A multi-phase particle shifting algorithm for SPH simulations of violent hydrodynamics with a large number of particles

ATHANASIOS MOKOS, Research Associate, School of Mechanical, Aerospace and Civil Engineering, University of Manchester, Manchester, UK
Email: athanasios.mokos@manchester.ac.uk (author for correspondence)

BENEDICT D. ROGERS, Reader, School of Mechanical, Aerospace and Civil Engineering, University of Manchester, Manchester, UK
Email: benedict.rogers@manchester.ac.uk

PETER K. STANSBY (IAHR Member), Osborne Reynolds Professor of Fluid Mechanics, School of Mechanical, Aerospace and Civil Engineering, University of Manchester, Manchester, UK
Email: p.k.stansby@manchester.ac.uk

ABSTRACT
A numerical inconsistency has emerged for multi-phase smoothed particle hydrodynamics simulations when using very high resolution, made possible by graphical processing units. In violent flows unphysical voids and phase separation occur ultimately leading to numerical instability. New Fickian-based particle shifting algorithms with a selectively activated free-surface correction are developed for air–water simulations to prevent the creation of unnatural voids and maintain numerical stability through nearly uniform distributions. Using the shifting algorithm without surface correction in the air phase is recommended, with marginal improvements if the shifting algorithm is not applied in water. However, maintaining shifting in water would avoid possible void occurrence. The improvement is demonstrated using a dry-bed dam break and a sloshing tank case. A 3D case involving the impact of the water flow on an obstacle is compared with experimental data. The multi-phase SPH scheme gives closer agreement with experiment than a single-phase simulation.

Keywords: Air–water interface interaction; gravity wave; interfacial flow; multiphase and stratified flow; smoothed particle hydrodynamics model; three-dimensional model

1 Introduction
Multi-phase flows with mixing and violent free-surface hydrodynamic interaction exist in various industrial and research fields such as coastal and nuclear engineering. They include a diverse range of problems such as overturning or breaking waves and multi-phase pipe flow and are an ideal application of smoothed particle hydrodynamics (SPH) (Gingold & Monaghan, 1977). Including multiple phases in SPH is relatively straightforward, as it is possible to assign a separate set of particles to each phase with minimal treatment of the interface (Colagrossi & Landrini, 2003; Grenier, Antuono, Colagrossi, Le Touzé, & Alessandrini, 2009; Hu & Adams, 2006).

A major disadvantage of SPH is its extremely high computational requirements, especially when considering 3D multi-phase flows. To reduce the computational run times, the use of graphics processing units (GPUs) has emerged as a viable option for accelerating the SPH simulations (Dominguez, Crespo, & Gomez-Gesteira, 2013; Hérault, Bilotta, & Dalrymple, 2010) enabling the modelling of very large numbers of particles (Dominguez, Crespo, Gomez-Gesteira, & Rogers, 2013). The large number of multi-processors on a GPU enables speed-ups close to two orders of magnitude compared to an optimized single-thread CPU code (Crespo, Dominguez, Barreiro, Gomez-Gesteira, & Rogers, 2011) and has already been used for accelerating multi-phase flows with significant speed-ups (Mokos,
Rogers, Stansby, & Dominguez, 2015). However, in this paper a number of previously unreported issues have been identified at finer resolutions. Specifically, the creation of voids within the lighter air phase is observed, especially with high flow velocities and in flows with entrainment.

As a solution to this issue, this paper proposes using a modification of the shifting algorithm, initially presented by Xu, Stansby, and Laurence (2009) within a divergence-free incompressible SPH approach to prevent the instability caused by anisotropic particle spacing. The algorithm was improved by Lind, Xu, Stansby, and Rogers (2012) who used Fickian diffusion to provide shifting towards areas with lower concentration extending to free-surface flows. This algorithm leads to improved accuracy, smoother pressure fields and numerical stability in the simulation. For compressible flows, both Shadloo, Zainali, Yildiz, and Suleman (2012) and Vacondio, Rogers, and Fekken, Veldman, Iwanowski, & Buchner (2005) have used similar shifting approaches, whereas Tsuruta, Khayyer, and Gotoh (2015) proposed the use of space voids in the flow. A more detailed investigation on the effects on the interface is performed. The code is compared with a BEM algorithm using a dam break simulation. Finally, the interface is validated with experimental results for a 2D sloshing tank case and a level-set algorithm using a level-set algorithm by Colicchio, Landrini, and Chaplin (2005). The algorithm has been extended to 3D obstacle impact case (Botia-Vera, Souto-Iglesias, Bulian, & Lobovsky, 2010; Souto-Iglesias, Botia-Vera, Martin, & Perez-Arribas, 2011). The new shifting algorithm has also been expanded to 3D space with results presented for a wave impact case (Kleebsman, Fekken, Veldman, Iwanowski, & Buchner, 2005).

This paper is structured as follows: first the unmodified multi-phase SPH formulation is presented. Then, we highlight the issues for finer resolutions and present new shifting algorithms to provide numerical stability and eliminate unphysical voids in the flow. A more detailed investigation on the effects on the interface is performed. The code is compared with a BEM and a level-set algorithm using a dam break simulation. Finally, it is validated with experimental results for a 2D sloshing tank case and a 3D obstacle impact case.

