

Numerical modelling of extreme waves by Smoothed Particle Hydrodynamics

M. H. Dao¹, H. Xu¹, E. S. Chan², and P. Tkalich¹

¹Tropical Marine Science Institute, National University of Singapore, Singapore

²Department of Civil Engineering, National University of Singapore, Singapore

Received: 31 August 2010 – Revised: 5 January 2011 – Accepted: 5 January 2011 – Published: 11 February 2011

Abstract. The impact of extreme/rogue waves can lead to serious damage of vessels as well as marine and coastal structures. Such extreme waves in deep water are characterized by steep wave fronts and an energetic wave crest. The process of wave breaking is highly complex and, apart from the general knowledge that impact loadings are highly impulsive, the dynamics of the breaking and impact are still poorly understood. Using an advanced numerical method, the Smoothed Particle Hydrodynamics enhanced with parallel computing is able to reproduce well the extreme waves and their breaking process. Once the waves and their breaking process are modelled successfully, the dynamics of the breaking and the characteristics of their impact on offshore structures could be studied. The computational methodology and numerical results are presented in this paper.

1 Introduction

The breaking dynamics and impact loading of ocean waves are among the most important factors in the design of marine and coastal structures. Past researches have shown that extreme breaking waves, especially the plunging type, are the most dangerous. Despite their low frequency of occurrence, extreme waves have caused significant loss of property and human lives. Some of the largest incidents involving supercarriers between 1969 and 1994, where rogue waves were suspected, were listed by Kharif and Pelinovsky (2003). Recently, the Category 5 hurricanes Katrina and Rita hit the Gulf of Mexico resulting in more than 3000 oil and gas production platforms in the Gulf being directly impacted and 113 platforms being destroyed (<http://www.boemre.gov/ooc/press/2006/press0501.htm>).

The basic mechanisms of the extreme/rogue wave generation were discussed in the review article by Kharif and Pelinovsky (2003). Extreme waves result from excessive energy concentration over a small domain in space and time due to or reinforced by several factors, such as wind, opposing current, frequency/directional focusing, or bottom topography. The nonlinearity and instability in the evolution of the waves could also lead to or influence the self-focusing. These mechanisms have been tried in numerical and experimental studies to generate breaking waves. Chan and Melville (1988), Kway et al. (1998) used frequency focusing to generate strong plungers in numerical simulations and laboratory experiments. In Fochesato et al. (2007) the directional focusing method was used to simulate a rogue wave generation in three-dimensional space. Guyenne and Grilli (2006) simulated shoaling and overturning waves in shallow water. The nonlinear evolution of a large-amplitude sinusoidal wave led to a powerful plunger in the numerical simulation of Lubin et al. (2003). Apart from the generation mechanism, extreme wave breaking is known to entrain large amount of air bubbles and generate current, vorticity and turbulence in the water column and exert highly impulsive impact loadings (Peregrine, 1983; Chan and Melville, 1988; Bonmarin, 1989). However, the kinematics and dynamics of these waves are still poorly understood.

Our study aims to gain a deeper understanding of general wave breaking phenomena including that of extreme/rogue waves. To supplement laboratory experiments which could be costly and are limited in resolution and reliability of measured data in bubbly and turbulent flow areas, we have chosen to conduct a numerical study. In this paper, we present our first step of the study, which is the development of a robust algorithm and a numerical code in order to perform high resolution simulations of the whole breaking process.

The numerical model used in this study was developed from the SPHysics code v1.0, which is based on the Smoothed Particle Hydrodynamics (SPH) method for



Correspondence to: M. H. Dao
(tmsdmh@nus.edu.sg)

general single-phase flows (Gómez-Gesteira et al., 2007). Major enhancements to the original SPH code have been made. These include the code parallelization, rewriting of the SPH formulation for air-water simulation, kernel correction, and implementation of the “ghost particle” boundary condition. SPH has been widely used by various researchers in the past to simulate wave breaking (Gotoh et al., 2004; Dalrymple and Rogers, 2006; Gotoh and Sakai, 2006; Khayyer et al., 2008). However, most of these studies were conducted at low resolution and single-phase flow. Our preliminary simulations of wave breaking problems have shown that the dynamics of air plays an important role in capturing the correct physics of the wave breaking processes. The two important processes of wave breaking have been described in review papers by Peregrine (1983) and Bonmarin (1989), which include the formations and collapses of the entrapped air pocket and splash-up during the breaking of a plunger, but were not fully captured in the SPH simulations in the past. These processes have much smaller time and length scales than those of wave propagation. Significant improvements in capturing these processes have been observed when the temporal and spatial resolutions are increased. Therefore, we are improving the SPH model for high resolution air-water simulations of breaking waves to capture these fine features.

In this paper, we present the methodology used to simulate a high resolution air-water breaking wave. The frequency focusing method of Chan and Melville (1988) and Kway et al. (1998) is used to generate well-controlled breaking waves in 2-D water flumes. This is a simplification of the wave-wave interaction mechanism that generates breaking waves in real oceans. Obtained results of the breaking processes of a plunging wave are presented and discussed.

2 Numerical methodology

2.1 Governing equations

The two-dimensional surface water wave problem is governed by the Navier-Stokes equations:

$$\frac{d\rho}{dt} = -\rho \nabla \cdot \mathbf{u} \quad (1)$$

$$\frac{d\mathbf{u}}{dt} = -\frac{1}{\rho} \nabla p + \left(\frac{1}{\rho} \nabla \cdot \mu \nabla \right) \mathbf{u} + \mathbf{g} + \nabla \cdot \boldsymbol{\tau} \quad (2)$$

Where ρ , p and $\mathbf{u} = (u, w)$ are the density, pressure and velocity of the fluids (water and air, in particular); $\mathbf{g} = (g_x, g_z)$ is the gravitational acceleration; μ is kinetic viscosity and $\boldsymbol{\tau}$ is turbulence shear stress. The effect of surface tension could be neglected (Peregrine, 1983).

2.2 SPH approximation to the governing equations

The SPH method was first proposed by two independent works of Gingold and Monaghan (1977) and Lucy (1977).

