



## Turbulence particle models for tracking free surfaces

## Modèles particuliers turbulents pour suivre les surfaces libres

SONGDONG SHAO, Formerly Postdoctoral Fellow of Japan Society for the Promotion of Science (JSPS). *Coastal and Offshore Engineering, Department of Urban and Environmental Engineering, Kyoto University, Kyoto 606-8501, Japan.*

*E-mail: shao@coast.kuciv.kyoto-u.ac.jp. Currently Joint Research Fellow of School of Mathematics and Statistics and School of Engineering, University of Plymouth, Drake Circus, Plymouth PL4 8AA, UK. Tel.: +44 175223 2791;*

*fax: +44 175223 2780; e-mail: songdong.shao@plymouth.ac.uk (author for correspondence)*

HITOSHI GOTOH, Associate Professor, *Waterfront Environmental Engineering, Department of Urban and Environmental Engineering, Kyoto University, Kyoto 606-8501, Japan. Tel.: +81 75 753 5098; fax: +81 75 753 5098;*

*e-mail: gotoh@coast.kuciv.kyoto-u.ac.jp*

### ABSTRACT

Two numerical particle models, the Smoothed Particle Hydrodynamics (SPH) and Moving Particle Semi-implicit (MPS) methods, coupled with a sub-particle scale (SPS) turbulence model, are presented to simulate free surface flows. Both SPH and MPS methods have the advantages in that the governing Navier–Stokes equations are solved by Lagrangian approach and no grid is needed in the computation. Thus the free surface can be easily and accurately tracked by particles without numerical diffusion. In this paper different particle interaction models for SPH and MPS methods are summarized and compared. The robustness of two models is validated through experimental data of a dam-break flow. In addition, a series of numerical runs are carried out to investigate the order of convergence of the models with regard to the time step and particle spacing. Finally the efficiency of the incorporated SPS model is further demonstrated by the computed turbulence patterns from a breaking wave. It is shown that both SPH and MPS models provide a useful tool for simulating free surface flows.

### RÉSUMÉ

Deux modèles numériques particuliers, SPH (hydrodynamique de particules lissée) et MPS (Méthodes semi-implicites de particules mobiles), couplés à un modèle de turbulence à une échelle sub-particulaire (SPS), sont présentés pour simuler des écoulements à surface libre. Les deux méthodes SPH et MPS ont l'avantage de résoudre les équations de Navier-Stokes utilisées, par une approche lagrangienne sans nécessité de mailler le calcul. Ainsi la surface libre peut se définir facilement et exactement par des particules sans diffusion numérique. Dans cet article différents modèles de l'interaction particulaire pour les méthodes SPH et MPS sont résumés et comparés. La robustesse des deux modèles est validée par des données expérimentales de l'écoulement d'une rupture de barrage. En outre, une série d'exécutions numériques sont effectuées pour étudier la convergence des modèles en fonction du pas de temps et de l'espacement des particules. Enfin l'efficacité du modèle SPS incorporé est de plus démontrée par le calcul de la configuration turbulente d'une vague déferlante. On montre ainsi que les deux modèles SPH et MPS fournissent un outil utile pour simuler des écoulements à surface libre.

*Keywords:* Smoothed Particle Hydrodynamics, Moving Particle Semi-implicit, sub-particle scale, free surface.

### 1 Introduction

Modeling free surface flows is of great significance in the field of hydrodynamics and hydraulic engineering. However, these flows are difficult to simulate because the surface boundary conditions are specified on an arbitrarily moving surface. The MAC (Harlow and Welch, 1965) and VOF (Hirt and Nichols, 1981) methods are two of the most flexible and robust approaches for treating free surfaces in Eulerian grid methods. The former uses marker particles to define the free surface while the latter solves a

transport equation for the volume fraction of the fluid. However, as for both the MAC and VOF methods, the Navier–Stokes (N–S) equations are solved on a fixed grid, which brings the problem of numerical diffusion due to the existence of the advection term in the N–S equations. The numerical diffusion becomes severe especially when the deformation of free surface is very large and complicated.

Thus numerical models that do not use the grid have been developed to overcome this problem and the mesh-free modeling has become increasingly popular. Among them, the deterministic

meshless or gridless methods can be roughly classified into two groups, i.e., those based on the gridless Eulerian methods, such as the gridless Euler/N–S solution algorithm (Batina, 1993) and element free Galerkin (Belytschko *et al.*, 1994); and those based on fully Lagrangian particle approaches, such as the Smoothed Particle Hydrodynamics (SPH) (Lucy, 1977) and Moving Particle Semi-implicit (MPS) (Koshizuka *et al.*, 1995) methods. The second group of particle models is the focus of interest in this paper.

Both the SPH and MPS methods have the common feature in that the modeled media, such as the water, is modeled as an assembly of particles. The interaction zone is supposed around each individual particle. All of the terms in governing equations are described as the interaction between the reference particle and its neighbors and thus the computational grid is not required. If combined with necessary initial and boundary conditions, a particular hydrodynamic problem can be solved exclusively through particle properties. Unlike the particle-in-cell (PIC) method (Shao *et al.*, 2002), the SPH and MPS models do not need the grid to calculate spatial derivatives and therefore, the back and forth interpolations between the grid and particles are avoided.

Initially developed by Koshizuka *et al.* (1995), the MPS method has been widely applied in both the hydrodynamics and nuclear mechanics, to deal with a variety of practical problems such as the dam-breaking (Koshizuka and Oka, 1996), vapor explosion (Koshizuka *et al.*, 1999) and two-phase gas–liquid flows (Yoon *et al.*, 1999). Later it was improved and extended by Gotoh and Sakai (1999) and Gotoh *et al.* (2001a) into coastal engineering to study wave breaking and two-phase sediment–water interactions. On the other hand, the conception of SPH method was first put forward by Lucy (1977) for astrophysics applications. Later Monaghan (1992) summarized the SPH theory and extended its applications into many other fields including the fluid flow. The initial SPH algorithm was fully explicit and the fluid flow was assumed to be weakly compressible when it was applied to a solitary wave running up a beach (Monaghan and Kos, 1999) and a low Reynolds number flow inside the porous media (Zhu *et al.*, 1999). Based on the classical SPH theory, an incompressible semi-implicit SPH projection scheme was later developed by Cummins and Rudman (1999) for a non-free surface flow. Since both the SPH and MPS methods employ a similar semi-implicit numerical scheme to treat incompressible flow in this paper, the major differences between the two lie only in the use of different kernel functions or particle interaction models. The basic governing equations and equation solvers have the same form.

The current paper is organized in the following ways. First, the governing equations and equation solvers are presented for both the SPH and MPS methods. The Large Eddy Simulation (LES) conception is employed to formulate a sub-particle scale (SPS) model to address the turbulence issues. Next the treatment of free surfaces and wall boundaries by particles is described. Then various kernel functions or particle interaction models employed for the formulation of gradient, divergence and Laplacian operators

are presented and compared. In the model applications, both models are used to reproduce a documented dam-break experiment to verify their robustness. Meanwhile a series of numerical runs using different particle spacing and time steps are carried out to investigate the order of convergence. Finally the efficiency of the coupled SPS turbulence model is further demonstrated by comparing the computed surface profiles and turbulence intensities of a breaking wave with those found in the literatures.

## 2 Governing equations and equation solvers

### 2.1 Governing equations

Both the SPH and MPS models employ the Lagrangian form of N–S equations, which are represented in the following form as

$$\frac{1}{\rho} \frac{D\rho}{Dt} + \nabla \cdot \mathbf{u} = 0 \quad (1)$$

$$\frac{D\mathbf{u}}{Dt} = -\frac{1}{\rho} \nabla P + \mathbf{g} + \nu_0 \nabla^2 \mathbf{u} + \frac{1}{\rho} \nabla \cdot \vec{\tau} \quad (2)$$

where  $\rho$  = fluid density;  $t$  = time;  $\mathbf{u}$  = velocity;  $P$  = pressure;  $\mathbf{g}$  = gravitational acceleration;  $\nu_0$  = laminar kinematic viscosity and  $\vec{\tau}$  = SPS or turbulence stress contributed by the unresolved small motions.

In order to account for the above unresolved small motion term  $\vec{\tau}$ , the philosophy of LES (Rogallo and Moin, 1984) is referenced here. According to the original LES conception, eddies capable of being resolved by the computational grid are allowed to evolve according to the N–S equations and a model is employed to represent the turbulence at sub-grid scales. By introducing the turbulence eddy viscosity  $\nu_t$ , the unresolved SPS turbulence stress  $\tau_{ij}$  in Eq. (2) can be written as

$$\frac{\tau_{ij}}{\rho} = 2\nu_t S_{ij} - \frac{2}{3} k \delta_{ij} \quad (3)$$

where  $\delta_{ij}$  is Kronecker's operator; and  $S_{ij}$  = strain rate and  $k$  = turbulence kinetic energy, which can be incorporated into the pressure term when solving the momentum equation (2).

The widely used Smagorinsky (1963) model is employed here to formulate the turbulence eddy viscosity as

$$\nu_t = (C_s \Delta X)^2 |\bar{S}| \quad (4)$$

where  $C_s$  = Smagorinsky constant (taken as 0.1 in the computations);  $\Delta X$  = particle to particle spacing (equivalent of the mixing length in a grid method) and  $|\bar{S}|$  = local strain rate, which can be calculated from the resolved variables.