2 Multi-phase SPH formulation

2.1 Governing equations

The multi-phase flows considered here can be described by the Navier–Stokes equations for the conservation of mass and momentum. Expressed in Lagrangian form the governing equations are:

\[
\frac{d\rho}{dt} = -\rho \nabla \cdot \mathbf{u} \\
\frac{d\mathbf{u}}{dt} = -\frac{1}{\rho} \nabla p + \mathbf{g} + \frac{1}{\rho} (\nabla \cdot \mathbf{v}) \mathbf{u}
\]

where \(\rho\) is density, \(v\) is laminar viscosity, \(\mathbf{u}\) is velocity, \(p\) is pressure, \(\mathbf{g}\) is gravity and \(t\) is time.

The SPH formulation of Colagrossi and Landrini (2003) is used as it has successfully simulated violent air–water mixtures (Rogers, Leduc, Marongiu, & Leboeuf, 2009) and its implementation on a GPU is relatively straightforward (Mokos et al., 2015). For an air–water mixture the Colagrossi and Landrini (2003) multi-phase model follows the work of Nugent and Posch (2000), who propose the use of a modified version of Tait’s equation of state (Batchelor, 1967) for incompressible and inviscid fluids:

\[
p(\rho) = c_s^2 \rho_0 \left[ \left( \frac{\rho}{\rho_0} \right)^\gamma - 1 \right] + X - a\rho^2
\]

where \(\gamma\) is the isentropic expansion factor, \(\rho_0\) is the initial density of the fluid, \(c_s\) is the speed of sound, \(X\) signifies a constant background pressure, while the last term represents the cohesion forces between the particles of a single phase. As proposed by Nugent and Posch (2000), the coefficient is based on the properties of the different phases and a characteristic length scale of the problem, \(L\):

\[
a = 1.5 g \left( \frac{\rho_w}{\rho_a} \right) L
\]

where \(\rho_w\) and \(\rho_a\) are the initial densities of the two phases (water and air in this case).

2.2 SPH formulation

As a Lagrangian method, SPH simulates the domain as a set of discrete particles. The value of the function \(A\) at each particle is interpolated with a weighting function called the smoothing kernel \(W\). The kernel depends on the pairwise particle distance and a characteristic length referred to as the smoothing length \(h\). The present study will use the quintic Wendland kernel:

\[
\langle A(r) \rangle = \int_{\Omega} W(r - r', h) A(r')d\Omega
\]

where \(r\) is a position vector, \(\Omega\) is the interpolation domain, and \(\langle \rangle\) represents the approximation. In a discrete SPH form, this is
The zeroth-order Shepard filter (Shepard, 1968) is applied everywhere $N_j$ and $m_j W_{ij}$ 20 time steps for each phase independently: the interface (Colagrossi & Landrini, 2003). Using Eq. (3) as tum equations as the standard SPH formulation is not applicable pressure gradient are used to update the continuity and momen-
(c) and for testing its application in impulsively-started, rapidly-
evolving free-surface flows (Koshizuka & Oka, 1996). The 
simulated using the dynamic boundary condition method (Crespo, Gomez-Gesteira, & Dalrymple, 2007), which represents solid walls using stationary liquid particles.

3 Multi-phase shifting algorithms

3.1 Unphysical void formation

While the previous algorithm has been shown to provide close agreement for a dry-bed dam break test case (Mokos et al., 2015), an issue arose when simulating large numbers of particles (over 200,000 fluid particles corresponding to 25,000 water particles or a resolution of $dx/h_0 = 0.008$) which persists even when using finer resolutions to model the case. The issue is demonstrated here using a dry dam break case as used by Mokos et al. (2015). The dry dam break case is a well-used benchmark for demonstrating the robustness of many SPH schemes and for testing its application in impulsively-started, rapidly-evolving free-surface flows (Koshizuka & Oka, 1996). The definition sketch for the case is shown in Fig. 1 with dimensions $H = L = 4 \text{ m}$, $l_0 = 1 \text{ m}$ and $h_0 = 2 \text{ m}$. (CFL) condition:

$$\Delta t = C_{CFL} \cdot \min(\delta t_f, \delta t_v)$$

(13)

The restrictions are imposed by the inter-particle forces $f_i$ shown in Eq. (14) or the viscous conditions in Eq. (15) (Mon-
agham & Kos, 1999):

$$\delta t_f = \min_i \left( \frac{\sqrt{h_i}}{|f_i|} \right)$$

(14)

$$\delta t_v = \min_i \left( \frac{h_i}{c_{ij} \max + \max |\sigma_{ij}|} \right) \text{ where } \sigma_{ij} = \left| \frac{h_i u_i \cdot r_{ij}}{r_{ij}^2} \right|$$

(15)

For the simulations presented herein, we use the DualSPHysics code (Crespo et al., 2015) that was modified and optimized for multi-phase SPH simulations on a GPU as proposed by Mokos et al. (2015). In the DualSPHysics code, the boundaries are sim-
ulated using the dynamic boundary condition method (Crespo, Gomez-Gesteira, & Dalrymple, 2007), which represents solid walls using stationary liquid particles.

