.12$  (1), 4.85 (2) and 5.58 (3).

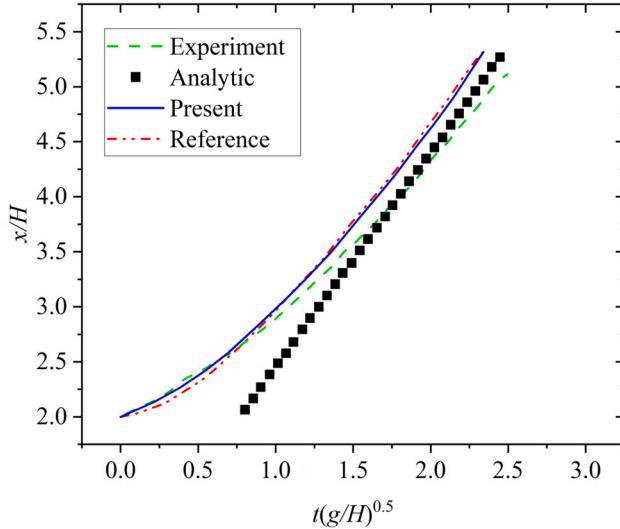


Fig. 33. Time series of water front. (Experimental data from [41]; Analytical resolution from [56]; Reference results from [10] obtained by two-phase SPH model.)

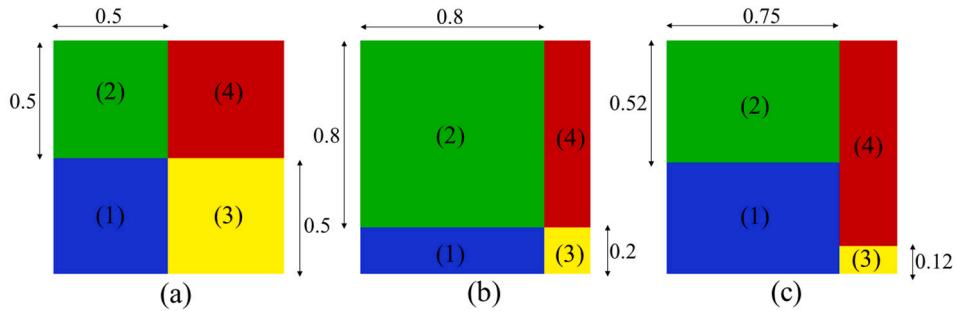


Fig. 34. Schematic of three different divisions of water tank.

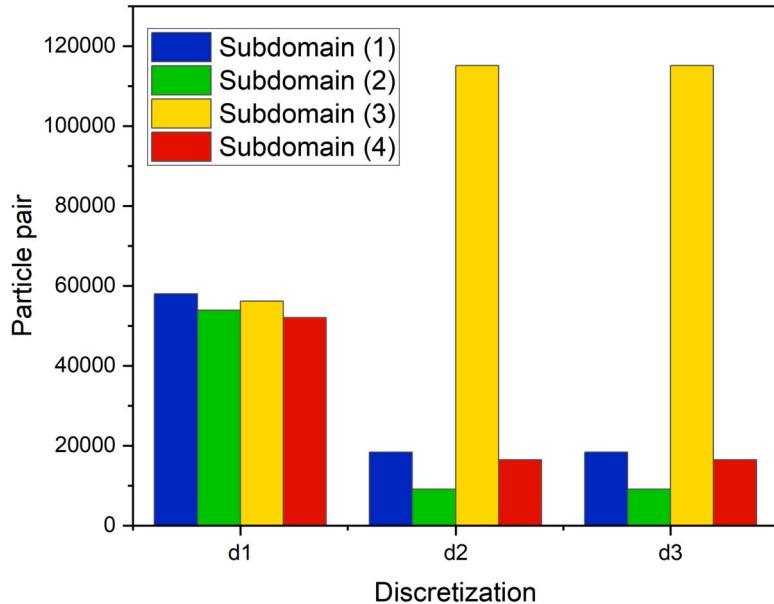


Fig. 35. The number of particle pairs of Division (a) with different discretization.