### 2.2 Equation solvers

The prediction–correction numerical schemes of the SPH and MPS methods consist of two steps, similar to the two-step projection method by Chorin (1968) to solve the N–S equations. The first prediction step is an explicit integration in time without enforcing incompressibility. Only the stress tensor, laminar viscosity and gravitational terms in the N–S equation (2) are used and

thus an intermediate particle velocity and positions are obtained

$$\Delta \mathbf{u}_* = \left( \mathbf{g} + \nu_0 \nabla^2 \mathbf{u} + \frac{1}{\rho} \nabla \cdot \vec{\tau} \right) \Delta t \quad (5)$$

$$\mathbf{u}_* = \mathbf{u}_t + \Delta \mathbf{u}_* \quad (6)$$

$$\mathbf{r}_* = \mathbf{r}_t + \mathbf{u}_* \Delta t \quad (7)$$

where  $\Delta \mathbf{u}_*$  = changed particle velocity during the prediction step;  $\Delta t$  = time increment;  $\mathbf{u}_t$  and  $\mathbf{r}_t$  = particle velocity and position at time  $t$ ; and  $\mathbf{u}_*$  and  $\mathbf{r}_*$  = intermediate particle velocity and position.

In the second correction step, the pressure term is used to update the particle velocity obtained from the prediction step

$$\Delta \mathbf{u}_{**} = - \frac{1}{\rho_*} \nabla P_{t+1} \Delta t \quad (8)$$

$$\mathbf{u}_{t+1} = \mathbf{u}_* + \Delta \mathbf{u}_{**} \quad (9)$$

where  $\Delta \mathbf{u}_{**}$  = changed particle velocity during the correction step;  $\rho_*$  = intermediate particle density after the prediction step; and  $P_{t+1}$  and  $\mathbf{u}_{t+1}$  = particle pressure and velocity at time  $t + 1$ .

Finally the positions of a particle are calculated by

$$\mathbf{r}_{t+1} = \mathbf{r}_t + \frac{(\mathbf{u}_t + \mathbf{u}_{t+1})}{2} \Delta t \quad (10)$$

where  $\mathbf{r}_t$  and  $\mathbf{r}_{t+1}$  = positions of particle at time  $t$  and  $t + 1$ .

The pressure for updating the particle velocity in the correction step is obtained by setting  $D\rho/Dt = 0$  at each particle according to the mass conservation equation (1) after the prediction step. The aim of this semi-implicit process is to enforce the incompressibility or mass conservation, which has been violated by the prediction computations. By combining Eqs (1) and (2), a pressure Poisson equation for the SPH numerical schemes is formulated as follows

$$\nabla \cdot \left( \frac{1}{\rho_*} \nabla P_{t+1} \right) = \frac{\rho_0 - \rho_*}{\rho_0 \Delta t^2} \quad (11)$$

where  $\rho_0$  = initial density at each of the particles.

The source term on the right side of the above equation is the deviation of particle density, while it is usually the divergence of intermediate velocity in a grid method. Here it should be pointed out that the MPS model employs a slightly different form of the source term, in which the particle density  $\rho$  is replaced by the particle number density  $n$ . However, since the number density  $n$  is proportional to  $\rho$  only by a constant (which will be shown in the later section), Eq. (11) is applicable to both numerical schemes.

Besides, for the purpose of stable and accurate computations, the time step  $\Delta t$  is controlled in SPH and MPS simulations to constrain the particle displacement in one step. Therefore, the requirement of Courant condition and viscous diffusion can be satisfied, which is the same as the stability requirement in a grid method.

### 3 Numerical treatment of free surfaces and wall boundaries

#### 3.1 Free surfaces

For SPH and MPS models free surfaces can be easily identified and tracked by particles. Since no particle exists in the outer region of the free surface, the particle density  $\rho$  or number density  $n$  drops abruptly on the surface. A particle is regarded as a surface particle if its density is 1% (for SPH) or 3% (for MPS) lower than that of the inner fluid.

The kinematic free surface boundary condition states that the fluid particles at a free surface  $\eta = \eta(x, t)$  should remain on the free surface, represented by

$$\frac{\partial \eta}{\partial t} + u \frac{\partial \eta}{\partial x} = w \quad (12)$$

where  $w$  = vertical velocity of surface particles. In the computation, this requirement can be easily satisfied by assigning all surface particles an identifier at the beginning and then continuously maintaining their identity in the code.

The dynamic free surface boundary condition requires that, along the free surface boundary, the normal stress is equal to the atmospheric pressure and the tangential stress is zero. This condition is also simplified by giving a prescribed pressure value to all surface particles.

#### 3.2 Solid walls

Solid walls such as the impermeable bottom bed and inclined slope are also modeled by particles, which balance the pressure of inner fluid particles and prevent them from penetrating the wall. The pressure Poisson equation (11) is solved on these wall particles to repulse the inner fluid particles accumulating in the vicinity of the wall. The homogeneous Neumann boundary conditions are enforced when solving the pressure equation. Usually the velocities of wall particles are set zero to represent the non-slip boundary condition.

### 4 SPH and MPS formulations

In this section, the major differences between the SPH and MPS formulations are presented and compared to show their respective characteristics.

#### 4.1 Kernel functions/interaction models

SPH and MPS methods employ different kernel functions or interaction models in their formulations. The use of different interaction models is similar to the use of different difference schemes in a finite difference method. While different governing equations can employ different interaction models, usually the same interaction model is used throughout all governing equations in one numerical method. By balancing the computational accuracy and efficiency, the following kernel based on the spline

function and normalized in 2-D is widely adopted in the SPH formulation (Monaghan, 1992)

$$\begin{aligned}\omega(r) &= \frac{10}{7\pi r_e^2} \left[ 1 - \frac{3}{2} \left( \frac{r}{r_e} \right)^2 + \frac{3}{4} \left( \frac{r}{r_e} \right)^3 \right] & 0 \leq \frac{r}{r_e} \leq 1 \\ \omega(r) &= \frac{10}{28\pi r_e^2} \left[ 2 - \left( \frac{r}{r_e} \right)^3 \right] & 1 < \frac{r}{r_e} \leq 2 \\ \omega(r) &= 0 & \frac{r}{r_e} > 2\end{aligned}\quad (13)$$

where  $r$  = separation distance between two particles and  $r_e$  = smoothing distance or interaction range, which determines the degree by which a particle interacts with its neighbors. In the current SPH computations,  $r_e$  is set to be  $2.0 \Delta x$ , where  $\Delta x$  is the initial particle spacing. The SPH kernel equation (13) has the advantage of possessing compact support. Its second derivative is continuous and the dominant error term in the integral interpolant is the order of  $r_e^2$ . The continuity of the second derivative means that the kernel is not too sensitive to particle disorder and the errors in approximating the integral interpolants by summation interpolants are small provided the particle disorder is not too large.

On the other hand, the following basic interaction model by Koshizuka *et al.* (1995) is often employed in the MPS formulation

$$\begin{aligned}\omega(r) &= \frac{r_e}{r} - 1 & 0 < \frac{r}{r_e} \leq 1 \\ \omega(r) &= 0 & \frac{r}{r_e} > 1\end{aligned}\quad (14)$$

This function, which also has the finite interaction zone, is equally effective for saving the computational time and memory. Besides, the value of this interaction model is infinity at  $r = 0$ , which is quite efficient of avoiding the clustering of particles and thus improves the numerical stability. In comparison, the SPH kernel equation (13) has a finite value at  $r = 0$ . However, the derivatives of MPS kernel are not always continuous and thus it is less smooth than the SPH counterpart.

In the MPS computation, different sizes of the interaction range are used, i.e., the size  $r_e$  is  $2.1\Delta x$  for the calculation of density and gradient, while it is  $4.0\Delta x$  for the calculation of Laplacian and viscosity. These values have been selected through a large number of optimum MPS numerical tests (Koshizuka and Oka, 1996).

#### 4.2 Density models

Using both the SPH and MPS conceptions, any quantity of particle  $a$  can be approximated by the direct summation of the relevant quantities of its neighboring particles  $b$ . For example, the density of particle  $\rho_a$  is represented by SPH formulation as

$$\rho_a = \sum_b m_b \omega(|\mathbf{r}_a - \mathbf{r}_b|) \quad (15)$$

where  $a$  and  $b$  = reference particle and its neighbors;  $\mathbf{r}_a$  and  $\mathbf{r}_b$  = position of particles; and  $m_b$  = particle mass.

Another kind of particle density, the number density  $n_a$ , is defined in the MPS model as follows

$$n_a = \sum_b \omega(|\mathbf{r}_a - \mathbf{r}_b|) \quad (16)$$

It is very obvious that there is almost no difference between the particle density defined by Eq. (15) and number density defined by Eq. (16), since they are simply proportional to each other through a constant  $m_b$ . Although the kernel functions  $\omega(r)$  in Eqs (15) and (16) are different, the integration of  $\omega(r)$  over the entire interaction area keeps constant if  $r_e$  is fixed (Koshizuka and Oka, 1996).

#### 4.3 Gradient and divergence models

In the SPH conception, the gradient term has many different forms depending on the derivation used. The following symmetric form is widely used since it conserves linear and angular momentum exactly (Monaghan, 1992)

$$\left( \frac{1}{\rho} \nabla P \right)_a = \sum_b m_b \left( \frac{P_a}{\rho_a^2} + \frac{P_b}{\rho_b^2} \right) \nabla_a \omega_{ab} \quad (17)$$

where  $\nabla_a \omega_{ab}$  = gradient of the kernel taken with respect to the positions of particle  $a$ . Similarly, the divergence of a vector  $\mathbf{u}$  at particle  $a$  can be formulated by

$$\nabla \cdot \mathbf{u}_a = \rho_a \sum_b m_b \left( \frac{\mathbf{u}_a}{\rho_a^2} + \frac{\mathbf{u}_b}{\rho_b^2} \right) \cdot \nabla_a \omega_{ab} \quad (18)$$

On the other hand, the following more intuitive formulations in the form of particle interactions are employed by the MPS method for gradient and divergence

$$\left( \frac{1}{\rho} \nabla P \right)_a = \frac{1}{\rho_a} \frac{\text{Dim}}{n_0} \sum_b \omega(|\mathbf{r}_a - \mathbf{r}_b|) \frac{P_a - P_b}{|\mathbf{r}_a - \mathbf{r}_b|^2} (\mathbf{r}_a - \mathbf{r}_b) \quad (19)$$

$$\left( \frac{1}{\rho} \nabla \cdot \mathbf{u} \right)_a = \frac{1}{\rho_a} \frac{\text{Dim}}{n_0} \sum_b \omega(|\mathbf{r}_a - \mathbf{r}_b|) \frac{\mathbf{u}_a - \mathbf{u}_b}{|\mathbf{r}_a - \mathbf{r}_b|^2} \cdot (\mathbf{r}_a - \mathbf{r}_b) \quad (20)$$

where Dim = the number of space dimensions and  $n_0$  = initial particle number density.

