Computers and Fluids 274 (2024) 106224

Contents lists available at ScienceDirect

Computers and Fluids

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

A consistent second order ISPH for free surface flow

Ningbo Zhang [a], Shiqiang Yan [a,*], Qingwei Ma [a], Abbas Khayyer [b], Xiaohu Guo [c], Xing Zheng [d]

[a] Department of Engineering, School of Science and Technology, City, University of London, London, UK
[b] Department of Civil and Earth Resources Engineering, Kyoto University, Kyoto, Japan
[c] Hartree Centre, Science and Technology Facilities Council, Daresbury Laboratory, Warrington, UK
[d] College of Shipbuilding Engineering, Harbin Engineering University, Harbin, Heilongjiang, China

ARTICLE INFO

Keywords:
SPH
Consistent second order ISPH
Convergence
Mass/energy conservation
Free surface flow
QSFDI

ABSTRACT

The Incompressible Smoothed Particle Hydrodynamics (ISPH) is now a popular numerical method for modelling free surface flows, in particular the breaking waves and violent wave-structures interaction. The ISPH requires the projection approach, leading to solving a pressure Poisson's equation (PPE). Although the accuracy and convergence of the numerical scheme to discretise the Laplacian operator involved in PPE is critical for securing a satisfactory solution of the PPE, the overall performance of the ISPH is also influenced by other key numerical implementations, including (1) estimation of the viscous terms; (2) calculation of the velocity divergence; (3) discretisation of the boundary conditions for the PPE; and (4) evaluation of the pressure gradient. In our previous paper [29], the quadratic semi-analytical finite difference interpolation scheme (QSFDI), which has a leading truncation error at third order derivatives, has been adopted to discretise the Laplacian operator. In this paper, the QSFDI will be adopted, not only for discretising the Laplacian operator, but also for approximating viscous terms, velocity divergence, boundary conditions and pressure gradient. The performance of the newly formulated consistent second order ISPH is assessed by various cases including the oscillating liquid drop, the wave propagation, and the liquid sloshing. The results do not only demonstrate a second order convergence over a limited range of conditions and a higher computational efficiency, i.e., requiring less computational time to achieve the same accuracy, but also show a better mass/energy conservation property and capacity of reproducing a smooth pressure field, than other ISPH models considered in this study.

1. Introduction

The Incompressible Smoothed Particle Hydrodynamics (ISPH) proposed by Cummins and Rudman [1] and Shao and Lo [2] is an important stream of the Smoothed Particle Hydrodynamics (SPH) [3,4], in which the computational domain is spatially discretised by particles that are either fixed (Eulerian particles) [5] or moving with a velocity, such as with the material velocity (Lagrangian particles) [6]. The Lagrangian ISPH has been shown to have some advantages on modelling problems involving large deformations, where the conventional mesh-based methods may suffer from severe mesh distortions in the Lagrangian applications (e.g. [7]) or undesirable energy dissipations due to the numerical approximation of convective terms in the Eulerian or Arbitrary Lagrangian Eulerian (ALE) applications, e.g. [8]. Compared with the traditional SPH (the weakly compressible SPH), the ISPH has been shown to have several superiorities in terms of convergence [9,10], stability and accuracy [6,11] as well as volume conservation [12]. The ISPH has become a popular method for modelling free surface flows, in particular the breaking waves and violent wave-structure interactions [13–22].

The ISPH solves the incompressible Navier-Stokes (NS) equation using the projection method [23], where the pressure is obtained by solving a pressure Poisson Equation (PPE). There are two approaches to derive the PPEs. In the first approach, the PPE is obtained by enforcing the density invariance [2,22,24,25]. The right-hand side (RHS) of the PPE is termed by the time derivative of the intermediate density. In the second approach, the PPE is derived to satisfy a divergence free condition and, therefore, the RHS of the PPE is formulated as the divergence of the intermediate velocity [13,26]. For both approaches, additional terms may be introduced to the RHS of the PPE to improve the overall performance of the ISPH. For example, Khayyer and Gotoh [27] introduced an error compensation term to improve the volume conservation. Zhang et al. [28,29] implemented a density variation term in the divergence-free PPE to form a blended/hybird variant of the PPE for

* Corresponding author.
E-mail address: shiqiang.yan.1@city.ac.uk (S. Yan).

https://doi.org/10.1016/j.compfluid.2024.106224
Received 22 November 2022; Received in revised form 13 December 2023; Accepted 26 February 2024
Available online 28 February 2024

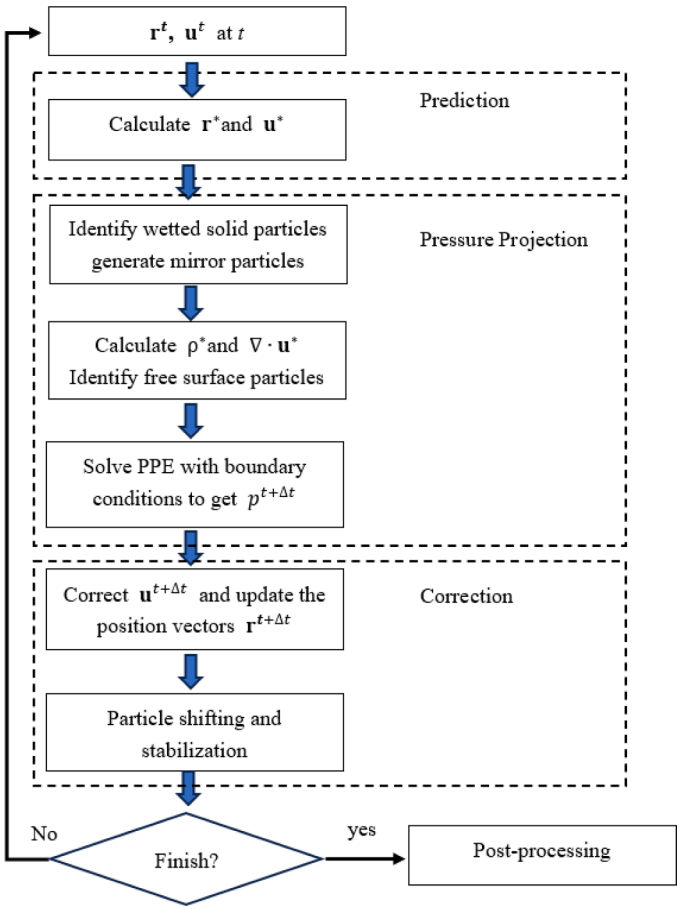

Fig. 1. Flowchart of the projection-based ISPH method.

improving the numerical stability. These developments have delivered a promising performacne of the ISPH on modelling free surface flows.

In addition to developing different forms of the PPE, discretising the PPE is critical and often dominates the accuracy and the convergence of the ISPH methods [29,30]. There are also two types of approaches. One of them is to discretise the PPE after converting it into a weak form, which avoids the discretisation of high order derivatives. This approach was originally developed for the meshless local Petrov-Galerkin method, e.g. the MLPG-R [31–36]. It was also extended to the ISPH by Zheng et al. [6]. Despite the incorporation of a linear semi-analytical finite difference interpolation scheme (SFDI) [33] for the required interpolation and gradient calculation, a second order convergence has been observed in modelling violent liquid sloshing [6,30]. Another approach is to directly approximate the second and first order derivatives involved in the PPE. However, the Lagrangian nature of the particle movement results in an irregular or random particle distribution even if a uniform and regular particle distribution is used in the initial state. This brings a grand challenge to the Lagrangian particle methods, i.e., how to secure a higher order convergence and consistency for irregular particle distributions [37,38]. The problem of particle irregularity may be eliminated by adopting the Eulerian-based ISPH [5] or a hybrid approach coupling the Eulerian and Lagrangian SPH using the domain-decomposition strategy [39]. Nevertheless, most of the related research focuses on the developments of high order kernel correction [26,37], high order discretisation schemes for the Laplacian [40–45] and the gradient [11, 46] operators, whereas keeps the advantages of the Lagrangian particle methods. More reviews on the high order Laplacian discretisation can be found in our previous work such as Refs. [30,45]. These developments are generally based on the Taylor's expansion [26,40,41], moving least squares method (MLS) [43] or weighted least squared method (WLS) [24,25,44,47]. Recently, Yan et al. [45] developed a quadratic semi-analytical finite difference interpolation scheme (QSFDI), which is based on the principle of the linear SFDI developed by Ma [33]. The QSFDI includes the schemes for interpolation, gradient and Laplacian estimation, which are derived consistently from the same second order Taylor's expansion and have a leading truncation error at the third order. All these schemes in the QSFDI are named as second-order schemes in this paper. The patch test carried out by Yan et al. [45], in which particles are randomly distributed, confirmed that (1) the QSFDI has a first-order convergence and consistency for directly estimating the Laplacian of a function; (2) when the QSFDI is applied to discretise the Laplacian of a Poisson equation formulated by a known function, the solutions exhibit a second-order convergence rate [45]; and (3) the QSFDI has a higher computational efficiency to achieve the same order of accuracy and mathematical consistency, compared with the second order MLS and the second order WLS scheme developed by Tamai et al. [43,44]. Zhang et al. [29] applied the QSFDI to discretising the Laplacian in the PPE of the ISPH (referred to as the ISPH_QSFDI hereafter). They have demonstrated that, compared with the classic ISPH [28], the ISPH_QSFDI results in a better accuracy with the same number of particles or requires a shorter computational time to achieve the same accuracy for wave-structure interaction problems. Nevertheless, it does not show a second order convergence as what has been observed in the theoretical patch test [45]. This is because the other differential terms in the ISPH_QSFDI, such as the velocity divergence at the RHS of the PPE, the viscous terms, the Neumann conditions and the pressure gradient, are dealt with by the linear SFDI.

In this paper, the QSFDI is consistently applied to both the Laplacian and the discretisation of other differential terms involved in the ISPH procedure. Compared with the ISPH_QSFDI, although the degree of the accuracy of the Laplacian discretisation remains the same, i.e. a first-order accuracy, the accuracy of the gradient discretisation and the treatment of the boundary condition are improved to a second-order accuracy. The newly formulated ISPH model is referred to as the Consistent Second Order ISPH (ISPH_CQ) hereafter. Its accuracy, convergence, efficiency, mass and energy conservation, the capability of reproducing a smooth pressure field and capturing a violent pressure will be examined for various free surface problems, including the oscillating liquid drop, the wave propagation, and the liquid sloshing.

2. Mathematical model and numerical implementations

2.1. Governing equations, boundary conditions and projection method

In the present ISPH, the fluid is assumed to be incompressible and is governed by the incompressible NS equation and the continuity equation,

$$\frac{D\mathbf{u}}{Dt} = -\frac{1}{\rho}\nabla p + \mathbf{g} + \nu\nabla^2\mathbf{u} \tag{1}$$

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

where, D/Dt is the material derivative following the fluid particle and ∇ is the spatial differential operator; ρ is the fluid density; $\mathbf{u}$ is the particle velocity; p is the pressure; $\boldsymbol{g}$ is the gravitational acceleration; and ν is the kinematic viscosity. On the solid boundaries, the following Neumann boundary conditions for velocity and pressure, respectively, are imposed [30],

$$\mathbf{u}\cdot\mathbf{n} = \mathbf{U}\cdot\mathbf{n} \tag{3}$$

$$\mathbf{n}\cdot\nabla p = \rho(\mathbf{n}\cdot\mathbf{g} - \mathbf{n}\cdot\ddot{\mathbf{U}}) \tag{4}$$

in which $\mathbf{n}$ is the unit normal vector of the solid boundary; $\mathbf{U}$ and $\ddot{\mathbf{U}}$ are the velocity and acceleration of the solid boundary. Eq. (3) specifies a no-penetration boundary condition but does not enforce a no-slip. This is

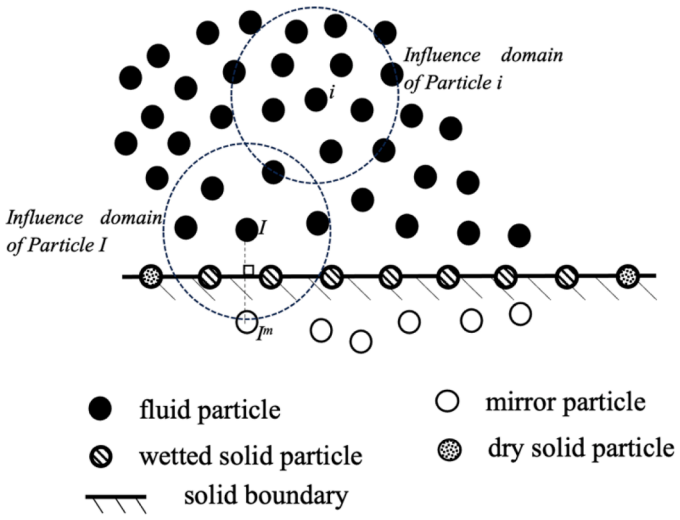

Fig. 2. Illustration of fluid particles, solid particles and mirror particles.

acceptable if the boundary layer is not the focus of the simulation. On the free surface,

$$p = 0 \tag{5}$$

The governing equations and boundary conditions are solved using the projection method, which is illustrated by the flowchart shown in Fig. 1.

Assuming the position ($\mathbf{r}^t$) and the velocity ($\mathbf{u}^t$) of the particles at time t are known, the pressure, velocity and position of the particles at the new time step, $t + \Delta t$, where Δt is the time step size, can be predicted using three stages,

(1) Prediction (intermediate) stage

Calculate the intermediate fluid velocity $\mathbf{u}^*$ at the particle i using

$$\mathbf{u}_i^* = \mathbf{u}_i^t + \left(\mathbf{g} + \nu\nabla^2\mathbf{u}_i^t\right)\Delta t \tag{6}$$

and the intermediate position vector $\mathbf{r}^*$ of the particle i by

$$\mathbf{r}_i^* = \mathbf{r}_i^t + \mathbf{u}_i^*\Delta t \tag{7}$$

During this stage, the numerical discretisation is required for finding the viscous stress $\nu\nabla^2\mathbf{u}_i^t$ in Eq. (6).

(2) Pressure projection stage

Eqs. (1) and (2) lead to a PPE $\nabla^2 p^{t+\Delta t} = \frac{\rho\nabla\cdot\mathbf{u}^*}{\Delta t}$ for the pressure at the new time step $p^{t+\Delta t}$. As discussed in the Introduction, this approach is derived to satisfy a divergence free condition [13,26]. Alternatively, one may derive a PPE to enforce the density invariance, i.e., $\nabla^2 p_i^{t+\Delta t} = \frac{\rho_i\nabla\cdot\mathbf{u}_i^*}{\Delta t}$ [2,22,24,25]. Existing literatures, e.g. Xu et al. [48], revealed that the divergence-free approach may suffer from a numerical instability caused by ill-distributed particles following the Lagrangian movement of particles; whereas the density-invariance approach may overcome such problem, but was shown to give pressure predictions with an extremely high pressure fluctuation, and to have difficulty in securing satisfactory mass/energy conservations. Although these problems can be partially solved by introducing an effective particle shifting (e.g. [14,18]), we apply the following blended form to all internal fluid particles in this paper, such as the particle i illustrated in Fig. 2,

$$\nabla^2 p_i^{t+\Delta t} = \alpha\frac{\rho_i - \rho_i^*}{\Delta t^2} + (1-\alpha)\frac{\rho_i\nabla\cdot\mathbf{u}_i^*}{\Delta t} \tag{8}$$

which shows a better numerical stability in the cases with violent free surface flows based on our previous numerical investigations [6,28,29,34] compared with the conventional divergence-free approach. According to the numerical investigation in Refs. [28,29], the blending coefficient α is taken as 0.01. Numerical algorithms are required to discretise the Laplacian in the left-hand side (LHS) of Eq. (8), the intermediate density $\rho^*$and the velocity divergence at the RHS of Eq. (8), using the corresponding physical quantities at particles in the influence domain, Ω_i, which is centred at the particle i and has a specific radius. The intermediate density ρ^* at the particle i is calculated by $\sum_{j=1}^{N} m_j W(\mathbf{r}_{ij}^*)$ where N is the number of the particles in the influence domain (referred to as the neighbouring particles) of the particle i (Fig. 2), j is the local particle number ranging from 1 to N, m_j is the particle mass of the local particle j and $W(\mathbf{r}_{ij})$ is a kernel function corresponding to the position vector $\mathbf{r}_{ij} = \mathbf{r}_i - \mathbf{r}_j$.

For fluid particles near the solid boundary, e.g. the particle I illustrated in Fig. 2, boundary treatments are required. As other SPH works, the solid boundaries are represented by solid particles and mirror particles. Following our previous work, e.g. [28,29], the solid particles are generated in the beginning of the simulation and are distributed on the entire solid boundaries, however, only those located within the influence domain of the fluid particles (referred to as the wetted solid particles) are involved in the simulation. At each time step, mirror particles are generated instantaneously by projecting fluid particles near the boundaries, that are determined using the distance to the solid boundary being smaller than the radius of the influence domain, to the outer side of the surface. At the mirror particle I^m, the density and the velocity components tangential to the solid boundary are taken as the same as those at the corresponding fluid particle I, the velocity component normal to the solid boundary is taken as the negative normal velocity component at the internal particle I to ensure the satisfactory of Eq. (3).

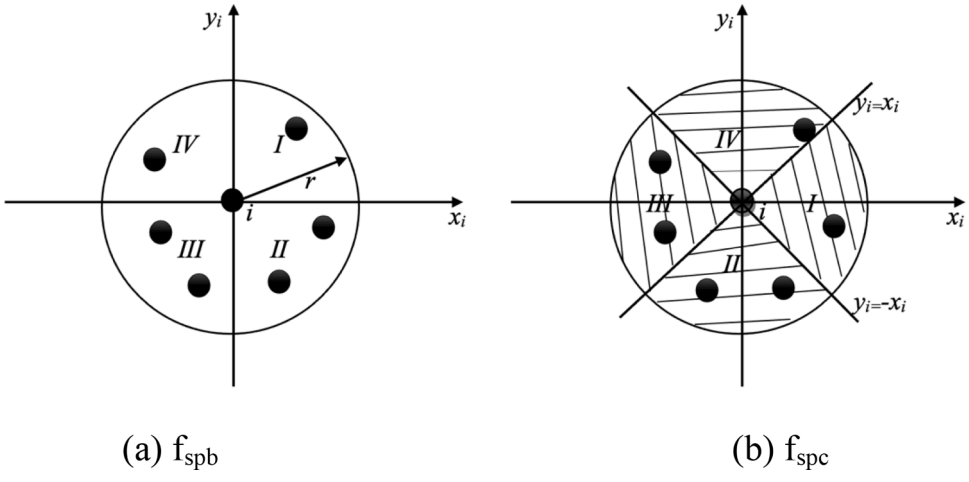

Fig. 3. Illustration of definitions of auxiliary functions for the free surface particle identification.

The instantaneous generation of the mirror particles and identification of the wetted solid particles can minimise the number of particles involved in solving the PPE and thus maximise the computational efficiency. The wetted solid and mirror particles are involved in the calculation of the particle density and the velocity divergence at the RHS of Eq. (8). The Neumann boundary condition Eq. (4) is imposed on the wetted solid particles and relevant gradient discretisation is implemented only using the fluid and wetted solid particles in their influence domain. For this purpose, appropriate numerical scheme to discretise the gradient operator is required.

