Computers & Fluids 67 (2012) 138–151

Contents lists available at SciVerse ScienceDirect

Computers & Fluids

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

Incompressible SPH for free surface flows

Arne Bøckmann [a,*], Olga Shipilova [b], Geir Skeie [b]

[a] University of Oslo (UiO), Department of Mathematics, PO Box 1053, Blindern, NO-0316 Oslo, Norway
[b] Det Norske Veritas (DNV), Veritasveien 1, 1363 Høvik, Norway

ARTICLE INFO

Article history:
Received 1 September 2010
Received in revised form 12 June 2012
Accepted 3 July 2012
Available online 3 August 2012

Keywords:
Incompressible free surface flow
Smoothed particle hydrodynamics
Projection method
Marine applications

ABSTRACT

Violent free surface flows exist in a number of physical processes related to marine engineering. Traditionally, these are simulated using mesh-based methods formulated in an Eulerian frame of reference. Lagrangian meshless and particle-based methods, like Smoothed Particle Hydrodynamics (SPH), represent an interesting alternative as they circumvent several common problems related to the free surface. This work is devoted to the properties and implementation of an SPH method based on pressure projection for incompressible inviscid flows with free surfaces. Novel features include a time integration scheme designed to improve stability, and a new treatment of the free surface pressure boundary condition. The mechanisms and performance of the algorithm are addressed with a study of the evolution of an elliptical drop, and the more practical problems of rigid body water entry and internal tank sloshing. The results are compared to those of competing methods.

1. Introduction

A number of physical processes within the field of marine engineering involve violent free surface flows. Water entry problems found in bow slamming and lifeboat impact, sloshing in internal tanks and green water on deck are all associated with a free surface and pose challenges in both design and simulation.

Computational Fluid Dynamics (CFD) is a well established tool in many fields of engineering. As increased CPU power continually opens up new possibilities, the feasible engineering challenges tend to increase in complexity. This has spurred the development of many specialized numerical methods. In the case of free surface flows, a number of different techniques are usually included – in particular those that handle the changing topology of a free surface, which may cause mass loss and other spurious effects.

Eulerian and Arbitrary Lagrangian–Eulerian mesh-based methods, like the Finite Element Method (FEM), Finite Volume Method (FVM) and Finite Difference Method (FDM), supplemented with techniques to capture free surfaces have been successfully applied for certain classes of problems in fluid dynamics. Being very useful and effective for problems with moderate deformations of the free surface, these methods may fail in the case of large deformations. They often suffer from numerical diffusion and/or severe distortion of the background mesh. Over the recent decades, particle based-methods, like *Smoothed Particle Hydrodynamics (SPH)*, have gained popularity within CFD, especially within free surface problems. The popularity of SPH may stem from the fact that it is capable, yet conceptually simple. The Lagrangian frame of reference automatically captures the free surface, discretization stencils are easily updated and there is no bookkeeping associated with a mesh. It may also be argued that SPH is a natural approach to fluid simulations, as the computational nodes may be viewed as representations of fluid segments with reciprocal forces acting in between.

Originally designed for astrophysical problems [1,2], SPH has since developed to encompass many other areas of applicability such as visco-elastic fluids [3] and fracture mechanics [4]. It has been successfully coupled with other numerical techniques, for instance, with the level set method to capture foam and sprays [5], or with mesh-based FEM solvers [6]. The methods originating from smoothed particle hydrodynamics can generally be divided into two classes with respect to the formulation of incompressible flow – weakly compressible and incompressible algorithms.

Weakly Compressible SPH (WCSPH) was introduced by Monaghan in [7] and became the first successful SPH formulation of incompressible free surface flows. This method is still the most commonly used particle-based method in fluid dynamics. WCSPH introduces an artificial equation of state to couple pressure and density under the incompressibility constraint. To obtain reasonable computational efficiency, the numerical speed of sound is usually chosen much lower than the actual speed of sound, so that the Courant–Friedrichs–Lewy (CFL) condition allows for larger time steps. However, the length of the time step needed to maintain low compressibility is very short. Moreover, due to artificial pressure waves and the strong dependence on particle positions, several more or less artificial techniques are usually applied to stabilize simulations. These often employ arbitrary numerical parameters, which may impair the robustness and increase the complexity of the method.

[*] Corresponding author.
E-mail address: arnbo@math.uio.no (A. Bøckmann).

0045-7930/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.compfluid.2012.07.007

To overcome the troublesome features that plague WCSPH, the *pressure projection method* was used in [8] to enforce the incompressibility constraint. Formulated independently by Chorin [9] and Temam [10] more than 40 years ago, this method is very common in mesh-based simulations of incompressible flows. Although Cummins and Rudman referred to the method as 'Projection SPH' [8], the presently common name used in literature is 'Incompressible SPH' (ISPH), which we adopt here. A similar particle method called the Moving-Particle Semi-Implicit method (MPS), based on the Poisson equation for pressure was introduced by Koshizuka and Oka in [11]. In [12], ISPH was extended to free surface flows, where incompressibility was enforced as in the MPS method, namely, by approximating the velocity divergence by the discrete time derivative of particle volume. Later, this was shown to produce inaccurate and noisy solutions, see [13,14]. A methodology for both divergence free and density invariant SPH simulations was developed in [15]. In the limit of continuity, these two approaches are equivalent. In [16,17], iterative procedures were applied to the pressure equation to modify particle positions until the particle density is invariant. An interesting question is how the information propagates in such methods and whether numerical pressure waves are present or not.

In the current study, the implementation and properties of an incompressible SPH method for inviscid free surface flows are investigated. A novel time integration scheme is introduced, using a second order backward difference scheme in time, and linear/quadratic velocity profiles in time for the integration of particle positions. Both the position and velocity at the end of a time step are interpolated to the half-step in time. A new treatment of the free surface pressure boundary condition is also implemented.

The paper is organized as follows. Section 2 formulates the class of problems to be considered. The details of the pressure projection method in a Lagrangian frame of reference is given in Section 3. Next, the SPH discretization and fluid-body interaction implemented in this work are presented in Section 4. Section 5 is devoted to testing the method on physical problems. The proposed time integration scheme is first tested on the evolution of a 2D elliptical drop, and compared to the semi-analytical solution for the pressure, and results from similar time integration schemes. The conservation of kinetic energy is also studied. Next, the 2D problems of rigid body water entry and internal tank sloshing are simulated, which represent short-term and long-term violent flows respectively. The water entry problem reproduces the wave generator experiment presented by Monaghan and Kos [18], experimental data and WCSPH results from [18] are compared to the results obtained by ISPH. For the sloshing cases, ISPH simulations are compared with experiments, and results obtained by the WCSPH code SPHysics [19] and the Finite Volume (FV)/Volume of Fluid (VOF) implementation in OpenFOAM [20]. Conclusions are drawn in Section 6.

2. Governing equations

Inviscid flow can be described by the Euler equations in Lagrangian (material) coordinates as

$$\frac{D\mathbf{u}}{Dt} = -\frac{1}{\rho}\nabla p + \mathbf{f} \quad \text{in } \Omega \times (0,T), \tag{1}$$

$$\frac{D\rho}{Dt} = -\rho(\nabla \cdot \mathbf{u}) \quad \text{in } \Omega \times (0,T), \tag{2}$$

where $\mathbf{u} = \mathbf{u}(\mathbf{x},t), p = p(\mathbf{x},t)$ and $\rho = \rho(\mathbf{x},t)$ are velocity, pressure and density respectively, and $\mathbf{f}$ is a body force, which includes gravity and reference system accelerations. For two-dimensional flows, the space coordinates $\mathbf{x}$ are defined in the computational domain $\Omega \subset \mathbb{R}^2$ and the time t varies within the interval $(0,T)$.

The Lagrangian formulation of the momentum Eq. (1) has the advantage of eliminating the non-linear convection term $\nabla \cdot (\mathbf{u} \otimes \mathbf{u})$, which is challenging to model accurately. Moreover, by following the material elements of the flow, the free surface is tracked automatically without additional measures such as a distance function or sub-grid reconstruction.

In the present work, the flow is assumed incompressible, i.e. the mass density is constant following a material element,

$$\frac{D\rho}{Dt} = 0.$$

The velocity field should thus be divergence free,

$$\nabla \cdot \mathbf{u} = 0 \quad \text{in } \Omega \times (0,T). \tag{3}$$

The boundary of the computational domain Ω consists of solid walls, Γ_w, and the free surface, Γ_f, of the fluid. On Γ_w free-slip conditions are assumed, i.e. the flow is not allowed to permeate through the wall;

$$\mathbf{u} \cdot \mathbf{n}_w = U_n \quad \text{on} \quad \Gamma_w \times (0,T), \tag{4}$$

where $\mathbf{n}_w$ is the wall normal vector and U_n is the wall normal velocity. If the free surface is not exposed to an ambient fluid, the surface pressure is zero,

$$p(\mathbf{x},t) = 0 \quad \text{on } \Gamma_f \times (0,T). \tag{5}$$

Otherwise, if there is ambient fluid with low density and viscosity as compared to the fluid phase in question, it might be omitted from the model as long as entrainment is not significantly affecting the flow. In this case, the surface pressure can be assumed constant. Note that although Eq. (1) does not discriminate between p and $p + const$, the SPH gradient operator might, see [21].

From a physical perspective, this is a result of reciprocal forces, which are proportional to the absolute value of the pressure. Perturbing the position of a neighbor particle will thus induce a larger net force at higher pressures, and simulation results will be more sensitive to the particle configuration. On the surface, the particle distribution is particularly non-uniform, so zero pressure is always assigned, regardless of the original problem.

Eqs. (1) and (2) are supplemented with initial conditions for velocity and surface configuration.

3. The pressure projection method

The pressure projection method is used to enforce incompressibility, and relies on Helmholtz-Hodge decomposition of the velocity field into a divergence free (solenoidal) and a curl free part.

The implementation of the pressure projection method in a Lagrangian frame of reference introduces mutual dependencies between velocity, discretization points and spatial stencils that are not present in an Eulerian frame of reference. There are thus at least two new aspects that should be taken into account when designing time integration schemes.