This method allows a function to be estimated at a location in a domain by an integration of the function itself weighted by a kernel function over the whole domain. This expression has the form of

$$A(\mathbf{x}) = \int_{\Omega} A(\mathbf{x}') W(\mathbf{x} - \mathbf{x}', h) d\mathbf{x}' \quad (3)$$

where $W(\mathbf{x} - \mathbf{x}', h)$ is the kernel function; h is smoothing length which is a parameter whose value is in order of the distance between two neighbouring points. The integration is taken over the entire computation domain Ω . The kernel function must satisfy the following two important properties

$$\int_{\Omega} W(\mathbf{x} - \mathbf{x}', h) d\mathbf{x}' = 1 \quad (4)$$

$$\lim_{h \rightarrow 0} W(\mathbf{x} - \mathbf{x}', h) = \delta(\mathbf{x} - \mathbf{x}') \quad (5)$$

Other properties of the kernel function might be required for better accuracy and efficiency of the kernel approximation (Monaghan, 1992, 2005; Liu and Liu, 2003). These properties are:

1. The kernel function has a compact support (or compact support domain), i.e. $W(\mathbf{x} - \mathbf{x}', h) = 0$ for $|\mathbf{x} - \mathbf{x}'| > kh$, where k is a parameter which determines the spread of the kernel function. This property transforms a kernel approximation from involving integration of the whole computation domain to a local domain which has much smaller area. As a result, computational effort is significantly reduced.
2. The kernel function is non-negative in the support domain. It is not strictly required, but important to ensure a meaningful representation of some physical phenomena.
3. The kernel function is monotonically decreasing with the increase of the distance away from the center. This property means that with the increase of the distance from its center, the contribution to the integral is smaller.
4. The kernel function is symmetric. This property means that locations that have the same distance to the center will have the same contributions to the integral.
5. The kernel function is sufficiently smooth. When used in approximation, smoother kernel function and derivatives would usually yield better results.

The estimate of the function $A(\mathbf{x})$ could be differentiated exactly provided the kernel function is differentiable. Different kernel functions could be used (Monaghan, 2005; Gómez-Gesteira et al., 2007). In this study the two-dimensional 5th order polynomial function, as shown in Eq. (6), is used. From

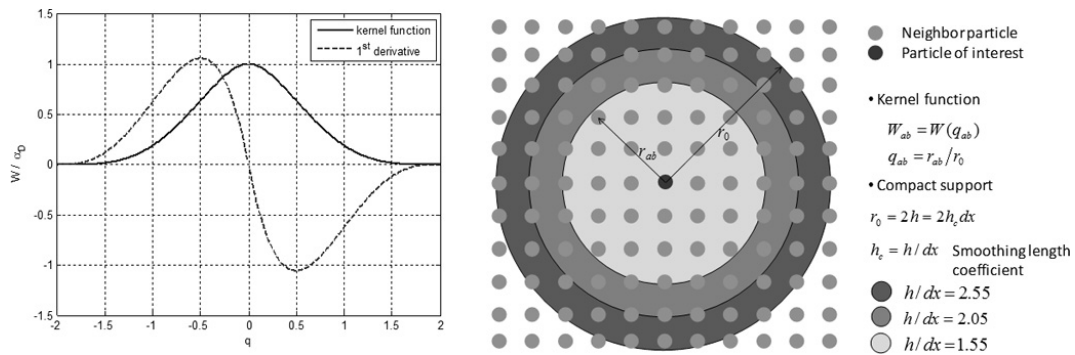


Fig. 1. The 5th order kernel function and its 1st derivative (left) and different compact support domains (right).

our preliminary simulations, adequate results are obtained at high resolution simulations if the 5th order polynomial function is used. The function and its derivative are shown in Fig. 1.

$$W(q) = \alpha_D \left(1 - \frac{q}{2}\right)^4 (2q + 1),$$

$$q = (\mathbf{x} - \mathbf{x}_b)/h; 0 \leq q \leq 2; \quad \alpha_D = 7/(4\pi h^2) \quad (6)$$

In discrete fluid problems, the fluid domain is represented by a set of N particles interacting with each other. A particle having an index b carries the basic properties: position \mathbf{x}_b , velocity \mathbf{u}_b , mass m_b , density ρ_b , pressure p_b , kinetic viscosity μ_b and turbulence shear stress $\boldsymbol{\tau}_b$. Using the kernel function, the integration (3) is replaced by a summation

$$A(\mathbf{x}) = \sum_b \frac{m_b}{\rho_b} A_b W(\mathbf{x} - \mathbf{x}_b, h) = \sum_b \frac{m_b}{\rho_b} A_b W_{ab} \quad (7)$$

where A_b is the value of function $A(\mathbf{x})$ at location \mathbf{x}_b and $W_{ab} = W(\mathbf{x}_a - \mathbf{x}_b, h)$. The subscript a or b denote the particle where the values of the parameters or functions are taken.

Using the method, the governing equations (the approximation of viscosity and turbulence terms will be shown separately) are approximated as

$$\frac{d\rho_a}{dt} = -\rho_a \sum_b \frac{m_b}{\rho_b} (\mathbf{u}_b - \mathbf{u}_a) \cdot \nabla_a W_{ab} \quad (8)$$

$$\frac{d\mathbf{u}_a}{dt} = -\sum_b m_b \left(\frac{p_a}{\rho_a \rho_b} + \frac{p_b}{\rho_a \rho_b} \right) \nabla_a W_{ab} + \mathbf{g} \quad (9)$$

$$\frac{d\mathbf{x}_a}{dt} = \mathbf{u}_a \quad (10)$$

These formulas were used in Colagrossi and Landrini (2003) for simulations of air-water flows. In this formulation, water and air are treated as compressible fluids. An equation of state (Batchelor, 1967) is used to relate the density and pressure of each fluid:

$$p(\rho) = B \left[\left(\frac{\rho}{\rho_o} \right)^\gamma - 1 \right] + p_o \quad (11)$$

where, p_o is atmospheric pressure and usually set zero, ρ_o is reference density of the fluids. In this study $\rho_o = 997 \text{ kg m}^{-3}$ for water and $\rho_o = 1.2 \text{ kg m}^{-3}$ for air are used. γ is a constant for each fluid ($\gamma = 7$ for water and $\gamma = 1.4$ for air).

The coefficient B in the equation of state is used to control the compressibility of the fluid. It is a function of the compressibility and density of the fluid. The derivation of B is as follows. The density variation is proportional to square of the Mach number:

$$\frac{|\delta\rho|}{\rho} \propto \left(\frac{u_{\max}}{c_s} \right)^2 \leq \delta_{\max} \Leftrightarrow \frac{\rho_o u_{\max}^2}{B\gamma} \leq \delta_{\max} \quad (12)$$

where u_{\max} is the maximum speed of fluid, δ_{\max} is the maximum density variation and c_s is the sound speed in the fluid which is computed as

$$c_s = \sqrt{\left. \frac{\partial p}{\partial \rho} \right|_{\rho_o}} = \sqrt{\frac{B\gamma}{\rho_o}} \quad (13)$$

From Eqs. (12) and (13), we have

$$B \geq \frac{\rho_o u_{\max}^2}{\delta_{\max} \gamma} \quad (14)$$

Given the actual sound speed in water, the density variation, δ_{\max} , is very small and thus a very large value of B is required. However, the computational time step will be extremely small since it is inversely proportional to the sound speed as shown in Eq. (17). This small time step will deteriorate the efficiency of the computation. Hence, in actual simulations, the value of B is chosen so that the density variation and sound speed are within desired ranges. This sound speed used in the computation is called artificial sound speed and is not necessary equal to the actual sound speed. Similarly, an artificial sound speed is also used for the simulation of air. In this study, artificial sound speeds of 20 m s^{-1} and 40 m s^{-1} are used for water and air, respectively.