To ensure the numerical stability of the scheme, the artificial viscosity term by Monaghan and Pongracic (1985), which simulates the effects of the viscous term in the momentum equation will be used:

$$\Pi_{ij} = -\frac{\alpha_v \tilde{c}_{ij} \mu_{ij}}{\tilde{\rho}_{ij}} \text{ if } (u_i - u_j) \cdot (r_i - r_j) < 0$$

(11)

where $\alpha_v$ is a numerical parameter, $\tilde{c}_{ij} = \frac{1}{2}(c_i + c_j)$, $\tilde{\rho}_{ij} = \frac{1}{2}(\rho_i + \rho_j)$, and $\mu_{ij}$ is given by:

$$\mu_{ij} = \frac{h(u_i - u_j) \cdot r_i}{r_{ij}^2 + 0.01h^2}$$

(12)

Due to the violent nature of the flows investigated here, the iner-
tial forces are significantly larger that the surface tension forces; the latter will not be simulated. To march the scheme forward in time a Verlet formulation (Verlet, 1967) is used. The time step used for this scheme is defined by a Courant–Friedrichs–Lewy approximation as (Monaghan, 2005; Violeau & Rogers, 2016):

$$\langle A(r) \rangle = \sum_{j=1}^{N} W(r - r_j, h) A_j m_j \rho_j$$

(6)

where $N$ is the number of particles within the support domain of $W$, and $m_j \rho_j$ is the volume of the $j$th particle where $m$ is mass.

Variationally consistent forms of the velocity divergence and pressure gradient are used to update the continuity and momentum equations as the standard SPH formulation is not applicable for air–water flows due to the large density discontinuity at the interface (Colagrossi & Landrini, 2003). Using Eq. (3) as the equation of state leads to the use of an extra term for the cohesion forces in the momentum equation for the lighter phase (Colagrossi & Landrini, 2003):

Continuity: \[ \frac{d\rho_i}{dt} = \rho_i \sum_j \left[ (u_i - u_j) \cdot \nabla_i W_{ij} \frac{m_j}{\rho_j} \right] \]

Momentum (water phase): \[ \frac{dm_i}{dt} = - \sum_j m_j \times \left( \frac{p_i + p_j}{\rho_i \rho_j} + \Pi_{ij} \right) \nabla_i W_{ij} \]

Momentum (air phase): \[ \frac{dm_i}{dt} = - \sum_j m_j \left( \frac{p_i + p_j}{\rho_i \rho_j} + \Pi_{ij} \right) \times \nabla_i W_{ij} - 2 \alpha_i \mu_{ij} \sum_j m_j \frac{m_j}{\rho_j} \nabla_i W_{ij} \]

(9)

The zeroth-order Shepard filter (Shepard, 1968) is applied every 20 time steps for each phase independently:

$$\langle \rho_i \rangle = \sum_j \rho_j \tilde{W}_{ij} \text{ where } \tilde{W}_{ij} = \frac{W_{ij}}{\sum_j W_{ij}(m_j/\rho_j)}$$

(10)

To ensure the numerical stability of the scheme, the artificial viscosity term by Monaghan and Pongracic (1985), which simulates the effects of the viscous term in the momentum equation will be used:

$$\Pi_{ij} = -\frac{\alpha_v \tilde{c}_{ij} \mu_{ij}}{\tilde{\rho}_{ij}} \text{ if } (u_i - u_j) \cdot (r_i - r_j) < 0$$

(11)

where $\alpha_v$ is a numerical parameter, $\tilde{c}_{ij} = \frac{1}{2}(c_i + c_j)$, $\tilde{\rho}_{ij} = \frac{1}{2}(\rho_i + \rho_j)$, and $\mu_{ij}$ is given by:

$$\mu_{ij} = \frac{h(u_i - u_j) \cdot r_i}{r_{ij}^2 + 0.01h^2}$$

(12)

Due to the violent nature of the flows investigated here, the iner-
tial forces are significantly larger that the surface tension forces; the latter will not be simulated. To march the scheme forward in time a Verlet formulation (Verlet, 1967) is used. The time step used for this scheme is defined by a Courant–Friedrichs–Lewy approximation as (Monaghan, 2005; Violeau & Rogers, 2016):
Results are shown here for three different particle resolutions of $dx/h_0 = 0.016, 0.008$ and $0.004$ where $dx$ is the initial particle spacing. The problem appears after the impact of the water flow on the opposite wall as shown in Fig. 2. After the wave overturns an air pocket is formed. However, the air particles within the air pocket do not adapt to the constantly evolving volume and shape of the bubble leading to the generation of voids, which persist until the pocket disappears. The problem does not appear in Fig. 2a because of the coarse resolution; it begins appearing in Fig. 2b and is especially visible in Fig. 2c.

The formation of the voids is due to the treatment of the air phase as a compressible fluid using Eq. (3), whereby it will not expand to fill the void. The behaviour is reinforced by the use of the cohesion force which maintains a consistent phase interface. The problem is not apparent in coarser resolutions due to the increased particle volume and kernel radius which allows each particle to have greater movement flexibility, covering larger areas with fewer particles. This is especially important for the air pocket, whose dimensions depend on the flow, not the resolution. A secondary issue also shown in Fig. 2 is the empty $2h$ zone around isolated water particles within the air flow occurring for the same reasons.

