Incompressible SPH method based on Rankine source solution for violent water wave simulation

X. Zheng[1], Q.W. Ma[2*] , W.Y. Duan[1]

1. College of Shipbuilding Engineering, Harbin Engineering University, Harbin 150001, China

2. Schools of Engineering and Mathematical Science, City University London, London EC1V 0HB, UK

Abstract

With wide applications, the smoothed particle hydrodynamics method (abbreviated as SPH) has become an important numerical tool for solving complex flows, in particular those with a rapidly moving free surface. For such problems, the incompressible Smoothed Particle Hydrodynamics (ISPH) has been shown to yield better and more stable pressure time histories than the traditional SPH by many papers in literature. However, the existing ISPH method directly approximates the second order derivatives of the functions to be solved by using the Poisson equation. The order of accuracy of the method becomes low, especially when particles are distributed in a disorderly manner, which generally happens for modelling violent water waves. This paper introduces a new formulation using the Rankine source solution. In the new approach to the ISPH, the Poisson equation is first transformed into another form that does not include any derivative of the functions to be solved, and as a result, does not need to numerically approximate derivatives. The advantage of the new approach without need of numerical approximation of derivatives is obvious, potentially leading to a more robust numerical method. The newly formulated method is tested by simulating various water waves, and its convergent behaviours are numerically studied in this paper. Its results are compared with experimental data in some cases and reasonably good agreement is achieved. More importantly, numerical results clearly show that the newly developed method does need less number of particles and so less computational costs to achieve the similar level of accuracy, or to produce more accurate results with the same number of particles compared with the traditional SPH and existing ISPH when it is applied to modelling water waves.

Keywords: Meshless method; SPH; ISPH; ISPH_R; Free surface flow; Wave impact; Violent water waves

1. Introduction

Smoothed Particle Hydrodynamics (SPH) is a Lagrangian meshless particle method. It was originally developed to simulate astrodynamics [1, 2] but has been extended to model dynamics problems with violent motions in many areas [3-23]. The work on the method has been continuously reviewed by many authors. We mainly take some of these related to water wave modelling here.

When the SPH is applied to modelling water waves, there are largely two different formulations in literature. The first one is weakly compressible SPH (WCSPH, also called traditional SPH in this paper), in which water is considered as slightly compressible and its pressure is related to its density through an equation of state with artificially specified sound speed [6]. The second formulation is incompressible SPH, also called ISPH, in which water is considered as incompressible and having constant density with pressure found by solving a boundary value problem. As indicated by many researchers, e.g., Rafiee et al. [13] and Lee et al. [22], the WCSPH has several advantages, such as that it is easy to be programmed and does not need to solve pressure boundary value problem. However, it has at least two weaknesses [13, 22]: (a) requiring use of very small time steps and (b) resulting in significant spurious pressure fluctuations in space and time domain. The first one is inherent because the sound speed in the equation of state has to be large enough (even though much smaller than real sound speed), and so leads to smaller time steps. This is also related to that fact that the traditional SPH usually needs a large number of particles; in other words, very small distance between particles or very high resolution to obtain sufficiently accurate results. The more the particles in a given computational domain, the smaller the time steps should be used. The second weakness is perhaps because the pressure is very sensitive to the density and so a small error in density can lead to a significant error in pressure, inducing spurious pressure fluctuations. The spurious pressure fluctuations do not only give wrong values of pressure but can also cause simulation to become unstable when the method is employed to model wave-structure interactions. A great amount of effort has been made to overcome the weaknesses, for example, use of K2_SPH [8] and SPH based on the solution for Riemann problem [9-12]. K2_SPH improves the accuracy of kernel approximation for partial derivatives in continuity equation and momentum equations, and so can give more accurate results of density and velocity. In the SPH based on the solution for a Riemann problem, the Riemann problem is solved for each pair of particles to calculate the associated parameters. Basically, this approach improves accuracy of estimating the gradient involved. Its results are much more accurate and smoother than these from the traditional SPH, but it is significantly more expensive (in the order of 5 or 6 times) than the latter [13].

The ISPH has also been widely applied in the field of water wave dynamics [15-23]. This method projects the intermediate velocity field onto a divergence-free space by solving a Poisson equation for pressure. According to comparative studies carried out by Lee et al [22], the time step used for the ISPH can be much larger (50 times larger in one of cases presented by them). In addition, the results from ISPH can be much more accurate than these from the WCSPH for a given number of particles. In other words, the convergence rate of ISPH results is much higher than that of the WCSPH. The drawback of this formulation is obvious as it needs to solve the boundary value problem defined by the Poisson equation at each time step, which is recognised to consume a significant amount of computational time. Nevertheless, the total computational time taken by the ISPH can be shorter than that by the WCSPH, as indicated also by Lee et al. [22]. However, the second order derivatives of pressure need to be approximated when discretising the Poisson equation. In all publications found so far in literature about ISPH, the second derivatives are directly approximated using a scheme similar to that for finite difference method. No matter what scheme to be used, direct numerical approximation to second derivatives always has a difficulty with accurately modelling the functions to be solved, in particular when particles are distributed in a disorderly manner. Distribution of particles always becomes disorderly when modelling violent waves even they are regularly distributed at the start of simulation. Therefore, it is obviously advantageous to eliminate use of direct numerical approximation to second derivatives when solving the pressure Poisson equation in the ISPH formulation.

The distinct feature of this paper, compared with other papers on ISPH lies in that the pressure Poisson equation is first transformed into another equation based on a Rankine source solution using the same idea employed in Meshless Local Petrov-Galerkin Method based on Rankine Source Solution (MLPG_R) [24-30]. In the new formulation of ISPH, the governing equation for pressure does not include any derivatives of the functions to be solved and so overcomes the problems associated with direct numerical approximation to second derivatives in existing ISPH formulation. This new formulated ISPH is named as ISPH_R for convenience in this paper. According to our benchmark tests presented in this paper below, the ISPH_R method can give more accurate results and consume less computational time when modelling water waves.

2. Governing equations and numerical schemes

2.1. Traditional SPH method

The formulation of the traditional SPH can be found in many publications but it will be outlined in this section for completeness. The method is generally based on the Lagrangian form of continuity equation and the Navier-Stokes equation for compressible flow, which may be written as

$$\frac{\mathrm{D}\rho}{\mathrm{D}t} + \rho\nabla\cdot\mathbf{u} = 0 \tag{1}$$

$$\frac{\mathrm{D}\boldsymbol{u}}{\mathrm{D}t} = -\frac{1}{\rho}\nabla p + \boldsymbol{g} + \nu\nabla^2\boldsymbol{u} \tag{2}$$

where ρ is the fluid density, $\mathbf{u}$ is the fluid velocity, t is the time, p is the fluid pressure, $\mathbf{g}$ is the gravitational acceleration, and ν is the kinematic viscosity. In WCSPH, the pressure and density are usually related by the following equation of state for sound waves

$$p = \frac{c_0^2\rho_0}{\kappa}[(\frac{\rho}{\rho_0})^\kappa - 1] \qquad \text{or} \qquad p = c_0^2(\rho - \rho_0) \tag{3}$$

where $\kappa = 7$ is used for liquid water simulation, c_0 is the artificial sound speed and usually chosen as 10 times of maximum fluid velocity, ρ_0 is the initial density of water. In the SPH formulation (see for example, Monaghan, 2005 [23]), the pressure gradient and velocity divergence may be estimated, respectively, by

$$\nabla p_i \approx \rho_i \sum_{j=1}^{N} m_j (\frac{p_j}{\rho_j^2} + \frac{p_i}{\rho_i^2}) \nabla_i W(\boldsymbol{r}_{ij}) \tag{4a}$$

$$\nabla \cdot \boldsymbol{u}_i \approx -\frac{1}{\rho_i} \sum_{i=1}^{N} m_j \boldsymbol{u}_{ij} \cdot \nabla_i W(\boldsymbol{r}_{ij}) \tag{4b}$$

where $\mathbf{u}_{ij} = \mathbf{u}_i - \mathbf{u}_j$, m_j is the mass bearded by particle j and W is a kernel function. There are many forms of the kernel function. The one used in this paper is given by

$$W(\boldsymbol{r}_{ij}) = \alpha_d \begin{cases} \frac{2}{3} - (\frac{|\boldsymbol{r}_i - \boldsymbol{r}_j|}{h})^2 + \frac{1}{2}(\frac{|\boldsymbol{r}_i - \boldsymbol{r}_j|}{h})^3 & 0 \le \frac{|\boldsymbol{r}_i - \boldsymbol{r}_j|}{h} < 1 \\ \frac{1}{6}(2 - \frac{|\boldsymbol{r}_i - \boldsymbol{r}_j|}{h})^3 & 1 \le \frac{|\boldsymbol{r}_i - \boldsymbol{r}_j|}{h} < 2 \\ 0 & \frac{|\boldsymbol{r}_i - \boldsymbol{r}_j|}{h} > 2 \end{cases} \tag{5}$$

where h is equal to $1.8s$ (s is the initial distance between particles) in this paper and α_d is taken as 15/(7*π*h^2) in 2D cases. Using Eq. (4), the velocity and density of each particle may be updated by the following equations:

$$\frac{D\boldsymbol{u}_i}{Dt} \approx -\sum_{j=1}^{N} m_j (\frac{p_i}{\rho_i^2} + \frac{p_j}{\rho_j^2} + \Pi_{ij}) \nabla W(\boldsymbol{r}_{ij}) + \boldsymbol{g_i} \tag{6}$$

$$\frac{D\rho_i}{Dt} \approx \rho_i \sum_{j=1}^{N} \frac{m_j}{\rho_j} \boldsymbol{u}_{ij} \nabla W(\boldsymbol{r}_{ij}) \tag{7}$$

$$\Pi_{ij} = \begin{cases} \frac{-\omega c_0 \tilde{\mu}_{ij} + \xi \tilde{\mu}_{ij}^2}{\bar{\rho}_{ij}} & \mathbf{u}_{ij} \cdot \mathbf{r}_{ij} < 0 \\ 0 & \mathbf{u}_{ij} \cdot \mathbf{r}_{ij} \ge 0 \end{cases} \tag{8}$$

where the artificial viscosity is given by $\tilde{\mu}_{ij} = \frac{h\mathbf{u}_{ij} \cdot \mathbf{r}_{ij}}{\mathbf{r}_{ij}^2 + 0.01h^2}$, $\mathbf{r}_{ij} = \mathbf{r}_i - \mathbf{r}_j$, $\bar{\rho}_{ij} = (\rho_i + \rho_j)/2$. The value of ω and ξ in Eq. (8) is constant and is problem-dependent, which are taken as $\omega = 0.3$ and $\xi = 0$ in this paper. It is noted that the term defined in Eq. (8) is not the exact physical viscous term, instead, it is artificial in some sense but can lead to more stable results than the viscous force directly estimated by derivatives of velocity. Other artificial viscosity models are available [31] but will not be discussed here as they are not focus of this paper. Although the positions of particles may be found by directly integrating velocity, Monaghan [6] suggested the following equation for calculating positions

$$\frac{D\boldsymbol{x_i}}{Dt} \approx \boldsymbol{u}_i - \varepsilon \sum_{j=1}^{N} \frac{m_j}{\rho_j} \boldsymbol{u}_{ij} W(\boldsymbol{r}_{ij}) \tag{9}$$

where ε is a constant in the range of $0 \le \varepsilon \le 1$. In this paper, $\varepsilon = 0.01$.

2.2. ISPH method

In the incompressible SPH method, the fluid density is considered as a constant, and as a result, the continuity equation can be written as

$$D\rho / Dt = 0 \tag{10a}$$

or

$$\nabla \cdot \mathbf{u} = 0 \tag{10b}$$

The momentum equation remains the same as in Eq. (2). The computation in the ISPH method is composed of two basic steps, following the procedure in [41]. The first step is a prediction, in which the velocity field is computed without imposing incompressibility. The second step is a correction in which incompressibility is enforced, leading to the Poisson equation for solving pressure. More details can be found in, e.g., Shao et al. [17]. Summary will be given below.

(a) Prediction step

Assuming that velocities and positions of particles at time t have been found, their velocities and positions at $t + \Delta t$ are first predicted by considering gravitational term and viscous term in Eq. (2) using the following equations,

$$\boldsymbol{u}_* = \boldsymbol{u}_t + \Delta \boldsymbol{u}_* \tag{11}$$

$$\Delta \boldsymbol{u}_* = (\boldsymbol{g} + \nu \nabla^2 \boldsymbol{u}) \Delta t \tag{12}$$

$$\boldsymbol{r}_* = \boldsymbol{r}_t + \boldsymbol{u}_* \Delta t \tag{13}$$

where $\boldsymbol{u}_t$ and $\boldsymbol{r}_t$ are the velocities and positions at time t, respectively; Δt is the time step; $\boldsymbol{r}_*$ and $\Delta \boldsymbol{u}_*$ are the predicted intermediate position and velocity of particles at the new time step.

(b) Correction step

The velocity change during the correction step is estimated by

$$\boldsymbol{u}_{**} = -\frac{\Delta t}{\rho} \nabla p_{t+\Delta t} \tag{14}$$

where $p_{t+\Delta t}$ is the pressure at $t + \Delta t$. The velocities and positions of particles at $t + \Delta t$ are then given by

$$\boldsymbol{u}_{t+\Delta t} = \boldsymbol{u}_* + \boldsymbol{u}_{**} \tag{15}$$

$$\boldsymbol{r}_{t+\Delta t} = \boldsymbol{r}_t + \frac{\boldsymbol{u}_t + \boldsymbol{u}_{t+\Delta t}}{2} \Delta t \tag{16}$$

Combining Eq. (10b) with Eq.(14), one obtains the following equation for pressure

$$\nabla^2 p_{t+\Delta t} = \frac{\rho \nabla \cdot \boldsymbol{u}_*}{\Delta t} \tag{17}$$