For modelling the free surface flows, it is critical to identify the free surface particles, on which Eq. (5) is imposed. For this purpose, Zheng et al. [6] has developed an effective approach based on the particle density ratio and the particle distribution. In this approach, three auxiliary functions relevant to the particle distribution are defined. The first one $f_{spa} = 1$ if there are more than one free surface particles in the influence domain of the particle i, and zero otherwise. The 2nd one $f_{spb} = 1$ if particles are detected in all quadrants of the particle i defined using the local x_i and y_i axis (see Fig. 3(a)), and $f_{spb} = 0$ otherwise. The third one $f_{spc} = 1$ if particles are detected in all quadrants of the particle i divided by $y_i = x_i$ and $y_i = -x_i$ (see Fig. 3(b)), and $f_{spc} = 0$ otherwise. The particle i can be identified as a free surface particle if (1) no inner particles are observed in the its influence domain; or (2) if the particle density ratio is not higher than 0.9, either $f_{spa} = 1$ or particles are observed in no more than 2 quadrants of the particle i defined for both f_{spb} and f_{spc}; or (3) if the particle density ratio is higher than 0.9 and $f_{spa} = 1$, either f_{spb} or $f_{spc} = 0$.

(3) Correction stage

After the PPE with the boundary conditions is solved, the velocity and position vectors of particle i at $t + \Delta t$ are corrected using

$$\mathbf{u}_i^{t+\Delta t} = \mathbf{u}_i^* - \frac{1}{\rho}\nabla p_i^{t+\Delta t}\Delta t \tag{9}$$

$$\mathbf{r}_i^{t+\Delta t} = \mathbf{r}_i^t + \frac{\mathbf{u}_i^t + \mathbf{u}_i^{t+\Delta t}}{2}\Delta t \tag{10}$$

During this stage, numerical schemes to calculate the pressure gradient in Eq. (9) are required. After the velocity at and the position vector of the particles are updated using Eqs. (9) and (10). The hybrid particle stabilization and shifting scheme proposed by Zhang et al. [28] are used to further improve the stabilization of the PPE solution.

As shown, the temporal scheme used to update the velocity at the particle i is specified by Eqs. (6) and (9), that yield $\mathbf{u}_i^{t+\Delta t} = \mathbf{u}_i^t + \mathbf{a}_i^t\Delta t$ with the acceleration $\mathbf{a}_i^t = \left(\mathbf{g} + \nu\nabla^2\mathbf{u}_i^t - \frac{1}{\rho}\nabla p_i^{t+\Delta t}\right)$; that used to update the position vector is Eq. (10), which results in $\mathbf{r}_i^{t+\Delta t} = \mathbf{r}_i^t + \mathbf{u}_i^t\Delta t + \frac{1}{2}\mathbf{a}_i^t\Delta t^2$. Both of them are widely implemented in the ISPH applications. The leading truncation errors of estimating the velocity and the positions are, respectively, $O(\Delta t)$ and $O(\Delta t^2)$ due to the error of estimating the acceleration of the fluid particle (E_a) and the fluid velocity $\mathbf{u}_i^t$. The accuracy of the temporal scheme may be improved by using other schemes, e.g. the Runge-Kutta method. However, that does not affect the degrees of the accuracy unless an additional term is introduced to compensate the leading error $E_a\Delta t$. Alternatively, this work aims to improve the accuracy of the time integration through improving the accuracy of estimating the acceleration of the fluid particle, i.e., reducing E_a. This is achieved by improving the accuracy of spatial discretisation of the quantities and the solution to the PPE.

2.2. Consistent second order ISPH (ISPH_CQ)

As stated above, the QSFDI is consistently applied for all Laplacian and gradient operations outlined above, yielding a second order gradient estimation and a first-order Laplacian discretisation. Yan et al. [45] presented the detailed derivation of the QSFDI and relevant patch tests using theoretical functions and ramdonly distributely particles; Zhang et al. [29] gave a summary of the QSFDI and applied it to descritising the Laplacian of the PPE. For completeness, necessary description of the QSFDI is presented here. More detailed derivation is duplicated from Refs. [29,45], and presented in the Appendix.

For each neighbouring particle j with a position vector of $\mathbf{r}_j$, that is located inside the influence domain of the particle i at $\mathbf{r}_i$, a function P can be expressed as the Taylor's expansion,

$$P_j - P_i = \mathbf{r}_{ji}^{\mathrm{T}}\nabla P_i + \frac{1}{2}\left(\mathbf{r}_{ji}^{(2s)}\right)^{\mathrm{T}}\nabla^{(2s)}P_i + \left(\mathbf{r}_{ji}^{(2c)}\right)^{\mathrm{T}}\nabla^{(2c)}P_i + \frac{1}{6}\left(\mathbf{r}_{ji}^{\mathrm{T}}\nabla\right)^3 P_i + \ldots \tag{11}$$

where, $\mathbf{r}_{ji}^{(2s)} = \begin{bmatrix} x_{ji}^2 & y_{ji}^2 & z_{ji}^2 \end{bmatrix}^{\mathrm{T}}$, $\mathbf{r}_{ji}^{(2c)} = \begin{bmatrix} x_{ji}y_{ji} & x_{ji}z_{ji} & y_{ji}z_{ji} \end{bmatrix}^{\mathrm{T}}$, $\nabla^{(2s)} = \begin{bmatrix} \frac{\partial^2}{\partial x^2} & \frac{\partial^2}{\partial y^2} & \frac{\partial^2}{\partial z^2} \end{bmatrix}^{\mathrm{T}}$ and $\nabla^{(2c)} = \begin{bmatrix} \frac{\partial^2}{\partial x\partial y} & \frac{\partial^2}{\partial x\partial z} & \frac{\partial^2}{\partial y\partial z} \end{bmatrix}^{\mathrm{T}}$. The splitting of 2nd-derivative term $\frac{1}{2}(\mathbf{r}_{ji}^T\nabla)^2P_i$ in the conventional Taylor's expansion (e.g. [40,44]) into two terms $\frac{1}{2}(\mathbf{r}_{ji}^{(2s)})^{\mathrm{T}}\nabla^{(2s)}P_i$ and $(\mathbf{r}_{ji}^{(2c)})^{\mathrm{T}}\nabla^{(2c)}P_i$ in Eq (11) results in the reduction of the size of matrix to be inversed when formulating the algorithms for approximating P_i, estimating the gradient ∇P_i, the Laplacian $\nabla^2 P_i$, as well as other 2nd derivatives at $\boldsymbol{r}_i$. The derivation follows the approach of weighted summation suggested by Ma [33] who derived the linear SFDI. The details can be found in the appendix or our previous work, only the final formulas are given here,

$$\nabla P_i = \sum_{j=1}^{N}\boldsymbol{\Phi}_{ji}^g\left(P_j - P_i\right) \tag{12}$$

$$\nabla^2 P_i = \boldsymbol{I}^{\mathrm{T}}\sum_{j=1}^{N}\boldsymbol{\Phi}_{ji}^s\left(P_j - P_i\right) \tag{13}$$

For convenience, Eqs. (12) and (13) are written in a summation form. In the above procedure, $\boldsymbol{I} = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}^{\mathrm{T}}$ and

$$\boldsymbol{\Phi}_{ji}^g = \boldsymbol{M}_{1q,i}^{-1}\left(\frac{W(\mathbf{r}_{ji})}{d_{ji}^2}\boldsymbol{q}_{ji} - \sum_{k=1}^{N}\frac{W(\mathbf{r}_{ki})}{d_{ki}^2}\boldsymbol{q}_{ki}\left(\mathbf{r}_{ki}^{(2c)}\right)^{\mathrm{T}}\boldsymbol{M}_{2c,i}^{-1}\frac{W(\mathbf{r}_{ji})}{d_{ji}^4}\mathbf{r}_{ji}^{(2c)} - \sum_{k=1}^{N}\frac{w_{kl}}{d_{ki}^2}\boldsymbol{q}_{ki}\boldsymbol{\Pi}_{ki}^{\mathrm{T}}\boldsymbol{M}_{2s,i}^{-1}\boldsymbol{\Gamma}_{ji}\right) \tag{14}$$

$$\boldsymbol{\Phi}_{ji}^s = 2\boldsymbol{M}_{2s,i}^{-1}\left(\boldsymbol{\Gamma}_{ji} - \sum_{k=1}^{N}\frac{w_{kl}}{d_{ki}^4}\boldsymbol{\Pi}_{ki}\boldsymbol{G}_{ki}^{\mathrm{T}}\boldsymbol{\Phi}_{ki}^g\right) \tag{15}$$

Definitions of matrices in Eqs. (14) and (15) are as follows

$$\boldsymbol{M}_{2c,i} = \sum_{j=1}^{N}\frac{W(\mathbf{r}_{ji})}{d_{ji}^4}\mathbf{r}_{ji}^{(2c)}\left(\mathbf{r}_{ji}^{(2c)}\right)^{\mathrm{T}} \tag{16a}$$

$$\boldsymbol{\Pi}_{ji} = \left\{\left(\mathbf{r}_{ji}^{(2s)}\right)^{\mathrm{T}} - \left(\mathbf{r}_{ji}^{(2c)}\right)^{\mathrm{T}}\boldsymbol{M}_{2c,i}^{-1}\sum_{k=1}^{N}\frac{W(\mathbf{r}_{ki})}{d_{ki}^4}\mathbf{r}_{ki}^{(2c)}\left(\mathbf{r}_{ji}^{(2s)}\right)^{\mathrm{T}}\right\}^{\mathrm{T}} \tag{16b}$$

$$\boldsymbol{M}_{2s,i} = \sum_{j=1}^{N}\frac{W(\mathbf{r}_{ji})}{d_{ji}^4}\boldsymbol{\Pi}_{ji}\boldsymbol{\Pi}_{ji}^{\mathrm{T}} \tag{16c}$$

$$\boldsymbol{G}_{ji}=\left\{\mathbf{r}_{ji}^{\mathrm{T}}-\left(\mathbf{r}_{ji}^{(2c)}\right)^{\mathrm{T}}\boldsymbol{M}_{2c,i}^{-1}\sum_{k=1}^{N}\frac{W(\mathbf{r}_{ki})}{d_{ki}^{4}}\mathbf{r}_{ki}^{(2c)}\mathbf{r}_{ki}^{T}\right\}^{\mathrm{T}} \tag{16d}$$

$$\boldsymbol{q}_{ji}=\left(\boldsymbol{G}_{ji}^{\mathrm{T}}-\boldsymbol{\Pi}_{ji}^{\mathrm{T}}\boldsymbol{M}_{2s,I}^{-1}\sum_{k=1}^{N}\frac{W(\mathbf{r}_{ki})}{d_{ki}^{4}}\boldsymbol{\Pi}_{ki}\boldsymbol{G}_{ki}^{\mathrm{T}}\right)^{\mathrm{T}} \tag{16e}$$

$$\boldsymbol{\Gamma}_{ji}=\left(\frac{W(\mathbf{r}_{ji})}{d_{ji}^{4}}\boldsymbol{\Pi}_{ji}-\boldsymbol{\Pi}_{ji}\left(\mathbf{r}_{ji}^{(2c)}\right)^{\mathrm{T}}\boldsymbol{M}_{2c,I}^{-1}\sum_{k=1}^{N}\frac{w_{ki}}{d_{ki}^{4}}\mathbf{r}_{ki}^{(2c)}\right) \tag{16f}$$

$$\boldsymbol{M}_{1q,i}=\sum_{j=1}^{N}\frac{W(\mathbf{r}_{ji})}{d_{ji}^{2}}\boldsymbol{q}_{ji}\mathbf{q}_{ji}^{\mathrm{T}} \tag{16g}$$

The leading truncation errors of Eqs. (12) and (13) are

$$\boldsymbol{E}_{\nabla p_i}=-\frac{1}{6}\boldsymbol{M}_{1q,i}^{-1}\sum_{j=1}^{N}\frac{W(\mathbf{r}_{ji})}{d_{ji}^{2}}\boldsymbol{q}_{ji}\left(\boldsymbol{F}_{ji}^{\mathrm{T}}-\boldsymbol{\Pi}_{ji}^{\mathrm{T}}\boldsymbol{M}_{2s,i}^{-1}\sum_{k=1}^{N}\frac{W(\mathbf{r}_{ki})}{d_{ki}^{4}}\boldsymbol{\Pi}_{ki}\boldsymbol{F}_{ki}^{\mathrm{T}}\right)\nabla^{(3)}p_i \tag{17a}$$

and

$$\boldsymbol{E}_{\nabla^2 P_i}=\boldsymbol{I}^{\mathrm{T}}\left\{-\frac{1}{3}\boldsymbol{M}_{2s,i}^{-1}\sum_{j=1}^{N}\frac{W(\mathbf{r}_{ji})}{d_{ji}^{4}}\boldsymbol{\Pi}_{ji}\boldsymbol{F}_{ji}^{\mathrm{T}}\nabla^{(3)}p_i-2\boldsymbol{M}_{2s,i}^{-1}\times\sum_{j=1}^{N}\frac{W(\mathbf{r}_{ji})}{d_{ji}^{4}}\boldsymbol{\Pi}_{ji}\boldsymbol{G}_{ji}^{\mathrm{T}}\boldsymbol{E}_{\nabla P_i}\right\} \tag{17b}$$

where, $\boldsymbol{F}_{ji}=\left\{\left(\mathbf{r}_{ji}^{(3)}\right)^{\mathrm{T}}-\left(\mathbf{r}_{ji}^{(2c)}\right)^{\mathrm{T}}\boldsymbol{M}_{2c,I}^{-1}\sum_{k=1}^{N}\frac{W(\mathbf{r}_{ki})}{d_{ki}^{4}}\mathbf{r}_{ki}^{(2c)}\left(\mathbf{r}_{ki}^{(3)}\right)^{\mathrm{T}}\right\}^{\mathrm{T}}$ and $\mathbf{r}_{ji}^{(3)}=[x_{ji}^{3}\ \ 3x_{ji}^{2}y_{ji}\ \ 3x_{ji}^{2}z_{ji}\ \ 3x_{ji}y_{ji}^{2}\ \ 6x_{ji}y_{ji}z_{ji}\ \ 3x_{ji}z_{ji}^{2}\ \ y_{ji}^{3}\ \ 3y_{ji}^{2}z_{ji}\ \ 3y_{ji}z_{ji}^{2}\ \ z_{ji}^{3}]^{\mathrm{T}}$. It is clear that Eqs. (17a) and (17b) are all termed by the third order derivative of P. Theoretically, such leading truncation terms confirm that Eqs. (12) and (13) provide exact solutions for the gradient and Laplacian of a function that satisfy a second-order polynomial. It further advises a second order convergence rate for finding the gradient of a function and a first order convergence rate for directly estimating the Laplacian of a function, as confirmed by the theoretical patch test in Ref. [45]. In the present ISPH_CQ, Eq. (13) is used to discretise the Laplacian in Eq. (8), Eq. (12) is applied to implementing the pressure boundary condition (Eq. (4)) and to calculating the pressure gradient in Eq. (9). In addition, Eqs. (12) and (13) are also extended to finding the velocity divergence and the viscous term, respectively,

$$\nabla\cdot\mathbf{u}_i=\sum_{j=1}^{N}\boldsymbol{\Phi}_{ji}^{g}\cdot\left(\mathbf{u}_j-\mathbf{u}_i\right) \tag{18}$$

$$\nabla\cdot(\nu_i\nabla\mathbf{u}_i)=\sum_{j=1}^{N}\nu_i\boldsymbol{\Phi}_{ji}^{s}\cdot\left(\mathbf{u}_j-\mathbf{u}_i\right) \tag{19}$$

Overall, all key numerical implementations required by the above-mentioned projection procedure are consistently dealt with by the QSFDI, except the estimation of the intermediate density in the prediction stage, whose influence on the accuracy of the PPE may be minimal due to the fact that a small blending coefficient of 0.01 is applied. Consequently, the overall turncation error of the ISPH_CQ resulting from the spatial discretisation is dominated by the third order derivative of physical quantities. A systematic numerical investigation will be carried out in the following section to confirm the theoretical analysis. It is worth noting that there are few attempts in literature to develop and apply consistent gradient and Laplacian schemes for Lagrangian particle methods. For example, Nasar et al. [26] developed a high order consistent SPH with the pressure projection method and applied it to some benchmark problems such as the Taylor Couette problems; Sibilla [49] proposed an implicit consistency correction scheme to improve the consistency of the SPH approximation; Koh et al. [24] and Luo et al. [25] developed the consistent particle method (CPM) and applied the CPM to solving free surface problems. Recently, Shimizu et al. [47] developed a consistent second order ISPH and investigated its performance for modelling free surface flow. However, the PPE solved in Shimizu's ISPH model [47] is written to enforce a density invariance, whereas that used in this paper is written as a blended form that consists of both the velocity divergence term and the density variation term [6,11,28,29]. The small value of the blending coefficient, i.e. 0.01 in Zhang et al. [29], implies that the velocity divergence term is the dominant term and the accuracy of the gradient discretisation may play more important role for securing a satisfactory convergence, compared with Shimizu's ISPH model [47]. It is admitted that the second order algorithms developed by Shimizu et al. [47] and that developed by Tamai et al. [44], as well as the second order WLS or MLS methods, can also be applied to discretising the present PPE and dealing with the gradient operators. The theoretical patch tests by Shimizu et al. [47] and Tamai et al. [44] have shown a first-order convergence for approximating the Laplacian of specified function and the second-order convergence for estimating the gradient. These are consistent with the QSFDI patch test by Yan et al. [45]. Nevertheless, to discretise the Laplacian and the gradient, the method proposed by Shimizu et al. [47] needs to calculate inverse matices with sizes of 5 × 5 for 2D problems and 9 × 9 for 3D problems, larger than 2 × 2 and 3 × 3, respectively, in the QSFDI, i.e. $\boldsymbol{M}_{1q,i}$, $\boldsymbol{M}_{2c,i}$ and $\boldsymbol{M}_{2s,i}$ are all 3 × 3 matrices for 3D problems and they are 2 × 2 matrix, scalar value and 2 × 2 matrix for 2D problems, respectively. As confirmed by Yan et al. [45] and Zhang et al. [29], reducing the size of the matrices to be inversed brings considerable benefit on improving the computational efficiency for both the theoretical patch test and practical ISPH simulation of free surface flow.