The unphysical voids are observed here while using the Colagrossi and Landrini (2003) multi-phase model. The inherent complexity of multi-phase simulations coupled with the large computational demands of SPH have made observing the issue for more complicated models a difficult task; however, it has been recently reported (Ghaitanellis, Violeau, Leroy, Joly, & Ferrand, 2015) that issues with anisotropic particle density occur while using a particle density based multi-phase model (Hu & Adams, 2006). In general, similar voids are expected with any multi-phase model using the weakly compressible SPH formulation for the air phase, regardless of the interface treatment.

To eliminate the voids within the SPH simulation, the air particles should move to lower particle concentration areas adapting their volume, as expected by a gas. The SPH particle shifting algorithms developed by Xu et al. (2009) and improved by Lind et al. (2012) and Skillen, Lind, Stansby, and Rogers (2013) appear ideal for this situation to maintain a more uniform particle arrangement.

### 3.2 Shifting algorithm formulation

Xu et al. (2009) proposed a particle shifting algorithm to prevent the instability caused by anisotropic particle spacing for an incompressible SPH model. This is a non-conservative approach due to the interpolation of the hydrodynamic variables when shifting. An improvement was proposed by Lind et al. (2012) who used Fick’s law to control the shifting magnitude and

---

**Figure 2** Dry dam break water flow at the lower right part of the domain for three different resolutions (a) $dx/h_0 = 0.016$, (b) $dx/h_0 = 0.008$, (c) $dx/h_0 = 0.004$ after 2.2 s (black particles denote air particles and grey particles denote water particles)
direction. Using concentration gradients, the particle shifting distance \( \delta r_s \) is given by:

\[
\delta r_s = -D \nabla C_i
\]  

(16)

where \( D \) is a diffusion coefficient that controls the shifting magnitude and \( C \) is the particle concentration which can be estimated using the sum of the smoothing kernel:

\[
C_i = \sum_j \frac{m_j}{\rho_j} W_{ij}.
\]  

(17)

The gradient of the particle concentration can be found using a standard SPH gradient approximation through the smoothing kernel (Monaghan, 2005; Violeau & Rogers, 2016):

\[
\nabla C_i = \sum_j \left( C_j - C_i \right) \frac{m_j}{\rho_j} \nabla W_{ij}.
\]  

(18)

To evaluate the diffusion coefficient the approach proposed by Skillen et al. (2013) was used, which imposes a restriction based on the particle velocity magnitude:

\[
D = A_s h ||u|| \Delta t,
\]  

(19)

where \( A_s \) is a parameter and \( ||u|| \) is the velocity magnitude of particle \( i \). The parameter value is in the range \([1,6]\) with a value of 2 recommended by Skillen et al. (2013). To prevent the unphysical movement at the free surface Lind et al. (2012) proposed that the concentration gradient near the surface is controlled using the local tangent and normal vectors at the free surface:

\[
\delta r_s = -D \left( \frac{\partial C_i}{\partial s} s + \alpha_n \left( \frac{\partial C_i}{\partial n} - \beta_n \right) n \right),
\]  

(20)

where \( s \) and \( n \) are the tangent and normal vectors to the free surface in 2D, while \( \beta_n \) is a reference concentration gradient (usually taken as the initial value). The parameter \( \alpha_n \) limits the diffusion in the direction normal to the free surface; for violent flows, such as the ones simulated in this article, \( \alpha_n \) is set equal to 0. However, for long slow flows, such as standing gravity waves, errors can potentially accumulate at the free surface due to the incompleteness of the kernel (Lind et al., 2012). A small degree of diffusion at the normal direction is then allowed with the parameter \( \alpha_n \) set equal to 0.1.

To identify the free surface, threshold values for the divergence of the particle position are used according to Lee et al. (2008):

\[
\nabla \cdot r = \sum_j \frac{m_j}{\rho_j} r_{ij} \cdot \nabla_i W_{ij}
\]  

(21)

The value of the position divergence for a kernel with complete support is equal to 2 for a 2D case and 3 for a 3D case; the value of \( \nabla \cdot r \) for a particle at the free surface is naturally much less.

The values of 1.5 and 2.5 are proposed as threshold values for a 2D and a 3D computation, respectively, to locate the surface particles. It is possible for some free surface particles to be ignored (Lee et al., 2008), however the error appears to be sufficiently small to be neglected (Lind et al., 2012).

Both Xu et al. (2009) and Lind et al. (2012) used a correction to the fluid velocities after shifting based on a linear interpolation. This was tested here, but the results showed that the effect, especially for high resolutions, is negligible confirming the conclusions of Vacondio, Rogers, Stansby, Migliorini, and Feldman (2013) for weakly compressible SPH. A similar conclusion was reached for correcting the density (Vacondio et al., 2012).

To impose boundary conditions, the present study uses the dynamic boundary particles method (Crespo et al., 2007) which represents solid walls using stationary fluid particles. Their inclusion in the calculation of the concentration becomes necessary to prevent fluid particles from being shifted towards the boundary. The use of a second row of boundary particles is also recommended to lower the concentration gradient for fluid particles near the edge of the domain. Hence, Eq. (18) is modified as follows:

\[
\nabla C_i = \sum_{j \in F \cup B} \left( C_j - C_i \right) \frac{m_j}{\rho_j} \nabla W_{ij}
\]  

(22)

where \( F \) and \( B \) denote the set of fluid and boundary particles respectively.