Similarly, Shao and Lo [16] proposed a projection-based incompressible method to impose density invariance Eq. (10a), which leading to the equation below

$$\nabla\cdot\left(\frac{1}{\rho^*}\nabla p_{t+\Delta t}\right)=\frac{\rho-\rho^*}{\rho\Delta t^2} \tag{18}$$

where ρ^* is the density at the intermediate time step and can be estimated by $\rho^*=\sum_{j=1,}^{N}m_jW_{ij}$. For the incompressible fluids, the intermediate density is not much different from the specified fluid density. As indicated by Hu and Adams [19], Eq. (17) and (18) are equivalent and both valid for incompressible fluids theoretically. They suggested solving the two incompressibility equations simultaneously. The solution of the density invariant equation (Eq. 18) was used to adjust the positions of particles while the solution of the velocity-divergence-free equation (Eq. 17) was used to adjust their velocity. In contrast, Zhang et al [43] proposed using the mixed one given below

$$\nabla^2 p_{t+\Delta t}=\gamma\frac{\rho-\rho^*}{\Delta t^2}+(1-\gamma)\frac{\rho\nabla\cdot\boldsymbol{u}_*}{\Delta t} \tag{19}$$

which was also used by Ma et al. [26] for the MLPG_R method, where γ is the artificial value and in the range of 0- 1. According to numerical tests presented in Ma and Zhou [26] and also suggested by Zhang et al [43], the results for violent water waves obtained by using Eq. (19) seems to be better if γ is specified a proper small value than those for $\gamma=0$(velocity-divergence-free equation). In order to achieve good results without need of the density term (i.e. $\gamma=0$), the position of particles may be shifted at each time step in a way similar to remeshing or dynamic regridding, such as that based on the Fick's law employed by Lind et al [21], or that according to the velocity calculated from pressure gradients relative to a minimum pressure proposed by Sriram and Ma [28]. These techniques had been found to make the distribution of particles significantly more regular and lead to much better results compared with these from using either Eq. (17) or Eq. (18) separately. Xu et al [44] compared four methods (two of them are based on Eq. (17) and Eq. (18), respectively; the third one is the approach of Hu and Adams [19] and the fourth one is based on the approach of shifting the positions of particles they proposed). They concluded that the approach of using only Eq. (17) may exhibit instability in some cases, the one with Eq. (18) may overcome the instability caused by ill-distributed particles but was shown giving inaccurate predictions with extremely high noise in results, and the particle shifting technique can help achieving accurate and stable simulations as the approach of Hu and Adams [19] does but needs much less computational time than the latter. Such comparative studies are not the focus of this paper, though the studies may be necessary to judge if the approach based on Eq. (19) and the one proposed by [28] could be as good as the particle shifting technique and the approach of Hu and Adams [19] in the same cases. Based on our experience in modelling violent water waves, we choose in this paper to use Eq. (19) which does not need extra computation related to shifting particles but we accept the possibility that combining the particle shifting technique with the new technique for solving the pressure Poisson introduced in the paper may give better results than these presented in the current paper. For solving the pressure Poisson equation (either Eq. 17, Eq. 18 or Eq. 19), several different discretised schemes of the Laplacian operator have been suggested previously but the often employed scheme (e.g., [4],[16] and [22]) is the one given below:

$$\nabla^2 p_i\approx\sum_{j=1}^{N}2\frac{m_j}{\rho_j}\frac{\left(\boldsymbol{r}_{ij}\right)_\alpha W(\boldsymbol{r}_{ij})_{,\alpha}}{r_{ij}^2+\delta^2}p_{ij} \tag{20a}$$

where $p_{ij}=(p_i-p_j)$, subscript α denotes the direction and corresponds to x and y in two-dimensional (2D) cases, $W(\boldsymbol{r}_{ij})_{,\alpha}$ is the partial derivative of the kernel function with respect to the coordinate in α direction, $\left(\boldsymbol{r}_{ij}\right)_\alpha$ is the component of the vector $\boldsymbol{r}_{ij}$ in α direction, $\left(\boldsymbol{r}_{ij}\right)_\alpha W(\boldsymbol{r}_{ij})_{,\alpha}$ is $\boldsymbol{r}_{ij}\cdot\nabla W(\boldsymbol{r}_{ij})$. These explanations are also applied to similar expressions below.

It has been well known that the discretised Laplace operator given by Eq. (20a) may produce large errors in particular when the particles are distributed in a disorderly manner as happen in violent water wave problems. To improve the accuracy of the discretised Laplace operator, Schwaiger [39] proposed several forms of a higher order method for Laplace operator discretisation. Two of them are based on the following equation

$$\nabla^2 p_i \approx \frac{\boldsymbol{\Gamma}^{-1}_{\beta\beta}}{n}\left\{2\sum_{j=1}^{N}\frac{m_j}{\rho_j}(p_j - p_i)\frac{(\boldsymbol{r}_{ij})_\alpha W(\boldsymbol{r}_{ij})_{,\alpha}}{|r_{ij}|^2} - 2p_{i,\alpha}\cdot(\sum_{j=1}^{N}\frac{m_j}{\rho_j}W(\boldsymbol{r}_{ij})_{,\alpha})\right\} \tag{20b}$$

where $\boldsymbol{\Gamma}$ is a tensor that is defined as $\Gamma_{\beta\chi} = \int_{\Omega_i}\frac{m_j}{\rho_j}\frac{(\boldsymbol{r}_{ij})_\alpha W_{,\alpha}}{|\boldsymbol{r}_{ij}|^2}(\boldsymbol{r}_{ij})_\beta(\boldsymbol{r}_{ij})_\chi$, n is the number of dimension (n=2 for 2D cases), α and β denote different directions and correspond to x and y in 2D cases. The difference between the two forms of the high order Laplace operator discretisation lies in how to estimate the gradient in the second term of the right hand of Eq. (20b). In one form, the gradient in the term is estimated by

$$p_{i,\alpha} \approx \sum_{j=1}^{N}\frac{m_j}{\rho_j}(p_j - p_i)W_{,\alpha} \tag{20c),}$$

while in the other form, the gradient is estimated by

$$p_{i,\alpha} \approx \sum_{j=1}^{N}\frac{m_j}{\rho_j}(p_j - p_i)L_{\alpha\beta}W_{,\beta} \tag{20d}$$

where $L_{\alpha\beta} = \left(\sum_{j=1}^{N}\frac{m_j}{\rho_j}(x_{j,\alpha} - x_{i,\alpha})W_{,\beta}\right)^{-1}$, which is an inverse matrix.

In Schwaiger [39], the method based on Eq. (20b) and Eq. (20c) was called as SPH2, while the method based on Eq. (20b) and Eq. (20c) is called as CSPH2Γ. They also suggested another method based on solving all the second order derivatives and named it as CSPM. Schwaiger [39] compared the convergence properties of all the forms including the three methods (SPH2, CSPH2Γ and CSPM) by applying them to estimate the second order derivatives on a set of particles distributed in a disorderly manner. They demonstrated that their CSPH2Γ and CSPM methods have similar convergence properties, but the CSPM method is expected to take much more CPU time. Lind et al [21] applied the CSPH2Γ method to study water wave problems, and indicated that it can give better results by combining it with a particle-shifting technique they suggested. Fatehi and Manzari [40] also suggested a high order method for Laplace operator discretisation, which is similar to the CSPM method in the sense that both need to solve all the second order derivatives.

In this paper, we will compare several forms of ISPH methods with our proposed method. The different forms of the methods used are defined as below:

1) SPH: tradional SPH as discussed in Section 2.1
2) ISPH - incompresible SPH based on Eq. (20a)
3) CISPH1-incompresible SPH based on Eq. (20b) and Eq. (20c)
4) CISPH2- incompresible SPH based on Eq. (20b) and Eq. (20d)

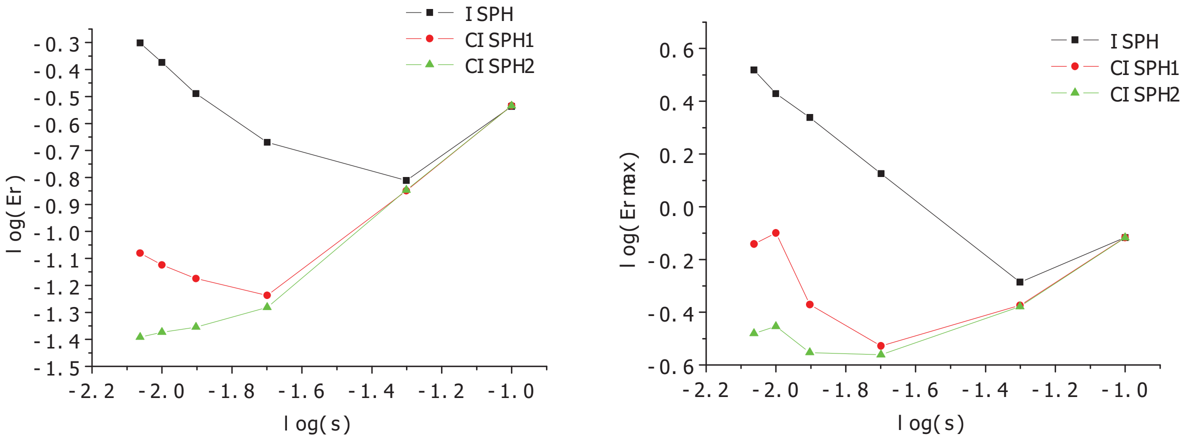
Although Schwaiger [39] and Lind et al [21] had carried out patch tests by applying Eq. (20b) to estimate the second order derivatives of the several functions on a set of particles distributed in a disorderly manner, more tests will be carried out in this section to further show the behaviours of ISPH, CISPH1 and CISPH2, in particular when the disorderliness of particle distribution varies.

For this purpose, we will consider the function of $f(x,y)=\cos(4\pi x+8\pi y)$. The space domain is chosen as a square with the length of sides 1 for $2\leq x\leq 3$ and $2\leq y\leq 3$. The domain is first divided into small squared elements ($\Delta x\times\Delta y$ with $\Delta x=\Delta y=s$). The particles are then redistributed according to $\Delta x'=\Delta y'=s[1+k(Rn-0.5)]$, where Rn is a random number between 0 and 1.0 and k is a constant that can be taken as a value between 0 and 1. Clearly, k=0 leads to regular distribution of particles. k>0 makes the distribution of particles become disorderly. As k increases, the distribution is more disorderly. The Laplacian of $f(x,y)$ is calculated by directly taking mathematical derivatives, which is denoted as $\nabla^2 f_{i,a}$, and estimated numerically by using the three methods mentioned in the previous paragraph, which is denoted as $\nabla^2 f_{i,c}$. The accuracy of the numerical methods is quantified, in a similar way to that used in Schwaiger [39], by evaluating their average relative errors given as

$$Er=\sqrt{\sum_{i=1}^{N}\left(\frac{\nabla^2 f_{i,c}-\nabla^2 f_{i,a}}{\nabla^2 f_{i,a,m}}\right)^2\frac{m_i}{\rho_i}},$$

where $\nabla^2 f_{i,a,m}$ is the magnitude of $\nabla^2 f_{i,a}$, e.g., $\nabla^2 f_{i,a,m}=80\pi^2$, for $f(x,y)=\cos(4\pi x+8\pi y)$. When estimating the error using the above equation, only the particles within the region of $2.2\leq x\leq 2.8$ $2.2\leq y\leq 2.8$ are considered. The accuracy of the methods is also quantified by estimating their maximum relative errors given as

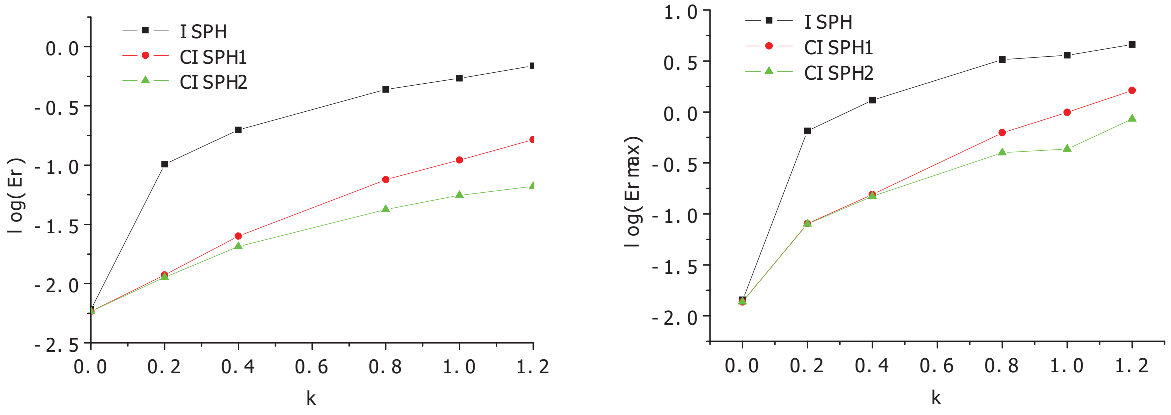
$$Er_{\max}=\max\left(\left|\frac{\nabla^2 f_{i,c}-\nabla^2 f_{i,a}}{\nabla^2 f_{i,a,m}}\right|\right), i=1,2,3\cdots N$$



(a) (b)

Fig. 1 Relative errors of different discretised Laplace operators for different values of s with the value of k fixed to be 0.8 (a) average error; (b) maximum error.

The cases with s=0.1, 0.05, 0.0333, 0.025, 0.02, 0.01667, 0.01429, 0.0125, 0.0111 or 0.01 and k=0, 0.2, 0.4, 0.8, 1.0, 1.2 are considered. The results are presented in Figs. 1-3. Fig. 1(a) presents the average errors for different values of s with a value of k being fixed to be 0.8. From the figure, one can see that the average error of CISPH2 is consistently reduced with reduction of s while it was shown to remain to be constant in Schwaiger [39] for a function of x^2+y^2. The averages errors of other two methods can increase with the reduction of s, which is a divergent behaviour. The behaviour of ISPH is similar to what was shown in Schwaiger [39] but the behaviour of CISPH1 is slightly different as they showed that its average error remained as constant. The results of maximum error given in Fig. 1(b) demonstrate that the error of all three methods can increase with increasing the resolution of the particles. In addition, the smallest value of the error is Log($Er_{\max}$)>-0.6, corresponding to $Er_{\max}$ =25%, which is considerably larger than the value of average error for the same case.



(a) (b)

Fig. 2 Relative errors of different discretised Laplace operators for different values of k with the value of s fixed to be 0.01 (a) average error; (b) maximum error.

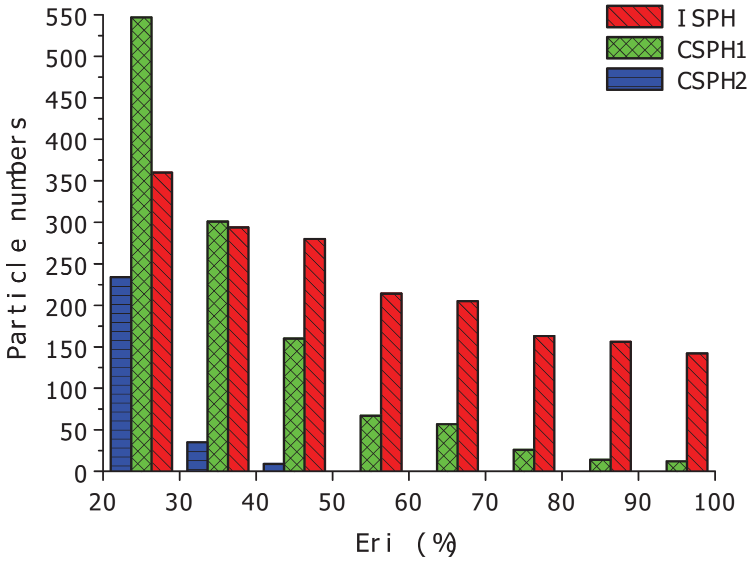


Fig. 3. Number of particles with a large relative error (>20%) for s=0.01 and k=1.2

Fig. 2 plots the average and maximum errors for different values of k with s=0.01. One can see from the figures that the errors of the three methods increase with the increase of k values, i.e., with particle being more disorderly. Furthermore, the maximum error can be become very large, for example, Log($Er_{\max}$)>-0.25, corresponding to $Er_{\max}$ = 56%, at k=1.2. It is noted that overall accuracy of numerical methods are controlled by the maximum error, not the average error, in water wave problems.

In Fig. 3, the horizontal axis shows the different ranges of relative error, e.g., [10%, 20%] and [20%, 30%], while the vertical axis shows the number of particles whose error lies in a range. For example, in the range of [10%, 20%], there are about 550 particles for the CISPH1 method and about 225 particles for the CISPH2 method. The relative error at each individual particle used in this figure is defined as $Er_i = \left|\frac{\nabla^2 f_{i,c} - \nabla^2 f_{i,a}}{\nabla^2 f_{i,a,m}}\right|$ $(i = 1,2,3\cdots N)$. This figure demonstrates that the quite large relative error (>20%) can happen at a considerable number of particles for all the three methods even when they are applied to computing the Laplacian of the quite simple function, though the number for the CISPH1 and CISPH2 method is smaller than for ISPH method.