The approximation of the viscosity term has the form of

$$\left[\left(\frac{1}{\rho} \nabla \cdot \mu \nabla \right) \mathbf{u} \right]_a = \sum_b \frac{m_b}{\rho_a \rho_b} \frac{(\mu_a + \mu_b) \mathbf{r}_{ab} \cdot \mathbf{u}_{ab}}{r_{ab}^2 + 0.01h^2} \cdot [\nabla W_{ab}]_a,$$

$$\mathbf{r}_{ab} = \mathbf{x}_b - \mathbf{x}_a, \quad \mathbf{u}_{ab} = \mathbf{u}_b - \mathbf{u}_a \quad (15)$$

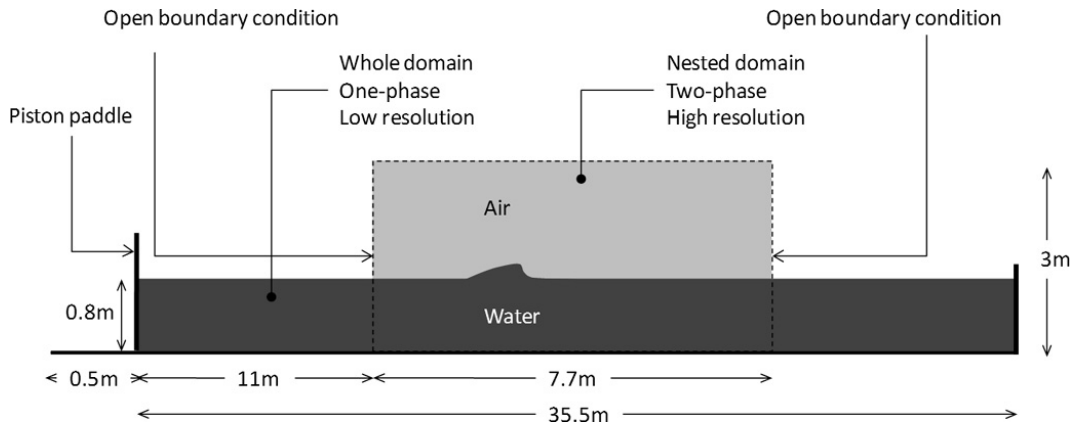


Fig. 2. Numerical setup for simulations of generation and breaking of an extreme wave.

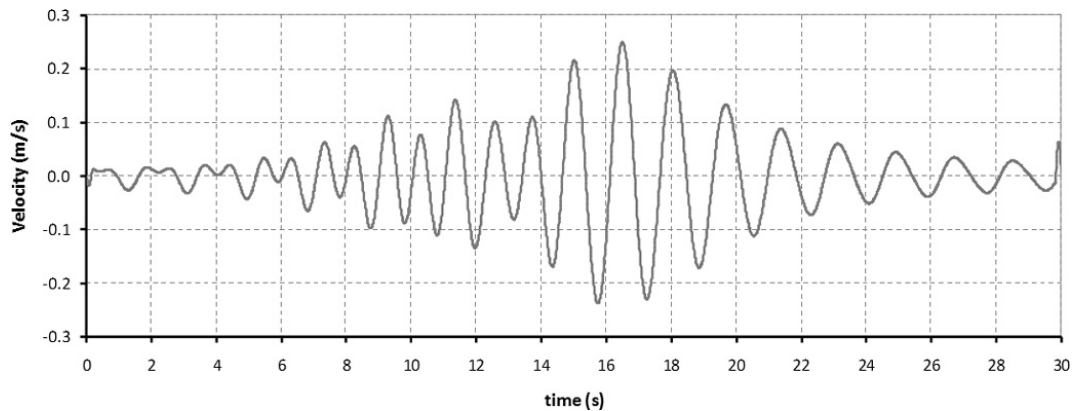


Fig. 3. Velocity of the wave paddle derived from the experiment.

The turbulence model in Gotoh et al. (2004) has been used. The model uses the concept of Large Eddy Simulation where the large scale turbulence (particle scale – PS) is computed directly in the numerical simulation and the small scale (sub-particle scale – SPS) stresses are modelled using a subgrid-scale model (SGS). The SPH expression of the SPS term in two-dimensional space (x, z) is

$$\left(\frac{1}{\rho} \nabla \cdot \boldsymbol{\tau}\right)_a = \sum_b m_b \left(\frac{\boldsymbol{\tau}_a + \boldsymbol{\tau}_b}{\rho_a \rho_b}\right) \cdot \nabla_a W_{ab} \quad (16)$$

where

$$(\nabla \cdot \boldsymbol{\tau})_x = \frac{\partial}{\partial x} \tau_{xx} + \frac{\partial}{\partial z} \tau_{xz}; \quad (\nabla \cdot \boldsymbol{\tau})_z = \frac{\partial}{\partial x} \tau_{xz} + \frac{\partial}{\partial z} \tau_{zz}$$

$$\tau_{xx} = 2\rho\nu_t \frac{\partial u}{\partial x} - \frac{2}{3}\rho k; \quad \tau_{zz} = 2\rho\nu_t \frac{\partial w}{\partial z} - \frac{2}{3}\rho k;$$

$$\tau_{xz} = \rho\nu_t \left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x}\right)$$

$$\nu_t = (C_s \Delta)^2 P_r^{1/2}; \quad k = \frac{C_v}{C_\epsilon} \Delta^2 P_r$$

$$P_r = 2\left(\frac{\partial u}{\partial x}\right)^2 + 2\left(\frac{\partial w}{\partial z}\right)^2 + \left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x}\right)^2$$

Here, Δ is the filter width which is equal to the initial particle spacing. The constants used by Gotoh et al. (2004) are chosen as $C_s = 0.15$, $C_v = 1.0$, and $C_\epsilon = 0.08$.