The shifting algorithm of Lind et al. (2012) uses a tensile instability term (Monaghan, 2000) to supplement the shifting algorithm in order to compensate for the discontinuities of the quintic spline kernel used in the incompressible SPH algorithm. The present study however, uses the continuous quintic Wendland kernel so this correction is not used.

3.3 Modified particle shifting algorithms for multi-phase weakly compressible SPH

3.3.1 Surface behaviour of particle shifting algorithms for a circular patch

When treated with the particle shifting algorithms of section 3.2, the behaviour of the air particles can be further examined by creating an artificial test case: an initially quiescent circular volume of air particles. This case is a single-phase test case; only the circular air volume shown in Fig. 3a is considered. A pressure difference is created between the volume and the surrounding area, forcing the air volume to gradually expand. The aim of this case is to demonstrate the effect of the shifting algorithm on the anisotropic particle spacing and test the applicability of the free surface correction to the air phase. Figure 3 shows snapshots for the different particle shifting algorithms for a constant background pressure of 50 Pa applied to 20,000 particles with initial particle spacing \( dx = 0.01 \text{ m} \).

Figure 3b shows the non-uniform expansion of the air particles without any shifting algorithm, where large voids are
Figure 3  Multi-phase model issue demonstrated in an artificial test case following the expansion of an air volume of radius 1 m and $dx = 0.01$ m. All particles have a constant background pressure of $X = 50$ Pa

created due to particles following streamlines. Figure 3c shows that applying the shifting algorithm of Xu et al. (2009) of Eq. (16) allows a consistent expansion in each direction without any voids. The asymmetrical expansion of the outer particles is due to the incomplete kernel and the single precision of the GPU.

Figure 3d shows that if the shifting free-surface correction (FSC) of Eq. (20) of Lind et al. (2012) is used, the air volume does not expand, maintaining the initial interface which is not desired for the density ratios used here. Consequently, when simulating the multi-phase dam break of section 3.1 using the shifting algorithm with the FSC the voids still remain as shown in Fig. 4, with the only advantage being a smoother interface. This poses an interesting question regarding the optimal configuration of a shifting algorithm for a multi-phase simulation.

### 3.3.2 Multi-phase particle shifting algorithms

The new multi-phase particle shifting algorithm is now developed. As stated in Eq. (17), the concentration is calculated across all particles. If the particles of the other phase are not included in calculating the concentration, the shifting algorithm is not able to distinguish between voids and particle-occupied areas as it is executed semi-independently of the SPH simulation. As a result, particles of different phases would move through each other, destroying the interface.

However, it was observed that, if particle shifting is activated in both phases without the FSC, the water flow adapts to an incorrect interface profile, subsequently altering the positions of the air particles, a problem also encountered by Lind et al. (2012). Two reasons were identified leading to that behaviour. The first is that identifying the normal vector on the interface
in SPH is not perfect, especially in violent flows. The second is the anisotropic particle spacing in the interface. Due to the multi-phase treatment and the complicated flow geometries, the particle distribution will not be consistent, leading to particles being shifted in the normal direction to the interface. This is desirable for the gas phase but will alter the profile of the water flow leading to the incorrect results observed (Lind et al., 2012). It can therefore be concluded that for the water phase, the surface correction is essential in restricting the water movement and preventing an unnatural expansion.

For the air phase, the phase expansion is the desired result and as shown in Fig. 3d, the FSC actively prevents the movement of air particles towards areas of lower concentration. Hence, the correction will not be used for the air phase, but only for the water phase. Figure 5 shows the dam break simulation for this configuration (FSC only in the water phase). No voids are occurring in the air pocket due to the air particles moving freely and eliminating the voids. Furthermore, the 2h separation around individual water particles moving through the air has also been eliminated.

While the improvements of the shifting algorithm to the air phase are quite significant, the water flow profile is very similar to Fig. 2. Figure 6 displays a simulation where the shifting algorithm was only applied to the air particles (with no FSC). The water profile is similar to Fig. 5 with no voids appearing in the air pocket or within the air flow. The secondary splash-up caused by the plunging water wave is slightly different, with the interface being fractured by the air phase. The splash-up traps a second volume of air and minor separation can be seen between the two phases at that point. Bearing in mind that a real dam break will never be repeated identically, Figs 5 and 6 now raise the question of which combinations of the shifting algorithms are appropriate requiring more precise validation.

4 Validation results

The results presented so far show that the shifting algorithm should always be used for the air phase without the surface correction term to ensure a smooth expansion of the air particles. They do not, however, offer conclusive evidence on the best approach for the water phase. The following cases, simulating multi-phase flows of water (initial density 1000 kg m$^{-3}$, isentropic expansion factor 7) and air (initial density 1.18 kg m$^{-3}$,
isentropic expansion factor 1.4), will further investigate these two approaches and their respective differences. To facilitate discussion we refer to the different variants of the shifting algorithms as follows:

- no shifting (NS) in either phase;
- shifting algorithm 1 (SA1): shifting only in the air phase;
- shifting algorithm 2 (SA2): shifting in both phases but with the surface correction only activated in the water phase.

4.1 Dry dam break

4.1.1 Case description

Herein, we use the data from the incompressible boundary element model (BEM) simulation by Greco et al. (2004) used by Colagrossi and Landrini (2003) for their validation. With the definition sketch shown in Fig. 1, we use the same geometry and numerical parameters as Colagrossi and Landrini (2003): \( l_0/h_0 = 2, H/h_0 = 3 \) and \( L/h_0 = 5.366 \).