2.3. Distinguished difference from classic ISPH and ISPH_QSFDI

As indicated in the Introduction, the novelty of the present ISPH_CQ is the application of the QSFDI consistently to the Laplacian and gradient discretisation/approximations in the ISPH. This is different from the classic ISPH [28] and the ISPH_QSFDI [29]. In the classic ISPH [28], the velocity divergence and the viscous stress term are discretised at the particle i, respectively, by

$$\nabla\cdot\mathbf{u}_i=-\frac{1}{\rho_i}\sum_{j=1}^{N}m_j\left(\mathbf{u}_i-\mathbf{u}_j\right)\cdot\nabla_i W(\mathbf{r}_{ij}) \tag{20}$$

$$\nabla\cdot(\nu_i\nabla\mathbf{u}_i)=\sum_{j=1}^{N}8m_j\left(\frac{\nu_i+\nu_j}{\rho_i+\rho_j}\frac{\mathbf{u}_{ij}\cdot\mathbf{r}_{ij}}{{d_{ij}}^2+\varepsilon^2}\right)\cdot\nabla_i W(\mathbf{r}_{ij}) \tag{21}$$

where, ϵ is a small number to avoid the singularity caused by $\mathbf{r}_{ij}=0$; $\mathbf{u}_{ij}=\mathbf{u}_i-\mathbf{u}_j$. These equations are commonly used in other SPH applications, e.g. [50,51]. The pressure gradient in the correction stage and the LHS of the pressure boundary condition (Eq. (4)) are discretised by using the linear SFDI [11,29,46],

$$\nabla p_i=\boldsymbol{M}_{1,i}^{-1}\sum_{j=1,j\neq i}^{N}\frac{W(\mathbf{r}_{ij})\mathbf{r}_{ij}}{d_{ij}^{2}}\left(p_j-p_i\right) \tag{22a}$$

where, the matrix $\boldsymbol{M}_{1,i}=\sum_{j=1}^{N}\frac{W(\mathbf{r}_{ji})\mathbf{r}_{ji}\mathbf{r}_{ji}^{\mathrm{T}}}{d_{ij}^{2}}$ has a size of 2 × 2 and 3 × 3 for 2D and 3D problem, respectively. For 2D problems, $\boldsymbol{M}_{1,i}^{-1}$ can be directly calculated and consequently,

$$\nabla p_i=\sum_{j=1,j\neq i}^{N}\frac{n_i^{x_k}B_{ij}^{x_m}-n_i^{xy}B_{ij}^{x_k}}{n_i^{x}n_i^{y}-n_i^{xy}n_i^{xy}}\left(p_j-p_i\right) \tag{22b}$$

where, $n_i^{xy}=\sum_{j=1,j\neq i}^{N}\frac{(\mathbf{r}_j^{x_m}-\mathbf{r}_i^{x_m})(\mathbf{r}_j^{x_k}-\mathbf{r}_i^{x_k})}{d_{ij}^{2}}W(\mathbf{r}_{ij})$, $n_i^{x_m}=\sum_{j=1,j\neq i}^{N}\frac{(\mathbf{r}_j^{x_m}-\mathbf{r}_i^{x_m})^2}{d_{ij}^{2}}W(\mathbf{r}_{ij})$ $B_i^{x_m}=\sum_{j=1,j\neq i}^{N}\frac{(\mathbf{r}_j^{x_m}-\mathbf{r}_i^{x_m})}{d_{ij}^{2}}W(\mathbf{r}_{ij})$, $x_m=x$ when $x_k=y$

Table 1
Summary of numerical implementations of ISPH models.

Numerical implementations		ISPH_CQ	ISPH_QSFDI	ISPH
prediction	Intermediate velocity and position	Eq. (6) and Eq. (7), respectively		
	Intermediate density	$\rho^* = \sum_{j=1}^{N} m_j W(\mathbf{r}_{ij})$		
	Viscous stress	Eq. (19)	Eq. (21)	Eq. (21)
Pressure Projection	PPE	Eq. (8)		
	Laplacian discretisation	Eq. (13)	Eq. (13)	Eq. (23)
	Velocity divergence	Eq. (18)	Eq. (20)	Eq. (20)
	Solid boundary condition	Eq. (12)	Eq. (22)	Eq. (22)
	Free surface condition	Eq. (5), free surface particle is identified by using a method based on the auxiliary functions [5]		
	Linear algebraic solver	Bi-CGSTAB method [52] without pre-conditioning. Tolerance 10^{-5}.		
Correction	Pressure gradient	Eq. (12)	Eq. (22)	Eq. (22)
	Correction of particle velocity and position	Eq. (9) and Eq. (10), respectively		
	Particle stabilization and shifting	The hybrid particle stabilization and shifting scheme proposed by Zhang et al. [28]		
Kernel/Weighting function		The cubic B-spline kernel proposed by Monaghan and Lattanzio [53]		

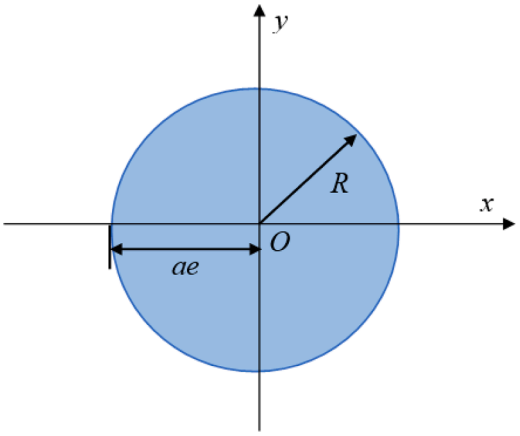

Fig. 4. Schematic sketch of the oscillating drop under a conservative force field.

or $x_m = y$ when $x_k = x$, and $\mathbf{r}^{x_m}$ is the component of the position vector in x_m direction. The leading truncation error of Eq. (22) is termed by the second order derivatives and, therefore, Eq. (22) gives exact solutions for a linear distribution of the pressure. The Laplacian in Eq. (8) is discretised by

$$\nabla^2 p_i = \sum_{j=1}^{N} \frac{4m_j}{\rho_i + \rho_j} \frac{(p_i - p_j)\mathbf{r}_{ij}}{d_{ij}^2 + \epsilon^2} \cdot \nabla_i W(\mathbf{r}_{ij}) \tag{23}$$

The overall accuracy of the spatial discretization in the classic ISPH [28] is dominated by the linear schemes for the gradient estimation and the Laplacian discretisation.

In the ISPH_QSFDI [29], only Eq. (23) is replaced by Eq. (12) for the Laplacian discretisation, all other numerical implementations remain the same as the classic ISPH, i.e., Eqs. (20)–(22). In particular, the linear discretisation of the Neumann boundary condition on the solid boundaries significantly constraints the order of the accuracy and convergence of the PPE solution, even though the QSFDI is applied to discretise the Laplacian. The overall accuracy of the spatial discretization in the ISPH_QSFDI is still dominated by the linear scheme for the gradient estimation due to the fact that the overall degree of accuracy is determined by the scheme with the lowest order. This has been confirmed by the first-order convergence of the ISPH_QSFDI observed in the numerical investigation of wave-structure interaction problems in Ref. [29]. For clarity, numerical implementations of different ISPH models are summarised in Table 1. In the present ISPH_CQ, the QSFDI is consistently adopted to the Laplacian and gradient discretisation/approximations, involved in the viscous stress, Laplacian operator, velocity divergence and corresponding Neumann boundary condition and the pressure gradient. Through the use of the second-order gradient estimation in the ISPH_CQ, instead of the first-order estimation in the ISPH_QSFDI, the overall degree of accuracy and convergence of the ISPH_CQ are expected to be improved. It is noted that $W(\mathbf{r}_{ij})$ in the SFDI and QSFDI, e.g. Eqs. (11), (12) and (22), is a weighting function that can be different from the SPH kernel function, e.g. $W(\mathbf{r}_{ij})$ in Eqs. (20), (21) and (23). In this paper, the cubic B-spline kernel proposed by Monaghan and Lattanzio [53] is used for the SPH kernel function and the weighting function of the SFDI and QSFDI.

3. Numerical investigations

In this section, the present ISPH_CQ is applied to various cases with the free surface, including the oscillating drop, wave propagation and liquid sloshing. Its performance in terms of accuracy, convergence, computational efficiency, mass (volume) and energy conservation and of producing satisfactory pressure field is assessed. For the purpose of comparison, the classic ISPH [28] and the ISPH_QSFDI [29] are also employed. Although the mathematical models and the numerical procedure described above are valid for both 2D and 3D problems, only 2D cases are considered in the investigation. In the convergence investigations, following the concept of the Courant–Friedrich–Lewy (CFL) condition, the time step size dt is determined using the mean particle spacing dx, i.e. $Co = u_{max}dt/dx$, where u_{max} is the maximum particle speed. As discussed above, the temporal scheme adopted in the present projection approach is first order for the particle velocity and second order for the particle displacement. In order to minimise the impact of the time step size (the temporal scheme) on the convergence investigation, the Courant number Co is taken as a small value, i.e. $0.1u_{max}$, according to our numerical investigation.

3.1. Oscillating drop under a conservative force field

The oscillating drop under a conservative force field has been widely considered as a benchmark case for assessing the accuracy, convergence and energy conservation of particle methods [18,19,54]. In this case, the fluid is incompressible and inviscid, and is subjected to a conservative force field of $-\Psi^2\mathbf{r}$, where Ω is a dimensional parameter and $\Psi^2 = 1.44$ N/m is used in the present study; $\mathbf{r}$ (x, y) is the position vector of a particle, measured from the centre of the drop at its initial state when the drop exhibits a circular shape with a radius R of 0.5 m, as sketched in Fig. 4. The particle velocity at the initial state is specified as $(\sigma x, -\sigma y)$, where the coefficient σ is taken as 0.4 s^{-1}. The free surface boundary condition is imposed to the outer edge of the drop. For this problem, the corresponding exact theoretical solution is available and is given in Ref. [54]. All numerical models use $dt/dx = 0.1$ s/m to determine the time step size. Correspondingly, the Courant number is 0.02 considering the maximum particle velocity of 0.2 m/s in this case. For convenience of the analysis, the parameters with a length scale are non-dimensionalised by the drop radius R and the time by $\tau = t\sqrt{g/R}$, unless mentioned otherwise.

Fig. 5 displays the particle distributions and the pressure contour for the drop at $\tau = 88.58$. In this case, the particle resolution is taken as $dx = 0.02$. As expected, all ISPH models reproduce a smooth pressure distribution. However, a close look at the particle distribution, e.g., the zoom-

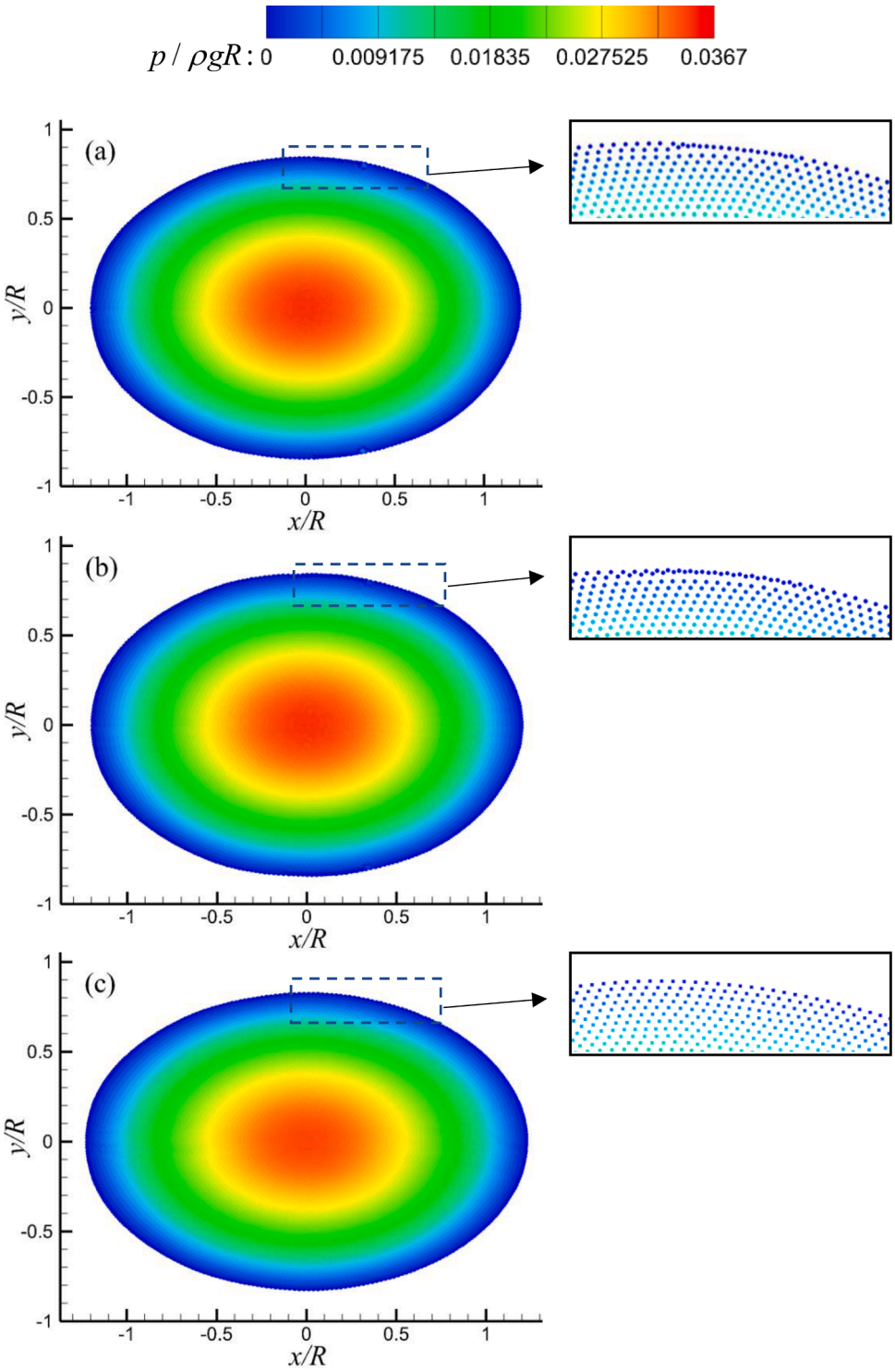

Fig. 5. The particle distributions and pressure contour for the oscillating drop at $\tau = 88.58$ (a: ISPH; b: ISPH_QSFDI; and c: ISPH_CQ. $dx = 0.02$).

in plots in the right column of Fig. 5, suggests that the present ISPH_CQ results in a better (more evenly distributed, lower irregularity) particle distribution near the free surface than other ISPH models.

Fig. 6 illustrates the time histories of the semi-major axis *ae* of the oscillating drop resulted from different ISPH models. For comparison, the corresponding theoretical solution [54] is also plotted. It is shown that the numerical results by the ISPH_CQ agree well with the theoretical solution, whereas the classic ISPH and the ISPH_QSFDI do not only underestimate the peak value but also lead to a visible phase shift. To quantitatively assess the accuracy of the numerical results, the averaged errors Err_a (L-2 norm) of the numerical results with reference to the analytical solution are analysed. The error is defined as

$$Err_a = \frac{\sqrt{\sum_{i=1}^{N}\left(ae_{i,n} - ae_{i,a}\right)^2}}{\sqrt{\sum_{i=1}^{N} ae_{i,a}^2}} \tag{24}$$

where, $ae_{i,a}$ is the theoretical value of the semi-major axis *ae* at *i*th time step and $ae_{i,n}$ is the corresponding numerical results, N is the total number of time steps considered for the error evaluation. In this case,

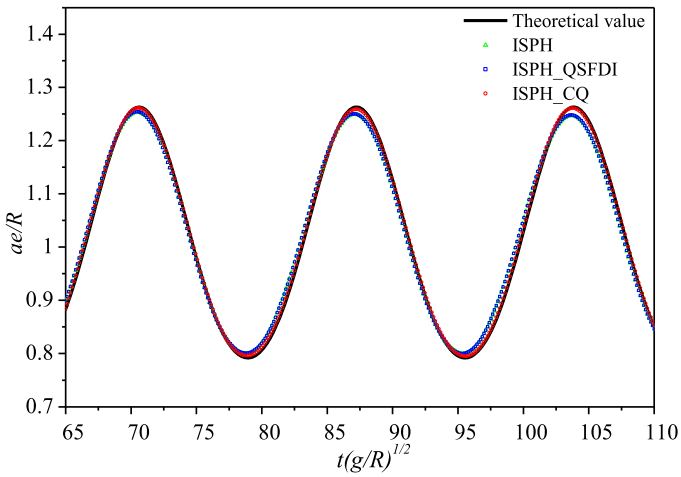

Fig. 6. Time histories of semi-major axis *ae* of the oscillating drop ($dx = 0.02$).

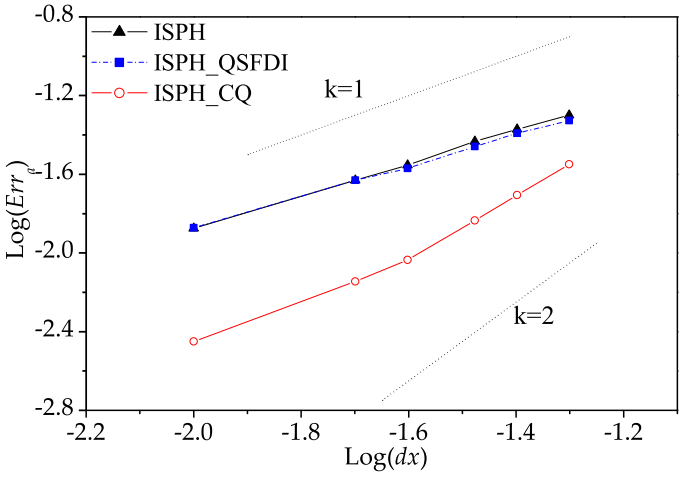

Fig. 7. Numerical errors for modelling oscillating drop using different particle resolutions ($R = 0.5$ m).

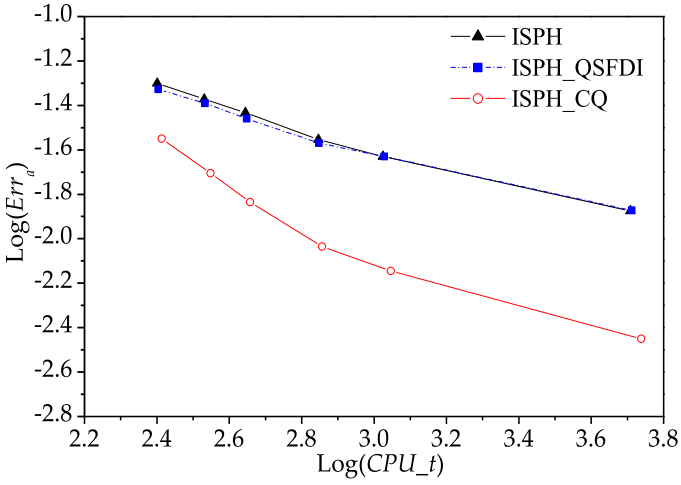

Fig. 8. CPU times spent by different ISPH models for modelling the oscillating drop ($R = 0.5$ m).

the duration of the simulation is taken as 110.73 and the numerical results at the final two periods, i.e. 82.38–110.73, are used for the error evaluation. For all cases with different ISPH methods and different

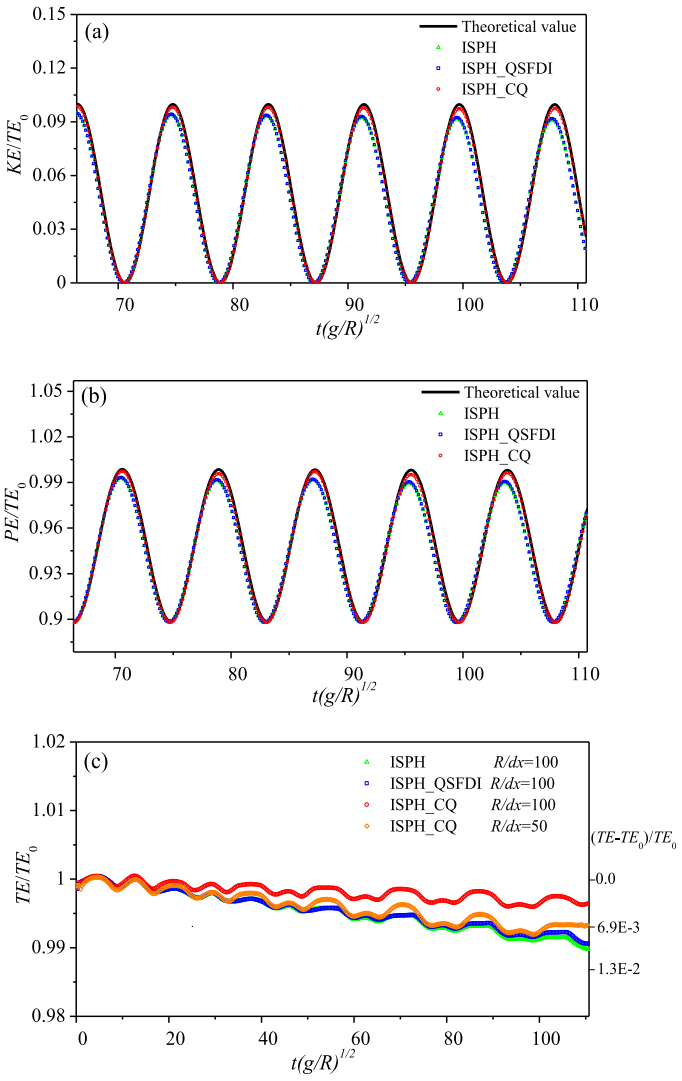

Fig. 9. Time histories of (a) kinetic energy (*KE*); (b) potential energy (*PE*) and (c) total energy (*TE*) in the cases with oscillating drop ($dx = 0.02$).

particle resolutions, the same time interval (yield the same value of *N*) is used for post-processing the data and estimating the error.