These results demonstrate that although a lot of effort has been made to develop the approximations to the Laplacian operator discretisation, it still needs to be improved in particular for modelling violent water waves where the particles can become severely disordered without regularisation by shifting even they are uniformly distributed at the start of simulation. The new approach suggested below is to overcome the weakness associated with discretisation of Laplacian operator in the ISPH method.

2.3. ISPH_R method

The main difference between the existing ISPH method and the new method named as ISPH_R method lies in the approach to discretisation of the pressure Poisson equation defined in Eq. (19). The main idea of the new approach comes from another meshless method called as the Meshless Local Petrov-Galerkin Method based on Rankine Source Solution (MLPG_R) [25][26], that is, reformulating Eq. (19) into another form which does not include any derivative of pressure and velocity. For this purpose, Eq. (19) is integrated over a small sub-domain Ω_I (to be distinctive, notation of particles for the ISPH_R method is denoted by capital I or J) surrounding a particle after multiplication by the Rankine source solution φ, and then it reads

$$\int_{\Omega_I} \varphi \nabla^2 p_{t+\Delta t} d\Omega_I = \int_{\Omega_I} [\gamma \frac{\rho - \rho^*}{\Delta t^2} + (1-\gamma)\frac{\rho}{\Delta t}\nabla\cdot \boldsymbol{u}_*]\varphi\, d\Omega_I \tag{21}$$

where φ can be chosen as

$$\varphi = \frac{1}{2\pi}\ln(r/R_I) \quad \text{for 2D cases} \tag{22}$$

that satisfies $\nabla^2\varphi = 0$, in Ω_I except for the center and $\varphi = 0$, on $\partial\Omega_I$, which is the boundary of Ω_I and R_I is its radius. The radius is usually smaller than the distance between two particles. After some mathematical manipulations, Eq.(21) becomes the following form

$$\int_{\partial\Omega_I} \vec{\boldsymbol{n}}\cdot(p_{t+\Delta t}\nabla\varphi)dS - \left(p_{t+\Delta t}\right)_I = \gamma\frac{\rho_I - \rho_I^*}{\Delta t^2}\frac{R_I^2}{4} + (1-\gamma)\int_{\Omega_I}\frac{\rho}{\Delta t}\boldsymbol{u}_*\cdot\nabla\varphi d\Omega \tag{23}$$

which will be applied to each of inner particles. More details of mathematical manipulations can be found in Ma and Zhou [26]. It has been noted that the increment of the density ($\rho - \rho^*$) assumed to a constant within the sub-domain and so equal to its value at Particle *I* when Eq. (23) is derived. This may not cause unacceptable error not only because the density should not change much due to the change in the intermediate position of the particle as pointed above but also because the small error caused due to the assumption is further reduced by multiplying the coefficient γ that is normally chosen in a range of 0~0.3, taken as 0.1 in this paper. The term may be evaluated in the same way as that for the second term but such a way will not improve the accuracy significantly due to the reasons discussed here.

Compared with Eq. (19), the major difference of Eq. (23) is that it does not include any derivative of the functions to be solved while Eq. (19) contains the second order derivative of pressure and the first-order derivative of velocity. Approximation to the functions in Eq. (23) does not require them to have any continuous derivatives, while approximation to the functions in Eq. (19) requires them to have finite, or at least integrable second order derivatives. Therefore, use of Eq. (23) for further discretisation has a great numerical advantage over use of Eq. (19) directly, and so potentially makes the ISPH_R more accurate for the same number of particles or require less number of particles to obtain the solution with the same order of accuracy than ISPH, which will be demonstrated in the later sections of this paper.

2.4. Boundary conditions

Generally, there are two kinds of boundary conditions for water wave problems. One is on solitary boundaries and one on the free surface. They are outlined separately below.

2.4.1. Solid boundary conditions

On solid boundaries, the following conditions (e.g. Ma and Zhou [26] or Sriram and Ma [28]) should be satisfied

$$\boldsymbol{u}\cdot\boldsymbol{n}=\boldsymbol{U}\cdot\boldsymbol{n} \tag{24}$$

and

$$\boldsymbol{n}\cdot\nabla p=\rho\left(\boldsymbol{n}\cdot\boldsymbol{g}-\boldsymbol{n}\cdot\dot{\boldsymbol{U}}+\nu\,\boldsymbol{n}\cdot\nabla^2\boldsymbol{u}\right) \tag{25}$$

where $\boldsymbol{n}$ is the unit normal vector of the solid boundaries, $\mathbf{g}$ is the vector of gravitational acceleration, $\boldsymbol{U}$ and $\dot{\boldsymbol{U}}$ are the velocity and acceleration of the solid boundaries, respectively. It is noted that the traditional SPH does not need the condition in Eq. (25) as it does not need to solve the boundary value problem for pressure. In the ISPH method, the condition in Eq. (25) is necessary.

It is obvious that one must compute the term $\nabla^2\boldsymbol{u}$ when applying this condition in Eq. (25), which needs to estimate the second order derivative at the rigid boundary. To avoid the computation of the second order derivative in the equation, Ma and Zhou [26] combined Eqs. (11) with (25) and gave an alternative as follows:

$$\boldsymbol{n}\cdot\nabla p=\frac{\rho}{\Delta t}\boldsymbol{n}\cdot(\boldsymbol{u}^*-\boldsymbol{U}) \tag{26}$$

This one is used in this paper.

Numerical implementation of Eq. (24) is relatively straightforward, i.e., the normal velocity of fluid particles is imposed to be equal to the normal velocity of wall that is given in the cases of this paper. Numerical implementation of Eq. (26) is different for different SPH methods described above. For the traditional SPH, there is no need to solve the Poisson's equation (Eq. (19)) for pressure and so it is not necessary to use this equation. However, in order to improve the computation of pressure gradients in Eq. (4a) for updating the velocity, the two-layer ghost particles are adopted as described, such as in [22]. For ISPH and ISPH_R methods, Eq. (26) is discretised at the points on solid boundaries, in which the normal gradient of pressure is estimated at the boundaries without the assistance of any ghost particles by using the following equations

$$\vec{\boldsymbol{n}}\cdot\nabla p\Big|_{\vec{r}_I}\approx\sum_{J=1}^{N}\Psi_{IJ}\,p\left(\vec{r}_J\right) \tag{26b}$$

where Ψ_{IJ} is defined by Eq. (A6) in Appendix A. It is noted that Eq. (26b) is based on the approximation to the gradient of a function as detailed in Ma (2007) and adopted by Ma and Zhu (2009). However, the form used in this paper is slightly different from that given in Ma (2007), in the sense that the expression of Ψ_{IJ} is directly given here.

2.4.2. Free surface condition and Free surface particle identification

The condition on the free surface is very simple, which is stated that the pressure of water on its free surface is equal to the atmospheric pressure which can be taken as zero, i.e.,

$$p = 0 \tag{27}$$

In the traditional SPH method, this condition is automatically satisfied as long as the density on the free surface is estimated correctly, as one would see in Eq. (3). However, in the ISPH method, this condition has to be imposed when solving the boundary value problem defined by Eqs. (17), (18), (19) or (23). In order to impose this condition, one needs to know which particles are on the free surface. This is not a problem for non-broken water waves, where the water particles on the free surface at start always remain on the free surface and does not need to be identified during simulation. However, for breaking or violent water waves, the particles on the free surface at start can become inner particles and inner particles can become the free surface particles during a simulation. Therefore, the free surface particles have to be identified at every time step after wave breaking occurs. Many publications (e.g., Shao et al. [18]) on the ISPH method use the ratio of $\alpha_I = \frac{\rho_I}{\rho_0}$, where ρ_I is estimated by $\rho_I \approx \sum_{J=1,}^{N} m_J W_{IJ}$, to identify the free surface particles but often shows wrong identification. A number of researchers have tried to address the inaccuracy of the approach based only on $\alpha_I = \frac{\rho_I}{\rho_0}$, for example, Lee at el [22] employed the gradient of position vector, $\nabla \cdot \boldsymbol{r}$, to judge if a particle is on the free surface ($|\nabla \cdot \boldsymbol{r}| < 1.5$) or not. However as indicated by Lind et al [21], some internal fluid particles may satisfy$|\nabla \cdot \boldsymbol{r}| < 1.5$, while some free-surface particles may $|\nabla \cdot \boldsymbol{r}| > 1.5$, leading to misjudgement. Another method has been recently proposed for the SPH method by Zheng et al. [32]. For completeness, this method is summarised here and more details can be found in Zheng et al. [32]. The main idea of the method is to define three auxiliary functions on the influence domain around a particle, as shown in Fig. 4. The influence domain is determined by the kernel function, i.e., the kernel function at particle *I* is zero outside the influence domain. The influence domain is divided into four parts in two different ways, as seen in Fig. 4. One way is that it is divided into four quadrants by x- and y- axes and the other is divided into four shaded areas. The angle of each shaded part covers an area of 90 degrees, and is symmetrical to the x or the y axis, respectively. The first auxiliary function is defined as

$$fsp\_a(I) = \begin{cases} 1 & NumA \geq 1 \\ 0 & NumA = 0 \end{cases} \tag{28}$$

where $NumA$ represents the number of free surface particles existing in the influence domain of Particle *I* in previous time step. The value of *NumA* is equal to or larger than 1 if there is any neighbour free surface particle in the influence domain in Fig. 4; otherwise it is zero. The second auxiliary function is

$$fsp\_b(I) = \begin{cases} 1 & NumB = 4 \\ 0 & NumB \leq 3 \end{cases} \tag{29}$$

where $NumB$ represents the number of quadrants of the influence domain occupied by the particles in a local coordinate system centred at Particle *I*. $NumB = 4$ if all the 4 parts in Fig.4a have neighbour particles and $NumB = 3$ means that only three of four parts in Fig.4a have neighbour particles. The third auxiliary function is given by

$$fsp\_c(I) = \begin{cases} 1 & NumC = 4 \\ 0 & NumC \leq 3 \end{cases} \tag{30}$$

where $NumC$ represents the number of shaded parts of the influence domain containing at least one fluid particle. If all the 4 shaded parts in Fig.4b have neighbour particles, then *NumC*=4. Similarly, if 3 shaded parts in Fig.4b have neighbour particles, *NumC*=3. At each time step, each of potential free surface particles is checked in the following sequence:

(a) No inner particle in the influence domain except for Particle *I*
(b) $\alpha_I \leq 0.90$ and $fsp\_a(I) = 1$
(c) $\alpha_I > 0.90$, $fsp\_a(I) = 1$ and $fsp\_b(I) = 0$
(d) $\alpha_I > 0.90$, $fsp\_a(I) = 1$ and $fsp\_c(I) = 0$
(e) $\alpha_I \leq 0.90, NumB \leq 2$ and $NumC \leq 2$

If any of expressions is true during checking, Particle *I* is identified as a free surface particle. The group of potential free surface particles is selected according to the free surface particles and their neighbours in previous two time steps. In the sense that only the potential free surface particles, not all, are checked, the aspect of the method is similar to what was suggested by Marrone et al [41].

As tested by Zheng et al. [32], this technique can give significant improvement on identifying the particles on the free surface, as shown in Fig. 5. The left one shows the configuration of particles at a time instant with the free surface particles (blue dots) identified by only using the ratio ($\alpha_I = 0.9$) of the density while the right one shows that with the free surface particles (blue dots) identified by using the above method. It can be seen that many inner particles are identified as the free surface particles by the method based only on the density ratio while the method described above can correctly identify almost of them on and near the free surface. It is noted nevertheless that a few particles near the free surface may still be identified as free surface particles but such incorrect identification may not lead to significant error on pressure. That is because the pressures of these particles are very close to the pressure on the free surface. More details of verification and comparison is referred to Zheng et al. [32].

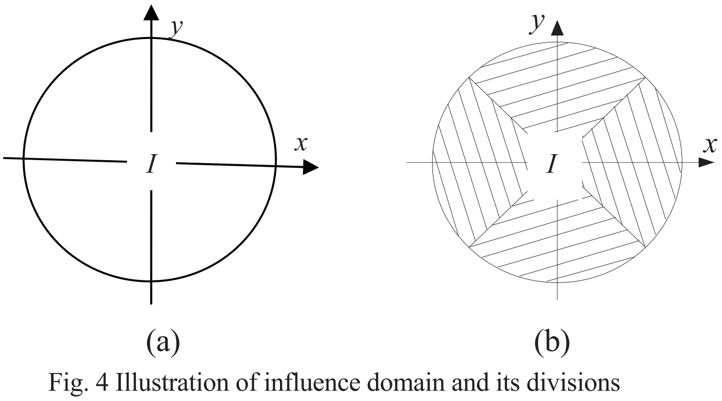
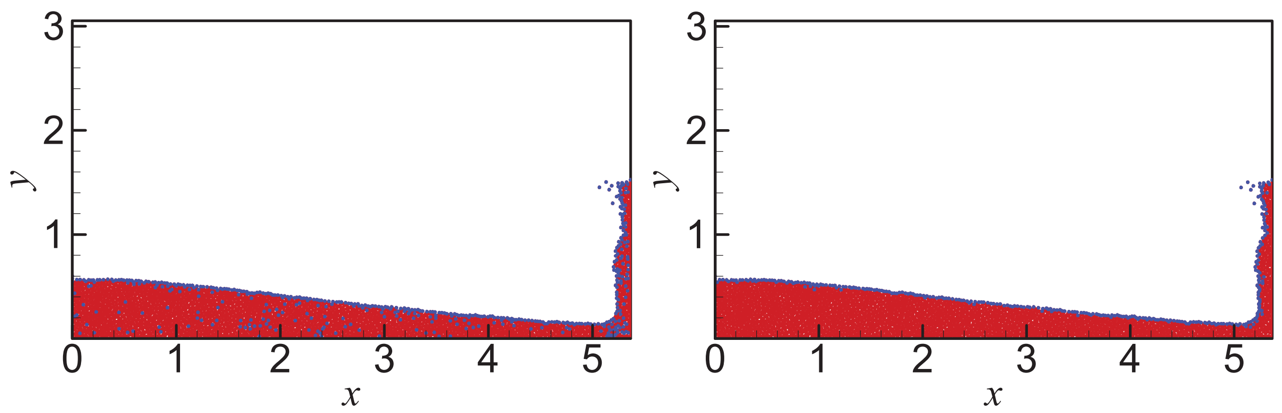


Fig. 4 Illustration of influence domain and its divisions



(a)based on density ratio only

(b) Method of Zheng et al. [32]

Fig. 5 Comparison of free surface particle identification methods
(blue dotes: free surface particles; red dots: inner particle)

2.5. Discretisation of pressure governing equation for ISPH and ISPH_R methods

As has been indicated above, the ISPH and ISPH_R methods require solving the boundary value problem for pressure defined by Eqs. (19) or (23), (26) and (27) at each time step. For this purpose, they must be discretised by using a set of particles to form the following algebraic system.

$$\boldsymbol{A}\cdot\boldsymbol{P}=\boldsymbol{B} \tag{31}$$

where $\boldsymbol{P}$ is a column vector formed by the pressure, $\boldsymbol{A}$ and $\boldsymbol{B}$ are the matrices. The specific expressions of entries in $\boldsymbol{A}$ and $\boldsymbol{B}$ depend on the scheme for discretising the governing equations. For the ISPH method, if one uses Eq. (20a) to discretise the Laplace operator at inner particles and Eq. (26b) to discretise the boundary condition on the solid boundaries, the expressions of entries in $\boldsymbol{A}$ and $\boldsymbol{B}$ are given by

$$A_{IJ}=\begin{cases}\displaystyle\sum_{J=1}^{N}2\frac{m_J}{\rho_J}\frac{r_{IJ}^{\alpha}W_{,\alpha}}{r_{IJ}^2+\delta^2} & \text{for inner particles}\\ \Psi_{IJ} & \text{for particles on solid boundaries}\end{cases} \tag{32a}$$

$$B_I=\begin{cases}\left[\alpha\dfrac{\rho-\rho^*}{\Delta t^2}+(1-\alpha)\dfrac{\rho\nabla\cdot\mathbf{u}_*}{\Delta t}\right]_I & \text{for inner particles}\\ \left[\dfrac{\rho}{\Delta t}\mathbf{n}\cdot(\mathbf{u}^*-\mathbf{U}^{n+1})\right]_I & \text{for solid boundary particles}\end{cases} \tag{32b}$$

Similar expressions can be written out if one employs Eq. (20b) to discretise the Laplace operator at inner particles but details are omitted here.