4.1.2 Comparison with a BEM simulation

A comparison with the BEM simulation (Greco et al., 2004) is done for a simulation with a particle number close to 160,000 and a smoothing length given by \( h/h_0 = 1.3 \times 10^{-2} \). Figure 7 shows the results at \( t = 1.9 \) s (or \( t\sqrt{g/h_0} = 5.95 \)) for the different shifting algorithms NS, SA1 and SA2 for the two cases of viscosity coefficient simulated by Colagrossi and Landrini (2003) \( \alpha_v = 0 \) or \( 0.03 \). In each case, the pressure field of the water particles is displayed. Perfect agreement of the correct SPH results and the BEM results is not expected as the latter is a strictly incompressible model.

The use of the artificial viscosity leads to a shorter reflected wave. Without it, the shape of the plunging wave has a good agreement to the BEM solution with only the upper part of

![Figure 7](image-url)
the wave being underestimated. The viscosity also leads to the appearance of voids with the NS algorithm as seen in Fig. 7a.

Shifting affects the shape of the plunging wave in a similar manner. The effects are lessened if the SA1 algorithm is used. The shifting algorithm appears to increase the forces exerted in the plunging wave by the air particles and facilitates the downward movement of the water particles in the wave toe.

An increased pressure can be observed for SA2 in the bottom right-hand corner in Fig. 7c although the use of the artificial viscosity model limits the increase. A slight increase is also observed in Fig. 7b for algorithm SA1. The inclusion of the shifting algorithm therefore alters the shape of the overturning wave and affects the pressure field of the simulation leading to an increase in pressure. These effects are lessened if SA1 is used.

4.1.3 Comparison with a level-set algorithm

The same case has been executed with a level-set algorithm by Colicchio et al. (2005). Their results will be compared with SPH simulations using different shifting algorithms at two times given by $t\sqrt{g/h_0} = 6.76$ and $t\sqrt{g/h_0} = 7.14$ displayed in Figs 8 and 9, respectively. The numerical and physical parameters are the same as in the comparison with the BEM simulation. With the exception of the NS results, the results from the level-set algorithm are in generally good agreement with the SPH simulations with only the water flow height in the right side being slightly underestimated. Using the artificial viscosity leads to a lower splash-up than predicted by the level-set algorithm.

Figure 8a shows the creation of voids if no shifting is used. Small voids are visible for $\alpha_v = 0$ while large voids are clear for $\alpha_v = 0.03$. The shifting algorithms eliminate the voids but show differences in the water profile. SA1 has the best agreement with the level-set algorithm, while SA2 shows a more well-defined splash-up, but also an increased pressure field. The artificial viscosity also significantly alters the SA2 profile.

Figure 9, showing the evolution of the flow at $t\sqrt{g/h_0} = 7.14$, confirms the previous observations. The SA2 water profile further deviates from the level-set simulation especially when viscosity is used. SA1 maintains an agreement with the reference result, showing only a slightly increased pressure field. Its water profile is very similar to the NS simulation but the voids are eliminated due to the shifting in the air phase.

In general, the shifting algorithms prevent the creation of voids within the air phase and a smooth interface is maintained. SA1 has a similar profile to both Eulerian and the NS simulations while SA2 shows significant differences and an increased pressure field. Using the artificial viscosity model leads to a lower water height especially in the splash-up area, but avoids the increase in the pressure field.

Figure 8  Dry dam break – comparison with a level-set algorithm (Colicchio et al., 2005) shown by the black squares at $t\sqrt{g/h_0} = 6.76$ for $\Delta x/h_0 = 0.001$ for (a) present model with no shifting (NS), (b) present model with shifting only in the air phase (SA1), (c) shifting in both phases (SA2)
4.2 Sloshing tank

4.2.1 Case description

A sloshing tank case with experimental data (Botia-Vera et al., 2010) is selected to validate the multi-phase SPH code. The experiments studied the water movement and the pressures on the tank walls which are significant especially when a flip-through effect is observed.

We present the case where the water level corresponds to 18% of the maximum height, selected to maximize the wall impact. Figure 10 shows the sensor position investigated in this article. The dimensions are $H = 0.508 \text{ m}$, $L = 0.9 \text{ m}$ and $h_0 = 0.093 \text{ m}$. The case will be modelled with the multi-phase model described in section 2 and results without the shifting algorithm will be compared to results with SA1 and SA2. The particle resolution is $dx/h_0 = 0.0215$ giving 25,000 water particles and 90,000 air particles. This case has been previously modelled with a single-phase SPH simulation by Leonardi, Manenti, and Sibila (2011) who found that the pressure results were very sensitive to flow parameters, especially the viscosity model. The artificial viscosity value ($\alpha_v = 0.01$) they proposed is used here.

4.2.2 Validation

Figure 11 shows that using the shifting algorithm has a significant effect in the air phase. Because of the large forces involved,

![Figure 11](image-url)
in the flow, the shifting algorithm used does not have a significant effect on the water profile; SA1 and SA2 give nearly identical profiles, so only the SA2 profile is shown. With no shifting (NS), Fig. 11a shows water particles separated from the flow; voids are created in their wake due to their high velocity after the impact. Particle shifting shown in Fig. 11b eliminates these voids and improves the treatment of the interface.