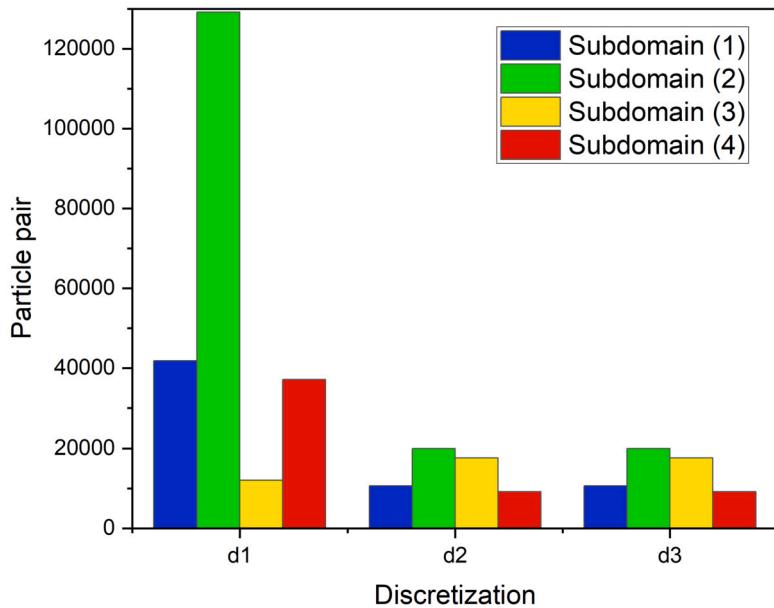


Fig. 36. The number of particle pairs of Division (b) with different discretization.

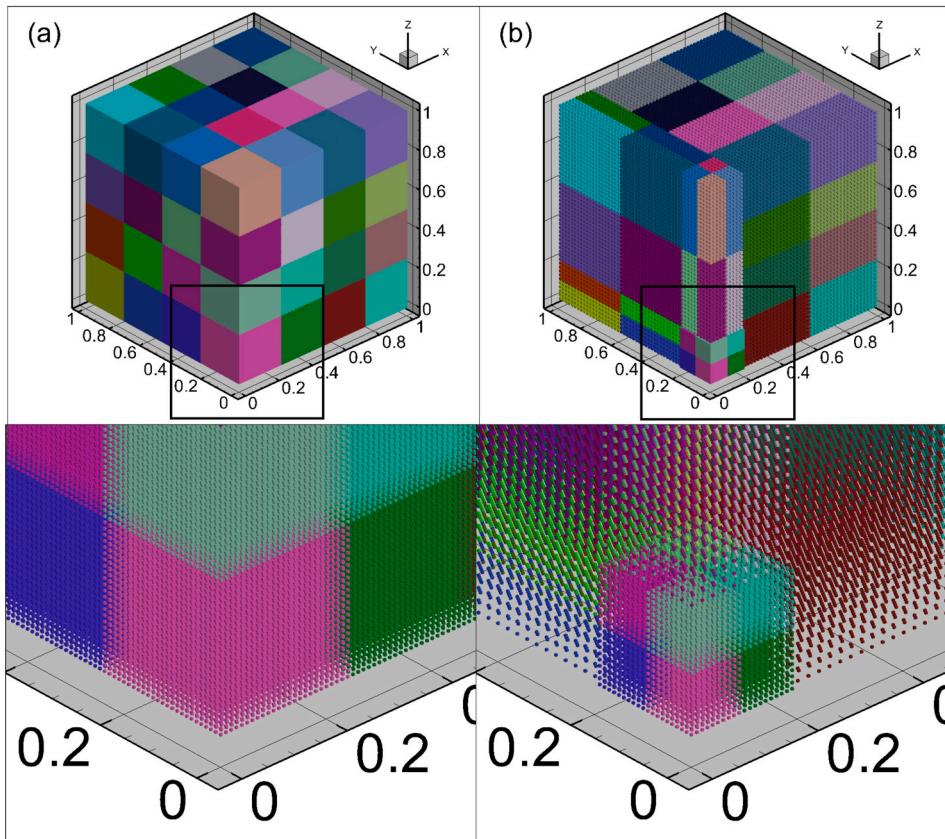


Fig. 37. Initial subdomain distributions for 3D hydrostatic tests. (a): Case (1); (b): Case (2).