The computational time-step is controlled by the Courant-Friedrichs-Lewy (CFL) condition. The CFL condition requires the distance that a fluid particle moves in one time step to be less than a smoothing length. In this study, the following formulation is used

$$\Delta t \leq \frac{h}{u_{\max}}; \quad u_{\max} = 0.2 \max_a(c_s) \quad (17)$$

2.3 Solid boundary condition

The ghost particle method is used to approximate a solid boundary. A similar method was used in Colagrossi and Landrini (2003). In this method, a solid boundary is represented by a straight line or a curve. The outer region of the solid boundary is filled by a set of so-called “ghost” particles. Essentially, the ghost particles are the “image” or the “mirror”

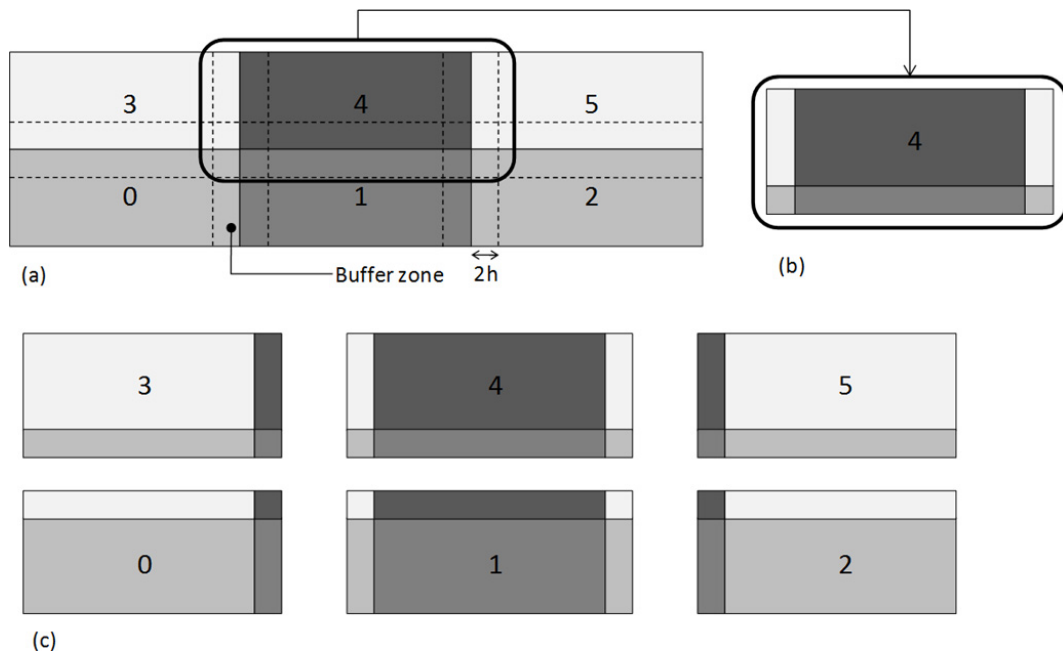


Fig. 4. Domain decomposition and buffer zones for data exchange.

of the fluid particles on the other side of the solid boundary through the boundary itself. The position of the ghost particles, a_G , is updated at every time step according to the positions of the boundary and fluid particles a

$$\mathbf{x}_{a_G} = 2\mathbf{x}_B - \mathbf{x}_a \quad (18)$$

where \mathbf{x}_B is the mirror point on the boundary.

In Colagrossi and Landrini (2003), the pressure and density of ghost particles are set equal to that of the fluid counterparts. In this application, the pressure of the ghost particle a_G is extrapolated from a fluid particle a using the hydrostatic hypothesis. The density of the particle a_G is then calculated from its pressure using the inverse of the equation of state. The equations are as follows:

$$p_{a_G} = p_a + \rho_a \mathbf{g}(\mathbf{x}_{a_G} - \mathbf{x}_a) \quad (19)$$

$$\rho_{a_G} = (\rho_o)_a \left(\frac{p_{a_G} - p_o}{B_a} + 1 \right)^{\frac{1}{\gamma_a}} \quad (20)$$

The velocity of the ghost particle a_G is calculated from its physical counterpart a , depending on the slip condition applied at the boundary. The equations are

$$\begin{aligned} u_{na_G} &= 2u_{nB} - u_{na} \\ u_{ta_G} &= (1 - \alpha_s)u_{tB} + \alpha_s u_{ta} \end{aligned} \quad (21)$$

Here, the subscripts “n” and “t” denote the normal and tangential velocity components with respect to the instantaneous position of the boundary; u_{nB} and u_{tB} are the normal and tangential velocity components at the mirror point \mathbf{x}_B ; α_s is the

slip coefficient, $\alpha_s = 1$ for free-slip condition and $\alpha_s = -1$ for non-slip condition. An intermediate slip condition is modelled by a value of α_s between -1 and 1 .

3 Numerical simulations of extreme wave breaking

3.1 Numerical setups

The setup of a two-dimensional numerical flume is shown in Fig. 2. The full dimension and setup of the actual laboratory experiment by Kway et al. (1998) have been used. The length of the flume is 36 m and the water depth is 0.8 m. The bottom and the right wall are fixed, smooth solid walls. A piston paddle is located on the left. The paddle is able to move horizontally according to the input signal which is either its horizontal velocity or position. In the SPH program, the piston paddle is modelled as a moving solid wall.

The frequency focusing method of Chan and Melville (1988) and Kway et al. (1998) was used to generate well-controlled breaking waves in 2-D water flumes. Based on this method, a modulated wave packet was generated by summing up sinusoidal wave components of discrete frequencies. Due to the frequency dispersion, different wave components in the wave packet propagated at different speeds. The wave-wave interaction could have led to a build up of a big and steep wave that would break. The height of the breaking wave is then further controlled through a gain factor to generate a desired breaking wave.

In this study, the wave packet components of constant steepness (i.e. constant $a_c k_c$, where a_c is the amplitude and