It is shown that in the SPH formulation, the gradient or divergence is always related to the differentiation of the kernel function in form of  $\nabla_a \omega_{ab}$ , while in the MPS approach it has nothing to do with the differentiation of kernel itself, but calculated directly through the difference of particle properties. The MPS kernel serves as something like a weight function. The same characteristics are also observed in the following formulations of Laplacian and viscosity terms. Besides, the turbulence shear stress in Eq. (2) is also formulated in the same way as the divergence model.

#### 4.4 Laplacian and viscosity formulations

In the SPH model, for the purpose of particle pressure stability, the Laplacian is formulated as a hybrid of a standard SPH

first derivative with a finite difference approximation for the first derivative, and thus it is globally first-order accurate

$$(\nabla^2 P)_a = \sum_b m_b \frac{4}{\rho_a + \rho_b} \frac{(P_a - P_b)(\mathbf{r}_a - \mathbf{r}_b) \cdot \nabla_a \omega_{ab}}{|\mathbf{r}_a - \mathbf{r}_b|^2} \quad (21)$$

The same approach is applied to treat the laminar viscosity term so that it has a clear physical meaning

$$(\nu_0 \nabla^2 \mathbf{u})_a = \sum_b m_b \frac{2(\nu_a + \nu_b)}{\rho_a + \rho_b} \frac{(\mathbf{u}_a - \mathbf{u}_b)(\mathbf{r}_a - \mathbf{r}_b) \cdot \nabla_a \omega_{ab}}{|\mathbf{r}_a - \mathbf{r}_b|^2} \quad (22)$$

In the MPS formulation, the Laplacian is discretized as the re-distribution of the scalar quantity in the following form

$$(\nabla^2 P)_a = -\frac{2\text{Dim}}{n_0 \lambda} \sum_b (P_a - P_b) \omega(|\mathbf{r}_a - \mathbf{r}_b|) \quad (23)$$

The above equation is derived based on a simple time-dependant diffusion problem by Koshizuka and Oka (1996). The model is conservation because the quantity lost by particle  $a$  is gained by particle  $b$ . The coefficient  $\lambda$  is introduced to make the increase of variance due to the re-distribution of Eq. (23) equal to the increase of variance estimated from the unsteady diffusion equation, which is approximately estimated by (Gotoh and Sakai, 1999)

$$\lambda = \frac{\sum_b \omega(|\mathbf{r}_a - \mathbf{r}_b|) |\mathbf{r}_a - \mathbf{r}_b|^2}{\sum_b \omega(|\mathbf{r}_a - \mathbf{r}_b|)} \quad (24)$$

By using a same approach, the laminar viscosity term for MPS model is represented by

$$(\nu_0 \nabla^2 \mathbf{u})_a = -\frac{2\nu_0 \text{Dim}}{n_0 \lambda} \sum_b (\mathbf{u}_a - \mathbf{u}_b) \omega(|\mathbf{r}_a - \mathbf{r}_b|) \quad (25)$$

Using the present Laplacian model of either SPH or MPS, the left side of the pressure Poisson equation (11) is discretized into simultaneous linear equations. The linear equations are solved using the Incomplete Cholesky Conjugate Gradient (ICCG) method, which is the most time-consuming part of simulations. The particle-list generation optimum method by Koshizuka *et al.* (1998) and Monaghan and Lattanzio (1985) significantly reduced the computational load from  $\sim N^2$  to  $\sim N$ , where  $N$  is particle numbers.

## 5 Model applications

The SPH and MPS models are validated here using a classic dam-break problem for two reasons. Firstly, dam-break flow is an important practical problem in civil engineering and its prediction is the required element in the design of a dam and its surrounding environment. Secondly, this is also a good test for the numerical model because the existence of both a vertical and horizontal free surfaces provides a convincing check on the accuracy of numerical schemes. In this example, a rectangular column of water in hydrostatic equilibrium is confined between two vertical and one bottom walls. The water column is  $a = 0.1$  m wide and  $H = 0.2$  m high. At the beginning of the computation, the right

wall (dam) is instantaneously removed and the water is allowed to flow out along a dry horizontal bed.

During the simulations the total number of fluid particles is  $N = 20 \times 40$ , corresponding to a particle spacing of 0.005 m in the initial configuration. The fluid particles were initially arranged in a regular, equally spaced grid, with boundary particles added to form the left-hand wall and bed. In the computation a constant time step of  $\Delta t = 0.0004$  s was employed. The laminar viscosity is  $\nu_0 = 10^{-6}$  m<sup>2</sup>/s.

The particle configurations computed by the SPH and MPS models for the flow at different times  $t = 0.06, 0.12$  and  $0.20$  s, are shown in Fig. 1(a)–(c), respectively. It is seen that both models well reproduce the formation of the shock as well as the subsequent turbulence bore, which is due to the advantage that particle methods can clearly describe the free surface without numerical diffusion. The simulated flow patterns are very similar to the computational results by the MAC (Harlow and Welch, 1965) and VOF (Hirt and Nichols, 1981) approaches. In order to quantitatively evaluate the accuracy of the numerical results, the computed time-dependent leading edge is given in Fig. 2, in which the experimental data from Martin and Moyce (1952) are shown for comparison. In the figure, the non-dimensional time  $T = t\sqrt{2g/a}$  and leading edge  $X = x/a$  are used. The good agreement between them is quite satisfactory.

Fig. 1 clearly shows that the instantaneous collapse of the dam causes an abrupt change of the flow pattern, characterized by the resulting unsteady motion. Computations by both the SPH and MPS models indicate the formation of the shock on the right corner near the bottom as shown in Fig. 1 (b) at time  $t = 0.12$  s and the propagation of subsequent turbulence bores as shown in Fig. 1 (c) at time  $t = 0.20$  s. Almost the same flow patterns have been obtained by both numerical methods. However, as can be seen from Fig. 1 (c), the turbulence bore front simulated by the MPS model is much more splashing and violent than that obtained by the SPH. This is due to the use of different interaction models in the numerical schemes. The MPS interaction model of Eq. (14) is much “stiffer” than the corresponding kernel equation (13) employed by the SPH, in that it approaches an infinite value as the distance between two particles becomes too small. Thus it created a huge repulsive force to disperse particles. Besides, according to Figs 1 and 2, in which the time-dependent leading edge tends to be a straight line after the initial non-linear increase, it can be roughly estimated that the stable turbulence bore forms around  $T = 2.8$  or  $t = 0.2$  s. Also the bore velocity is estimated to be about 2.0 m/s. For engineering purpose, a useful relationship can thus be established between these two parameters and the dam height  $H$ . For example, the bore velocity is estimated by  $\sqrt{2gH}$  and the time of bore formation is around  $\sqrt{2H/g}$ . Similar conclusions have also been reported in other literatures (Monaghan, 1992).

In order to further disclose some important flow features of the dam-break, the computed dynamic pressure contours by MPS and turbulence eddy viscosity contours by SPH are shown in Figs 3(a)–(c) and 4(a)–(c), respectively. Since the distribution patterns from two numerical models are almost the same, the results are presented here only for one of them. Dynamic pressure