**Table 4**

Time cost of different division conditions and discretization for 3D hydrostatic tests. Total particles: particle number for all subdomains; Maximum particle pairs: maximum particle pairs among subdomains; communication particles: particle number participating in MPI communication.

	Total particles	Maximum particle pairs	Communication particles	Time cost (s)
Case (1)	1213060	1082340	723816	231.23
Case (2)	188656	919702	285831	162.78

Table 4 shows the computation cost for various cases in the 3D hydrostatic tests. It can be found that the calculation time can be reduced by using the present model. As discussed in the 2D hydrostatic example, the computation time of an MPI parallel program depends on the subdomain that takes the most time, and it is proportional to the particle pairs. Utilizing the model presented in this paper has the potential to decrease the number of particle pairs, thereby resulting in a reduction of computational time. Furthermore, there's a noticeable reduction in the count of communicating particles. Within the SPH MPI-based parallel framework, inter-subdomain communication primarily involves sharing particle information. Hence, diminishing the quantity of communicating particles can lead to decreased communication time.

Overall, the current model can reduce calculation time, thanks to the reduction in the number of particle pairs. At the same time, it can also reduce communication costs because the number of communicating particles is reduced. Moreover, it can also reduce memory requirements due to a significant reduction in the total number of particles. However, there are certain limitations in how the current parallel multi-resolution models can effectively reduce computing time. Multi-resolution models can effectively reduce computational costs when low-resolution particles are used in areas with the greatest computational load. By incorporating a dynamic load balancing strategy, the distribution of computational load among subdomains can be equitably managed, thereby decreasing the refinement area and addressing this contradiction dynamically.

## 5. Conclusions

In this study, an innovative parallel multi-resolution SPH model with a local time step strategy is presented. By integrating the multi-resolution model with an MPI-based parallelization framework, the computational domain was discretized into subdomains with varying resolutions. The model under the MPI parallel framework makes each subdomain relatively independent, which facilitates the development of multi-resolution models. The transfer of particles between different subdomains involves dynamic changes in particle resolution. This is achieved by merging several fine particles flowing through the subdomain interface into coarse particles or by splitting coarse particles into fine particles. A local fourth-order Runge-Kutta time step is first introduced to implement different time steps in different resolution subdomains in the multi-resolution SPH model. Specifically, the coarse-particle subdomain adopts a larger time step compared to the time step of the fine-particle region. In this strategy, it is assumed that the time gradient of different subdomain interfaces is consistent, and the relationship between different time step values can be obtained. Through the relationship of these intermediate values, the particle communication information between subdomains is changed, and the particle information under different time steps is obtained.

The proposed model has been tested for its discretization errors at the interface of multi-resolution SPH particles. The results show that Eqs. (9) and (19) lead to smaller errors relative to other discrete schemes. Simultaneously, when employing a discrete scheme, it is essential to take into account the effect of the free surface boundary on discrete schemes. In the 2D dam breaking cases, the results prove that the current multi-resolution model can guarantee the accuracy of numerical results compared with the single-resolution model. The local time step strategy takes less computation time than the global time step strategy. However, the current model does not strictly guarantee the conservation of mass and momentum, which may be mainly attributed to the splitting and merging strategy. The current model's accuracy and reliability are further confirmed through additional numerical examples showcasing the flow around a cylinder. The oscillating drop and Taylor-Green flow cases show the accuracy, energy conservation and convergence features of the present model. The model's capacity to handle 3D challenges is also confirmed through the examination of a 3D dam breaking scenario.