For the ISPH_R method, the pressure in Eq. (23) is approximated by $p(\vec{x})\approx\sum_{j=1}^{N}\Phi_j(\vec{x})p_j$ at all inner particles, where $\Phi_j(\vec{x})$ is the shape function which may be formulated by using the moving least square method (MLS) as in [24-26], or the interpolation scheme developed by Ma [29]. In this paper, the MLS is used. The details about formulating the shape function can be found in Ma [24], and so will not be given here. With the approximation to pressure and after converting Eq. (23) and (26) into Eq. (31), the entries of $\boldsymbol{A}$ and $\boldsymbol{B}$ are given, respectively, by

$$A_{IJ}=\begin{cases}\int\limits_{\partial\Omega_I}\varPhi_J(\vec{x})\vec{n}\cdot\nabla\varphi ds-\varPhi_I(\vec{x}) & \text{for inner nodes}\\ \varPsi_{IJ} & \text{for solid boundary nodes}\end{cases}\tag{33a}$$

$$B_I=\begin{cases}\alpha\dfrac{\rho_I-\rho_I^{\,*}}{\Delta t^2}\dfrac{R_I^2}{4}+(1-\alpha)\int\limits_{\Omega_I}\dfrac{\rho}{\Delta t}\vec{u}^{\,*}\cdot\nabla\varphi d\varOmega & \text{for inner nodes}\\ \dfrac{\rho}{\Delta t}\vec{\mathbf{n}}\cdot(\vec{\mathbf{u}}_*-\vec{\mathbf{U}}^{n+1}) & \text{for solid boundary nodes}\end{cases}\tag{33b}$$

When forming the above equations, the pressure at the free surface particles has been imposed to be zero, according to Eq. (27). In Eq. (33), one needs to evaluate the integrals at each particle over its sub-domain. This potentially takes significant computational time but the semi-analytical technique suggested by Zhou and Ma [27] helps reducing the costs considerably and is adopted in this paper.

3. Numerical tests, validation and discussions

In this part, the method described above will be tested for the several water wave problems and also for lid-driven flow, and validated by comparing its results with those in literature. Its behaviours will also be compared with the traditional SPH and different forms of existing ISPH methods in the cases for water waves.

3.1. Dam breaking flow

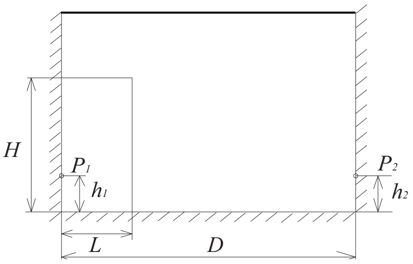


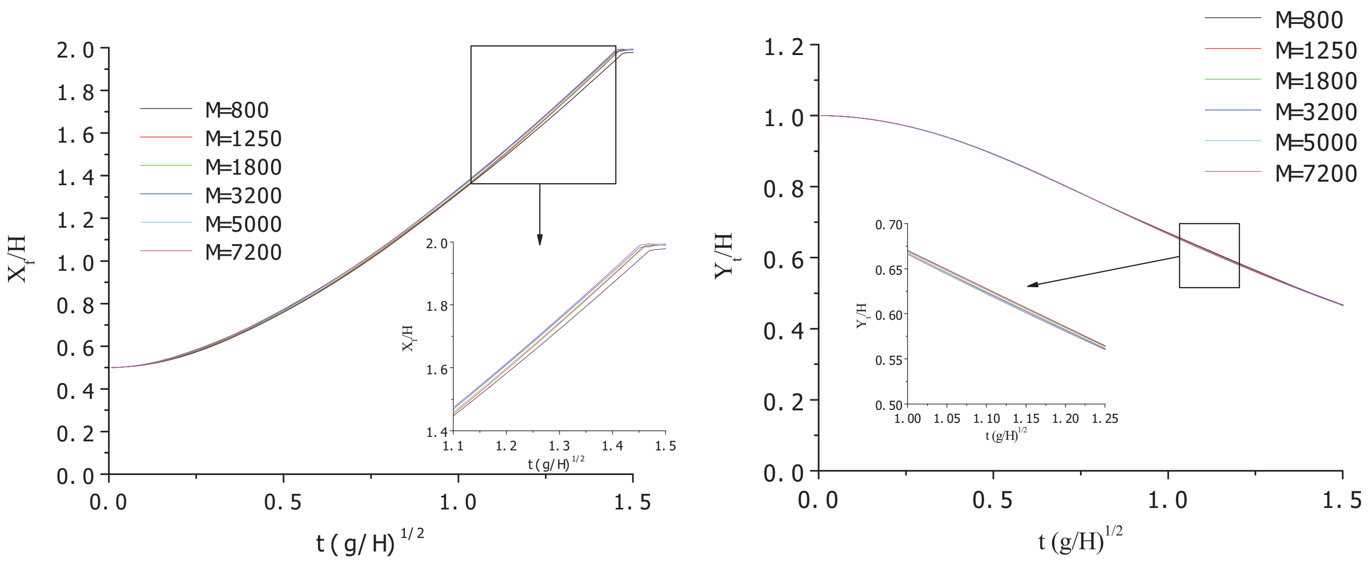
Fig. 6 Sketch of dam breaking flow

Dam breaking is often used as a benchmark for violent free surface flow. In this numerical test, a rectangular water column is confined between bottom, top wall and two vertical walls, as illustrated in Fig. 6. The width of the water column is L and its height is H. At the beginning of the computation, the dam is instantaneously removed and the water collapses and flows out along a dry horizontal bed. D is the distance between two vertical walls. There are two pressure sensors p_1 and p_2 on the left and right vertical walls, respectively. The height of p_1 and p_2 from the bottom is h_1 and h_2 respectively. In this section, all variables and parameters are non-dimensionalised using H and g, such as $\tilde{t}=t\sqrt{g/H}$ unless mentioned otherwise.

3.1.1. Dam breaking flow with non-breaking waves

Although the case of non-breaking waves is not necessarily dealt with by the ISPH_R method that is developed for modelling the violent breaking waves, it is used here for preliminary validation of the method. That is because the error could be estimated precisely against the experimental data in such a case. For this purpose, the parameters in Fig. 6 are taken as L=0.5m, $H/L=2.0$ and $D/L=4.0$, and the number of participles is chosen to be M=800, 1250 1800, 3200, 5000, 7200, corresponding to $s=0.025$, 0.02, 0.0167, 0.0125, 0.01, 0.00833, respectively. The time step length is selected as $\Delta\tilde{t}=0.008$. The time histories of the water front (X_f) and the water column height (Y_t) at the left vertical wall after collapsing of the water column corresponding to different numbers of particles obtained by using the ISPH_R are plotted in Fig. 7. It can be seen that the results are very close to each other, which indicates that 800 particles can lead to sufficiently accurate results in this case. Fig. 8 gives the results of water front and water column height at the left vertical wall obtained by using the ISPH_R method, all three forms of ISPH method and traditional SPH described above and their comparison with published experimental results [33]. According to the results in Fig. 8(a), the water front results obtained by ISPH_R and two SPH methods agree quite well with experimental results, though there are visible differences between the computational and experimental results. However, the results are very close to the results obtained by using MAC [34] and VOF [35] methods. From Fig. 8(b), one may observe that the results of water column height at the left vertical wall from all the methods (ISPH_R, CISPH2 and SPH) do not have much difference with experimental results.

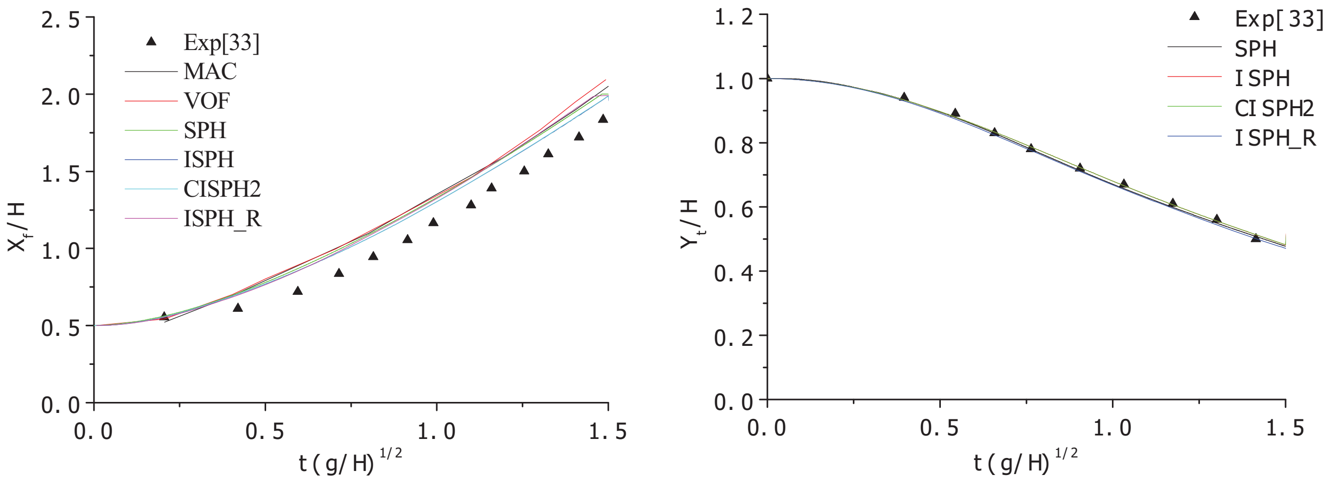
The time histories of pressure recorded at $h_1=0.1H$ corresponding to different particle numbers are presented in Fig. 9(a), which are computed by using the ISPH_R method. It can be seen that when the number of particles is not large enough, such as M=800, there are some spurious fluctuations in the pressure time histories, while they can be suppressed with increase of particle numbers, for example, $M>3200$ in this case. This case indicates that spurious fluctuations in the pressure results computed by the new developed ISPH_R method become insignificant when the particle number is large enough, similar to the features of the existing ISPH method as indicated by Lee et al. [22] where the results of the SPH were shown to have large fluctuations. Fig. 9(b) compares the results corresponding to the different values of h_1 of Point p_1 with $M=7200$, i.e., $h_1=0.05H, 0.1H, 0.15H, 0.20H$ and $0.25H$. One can see from this figure that all the pressure time histories are quite smooth.



(a)

(b)

Fig. 7 Comparison of (a) water fronts represented by X_f and (b) water column heights represented by Y_t corresponding to different particle numbers



(a) (b)

Fig. 8 Comparison of water fronts (a) and water column heights (b) with experimental data [33] when $M = 7200$

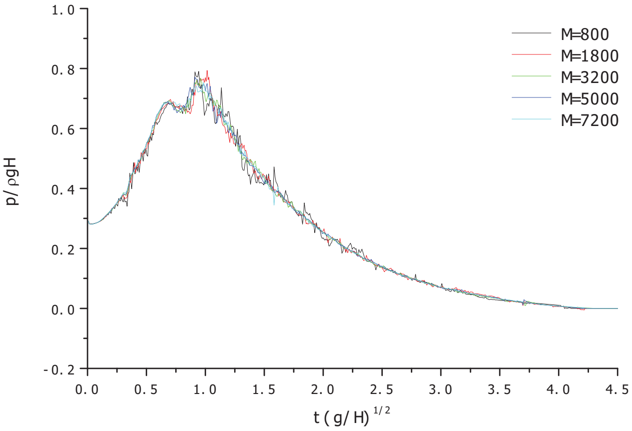


Fig. 9(a) Pressure at h_1=0.1H corresponding to different number of particles

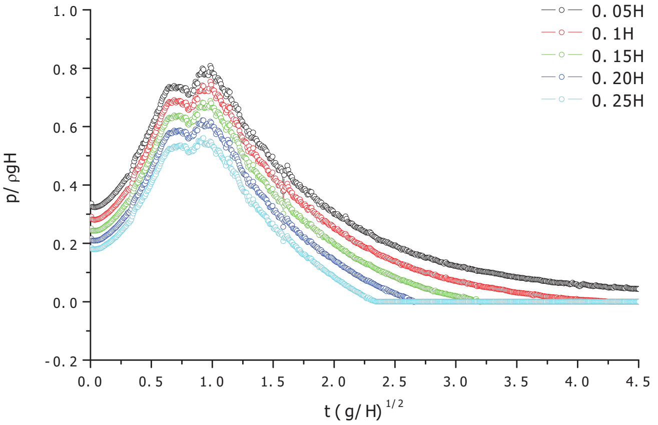
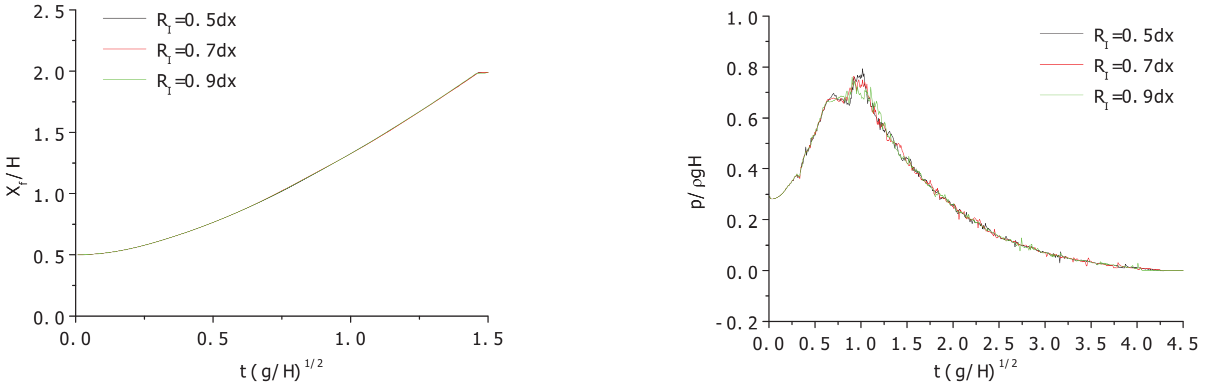


Fig. 9(b) Pressure at different positions using M=7200

Fig. 9 Time histories of pressure computed by using ISPH_R method

Another issue is also addressed by using the above case, which is about the size (R_I) of the integration domain (Ω_I) related to Eq. (23) for the ISPH_R method. It is noted that Eq. (23) is similar to that for MLPG_R method [24-30] as mentioned before. The effects of R_I have been discussed, for example, by Ma et al [25], which showed that R_I could be selected in the range of 0.3 to 0.9 *dx* (*dx* here is the minimum distance of two particles). Three values (0.5 *dx*, 0.7 *dx* and 0.9 *dx*) of R_I are tested for the case presented in Figs. 7-9.