Firstly, with moving discretization points, where all quantities (velocity, pressure, density) are stored, a discretely divergence free velocity field at time t_0 is in general not discretely divergence free at time $t > t_0$ if the points move, but the point velocities remain unchanged. The scheme should thus take into account the instantaneous mutual dependence between velocity, position and divergence, and promote consistency in time and space. It should be noted that with *approximate* projection, used in the current work, the velocity field is not completely discretely divergence free due to the formulation of the Laplace operator.

Secondly, these mutual dependencies, which also include perpetually changing discretization stencils, may introduce stability problems that are not present in an Eulerian frame of reference.

There might e.g. exist feedback loops between position, discretization stencil and velocity that destabilize simulations.

Since these aspects are not present in an Eulerian frame of reference, they are sometimes neglected. For instance, in [22], the particles were not moved to their estimated positions at the end of the time step before the divergence free projection took place. As a result, the velocity field was corrected at the particle positions at the start of each time step, causing an increasing discrete divergence when the particles were moved along straight lines during a single time step. This is a mild violation of the first aspect. Instability problems occured in the violent flow test cases when this approach was implemented in the present work.

For the second aspect, we have found few remedies in the existing literature. In [13], a proposed solution is to forcibly shift the particle positions slightly away from the streamlines, with a corresponding interpolation of properties. This apparently works by better allowing long, coherent particle filaments to fold together and avoid a complete breakdown. In the present code, such long filaments were present in the elliptical drop test case (Section 5.1) when the particles were initially placed in a rectangular configuration, but the problem was strongly reduced by changing the initial particle configuration.

The occurence of instabilities has also been reduced by taking half steps in time. That is, a full step is taken when determining the new velocity field, but the final particle positions are integrated to the point in time midway between the current and next time step, based on an interpolated velocity profile. The velocity field is also interpolated to the midpoint in time. As such, each full time step consists of two sub-steps. This approach is not the same as merely halving the length of the time step, which hardly reduces the problem, see Section 5.1.2.

The following equations describe one full time step with the above-mentioned method, assuming time steps of constant length::

1. Compute estimated nodal positions $\mathbf{x}^*$ at time instance $n+1$ by linearly extrapolating the velocity from $n-1$ and n and integrating:

$$\mathbf{x}^* = \mathbf{x}^n + \Delta t\left(\frac{3}{2}\mathbf{u}^n - \frac{1}{2}\mathbf{u}^{n-1}\right) \tag{6}$$

2. Compute the tentative velocity field $\mathbf{u}^*$ via a second order backward difference scheme in time:

$$\mathbf{u}^* = \frac{4}{3}\mathbf{u}^n - \frac{1}{3}\mathbf{u}^{n-1} + \frac{2}{3}\Delta t\mathbf{f}^{n+1} \tag{7}$$

3. Solve the pressure equation:

$$\nabla\cdot\left(\frac{\nabla p^{n+1}}{\rho}\right) = \frac{3}{2\Delta t}\nabla\cdot\mathbf{u}^* \tag{8}$$

4. Find the velocity field at the $n+1$ time instant:

$$\mathbf{u}^{n+1} = \mathbf{u}^* - \frac{2}{3}\frac{\Delta t}{\rho}\nabla p^{n+1} \tag{9}$$

5. Find the velocity field and nodal positions at time instant $n+1/2$ using quadratic interpolation of the velocity field through time instances $n-1, n$ and $n+1$:

$$\mathbf{u}^{n+1/2} = \frac{1}{8}\left(3\mathbf{u}^{n+1} + 6\mathbf{u}^n - \mathbf{u}^{n-1}\right) \tag{10}$$

$$\mathbf{x}^{n+1/2} = \mathbf{x}^n + \frac{\Delta t}{24}\left(2\mathbf{u}^{n+1} + 11\mathbf{u}^n - \mathbf{u}^{n-1}\right) \tag{11}$$

6. Repeat steps 1–5 starting from time instant $n+1/2$, meaning that in the above equations, the time indices $n-1, n, n+1/2$ and $n+1$ become $n-1/2, n+1/2, n+1$ and $n+3/2$, respectively.

To solve the discretized version of Eq. (8), we use the Conjugate Gradient Method without preconditioning, and a convergence criterion on the relative residual at 1.0×10^{-8}.

4. SPH discretization

4.1. Kernel function

In the present work, the truncated, renormalized Gaussian kernel proposed in [23] is used for SPH discretization in space. In 2D, it is given by

$$W(s,h) = \begin{cases} \frac{9}{4\pi h^2}\frac{e^{-\left(\frac{3s}{2h}\right)^2} - e^{-9}}{1-10e^{-9}} & \text{if} \quad s \leqslant 2h \\ 0 & \text{if} \quad s > 2h \end{cases} \tag{12}$$

where s is a distance, and h is the *smoothing length*. We will henceforth use a vector $\mathbf{x}$ as an argument to the distance function $W(\mathbf{x},h)$, such that $s = ||\mathbf{x}||$. The density, and thus the volume of each particle is kept constant in time, in all of the operators presented later. This differs from WCSPH, where the density is either tracked through the continuity equation, or evaluated directly from the kernel. Alternatively, we can let the density vary, and compute the particle volumes from

$$V_j = \frac{m_j}{\rho_j} = \left(\sum_k W(\mathbf{x}_j - \mathbf{x}_k, h)\right)^{-1} \tag{13}$$

where V_j denotes the volume of particle j, and k denotes all of the particles in the domain. In practice, k is limited to the neighborhood $s < 2h$ due to the compact support. Eq. (13) has been used later to approximate the total particle volume.

Although keeping the density constant is consistent with the incompressible methodology, the particle volumes may in practice vary slightly. This results in overestimation of inter-particle forces when particles cluster, which has a stabilizing effect on the total fluid volume.

To reduce the total discretization error, h should be increased as the total number of particles increases. However, increasing h quickly increases the density of the sparse coefficient matrices and thus the computational load. A common pragmatic approach is to fix h at a computationally feasible level. Unless otherwise specified, $h \approx 1.38\Delta x$ has been used in the subsequent work, where Δx is the particle spacing when the particles are organized in a square grid.

4.2. Operators

In [24], it is recommended to reformulate the differential operations to include the density. This may produce several desirable properties in the discretized operators such as conservation of momentum and reduced sensitivity to kernel truncation. The following SPH operators, taken from [24,25], are employed in the present work:

$$\left(\frac{1}{\rho}\nabla p\right)_i \approx \sum_j m_j\left(\frac{p_i}{\rho_i^2} + \frac{p_j}{\rho_j^2}\right)\nabla W_{ij} \tag{14}$$

$$(\nabla\cdot\mathbf{u}^*)_i \approx \sum_j \frac{m_j}{\rho_i}\mathbf{u}^*_{ji}\cdot\nabla W_{ij} \tag{15}$$

$$\left(\nabla\cdot\frac{1}{\rho}\nabla p\right)_i \approx \sum_j m_j\frac{8}{(\rho_i+\rho_j)^2}\frac{p_{ij}\mathbf{x}_{ij}\cdot\nabla W_{ij}}{||\mathbf{x}_{ij}||^2} \tag{16}$$

where the subscript ij applied to a vector or scalar denotes quantity evaluated at i minus quantity evaluated at j and ∇W_{ij} denotes $\nabla W(\mathbf{x}_i - \mathbf{x}_j, h)$.

4.3. Boundary conditions

In the present work, two types of boundary conditions are considered; solid walls and free surface.

4.3.1. Solid walls

Solid walls are represented by particles, just like the fluid. Two layers of boundary particles are used in the simulations presented in this paper. To prevent fluid particles from penetrating walls, the normal component of velocity on the wall should always be zero. To enforce this, the particles that represent solid walls are kept at zero velocity in (7) and (9). This generates a strong divergence where fluid particles impinge on walls, which acts as a source term in Eq. (8). The same approach is used in [26,27]. While being simple and providing a symmetric coefficient matrix in Eq. (8), a drawback of the method is that contact lines, i.e. the lines in 3D and points in 2D, at which the free surface and wall intersect, are not represented correctly. Even at purely tangential fluid motion relative to the wall, the divergence near a contact line is not zero in single-phase simulations. This pushes advancing contact lines away and attempts to suspend receding contact lines on the wall. However, this small artefact is negligible in the simulations presented in Section 5. To prevent fluid particles from getting stuck between the two layers of boundary particles, a pressure increment of $\rho g \Delta x$ is added to the second layer of boundary particles after solution of Eq. (8), where Δx is the particle spacing. This gently pushes trapped particles back into the container. As a quantitative example, it introduces an additional acceleration approximately normal to the walls of $\sim 1.5\,(\mathrm{mm/s^2})$ in the sloshing cases presented below, where the tank measures 1.73 (m). In this example, it has been assumed that the particles are positioned in the initial configuration, and the additional acceleration applies to a particle adjacent to the wall.

4.3.2. Free surface

Particles belonging to the free surface are characterized by truncated kernels. As in [22], the divergence of the position vectors is used, which has a theoretical value of 2 in 2D. However, with a truncated kernel, the evaluated divergence will be smaller. We relate the cutoff to the largest divergence of position vectors found in the whole domain. A surface particle is thus identified by

$$\begin{aligned} &\mathrm{div}(\mathbf{x})_j < \alpha \max_i(\mathrm{div}(\mathbf{x})_i) \\ &\mathrm{div}(\mathbf{x})_i \approx \sum_j \frac{m_j}{\rho_i} \mathbf{x}_{ji} \cdot \nabla W_{ij}. \end{aligned} \tag{17}$$

Unless otherwise specified, we have used $\alpha = 0.8$ in our simulations.

On the free surface, a constant reference pressure should be prescribed. In theory, any reference pressure may be applied. However, due to the symmetric formulation of Eq. (14), the sensitivity of the pressure gradient operator to kernel truncation heavily depends on the absolute value of the field. Assigning a reference pressure of zero minimizes the error caused by kernel truncation. However, a constant reference pressure also causes an undesirable effect – if the pressure gradient between surface particles is zero, they may cluster together or drift apart without any forces opposing the motion. Hence, it is preferable to keep the surface particles *close* to zero pressure while still allowing Eq. (8) to generate pressure gradients between the surface particles. A strict imposition of Eq. (5) on the free surface yields

$$p_i = 0, \tag{18}$$

and consequently

$$Cp_i \sum_j m_j \frac{8}{(\rho_i + \rho_j)^2} \frac{\mathbf{x}_{ij} \cdot \nabla W_{ij}}{||\mathbf{x}_{ij}||^2} = 0, \tag{19}$$

where C is some constant. Upon adding Eq. (19) to Eq. (16), the constant C determines the relative weighting between satisfaction of Eq. (5) and Eq. (8). By empirical investigation, $C = 1$ has been found to provide a suitable balance between the two. This means that for surface particles identified by Eq. (17), Eq. (16) is replaced by

$$\left(\nabla \cdot \frac{1}{\rho} \nabla p\right)_i \approx \sum_j m_j \frac{8}{(\rho_i + \rho_j)^2} \frac{(2p_i - p_j)\mathbf{x}_{ij} \cdot \nabla W_{ij}}{||\mathbf{x}_{ij}||^2}. \tag{20}$$

A similar approach was used in [28], although not with the Dirichlet boundary condition.