Performance tests present that the current MPI-based parallel multi-resolution model is effective in reducing computing time by reasonably dividing subdomains. At the same time, the current model can reduce memory requirements and MPI communication load. Multi-resolution models lead to an increase in computational time when subdomains that take more time are refined. Properly setting subdomains and considering dynamic load balancing strategies are key factors in enabling multi-resolution models to reduce computing time.

In the future, it is necessary to further develop the dynamic loading model strategy to optimize the region partitioning scheme and reduce the computational costs. The discrete scheme needs more discussion when dealing with the free surface flows. The strategy of particle splitting and merging should be strengthened to avoid the non-conservation problem.

## CRediT authorship contribution statement

**Guixun Zhu:** Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Conceptualization. **Yongdong Cui:** Writing – review & editing, Supervision, Funding acquisition. **Boo Cheong Khoo:** Writing – review & editing, Supervision, Funding acquisition. **Siming Zheng:** Writing – review & editing, Writing – original draft,

Investigation, Formal analysis. **Zongbing Yu:** Writing – review & editing, Funding acquisition. **Yelin Gao:** Writing – review & editing, Funding acquisition.

## 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.

## Data availability

Data will be made available on request.

## Acknowledgements

G.Z., Y.C., and B.C.K. would like to acknowledge the financial support of Agency for Science, Technology and Research (FM013101). Z.Y. and Y.G. would like to acknowledge the support from The National Natural Science Foundation of China (52231011 and 52071056). Z.Y. also would like to acknowledge the support from National Natural Science Foundation of China (51979032), State Key Laboratory of Ocean Engineering (GKZD010084), and the Fundamental Research Funds for the Central Universities (Dut21ZD402 and Dut23RC(3)013).