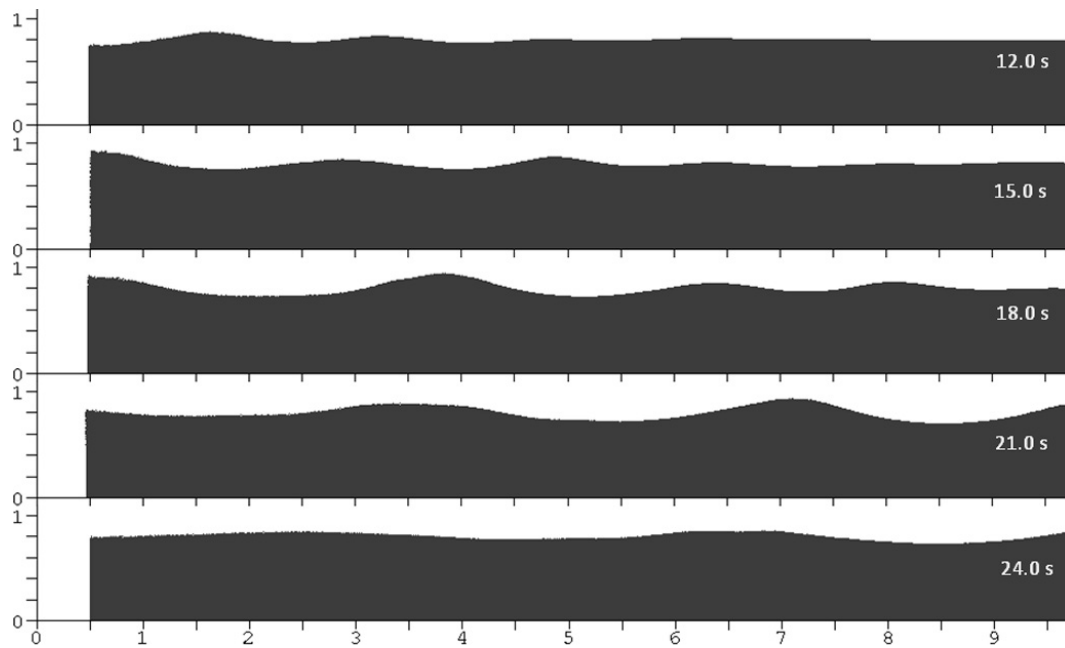


Fig. 5. Wave components being generated near the paddle.

k_c is the wave number of the central wave component of the packet) of Kway et al. (1998) is employed. The wave packet is comprised of 28 wave frequencies ranging from 0.56 Hz to 1.1 Hz and was designed so that it will focus and create a strong plunger at a location approximately 15.2 m from the initial position of the paddle. The paddle velocity derived from the voltage signal input to the paddle in the experiment is shown in Fig. 3. For the details of the experiment, readers should refer to the paper by Kway et al. (1998).

A two-stage simulation approach is used. At the first stage, the SPH simulation is carried out for 30 s of wave evolution in the whole wave flume. The particle size, time step and smoothing length are 0.005 m, 1×10^{-5} s and $h/dx = 1.55$, respectively. The dynamics of air is not of importance during the propagation of the wave package; thus it is excluded to reduce the computational cost. Once the waves have focused, the second stage is carried out. A nested domain will be extracted and used in a separate simulation which has a higher resolution. The simulation of the nested domain is carried out for 2 s from the time of wave focusing. The particle size of 0.0025 m, time step of 2.5×10^{-6} s and smoothing length of $h/dx = 2.1$ are used.

The nested domain is chosen as shown as the dash box in Fig. 2. The initial conditions of density, pressure, velocity of the wave are interpolated from stage 1 simulation at the focused time. The area above the water surface is also filled up by a layer of air. Initial gauge pressure and velocity of the air layer are set to zero. A solid wall is applied at the top of the domain and a periodic boundary condition is applied at the two lateral boundaries of the domain. The air layer is chosen to be about three times thicker than the water layer to remove the effect of the ceiling on the wave breaking process.

There will be 1 136 000 water particles involved in the first stage and 3 673 200 particles (966 185 water particles) used in the second stage. Due to the huge number of particles used, domain decomposition and parallel computation are required.

3.2 Domain decomposition and parallel computation

The SPH code is parallelized to deal with the extensive problems. The computational domain is partitioned into sub-domains and the Message Passing Interface (William et al., 1994) is used to exchange data among the sub-domains. The domain decomposition and data exchange are described as follows.

The original computational domain is decomposed into several sub-domains; each will be handled by a separate processor. For a demonstration, we assume that the original domain is divided into 6 sub-domains (shown as colour coded boxes in Fig. 4a). A sub-domain is required to exchange data with its adjacent sub-domains, e.g. domain #4 exchanges data with domains #0, 1, 2, 3, 5.

To serve for the data exchange, small zones at the interface between sub-domains are created. These zones are called buffer zones and will have widths of $2h$ (which is equal to the radius of the compact support domain). A copy of particles which lie inside a buffer zone that shares the same edge or corner with a sub-domain is added into the particle list of that sub-domain to create an extended sub-domain (as shown in Fig. 4a–b). In Fig. 4b, small colour coded boxes surrounding the sub-domain #4 have been received from adjacent sub-domains with the same colour codes.

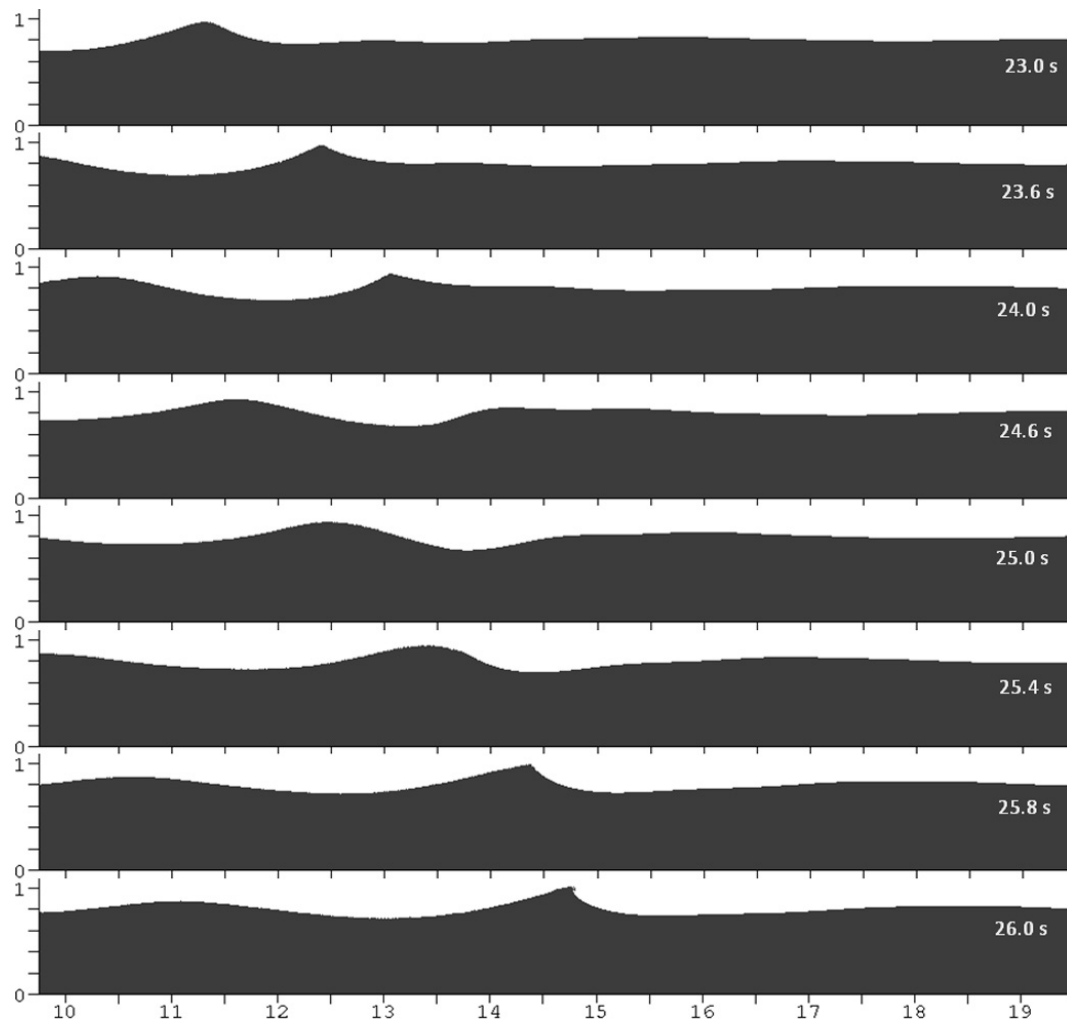


Fig. 6. Wave packet evolution and focusing.