4.4. Fluid-body interaction

To model the interaction between fluid and rigid objects, the algorithm described in [26] is used. The basic idea is that forces acting on a rigid object are given by changes in its interior velocity field. The rigid object is first treated as part of the surrounding fluid, and its velocity field is subsequently constrained to that of rigid body motion. This way, the forces acting on the object are implicitly taken into account. After updating the velocity field according to Eq. (9), the velocity field inside the rigid object undergoes a second (explicit) projection given by the following equations:

$$\begin{aligned} \mathbf{x}_{CoG} &= \frac{\sum_i m_i \mathbf{x}_i}{\sum_i m_i} \\ \mathbf{r}_i &= \mathbf{x}_i - \mathbf{x}_{CoG} \\ \mathbf{U} &= \frac{\sum_i m_i \mathbf{u}_i}{\sum_i m_i} \\ \mathbf{R} &= \frac{\sum_i \mathbf{r}_i \times m_i \mathbf{u}_i}{\sum_i m_i \mathbf{r}_i \cdot \mathbf{r}_i} \\ \mathbf{u}_i^r &= \mathbf{U} + \mathbf{R} \times \mathbf{r}_i \end{aligned} \tag{21}$$

where i indicates all particles belonging to the rigid object. As can be seen from (21), the total linear and angular momentum are found, and then redistributed back to the particles as if the fluid were rigid. The above approach violates the continuity equation on the boundary between the rigid body and surrounding fluid, but despite this discrepancy, the method is still applicable for our purposes.

5. Numerical experiments

5.1. Test 1 – The evolution of an elliptical drop

In order to demonstrate the effectiveness of the proposed time integration scheme, as well as provide a comparison between some

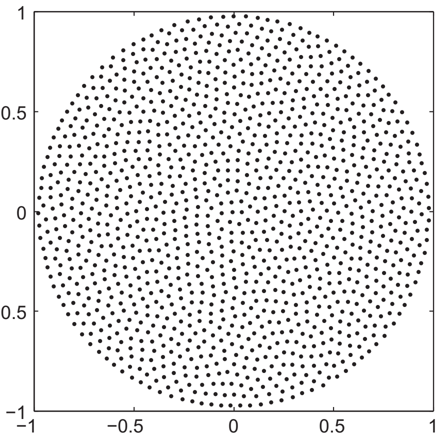

Fig. 1. Initial particle configuration for the elliptical drop test case.

Table 1
The evolution of an elliptical drop. Distinguishing features of the time integration schemes; method 1 [22], method 2 [8] and method 3 (Section 3).

	Extrapolated velocity profile for predicted position	Finite Difference scheme for velocity time derivative	Interpolated velocity profile for integration of position	Half-step interpolation
Analogous to equation	Eq. (6)	Eq. (7)	Eq. (11)	
Method 1	–	First order	Constant	No
Method 2	Constant	First order	Linear	No
Method 3	Linear	Second order	Quadratic	Yes

Table 2
Rigid body water entry, comparison of proposed ISPH with WCSPH and experimental data in [18].

	Resolution	B (m)	H (m)	R (m)	Wave height (m)
ISPH	$N_y = 20$	0.254	0.310	0.133	0.304
	$N_y = 40$	0.279	0.311	0.133	0.314
WCSPH	$N_y = 20$	0.273	0.308	0.075	0.303
	$N_y = 40$	0.273	0.309	0.109	0.306
Experiment	–	0.303	0.333	0.133	0.302

of the existing alternatives, we apply different time integration schemes to compute the motion and pressure in a circular fluid patch subjected to a straining velocity field. The 2D test was applied to WCSPH in [7,23], and also extended to 3D in [29]. Two aspects of the method are tested; accuracy/convergence, and stability. The corresponding simulation setups are distinguished by different kernel sizes. In the present work, the 2D problem is set up as follows:

A circular fluid patch of radius R, centered at $(x, y) = (0, 0)$ is subjected to an initial velocity field

$$\begin{aligned} u(x,y) &= -A_0 x \\ v(x,y) &= A_0 y \end{aligned} \tag{22}$$

with the pressure boundary condition

$$p = 0 \quad \text{on } \Gamma_f. \tag{23}$$

This velocity field will stretch the fluid patch into an elliptical shape. A semi-analytical solution to this problem can be found in the form of an ordinary, non-linear differential equation, which may be integrated using standard Runge–Kutta methods. Defining $\tau = A_0 t$, where t is physical time, the dimensionless equation reads

$$\mathbf{Y}(\tau) = \begin{bmatrix} Y_1 \\ Y_2 \\ Y_3 \end{bmatrix}, \quad \dot{\mathbf{Y}}(\tau) = \begin{bmatrix} Y_2 \\ 4\frac{Y_2^2}{Y_1} - 2Y_1^3 \\ -Y_1 \\ Y_3 \end{bmatrix} \mathbf{Y}(0) = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} \tag{24}$$

where the dot denotes differentiation with respect to τ. The physical interpretation is $Y_3 = \frac{a}{R}$ where a is the time dependent semi-minor axis of the evolving ellipse. Y_1 and Y_2 follow from Y_3. The dimensionless pressure at the origin as a function of dimensionless time is obtained by

$$\begin{aligned} \tilde{p}(\mathbf{0}, \tau) &= \frac{1}{2} Y_3^2 (Y_1^2 - Y_2) \\ \tilde{p} &= \frac{p}{\rho A_0^2 R^2} \end{aligned} \tag{25}$$

In the following, the solution of Eq. (24) has been obtained by a Runge–Kutta (4,5) method [30] with a relative tolerance of 10^{-10}, roughly corresponding to the number of correct digits. Though conceptually very simple, this type of problem, where fluid elements are subjected to regular straining velocity fields for longer periods of time, can be challenging for SPH methods. This is demonstrated in [13], for Taylor–Green vortices. The reason appears to be that strong particle clustering and anisotropy impedes the numerical approximation of the kernel integrals. The same problem is present in [31], where the current test case is simulated with various formulations of SPH with an artificial equation of state. To reduce this problem, which is particularly inhibitive in the current test case, the initial particle configuration was obtained from a variant of the bubble packing method [32]. This produces a semi-regular particle distribution, which better allows the particles to redistribute themselves when subjected to uniform strain.

In the practical simulations, $R = 1$ (m), $\rho = 1$ (kg/m^3) and $A_0 = 1$ (m/s).

To obtain the initial particle configuration, a mass-spring-damper system of the type

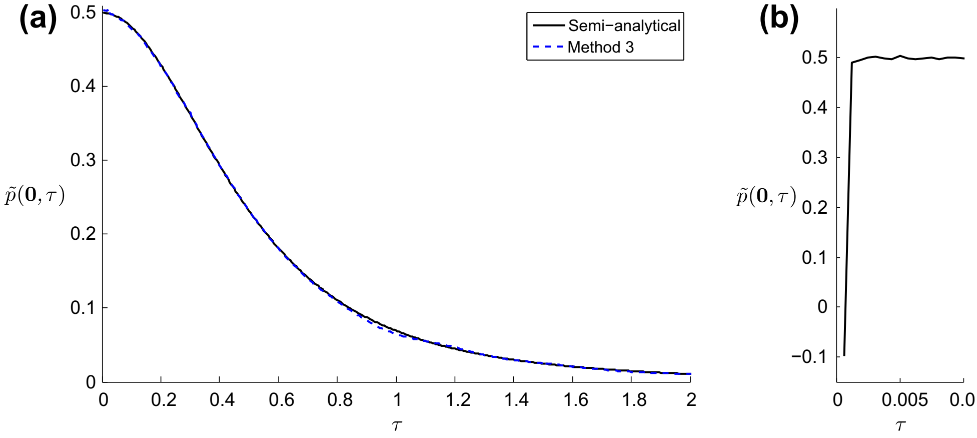

Fig. 2. Comparison of semi-analytical solution and best results produced by ISPH; method 3, $\Delta\tau = 0.01$ – (a), initial pressure with subsequent high-frequent noise, method 1, $\Delta\tau = 1.25 \times 10^{-3}$ – (b).

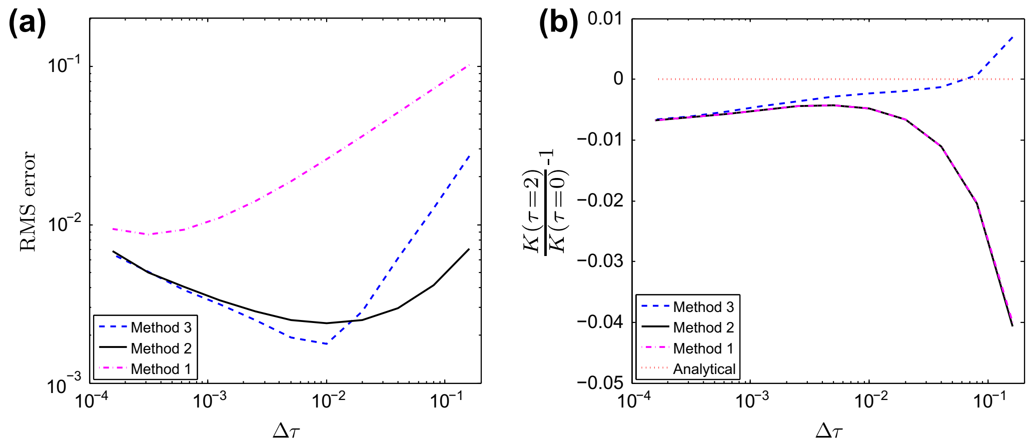

Fig. 3. Time series RMS error of pressure at the origin – (a), changes in kinetic energy – (b).