## References

- [1] S. Adami, X. Hu, N.A. Adams, A transport-velocity formulation for smoothed particle hydrodynamics, *J. Comput. Phys.* 241 (2013) 292–307.
- [2] C. Altomare, B. Tagliaferro, J. Dominguez, T. Suzuki, G. Viccione, Improved relaxation zone method in SPH-based model for coastal engineering applications, *Appl. Ocean Res.* 81 (2018) 15–33.
- [3] M. Antuono, S. Marrone, A. Colagrossi, B. Bouscasse, Energy balance in the  $\delta$ -SPH scheme, *Comput. Methods Appl. Mech. Eng.* 289 (2015) 209–226.
- [4] D.A. Barcarolo, D. Le Touzé, G. Oger, F. De Vuyst, Adaptive particle refinement and derefinement applied to the smoothed particle hydrodynamics method, *J. Comput. Phys.* 273 (2014) 640–657.
- [5] B. Buchner, Green water on ship-type offshore structures, Ph.D. thesis, Delft University of Technology, Delft, the Netherlands, 2002.
- [6] J. Chen, J. Beraun, C. Jih, An improvement for tensile instability in smoothed particle hydrodynamics, *Comput. Mech.* 23 (1999) 279–287.
- [7] J.M. Cherfils, G. Pinon, E. Rivoalen, JOSEPHINE: a parallel SPH code for free-surface flows, *Comput. Phys. Commun.* 183 (2012) 1468–1480.
- [8] L. Chiron, G. Oger, M. De Leffe, D. Le Touzé, Analysis and improvements of adaptive particle refinement (APR) through CPU time, accuracy and robustness considerations, *J. Comput. Phys.* 354 (2018) 552–575.
- [9] A.D. Chow, B.D. Rogers, S.J. Lind, P.K. Stansby, Incompressible SPH (ISPH) with fast Poisson solver on a GPU, *Comput. Phys. Commun.* 226 (2018) 81–103.
- [10] A. Colagrossi, M. Landrini, Numerical simulation of interfacial flows by smoothed particle hydrodynamics, *J. Comput. Phys.* 191 (2003) 448–475.
- [11] A. Crespo, J. Dominguez, A. Barreiro, M. Gomez-Gesteira, B.D. Rogers, DualSPHysics, new GPU computing on SPH models, in: Proc. 6th International SPHERIC Workshop, 2011, pp. 348–354.
- [12] A.J. Crespo, J.M. Dominguez, B.D. Rogers, M. Gómez-Gesteira, S. Longshaw, R. Canelas, R. Vacondio, A. Barreiro, O. García-Feal, DualSPHysics: open-source parallel CFD solver based on smoothed particle hydrodynamics (SPH), *Comput. Phys. Commun.* 187 (2015) 204–216.
- [13] M.O. Domingues, S.M. Gomes, O. Roussel, K. Schneider, An adaptive multiresolution scheme with local time stepping for evolutionary PDEs, *J. Comput. Phys.* 227 (2008) 3758–3780.
- [14] J. Domínguez, A. Crespo, D. Valdez-Balderas, B. Rogers, M. Gómez-Gesteira, New multi-GPU implementation for smoothed particle hydrodynamics on heterogeneous clusters, *Comput. Phys. Commun.* 184 (2013) 1848–1860.
- [15] R. Fatehi, M.T. Manzari, Error estimation in smoothed particle hydrodynamics and a new scheme for second derivatives, *Comput. Math. Appl.* 61 (2011) 482–498.
- [16] J. Feldman, J. Bonet, Dynamic refinement and boundary contact forces in SPH with applications in fluid flow problems, *Int. J. Numer. Methods Eng.* 72 (2007) 295–324.
- [17] T. Gao, H. Qiu, L. Fu, A block-based adaptive particle refinement SPH method for fluid–structure interaction problems, *Comput. Methods Appl. Mech. Eng.* 399 (2022) 115356.
- [18] T. Gao, H. Qiu, L. Fu, Multi-level adaptive particle refinement method with large refinement scale ratio and new free-surface detection algorithm for complex fluid–structure interaction problems, *J. Comput. Phys.* 473 (2023) 111762.
- [19] R.A. Gingold, J.J. Monaghan, Smoothed particle hydrodynamics: theory and application to non-spherical stars, *Mon. Not. R. Astron. Soc.* 181 (1977) 375–389.
- [20] G. Graziani, M. Ranucci, R. Piva, From a boundary integral formulation to a vortex method for viscous flows, *Comput. Mech.* 15 (1995) 301–314.
- [21] X. Guo, B.D. Rogers, S. Lind, P.K. Stansby, New massively parallel scheme for incompressible smoothed particle hydrodynamics (ISPH) for highly nonlinear and distorted flow, *Comput. Phys. Commun.* 233 (2018) 16–28.
- [22] W. Hu, W. Pan, M. Rakhsa, Q. Tian, H. Hu, D. Negruț, A consistent multi-resolution smoothed particle hydrodynamics method, *Comput. Methods Appl. Mech. Eng.* 324 (2017) 278–299.
- [23] X.T. Huang, P.N. Sun, H.G. Lyu, A.M. Zhang, Numerical investigations on bionic propulsion problems using the multi-resolution Delta-plus SPH model, *Eur. J. Mech. B, Fluids* 95 (2022) 106–121.
- [24] M. Ihmsen, N. Akinci, M. Becker, M. Teschner, A parallel SPH implementation on multi-core CPUs, in: Computer Graphics Forum, Wiley Online Library, 2011, pp. 99–112.
- [25] T. Imamura, K. Suzuki, T. Nakamura, M. Yoshida, Acceleration of steady-state lattice Boltzmann simulations on non-uniform mesh using local time step method, *J. Comput. Phys.* 202 (2005) 645–663.
- [26] Z. Ji, L. Fu, X.Y. Hu, N.A. Adams, A new multi-resolution parallel framework for SPH, *Comput. Methods Appl. Mech. Eng.* 346 (2019) 1156–1178.
- [27] J.W. Kaiser, N. Hoppe, S. Adami, N.A. Adams, An adaptive local time-stepping scheme for multiresolution simulations of hyperbolic conservation laws, *J. Comput. Phys. X* 4 (2019) 100038.
- [28] A. Khayyer, H. Gotoh, Y. Shimizu, Comparative study on accuracy and conservation properties of two particle regularization schemes and proposal of an optimized particle shifting scheme in ISPH context, *J. Comput. Phys.* 332 (2017) 236–256.
- [29] A. Khayyer, H. Gotoh, Y. Shimizu, K. Gotoh, On enhancement of energy conservation properties of projection-based particle methods, *Eur. J. Mech. B, Fluids* 66 (2017) 20–37.