For the multi-phase case we found that the numerical speed of sound used for the equation of state also has a significant effect on the pressure results. A suitable value for the parameters was found through a still water simulation: the tank geometry in Fig. 10 was simulated without the sloshing movement for a significant length of time (~20 s). The values of the speed of sound can then be selected based on the agreement of the pressure with the hydrostatic pressure distribution. Using speeds of sound of $c_{s,w} = 20 \text{ m s}^{-1}$, $c_{s,a} = 150 \text{ m s}^{-1}$ in water and air respectively, Fig. 12 shows the pressure results for the first 3 s of the simulation.

The results are very similar between all the SPH algorithms, with or without shifting. Close agreement with the experimental results is achieved. Small differences can only be seen in the latter stages with the NS simulation having a slightly lower pressure after the impact and the SA2 simulation predicting a slightly lower pressure peak.

From Fig. 12 it is possible to conclude that the impact on the wall is dominated by the forces exerted in the liquid phase. The instantaneous impact and the large velocities reduce the impact of the air phase leading to similar pressure results, regardless of the simulation of the air phase.

4.3 Extension to 3D

For 3D cases, the surface correction term needs to be slightly modified as Eq. (20) is only suitable for a two-dimensional simulation. For a 3D case, the bi-tangent vector, $b$, also needs to be taken into consideration which allows shifting only on the tangent, $s$, and bi-tangent directions, treating the normal direction in the same way as the two-dimensional implementation:

$$\delta r_i = -D \left( \frac{\partial C_i}{\partial s} s + \frac{\partial C_i}{\partial b} b + \alpha_n \left( \frac{\partial C_i}{\partial n} - \beta_n \right) n \right)$$

where $b$ is a unit vector orthogonal to $s$ and $n$.

To demonstrate the performance of the 3D extension of the FSC for the water phase, a 3D dry-bed dam break simulation in a narrow tank is performed with no shifting (NS) and with shifting algorithm 2 (SA2) with SA1 showing near identical results. The simulation is performed with the same numerical and physical parameters as the validation case in section 4.1, with the artificial viscosity coefficient set to 0.03. The domain is the same as in section 3.2 but with a lateral width of 4 $h$ and the particle distance is $dx/h_0 = 0.002$. Figure 13 shows snapshots at $t \sqrt{g/h_0} = 6.76$ for the NS and SA2 algorithms. The issues described in section 3.1 are less pronounced for the third dimension with more particles in the support of each kernel. However, void formation is still evident in Fig. 13a with a separation at the interface for the entrained air flow. This separation is significantly reduced if the shifting algorithm is used as shown in Fig. 13b.

4.4 Obstacle impact

4.4.1 Case description

To investigate the ability of the code to model complex 3D cases as well as provide experimental validation the SPH multi-phase code will be compared to the experiments of Kleefsman et al. (2005). This case has been simulated using a single-phase SPH model (Lee, Violeau, Issa, & Ploix, 2010) and using GPU acceleration (Crespo et al., 2011; Rooney et al., 2011).

Figure 14 shows the experimental set-up where a lock gate holds a quiescent volume of water. The gate movement is instantaneous, so it will not be included in the simulation. Four probes
measure the water height, while the box is covered by eight pressure sensors shown in Fig. 14.

The experiment was performed in a tank open at the top. This is approximated as an open boundary where air particles are allowed to exit the domain and because of the short timescale of the simulation they do not re-enter. The outflow and hence inflow volumes of air are considered negligible.

To reduce the number of particles leaving the system, the domain is extended in the vertical direction (the height of the domain is doubled) and a zero normal pressure gradient is imposed on the top boundary.

4.4.2 Validation

Crespo et al. (2011) simulated this case using a single-phase DualSPHysics code and presented results for height probes H4, H3 and H2. Results from this study will be compared with their 100,000 particle simulation using the same particle resolution ($h = 3.075 \times 10^{-3}$ m). A convergence study for this case has already been presented by the authors (Mokos et al., 2015). The speed of sound in the water phase for this simulation is 30 m s$^{-1}$, about 15 times the wave celerity, while the speed of sound ratio between the two phases is 4. The Verlet time-stepping algorithm and SA1 are used.

Figure 15 shows the water height evolution for probes H4, H3, H2 for the single-phase DualSPHysics code, the multi-phase code using shifting algorithm SA1 and the experimental data (Kleefsman et al., 2005). As seen in Fig. 15, the multi-phase results show a better agreement with the experiment as the single-phase simulation consistently overestimates the height. The only difference is the delay of the appearance of the reflected wave, clearly seen for probe H4 after 2 s.
Figure 15  Obstacle impact – comparison of a multi-phase simulation using the SA1 algorithm and the single-phase results of Crespo et al. (2011) for $dx/h_0 = 0.0364$ with the experimental results of Kleefsman et al. (2005)

The overestimation of the height can be observed for both simulations and can be traced to the interaction with the boundaries (Crespo et al., 2007). The boundary forces lead to upward movement of the neighbouring fluid particles especially near the water toe where the number of water particles is reduced. The problem is diminished for the multi-phase simulation due to the presence of the air particles.