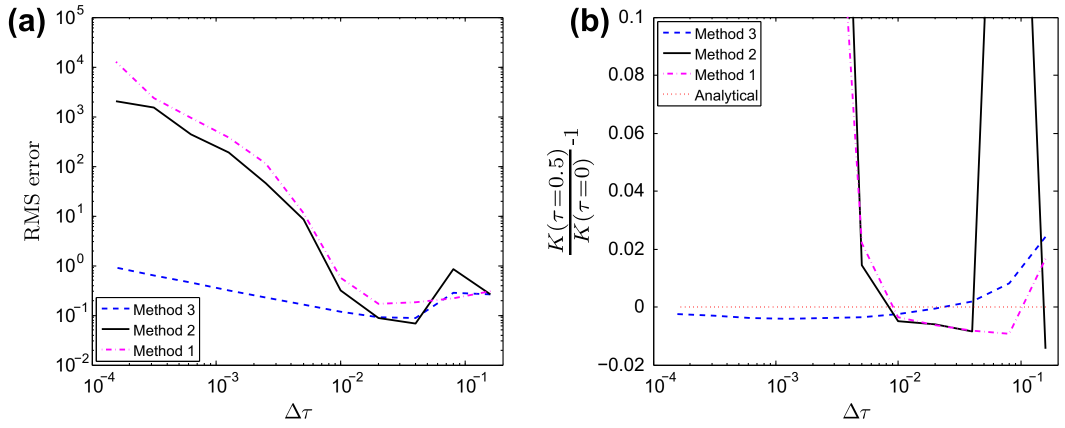

Fig. 4. Simulations with inappropriately small kernel ($h \approx 1.17\Delta x$); time series RMS error of pressure at the origin – (a), changes in kinetic energy – (b).

$$m\frac{D^2\mathbf{x}}{Dt^2} + c\frac{D\mathbf{x}}{Dt} = \mathbf{f} \tag{26}$$

was integrated in time, with a rigid circular boundary represented by five layers of particles, used to keep the free particles in place. The free particles were initially randomly distributed within a circle of radius $R = 1$ (m). Rather than using Van der Waals type forces for $\mathbf{f}$ as in [32], we used the SPH pressure gradient operator, and Eq. (26) was recast as

$$\frac{D^2\mathbf{x}}{Dt^2} + c\frac{D\mathbf{x}}{Dt} = -\sum_j m_j\left(\frac{p_i}{\rho_i^2} + \frac{p_j}{\rho_j^2}\right)\nabla W_{ij} \tag{27}$$

In Eq. (27), $p_i = p_j = 200$ (Pa), $\rho_i = \rho_j = 1$ (kg/m^3), $c = 200$ (1/s), and $m_j = \frac{\pi}{n}$ (kg), where n is the number of particles. In the solution of Eq. (27), a relatively wide kernel, $h = 1.8\Delta x$ was used, as it was found to produce a more uniform particle density. The stop criterion for the time integration was chosen as

$$\frac{s_V}{\mu_V} < 4 \times 10^{-4} \tag{28}$$

where s_V and μ_V are the sample standard deviation and mean of the particle volumes respectively, and the particle volume is approximated by Eq. (13). The drop was represented by 1250 particles, which was the lowest spatial resolution tested in [29]. The use of Eq. (27) allows imposition of the exact number of particles whilst keeping a smooth boundary. The initial particle configuration is shown in Fig. 1, and the same configuration was used in all simulations.

The considered time integration schemes are those presented in [22,8], and the scheme presented in Section 3, which will be

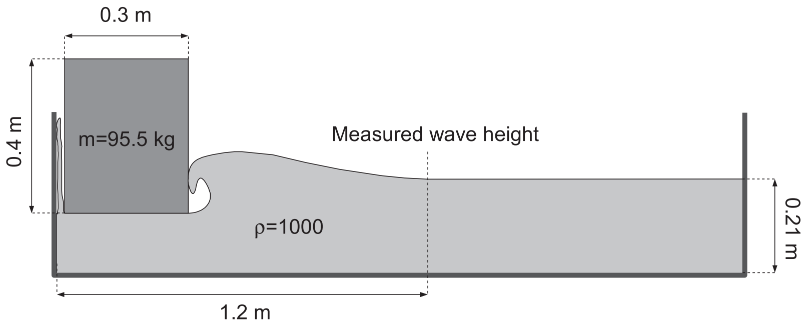

Fig. 5. Rigid body water entry, the experimental setup from Monaghan and Kos [18].

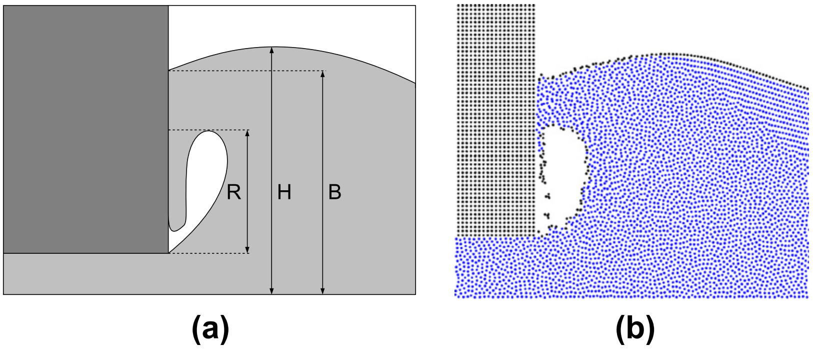

Fig. 6. Rigid body water entry, key parameters describing the shape of the cavity formed by the rebounding wave – (a) and ISPH simulation with $N_y = 40$ at $t = 0.285$ (s) – (b).

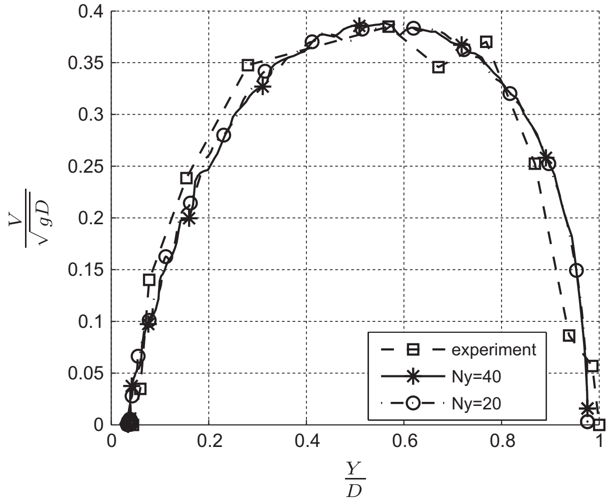

Fig. 7. Rigid body water entry, dimensionless box velocity, ISPH vs. experimental data.

Table 3
Sloshing cases.

	Fill level (m)	Excitation period (s)	Closest linear natural frequency (s)	Description
Case 1	0.3	2.40	2.11 (1st)	Breaking bore causes spray
Case 2	0.2	2.60	2.52 (1st)	Travelling bore gives impact at wall

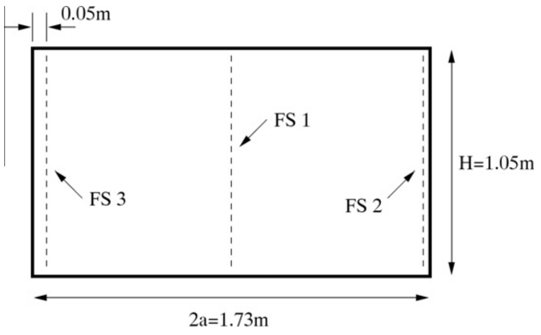

Fig. 8. Sloshing, tank dimensions and sensor positions.

referred to as methods 1–3, respectively. These are relatively similar as the divergence is approximated by the same SPH operator, but differ in the way the position, velocity and divergence are coupled. The differences are summarized in Table 1. It should also be noted that none of the three produce discretely divergence free velocity fields, as they are based on *approximate* rather than *exact* projection, i.e., the Laplacian operator is not the divergence operator applied to the gradient operator in the discrete sense.

The time history of pressure at the origin and total kinetic energy have been extracted. A normalized kernel of the same size as that of the operators has been used to extract the pressure. Since the flow is inviscid, kinetic energy should be perfectly conserved. This case also serves to test the validity of Eq. (20), as it is used to impose the pressure boundary condition on the surface. Eleven different time steps have been tested, starting at $\Delta\tau = 0.16$ and successively halving the step sizes. $\Delta\tau$ spans two half steps of our proposed scheme (method 3), and to make a fair comparison, we use half of the above steps in method 1 and method 2, so that the computational work for a given step size is approximately the same in all methods. The time step given in the subsequent comparisons are those used in method 3. Method 3 is not self-starting in the current case, and thus, two steps of method 2 are used to initialize method 3.

5.1.1. Accuracy and convergence

As uniform strain causes an undesirable particle clustering, the kernel size has been increased slightly in this problem, to $h \approx 1.70\Delta x$. Δx is found as $\sqrt{\frac{\pi R^2}{n}}$ where n is the number of particles. This has been done to incorporate a sufficient amount of particles in all spatial directions. The particles tend to self-organize into uniform distributions, however this process is sometimes too slow compared to the strain from the velocity field.

As can be seen in Fig. 2a, ISPH has the potential to give a highly accurate representation of the pressure at the origin. The accuracy is remarkable, especially when compared to WCSPH; in [29], the mean error of a simulation using 80000 particles was reported to be 1.6%. It is however unclear if this result was obtained with optimal parameters. At 1/64th of that particle count, we are able to obtain a mean error (not RMS) of 0.12%. However, we found that the accuracy of the pressure is highly dependent on the kernel size and initial particle configuration, and we have attempted to choose these as optimal as possible based on experience. It should be noted that in [29], the particles are initialized with the analytical pressure solution, and the time series are filtered to remove acoustic oscillations, none of which are necessary with ISPH.