- [30] A. Khayyer, Y. Shimizu, H. Gotoh, S. Hattori, Multi-resolution ISPH-SPH for accurate and efficient simulation of hydroelastic fluid-structure interactions in ocean engineering, *Ocean Eng.* 226 (2021) 108652.
- [31] A. Khayyer, Y. Shimizu, T. Gotoh, H. Gotoh, Enhanced resolution of the continuity equation in explicit weakly compressible SPH simulations of incompressible free-surface fluid flows, *Appl. Math. Model.* 116 (2023) 84–121.
- [32] P. Koumoutsakos, A. Leonard, High-resolution simulations of the flow around an impulsively started cylinder using vortex methods, *J. Fluid Mech.* 296 (1995) 1–38.
- [33] S.J. Lind, R. Xu, P.K. Stansby, B.D. Rogers, Incompressible smoothed particle hydrodynamics for free-surface flows: a generalised diffusion-based algorithm for stability and validations for impulsive flows and propagating waves, *J. Comput. Phys.* 231 (2012) 1499–1523.
- [34] J. Liu, X. Yang, Z. Zhang, M. Liu, A massive MPI parallel framework of smoothed particle hydrodynamics with optimized memory management for extreme mechanics problems, *Comput. Phys. Commun.* 295 (2024) 108970.
- [35] L. Liu, X. Li, F.Q. Hu, Nonuniform time-step Runge–Kutta discontinuous Galerkin method for computational aeroacoustics, *J. Comput. Phys.* 229 (2010) 6874–6897.
- [36] L. Liu, X. Li, F.Q. Hu, Nonuniform-time-step explicit Runge–Kutta scheme for high-order finite difference method, *Comput. Fluids* 105 (2014) 166–178.
- [37] M. Liu, W. Xie, G. Liu, Modeling incompressible flows using a finite particle method, *Appl. Math. Model.* 29 (2005) 1252–1270.
- [38] M. Liu, Z. Zhang, Smoothed particle hydrodynamics (SPH) for modeling fluid–structure interactions, *Sci. China, Phys. Mech. Astron.* 62 (2019) 984701.
- [39] M. Liu, Z. Zhang, D. Feng, A density-adaptive SPH method with kernel gradient correction for modeling explosive welding, *Comput. Mech.* 60 (2017) 513–529.
- [40] M.B. Liu, G.R. Liu, Smoothed particle hydrodynamics (SPH): an overview and recent developments, *Arch. Comput. Methods Eng.* 17 (2010) 25–76.
- [41] L. Lobovský, E. Botia-Vera, F. Castellana, J. Mas-Soler, A. Souto-Iglesias, Experimental investigation of dynamic pressure loads during dam break, *J. Fluids Struct.* 48 (2014) 407–434.
- [42] L.B. Lucy, A numerical approach to the testing of the fission hypothesis, *Astron. J.* 82 (1977) 1013–1024.
- [43] M. Luo, A. Khayyer, P. Lin, Particle methods in ocean and coastal engineering, *Appl. Ocean Res.* 114 (2021) 102734.
- [44] C. Lüthi, M. Afrasiabi, M. Bambach, An adaptive smoothed particle hydrodynamics (SPH) scheme for efficient melt pool simulations in additive manufacturing, *Comput. Math. Appl.* 139 (2023) 7–27.
- [45] H.G. Lyu, P.N. Sun, Further enhancement of the particle shifting technique: towards better volume conservation and particle distribution in SPH simulations of violent free-surface flows, *Appl. Math. Model.* 101 (2022) 214–238.
- [46] H.G. Lyu, P.N. Sun, J.M. Miao, A.M. Zhang, 3D multi-resolution SPH modeling of the water entry dynamics of free-fall lifeboats, *Ocean Eng.* 257 (2022) 111648.