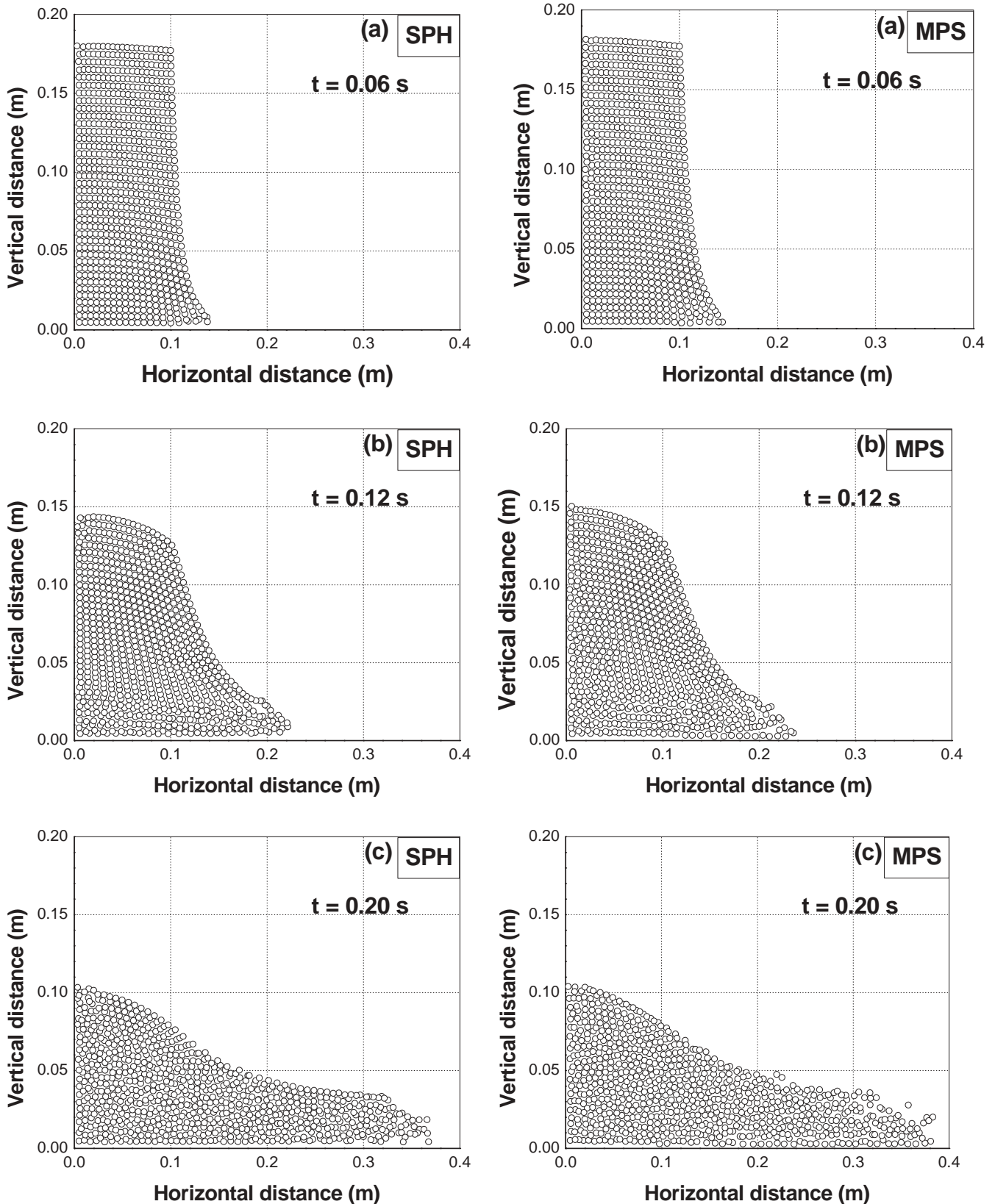


Figure 1 (a)–(c) Particle snapshots of dam-break flows at time  $t = 0.06, 0.12$  and  $0.20$  s by SPH and MPS models.

$P_d$  is the difference between the total pressure and hydrostatic pressure and its value was normalized by  $\rho g$  in Fig. 3. Turbulence eddy viscosity  $\nu_t$  is a measurement of turbulence intensities of the flow, which is also related to the turbulence energy  $k$  (Gotoh *et al.*, 2001b). In Fig. 4, its value was normalized by

the laminar viscosity  $\nu_0 = 10^{-6} \text{ m}^2/\text{s}$ . From the pressure contours in Fig. 3, it is shown that at the early stage of the dam-break at time  $t = 0.06$  s, the dynamic pressure is quite strong and globally distributed in the whole flow domain. The absolute value of peak negative pressure is close to the total pressure. This is

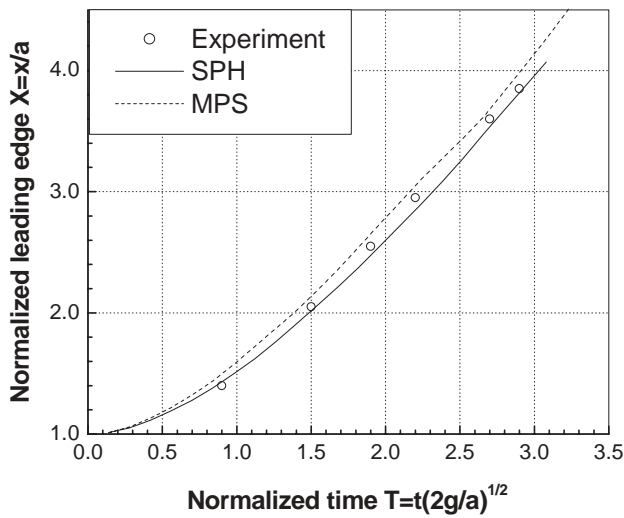


Figure 2 Comparisons between experimental and computational time-dependant leading edges by SPH and MPS models.

due to the large downwards particle acceleration after the sudden release of the right dam. As time goes on, the initial large acceleration decreases and the pressure distributions gradually become uniform at time  $t = 0.12$  s. The large dynamic pressure is locally concentrated near the initial dam-site and the amplitude of the negative pressure is only 1/3 of its previous value at  $t = 0.06$  s. After a stable turbulence bore forms at  $t = 0.2$  s, the pressure distribution is almost hydrostatic everywhere and the influence of dynamic pressure is so small as to be neglected. From the above pressure analysis, another useful conclusion can be drawn for practical purposes. The widely used shallow water equation (SWE), which is based on the hydrostatic assumption and uniform velocity over the depth, should be used with discretion when addressing the dam-break problem, especially at the early stage. SWE would be accurate enough during the later stages of the dam-break flow after a stable turbulence bore has fully evolved, which is characterized by the time scale of  $\sim \sqrt{2H/g}$ .

The capability of the incorporated SPS turbulence model to deal with turbulence is shown in Fig. 4, in which the time-dependant turbulence production and transportation processes are illustrated. It shows that at the early stages of dam-break at time  $t = 0.06$  s, the turbulence is initiated and confined to the right corner of the flow near the bottom. The turbulence level is quite low, the peak value of which is only 300 times higher than the laminar value. As time elapses on, the turbulence convects with the flow front and its intensity also increases rapidly, reaching two times higher at time  $t = 0.12$  s than the previous one. When a stable turbulence bore forms at time  $t = 0.20$  s, the high turbulence area is widely distributed inside the main flow and the general turbulence intensity is between 500–1000 times of that of the laminar flow. Based on the above findings, it is concluded that the turbulence levels of the dam-break flow vary significantly in both temporal and spatial domains. In the early simulation of a similar problem by Shao and Lo (2003), only a large constant viscosity was assumed throughout the simulation without using an appropriate turbulence model. Thus it could not treat this problem in a realistic way. Especially at the early stages

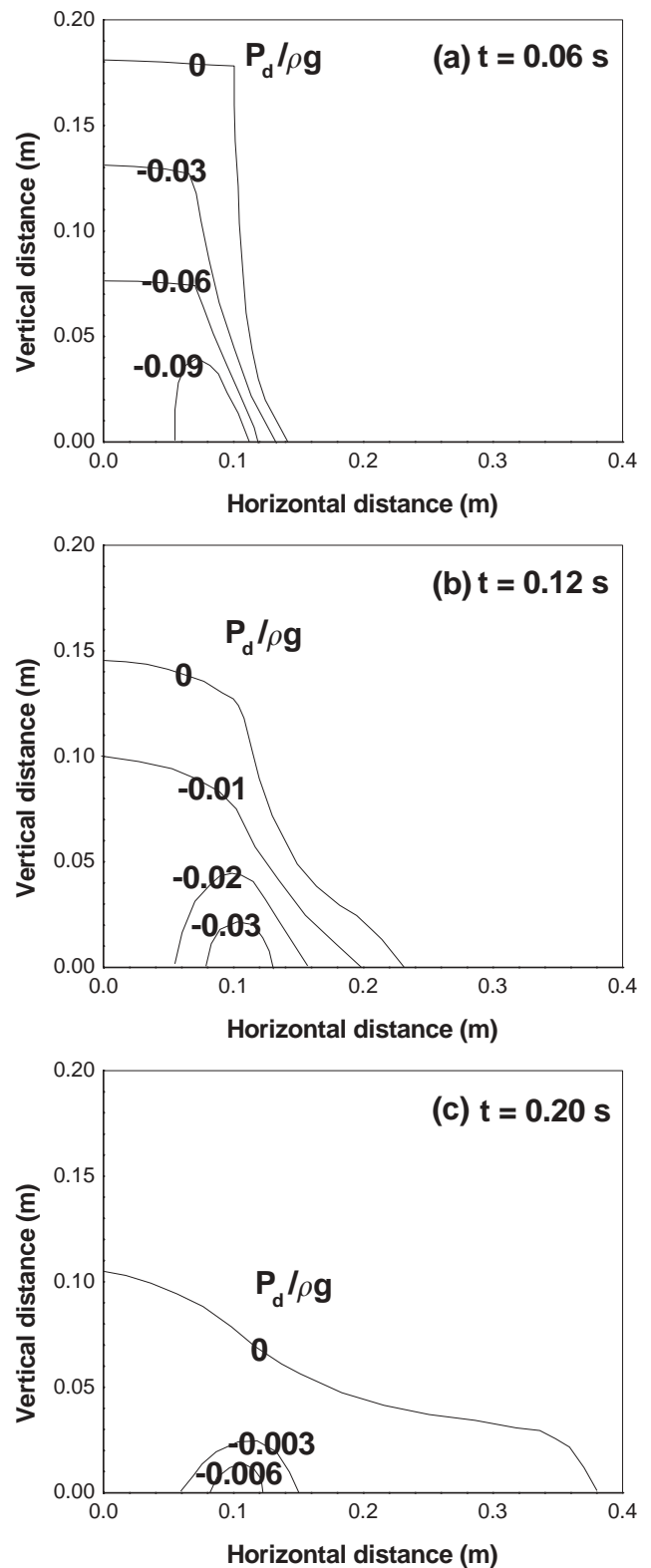


Figure 3 (a)–(c) Dynamic pressure contours of dam-break flow (by MPS model).

of the dam-break, the assumed large viscosity dampened the fluid motion, thus causing delay to the prediction of dam-break disaster. However, in the later stages of the dam-break after a stable turbulence bore has formed, we believe that numerical models employing a constant viscosity value would be accurate enough for engineering purposes.