Fig. 3a shows the RMS error of the time series of the pressure at the origin for different time steps. The error is computed according to

$$\text{RMS error} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\tilde{p}(\mathbf{0},\tau_i))^2} \tag{29}$$

where N is the total number of time steps. All of the tested integration methods provide good results for the pressure at the origin. Method 3 produces the smallest recorded error of all, with a time step $\Delta\tau = 0.01$. The differences between the three are however not striking. For the largest time step, the log–log slopes are estimated to 0.5, 0.76 and 1.09 for methods 1–3 respectively, but it should be noted that method 2 is not linear in the log–log plot. However the pressure results display a troublesome tendency to diverge as the time step is decreased beyond a certain size. Studying the time series, we have found that this is caused by low-amplitude, high-frequent noise that increases with decreasing time steps. We have not been able to relate this to either the free surface boundary condition or the accuracy of the pressure solver, and the apparent reason is that the two sources of divergence are the physical motion of the particles, and numerical error. As the time step decreases towards zero, the latter will be proportionally larger. Indeed, in method 1, at the first time step, the velocity field is set to the analytical, divergence free solution, and as the particle positions are not extrapolated in time, the computed divergence consists mainly of numerical error. Consequently, the first computed pressure field in method 1 is unphysical; in fact, the pressure at the origin is negative. The initial time history of pressure at the origin for method 1, $\Delta\tau = 1.25 \times 10^{-3}$, is shown in Fig. 2b, where also the high-frequent noise can be seen. In the subsequent test cases, this behavior has not posed any problems.

Table 4
Sloshing cases, parameter settings in SPHysics (WCSPH).

Kernel	Wendland kernel, $h \approx 1.27\Delta x$ with compact support at 2 h
Speed of sound	0.53 (m/s)
Artificial viscosity	None
Initial particle configuration	Staggered (every second column offset by $0.5\Delta x$)
Density filtering	Shepard filter applied every time step
Time integration	Beeman
Wall boundary condition	Repulsive

Fig. 3b shows the change in kinetic energy for different time steps, given as the ratio of the final to initial kinetic energy minus one. While methods 1 and 2 display a virtually identical behavior and show a strong increase in numerical dissipation at large time steps, method 3 shows a markedly better conservation of kinetic energy at large time steps.

5.1.2. Stability

To demonstrate the increased stability of our proposed time integration scheme, the kernel size was reduced to $h \approx 1.17\Delta x$. This is a deliberately bad choice; the kernel is inappropriately small compared to the nature of the problem, which induces severe pressure fluctuations and instability. To avoid losing numerical connectivity between the particles due to clustering, the simulation was stopped earlier – at $\tau = 0.5$. Simulations have been run with the same time steps as in Section 5.1.1. The results are shown in Fig. 4. Clearly, the smaller kernel increases the Courant number $\frac{V_{max}\Delta t}{h}$, and the 3–4 largest time steps suffer from this. The pressure is very poorly predicted with all methods, but it is interesting to note the differences; for the smallest time step, method 3 has errors 3–4 orders of magnitude below the others. The reason is seen in Fig. 4b, which shows that the kinetic energy has an erratic response to the time step size with methods 1 and 2, indicating a breakdown of the simulation. They are only stable for a small range of $\Delta\tau$, whereas method 3 is stable across the whole range. An empirical investigation revealed this to be a result of the half-step interpolation in time, which we believe act much like under-relaxation in iterative methods.

5.2. Test 2 – Rigid body water entry

To investigate the algorithm's ability to handle rigid body simulations, the wave generator experiment presented by Monaghan

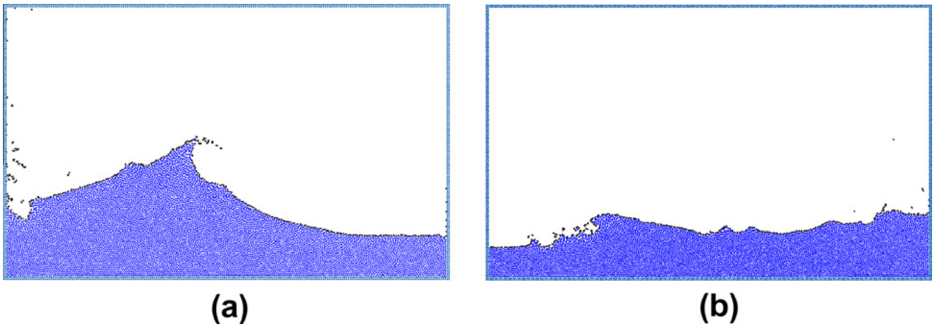

Fig. 9. Sloshing by ISPH, spray in Case 1 – (a) and bore in Case 2 – (b).

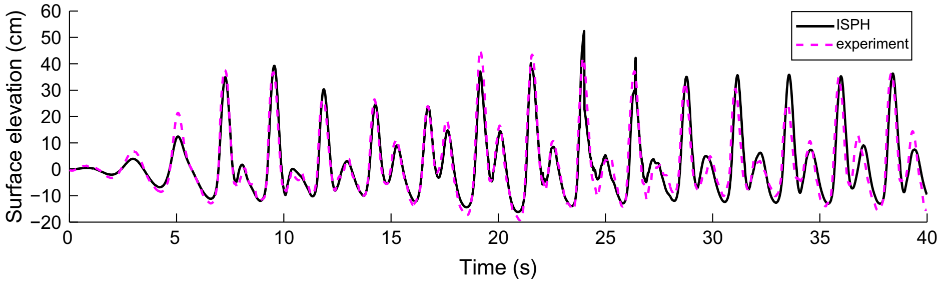

Fig. 10. Sloshing Case 1 (0.3 (m) fill), surface elevation along FS3, ISPH vs. experimental data.

and Kos in [18] was simulated. This problem was also simulated with ISPH in [33], although the box motion was prescribed rather than computed, so that the problem was reduced to a one-way coupling between body and fluid. A solitary wave is generated by dropping a box into a tank filled with water. The experiment is set up in such way that the resulting motion is mainly two dimensional. The box is constrained to vertical motion by guide rails. Fig. 5 shows the experimental setup. The tank measures 9 (m) in length and 0.4 (m) in width. Initially, the box, with mass 38.2 (kg), is positioned 2 (cm) from the left hand side of the tank and with the bottom side ~ 0.5 (cm) below the water surface in order to avoid a splash.

In the simulations, the actual mass of the box was converted to an appropriate representation in 2D, where the 'unit depth' is one meter. This gives a mass of 95.5 (kg). The box was represented by a full set of particles. The initial free surface height was 0.21 (m), and the height of the solitary wave was measured 1.2 (m) from the left wall. To save CPU time, the length of the tank was reduced to 3 (m), which is far enough not to influence the wave height 1.2 (m) from the wall.

Besides the height of the generated wave, the velocity of the dropped box and key parameters indicating the shape of the cavity formed by the rebounding plunging wave were extracted. The latter is caused by water flowing back from the steep solitary wave. The shape parameters are measured at $t = 0.285$ (s) and illustrated in Fig. 6a; a snapshot of the cavity generated by the ISPH simulations is shown in Fig. 6b.

The ISPH results are compared with experimental data and WCSPH results from Monaghan and Kos [18] in Table 2, where N_y indicates the initial number of particles in the vertical direction from the tank floor to the free surface. The uncertainty margins in the experiment are ±0.02 (m) for B and R and ±0.01 (m) for H and the wave height. In the ISPH simulations, we have used $\Delta t = 0.002$ (s) with $N_y = 20$ and $\Delta t = 0.001$ (s) with $N_y = 40$. As can be seen, ISPH reproduces the experimental data relatively well in terms of the cavity shape and wave height, and the results are comparable to the WCSPH simulations in [18].

Fig. 7 shows the dimensionless box velocity plotted against the dimensionless length, where Y is the distance from the bottom of the tank, D is the still water depth, V is the box velocity and g is

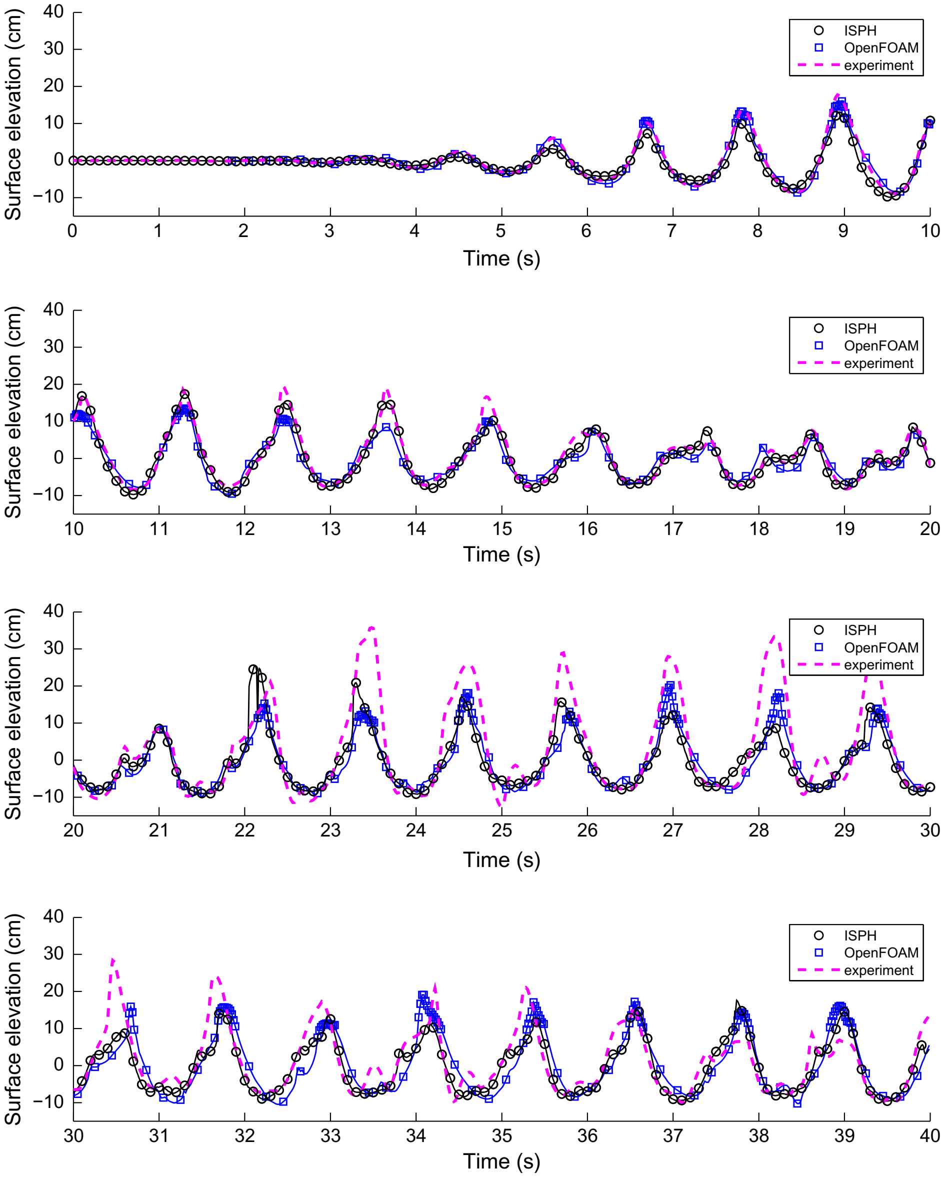

Fig. 11. Sloshing Case 1 (0.3 (m) fill), surface elevation along FS1, ISPH vs. OpenFOAM and experimental data.