Fig. 7 shows the numerical errors in the cases with different particle spacing *dx*, ranging from 0.05 to 0.01. For convenience, two dash lines representing the first-order ($k = 1$) and second-order ($k = 2$) convergence rates are illustrated, respectively. It is observed that the ISPH_CQ shows a second-order convergence when $dx > 0.025$ ($\log(dx) \approx -1.6$), and a slower convergent trend (approximately a first-order rate) when the particle spacing becomes further smaller. This is because other sources of errors may become relatively significant when the particle resolution is below a value, which are not reduced by decreasing the spacing. One source of error may be related to the imprecise enforcement of the dynamic free surface boundary condition, Eq. (5), due to the perturbation of the free surface (see the free surface particles in Fig. 5(a) and (b)), mainly caused by particle shifting and kernel-based approximations in presence of truncated kernel domains. Such perturbation becomes relatively sensitive when the particle resolution increases. The other error source may be related to the time integration scheme employed. This may become dominated when *dx* is sufficiently small. The high order time integration scheme would be used to reduce the error. These will be further investigated in our future work. Although a perfect second-order convergence was not achieved in the whole range of particle spacings, the smallest spacing within the range of second

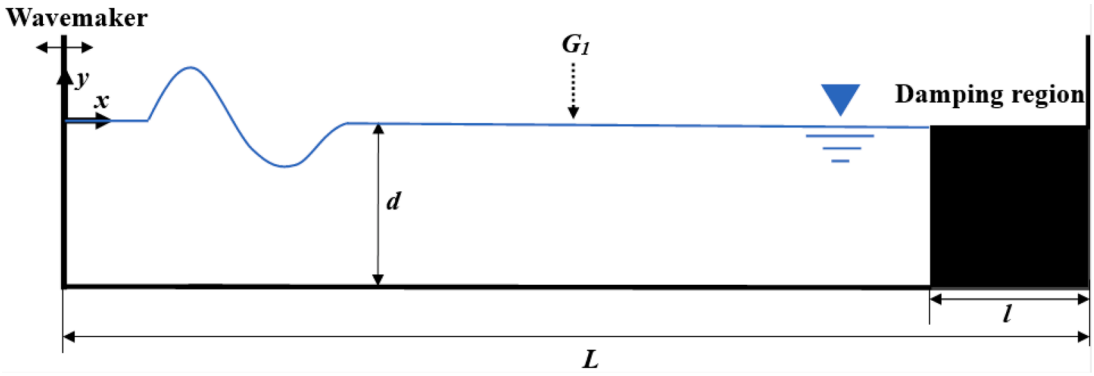

Fig. 10. Schematic setup of the numerical wave tank (G_1 demonstrates a wave gauge).

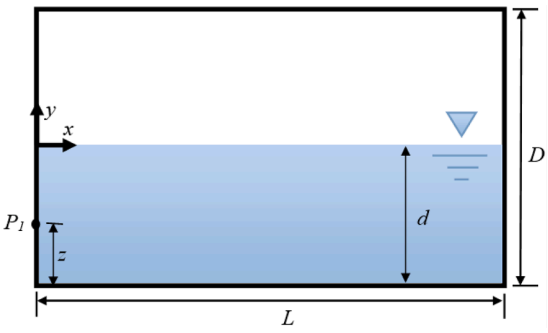

Fig. 11. Sketch of the sloshing tank.

order convergence can lead to numerical results that have invisible difference from the analytical solution as shown in Fig. 7. It is also observed that the accuracy of the ISPH_CQ in general is significantly higher than the accuracies of other models adopting the same particle spacing.

We recognise that discretising the Laplacian and gradient using high order schemes, such as the QSFDI in this paper and the consistent second order scheme developed by Shimizu et al. [47], requires more computational effort than the linear or other lower order schemes. One may question whether it is worth to sacrifice the time to improve the accuracy. To clarify this, the CPU time taken by different ISPH models to achieve the results shown in Fig. 7 are analysed. All these cases are run in the same workstation with Intel i7 3.3 GHz and 128 GB RAM using OpenMP for parallel computing with 16 cores. Fig. 8 illustrates the CPU time taken by all ISPH models to achieve the results from $\tau = 0$ to $\tau = 110.73$. It clearly reveals that the ISPH_CQ requires significantly less CPU time than other ISPH models to achieve the same level of accuracy.

The energy conservation is examined next. Fig. 9 displays the time histories of the kinetic energy (*KE*), potential energy (*PE*) and total energy (*TE*), in which TE_0 is the theoretical value of total energy. From Fig. 9(a) and (b), one can observe that the ISPH_CQ reproduces the energy time histories better than others, consistent with the observation from Fig. 6. Although the total energy of the drop seems to follow a dropping trend in all numerical simulations, the ISPH_CQ demonstrates a significantly better energy conservation, compared with other ISPH models using the same particle resolution. For a similar threshold value of energy loss, e.g., 1 %, the ISPH_CQ requires a much coarser particle resolution, e.g. $dx = 0.02$, than other ISPH models, i.e. 0.005. It is worth noting that the total energy in the present numerical simulation is not constantly dropping but slightly oscillating, similar to the observation in literature, e.g. [19]. In the present work, such oscillations may be caused by the particle shifting procedure, which does not guarantee the energy conservation [15].

3.2. Water wave propagation and liquid sloshing

On the basis of the above benchmarking test, the performance of the ISPH_CQ is now assessed by modelling water wave propagation problems where the implementation of the solid boundary condition is needed in addition to the free surface conditions. Unlike the inviscid case in Section 3.1, the viscosity of the fluid is taken into account and the kinematic viscosity of 10^{-6} m^2/s is applied in these cases. Although the accuracy of the viscous term estimation in the ISPH_CQ is first order, its contribution to the overall degree of accuracy and convergence is insignificant due to the fact that the viscous effect in the non-breaking free surface flow is negligible.

The first set of cases is the wave propagation in a numerical wave tank as sketched in Fig. 10, where d and L are the mean water depth and the length of the tank, respectively. A wavemaker is placed at the left end of the tank. A Cartesian coordinate system (x, y) is adopted. The horizontal axis x origins from the original location of the wavemaker and points to the right end of the tank; the vertical axis y points upwards and origins from the mean water surface. Both solitary waves and regular waves propagations in the wave tank are simulated. In the cases with solitary waves, $d = 0.25$ m, $L = 40d$, the right end of the tank is assigned to be a rigid wall and the length of the damping region is chosen as zero. The solitary wave with wave height h is generated by the wavemaker using the approach presented in Ref. [34]. It is expected that the total energy become a constant value after the wavemaker stops. For solitary waves, the analytical solution from Boussinesq equation [55] is widely used to describe their wave profiles. Although the analytical solution was derived based on the assumptions of incompressible, inviscid fluid and $h/d << 1$, it is used here to evaluate the accuracy of the numerical models. In the cases with regular waves, $d = 0.5$ m, $L = 60d$ and various wave gauges are placed along the direction of wave propagation. The regular waves are generated by a piston wavemaker using the linear wavemaker theory [56]. To prevent the undesirable wave reflections from the right end of the tank, an artificial damping region with a length of 3 wavelength is attached to the right end. In the damping region, the fluid velocity is corrected by a velocity damping during the correction stage of the projection procedure. More details about the damping technique can be found in Ref. [57]. We choose the case with wave height of $0.2d$ and wave period 1.2 s. For such wave condition, the second order Stokes wave theory [58] is valid and provides the reference value for the error evaluation in the present numerical investigation. In the analysis, the parameters with a length scale are nondimensionalised by the water depth d and the time by $\tau = t\sqrt{g/d}$.

Liquid sloshing is another classic free surface problem. The second set of cases considered in the investigation is the simulation of sloshing waves in an oscillating tank. The schematic diagram of the tank is illustrated in Fig. 11, where L is the tank length, D is its height, and d is the mean water depth. A pressure sensor P_1 is located on the left side

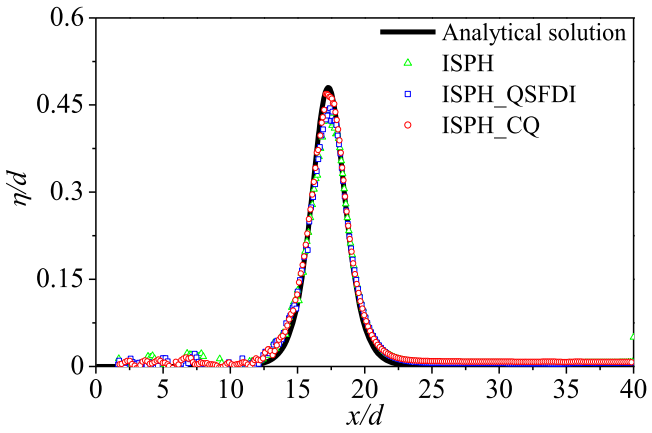

Fig. 12. Comparisons of solitary wave profiles between analytical and numerical results at τ =19.42 in the cases with $h = 0.48$ ($dx = 0.05$).

wall of the tank at a distance z from the bottom. The tank is subjected to a periodic sway motion with its displacement being specified by $X_s = a\sin(\omega t)$, where a and ω are the amplitude and the frequency of the motion, respectively. Two conditions are used for different purposes. In the first condition, $L = 1.0$ m, $d = 0.5\ L$, $a = 0.005\ L$ and $\omega = 0.8\omega_1$, where ω_1 is the natural frequency of the first mode of the tank, and the analytical solution by Faltinsen [59] and Wu et al. [60].

$$\eta = \frac{a}{g}\left(x\omega^2 + \sum_{n=0}^{\infty} C_n\omega\sin k_n x\right)\sin\omega t - \frac{a}{g}\sum_{n=0}^{\infty}\omega_n\left(C_n + \frac{H_n}{\omega^2}\right)\sin k_n x\sin\omega t \tag{25}$$

where, $k_n = \frac{2n+1}{L}\pi$, $\omega_n = \sqrt{gk_n\tanh k_n d}$, $H_n = \omega^3\frac{1}{L}\frac{4(-1)^n}{k_n^2}$, $C_n = \frac{H_n}{\omega_n^2-\omega^2}$ and η is the wave elevations, will be used to quantitively analyse the accuracy, convergence, computational robustness and the volume conservation, despite of the fact that the analytical solution may contain inexactness for a highly nonlinear sloshing. The second case involves a violent sloshing to assess the performances of the ISPH models on capturing the impact pressure. For this purpose, the experimental case carried out by Kishev et al. [61] is chosen. In the analysis, the parameters with a length scale are nondimensionalised by the tank length L and the time by $\tau = t\sqrt{g/L}$.

3.2.1. *Accuracy, convergence and efficiency*

Fig. 12 compares the free surface profiles at τ =19.42 for solitary wave with $h = 0.48$. The corresponding Courant number adopted in this case is approximately 0.2. The numerical results from different ISPH models, which adopt a mean particle resolution $dx = 0.05$, and the corresponding analytical solution based on the Boussinesq equation [55] are plotted for comparison. It is observed that all ISPH models produce the numerical results that reasonably agree with the analytical solution. However, the present ISPH_CQ leads to the best results in terms of

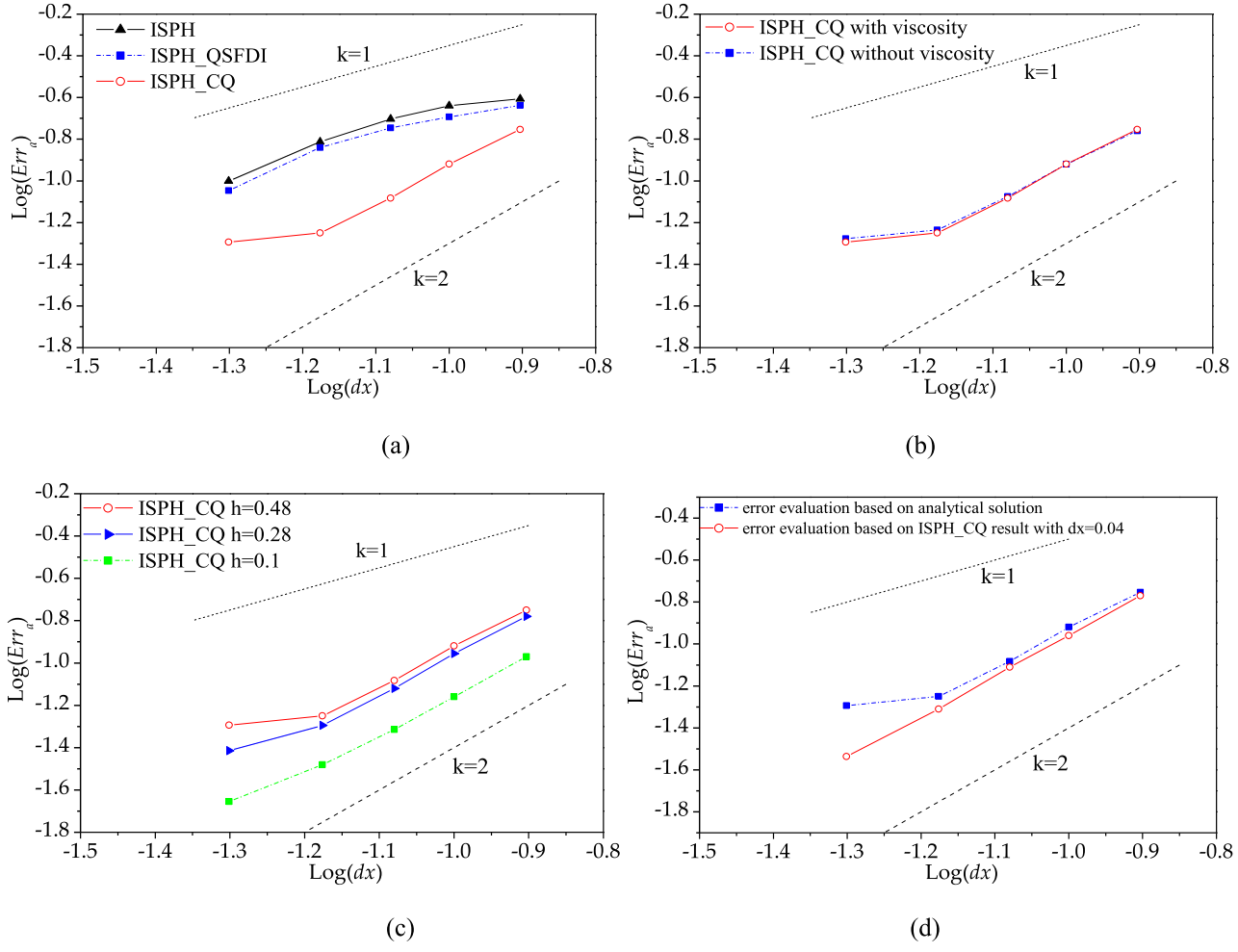

Fig. 13. Averaged errors Err_a (L-2 norm) of numerical results corresponding to different particle spacing in the cases for solitary wave propagation (a: $h = 0.48$, taking Boussinesq solution [55] as the reference solution; b: viscous effect, $h = 0.48$, taking Boussinesq solution [55] as the reference solution; c: nonlinear effect, taking Boussinesq solution [55] as the reference solution; d: $h = 0.48$, the errors of ISPH_CQ estimated by taking different references).

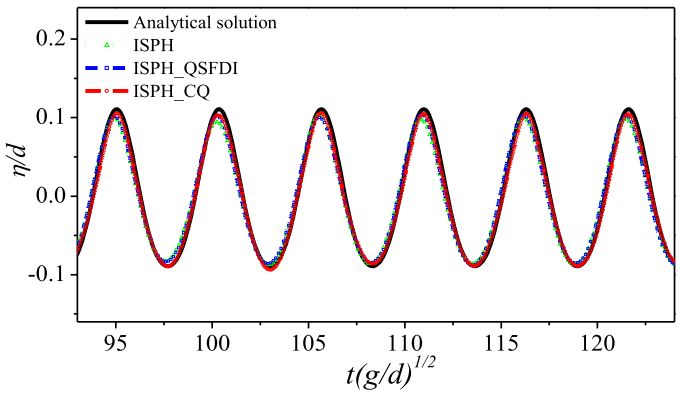

Fig. 14. Comparisons of regular wave time histories between the analytical and ISPH results at $x = 20$ (wave height $H = 0.2$ and wave period 5.315; $dx = 0.025$).

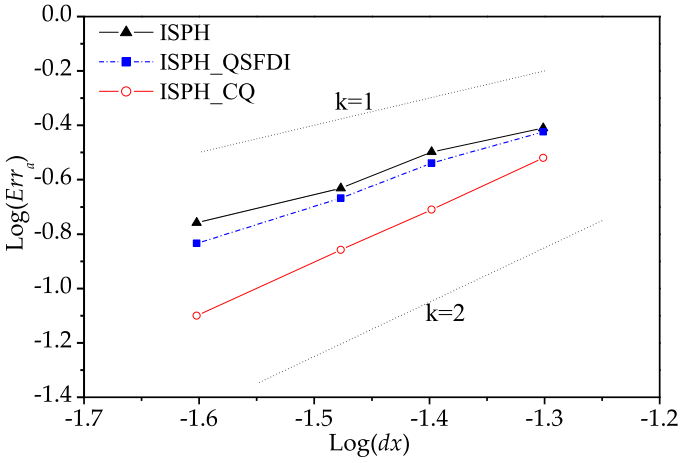

Fig. 15. Averaged errors Err_a (L-2 norm) of numerical results from different ISPH models in the cases with regular wave propagation.

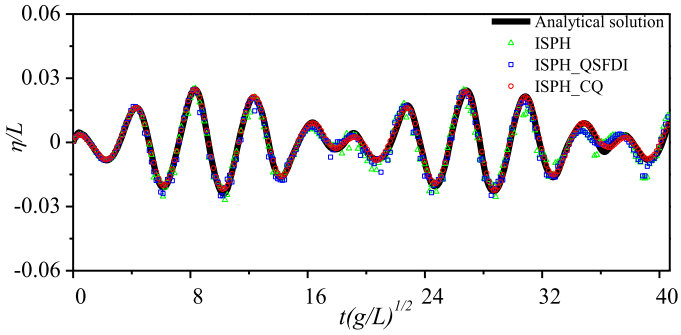

Fig. 16. The comparison of time histories of the surface elevations between the analytical solution [59,60] and ISPH results recorded at the left wall of the sloshing tank ($L = 1.0$ m, $d = 0.5$, $a = 0.005$ and $\omega = 0.8\omega_1$, $dx = 0.01$).

capturing the target wave crest at $\eta = 0.48$. Following the benchmarking test above, the averaged error in this case is estimated using the following equation,

$$Err_a = \frac{\sqrt{\sum_{i=1}^{N}\left(\eta_{i,n} - \eta_{i,a}\right)^2}}{\sqrt{\sum_{i=1}^{N}\eta_{i,a}^2}} \tag{26}$$

where, $\eta_{i,n}$ is the wave elevation at ith particle obtained from the numerical simulations and $\eta_{i,a}$ is the reference solution, N is the total number of particles in the sub-domain considered for the error evaluation. In this work, the sub-domain is chosen to be approximately centred at the wave crest and specified by $x = [14\ 22]$ for the case with $h = 0.48$. To investigate the convergence properties of the ISPH models, the averaged errors in the cases with different particle resolutions, ranging from $dx = 0.125$ to 0.05, are analysed.