The corresponding results of the wave front and pressure at h_1=0.1H are plotted in Fig. 10. It can be observed that there is no visible difference in Fig. 10(a) while very small one in Fig. 10(b). R_I= 0.5dx are used for the cases in this paper.



(a) Wave front

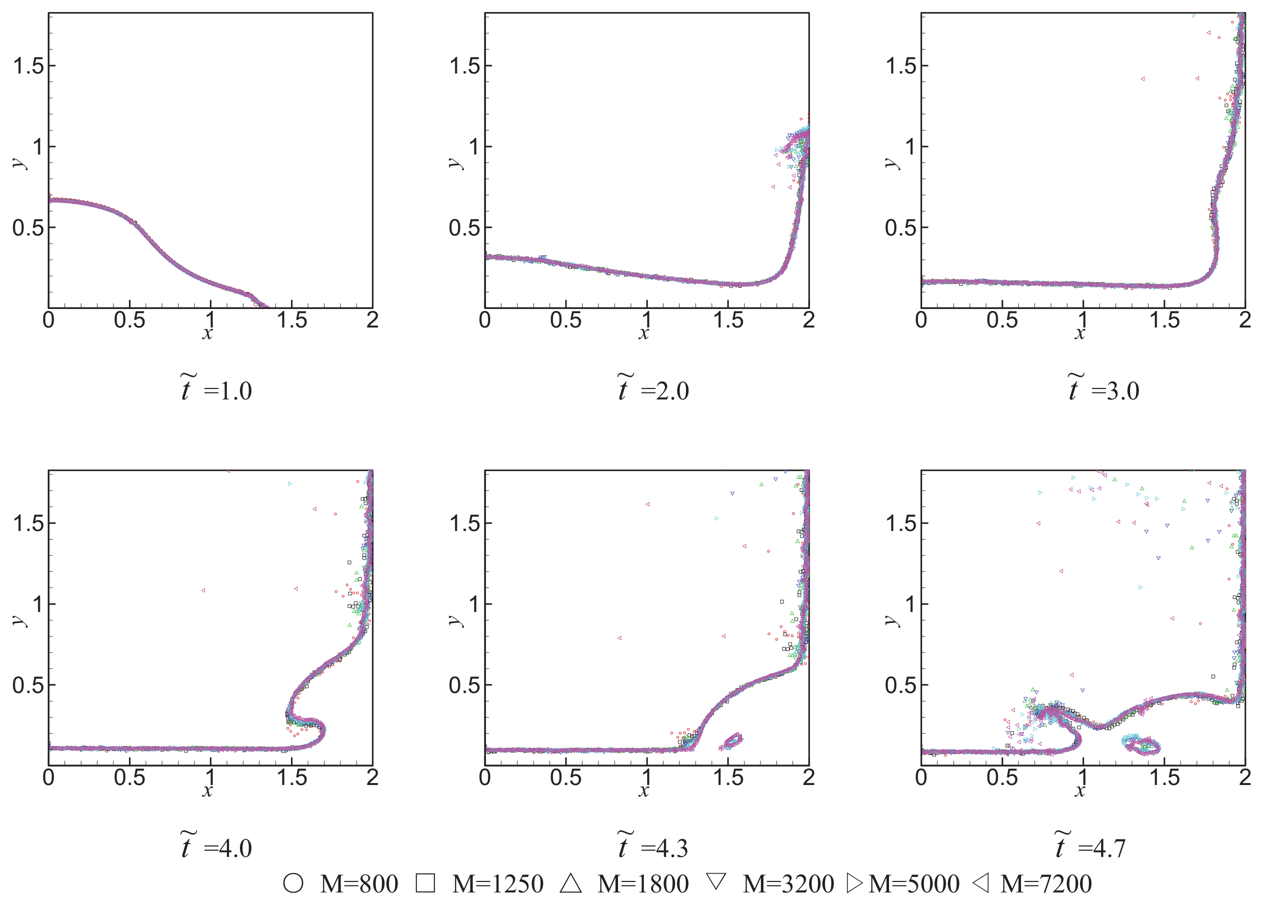
(b)Pressure at h_1=0.1H

Fig. 10 Effects of different sizes (R_I) of the integration domain in ISPH_R method ($M = 7200$)

3.1.2 Dam breaking flow with breaking waves

It is more important to consider the dam breaking flow with breaking waves, which will be investigated in this sub-section. For this purpose, a case with $L = 0.5$ m, $H/L = 2.0$, $D = 4L$ is also considered but the focus is now on the behaviour of breaking waves corresponding to longer simulation. To show how many particles should be used in this case, some results of free surface profiles at different time instants are presented in Fig.11, which are computed by using the ISPH_R method with a time step of $\Delta\tilde{t} = 0.008$. It can be seen from this figure that when particle number is larger than $M = 3200$, corresponding to the non-dimensional initial distance between particles of s=0.0125, the difference between the free surface profiles corresponding to different particle numbers become acceptably small.

In order to study how the numerical results changes with the different lengths of time steps, the ISPH_R method is employed to simulate the same case as in Fig. 11 by using $\Delta\tilde{t}$ =0.012, 0.010, 0.008, 0.006 (corresponding to 0.8, 0.67, 0.53 and 0.4 of the CFL (Courant–Friedrichs–Lewy) number calculated by using the velocity equal to $\sqrt{2gH}$) . The free surface profiles at different time instants are depicted in Fig. 12. Generally speaking, the difference between profiles is insignificant, in particular when the time step length is less than 0.01. From the discussions about Figs. 11 and 12, one would see that with use of a time step length of less than 0.01 and more than 3200 particles, the ISPH_R method can yield convergent results in terms of the free surface profiles. It is noted here that compared with Figs. 17 and 18 of Lind et al [21] where no top rigid wall was at y=2, the graphs in Fig. 11 and 12 show slight more isolated particles. This may be attributed partially to the reflection by the top rigid wall after impact. In addition, the results may also be improved by adopting an adaptive time-step technique and by combining the new technique for solving the pressure Poisson equation with other particle stabilising (e.g., particle shifting) techniques and with other free surface identification methods, other than these employed in this paper.



$\widetilde{t}$ =1.0

$\widetilde{t}$ =2.0

$\widetilde{t}$ =3.0

$\widetilde{t}$ =4.0

$\widetilde{t}$ =4.3

$\widetilde{t}$ =4.7

Fig. 11 Free surface profile comparison corresponding to different particle numbers by using the ISPH_R method

The convergence property is now examined in terms of the pressure, which is more critical. The time histories of pressure at Point p_2 on the right vertical wall with the height of $h_2 = 0.05H$ are presented in Fig.13(a) for the cases with the different particle numbers. The average error is calculated by $Er_p = \sqrt{\sum_{i=1}^{N}(p_i - p_{i,0})^2} \Big/ \sqrt{\sum_{i=1}^{N} {p_{i,0}}^2}$, where $p_{i,0}$ is the pressure for $M = 7200$ at different time step i, which is plotted in Fig. 13(b). The results of Fig. 13 confirm that with the particle number increasing, the results of pressure seem to be convergent. Other methods (traditional SPH, ISPH and CISPH2) are also applied to simulate this case. The results of pressure at Point p_2 from them for $M = 7200$ are compared in Fig. 14. It can be seen that the spurious oscillations in the pressure time histories produced by traditional SPH is very large, the fluctuations of pressure obtained by ISPH and CISPH2 methods much smaller but still visible while the time history from the ISPH_R method is much smoother. In addition, Table 1 compares the CPU times used by different methods to yield the results in Fig. 14. It can be seen that the ISPH_R method does not only provide smoother pressure time history but also spend much less CPU time, which is 86.7% of that used by the ISPH, 80% of that by CISPH2 and 40.2% of that used by the traditional SPH method for this case.

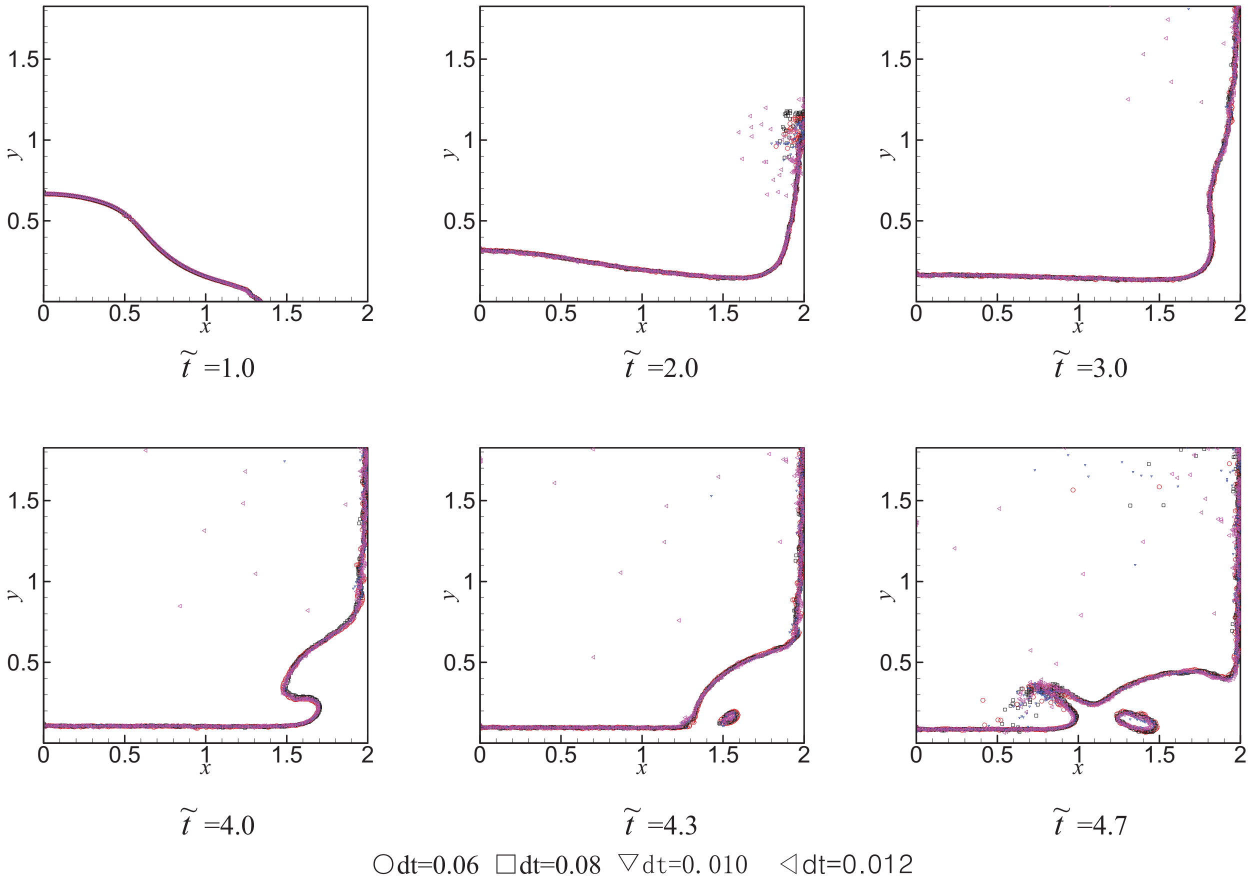
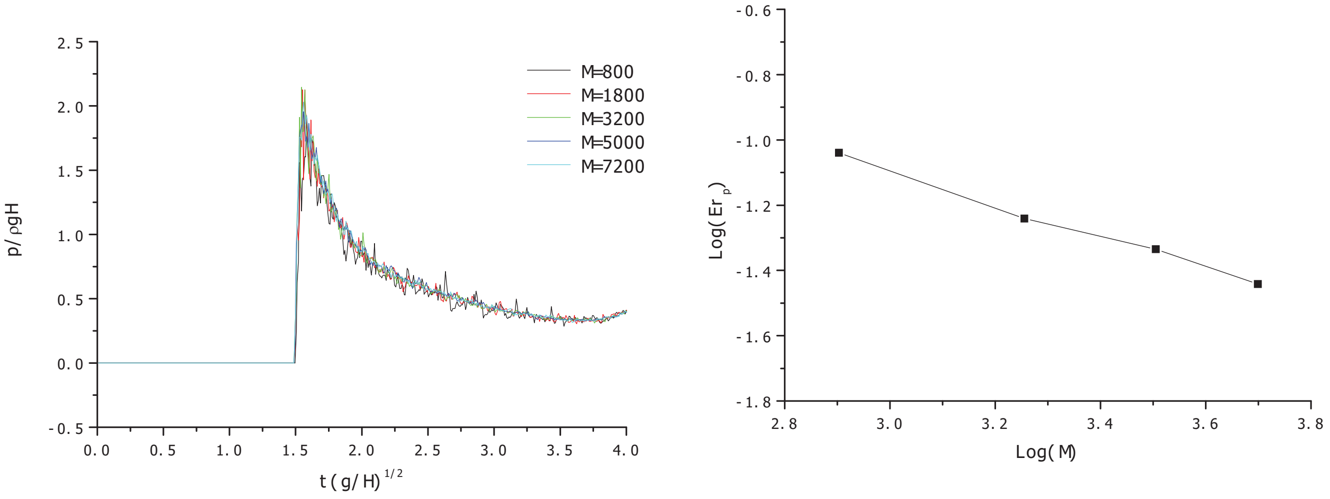


Fig. 12 Free surface profile comparison corresponding to different time step lengths by using the ISPH_R method ($M = 3200$)



(a) (b)

Fig. 13 Convergence test of pressure when $h_2 = 0.05H$ (a) the pressure time histories; (b) the errors, corresponding to different number of particles

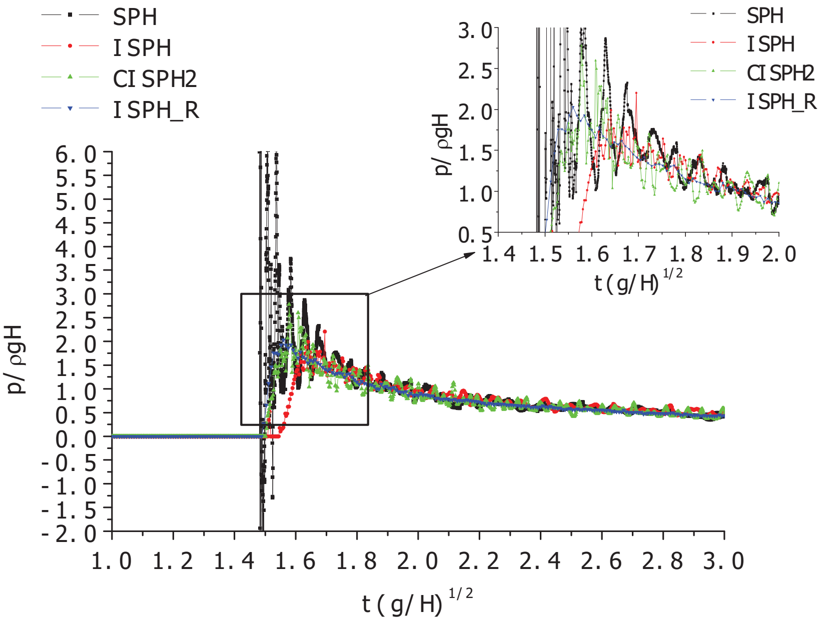
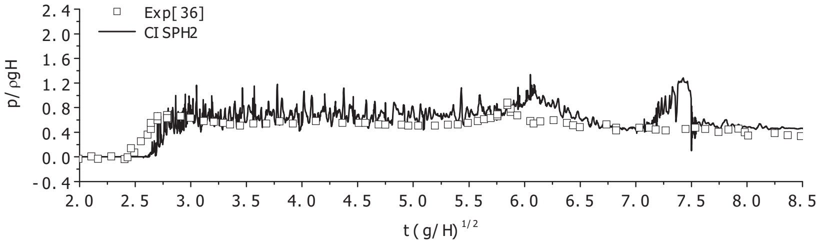