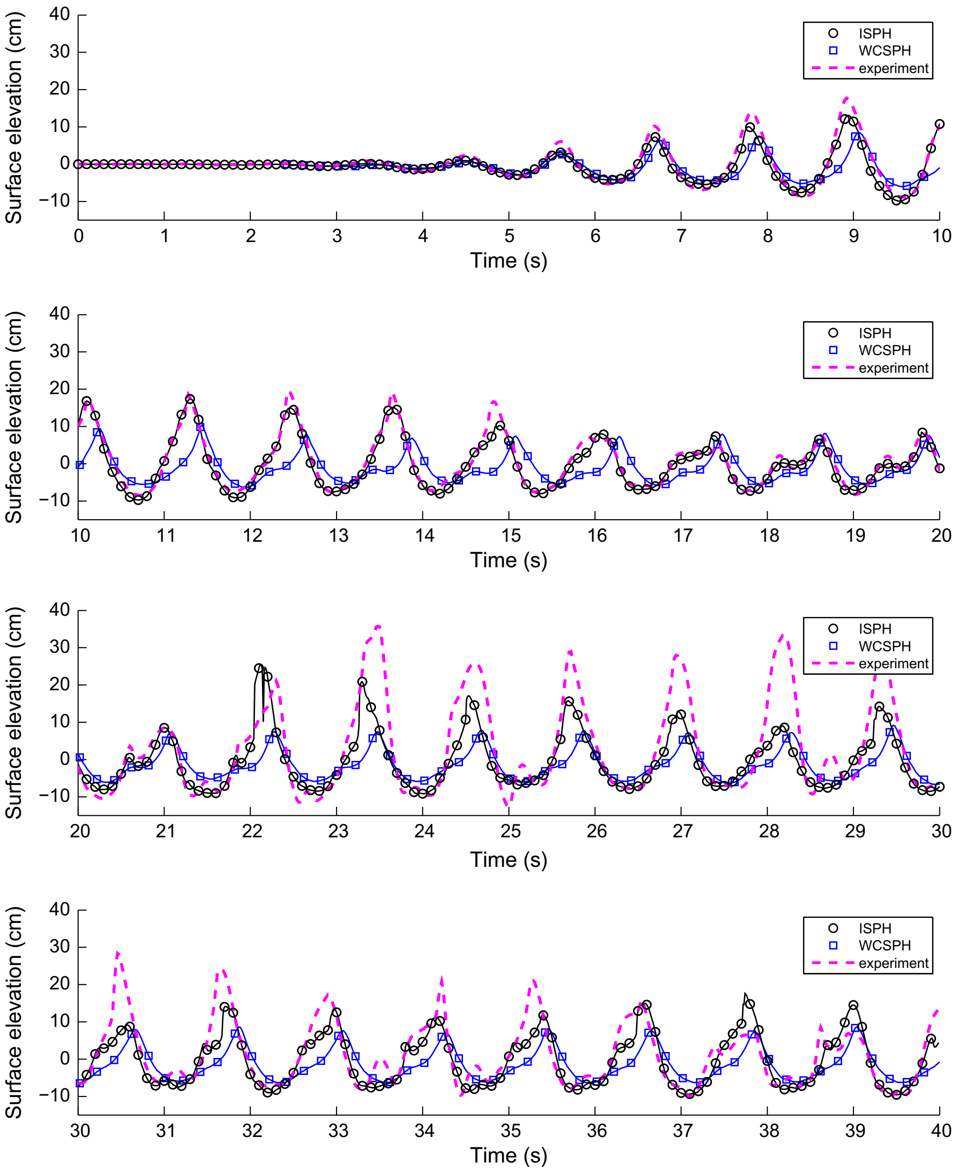

Fig. 12. Sloshing Case 1 (0.3 (m) fill), surface elevation along FS1, ISPH vs. WCSPH and experimental data.

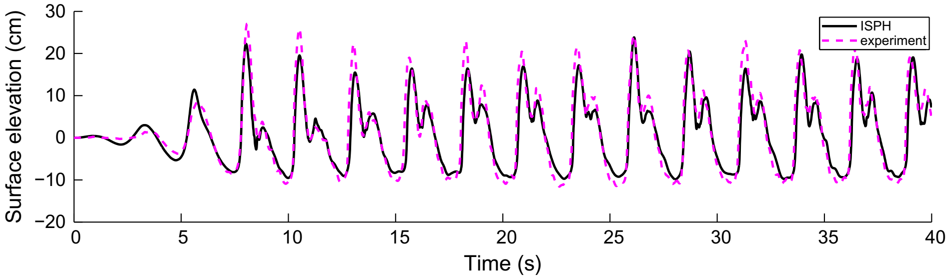

Fig. 13. Sloshing Case 2 (0.2 (m) fill), surface elevation along FS3, ISPH vs. experimental data.

the gravitational acceleration. As can be seen, the curve produced by ISPH is relatively smooth. It is also worth noting that for ISPH, the box velocities approximated with $N_y = 20$ and $N_y = 40$ are almost identical, indicating early convergence, while they are relatively different for WCSPH in [18]. Moreover, the dimensionless box velocity found by WCSPH exhibits significant oscillations. There might be indications of similar oscillations in the experiments as well, although there are too few data points and possibly too large tolerances to conclude whether they are physical or not. The experiment was also simulated using the volume of fluid method in [34], and the resulting dimensionless box velocity showed no clear oscillations.

5.3. Test 3 – Sloshing

In [35], SPH was tagged as a promising method to study sloshing problems, in particular, the ones involving violent flows and breaking waves. A number of works studying sloshing with SPH, see for example [36–39,23], have reported good results as compared to experiments. These works were all based on the weakly compressible formulation, and to the authors' knowledge, only one conference paper [40] has been devoted to studying sloshing with the incompressible formulation. The latter used a density-based rather than divergence-based formulation of the incompressibility constraint, and compared impact pressures on the walls with experimental results. Good agreement was found, although neither the ISPH code nor the commercial software used for comparison captured the largest peaks.

In 1998, a number of sloshing experiments were carried out at Det Norske Veritas (DNV) using a rectangular tank and horizontal oscillatory motion, see [41]. Depending on the fill level and frequency, anything from smooth harmonic surface waves to violent sloshing and travelling bores could be produced. In the present work, we have considered the two cases described in Table 3 for a tank measuring 1.73 (m) in length and 1.05 (m) in height.

The linear natural periods in a rectangular tank are given by

$$T_i = \sqrt{\frac{4\pi l}{ig\tanh\left(\frac{\pi i h}{l}\right)}}, \quad i = 1,2,\ldots, \tag{30}$$

where h is the water depth, l is the length of the tank and g is the gravitational acceleration. For small ratios $\frac{h}{l}$, $\tanh\left(\frac{\pi i h}{l}\right) \approx \frac{\pi i h}{l}$, meaning that $\frac{T_i}{T_1} = \frac{1}{i}$, i.e., higher mode frequencies are multiples of the lowest natural frequency. In [35], this limit is given as $\frac{h}{l} \lesssim 0.05 - 0.1$, under which conditions are termed 'shallow'. Shallow flows are characterized by strongly non-linear effects such as hydraulic jumps. Case 2 is bordering to this limit, resulting in a wave impacting on the wall which is steeper than in Case 1. This produces more violent impacts.

In the experiments, the water height was probed at different locations in the tank, see Fig. 8. In the simulations, the surface elevation was extracted at probe FS1 in the middle of the tank and probe FS3 positioned 0.05 (m) from the left wall. A linear 'ramp'