- [47] S. Marrone, M. Antuono, A. Colagrossi, G. Colicchio, D. Le Touzé, G. Graziani,  $\delta$ -SPH model for simulating violent impact flows, *Comput. Methods Appl. Mech. Eng.* 200 (2011) 1526–1542.
- [48] S. Marrone, A. Colagrossi, M. Antuono, G. Colicchio, G. Graziani, An accurate SPH modeling of viscous flows around bodies at low and moderate Reynolds numbers, *J. Comput. Phys.* 245 (2013) 456–475.
- [49] D.D. Meringolo, F. Aristodemo, P. Veltri, SPH numerical modeling of wave–perforated breakwater interaction, *Coast. Eng.* 101 (2015) 48–68.
- [50] J.J. Monaghan, A. Rafiee, A simple SPH algorithm for multi-fluid flow with high density ratios, *Int. J. Numer. Methods Fluids* 71 (2013) 537–561.
- [51] X. Ni, W. Feng, S. Huang, X. Zhao, X. Li, Hybrid SW-NS SPH models using open boundary conditions for simulation of free-surface flows, *Ocean Eng.* 196 (2020) 106845.
- [52] G. Oger, M. Doring, B. Alessandrini, P. Ferrant, An improved SPH method: towards higher order convergence, *J. Comput. Phys.* 225 (2007) 1472–1492.
- [53] G. Oger, D. Le Touzé, D. Guibert, M. De Leffe, J. Biddiscombe, J. Soumagne, J.G. Piccinali, On distributed memory MPI-based parallelization of SPH codes in massive HPC context, *Comput. Phys. Commun.* 200 (2016) 1–14.
- [54] Y. Reyes López, D. Roose, C. Recarey Morfa, Dynamic particle refinement in SPH: application to free surface flow and non-cohesive soil simulations, *Comput. Mech.* 51 (2013) 731–741.
- [55] F. Ricci, R. Vacondio, A. Tafuni, Multiscale smoothed particle hydrodynamics based on a domain-decomposition strategy, *Comput. Methods Appl. Mech. Eng.* 418 (2024) 116500.
- [56] A. Ritter, Die Fortpflanzung der Wasserwellen, *Z. Ver. Dtsch. Ing.* 36 (1892) 947–954.
- [57] A. Skillen, S. Lind, P.K. Stansby, B.D. Rogers, Incompressible smoothed particle hydrodynamics (SPH) with reduced temporal noise and generalised Fickian smoothing applied to body–water slam and efficient wave–body interaction, *Comput. Methods Appl. Mech. Eng.* 265 (2013) 163–173.
- [58] P. Sun, A. Colagrossi, S. Marrone, A. Zhang, The  $\delta$ -plus-SPH model: simple procedures for a further improvement of the SPH scheme, *Comput. Methods Appl. Mech. Eng.* 315 (2017) 25–49.
- [59] P.N. Sun, A. Colagrossi, A.M. Zhang, Numerical simulation of the self-propulsive motion of a fishlike swimming foil using the  $\delta+$ -SPH model, *Theor. Appl. Mech. Lett.* 8 (2018) 115–125.
- [60] N. Trask, M. Maxey, K. Kim, M. Perego, M.L. Parks, K. Yang, J. Xu, A scalable consistent second-order SPH for unsteady low Reynolds number flows, *Comput. Methods Appl. Mech. Eng.* 289 (2015) 155–178.
- [61] R. Vacondio, C. Altomare, M. De Leffe, X. Hu, D. Le Touzé, S. Lind, J.C. Marongiu, S. Marrone, B.D. Rogers, A. Souto-Iglesias, Grand challenges for smoothed particle hydrodynamics numerical schemes, *Comput. Part. Mech.* 8 (2021) 575–588.
- [62] R. Vacondio, B.D. Rogers, P.K. Stansby, P. Mignosa, Variable resolution for SPH in three dimensions: towards optimal splitting and coalescing for dynamic adaptivity, *Comput. Methods Appl. Mech. Eng.* 300 (2016) 442–460.