Fig. 14 Comparison of breaking wave impact obtained by different methods

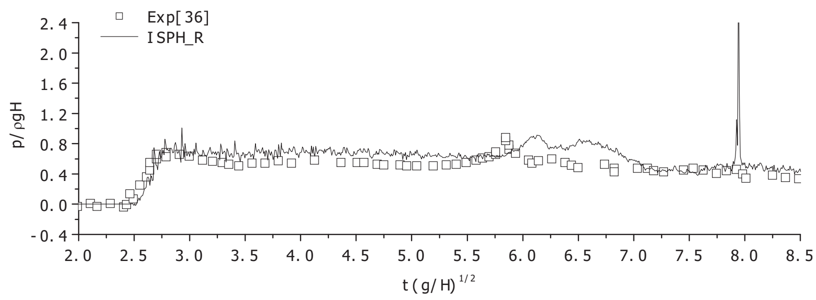
Table 1. CPU time comparison of different methods for dam breaking wave simulation

Scheme	Time step $(\Delta t\sqrt{g/H})$	Stepping number	CPU time(s)
SPH	0.000313	12771	4086
ISPH	0.002	2000	1893
CISPH2	0.002	2000	2046
ISPH_R	0.008	500	1641

In order to make further comparison between the results from the methods, another benchmark case for dam breaking flow is considered. In this case, $L = 2.0\,\mathrm{m}$, $H = 0.5L$, $D = 5.3667H$. For this case, the pressure time histories at p_2 obtained by the CISPH2 and ISPH_R methods with $M = 7200$ is compared with the experimental data [36] in Fig. 15. This shows that the pressure time history from the ISPH_R method is smoother, though both have similar agreement with the experimental data.



(a)



(b)

Fig. 15 Comparison of pressure time histories at p_2 with experimental results [36] and different SPH methods

3.2. *Sloshing waves in a moving tank*

In order to further show the properties of the ISPH_R method, this section gives the results of sloshing waves in a moving tank. The cases for small amplitude sloshing and violent sloshing will be considered to compare the numerical results with analytical solution [37], and with experimental data [38], respectively.

The geometry of the sloshing tank is illustrated in Fig.16. The tank length is L, its height h, and the water depth is d. There is a pressure sensor mounted on the left wall. The height between the sensor and bottom is h_1. The sway displacement of the tank is given by $X_s = a_0(1-\cos\Omega t)$, where a_0 and Ω are the amplitude and the frequency of the motion, respectively. In the simulation, the moving coordinate system fixed to the tank is employed to simplify the implementation of solid boundary conditions as in [42].

3.2.1. Small amplitude sloshing

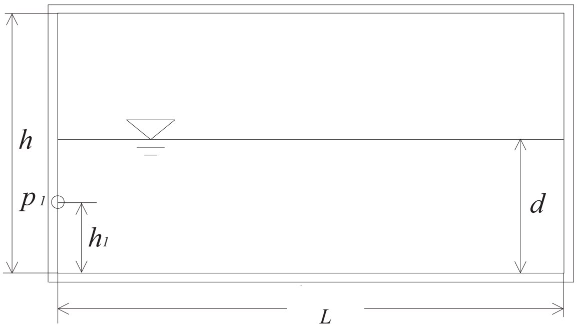


Fig. 16 Sketch of sloshing tank

There are two reasons to simulate this case. One is that when the motion amplitude, i.e. $a_0\Omega/\sqrt{gd}$, of the tank is small, and the frequency is not close to the natural frequency of sloshing waves in the tank, the free surface elevation can be analytically expressed by

$$\eta = -\frac{1}{g}\sum_{n=1}^{\infty}\dot{B}_n(t)\cos k_n(x+\frac{L}{2})+d \tag{34}$$

$$\dot{B}_n = \frac{1-(-1)^n}{k_n^2 L}2\omega_n U_0 \frac{\omega_n\Omega\cos\Omega t-\omega_n\Omega\cos\omega_n t}{\omega_n^2-\Omega^2} \tag{35}$$

where $k_n = n\pi / L$, $\omega_n = \sqrt{gk_n \tanh k_n d}$. More details about the solution can be found in [37]. The second reason for considering this case is that the error of numerical methods can be precisely estimated and so their behaviours can be systematically investigated. For this purpose, $d = 0.5$ m, $L = 2d$, $a_0 = 0.001d$ are selected and different numbers (*M*) of particles are used, that is 288, 450, 800, 1250, 1800, 3200, 5000 and 7200 with the corresponding non-dimensional particle sizes being $s = 0.0417$, 0.033, 0.025, 0.02, 0.0167, 0.0125, 0.01, 0.083, respectively. Fig. 17 gives the comparison of free surface elevations on the left tank wall, which are calculated by the analytical solution in Eq. (34) and by the ISPH_R method for $\Omega / \omega_1 = 0.8$ (only the results corresponding to 288, 800, 3200 and 7200 are plotted for clarity). This figure indicates that the results of the ISPH_R method for this case become very close to the analytical solution when the number of particles is 800 or more. The results of the ISPH_R method are also compared with those from the SPH, ISPH and CISPH2 in Fig. 18, which are also produced by using 800 particles but different time step length. Non-dimensional time stepping length ($\Delta\tilde{t}$) is 0.008 for the ISPH_R, 0.002 for ISPH and CISPH2, and 0.000313 for SPH. From Fig. 18, it can be seen that the results of the SPH has a good agreement with analytical solution at first three periods, but with the simulation going longer, numerical dissipation in the wave amplitude become evident. In addition, in the time range (such as 35 to 40) of short and small waves, the traditional SPH method cannot catch the detail correctly. In contrast, the results from the ISPH, CISPH2 and ISPH_R method do not exhibit such dissipation and can well catch the details of short and small waves. In order to quantitatively show the behaviours of the methods, Fig. 19 presents the errors of numerical results relative to the analytical solution corresponding to different numbers of particles employed. In this figure, the error is computed by using $Er_\eta = \sqrt{\sum_{i=1}^{N_t} |\eta_i - \eta_{i,a}|^2} \Big/ \sqrt{\sum_{i=1}^{N_t} |\eta_{i,a}|^2}$, where $\eta_{i,a}$ is the wave elevation on the left wall of analytical solution at *i*-th time step, η_i is the numerical result at the time instant, N_t is total time steps in the simulation duration of $\tilde{t} = 50.0$. Fig. 19 clearly shows that the error of results from the ISPH_R method is considerably smaller than those of SPH, ISPH and CISPH2 methods. It is a little surprising that the two curves for ISPH and CISPH2 methods are very close to each other in this case. In other words, to achieve any specified accuracy, the ISPH_R method needs much less number of particles (or larger particle sizes) than others. For example, corresponding to $\text{Log}(Er_\eta) = -3.4$, the particle size required by ISPH_R, CISPH2, ISPH and SPH methods are $\text{Log}(s) \approx$ -1.62, -1.85, -1.86 and -1.96, respectively. In order to explore the properties of the methods in another way, Fig. 20 depicts the CPU time spent by all the methods corresponding to different numerical errors on the same computer. One can see from Fig. 20 that the ISPH_R method needs much less CPU time to achieve the same level of accuracy. For example, corresponding to $\text{Log}(Er_\eta) = -3.35$, the CPU times spent by the ISPH_R, CISPH2, ISPH and traditional SPH methods are $\text{Log}(CPU\_t) \approx$ 2.76, 3.24, 3.19, and 3.70, which are CPU_t≈575, 1737, 1548 and 5011 seconds, respectively.

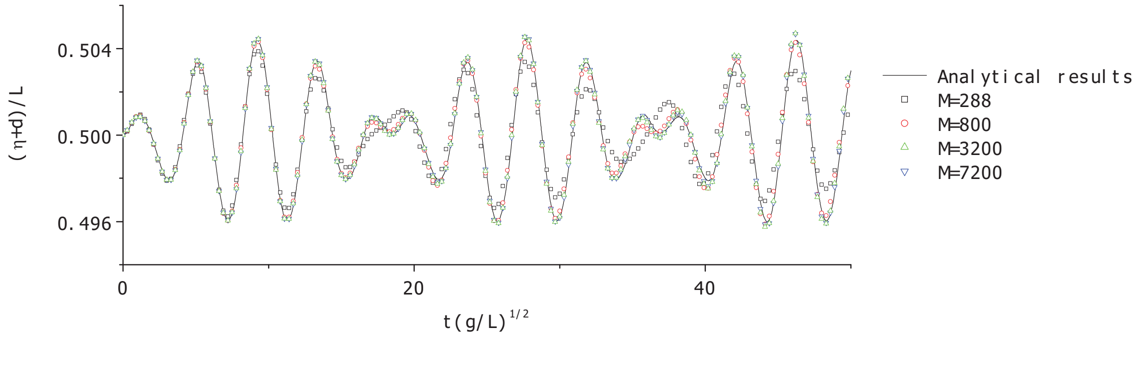


Fig. 17 Wave elevation time histories on the left wall obtained by using different number of particles and the ISPH_R method

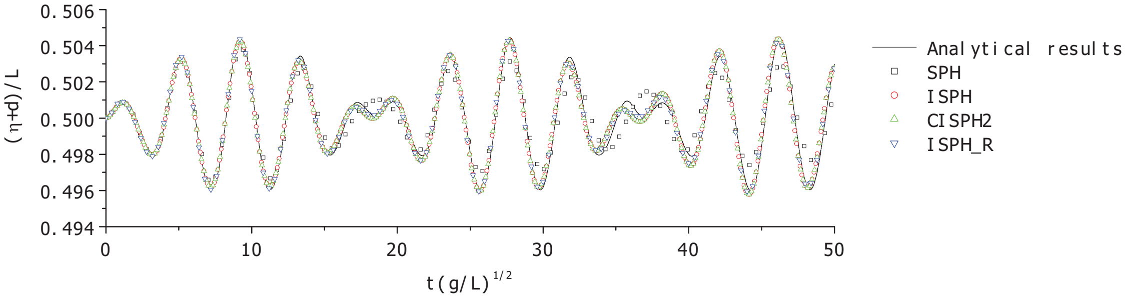
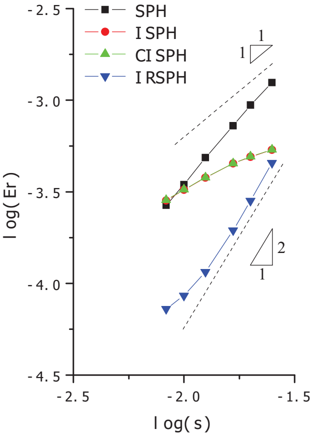


Fig. 18 Wave elevation time histories on the left wall obtained by four methods



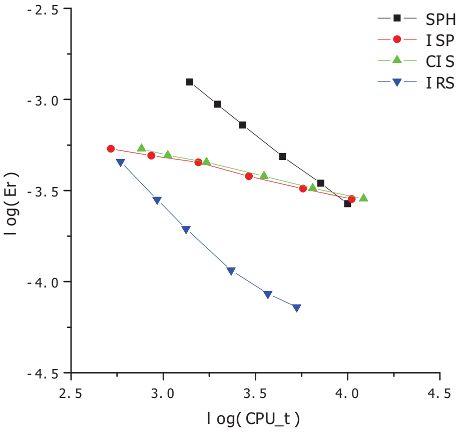
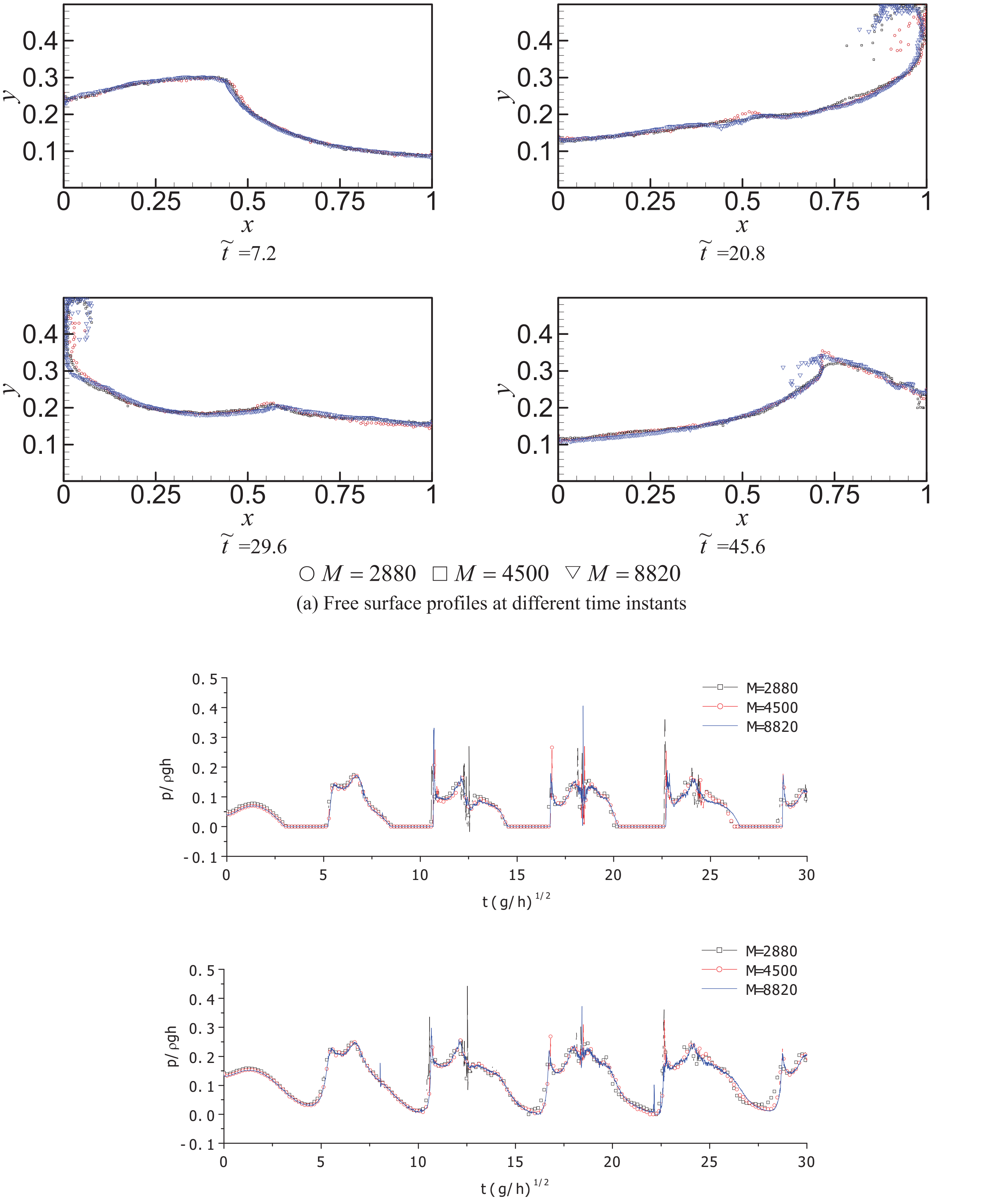