At each time step, each sub-domain will send necessary data to its neighbour and receive data from them to form extended sub-domains (Fig. 4c). A normal SPH calculation is performed for all particles in each extended sub-domain in a separate processor.

Efficiency of the parallel computation could be maximized by balancing the computational load and communicational load (for data exchange among processors). The communication time among processors is long if the data required for exchange is large. This communication time sometimes dominates the pure computation time. Thus we expect the speedup of a small problem and a large problem could be significantly different even the same program is used. Preliminary simulations of the two-phase wave breaking problem were performed to evaluate the parallel algorithms and estimate the required resources. The size of the testing problem is as below

- Number of particles: 3 673 200

- Number of step: 1000

The test was done on 25, 50, 75, 90 processors in a Linux cluster whose specifications are given as bellow:

- 23 nodes, 4 processors/node.

- Processor: Intel(R) Core(TM) i7 CPU 940 @ 2.93 GHz

- Memory (RAM): 12 GB

- Network card: Myri-10G PCIe NIC with MX

The number of processors, the average number of particles per processor, the time to finish 1000 steps of simulation and speedup ratios are shown in Table 1. Since the whole set of particles is too large to be simulated in a single processor, the 25-processor run will serve as a benchmarking point with a speedup ratio equal to 1. The SPH program shows a very good speedup compared to the ideal.

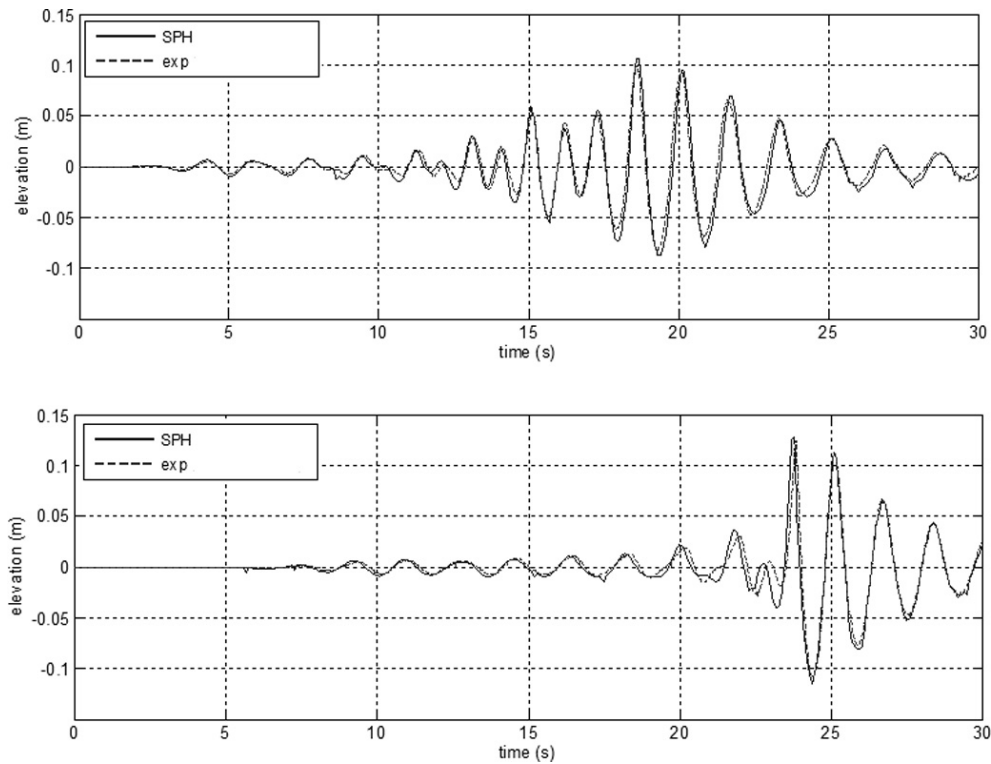


Fig. 7. Comparisons of simulated and measured water elevations at two locations 4.5 m (upper pane) and 12 m (lower pane) from the mean paddle position.

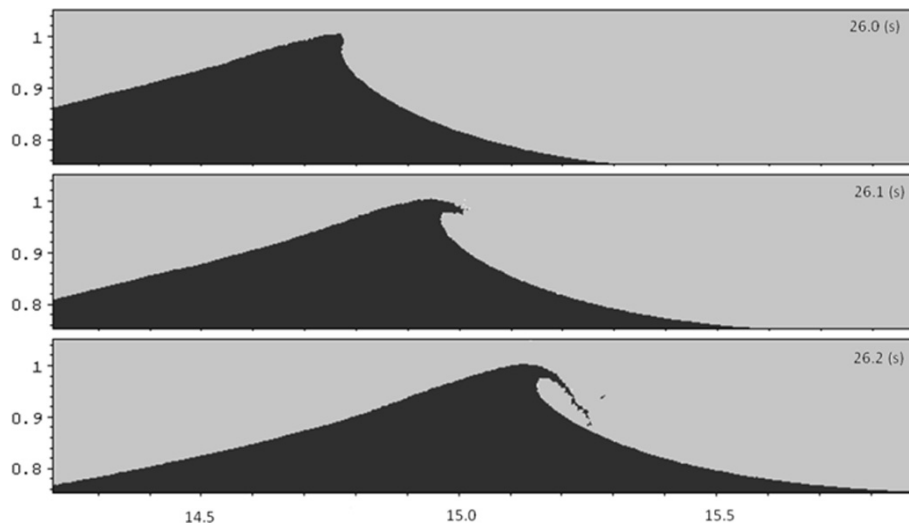


Fig. 8. The focused wave starts to break creating a plunger with a curling over plunging jet.