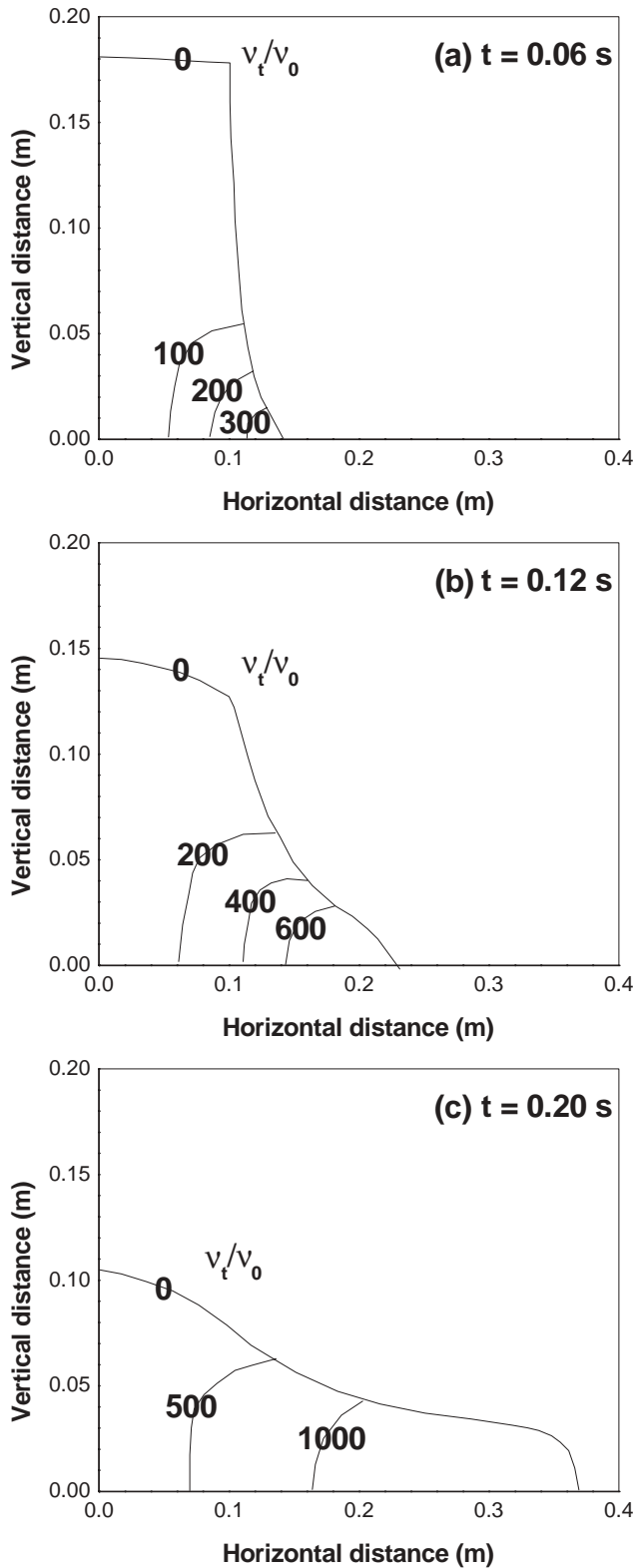


Figure 4 (a)–(c) Turbulence eddy viscosity contours of dam-break flow (by SPH model).

Conservation of mass or incompressibility provides a convincing self-check on the accuracy of incompressible numerical models. For the incompressible SPH and MPS models proposed in the paper, a quantitative measurement of the conservation of incompressibility is provided by evaluating the difference of time-dependant particle densities  $\rho(t)$  (or number density

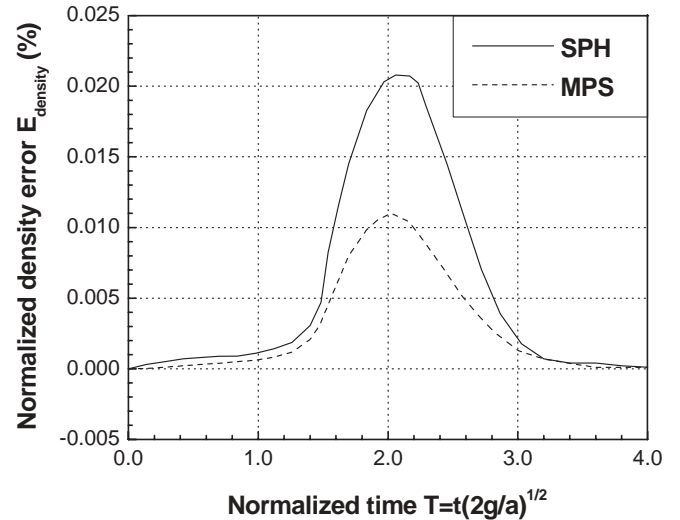


Figure 5 Time-dependant density errors by SPH and MPS computations.

$n(t)$  with the initial value  $\rho_0$  (or  $n_0$ ). It is calculated as the averaged-density variation of all particles, normalized with respect to the initial density, which is formulated by

$$E_{\text{density}}^{\text{SPH}}(t) = \frac{1}{N} \sum_{i=1}^N \text{abs}[\rho_i(t) - \rho_0] / \rho_0 \quad (26)$$

$$E_{\text{density}}^{\text{MPS}}(t) = \frac{1}{N} \sum_{i=1}^N \text{abs}[n_i(t) - n_0] / n_0 \quad (27)$$

In an ideal computation, particles should be moved to positions to satisfy  $E_{\text{density}}(t)$  to be zero at all times. However, it is not applicable in the real situation and the local particle density error cannot be avoided. This accumulation in the density error is not beyond expectation. For any incompressible numerical scheme, the resulting velocity divergence free or incompressibility is achieved only within a spatial truncation error either by the computer processor or by the numerical scheme itself. Errors in particle positions will lead to density errors. Similar problems were also reported in the SPH projection method by Cummins and Rudman (1999). In a stable numerical computation, the particle density error  $E_{\text{density}}(t)$  would also be expected to be stable.

For further analysis, the normalized time-dependant particle density errors  $E_{\text{density}}(t)$  are shown in Fig. 5 based on the SPH and MPS results. It is shown that the general accuracy of both models is equally satisfactory in that the density error is within the order of  $10^{-4}$ . The maximum density error occurs around  $T \sim 2.0$  (just prior to the formation of turbulence bore) and then decreases rapidly to near zero without accumulating. In addition, comparisons of the density error between two numerical models indicate that the peak density error from the MPS computation is about 50% of that from the SPH result, which means that the MPS conserves incompressibility better than the SPH. This can be attributed to the use of different kernels or interaction models. The infinite kernel of Eq. (14) employed by MPS is more effective in suppressing particle density error as compared with the finite kernel of Eq. (13) employed by SPH, which leads to higher density error accumulations.



## 6 Analysis of SPH and MPS numerical schemes

In this section, a series of numerical runs using different particle spacing  $\Delta X$  and time steps  $\Delta t$  are carried out to determine the order of convergence of the SPH and MPS models.

### 6.1 Numerical error with initial particle spacing $\Delta X$

To evaluate the accuracy of the numerical scheme with particle spacing  $\Delta X$ , additional two runs with coarser and finer particle spacing are implemented, i.e.,  $N = 10 \times 20$  and  $40 \times 80$ , respectively, corresponding to a particle spacing of 0.01 and 0.0025 m, as compared with the initial run with particle number of  $N = 20 \times 40$  and particle spacing of 0.005 m. The time step  $\Delta t$  is maintained at 0.0004 s throughout all runs for consistency. The normalized leading edge and time for the original and two additional runs are plotted in Fig. 6 (a) and (b) for SPH and MPS computations, respectively. It is shown clearly that in both cases the tendency of curves approaches an asymptotic line as the particle spacing  $\Delta X$  becomes smaller, i.e., the particle number  $N$

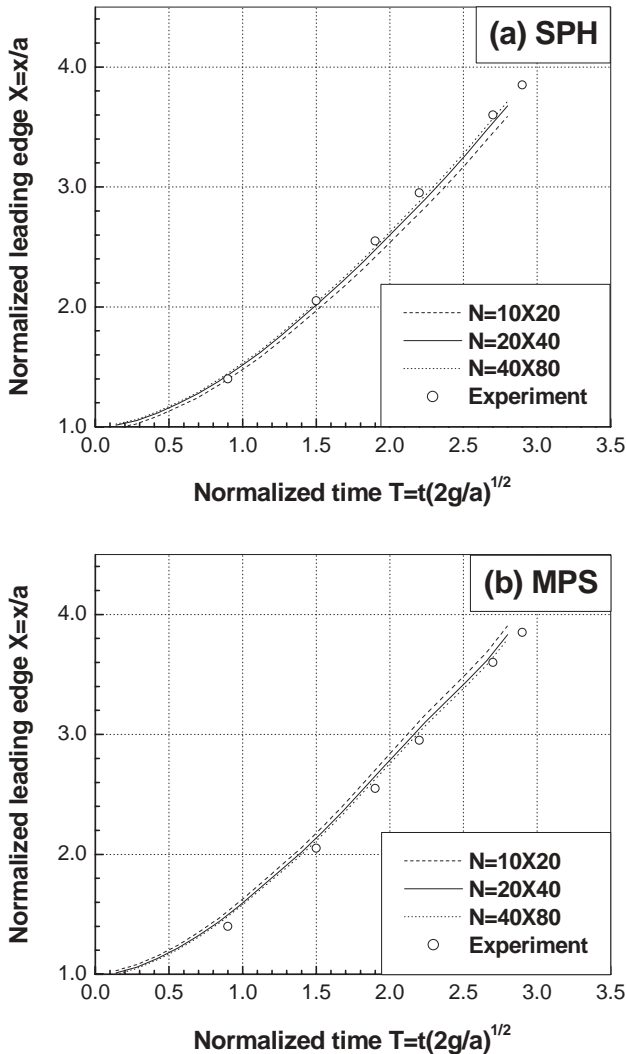


Figure 6 (a) Order of convergence with particle spacing  $\Delta X$  using different SPH runs. (b) Order of convergence with particle spacing  $\Delta X$  using different MPS runs.

becomes larger. However, the convergence rate of the MPS runs seems to be slower than that of the SPH.