Fig. 19 Error of numerical results of four numerical methods corresponding to different number of particles

Fig. 20 CPU times used by four numerical methods corresponding to different errors

3.2.2. Violent sloshing

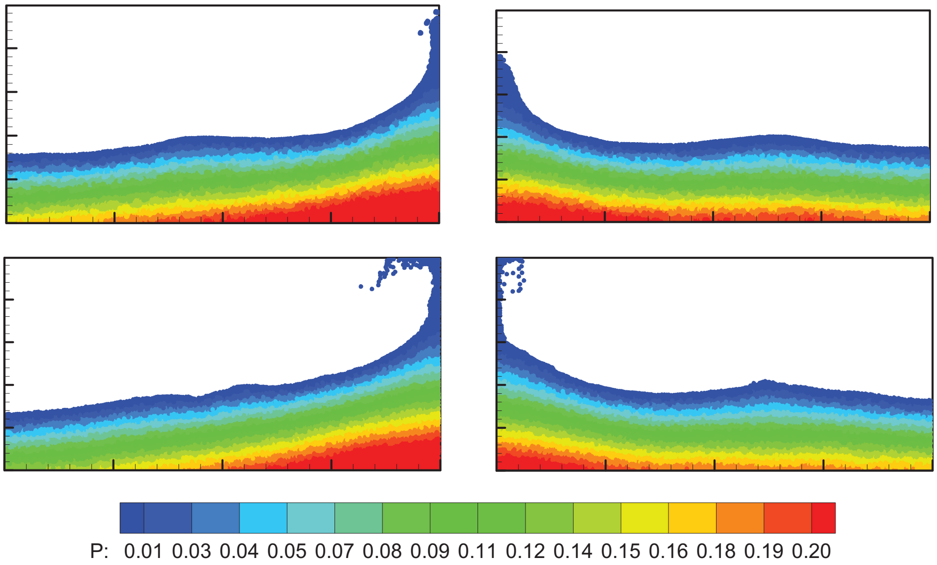
In this section, violent sloshing flow will be considered. The geometry of this tank is also rectangle with $L = 0.6m$, $h = 0.5L$, $d = 0.4h$. The parameters for this case are taken as $a_0 = 0.05m$ and $T_0 = 1.5s$, which are the same as those in Kishev et al. [38]. The behaviours of the ISPH_R method are further examined by modelling this case. For this purpose, different numbers of particles are employed with M =2880, 4500, 6480 and 8820 corresponding to non-dimensional particle size s = 0.005, 0.004, 0.00333 and 0.00286 respectively. Time step length is $\Delta\tilde{t} = 0.008$. Fig. 21 gives the free surface profiles at different time instants corresponding to different numbers of particles (the one for M=6480 is not shown for clarity). According to this figure, when the particle number is larger than 4500, the free surface profiles obtained by different particle numbers do not show significant differences, though there is some little difference near water jets. Fig.22 shows the pressure distribution at corresponding time instants for M =8820 calculated by using CISPH2 and ISPH_R methods, which demonstrates that both methods yield appropriate pressure field. Fig. 23 gives the comparison of pressure time histories at Point $h_1 / L = 0.1667$ on the left wall for M =8820 with experimental results [38]. It is noted that the experimental pressure time history given in Kishev et al. [38] did not show the transient period that must exist and they did not mention whether it was recorded on the left or on right walls. To compare our results with their experimental data, the time for the experimental results in the figure has been adjusted so that the 3rd pressure peak of the numerical results corresponds to the first peak in their paper. These figures show that the results from all the methods have similar patterns to the experimental one and that the time intervals between two consecutive impacts are almost the same. Nevertheless, there are indeed some differences. That is perhaps because of the nature of violent sloshing problems. As pointed out in Kishev et al. [38], “a large scattering of both experimental and numerical values can be observed" for this kind of sloshing waves. Furthermore, the time history from the ISPH_R method is smoother than those from CISPH2 and SPH, similar to what has observed in the cases for dam breaking above.



(a) Free surface profiles at different time instants

(b) Pressure time histories at different heights of sensor point: $h_1 / L = 0.1667$ (upper), $h_1 / L = 0.0833$ (lower)

Fig. 21 Results of the ISPH_R method obtained by using different number of particles



(a) $\widetilde{t}$ =20.8 (b) $\widetilde{t}$ =29.6

Fig. 22 Snapshots of pressure distribution of violent sloshing compared: CISPH2 (upper row) and ISPH_R (lower row)

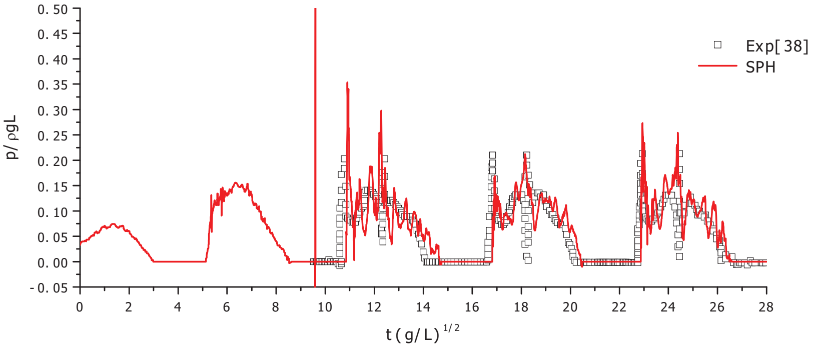


Fig. 23 (a)

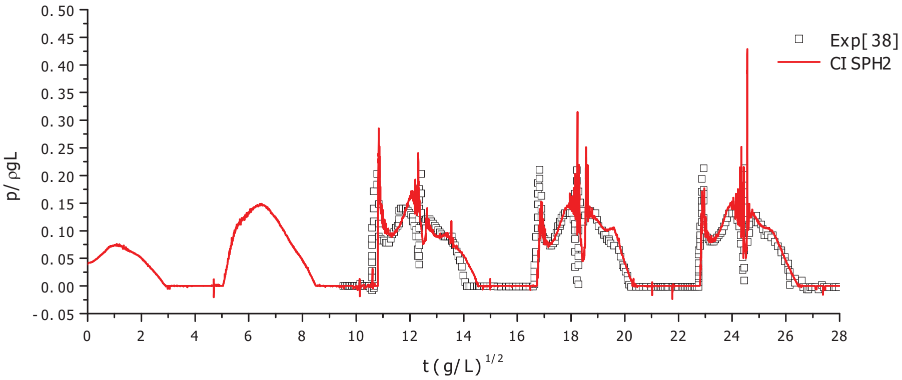


Fig. 23 (b)

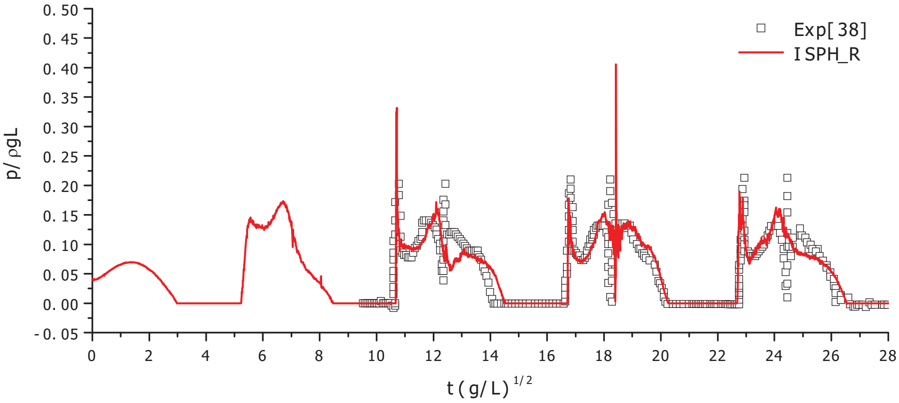


Fig. 23 (c)

Fig. 23 Comparison of pressure time histories obtained by using different methods (a)SPH, (b) CISPH2 and (c) ISPH_R

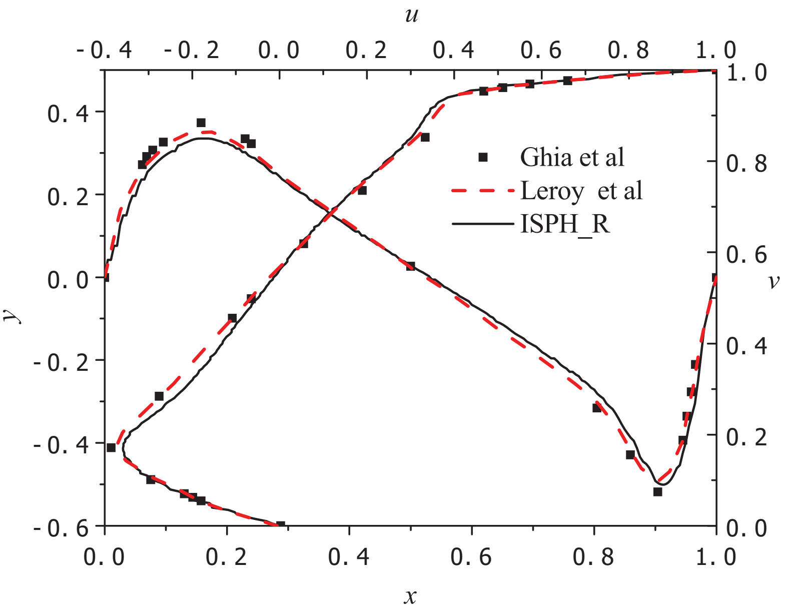
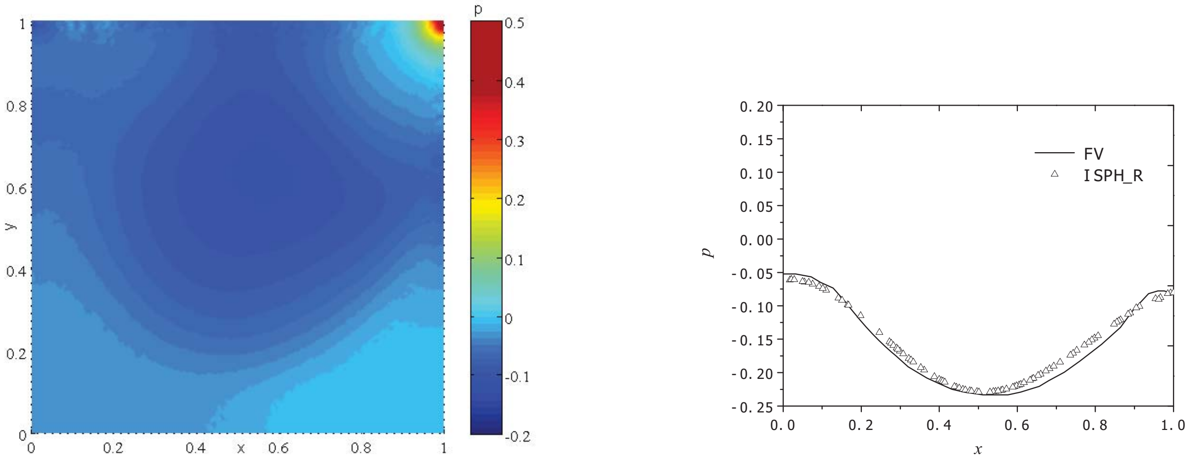


Fig. 24 Velocity comparison of lid-driven cavity flow with Ghia et al [45] and Leroy et al [46]

3.3. Lid-driven cavity flow

The last case considered is for modelling classical lid-driven cavity flow in a 2D square domain with the length of sides being 1. The flow inside the cavity is accelerated from rest by the constant motion of the lid on the top of the domain. This case is used to demonstrate that the ISPH_R method may also be employed to model fluid flow within a confined domain, though it is developed mainly for modelling violent water waves. Although the smaller Reynolds number (Re=UL/ν) corresponding to the speed (U) of the lid is tested, the results for the Reynolds number of 1000 are presented here. To model this case, the domain is discretised by 200*200 particles and non-dimensional time step is selected at 0.003 according to our numerical tests. The horizontal velocity on the line at x=0.5 and the vertical velocity on the line at y=0.5 are depicted in Fig. 24. The results from Ghia et al [45] and Leroy et al [46] are also plotted in the figure.

From this figure, one may see that although the results of the ISPH_R method are quite close to those from the cited papers, they are visibly less than others in the region near the bottom and left wall. The pressure contour and pressure along the section of y=0.5 corresponding to Fig. 24 are depicted in Fig. 25, where the result marked as FV is the pressure from Leroy et al [46], who obtained it by using a finite volume method. It is noted here that there are other results in [46] for the pressure along the section but we do not re-plot all of them for keeping clarification. Compared with the similar results given by Leroy et al [46] for the same case, the pattern of the contour is largely similar and the agreement of pressure with the FV result is at the similar level as those in [46]. It is noted that the flow in this case is driven mainly by the viscosity in particular at the earlier stage of simulation, which is quite different from water waves mainly driven by pressure gradient. The focus of this paper is to improve the ISPH by using the different method for solving the pressure equation but without improvement on the viscous stress term. If adopting the similar technique for estimating the viscous stress term to that in Leroy et al [46] which considered a non-homogeneous Neumann wall boundary condition on the velocity and the turbulent viscosity, the results may become better.



(a) Contour of pressure

(b) Pressure along the section of y=0.5

Fig. 25 Pressure of lid-driven cavity flow

4. Conclusion

This paper introduced a new incompressible SPH based on Rankine source solution (abbreviated as ISPH_R) to simulate 2D violent waves. The method adopts the Rankine solution to decrease the order of the derivatives in the Poisson equation defining the boundary value problem for pressure. The transformed Poisson equation does not include any derivative of the functions to be solved. Using the new formulation, one just needs to approximate the functions themselves during discretisation, instead of approximating their second order derivatives as in the existing incompressible SPH. In the cases of dam-breaking and sloshing wave problems, the results of the ISPH_R method are in good agreement with available experimental data and analytical solution. By comparing its results with those from traditional SPH and different forms of existing ISPH, the ISPH_R method presented in this paper is demonstrated to need a smaller number of particles and less computational costs to achieve the similar level of accuracy, or to produce more accurate results with the same number of particles for the water wave problems. Although the conclusion is obtained by using the limited cases studied in this paper, it is expected to be hold for general water wave cases where the pressure is more dominant than viscous stresses. That is because the feature of no derivatives in governing equation for pressure adopted in this paper is applied for all the water wave cases. This method can be extended to three dimensional problems, which will be presented in future. In future, comparative studies may also be carried out in order to find out if combining the particle shifting technique with the new technique for solving the pressure Poisson introduced in the paper may give better results, and if the other free surface identification technique (e.g. the one based on the free - surface divergence criteria, [21]) other than the one employed in this paper would be more accurate. In addition, an adaptive time-step technique should be adopted and the viscosity caused by turbulence should be considered.

Acknowledgements

This work is sponsored by The National Natural Science Funds of China (51009034, 51279041 and 51379051)，Foundational Research Funds for the central Universities (HEUCDZ1202，HEUCF120113) and Defense Pre Research Funds program (9140A14020712CB01158), to which the authors are most grateful. The second author also wishes to thank the Chang Jiang Visiting Chair professorship of Chinese Ministry of Education, hosted by the HEU.