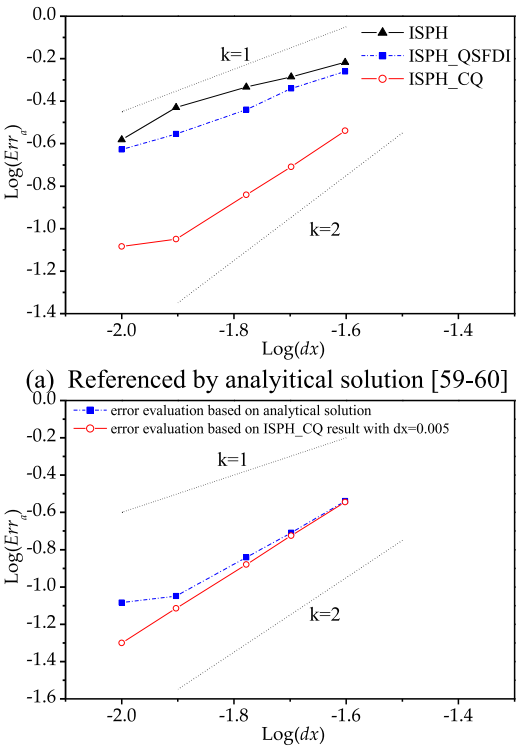

Fig. 17. Errors of numerical results in the sloshing cases with different particle resolutions ($L = 1.0$ m, $d = 0.5$, $a = 0.005$ and $\omega = 0.8\omega_1$).

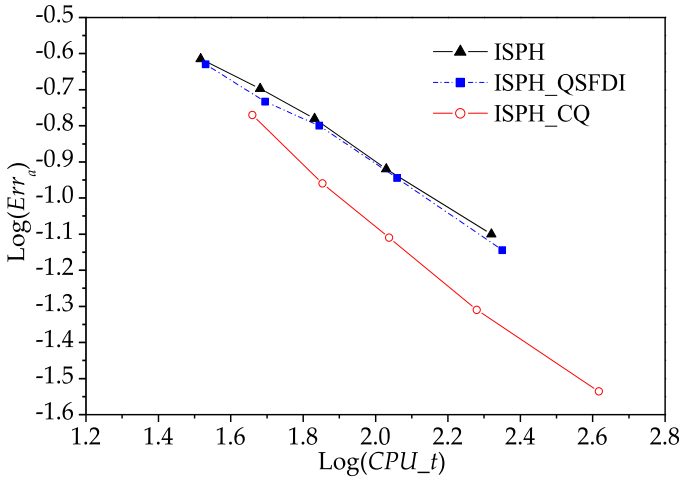

Fig. 18. CPU times spent by different ISPH models in the case with $h = 0.48$ (error is estimated with reference to the ISPH_CQ results with $dx = 0.04$).

Fig. 13(a) compares the L-2 norm error Err_a defined by Eq. (26), in which $\eta_{i,a}$ is given by the Boussinesq solution [55], resulted from different ISPH models for the cases with $h = 0.48$. It is observed that the accuracy of the ISPH_CQ is, in general, significantly higher than other schemes adopting the same particle spacing. One may observe a

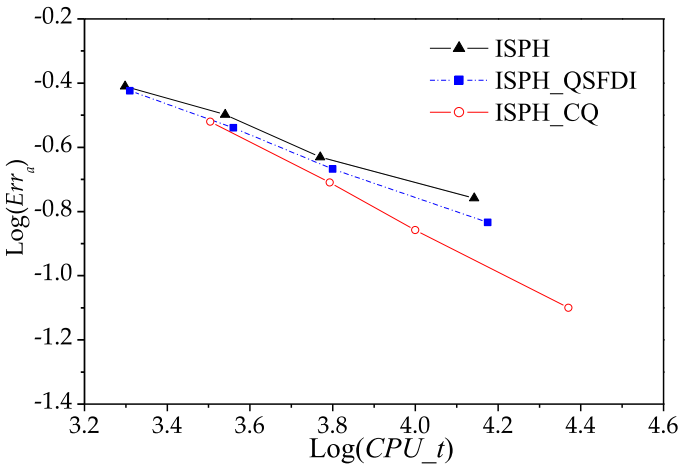

Fig. 19. CPU times spent by different ISPH models in the cases with regular wave propagation (wave height $H = 0.2$ and wave period 5.315).

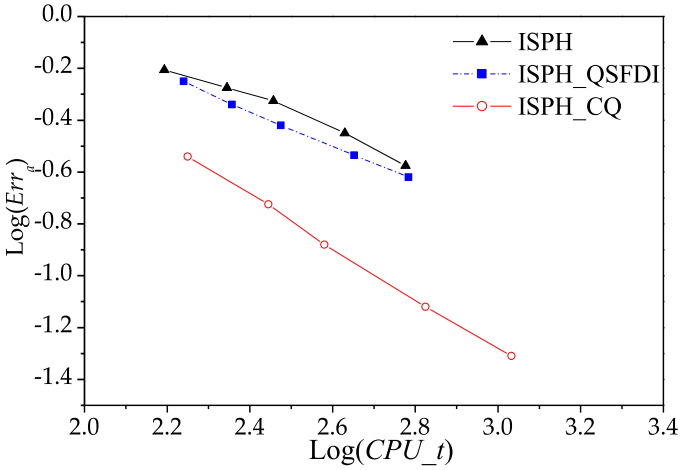

Fig. 20. CPU times spent by different SPH models to achieve the results at $\tau = 40.72$ for modelling the liquid sloshing ($L = 1.0$ m, $d = 0.5$, $a = 0.005$ and $\omega = 0.8\omega_1$, error is estimated using the ISPH_CQ results with $dx = 0.005$).

second-order convergence rate of the ISPH_CQ, when $dx > 1/15$ ($\log(dx) \approx -1.176$), one order higher than other ISPH models. Nevertheless, when the particle spacing is further reduced to $dx = 0.05$ ($\log(dx) \approx -1.3$), the convergence rate of the ISPH_CQ is less than a second order. In order to find the reasons, several possible causes are considered. The first one is that the viscosity is taken into account in the numerical model but not in the Boussinesq solution [55] that is taken as the reference in Eq. (26). The second is that the ISPH models are fully nonlinear while the Boussinesq solution is weakly nonlinear with the assumption of $h/d << 1$. For the case with $h = 0.48$, the assumption $h/d << 1$ is not well satisfied, yielding the inexactness of the Boussinesq solution, that does not depend on the particle spacing. When the error of numerical results is relatively large, such theoretical inexactness does not play a significant role and so not significantly affect the convergence rate of the numerical methods. However, when the error of the numerical results reaches to a certain level with reducing the particle spacing, the theoretical inexactness becomes relatively more evident. To test the perception, we simulated some cases without viscosity and some cases with different wave heights. Fig. 13(b) shows the L-2 norm error of the ISPH_CQ with and without considering the viscous term. It confirms that the viscous term plays insignificant role [62] and is not the cause leading to the lower convergence rate at $dx = 0.05$. Fig. 13(c) depicts the convergence rates corresponding to different wave heights. It is found

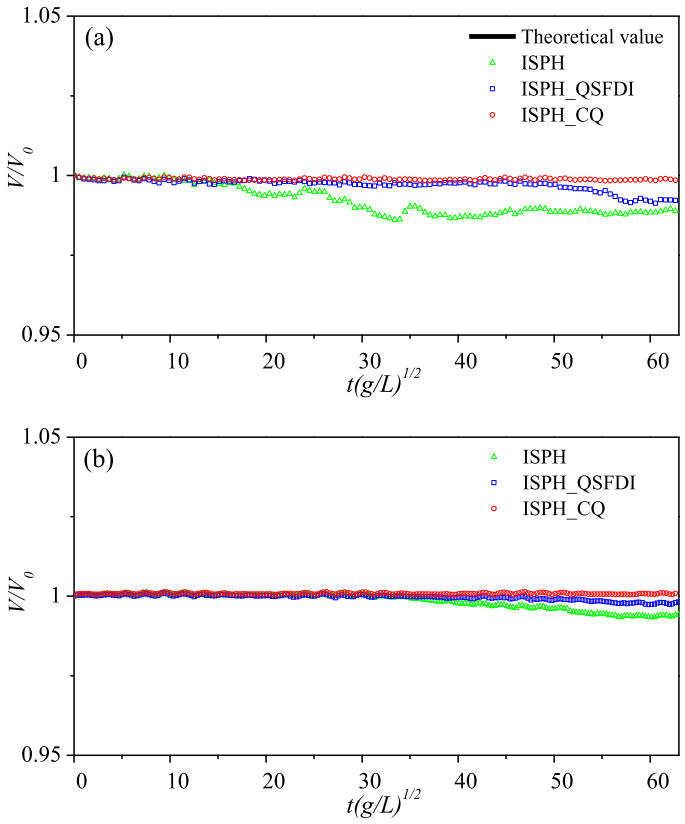

Fig. 21. Time history of fluid volume in the sloshing tank in the cases with (a) $dx = 1/60$; (b) $dx = 1/100$ ($L = 1.0$ m, $d = 0.5$, $a = 0.005$ and $\omega = 0.8\omega_1$).

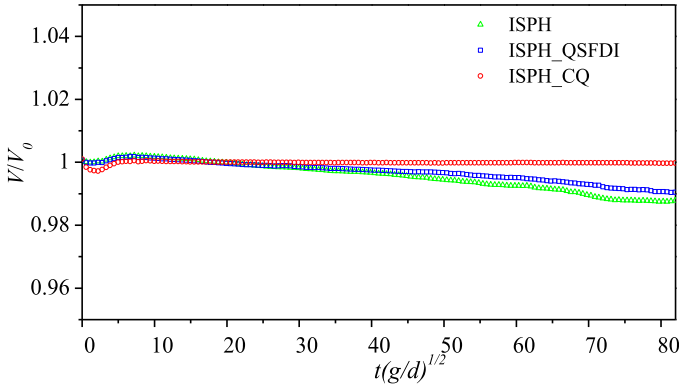

Fig. 22. Time histories of the volume of the fluid in the case with solitary wave propagation ($h = 0.4$ and $dx = 0.05$).

from Fig. 13(c) that as the wave height (wave nonlinearity) decreases, the convergence rate becomes faster at $dx = 0.05$; in particular, when $h = 0.1$, a nearly second-order convergence rate is observed in the entire range of the particle spacing. To further explore this aspect, the numerical error is estimated by taking the numerical results given by a finer particle spacing, i.e. $dx = 0.04$, as $\eta_{i,a}$ in Eq. (26). The convergence rate is plotted in Fig. 13(d), which shows a second order convergence of the ISPH_CQ. From these results, we believe that the slower convergence rate in Fig. 13(a) at $dx = 0.05$ is because the theoretical solution used as reference in Eq. (26) bears some inexactness due to the fact that h/d does not well satisfy the assumption of the Boussinesq solution [55]. For the cases falling in the valid range of the Boussinesq solution, i.e. $h = 0.1$, a second order convergence rate can be observed. This conclusion will be further consolidated by the following case with regular waves which can

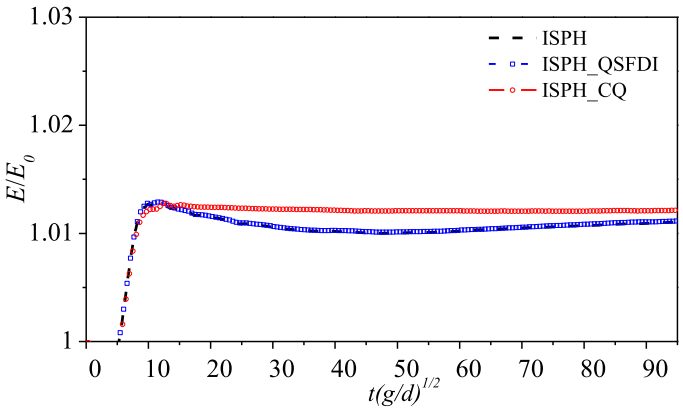

Fig. 23. Time histories of the total energy in the case with solitary wave propagation ($h = 0.2$ and $dx = 0.025$).

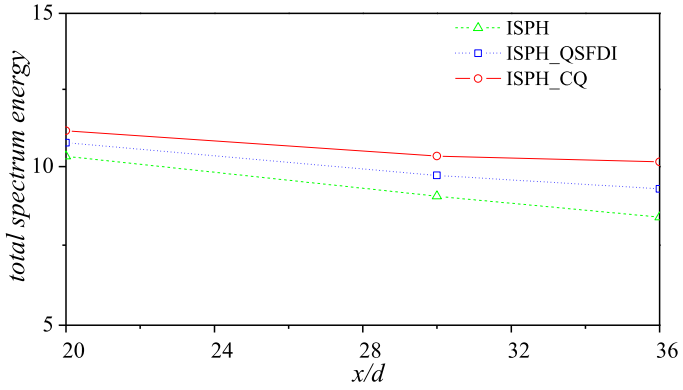

Fig. 24. Total spectrum energy recorded at different locations along the wave tank in the case with regular wave propagation (wave height $H = 0.2$ and wave period 5.315; $dx = 0.025$).

be accurately described by the second order Stokes wave theory [58].

Fig. 14 plots the time histories of the wave elevation recorded at $x = 20$ in the case with regular wave propagation, in which the analytical solution is given by the second order Stokes wave theory [58]. The corresponding Courant number adopted in this case is approximately 0.15 in which the maximum fluid speed is taken as $\sqrt{gd}$. In the ISPH modelling, the mean particle resolution $dx = 0.025$ is chosen. Similar to the comparisons in Figs. 7 and 13, the ISPH_CQ delivers the results with a better accuracy than the classic ISPH and the ISPH_QSFDI. This fact is clearer in Fig. 15 which quantitatively compares the variation of the average errors (Err_a) of different ISPH models as the mean particle spacing changes. Err_a is defined in a way similar to Eq. (24) but $ae_{i.a}$ is replaced by the wave elevation at ith time step from analytical solution and $ae_{i.n}$ by the corresponding analytical solution, N is the total number of time steps as defined in Eq. (24) during the time window considered for the error estimation, which is 6 wave periods starting from $\tau = 83.27$ in this case. Fig. 15 does not only clarify the observation in Fig. 14 but also confirms that the ISPH_CQ has a second-order convergence. The errors have also been estimated by taking the numerical results obtained by using the finest mesh in the tests as for Fig. 13, and same behaviour is observed, which are not presented as they do not add much value. This is perhaps due to the fact that the second order analytical solution is accurate enough for this case.

For the liquid sloshing problems, the corresponding Courant number adopted in this case is approximately 0.08. Fig. 16 compares the time histories of the surface elevations recorded at the left wall of the sloshing tank in the case with $L = 1.0$ m, $d = 0.5$, $a = 0.005$ and $\omega = 0.8\omega_1$. For the purpose of comparison, the analytical solution by Faltinsen [59] and Wu et al. [60] is also plotted. As expected, the result predicted by the ISPH_CQ agrees well with the analytical solution, whereas the results from other two ISPH models show a noticeable difference from the analytical solution, especially during $\tau = 15.66$ to 21.92 and 34.45 to 40.72. The corresponding error analysis is shown in Fig. 17. The definition of the error is the same as that used in Fig. 15 for the numerical results in the time window $\tau = 0$ to 40.72. Similar to the cases shown in Fig. 13, the analytical solution may contain inexactness for the cases with high nonlinearities, both the analytical solutions and the ISPH_CQ solution with a higher particle resolution, i.e. $dx = 0.005$, are used as the reference value to define the error in Fig. 17(b). Clearly, it leads to the same conclusion in terms of the accuracy and convergence as that was observed in Fig. 13.

Overall, one may agree that the ISPH_CQ results in a second-order convergence over a limited range of conditions considered in this paper, when the viscosity effect is not significant, and a better accuracy compared with the ISPH_QSFDI and the classic ISPH. In the ISPH_QSFDI, despite a more accurate Laplacian discretisation is applied and a slight improvement of the accuracy compared with the classic ISPH is observed in Figs. 7, 13, 15 and 17, the convergence is mainly first-order because a first-order gradient discretisation scheme (SFDI) is applied for calculating the velocity divergence and the pressure gradient, as well as implementing the solid boundary condition (see Table 1 for details). After the discretisation of relevant terms in the ISPH_QSFDI is replaced by the QSFDI in the ISPH_CQ, the overall accuracy and convergence are enhanced. This confirms our hypothesis stated above and in Zhang et al. [26], i.e., the order of accuracy/convergence of the ISPH depends on all numerical implementations, which can be generalised as the Laplacian and gradient discretisation/approximation, and may be dominated by the implementation/scheme with the lowest order of accuracy/convergence. This further confirms the necessity of developing consistent numerical models. This observation is also in line with a recent work by Pan et al. [36], who developed a consistent weak formulation for discretising both the PPE and the solid boundary condition using the MLPG-R, and demonstrated an improvement in the accuracy compared with the original MLPG-R with directly discretising the solid boundary condition using the SFDI.

Following Fig. 8, the CPU time spent by different ISPH models to achieve the results shown in Figs. 13, 15 and 17 are analysed. Fig. 18 illustrates the CPU time taken by all models to achieve the results from $\tau = 0$ to 19.42 for the solitary wave case with $h = 0.48$. It once again reveals that the ISPH_CQ requires significantly less CPU time than other ISPH models to achieve the same level of accuracy. The same observation is hold for the cases of the wave propagation and the liquid sloshing, as demonstrated by Figs. 19 and 20.