Since neither the converged nor the exact solution is known, the order of convergence can be quantitatively determined only through the differences among three numerical trials. A similar approach was also employed by Ogami (1999) to analyze a Lagrangian scheme for compressible flow. According to his proposal, the numerical error  $E_{\Delta X}$  of any run with the accurate solution is proportional to a function of the particle spacing  $(\Delta X)^{CR}$ , where  $CR$  is the order of convergence. Thus the following relationship can be approximately established to relate the numerical error with particle spacing as

$$\frac{E_{10 \times 20} - E_{20 \times 40}}{E_{20 \times 40} - E_{40 \times 80}} \approx \frac{(\Delta X_{10 \times 20})^{CR} - (\Delta X_{20 \times 40})^{CR}}{(\Delta X_{20 \times 40})^{CR} - (\Delta X_{40 \times 80})^{CR}} \quad (28)$$

According to the design of numerical runs, it has

$$\Delta X_{10 \times 20} = 2\Delta X_{20 \times 40} = 4\Delta X_{40 \times 80} \quad (29)$$

Thus Eq. (28) is simplified to

$$\frac{E_{10 \times 20} - E_{20 \times 40}}{E_{20 \times 40} - E_{40 \times 80}} \approx 2^{CR} \quad (30)$$

In order to calculate the numerical differences between two adjacent runs, the following sampling method is applied to Fig. 6

$$E_{10 \times 20} - E_{20 \times 40} = X_{10 \times 20}^s - X_{20 \times 40}^s \quad (31)$$

$$E_{20 \times 40} - E_{40 \times 80} = X_{20 \times 40}^s - X_{40 \times 80}^s \quad (32)$$

where  $X_{10 \times 20}^s$ ,  $X_{20 \times 40}^s$  and  $X_{40 \times 80}^s$  are sample values of the leading edge  $X$  and the sample points  $s$  are uniformly distributed along the horizontal time axis. Each sample point corresponds to a specified  $CR$ . We can use enough points to get a series of  $CR$  and then take the mean value to represent the order of convergence.

By using 100 points from Fig. 6 (a) and (b), the mean value of  $CR$  is calculated to be 1.25 for the SPH and 0.95 for the MPS. Thus the accuracy of SPH numerical scheme is represented by  $O(\Delta X^{1.25})$ . This conclusion is also consistent with the theoretical analysis by Monaghan and Kos (1999), who suggested that the error from the SPH integral interpolant is about the order of  $O(\Delta X^2)$ . But further error is introduced when the integral interpolant is replaced by the summation interpolant as shown in various SPH formulations. Also in developing the Laplacian and viscosity equations (21) and (22), a first-order finite difference scheme was incorporated. Thus the total accuracy of the proposed incompressible SPH model would be expected to be lower than the theoretical value of  $O(\Delta X^2)$ . On the other hand, the mean value of  $CR$  calculated from Fig. 6 (b) indicates that the accuracy of the current MPS numerical scheme is around  $O(\Delta X)$ . Although the MPS computations have lower density errors as shown in Fig. 5, its spatial accuracy lags behind the SPH scheme, which employed highly accurate integral interpolants.

### 6.2 Numerical error with time step $\Delta t$

For checking the accuracy of the numerical scheme with time step  $\Delta t$ , another additional two runs with finer time steps  $\Delta t = 0.0002$  and  $0.0001$  s are carried out, in comparison with

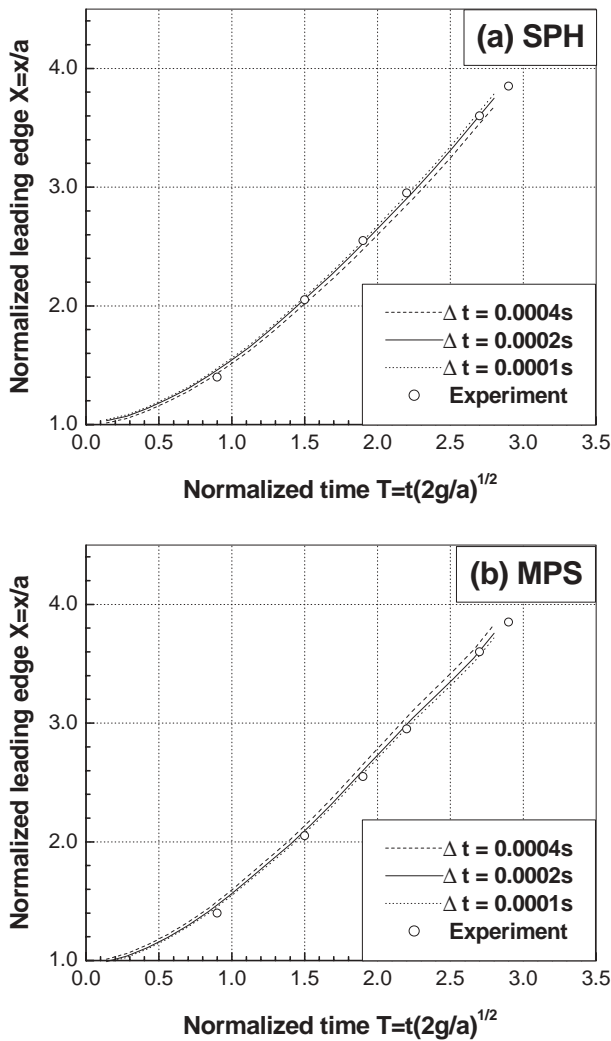


Figure 7 (a) Order of convergence with time step  $\Delta t$  using different SPH runs. (b) Order of convergence with time step  $\Delta t$  using different MPS runs.

the initial run with  $\Delta t = 0.0004$  s. In all runs the particle spacing  $\Delta X = 0.005$  m is kept unchanged, i.e., the particle number is  $N = 20 \times 40$ . The normalized leading edge and time are given in Fig. 7 (a) and (b) for the SPH and MPS runs, respectively. The convergence of the numerical results with the halved time step is very obvious.

To determine the convergence rate, a similar analysis as employed in the previous section is used. If we assume that the numerical error  $E_{\Delta t}$  of any run with the accurate solution is proportional to a function of  $(\Delta t)^{CR}$ , a relationship analogous to Eq. (30) can be established to relate the numerical errors to time step as

$$\frac{E_{0.0004} - E_{0.0002}}{E_{0.0002} - E_{0.0001}} \approx 2^{CR} \quad (33)$$

Again by using 100 points from Fig. 7 (a) and (b), the mean value of  $CR$  is calculated to be 0.97 and 0.98 for the SPH and MPS computations, respectively, suggesting that the numerical accuracy of both models is about  $O(\Delta t)$  in temporal domain. This finding is consistent with the formulations of SPH and MPS equation solvers, in that the pressure projection is performed at intermediate particle positions  $\mathbf{r}_*$  which were found from an

$O(\Delta t)$  integration of velocity  $\mathbf{u}$  through Eq. (7). Besides, the particle is updated to its corrected position by Eq. (10), which is also first-order accurate in time. Thus the general accuracy of both numerical schemes should be around  $O(\Delta t)$ .

Finally it should be noted that Fig. 6 (a) and (b) and the like clearly show the different sequences of three curves. For example, in Fig. 6 (a) for the SPH result, the  $N = 40 \times 80$  curve is above the  $N = 20 \times 40$  curve but the  $N = 10 \times 20$  curve is below. While with regard to the MPS runs in Fig. 6 (b), the sequence of three curves is just the opposite. This is attributed to the differences in the direction of convergence in two methods. The MPS converged to the final solution (which could be represented by the experimental data in the figures) in the downward direction while the SPH converged to the final solution in the upward direction. The same explanation is also applicable to Fig. 7.

## 7 Validation of SPS turbulence model

In order to fully demonstrate the robustness of the incorporated LES SPS turbulence model as well as the accuracy of SPH and MPS numerical schemes, a solitary wave breaking on the beach is used as another convincing test. Solitary wave breaking is also an important practical problem relevant to tsunami hazard mitigation in coastal regions. In this section, the laboratory breaking solitary wave data from Synolakis (1986) is used for verification.

The basic parameters are summarized here. In the experiment, the still water depth  $d$  varied from 0.21 to 0.29 m and the incident wave height was  $H/d = 0.28$ . The wave broke and ran up on a mild 1 : 20 slope. In the SPH and MPS simulations, the computational domain starts at a distance of 3.0 m upstream of the toe of the beach and extends to the location beyond the maximum runup point. Considering the spatial resolution, the still water depth of  $d = 0.29$  m is selected. The initial particle spacing is  $\Delta X = 0.01$  m and thus at least 29 particles are distributed along each of the vertical lines in the main flow area. Totally about  $N = 18,000$  particles are used in the simulations. The time step  $\Delta t$  is automatically adjusted to achieve computational efficiency and meanwhile, to satisfy the stability requirement. To be consistent with the experimental results, the time scale was normalized to  $t(g/d)^{1/2}$  and the water surface elevation was normalized to  $\eta/d$ . The initial solitary wave profile was produced according to Monaghan and Kos (1999).

In order to improve the computational accuracy in the near-wall region and swash zone where the wave runs up and down, a simple wall function is used to relate the velocity at the computational particles closet to the bed to the instantaneous shear stress at the bed. A quadratic resistant law based on a logarithmic velocity profile is used in the paper, similar to that presented by Christensen and Deigaard (2001) in their 3-D LES computations of plunging and spilling breakers.