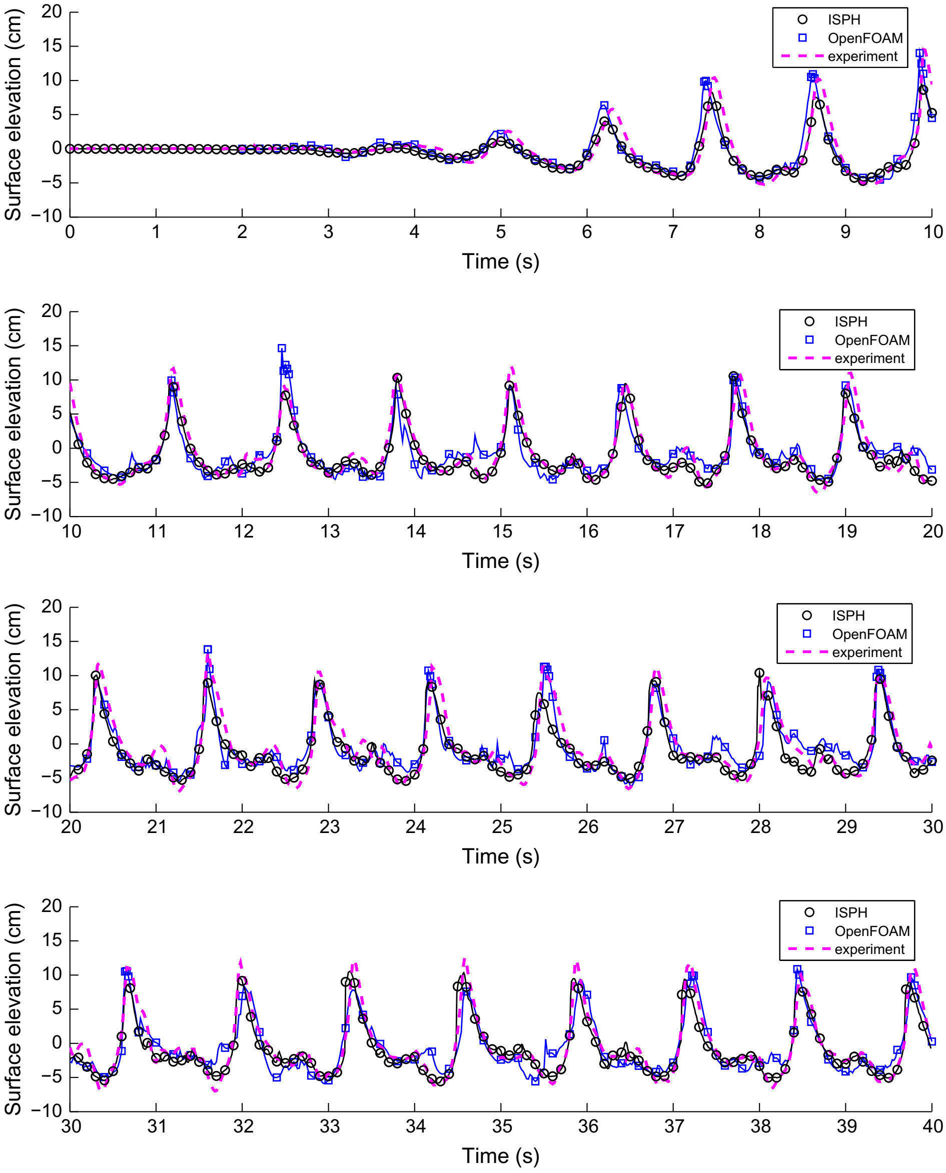

Fig. 14. Sloshing Case 2 (0.2 (m) fill), surface elevation along FS1, ISPH vs. OpenFOAM and experimental data.

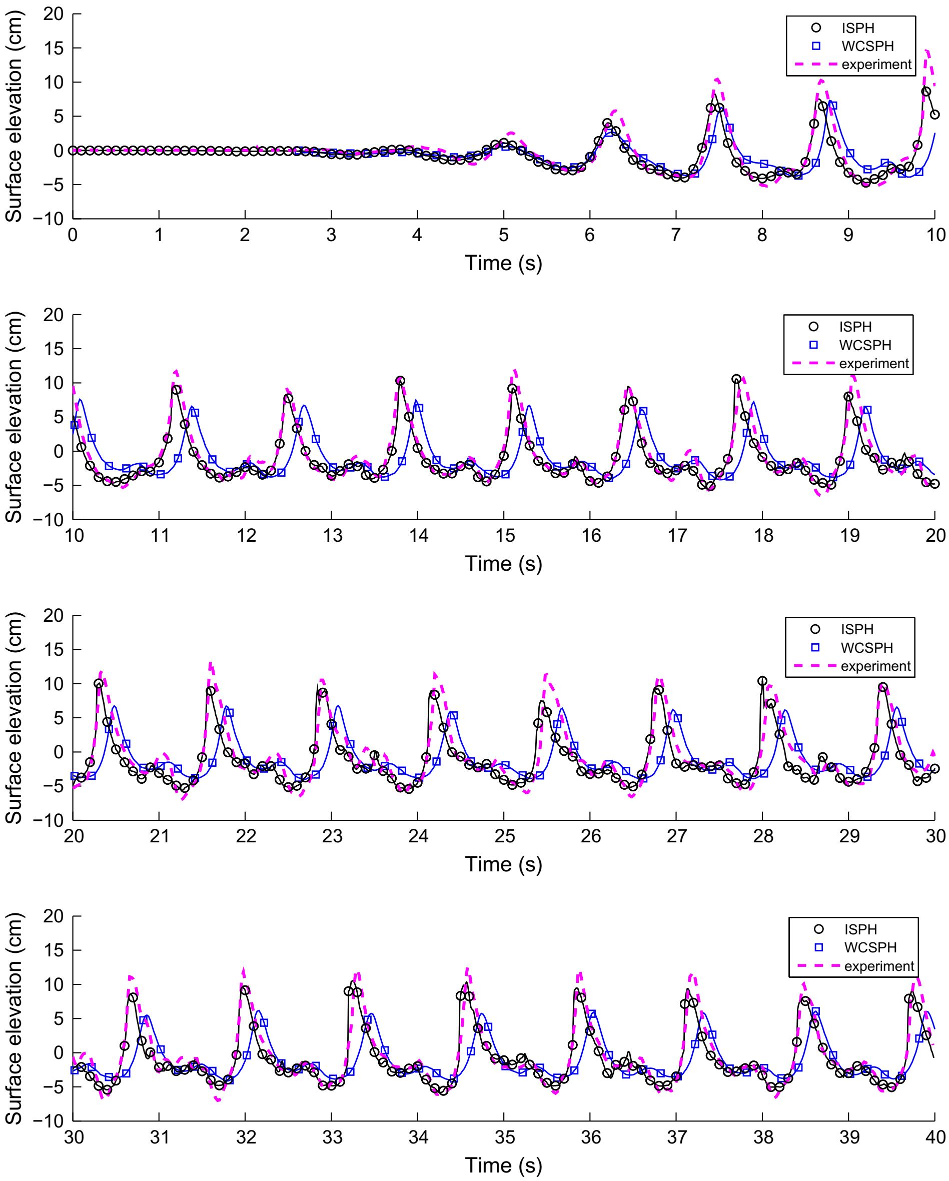

Fig. 15. Sloshing Case 2 (0.2 (m) fill), surface elevation along FS1, ISPH vs. WCSPH and experimental data.

spanning two excitation periods was applied to the tank motion in the ISPH simulations, which is relatively similar to the start-up conditions in the experiment. The tank motion was included by modifying the body force vector. For probe FS1, the ISPH results are also compared with simulations by WCSPH implemented in SPHysics and by the FV/VOF method implemented in OpenFOAM. For the latter, a Cartesian 2D mesh was used with cell spacing corresponding to the initial particle spacing in SPH; $\Delta x = 0.0075$ in Case 1 and $\Delta x = 0.0067$ in Case 2. The parameter settings in SPHysics are given in Table 4. The number of particles in both ISPH and WCSPH is ~11,000 for Case 1 and ~9000 for Case 2. The time step is 5×10^{-4} (s) in both cases. In the ISPH simulations, Courant numbers up to ~0.17 (or 0.34 with our definition of a full time step) were reached, and the simulations did not suffer from instability problems. Snapshots from the simulations are shown in Fig. 9a and b.

The surface elevation near the tank wall (FS3) for Case 1 is presented in Fig. 10, and at the middle of the tank (FS1) in Figs. 11 and 12. The surface elevation found by ISPH at FS3 is in very good agreement with the experiment. A series of high peaks are observed in the experiment along FS1 between ~22 and ~32 (s). The first peak is captured only by ISPH, but ISPH, OpenFOAM and WCSPH all underestimate the surface elevation at FS1 during this period. Note that the first peak is caused by a spray in ISPH, see Fig. 9a. It is remarkable that both the quantity and quality of the error are almost the same for ISPH and the FV/VOF method of OpenFOAM, which are, after all, very different methods. It thus seems plausible that the error may be caused by the lack of a viscosity/turbulence model.

Case 2 is presented in Fig. 13 for FS3 and in Figs. 14 and 15 for FS1. In the simulation, the peaks at FS1 represent the secondary wave generated from a steep, collapsing wave right next to the middle of the tank. The wave further propagates much like a bore until it hits the wall. This bore is depicted in Fig. 9b.

In addition to the surface elevation measurements, frequency analysis based on data from FS1 for the experimental data and simulations by ISPH, OpenFOAM and WCSPH was carried out. The results are presented in Fig. 16; here, twenty periods after 8 (s) were used as input to the Direct Fourier Transform, and the Power Spectral Density (PSD) was extracted. As predicted by Eq. (30), the

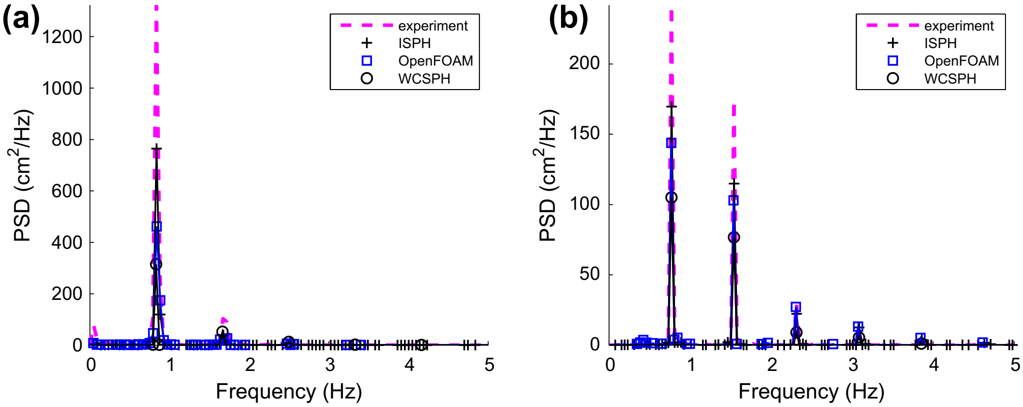

Fig. 16. Sloshing, frequency analysis based on data from FS1 in Case 1 – (a) and Case 2 – (b).

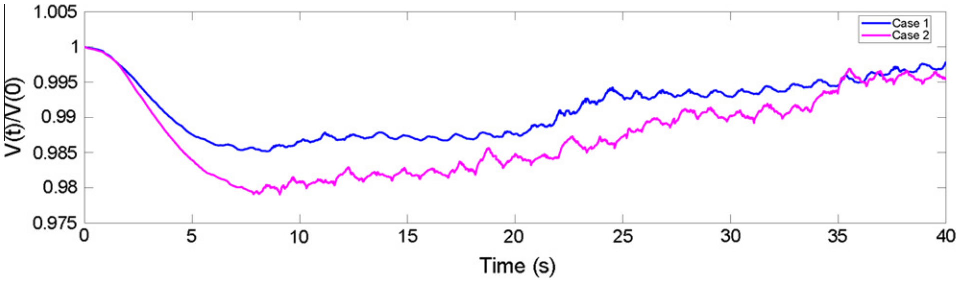

Fig. 17. Sloshing, volume fluctuations in the ISPH simulations.

frequency spectrum in Case 1 has evenly spaced peaks, see Fig. 11. Since the wave travels forth and back during one forced period of oscillation, the lowest frequency is double that of the excitation frequency. In shallow water sloshing, Case 2, a significant portion of the energy is contained in the higher frequencies, Fig. 16b. All three methods predict excitation at the correct frequencies for both cases. However, while ISPH gives higher PSD levels than OpenFOAM and WCSPH, all methods give low results compared to the experiment, partly due to the previously mentioned errors.