It is worth noting that the CPU time required to discretise the Laplacian and estimate the gradient in the QSFDI is longer than the linear SFDI and other lower-order schemes adopted in the classic ISPH and the ISPH_QSFDI for the same particle resolution, though the QSFDI results in a better accuracy. Attention shall be paid to balance the CPU time and the required accuracy. In some cases with low particle resolutions, the advantage of using the ISPH_CQ is not significant, e.g. Fig. 19, when the error is not small. Nevertheless, as the particle resolution increases, the advantage becomes more significant and consistent. For a threshold of error, e.g. 10 %, which is normally required, the advantage of the ISPH_CQ is observed to be significant in all the cases considered in this paper. It is also worth noting that the investigation of the CPU time presented above is based on a shared-memory OpenMP parallel computing, which may be ideal for small-scale problems that can be solved using a workstation. Although techniques for accelerating the ISPH have been developed, e.g., the pseudo-spectral ISPH [63], or the particle methods supported by the modern machine learning techniques (e.g. [64,65]), the most commonly used approach to accelerate the simulation is to develop massively parallel schemes supporting distributed memory and running in high-performance computer clusters (HPC) or super computers. There is a significant progress on parallelisation of

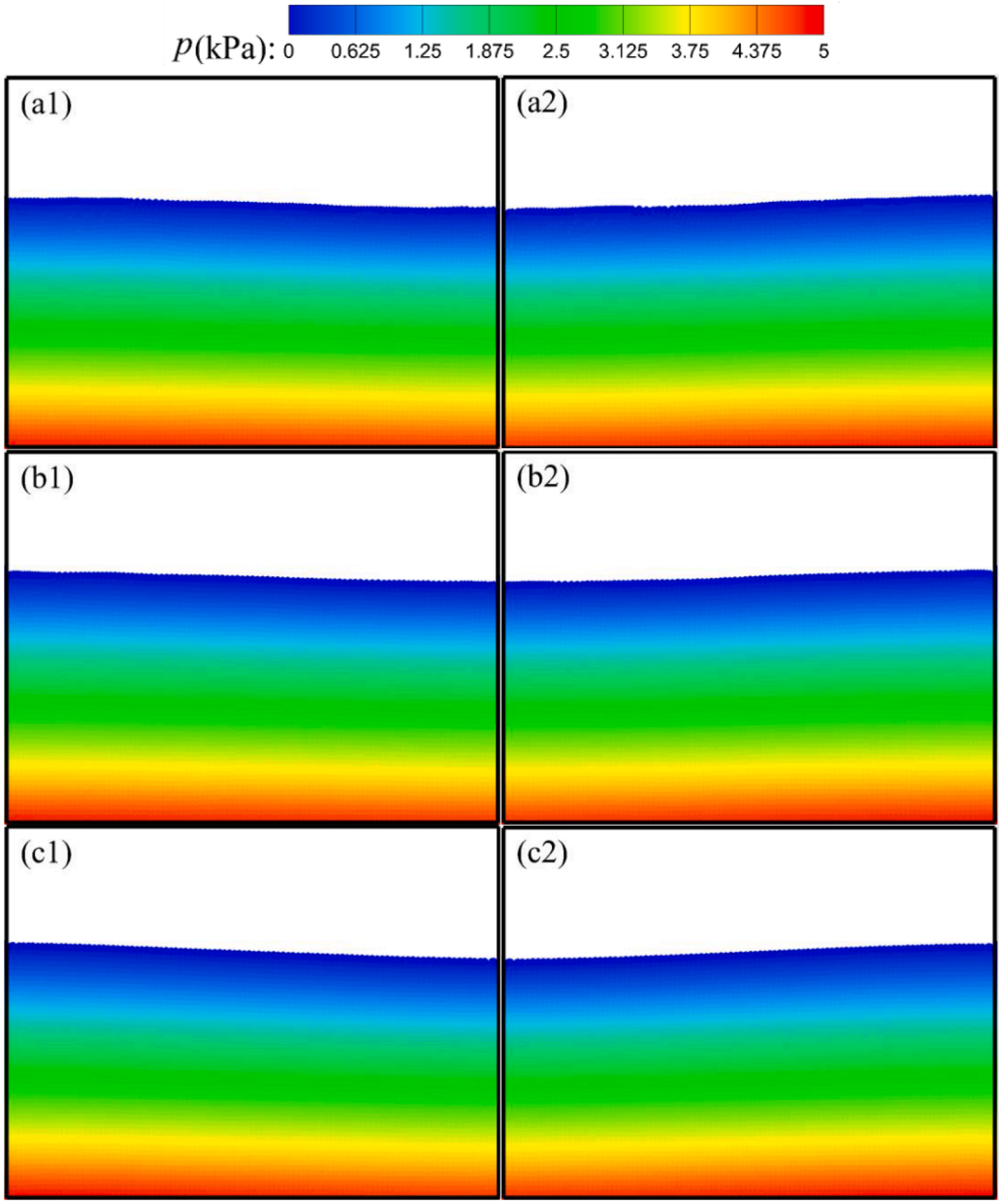

Fig. 25. The particle distributions and pressure contour: (a) ISPH; (b) ISPH_QSFDI and (c) ISPH_CQ at different time instants: $\tau = 50.11$(a1, b1 and c1) and $\tau = 51.68$ (a2, b2 and c2) ($L = 1.0$ m, $d = 0.5$, $a = 0.005$ and $\omega = 0.8\omega_1$ and $dx = 0.01$).

ISPH solvers recently, including the message passing interface (MPI) parallelisation [66,67] or graphics processing unit (GPU) computing [68]. For parallel computing using distributed memory, the parallel efficiency can be affected by particle data locality, communication due to distributed parallelism and load imbalance caused by the particle movement between partitions at the hierarchical architecture of CPU/GPU cores and HPC nodes, which are closely correlated with the kernel radius (the influence domain size) and the particle irregularity. The classic ISPH and the ISPH_QSFDI typically require a larger influence domain size for gradient/divergence approximations to ensure a smooth pressure field and a stable solution than the ISPH_CQ, as will be discussed in Section 3.2.4. This means that the number of neighbouring particles for a particle in its influence domain is smaller in the ISPH_CQ than that required in the classic ISPH and ISPH_QSFDI. Therefore, the ISPH_CQ is expected to need less amount of inter-processor communication due to neighbouring particles data dependency of the different processors. Furthermore, the particle distributions in the ISPH_CQ simulations seem to be more even or less irregular compared with those in other ISPH models, as demonstrated by Fig. 6. This helps maintaining a satisfactory conditional number of the matrix for the algebraic equations resulted from discretising the PPE, avoiding ill conditioned matrix or singularity due to the particle irregularity, and consequently leads to a quick convergence for solving the algebraic equations. Furthermore, it brings convenience to maintaining load balancing and a robust neighbour particle search by minimising the particle movement between partitions. However, these hypotheses cannot be confirmed until the corresponding parallel solver based on the ISPH_CQ can be developed in our future work.

3.2.2. *Mass/volume and energy conservations*

Mass/volume and energy conservations are other important criteria to assess the performance of a numerical model. Shimizu et al. [47] carried out a systematic investigation using the cases with standing waves and regular wave propagations, and concluded that the consistent

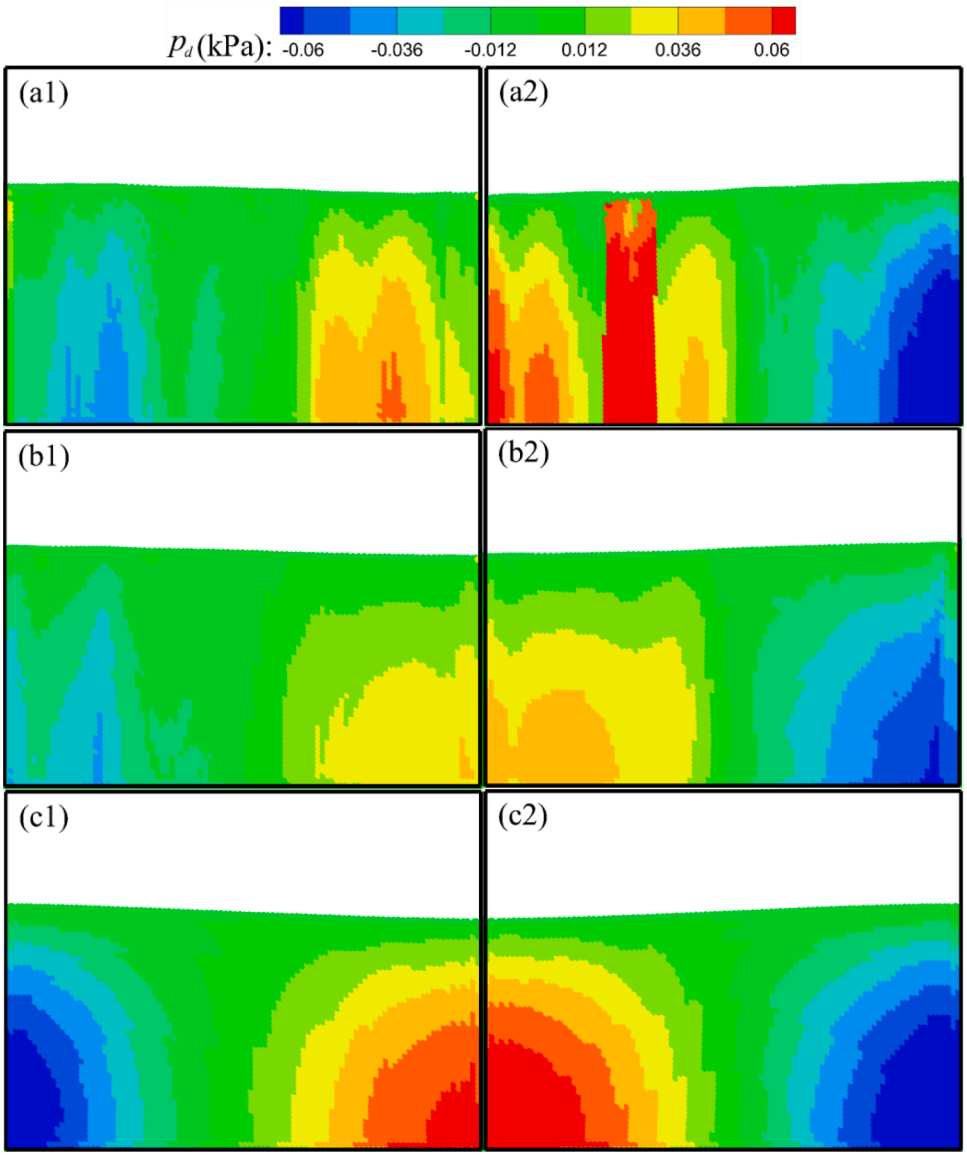

Fig. 26. The particle distributions and dynamic pressure contour: (a) ISPH; (b) ISPH_QSFDI and (c) ISPH_CQ at different time instants: $\tau = 50.11$(a1, b1 and c1) and $\tau = 51.68$ (a2, b2 and c2) ($L = 1.0$ m, $d = 0.5$, $a = 0.005$ and $\omega = 0.8\omega_1$ and $dx = 0.01$).

application of the second order Laplacian and gradient discretisation can significantly improve the energy conservation. In this section, the mass (volume) and energy conservations in the cases presented in the above section are examined.

In the ISPH, mass may be considered to be always conserved due to invariance of number of particles involved. Hence, numerical resolution and satisfaction of the continuity equation would be linked to the volume conservation including the total volume of the fluid. Fig. 21 displays the time histories of the fluid volume in the sloshing tank in the cases with $L = 1.0$ m, $d = 0.5$, $a = 0.005$ and $\omega = 0.8\omega_1$, corresponding to Figs. 16 and 17. The fluid volume in the tank is obtained by integrating $\eta + d$ over the entire numerical tank. A theoretical volume V_0 of the fluid, i.e., the fluid volume at the initial state, is also plotted in Fig. 21 for comparison. As seen, the ISPH_CQ results in a nearly constant fluid volume during the simulation for both particle resolutions, whereas both the classic ISPH and the ISPH_QSFDI lead to a visible loss of the volume, especially in the case with the coarser particle resolution ($dx = 1/60$). The observation proves a better performance of the ISPH_CQ than the classic ISPH and the ISPH_QSFDI in terms of mass/volume conservation.

The superiority of the ISPH_CQ over the other ISPH models in terms of volume conservation was also found in other cases with different particle resolutions. To save the space, only one set of results in the case with solitary wave propagations is presented in Fig. 22, where $d = 0.25$ m, $h = 0.4$, $L = 140$ and $dx = 0.05$. It is observed that the volume obtained by the ISPH_CQ slightly varies before the wavemaker stops at ~ 10, and tends to be a constant value thereafter. This suggests a good volume conservation of the ISPH_CQ, an insignificant difference between the ISPH_CQ results and the theoretical value (i.e., $V/V_0 = 1.0$). However, the fluid volumes in the simulations using the classic ISPH and the ISPH_QSFDI reduce continuously after the wavemaker stops.

On the other hand, the performances of the ISPH models in terms of energy conservation are also examined. Fig. 23 illustrates the time histories of the total energy for the same case shown in Fig. 22. In this case, the wavemaker feeds energy into the fluid and, consequently, the fluid energy increases until the wavemaker stops. During the period when the wavemaker moves, all models seem to yield similar results. After the wavemaker stops, the energy in the tank is expected to be constant. One can find from Fig. 23 that the total fluid energy in the ISPH_CQ simulation is well conserved after the wavemaker stops, but the

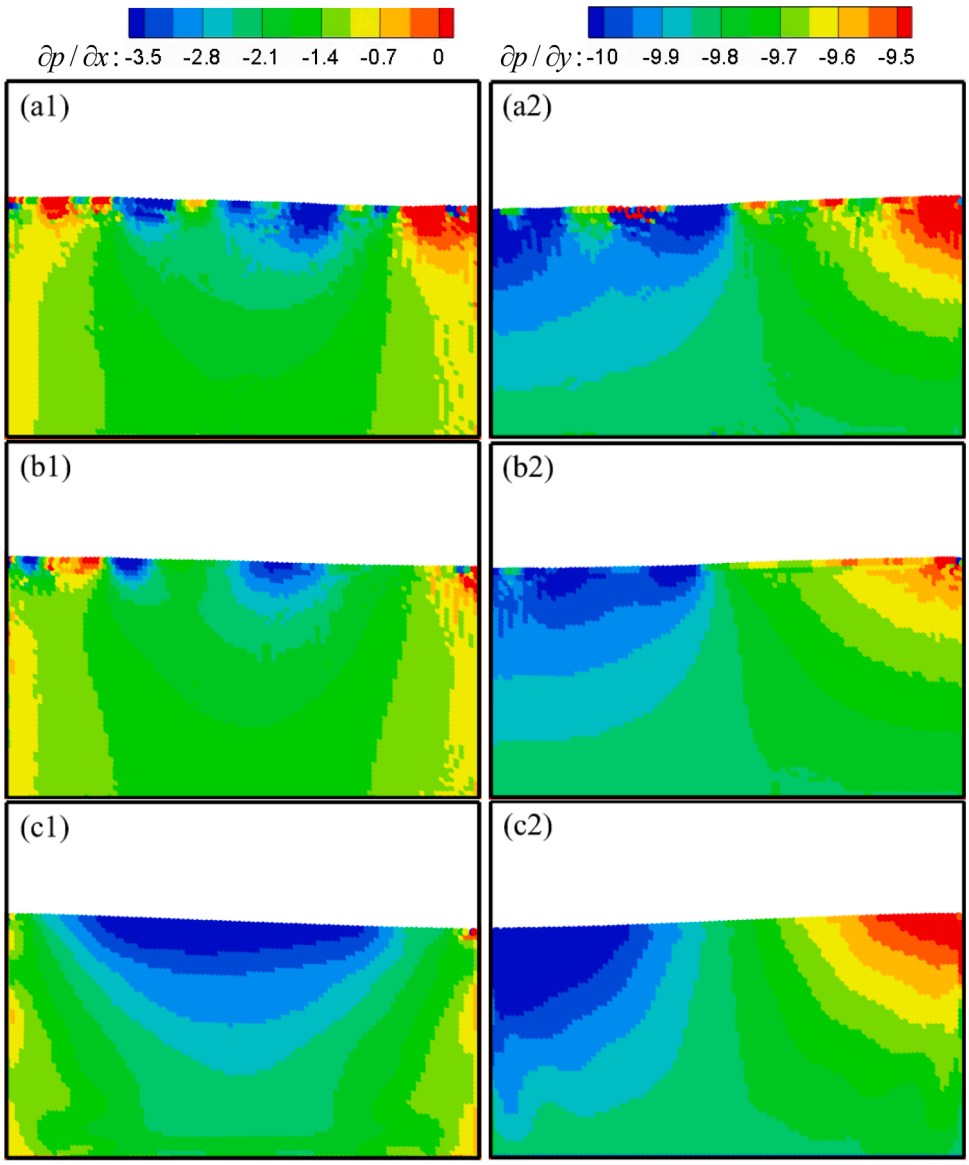

Fig. 27. The pressure gradient contour: (a) ISPH; (b) ISPH_QSFDI and (c) ISPH_CQ at different time instants (left column: $\partial p/\partial x$ at $\tau = 50.11$; right column: $\partial p/\partial y$ at $\tau = 51.68$) ($L = 1.0$ m, $d = 0.5$, $a = 0.005$ and $\omega = 0.8\omega_1$ and $dx = 0.01$).

corresponding results from other two ISPH models exhibit a continuous loss of energy after the wavemaker stops until a quasi-steady state is reached at ~ 32. This is consistent with the observation in Fig. 12 that the classic ISPH and ISPH_QSFDI underestimate the solitary wave height.

For the regular wave propagation, the energy conservation can be reflected by the spectrum energy recorded at different locations along the direction of the propagation. Fig. 24 compares the total spectrum energy recorded at different locations in the case corresponding to Fig. 14. The particle spacing yielding approximately 80 particles in each wavelength (~ 2 m) and 40 particles along the water depth is adopted. Generally speaking, the wave energies at different locations predicted by the ISPH_CQ are the highest and those by the classic ISPH are the lowest. This is consistent with the comparison of the wave elevation shown in Fig. 14. More importantly, the rate of the reduction in the case with ISPH_CQ is lower than others. The percentage of the energy reductions from $x/d = 20$ to $x/d = 36$ are 8.9 %, 13.8 % and 19 % for the ISPH_CQ, the ISPH_QSFDI and the classic ISPH, respectively. One may agree that this exhibits that the ISPH_CQ has a better energy conservation property than other ISPH models in terms of maintaining the wave energy during a long-distance wave propagation, although the energy loss can be further reduced if a finer particle resolution is applied.

3.2.3. *Pressure distribution*

As indicated in the Introduction, the ISPH generally produces a better pressure field, e.g. smoother distribution and less suspicious pressure fluctuation, than the weakly compressible SPH. In this section, the performance of the ISPH_CQ in terms of pressure prediction is systematically investigated using the case with liquid sloshing.

Fig. 25 illustrates the particle distribution and the pressure contour at different instants in the case corresponding to Figs. 16 and 21, where the particle resolution $dx = 0.01$ is adopted. The corresponding dynamic pressure distribution is plotted in Fig. 26. All ISPH models seem to reproduce a smooth pressure field (Fig. 25) but the ISPH_CQ results in a smoother dynamic pressure distribution (Fig. 26). A similar conclusion is also made by Shimizu et al. [47] who demonstrated the advantage of using the consistent high order ISPH on producing a smoother dynamic pressure than the lower order competitors.