The computed wave profiles by SPH and MPS models are shown in Fig. 8 (a)–(d), corresponding to the events of wave pre-breaking at time  $t(g/d)^{1/2} = 15$ , breaking at  $t(g/d)^{1/2} = 20$ , post-breaking at  $t(g/d)^{1/2} = 30$  and maximum runup at  $t(g/d)^{1/2} = 45$ , respectively. The experimental data from

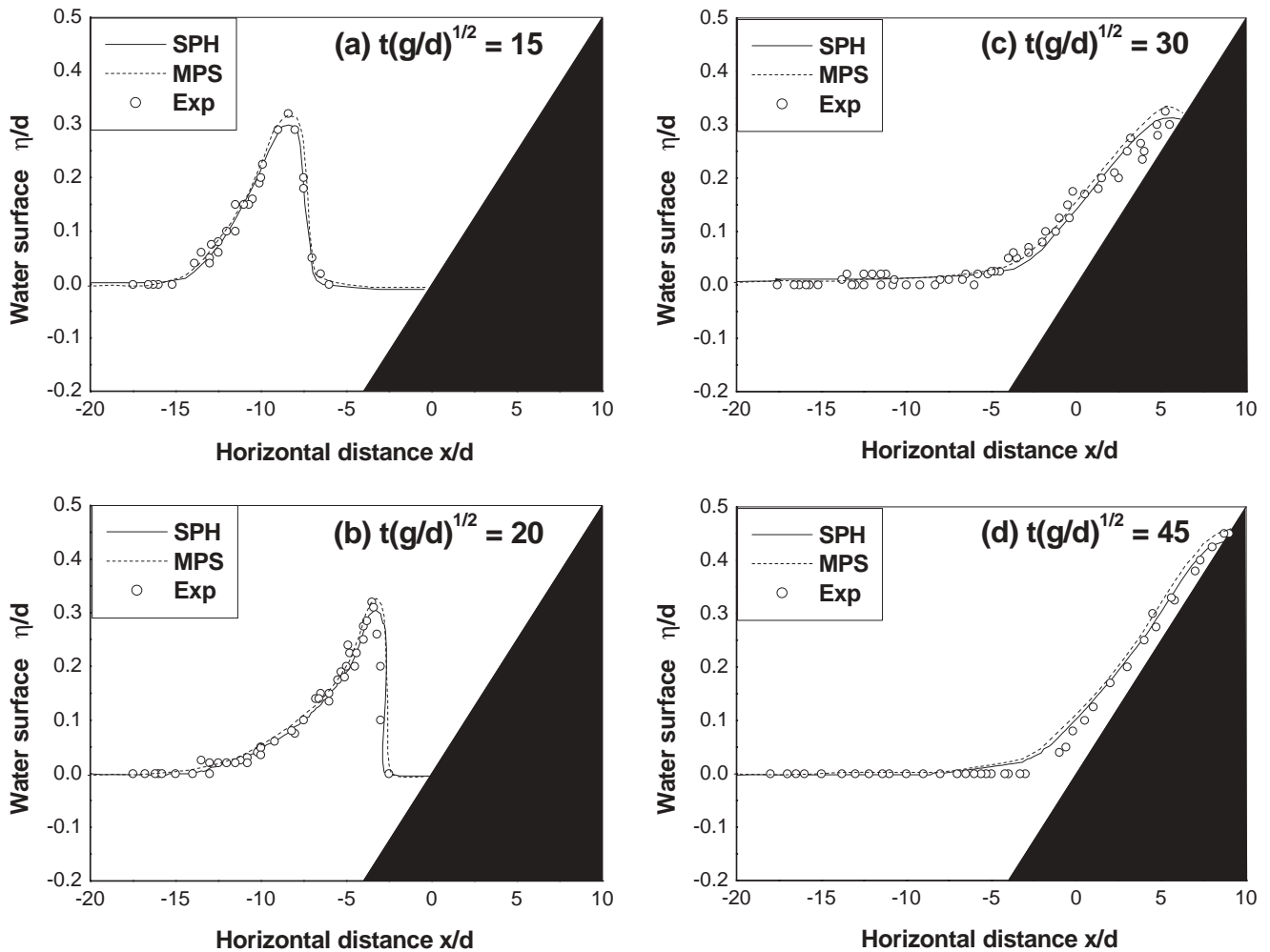


Figure 8 (a)–(d) Experimental and computational wave profiles by SPH and MPS models.

Synolakis (1986) are given for comparison. Once again, the good agreement between them verifies the accuracy of the proposed particle models for tracking free surfaces with less or no numerical diffusion. The maximum runup height in Fig. 8 (d) calculated by both SPH and MPS models is about  $0.45d$ , which is quite close to the numerical results of  $0.48d$  reported by Lin *et al.* (1999), using a RANS approach. This suggests that particle models could give out good results in the region where there is not sufficient number of particles on condition that a suitable treatment procedure is implemented. It is also seen that MPS computations always overestimate the height of wave profiles as compared with the SPH results. This is due to the employment of different particle interaction models as had been illustrated in the previous section.

In the computations, no obvious reflections of the wave from the slope are observed. This phenomenon is completely different from the reflections of a non-breaking solitary wave running up the beach, in which the reflection is continuous and quite strong. One of the main reasons is that the breaking solitary wave dissipates out much of the wave energy during the breaking process, while there is no significant energy loss in a non-breaking solitary wave.

In order to give some proofs to support this assumption, the computed time sequences of turbulence energy intensities

$(2k)^{1/2}/c$  (which have been normalized by the phase velocity of solitary wave in constant depth) by two numerical models are plotted in Fig. 9 (a)–(d). It is shown that the small scale turbulence begins to appear near the wave front prior to breaking at time  $t(g/d)^{1/2} = 15$ . The turbulence intensity increases to the peak level during the breaking at  $t(g/d)^{1/2} = 20$ , when the high turbulence area is widely distributed in the wave front and thus the wave energy is significantly dissipated by this process. Later the broken wave runs up the slope and the remaining turbulence energy has been gradually dampened out by the mean flow at  $t(g/d)^{1/2} = 30$ . When the wave tongue reaches its maximum runup height at  $t(g/d)^{1/2} = 45$ , the overall turbulence level is close to zero and only very small scale turbulence emerges near the still-water shore line. This indicates that the lower part of wave front had already begun to retreat and therefore, producing slightly higher turbulence levels. The similar turbulence production, transportation and dissipation processes have also been reported in detail by Lin *et al.* (1999), who solved the RANS equations coupled with a  $k-\varepsilon$  model. Besides, the general turbulence evolution features computed by both SPH and MPS models are analogous to each other. This is not beyond expectation since the same turbulence model is coupled to the two numerical solvers.

Finally it should be pointed out that the spatial resolution in the above SPH and MPS runs is still not fine enough so that

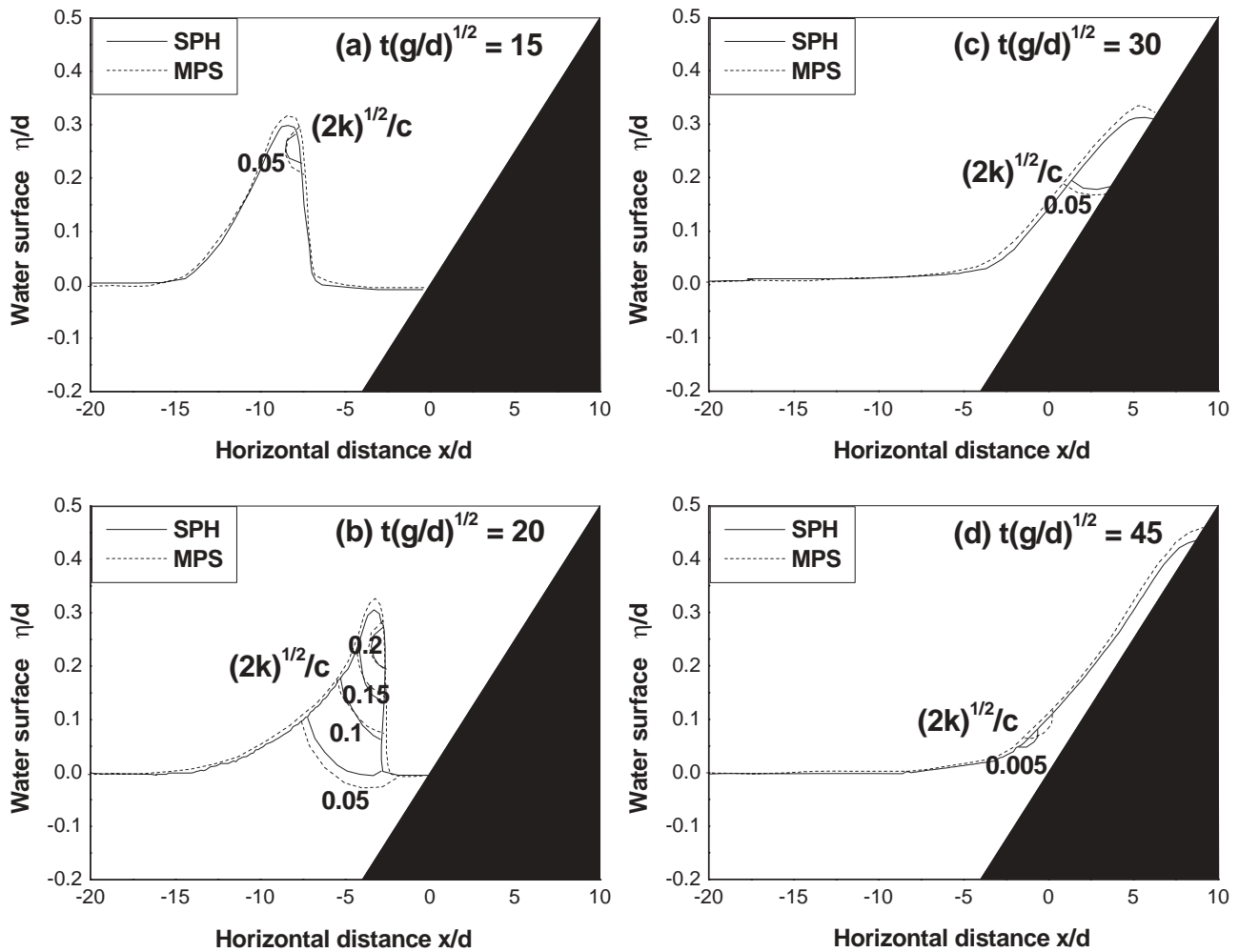


Figure 9 (a)–(d) Computed turbulence intensity distributions of breaking wave by SPH and MPS models.

the real essence of LES modeling has not been fully explored. Nonetheless, the good agreement of turbulence evolutions with those obtained by Lin *et al.* (1999) demonstrates that the LES modeling is a fairly promising approach in that the computational effort is reduced since there is no need to solve additional turbulence transportation and dissipation equations.