4 Numerical results

4.1 Wave packet propagation and focusing

As shown in Fig. 5, the wave packet was sent out from the moving paddle. Shorter waves with small amplitude were generated first, followed by longer and large ampli-

tude waves. Due to frequency dispersion, longer waves in the wave packet travel faster than short waves and will catch up with the components ahead. As shown in Fig. 6, the wave packet was modulated, focused and started to break at a location of around 14.6 m, which is 14.1 m from the paddle mean position. The time of wave breaking is 26 s. The wave height

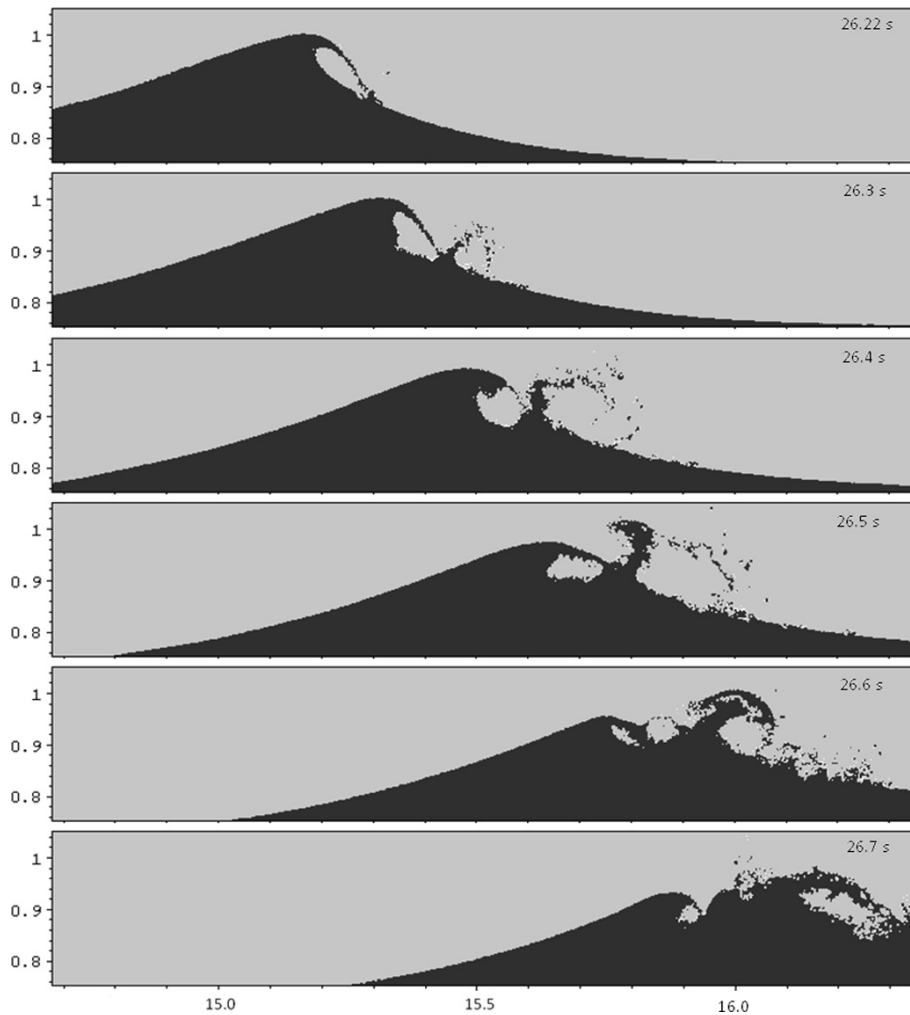


Fig. 9. Impingement of the plunging jet entraps an air pocket and creates a strong splash-up.

Table 1. Computation time and speedup ratio versus number of processors.

No. of proc	Particles/1 proc	Comp. time/1000 steps (min)	Speedup	Ideal speedup
25	147 000	13.065	1	1
50	73 000	6.495	2.01	2
75	49 000	4.71	2.77	3
90	41 000	4.178	3.13	3.6

of the breaking wave measured from its crest to its lowest trough ahead is 0.2 m. The distance between two troughs is around 2.5 m. These values of wave height and breaking location agree very well with the wave height of 0.22 m and breaking locations of 14 m reported in Kway et al. (1998). This simulated breaking wave is equivalent to a wave of 20 m high and 250 m long at 80 m water depth.

The time series of water elevation extracted at two locations (4.5 m and 12 m from the paddle mean position)

are verified against the laboratory experiments by Kway et al. (1998). Figure 7 shows that the time series of simulated water elevations at the two locations agree very well with those measured in the experiment. Quantification of the deviation is not done as the inputs to the wave paddle in the experiment and the numerical simulation are not exactly the same.

Considering an SPH simulation of an intermediate water depth wave group over a period of about $22T_c$ ($T_c = 1.2$ s is

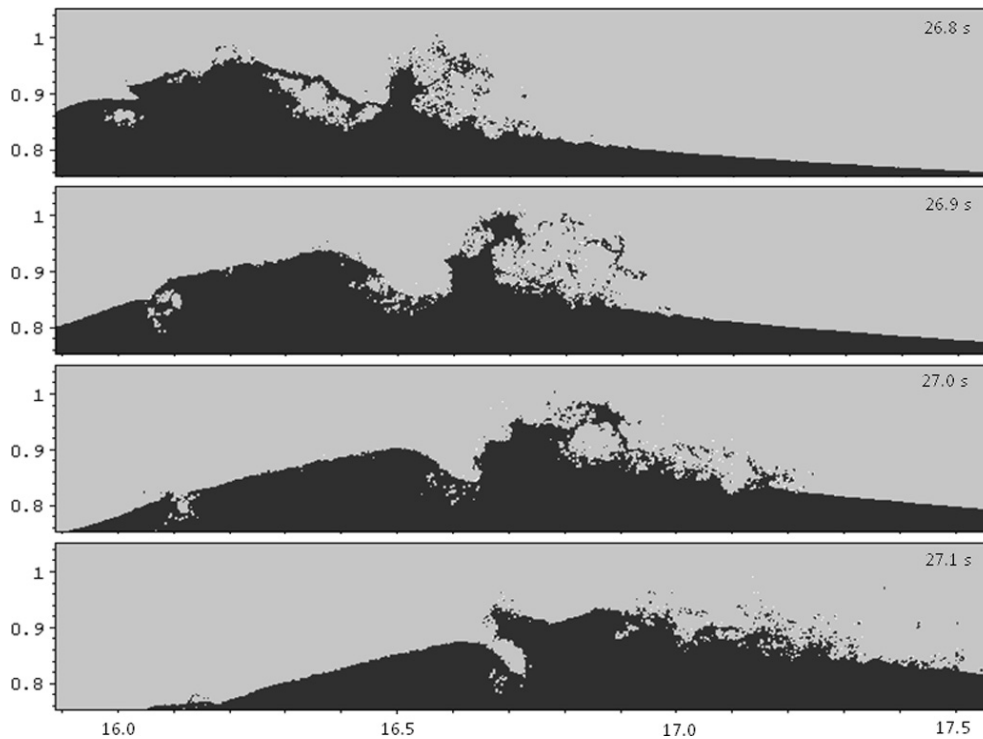


Fig. 10. The wave is collapsing. The jet from the splash-up creates another splash-up ahead.

the wave period of the central frequency component), good agreement of the numerical results with the experiment indicate that the numerical model has achieved an adequate numerical dissipation rate.