References

[1] L.B. Lucy, A numerical approach to the testing of the fusion process, Astron J. 88 (1977) 1013-1024.
[2] R.A. Gingold, J.J. Monaghan, Smoothed particle hydrodynamics: theory and application to non-spherical stars, Mon. Not. R. Astron. Soc. 181 (1977) 375-389.
[3] M.B. Liu, G.R. Liu, K.Y. Lam, Z. Zong, Smoothed particle hydrodynamics for numerical simulation of underwater explosion, Comput. Mech. 30(2) (2003) 106-118.
[4] J.P. Morris, P.J. Fox, Y. Zhu, Modeling low Reynolds number incompressible flows using SPH, J. Comput. Phys. 136 (1997) 214-226.
[5] P.W. Cleary, J.J. Monaghan, Conduction modeling using smoothed particle hydrodynamics, J. Comput. Phys. 148 (1999) 227-264.
[6] J.J. Monaghan, Simulation Free Surface Flows with SPH, J. Compu. Phys. 110 (4) (1994) 399-406.
[7] J.N. Fang, R.G. Owens, L. Tacher, A. Parriaux, A numerical study of the SPH method for simulating transient viscoelastic free surface flows, J. Non-Newtonian Fluid Mech. 139(1-2) (2006) 68-84.
[8] X. Zheng, Q.W. Ma, W.Y. Duan, Simulation of breaking waves by using an improved SPH method, in: Proceedings of the Twenty-second International Offshore and Polar Engineering Conference, Rhodes, Greece, June 17-22, 2012, pp.1051-1056.
[9] J.J. Monaghan, SPH and Riemann Solvers, J. Comput. Phys. 136 (1997) 298-307.
[10] M. Antuono, A. Colagrossi, S. Marrone, D. Molteni, Free-surface flows solved by means of SPH schemes with numerical diffusive terms, Computer Physics Communications 181 (2010) 532-549.
[11] S.I. Inutsuka, Reformulation of Smoothed Particle Hydrodynamics with Riemann solver, J. Comput. Phys. 179 (2012) 238-267.
[12] R.Gao B.Ren, G.Y. Wang, Y.X. Wang, Numerical modeling of regular wave slamming on subface of open-piled structures with the corrected SPH method, Applied Ocean Research 34 (2012) 172-186.
[13] A.R. Rafiee, S. Cummins, M. Rudman, K. Thiagarajan, Comparative study on the accuracy and stability of SPH schemes in simulating energetic free-surface flows, Europen Journal of Mechanics B/Fluids 36 (2012) 1-16.
[14] S.J. Cummins, Rudman M, An SPH projection method, J. Comput. Phys. 152 (1999) 584-607.
[15] Y.M. Lo Edmond, S.D. Shao, Simulation of near-shore solitary wave mechanics by an incompressible SPH method, Applied Ocean research 24 (2002) 275-286.
[16] S.D. Shao, Y.M. Lo Edmond, Incompressible SPH method for simulating Newtonian and non-Newtonian flows with a free surface, Adv. Water Resour. 26(7) (2003) 787-800.
[17] S D Shao, C.M. Ji, D.I. Graham, D.I. Graham, D.E. Reeve, P.W. James, A.J. Chadwick, Simulation of wave overtopping by an incompressible SPH model, Coast Eng. 53(9) (2006) 723-735.
[18] S.D. Shao, Incompressible SPH simulation of water entry of a free-falling object, Int. J. Numer. Meth. Fluids 59(1) (2009) 91-115.
[19] X.Y. Hu, N.A. Adams, An incompressible multi-phase SPH method, J. Comput. Phys. 227 (2007) 264-278.
[20] X. Liu , H. Xu , S.D. Shao, P. Lin, An improved incompressible SPH model for simulation of wave-structure interaction, Computers&Fluids, 71 (2013) 113-123.
[21] S.J. Lind , R. Xu, P.K. Stansby, B.D. Rogers, Incompressible smoothed particle hydrodynamics for free-surface flows: A generalised diffusion-based algorithm for stability and validations for impulsive flows and propagating waves, J. Comput.

Phys. 231 (2012) 1499-1523.

[22] E.S. Lee, C. Mouline, R. Xu, D. Violeau, D. Laurence, P. Stansby. Comparisons of weakly compressible and truly incompressible algorithms for the SPH, J. Comput. Phys. 227 (2008) 8417–8436.

[23] J.J. Monaghan, Smoothed Particle Hydrodynamics, Rep. Prog. Phys. 68 (2005) 1703-1759.

[24] Q.W. Ma, Meshless Local Petrov-Galerkin Method for Two-dimensional Nonlinear Water Wave Problems. J. Comput. Phys. 205 (2) (2005) 611-625.

[25] Q.W. Ma, MLPG Method Based on Rankine Source Solution for Simulating Nonlinear Water Waves, Computer Modeling in Engineering & Sciences 9 (2) (2005) 193-210.

[26] Q.W. Ma, J. Zhou, MLPG_R method for numerical simulation of 2D breaking waves, Computer Modeling in Engineering & Sciences 43 (3) (2009) 277-304.

[27] J.T. Zhou, Q.W. Ma, MLPG method based on Rankine source solution for modeling 3D breaking waves, Computer Modeling in Engineering & Sciences 56(2) (2010) 179-210.

[28] S. Sriram, Q.W. Ma, Improved MLPG_R method for simulating 2D interaction between violent waves and elastic structures, J. Comput. Phys. 231 (22) (2012) 7650-7670.

[29] Q.W. Ma, A new meshless interpolation scheme for MLPG_R method, Computer Modeling in Engineering & Sciences 23(2) (2008) 75-89.

[30] J.T. Zhou, Q.W. Ma, S. Yan, Numerical implementation of solid boundary condition in meshless methods, in: Proceedings of the Eighteenth International Offshore and Polar Engineering Conference, Vancouver, Canada, July 6-11, 2008, (3) pp.16-23.

[31] A. Colagrossi, M. Landrini, Numerical simulation of interfacial flow by Smoothed Particle Hydrodynamics, J. Comput. Physic. 191 (2003) 448-475.

[32] X. Zheng, W.Y. Duan, Q.W. Ma, A new scheme for identifying free surface particles in improved SPH. China Science: Physics, Mechanics & Astronomy 55(8) (2012) 1454-1463.

[33] J.C. Martin, W.J. Moyce, Part IV: An experimental study of the collapse of liquid columns on a rigid horizontal plane, Tansaction of the royal society of London, Series A, mathematical and physical sciences 244 (882) (1952) 312-324.

[34] C.H. Pan, X.Z. Xu, B.Y. Lin, Simulating free surface flows by MAC method, Estuar. Coastal Eng. 1-2 (1993) 51-58(in Chinese).

[35] C.W. Hirt, B.D. Nichols, Volume of fluid (VOF) method for the dynamics of free boundaries, J. Comput. Phys. 39 (1981) 201-225.

[36] Z.Q. Zhou, J.O. De Kat, B. Buchner, A nonlinear 3-D approach to simulate green water dynamics on deck, in: Proceeding of Seventh International Conference of Numerical Ship Hydrodynamics, Nantes, France, July 19-22, 1999, (5)1-15.

[37] O.M. Faltinsen, A numerical non-linear method for sloshing in tanks with two dimensional flow, J. Ship Research 18(4) (1976) 224-241.

[38] Z.R. Kishev, C.H. Hu, M. Kashiwagi, Numerical simulation of violent sloshing by a CIP-based method, J. Mar. Sci. Technol. 11 (2006) 111-122.

[39] R. Fatehi, M.T. Manzari. Error estimation in smoothed particle hydrodynamics and a new scheme for second derivatives, Computers and Mathematics with Applications. 61(2011)482-498

[40] H.F. Schwaiger. An implicit corrected SPH formulation for thermal diffusion with linear free surface boundary conditions, J. Numer. Meth. Engng., 75(2008):647-671.

[41] A.J. Chorin, Numerical Solution of Navier-Stokes Equations, Math. Computation, 22 (1968), 745-762.

[42] G.X. Wu, Q.W. Ma and R Eatock Taylor, Numerical simulation of sloshing waves based on finite element method, Applied Ocean Research, Vol. 20 (1998), pp. 337-355.

[43] S. Zhang, K. Morita, F. Kenji and N. Shirakawa, An improved mps method for numerical simulations of convective heat transfer problems, International Journal for Numerical Methods in Fluids, 51 (2006), 31-47.

[44] R. Xu, P. Stansby and D. Laurence, Accuracy and stability in incompressible SPH (ISPH) based on the projection method and a new approach, Journal of Computational Physics 228 (2009) 6703–6725.

[45] U. Ghia, K.N. Ghia, C.T. Shin, High-Re solutions for incompressible flow using the Navier–Stokes equations and a multigrid method, J. Comput. Phys., 48 (1982) 387–411.

[46] A. Leroy, D.Violeau, M.Ferrand, C.Kassiotis , Unified semi-analytical wall boundary conditions applied to 2-D incompressible SPH, Journal of Computational Physics, 261(2014), 106–129.

Appendix A: Discretisation of pressure gradient at a particle on a solid boundary

In Ma (2007), the gradient of a two-dimensional function, such as pressure *p(x,y)*, at a particle *I* is approximated by the following equations

$$\left.\frac{\partial p}{\partial x}\right|_{\vec{r}_I} \approx \frac{C_{I,1}-a_{I,12}C_{I,2}}{1-a_{I,12}a_{I,21}} \tag{A1}$$

$$\left.\frac{\partial p}{\partial y}\right|_{\vec{r}_I} \approx \frac{C_{I,2}-a_{I,21}C_{I,1}}{1-a_{I,21}a_{I,12}} \tag{A2}$$

$$C_{I,m} = \frac{1}{n_{I,x_m}} \sum_{J=1,J\neq I}^{N} \left[p(\vec{r}_J)-p(\vec{r}_I)\right] \frac{\left(\vec{r}_{J,x_m}-\vec{r}_{I,x_m}\right)}{\left|\vec{r}_J-\vec{r}_I\right|^2} w\left(\left|\vec{r}_J-\vec{r}_I\right|\right) \quad (m{=}1\ \ or\ \ 2)$$

$$a_{I,mk} = \frac{1}{n_{I,x_m}} \sum_{J=1,J\neq I}^{N} \frac{\left(\vec{r}_{J,x_m}-\vec{r}_{I,x_m}\right)\left(\vec{r}_{J,x_k}-\vec{r}_{I,x_k}\right)}{\left|\vec{r}_J-\vec{r}_0\right|^2} w\left(\left|\vec{r}_J-\vec{r}_I\right|\right) \quad (m{=}1\ \ or\ \ 2,\ k{=}1\ \ or\ \ 2,\ \mathrm{m}\neq\mathrm{k})$$

$$n_{I,x_m} = \sum_{J=1,J\neq I}^{N} \frac{\left(\vec{r}_{J,x_m}-\vec{r}_{I,x_m}\right)^2}{\left|\vec{r}_J-\vec{r}_I\right|^2} W\left(\left|\vec{r}_J-\vec{r}_I\right|\right) \quad (m{=}1\ \ or\ \ 2)$$

where N is the number of particles affecting Particle I, x_1=x, x_2=y, $\vec{r}_{J,x}$ is the component of the position vector in x (or y) direction. For the convenience of discretising Eq. (26), we may define

$$\overline{n}_{I,xy} = \sum_{J=1,J\neq I}^{N} \frac{\left(\vec{r}_{J,x_m}-\vec{r}_{I,x_m}\right)\left(\vec{r}_{J,x_k}-\vec{r}_{I,x_k}\right)}{\left|\vec{r}_J-\vec{r}_0\right|^2} w\left(\left|\vec{r}_J-\vec{r}_I\right|\right) \quad (m{=}1\ \ or\ \ 2,\ k{=}1\ \ or\ \ 2,\ \mathrm{m}\neq\mathrm{k})$$

$$B_{IJ,x_m} = \frac{\left(\vec{r}_{J,x_m}-\vec{r}_{I,x_m}\right)}{\left|\vec{r}_J-\vec{r}_I\right|^2} w\left(\left|\vec{r}_J-\vec{r}_I\right|\right)$$

$$D_{I,x_m} = \sum_{J=1,J\neq I}^{N} \left[p(\vec{r}_J)-p(\vec{r}_I)\right] B_{IJ,x_m}$$

and $a_{I,12}=\overline{n}_{I,xy}/n_{I,x}$, $a_{I,21}=\overline{n}_{I,xy}/n_{I,y}$ and $C_{I,m}=D_{I,x_m}/n_{I,x_m}$

After simple manipulation, Eq. (A1) and Eq. (A2) can be written as

$$\left.\frac{\partial p}{\partial x}\right|_{\vec{r}_I} \approx \frac{n_{I,y}D_{I,x}-\overline{n}_{I,xy}D_{I,y}}{n_{I,x}n_{I,y}-\overline{n}_{I,xy}^{\ 2}} = \sum_{J=1,J\neq I}^{N} \frac{n_{I,y}B_{IJ,x}-\overline{n}_{I,xy}B_{IJ,y}}{n_{I,x}n_{I,y}-\overline{n}_{I,xy}^{\ 2}} \left[p(\vec{r}_J)-p(\vec{r}_I)\right] \tag{A3}$$

$$\left.\frac{\partial p}{\partial y}\right|_{\vec{r}_I} \approx \frac{n_{I,x}D_{I,y}-\overline{n}_{I,xy}D_{I,x}}{n_{I,x}n_{I,y}-\overline{n}_{I,xy}^{\ 2}} = \sum_{J=1,J\neq I}^{N} \frac{n_{I,x}B_{IJ,y}-\overline{n}_{I,xy}B_{IJ,x}}{n_{I,x}n_{I,y}-\overline{n}_{I,xy}^{\ 2}} \left[p(\vec{r}_J)-p(\vec{r}_I)\right] \tag{A4}$$

Using the above two equations (A3 and A4), the gradient of pressure in the normal direction ($\vec{\mathbf{n}}$) of a solid boundary can be discretised as

$$\left.\vec{\boldsymbol{n}}\cdot\nabla p\right|_{\vec{r}_I} \approx \sum_{J=1,J\neq I}^{N} \left[\frac{n_{I,y}B_{IJ,x}-\overline{n}_{I,xy}B_{IJ,y}}{n_{I,x}n_{I,y}-\overline{n}_{I,xy}^{\ 2}} n_x + \frac{n_{I,x}B_{IJ,y}-\overline{n}_{I,xy}B_{IJ,x}}{n_{I,x}n_{I,y}-\overline{n}_{I,xy}^{\ 2}} n_y\right] \left[p(\vec{r}_J)-p(\vec{r}_I)\right]$$

or

$$\vec{\boldsymbol{n}} \cdot \nabla p\big|_{\vec{r}_I} \approx \sum_{J=1}^{N} \Psi_{IJ}\, p\left(\vec{r}_J\right) \tag{A5}$$

where

$$\Psi_{IJ} = \frac{n_{I,y} B_{IJ,x} - \overline{n}_{I,xy} B_{IJ,y}}{n_{I,x} n_{I,y} - \overline{n}_{I,xy}{}^{2}} n_x + \frac{n_{I,x} B_{IJ,y} - \overline{n}_{I,xy} B_{IJ,x}}{n_{I,x} n_{I,y} - \overline{n}_{I,xy}{}^{2}} n_y \qquad J \neq I \tag{A6}$$