## 8 Conclusion

The paper presented the SPH and MPS methods for simulating free surface flows. The two particle models solve the Lagrangian form of N–S equations and track free surfaces simply and accurately by particles without numerical diffusion. A series of numerical runs using different particle spacing and time steps for a dam-break simulation show that the spatial and temporal accuracy of the SPH numerical scheme is  $O(\Delta t + \Delta X^{1.25})$  and it is  $O(\Delta t + \Delta X)$  for the MPS scheme. In addition, the computed turbulence evolution patterns from a wave-breaking simulation indicate that the incorporated LES SPS turbulence model provides an effective tool to address turbulence issues in wave dynamics. Computationally, the advantages of the proposed SPH and MPS models lie in that they are easy to program. The dam-break runs were finished within 30 min and the wave breaking runs were finished within 3 h using a CPU 2.2 G and RAM 1.0 G PC.

Future work is needed to quantitatively validate the SPS turbulence model. Besides, the particle modeling conceptions should be extended into two-phase flows to study sediment motion under the breaking wave.

## Acknowledgment

This research work was supported by the Japan Society for the Promotion of Science (JSPS). We are grateful to Professor Tetsuo Sakai, Department of Urban and Environmental Engineering of Kyoto University, for his invaluable comments and suggestions. Besides, the first author extremely appreciates the guidance and help from Professor Edmond Lo Yat-Man, School of Civil and Environmental Engineering, Nanyang Technological University, during the coding process of the SPH model when the author was Research Fellow in Singapore.

## Notation

- $a$  = Dam width
- $c$  = Wave phase celerity
- $CR$  = Order of convergence
- $C_s$  = Smagorinsky constant
- $d$  = Constant water depth

Dim = Number of spatial dimensions  
 $E_{\text{density}}$  = Normalized density error  
 $E_{\Delta t}$  = Numerical error with time step  
 $E_{\Delta X}$  = Numerical error with particle spacing  
 $\mathbf{g}$  = Gravitational acceleration  
 $H$  = Dam height or wave height  
 $k$  = Turbulence energy  
 $m$  = Particle mass  
 $n$  = Particle number density  
 $n_0$  = Initial number density  
 $N$  = Number of particles  
 $P$  = Pressure  
 $P_d$  = Dynamic pressure  
 $r$  = Distance between particles  
 $\mathbf{r}$  = Position vector  
 $r_e$  = Kernel range or interaction distance  
 $|\bar{S}|$  = Local shear rate  
 $S_{ij}$  = Element of strain tensor  
 $T$  = Normalized time  
 $\mathbf{u}$  = Velocity vector  
 $w$  = Vertical velocity of surface particle  
 $X$  = Normalized leading edge  
 $\delta_{ij}$  = Kronecker operator  
 $\Delta t$  = Time increment  
 $\Delta \mathbf{u}$  = Changed velocity  
 $\Delta X$  = Initial particle spacing  
 $\eta$  = Water surface elevation  
 $\lambda$  = Diffusion coefficient  
 $\nu_0$  = Laminar viscosity  
 $\nu_t$  = Turbulence eddy viscosity  
 $\rho$  = Fluid density  
 $\rho_0$  = Initial density  
 $\tau_{ij}$  = Element of stress tensor  
 $\vec{\tau}$  = Sub-particle scale stress tensor  
 $\omega$  = Kernel function or interaction model

#### Subscripts and symbols

$a$  = Reference particle  
 $b$  = Neighboring particle  
 $ab$  = Values between particle  $a$  and  $b$   
 $s$  = Sample point  
 $t$  = Time  
 $*$  = Intermediate value  
 $**$  = Corrective value

#### References

- BATINA, T. (1993). "A Gridless Euler/Navier–Stokes Solution Algorithm for Complex Aircraft Applications". AIAA Paper 93-0333, American Institute of Aeronautics and Astronautics.
- BELYTSCHKO, T., LU, Y.Y. and GU, L. (1994). "Element Free Galerkin Methods". *Int. J. Numer. Methods Eng.* 37, 229–235.
- CHORIN, A.J. (1968). "Numerical Solution of the Navier–Stokes Equations". *Math. Comp.* 22, 745–762.
- CHRISTENSEN, E. D. and DEIGAARD, R. (2001). "Large Eddy Simulation of Breaking Waves". *Coast. Eng.* 42(1), 53–86.
- CUMMINS, S.J. and RUDMAN, M. (1999). "An SPH Projection Method". *J. Comp. Phys.* 152, 584–607.
- GOTOH, H. and SAKAI, T. (1999). "Lagrangian Simulation of Breaking Waves Using Particle Method". *Coast. Eng. J.* 41(3 and 4), 303–326.
- GOTOH, H., SAKAI, T. and HAYASHI, M. (2001a). "Lagrangian Two-Phase Flow Model for the Wave Generation Process due to Large-Scale Landslides". *Asian and Pacific Coast. Eng.*, October 18–21, Dalian China, 176–185.
- GOTOH, H., SHIBAHARA, T. and SAKAI, T. (2001b). "Sub-Particle-Scale Turbulence Model for the MPS Method—Lagrangian Flow Model for Hydraulic Engineering". *Comp. Fluid Dyn. J.* 9(4), 339–347.
- HARLOW, F.H. and WELCH, J.E. (1965). "Numerical Calculation of Time-Dependent Viscous Incompressible Flow of Fluid with Free Surface". *Phys. Fluids* 8(12), 322–329.
- HIRT, C.W. and NICHOLS, B.D. (1981). "Volume of Fluid (VOF) Method for the Dynamics of Free Boundaries". *J. Comp. Phys.* 39, 201–225.
- KOSHIZUKA, S., TAMAKO, H. and OKA, Y. (1995). "A Particle Method for Incompressible Viscous Flow with Fluid Fragmentation". *Comp. Fluid Dyn. J.* 4, 29–46.
- KOSHIZUKA, S. and OKA, Y. (1996). "Moving Particle Semi-Implicit Method for Fragmentation of Incompressible Fluid". *Nucl. Sci. Eng.* 123, 421–434.
- KOSHIZUKA, S., NOBE, A. and OKA, Y. (1998). "Numerical Analysis of Breaking Waves Using the Moving Particle Semi-Implicit Method". *Int. J. Numer. Methods Fluids* 26, 751–769.
- KOSHIZUKA, S., IKEDA, H. and OKA, Y. (1999). "Numerical Analysis of Fragmentation Mechanisms in Vapor Explosions". *Nucl. Eng. Design* 189, 423–433.
- LIN, P.Z., CHANG, K.A. and LIU PHILIP, L.F. (1999). "Runup and Rundown of Solitary Waves on Sloping Beaches". *J. Waterway Port, Coast. Ocean. Eng. ASCE* 125(5), 247–255.
- LUCY, L.B. (1977). "A Numerical Approach to the Testing of the Fission Hypothesis". *Astron. J.* 82(12), 1013–1024.
- MARTIN, J.C. and MOYCE, W.J. (1952). "An Experimental Study of the Collapse of Liquid Columns on a Rigid Horizontal Plane". *Philos. Trans. R. Soc. London Ser. A* 244, 312–324.
- MONAGHAN, J.J. and LATTANZIO, J.C. (1985). "A Refined Particle Method for Astrophysical Problems". *Astron. Astrophys.* 149, 135–143.
- MONAGHAN, J.J. (1992). "Smoothed Particle Hydrodynamics". *Annu. Rev. Astron. Astrophys.* 30, 543–574.
- MONAGHAN, J.J. and KOS, A. (1999). "Solitary Waves on a Cretan Beach". *J. Waterway, Port, Coast. and Ocean. Eng. ASCE* 125(3), 145–154.
- OGAMI, Y. (1999). "A Pure Lagrangian Method for Unsteady Compressible Viscous Flow". *Comp. Fluid Dyn. J.* 8(3), 383–392.

22. ROGALLO, R.S. and MOIN, P. (1984). "Numerical Simulation of Turbulent Flows". *Annu. Rev. Fluid Mech.* 16, 99–137.
23. SHAO, S.D., LO, E.Y.M. and WANG, G.Q. (2002). "Simulation of Fan Formation Using a Debris Mass Model". *J. Hydraul. Res.* 40(4), 425–433.
24. SHAO, S.D. and LO, E.Y.M. (2003). "Incompressible SPH Method for Simulating Newtonian and Non-Newtonian Flows with a Free Surface". *Adv. Water Resour.* 26(7), 787–800.
25. SMAGORINSKY, J. (1963). "General Circulation Experiments with the Primitive Equations, I. The Basic Experiment". *Mon. Weather Rev.* 91, 99–164.
26. SYNOLAKIS, C.E. (1986). "The Run-Up of Long Waves". Ph.D Thesis, California Institute of Technology, Pasadena, CA, USA.
27. YOON, H.Y., KOSHIZUKA, S. and OKA, Y. (1999). "A Mesh-Free Numerical Method for Direct Simulation of Gas–Liquid Phase Interface". *Nucl. Sci. Eng.* 133, 192–200.
28. ZHU, Y., FOX, P.J. and MORRIS, J.P. (1999). "A Pore-Scale Numerical Model for Flow through Porous Media". *Int. J. Numer. Anal. Methods. Geomech.* 23, 881–904.