4.2 Sequence of the wave breaking

After focusing, a wave starts to break. The details of the breaking process were captured in the nested domain with the higher resolution and inclusion of the air dynamics in the second stage. The sequence of the wave breaking snapshots is presented in Figs. 8–10.

As shown in Fig. 8, the front surface of the wave crest is steepening. A jet is projecting forwards from the wave crest, curling over and about to impinge on the water surface in the front. In Fig. 9, the plunging jet impinges on the water surface in the front, entraps an air pocket. The entrapped air pocket changes its shape during the wave breaking. Initially, it has an inclined elongated ellipsoid shape ($t = 26.22$ s). It becomes thicker and eventually collapses.

The impingement of the plunging tip on the water front generates a powerful water jet in the front which is called a splash-up (as shown in Fig. 9). This splash-up rises as high as the wave crest and is projecting forwards. As shown in Fig. 10, the splash-up is collapsing. Large amount of water from the splash-up fall down and impinge on the water surface ahead of it and create violent water fragmentation. The impingement of the splash-up also creates another smaller

splash-up ahead. Simultaneously, the rear part of the splash-up falls on the wave crest behind it, entrapping an additional air pocket. The entrapped air pockets are broken into many bubbles. Larger bubbles quickly resurface and burst off while smaller bubbles reside longer in the water column.

Air dynamics play an important role in the breaking process. Air entrapment under the plunging jet provides additional force to maintain the existence of the pocket longer. The drag force from the air acting on the front face of the splash-up contributes to its being built up vertically and reversely breaking on the wave crest, entraining a significant amount of air into the water column. A simulation of a single-phase breaking wave has showed that the pocket under the plunging jet collapses earlier and the splash-up projects forward without reversed breaking on the wave crest. As discussed in Bonmarin (1989), these processes are crucial for the generation of bubbles in the water column which are connected to the heat and mass transfer across the sea surface. These breaking processes takes place within 1 s which is very short compared to 26 s of the entire wave evolution.

5 Conclusion

The paper has presented the numerical methodology for simulations of extreme wave breaking. The methodology is based on the Smoothed Particle Hydrodynamics method, depicting well wave frequency focusing. Multi-scale nesting

and parallel computing have been employed to conduct extensive simulations. Simulated results have shown that the developed methodology successfully simulates the whole process of generation and breaking of an extreme wave. This work will be used for further studies on the kinematics and dynamics of wave breaking and its impact on offshore structures.

Edited by: E. Pelinovsky

Reviewed by: H. B. Branger and another anonymous referee

References

- Batchelor, G. K.: An Introduction to Fluid Dynamics, Cambridge University Press, ISBN 0521663962, 1967.
- Bonmarin, P.: Geometric properties of deep-water breaking waves, *J. Fluid Mech.*, 209, 405–433, 1989.
- Chan, E. S. and Melville, W. K.: Deep water plunging wave pressures on a vertical plane wall, *Proceeding of the Royal Society*, A417, 95–131, 1988.
- Colagrossi, A. and Landrini, M.: Numerical simulation of interfacial flows by Smoothed Particle Hydrodynamics, *J. Comput. Phys.*, 191, 448–475, 2003.
- Dalrymple, R. A. and Rogers, B. D.: Numerical modeling of water waves with the SPH method, *Coast. Eng.*, 53(2–3), 141–147, 2006.
- Fochesato, C., Grilli, S., and Dias, F.: Numerical modeling of extreme rogue waves generated by directional energy focusing, *Wave Motion*, 44(2007), 395–416, 2007.
- Gingold, R. A. and Monaghan, J. J.: Smoothed particle hydrodynamics: theory and application to non-spherical stars, *Mon. Not. R. Astron. Soc.*, 181, 375–389, 1977.
- Gómez-Gesteira, M., Rogers, B. D., Dalrymple, R. A., Crespo, A. J. C., and Narayanaswamy, M.: User Guide for the SPHysics Code v1.0, <http://wiki.manchester.ac.uk/sphysics>, 2007.
- Gotoh, H. and Sakai, T.: Key issues in the particle method for computation of wave breaking, *Coast. Eng. J.*, 53, 171–179, 2006.
- Gotoh, H., Shao, S. D., and Memita, T.: SPH-LES model for numerical investigation of wave interaction with partially immersed breakwater, *Coast. Eng. J.*, 46(1), 39–63, 2004.
- Guyenne, P. and Grilli, S.: Numerical study of three-dimensional overturning waves in shallow water, *J. Fluid Mech.*, 547, 361–388, 2006.
- Kharif, C. and Pelinovsky, E.: Physical mechanisms of the rogue wave phenomenon, *Eur. J. Mech. B-Fluid*, 22(2003), 603–634, 2003.
- Khayyer, A., Gotoh, H., and Shao, S. D.: Corrected Incompressible SPH method for accurate water-surface tracking in breaking waves, *Coast. Eng.*, 55, 236–250, 2008.
- Kway, J. H. L., Loh, Y. S., and Chan, E. S.: Laboratory study of deep-water breaking waves, *Ocean Eng.*, 25(8), 657–676, 1998.
- Liu, G. R. and Liu, M. B.: Smoothed particle hydrodynamics: a meshfree particle method, World Scientific, 2003.
- Lubin, P., Vincent, S., Abadie, S., and Caltagirone, J.-P.: Three-dimensional Large Eddy Simulation of air entrainment under plunging breaking waves, *Coast. Eng.*, 53, 631–655, 2003.
- Lucy, L. B.: A numerical approach to the testing of the fission hypothesis, *Astron. J.*, 82, 1013–1024, 1977.
- Monaghan, J. J.: Smoothed Particle Hydrodynamics, *Annu. Rev. Astron. Astr.*, 30, 543–574, 1992.
- Monaghan, J. J.: Smoothed Particle Hydrodynamics, *Rep. Prog. Phys.*, 68, 1703–1759, 2005.
- Peregrine, D. H.: Breaking waves on beaches, *Annu. Rev. Fluid Mech.*, 15, 149–178, 1983.
- SPHysics code v1.0, <http://wiki.manchester.ac.uk/sphysics>, last accessed August 2007.
- William, G., Ewing, L., and Anthony, S.: Using MPI: portable parallel programming with the message-passing interface, Cambridge, MA, USA: MIT Press Scientific and Engineering Computation Series, ISBN 0-262-57104-8., 1994.