It is noted that the dynamic pressure is mainly associated with the kinematics of the particles, which are dominated by the gravity and the

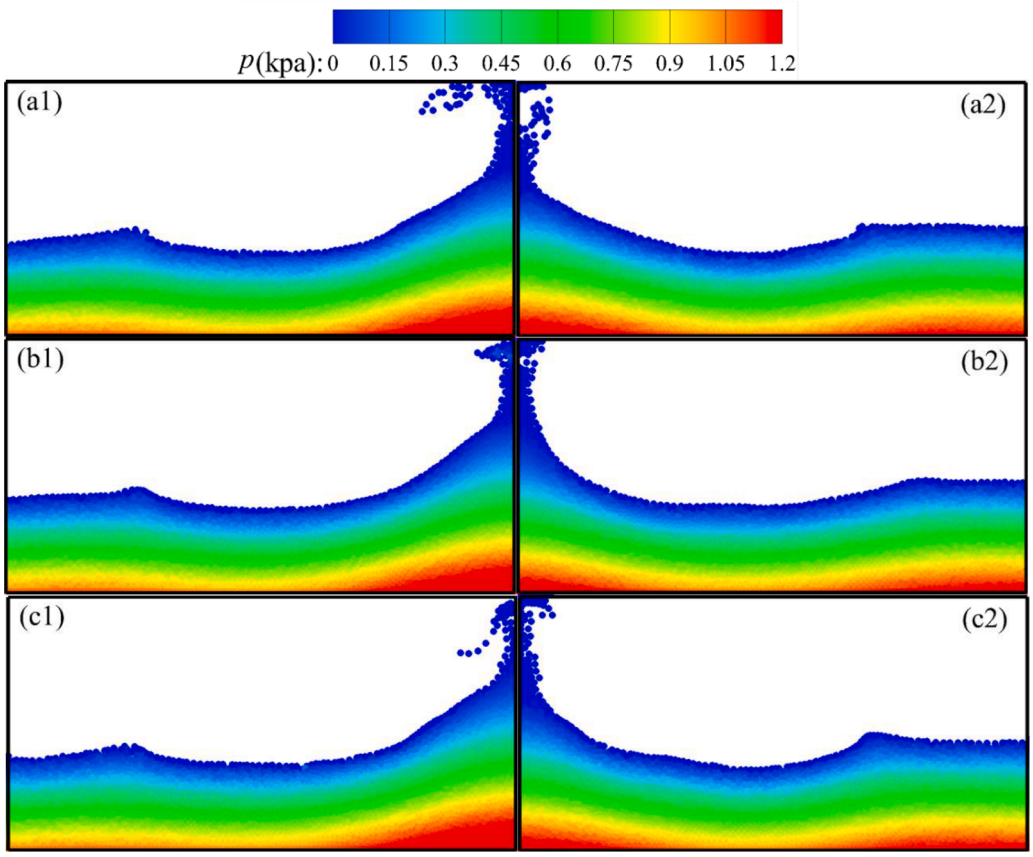

Fig. 28. The particle distributions and pressure contour in the case with violent sloshing: (a) ISPH; (b) ISPH_QSFDI and (c) ISPH_CQ at different time instants: τ = 13.06 (a1, b1 and c1) τ = 15.77 (a2, b2 and c2) (L = 0.6 m, d = 0.2, a = 0.0556 and period of 6.065).

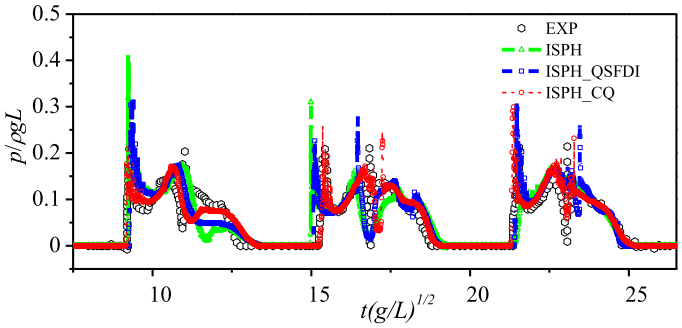

Fig. 29. Comparison of pressure time histories from different ISPH models (L = 0.6 m, d = 0.2, a = 0.0556 and period of 6.065; dx = 1/120, experimental data is duplicated from Kishev et al. [61]).

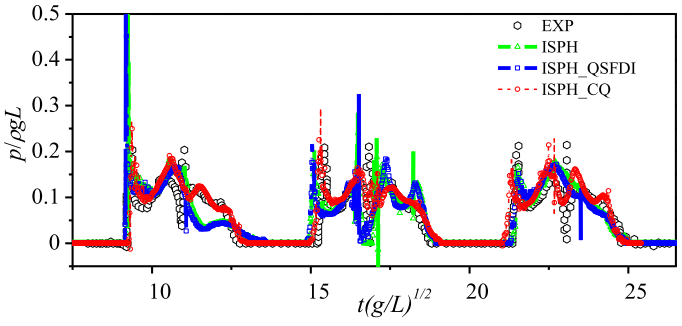

Fig. 30. Comparison of pressure time histories from different ISPH models (L = 0.6 m, d = 0.2, a = 0.0556 and period of 6.065; dx = 1/ 60, experimental data is duplicated from Kishev et al. [61]).

pressure gradient for modelling the free surface flows. Since the gravitational acceleration is uniform, consequently the smoothness of the pressure gradient evaluation is vital to ensure the smoothness of the dynamic pressure. In the classic ISPH and the ISPH_QSFDI simulation, the linear SFDI is applied for evaluating the pressure gradient. Although a larger influence domain may be taken, theoretically the scheme cannot guarantee the smoothness of the pressure gradient. The second order scheme QSFDI applied in the ISPH_CQ is expected to bring about better smoothness for gradient operation. This is confirmed by Fig. 27, in which the suspicious spatial wiggle of the pressure gradient is observed in the results from the classic ISPH and ISPH_QSFDI that apply the linear SFDI, but less evident in the present ISPH_CQ simulation.

Same analysis is also made for the cases with violent sloshing. In this case, the tank is specified by L = 0.6 m and d = 0.2. The motion amplitude and the period of the tank are a = 0.0556 and 6.065 $\sqrt{g/L}$, respectively. The configuration is the same as the experiments carried out by Kishev et al. [61]. Fig. 28 illustrates the snapshots of the particle distributions and the pressure contour at two instants, i.e. τ = 13.06 and τ = 15.77, corresponding to the occurrences of the violent wave impacts on the right and left sides of the tank, respectively. As expected, a smooth pressure field is reproduced by all ISPH models, although the formations of the breaking jet are different in different simulations.

Furthermore, the pressure time histories are recorded by a pressure sensor placed on the wall at z = 0.066 for the relevant simulations. Results are plotted in Fig. 29 for dx = 1/120 and Fig. 30 for dx = 1/60, together with the experimental results by Kishev et al. [61] for comparison. Similar to Ref. [61], we compared the numerical results from the 3rd period with the experimental result to avoid the potential

transient effects commonly occurring in the numerical simulation. It is observed that both the numerical and experimental results are not periodically repeated despite the tank is subjected to a periodic motion. This is a common phenomenon in the violent sloshing. Overall, the agreements between the numerical and the experimental results are acceptable, considering the complex nature of the violent sloshing problem where the random factor play a role; the ISPH_CQ seems to predict a better impact pressure than other ISPH models in terms of the impact magnitude and the time of occurrence. If a coarser particle resolution is applied, e.g. $dx = 1/60$ (Fig. 30), the ISPH_CQ may still deliver a result that reasonably agrees with the experimental data and is better than the other two models, especially in terms of peak pressure in the first period. This implies that the ISPH_CQ can provide a satisfactory result with a coarser computational resolution compared with other two considered ISPH models.

4. Conclusions and discussions

In this paper, the consistent second order ISPH model (ISPH_CQ) is developed for modelling free surface flows. In this model, the QSFDI is consistently adopted to numerically approximate derivatives, involved in the viscous stress, Laplacian operator, velocity divergence and corresponding Neumann boundary conditions and the pressure gradient. The schemes for discretising the Laplacian and approximating the gradient are consistently derived from the second order Taylor's expansion. Compared with the existing consistent second order ISPH model (e.g. [47]), the distinguishing feature is that the QSFDI requires inversion of matrices with much smaller sizes, i.e., 2×2 for 2D problems and 3×3 for 3D problems, compared with existing models, e.g., 5×5 for 2D problems in Ref. [47]. This feature is expected to bring about considerable benefit for improving computational robustness.

The performance of the present ISPH_CQ is examined by using various cases with free surface, including the oscillating drop, solitary/regular wave propagations and liquid sloshing with or without wave breaking. Its accuracy and convergence properties are compared with the classic ISPH [25] and the ISPH_QSFDI [26]. It is concluded that the ISPH_CQ leads to a significantly higher accuracy providing the same particle resolution or requires shorter CPU time to secure the same level of accuracy, than other ISPH models. Although the convergence property may be downgraded by the linear temporal schemes when the particle resolution is sufficiently high, the linear accuracy of the discretisation of the viscous term and, in some cases, the imprecise enforcement of the dynamic free surface boundary condition, the ISPH_CQ exhibits a second order convergence rate over a limited range of conditions for practical simulation of free surface problems, which has not been observed in literature for ISPH applications, to the best of our knowledge. The numerical investigations also show that the ISPH_CQ has better performance than the other two considered ISPH models in terms of mass/volume and energy conservation and reproducing smooth pressure field under the same conditions. The ISPH_CQ will be demonstrated to have similar good performance in three dimensional cases and/or using MPI parallelisation (GPU computing) in our future work.

CRediT authorship contribution statement

Ningbo Zhang: Investigation, Validation, Writing – original draft. **Shiqiang Yan:** Conceptualization, Methodology, Writing – review & editing. **Qingwei Ma:** Supervision, Writing – review & editing, Funding acquisition. **Abbas Khayyer:** Writing – review & editing. **Xiaohu Guo:** Writing – review & editing. **Xing Zheng:** Resources.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests:

Shiqiang Yan reports financial support was provided by Engineering and Physical Sciences Research Council. Qingwei Ma reports financial support was provided by Engineering and Physical Sciences Research Council. Ningbo Zhang reports financial support was provided by Engineering and Physical Sciences Research Council

Data availability

Data will be made available on request.

Acknowledgment

The authors acknowledge the financial support of EPSRC projects (EP/T026782, EP/T00424X and EP/V040235).

Appendix: Derivation of QSFDI

For each particle j with a position vector of $\mathbf{r}_j$, that is located inside the influence domain $\Omega_i$of the particle i at $\mathbf{r}_i$, a function P can be expressed as a second order Taylor's expanesion, i.e. Eq. (11). Multiplying Eq. (11) by $w(\boldsymbol{r}_{ji})\boldsymbol{r}_{ji}^{(2c)}/d_{ji}^4$, where $w(\boldsymbol{r}_{ji})$ is the weighting function for particle j related to $\mathbf{r}_i$, d_{ji} is the distance between particle i and its neighbouring particle and ignoring the truncation error and taking the sum of resultant equations for all particles in the influence domain of i, yields

$$\nabla^{(2c)}P_i \approx \boldsymbol{M}_{2c,i}^{-1}\sum_{j=1}^{N}\frac{W(\mathbf{r}_{ji})}{d_{ji}^4}\boldsymbol{r}_{ji}^{(2c)}\left(P_j-P_i\right)-\boldsymbol{M}_{2c,I}^{-1}\sum_{j=1}^{N}\frac{W(\mathbf{r}_{ji})}{d_{ji}^4}\boldsymbol{r}_{ji}^{(2c)}\boldsymbol{r}_{ji}^{\mathrm{T}}\nabla P_i-\frac{1}{2}\boldsymbol{M}_{2c,i}^{-1}\sum_{j=1}^{N}\frac{W(\mathbf{r}_{ji})}{d_{ji}^4}\boldsymbol{r}_{ji}^{(2c)}\left(\boldsymbol{r}_{ji}^{(2s)}\right)^{\mathrm{T}}\nabla^{(2s)}P_i-\frac{1}{6}\boldsymbol{M}_{2c,I}^{-1}\sum_{j=1}^{N}\frac{W(\mathbf{r}_{ji})}{d_{ji}^4}\boldsymbol{r}_{ji}^{(2c)}\left(\boldsymbol{r}_{ji}^{\mathrm{T}}\nabla\right)^3P_i \tag{A1}$$

in which $\boldsymbol{M}_{2c,i}=\sum_{j=1}^{N}\frac{W(\mathbf{r}_{ji})}{d_{ji}^4}\boldsymbol{r}_{ji}^{(2c)}\left(\boldsymbol{r}_{ji}^{(2c)}\right)^{\mathrm{T}}$. For convenience, $(\boldsymbol{r}_{ji}^{\mathrm{T}}\nabla)^3P_i$ is re-written as$(\boldsymbol{r}_{ji}^{(3)})^{\mathrm{T}}\nabla^{(3)}P_i$, where $\boldsymbol{r}_{ji}^{(3)}=[x_{ji}^3\ 3x_{ji}^2y_{ji}\ 3x_{ji}^2z_{ji}\ 3x_{ji}y_{ji}^2\ 6x_{ji}y_{ji}z_{ji}\ 3x_{ji}z_{ji}^2\ y_{ji}^3\ 3y_{ji}^2z_{ji}\ 3y_{ji}z_{ji}^2\ z_{ji}^3]^{\mathrm{T}}$, $\nabla^{(3)}=[\frac{\partial^3}{\partial x^3}\ \frac{\partial^3}{\partial x^2\partial y}\ \frac{\partial^3}{\partial x^2\partial z}\ \frac{\partial^3}{\partial x\partial y^2}\ \frac{\partial^3}{\partial x\partial y\partial z}\ \frac{\partial^3}{\partial x\partial z^2}\ \frac{\partial^3}{\partial y^3}\ \frac{\partial^3}{\partial y^2\partial z}\ \frac{\partial^3}{\partial y\partial z^2}\ \frac{\partial^3}{\partial z^3}]^{\mathrm{T}}$ are two 10×1 matrices. Substituting Eq. (A1) into Eq. (1), it leads to

$$P_j-P_i \approx \left(\boldsymbol{r}_{ji}^{(2c)}\right)^{\mathrm{T}}\boldsymbol{M}_{2c,i}^{-1}\sum_{k=1}^{N}\frac{W(\mathbf{r}_{ki})}{d_{ki}^4}\boldsymbol{r}_{ki}^{(2c)}\left(P_k-P_i\right)+\boldsymbol{G}_{jI}^{\mathrm{T}}\nabla P_I+\frac{1}{2}\boldsymbol{\Pi}_{jI}^{\mathrm{T}}\nabla^{(2s)}P_I+\frac{1}{6}\boldsymbol{F}_{jI}^{\mathrm{T}}\nabla^{(3)}P_i \tag{A2}$$

where,

$$\boldsymbol{\Pi}_{ji} = \left\{ \left(\mathbf{r}_{ji}^{(2s)}\right)^{\mathrm{T}} - \left(\mathbf{r}_{ji}^{(2c)}\right)^{\mathrm{T}} \boldsymbol{M}_{2c,i}^{-1} \sum_{k=1}^{N} \frac{W(\mathbf{r}_{ki})}{d_{ki}^4} \mathbf{r}_{ki}^{(2c)} \left(\mathbf{r}_{ji}^{(2s)}\right)^{\mathrm{T}} \right\}^{\mathrm{T}},$$

$$\boldsymbol{G}_{ji} = \left\{ \mathbf{r}_{ji}^{\mathrm{T}} - \left(\mathbf{r}_{ji}^{(2c)}\right)^{\mathrm{T}} \boldsymbol{M}_{2c,i}^{-1} \sum_{k=1}^{N} \frac{W(\mathbf{r}_{ki})}{d_{ki}^4} \mathbf{r}_{ki}^{(2c)} \mathbf{r}_{ki}^{T} \right\}^{\mathrm{T}},$$

$$\boldsymbol{F}_{ji} = \left\{ \left(\mathbf{r}_{ji}^{(3)}\right)^{\mathrm{T}} - \left(\mathbf{r}_{ji}^{(2c)}\right)^{\mathrm{T}} \boldsymbol{M}_{2c,I}^{-1} \sum_{k=1}^{N} \frac{W(\mathbf{r}_{ki})}{d_{ki}^4} \mathbf{r}_{ki}^{(2c)} \left(\mathbf{r}_{ki}^{(3)}\right)^{\mathrm{T}} \right\}^{\mathrm{T}}$$

Multiplying Eq. (A2) by $w(\boldsymbol{r}_{ji})\boldsymbol{\Pi}_{ji}/d_{ji}^4$ and taking the sum of resultant equations for all particles in the influence domain of i, it leads to the expression of $\nabla^{(2s)}P_i$,

$$\nabla^{(2s)}P_i \approx 2\boldsymbol{M}_{2s,i}^{-1} \sum_{j=1}^{N} \boldsymbol{\Gamma}_{jI}\left(P_j - P_i\right) - 2\boldsymbol{M}_{2s,i}^{-1} \sum_{j=1}^{N} \frac{W\left(\mathbf{r}_{ji}\right)}{d_{ji}^4} \boldsymbol{\Pi}_{ji}\boldsymbol{G}_{ji}^{\mathrm{T}}\nabla P_i - \frac{1}{3}\boldsymbol{M}_{2s,i}^{-1} \sum_{j=1}^{N} \frac{W\left(\mathbf{r}_{ji}\right)}{d_{ji}^4} \boldsymbol{\Pi}_{ji}\boldsymbol{F}_{ji}^{\mathrm{T}}\nabla^{(3)}P_i, \tag{A3}$$

where, $\boldsymbol{M}_{2s,i} = \sum_{j=1}^{N} \frac{W(\mathbf{r}_{ji})}{d_{ji}^4} \boldsymbol{\Pi}_{ji}\boldsymbol{\Pi}_{ji}^{\mathrm{T}}$ and $\boldsymbol{\Gamma}_{jI} = \left(\frac{W(\mathbf{r}_{ji})}{d_{ji}^4} \boldsymbol{\Pi}_{ji} - \boldsymbol{\Pi}_{ji}\left(\boldsymbol{r}_{ji}^{(2c)}\right)^{\mathrm{T}} \boldsymbol{M}_{2c,i}^{-1} \sum_{k=1}^{N} \frac{W(\mathbf{r}_{ki})}{d_{ki}^4} \boldsymbol{r}_{ki}^{(2c)} \right)$. Substituting Eqs. (A3) to (A2) leads to

$$P_j - P_i \approx \left(\boldsymbol{r}_{ji}^{(2c)}\right)^{\mathrm{T}} \boldsymbol{M}_{2c,i}^{-1} \sum_{k=1}^{N} \frac{W(\mathbf{r}_{ki})}{d_{ki}^4} \boldsymbol{r}_{ki}^{(2c)}(P_k - P_i) + \boldsymbol{\Pi}_{ji}^{\mathrm{T}}\boldsymbol{M}_{2s,i}^{-1} \sum_{k=1}^{N} \boldsymbol{\Gamma}_{ki}(P_k - P_i) + \left(\boldsymbol{G}_{ji}^{\mathrm{T}} - \boldsymbol{\Pi}_{ji}^{\mathrm{T}}\boldsymbol{M}_{2s,i}^{-1} \sum_{k=1}^{N} \frac{W(\mathbf{r}_{ki})}{d_{ki}^4} \boldsymbol{\Pi}_{ki}\boldsymbol{G}_{ki}^{\mathrm{T}} \right) \nabla P_i + \frac{1}{6} \left(\boldsymbol{F}_{ji}^{\mathrm{T}} - \boldsymbol{\Pi}_{ji}^{\mathrm{T}}\boldsymbol{M}_{2s,i}^{-1} \times \sum_{k=1}^{N} \frac{W(\mathbf{r}_{ki})}{d_{ki}^4} \boldsymbol{\Pi}_{ki}\boldsymbol{F}_{ki}^{\mathrm{T}} \right) \nabla^{(3)}P_i \tag{A4}$$