- [63] T. Verbrughe, J.M. Domínguez, A.J. Crespo, C. Altomare, V. Stratigaki, P. Troch, A. Kortenhaus, Coupling methodology for smoothed particle hydrodynamics modelling of non-linear wave–structure interactions, *Coast. Eng.* 138 (2018) 184–198.
- [64] D. Violeau, A. Leroy, On the maximum time step in weakly compressible SPH, *J. Comput. Phys.* 256 (2014) 388–415.
- [65] D. Violeau, A. Leroy, Optimal time step for incompressible SPH, *J. Comput. Phys.* 288 (2015) 119–130.
- [66] D. Violeau, B.D. Rogers, Smoothed particle hydrodynamics (SPH) for free-surface flows: past, present and future, *J. Hydraul. Res.* 54 (2016) 1–26.
- [67] P.P. Wang, Z.F. Meng, A.M. Zhang, F.R. Ming, P.N. Sun, Improved particle shifting technology and optimized free-surface detection method for free-surface flows in smoothed particle hydrodynamics, *Comput. Methods Appl. Mech. Eng.* 357 (2019) 112580.
- [68] F. Xu, J. Wang, Y. Yang, L. Wang, Z. Dai, R. Han, On methodology and application of smoothed particle hydrodynamics in fluid, solid and biomechanics, *Acta Mech. Sin.* 39 (2023) 722185.
- [69] 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 (2009) 6703–6725.
- [70] Q. Yang, F. Xu, Y. Yang, Z. Dai, J. Wang, A GPU-accelerated adaptive particle refinement for multi-phase flow and fluid–structure coupling SPH, *Ocean Eng.* 279 (2023) 114514.
- [71] X. Yang, S.C. Kong, Adaptive resolution for multiphase smoothed particle hydrodynamics, *Comput. Phys. Commun.* 239 (2019) 112–125.
- [72] X. Yang, S.C. Kong, M. Liu, Q. Liu, Smoothed particle hydrodynamics with adaptive spatial resolution (SPH-ASR) for free surface flows, *J. Comput. Phys.* 443 (2021) 110539.
- [73] A.m. Zhang, P.n. Sun, F.r. Ming, A. Colagrossi, Smoothed particle hydrodynamics and its applications in fluid–structure interactions, *J. Hydrodynam., Ser. B* 29 (2017) 187–216.
- [74] C. Zhang, M. Rezavand, X. Hu, A multi-resolution SPH method for fluid–structure interactions, *J. Comput. Phys.* 429 (2021) 110028.
- [75] C. Zhang, M. Rezavand, Y. Zhu, Y. Yu, D. Wu, W. Zhang, J. Wang, X. Hu, SPHinXsys: an open-source multi-physics and multi-resolution library based on smoothed particle hydrodynamics, *Comput. Phys. Commun.* 267 (2021) 108066.

- [76] H. Zhang, Z. Zhang, F. He, M. Liu, Numerical investigation on the water entry of a 3D circular cylinder based on a GPU-accelerated SPH method, *Eur. J. Mech. B, Fluids* 94 (2022) 1–16.
- [77] Z.X. Zhao, G. Bilotta, Q.E. Yuan, Z.X. Gong, H. Liu, Multi-GPU multi-resolution SPH framework towards massive hydrodynamics simulations and its applications in high-speed water entry, *J. Comput. Phys.* 490 (2023) 112339.
- [78] G. Zhu, Towards the development of smoothed particle hydrodynamics model for oscillating water column devices, Ph.D. thesis, University of Plymouth, 2023.
- [79] G. Zhu, J. Hughes, S. Zheng, D. Greaves, Development of a two-dimensional coupled smoothed particle hydrodynamics model and its application to nonlinear wave simulations, *Comput. Fluids* 266 (2023) 106044.
- [80] G. Zhu, J. Hughes, S. Zheng, D. Greaves, A novel MPI-based parallel smoothed particle hydrodynamics framework with dynamic load balancing for free surface flow, *Comput. Phys. Commun.* 284 (2023) 108608.