Fig. 17 shows the ratio of the total volume to the initial volume for ISPH. Here, the total volume is estimated as the sum of the individual particle volumes by Eq. (13). It can be seen that the largest volume error is around 1.5% in Case 1 and 2% in Case 2. This occurs soon after the beginning of the simulations and might result from rapid particle reorganization from the ordered grid to an unordered configuration. At the end of the simulations, the error in total volume is less than 0.5% for both cases.

6. Conclusions

An incompressible SPH method for inviscid free surface flows has been presented and tested. Apart from the smoothing length required in all types of SPH simulations, there are no other parameters used in ancillary procedures that need to be tuned. Moreover, the method does not rely on introducing artificial or unphysical effects for stability. From an industrial point of view this is an attractive feature, since it provides more confidence in the results and does not require the same amount of expertise from the user as WCSPH. Another attractive feature of SPH in general is that it is readily extendable to a variety of problems. The extension of ISPH to two-way coupled rigid body simulations, requiring only a few lines of additional code, has been demonstrated. In the elliptical drop test case, ISPH has shown its ability to capture the semi-analytical solution of the time dependent central pressure with very good accuracy. It has been discovered that with a good choice of initial particle configuration and kernel size, ISPH is able to provide a more accurate solution than WCSPH simulations [29] at 64 times the particle count. It should be noted however, that the initial particle configuration in the WCSPH results is not known. The test also demonstrated that our proposed time integration scheme improves stability, and better preserves kinetic energy. The sloshing test cases have shown that the method is able to reproduce violent sloshing with good accuracy. They are similar to those obtained by the FV/VOF implementation in the open source code OpenFOAM, and are superior in comparison to WCSPH results from SPHysics. It has been found that ISPH is able to preserve the total volume of fluid well, given that the average Courant number is sufficiently small. It should however be noted that very small Courant numbers may introduce pressure oscillations, apparently due to an increased ratio of numerical error to physically based divergence. The time step may of course be decreased towards zero, but this should be accompanied by an increase in spatial resolution. Throughout the simulations presented in the present work, we have observed one single particle escaping through a wall. Leakage may be a problem in some SPH implementations, as impermeability is not usually explicitly enforced.

Though ISPH is highly competitive and may even surpass Eulerian methods for violent free surface flows, comparisons for nonviolent motion have not been made in the present work. This is in fact considered to be the weakest point of SPH in general. Continuation of the present work should also include parallelization and an extension to three dimensions, which would enable the code to handle problems of more practical interest.

Although the equations and methodology of SPH are easily understood, there are still issues relating to the mass-based particle representation of continuum mechanics. Improved consistency between numerical procedures and actual physics may be the key to further elevate the integrity of SPH.

References

[1] Gingold RA, Monaghan JJ. Smoothed particle hydrodynamic: theory and application to non-spherical stars. Month Not Roy Astron Soc 1977;181:375–89.

[2] Lucy L. A numerical approach to the testing of the fission hypothesis. Astron J 1977;82:1013–24.

[3] Ellero M, Kröger M, Hess S. Viscoelastic flows studied by smoothed particle dynamics. J Non-Newton Fluid Mech 2002;105(1):35–51.

[4] Liu S, Chen J. Fracture of thin-walled structure with SPH shell formulation. Thin-Walled Struct 2010;48(2):118–26.

[5] Losasso F, Talton J, Kwatra N, Fedkiw R. Two-way coupled SPH and particle level set fluid simulation. IEEE Trans Visual Comput Graph 2008;14:797–804.

[6] Aktay L, Johnson AF. Advances in meshfree techniques. Computational Methods in Applied Sciences, vol. 5. FEM/SPH Coupling technique for high velocity impact simulations; 2007. p. 147–67. ISBN:978-1-4020-6094-6.

[7] Monaghan JJ. Simulating free surface flows with SPH. J Comput Phys 1994;110:399–406.

[8] Cummins SJ, Rudman M. An SPH projection method. J Comput Phys 1999;152(2):584–607.

[9] Chorin AJ. Numerical solution of the Navier–Stokes equations. Math Comput 1968;22(104):745–62.

[10] Temam R. Une méthode dGapproximation de la solution des équations de Navier–Stokes. Bull Soc Math France 1968;98:115–52.

[11] Koshizuka S, Oka Y. Moving-particle semi-implicit method for fragmentation of incompressible fluid. Nucl Sci Eng 1996;123(3):421–34.

[12] Shao S, Lo EYM. Incompressible SPH method for simulating Newtonian and non-Newtonian flows with a free surface. Adv Water Resour 2003;26:787–800.

[13] Xu R, Stansby P, Laurence D. Accuracy and stability in incompressible SPH (ISPH) based on the projection method and a new approach. J Comput Phys 2009;228(18):6703–25.

[14] Khayyer A. Improved particle methods by refined differential operator models for free-surface fluid flows. PhD thesis. Kyoto University; 2008.

[15] Hu XY, Adams NA. An incompressible multi-phase SPH method. J Comput Phys 2007;227(1):264–78.

[16] Ellero M, Serrano M. P. Español, incompressible smoothed particle hydrodynamics. J Comput Phys 2007;226(2):1731–52.

[17] Solenthaler B, Pajarola R. Predictive–corrective incompressible SPH. ACM Trans Graph (TOG) 2009;28(3):40.

[18] Monaghan JJ, Kos A. Scott Russell's wave generator. Phys Fluids 2000;12(3):622–30.

[19] Gómez-Gesteira M, Rogers B, Dalrymple R, Crespo A. Narayanaswamy. User guide for the sphysics code v2.0; 2010. <http://wiki.manchester.ac.uk/sphysics>.

[20] OpenFOAM User guide version 1.6. OpenCFD Ltd.; 2009. <http://www.openfoam.com>.

[21] Basa M, Quinlan NJ, Lastiwka M. Robustness and accuracy of SPH formulations for viscous flow. Int J Numer Methods Fluids 2009;60:1127–48.

[22] Lee E-S, Moulinec C, Xu R, Violeau D, Laurence D, Stansby P. Comparisons of weakly compressible and truly incompressible algorithms for the SPH mesh free particle method. J Comput Phys 2008;227(18):8417–36.

[23] Colagrossi A, A meshless Lagrangian method for free-surface and interface flows with fragmentation. Universitá di Roma La Sapienza. PhD thesis. <http://padis.uniroma1.it/>.

[24] Monaghan JJ. Smoothed particle hydrodynamics. Annu Rev Astron Astrophys 1992;30:543–74.

[25] Lo EYM, Shao S. Simulation of near-shore solitary wave mechanics by an incompressible SPH method. Appl Ocean Res 2002;24(5):275–86.

[26] Koshizuka S, Nobe A, Oka Y. Numerical analysis of breaking waves using the moving particle semi-implicit method. Int J Numer Methods Fluids 1998;26(7):751–69.

[27] Rafiee A, Thiagarajan KP. An SPH projection method for simulating fluid-hypoelastic structure interaction. Comput Methods Appl Mech Eng 2009;198(33–36):2785–95.

[28] Boroomand B, Tabatabaei A, Onate E. Simple modifications for stabilization of the finite point method. Int J Numer Methods Eng 2005;63(3):351–79.

[29] Le Touzé D, Colagrossi A. Free-surface prototype problems suitable to investigate particle methods. In: 20th International workshop on water waves and floating bodies; 2005.

[30] Shampine L, Reichelt M. The matlab ode suite. SIAM J Sci Comput 1997;18(1):1–22.

[31] Fang J, Parriaux A, Rentschler M, Ancey C. Improved SPH methods for simulating free surface flows of viscous fluids. Appl Numer Math 2009;59(2):251–71.

[32] Oger G, Leroy C, Jacquin E, Touzé DL, Alessandrini B. Specific pre/post treatments for 3-D SPH applications through massive HPC simulations. In: Proceedings of the 4th SPHERIC, workshop; 2009.

[33] Ataie-Ashtiani B, Shobeyri G. Numerical simulation of landslide impulsive waves by incompressible smoothed particle hydrodynamics. Int J Numer Methods Fluids 2008;56(2):209–32.

[34] Abadie S, Morichon D, Grilli S, Glockner S. VOF/Navier–Stokes numerical modeling of surface waves generated by subaerial landslides. La Houille Blanche 2008;1:21–6.

[35] Faltinsen OM, Timokha AN. Sloshing. Cambridge University Press; 2009. ISBN: 978-0-521-88111-1.

[36] Delorme L, Colagrossi A, Iglesias AS, Rodrìguez RZ, Vera EB. A set of canonical problems in sloshing, Part I: pressure field in forced roll-comparison between experimental results and SPH. Ocean Eng 2009;36(2):168–78.

[37] Iglesias AS, Rojas LP, Rodrìguez RZ. Simulation of anti-roll tanks and sloshing type problems with smoothed particle hydrodynamics. Ocean Eng 2004;31(8–9):1169–92.

[38] Iglesias AS, Delorme L, Rojas LP, Pèrez SA. Liquid moment amplitude assessment in sloshing type problems with smooth particle hydrodynamics. Ocean Eng 2006;33(11–12):1462–84.

[39] Pákozdi C. A smoothed particle hydrodynamics study of two-dimensional nonlinear sloshing in rectangular tanks. PhD thesis. Norwegian University of Science and Technology; 2008.

[40] Rafiee A, Thioagarajan KP, Monaghan JJ. SPH simulation of 2D sloshing flow in a rectangular tank. In: The proceedings of the nineteenth international offshore and polar engineering conference; 2009.

[41] Rognebakke O. Sloshing in rectangular tanks and interaction with ship motions PhD thesis. Norwegian University of Science and Technology; 2002.