Multiplying the further modified Eq. (A4) by $w(\boldsymbol{r}_{jI})\mathbf{q}_{ji}/d_{jI}^2$, where the coefficient $\mathbf{q}_{ji} = \left(\boldsymbol{G}_{ji}^{\boldsymbol{T}} - \boldsymbol{\Pi}_{ji}^{\mathrm{T}}\boldsymbol{M}_{2s,i}^{-1} \sum_{k=1}^{N} \frac{W(\mathbf{r}_{ki})}{d_{ki}^4} \boldsymbol{\Pi}_{ki}\boldsymbol{G}_{ki}^{\mathrm{T}} \right)^{\mathrm{T}}$, and taking the sum of resultant equations for all particles in the influence domain of i, it leads to the expression to approximate the gradient with a term associated with $(\mathbf{r}_{ji}^{\mathrm{T}}\nabla)^3 P_i$

$$\langle \nabla P_i \rangle = \boldsymbol{M}_{1q,i}^{-1} \sum_{j=1}^{N} \frac{W\left(\mathbf{r}_{ji}\right)}{d_{ji}^2} \boldsymbol{q}_{ji}\left(P_j - P_i\right) - \boldsymbol{M}_{1q,i}^{-1} \sum_{j=1}^{N} \frac{W\left(\mathbf{r}_{ji}\right)}{d_{jI}^2} \boldsymbol{q}_{ji} \left(\boldsymbol{r}_{ji}^{(2c)}\right)^{\mathrm{T}} \boldsymbol{M}_{2c,i}^{-1} \sum_{k=1}^{N} \frac{W(\mathbf{r}_{ki})}{d_{kI}^4} \boldsymbol{r}_{ki}^{(2c)}(P_k - P_i) - \boldsymbol{M}_{1q,i}^{-1} \sum_{j=1}^{N} \frac{W\left(\mathbf{r}_{ji}\right)}{d_{ji}^2} \boldsymbol{q}_{ji}\boldsymbol{\Pi}_{ji}^{\mathrm{T}}\boldsymbol{M}_{2s,i}^{-1} \sum_{k=1}^{N} \boldsymbol{\Gamma}_{ki}(P_k - P_I) \tag{A5}$$

$$\boldsymbol{E}_{\nabla p_i} = -\frac{1}{6}\boldsymbol{M}_{1q,i}^{-1} \sum_{j=1}^{N} \frac{W\left(\mathbf{r}_{ji}\right)}{d_{ji}^2} \boldsymbol{q}_{ji} \left(\boldsymbol{F}_{ji}^{\mathrm{T}} - \boldsymbol{\Pi}_{ji}^{\mathrm{T}}\boldsymbol{M}_{2s,i}^{-1} \sum_{k=1}^{N} \frac{W(\mathbf{r}_{ki})}{d_{ki}^4} \boldsymbol{\Pi}_{ki}\boldsymbol{F}_{ki}^{\mathrm{T}} \right) \nabla^{(3)} p_i \tag{A6}$$

where, $\boldsymbol{M}_{1q,i} = \sum_{j=1}^{N} \frac{W(\mathbf{r}_{ji})}{d_{ji}^2} \boldsymbol{q}_{ji}\boldsymbol{q}_{ji}^{\mathrm{T}}$. Substituting Eqs. (A5) to (A3), it leads to the formula to approximate$\nabla^{(2s)}p_i$, i.e.,

$$\nabla^{(2s)}P_i = 2\boldsymbol{M}_{2s,i}^{-1} \sum_{j=1}^{N} \boldsymbol{\Gamma}_{ji}\left(P_j - P_i\right) - 2\boldsymbol{M}_{2s,i}^{-1} \sum_{j=1}^{N} \frac{W\left(\mathbf{r}_{ji}\right)}{d_{jI}^4} \boldsymbol{\Pi}_{ji}\boldsymbol{G}_{ji}^{\mathrm{T}} \langle \nabla P_i \rangle \tag{A7}$$

with its leading truncation error

$$\boldsymbol{E}_{\nabla^{(2s)}P_i} = -\frac{1}{3}\boldsymbol{M}_{2s,i}^{-1} \sum_{j=1}^{N} \frac{W\left(\mathbf{r}_{ji}\right)}{d_{ji}^4} \boldsymbol{\Pi}_{ji}\boldsymbol{F}_{ji}^{\mathrm{T}}\nabla^{(3)}P_i - 2\boldsymbol{M}_{2s,i}^{-1} \sum_{j=1}^{N} \frac{W\left(\mathbf{r}_{ji}\right)}{d_{ji}^4} \boldsymbol{\Pi}_{ji}\boldsymbol{G}_{ji}^{\mathrm{T}}\boldsymbol{E}_{\nabla p_i} \tag{A8}$$

The Laplacian can therefore be approximated by using

$$\left\langle \nabla^2 P_i \right\rangle = \boldsymbol{I}^{\mathrm{T}} \left\langle \nabla^{(2s)} P_i \right\rangle \tag{A9}$$

where, $\boldsymbol{I} = [1 \quad 1 \quad 1]^{\mathrm{T}}$. The corresponding leading truncation error is

$$\boldsymbol{E}_{\langle \nabla^2 P_i \rangle} = \boldsymbol{I}^{\mathrm{T}}\boldsymbol{E}_{\nabla^{(2s)}P_i} \tag{A10}$$

In practice, Eq. (A9) can be applied to discretise the Poisson's equation at all particle positions and/or to directly approximate $\nabla^2 P(\boldsymbol{x})$ at a point $\boldsymbol{x}_i$ coinciding with a particle location, where P_i is known. However, to estimate $\nabla^2 p(\boldsymbol{x})$ at a point that does not coincide with any particles, P_i needs to be numerically interpolated using P_j. To do so, estimation of $\nabla^{(2c)}P_i$ in Eq. (1) is required and achieved by substituting Eqs. (A5)–(A8) to Eq. (A1),

$$\left\langle \nabla^{(2c)} p(\boldsymbol{x})\big|_{\boldsymbol{x}=\boldsymbol{x}_I} \right\rangle = \boldsymbol{M}_{2c,I}^{-1} \sum_{j=1}^{N} \frac{w_{jI}}{d_{jI}^4} \boldsymbol{r}_{jI}^{(2c)}\left(p_j - p_I\right) - \boldsymbol{M}_{2c,I}^{-1} \sum_{j=1}^{N} \frac{w_{jI}}{d_{jI}^4} \boldsymbol{r}_{jI}^{(2c)} \boldsymbol{r}_{jI}^{\mathrm{T}} \left\langle \nabla p(\boldsymbol{x})\big|_{\boldsymbol{x}=\boldsymbol{x}_I} \right\rangle - \frac{1}{2}\boldsymbol{M}_{2c,I}^{-1} \sum_{j=1}^{N} \frac{w_{jI}}{d_{jI}^4} \boldsymbol{r}_{jI}^{(2c)} \left(\boldsymbol{r}_{jI}^{(2s)}\right)^{\mathrm{T}} \left\langle \nabla^{(2s)} p(\boldsymbol{x})\big|_{\boldsymbol{x}=\boldsymbol{x}_I} \right\rangle \tag{A11}$$

For convenience of deriving the interpolation function, Eqs. (A5), (A7) and (A11) are, respectively, rewritten in a summation form, i.e.

$$\left\langle \nabla p(\boldsymbol{x})\big|_{\boldsymbol{x}=\boldsymbol{x}_I} \right\rangle = \sum_{j=1}^{N} \boldsymbol{\Phi}_{jI}^{g}\left(p_j - p_I\right) \tag{A13}$$

$$\left\langle \nabla^{(2s)} p(\boldsymbol{x})\big|_{\boldsymbol{x}=\boldsymbol{x}_I} \right\rangle = \sum_{j=1}^{N} \boldsymbol{\Phi}^{s}_{jI}\left(p_j - p_I\right) \tag{A14}$$

$$\left\langle \nabla^{(2c)} p(\boldsymbol{x})\big|_{\boldsymbol{x}=\boldsymbol{x}_I} \right\rangle = \sum_{j=1}^{N} \boldsymbol{\Phi}^{c}_{jI}\left(p_j - p_I\right) \tag{A15}$$

Consequently, Eq. (A9) can be re-written as

$$\left\langle \nabla^{2} p(\boldsymbol{x})\big|_{\boldsymbol{x}=\boldsymbol{x}_I} \right\rangle = \boldsymbol{I}^{\mathrm{T}} \sum_{j=1}^{N} \boldsymbol{\Phi}^{s}_{jI}\left(p_j - p_I\right) \tag{A16}$$

References

[1] Cummins SJ, Rudman M. An SPH projection method. J Comput Phys 1999;152(2):584–607.
[2] Shao S, Lo EY. Incompressible SPH method for simulating Newtonian and non-Newtonian flows with a free surface. Adv Water Resour 2003;26(7):787–800.
[3] Lucy LB. A numerical approach to the testing of fusion process. Astron J 1997;88:1013–24.
[4] Gingold RA, Monaghan JJ. Smoothed particle hydrodynamics - theory and application to non-spherical stars. Mon Not R Astron Soc 1977;181(3):375–89.
[5] Lind SJ, Stansby PK. High-order Eulerian incompressible smoothed particle hydrodynamics with transition to Lagrangian free-surface motion. J Comput Phys 2016;326:290–311.
[6] Zheng X, Ma QW, Duan WY. Incompressible SPH method based on Rankine source solution for violent water wave simulation. J Comput Phys 2014;276:291–314.
[7] Malcevic I, Ghattas O. Dynamic-mesh finite element method for Lagrangian computational fluid dynamics. Finite Elem Anal Des 2002;38(10):965–82.
[8] Colomés O, Badia S, Codina R, Principe J. Assessment of variational multiscale models for the large eddy simulation of turbulent incompressible flows. Comput Methods Appl Mech Eng 2015;285:32–63.
[9] Lee ES, Moulinec C, Xu R, Violeau D, Laurence D, Stansby P. Comparisons of weakly compressible and truly incompressible algorithms for the SPH mesh free particle method. J Comput Phys 2008;227:8417–36.
[10] Violeau D, Leroy A. Optimal time step for incompressible SPH. J Comput Phys 2015;288:119–30.
[11] Zheng X, Lv XP, Ma QW, Duan WY, Khayyer A, Shao SD. An improved solid boundary treatment for wave-float interactions using ISPH method. Int J Naval Archit Ocean Eng 2018;10(3):329–47.
[12] Gotoh H, Khayyer A. Current achievements and future perpectives for projection-based particle methods with applications in ocean engineering. J Ocean Eng Mar Energy 2016;2(3):251–78.
[13] Lo EY, Shao SD. Simulation of near-shore solitary wave mechanics by an incompressible SPH method. Appl Ocean Res 2002;24(5):275–86.
[14] Lind SJ, Xu R, Stansby PK, Rogers BD. Incompressible smoothed particle hydrodynamics for free-surface flows: a generalised diffusion-based algorithm for stability and validations for impulsive flows and propagating waves. J Comput Phys 2012;231(4):1499–523.
[15] Rafiee A, Cummins S, Rudman M, Thiagarajan K. Comparative study on the accuracy and stability of SPH schemes in simulating energetic free-surface flows. Eur J Mech B/Fluids 2012;36(10):1–16.
[16] Gui QQ, Dong P, Shao SD, Chen YQ. Incompressible SPH simulation of wave interaction with porous structure. Ocean Eng 2015;110:126–39.
[17] Liang DF, Jian W, Shao SD, Chen RD, Yang KJ. Incompressible SPH simulation of solitary wave interaction with movable seawalls. J Fluids Struct 2017;69:72–88.
[18] Khayyer A, Gotoh H, Shimizu Y. Comparative study on accuracy and conservation properties of two particle regularization schemes and proposal of an optimized particle shifting scheme in ISPH context. J Comput Phys 2017;332(1):236–56. a.
[19] Khayyer A, Gotoh H, Shimizu Y, Gotoh K. On enhancement of energy conservation properties of projection-based particle methods. Eur J Mech B/Fluids 2017;66:20–37. b.
[20] Khayyer A, Gotoh H, Falahaty H, Shimizu Y. An enhanced ISPH-SPH coupled method for simulation of incompressible fluid-elastic structure interactions. Comput Phys Commun 2018;232:139–64.
[21] Khayyer A, Shimizu Y, Gotoh H, Nagashima K. A coupled incompressible SPH-Hamiltonian SPH solver for hydroelastic FSI corresponding to composite structures. Appl Math Model 2021;94:242–71.
[22] Luo M, Khayyer A, Lin PZ. Particle methods in ocean and coastal engineering. Appl Ocean Res 2021;114:102734.
[23] Cummins SJ, Rudman M. An SPH projection method. J Comput Phys 1999;152(2):584–607.
[24] Koh CG, Gao M, Luo C. A new particle method for simulation of incompressible free surface flow problems. Int J Numer Methods Eng 2012;89(12):1582–604.
[25] Luo M, Reeve DE, Shao SD, Karunarathna H, Lin PZ, Cai H. Consistent Particle Method simulation of solitary wave impinging on and overtopping a seawall. Eng Anal Bound Elem 2019;103(2019):160–71.
[26] Nasar AMA, Fourtakas G, Lind SJ, King JRC, Rogers BD, Stansby PK. High-order consistent SPH with the pressure projection method in 2-D and 3-D. J Comput Phys 2021;444:110563.
[27] Khayyer A, Gotoh H. Enhancement of stability and accuracy of the moving particle semi-implicit method. J Comput Phys 2011;230(8):3093–118.
[28] Zhang NB, Zheng X, Ma QW, et al. A hybrid stabilization technique for simulating water wave–structure interaction by incompressible smoothed particle hydrodynamics (ISPH) method. J Hydro-Environ Res 2018;18:77–94.
[29] Zhang NB, Yan SQ, Ma QW, Zheng X. A QSFDI based Laplacian discretisation for modelling wave-structure interaction using ISPH. Appl Ocean Res 2021;117:102954.
[30] Ma QW, Zhou Y, Yan SQ. A review on approaches to solving Poisson's equation in projection-based meshless methods for modelling strongly nonlinear water waves. J Ocean Eng Mar Energy 2016;2(3):279–99.
[31] Ma QW. Meshless local Petrov–Galerkin method for two-dimensional nonlinear water wave problems. J Comput Phys 2005;205(2):611–25. a.
[32] Ma QW. MLPG method based on Rankine source solution for simulating nonlinear water waves. CMES-Comput Model Eng Sci 2005;9(2):193–209. b.
[33] Ma QW. A new meshless interpolation scheme for MLPG_R method. CMES Comput Model Eng Sci 2008;23(2):75–89.
[34] Ma QW, Zhou JT. MLPG_R method for numerical simulation of 2D breaking waves. CMES Comput Model Eng Sci 2009;43(3):277–303.
[35] Sriram V, Ma QW. Review on the local weak form-based meshless method (MLPG): developments and applications in ocean engineering. Appl Ocean Res 2021;116:102883.
[36] Pan X., Zhou Y., Dong P., Shi H. An improved impermeable solid boundary scheme for Meshless Local Petrov-Galerkin method. Eur J Mech B/Fluids 2022; 96:94–105.
[37] Quinlan N, Basa M, Lastiwka M. Truncation error in mesh-free particle methods. Int J Numer Methods Eng 2006;66(13):2064–85.
[38] Vacondio R, Altomare C, De Leffe M, et al. Grand challenges for smoothed particle hydrodynamics numerical schemes. Comput Part Mech 2021 2021;8:575–88.
[39] Fourtakas G, Stansby PK, Rogers BD, Lind SJ. An Eulerian–Lagrangian incompressible SPH formulation (ELI-SPH) connected with a sharp interface. Comput Methods Appl Mech Eng 2018;329:532–52.
[40] Chen JK, Beraun JE, Carney TC. A corrective smoothed particle method for boundary value problems in heat conduction. Int J Numer Methods Eng 1999;46(2):231–52.
[41] Schwaiger HF. An implicit corrected SPH formulation for thermal diffusion with linear free surface boundary conditions. Int J Numer Methods Eng 2008;75(6):647–71.
[42] Fatehi R, Manzari MT. Error estimation in smoothed particle hydro-dynamics and a new scheme for second derivatives. Comput Math Appl 2011;61(2):482–98.
[43] Tamai T, Koshizuka S. Least squares moving particle semi-implicit method. Comput Part Mech 2014;1(3):277–305.
[44] Tamai T, Murotani K, Koshizuka S. On the consistency and convergence of particle-based meshfree discretization schemes for the Laplace operator. Comput Fluids 2017;142:79–85.
[45] SQ Yan, Ma QW, Wang JH. Quadric SFDI for Laplacian discretisation in Lagrangian meshless methods. J Mar Sci Appl 2020;19(3):362–80.
[46] Zheng X, Shao S, Khayyer A, Duan WY, Ma QW, Liao KP. Corrected first-order derivative ISPH in water wave simulations. Coast Eng J 2017;59(1):1750010.
[47] Shimizu Y, Gotoh H, Khayyer A, Kita K. Fundamental investigation on the applicability of a higher-order consistent ISPH method. Int J Offshore Polar Eng 2022;32(03):275–84.
[48] Xu R, Stansby P, Laurence D. Accuracy and stability in incompressible SPH (ISPH) based on the projection method and a new approach. J Comput Phys 2009;228(18):6703–25.
[49] Sibilla S. An algorithm to improve consistency in smoothed particle hydrodynamics. Comput Fluids 2015;118:148–58.
[50] Cleary PW, Monaghan JJ. Conduction modelling using smoothed particle hydrodynamics. J Comput Phys 1999;148(1):227–64.
[51] Monaghan JJ. Smoothed particle hydrodynamics. Rep Progr Phys 2005;68(8):1703.
[52] Van der Vorst HA. Bi-CGSTAB: a fast and smoothly converging variant of Bi-CG for solution of non-symmetric linear system. SIAM J Sci Stat Comput 1992;13:631–44.
[53] Monaghan JJ, Lattanzio JC. A refined method for astrophysical problems. Astron Astrophys 1985;149(1):135–43.

[54] Monaghan JJ, Rafiee A. A simple SPH algorithm for multi-fluid flow with high density ratios. Int J Numer Methods Fluids 2013;71(5):537–61.
[55] Lee JJ, Skjelbreia JE, Raichlen F. Measurement of velocities in solitary waves. J Waterw Port Coast Ocean Div 1982;108(2):200–18.
[56] Gotoh H, Shao SD, Memita T. SPH-LES model for numerical investigation of wave interaction with partially immersed breakwater. Coast Eng J 2004;46(1):39–63.
[57] Wei G, Kirby JT. Time-dependent numerical code for extended Boussinesq equations. J Waterw Port Coast Ocean Eng 1995;121(5):251–61.
[58] Madsen OS. On the generation of long waves. J Geophys Res 1971;76(36):8672–83.
[59] Faltinsen OM. A numerical nonlinear method of sloshing in tanks with two-dimensional flow. J Ship Res 1978;22(03):193–202.
[60] Wu GX, Ma QW, Taylor RE. Numerical simulation of sloshing waves in a 3D tank based on a finite element method. Appl Ocean Res 1998;20(6):337–55.
[61] Kishev ZR, Hu C, Kashiwagi M. Numerical simulation of violent sloshing by a CIP-based method. J Mar Sci Technol 2006;11(2):111–22.
[62] Yan S, Ma QW. QALE-FEM for modelling 3D overturning waves. Int J Numer Methods Fluids 2010;63:743–68.
[63] Fourtakas G, Rogers BD, Nasar AMA. Towards pseudo-spectral incompressible smoothed particle hydrodynamics (ISPH). Comput Phys Commun 2021;226:108028.
[64] Wessels H, Weißenfels C, Wriggers P. The neural particle method–an updated Lagrangian physics informed neural network for computational fluid dynamics. Comput Methods Appl Mech Eng 2020;368:113127.
[65] Li ZJ, Farimani AB. Graph neural network – accelerated Lagrangian fluid simulation. Comput Graph 2022;103:201–11.
[66] Guo XH, Rogers BD, Lind SJ, Stansby PK. New massively parallel scheme for incompressible smoothed particle hydrodynamics (ISPH) for highly nonlinear and distorted flow. Comput Phys Commun 2018;233:16–28.
[67] Chow AD, Rogers BD, Lind SJ, Stansby PK. Incompressible SPH (ISPH) with fast Poisson solver on a GPU. Comput Phys Commun 2018;226:81–103.
[68] Monteleone A, Burriesci G, Napoli E. A distributed-memory MPI parallelization scheme for multi-domain incompressible SPH. J Parallel Distrib Comput 2022;170:53–67.