The obstacle impact case has also been simulated using different shifting algorithms. The results are presented for the same resolution and parameters as the simulation in Fig. 15 which corresponds to about 500,000 particles with the ratio between air and water particles being about 4:1. Figure 16 shows the results for the four height probes, including H1. The results are very similar, especially for the SA2 and SA1 simulations. The NS simulation shows small differences in the water profile, with an increased height predicted and a further delay for the reflected wave in probe H2.

The results for probe H1 show greater discrepancy with the experimental results. The recirculation of the water flow is delayed due to the fluid–boundary interaction and the gradual
water build-up and height variation observed in the experiment are not reproduced. This explains the delay in the appearance of the reflected wave observed for the other height probes. Issues with the fluid–boundary interaction occur due to particle inconsistencies (Fourtakas, Vacondio, & Rogers, 2015) and noise in the pressure field near the boundaries (Ferrand, Laurence, Rogers, Violeau, & Kassiotis, 2013). Consequently, the gradual water build-up and height variation observed in the experiment are not reproduced.

The pressure results for the different shifting algorithms are compared with the experiment in Fig. 17 for the four probes located on the face of the obstacle as shown in Fig. 14. The pressure computed at probe P1 with no shifting exhibits severe oscillations due to void formation as there are insufficient particles around the probe to capture the pressure. This effect is completely eliminated using the multi-phase shifting algorithms SA1 and SA2.

The results for pressure show reasonable agreement with the experimental results for all the probes, but the agreement is reduced as the vertical position of the probes increases, with the pressure peak not identified for P3 and P4. An underestimation of the pressure after the impact is visible after the impact for the higher probes, but they eventually converge to similar residual pressure. The multi-phase simulations show similar pressure results for all shifting algorithms as in the sloshing tank case. Differences can only be seen after the obstacle impact, where the pressure oscillates, creating peaks of different magnitude for each algorithm.

Further investigation of the case showed that the pressure values are greatly affected by the speed of sound (directly linked to the pressure via the equation of state). The effect is significantly less for the water height. Figure 18 shows the effect of choosing different speeds of sound, 30 m s$^{-1}$ or 60 m s$^{-1}$, in the water phase for two simulations using SA1. The speed of sound in the air phase is 200 m s$^{-1}$.

The results show that an increased speed of sound leads to small changes in the pressure outside of the pressure peak. Increasing the speed of sound reduces the density variations but smaller density changes have now an increased effect, as the weak compressibility of the fluid is altered. Similar differences occur if the speed of sound of the air is altered. In contrast, reducing the speed of sound to very low values results in phenomena such as the pressure peaks not being captured correctly.

With the weakly compressible formulation, the speed of sound is then a numerical parameter that should be carefully controlled depending on the case. It is especially important in this case due to the effect of the air phase in the water impact and
the subsequent mixing, which greatly affects the speed of sound (Peregrine & Thais, 1996). Here the SPH guideline regarding the relation between the maximum water velocity and the speed of sound (10–12 times higher) (Monaghan & Kos, 1999) is used. For the air phase a similar approach to the sloshing tank case in section 4.2 was used: a 2D still water simulation (identical geometry, but the water body is extended to cover the bottom of the tank) where the best speed of sound value was determined by comparing the pressure to the hydrostatic value.

A convergence study has been performed for algorithms SA1 and SA2. Results for the height of the water flow are plotted in Fig. 19 and for the pressure in Fig. 20. For brevity, the results are presented for a single probe. The agreement of the simulation is gradually improved for both algorithms, but algorithm SA1 shows better agreement for the height, evident in the latter
Figure 19 Obstacle impact – height convergence to the experimental results of Kleefsman et al. (2005) for three different resolutions for height probe H3 with domain height $H = 1.25$ m

Figure 20 Obstacle impact – pressure convergence to the experimental results of Kleefsman et al. (2005) for three different resolutions for pressure probe P1 with domain height $H = 1.25$ m

stages of the simulation in Fig. 19. The pressure results are quite similar with the pressure peak value in the medium resolution being the most significant difference.

Differences between the shifting algorithms are more evident in the lower resolution, especially for the pressure. For algorithm SA1, due to the lack of shifting, the water particles are pushed upwards by the boundaries, giving no reading for probe 1 which is very close to the bottom of the tank, whereas SA2 prevents this from occurring. Differences are also visible in Fig. 19, with SA1 showing a delayed reflected wave while SA2 only shows a gradual increase in height. The height differences are due to the larger volume of the particles; shifting them greatly smooths the interface.

Three snapshots of the simulation using algorithm SA1 are shown in Fig. 21, where the particles are coloured according to velocity showing the initial dam break, the impact and the flow rebounding off the far wall.

5 Conclusions

A new Fickian-based shifting algorithm for multi-phase SPH simulations accelerated using GPU has been presented. The investigation identified new problems with voids appearing in the gas phase. A modified shifting algorithm for multiple phases is used to treat this by shifting particles towards areas of lower concentration, reducing anisotropic particle spacing.

The free-surface term by Lind et al. (2012) was found to have a significant effect on the evolution of the flow, restricting the flow expansion of the water phase on the interface. It was removed from the air phase to allow for free expansion resulting in the elimination of the voids. Two algorithms were proposed, either shifting only in the air phase (SA1) or shifting in both phases but with the surface correction only activated in the water phase (SA2).
The algorithms were compared to a BEM and a level-set simulation for a dry dam break test case. SA1 shows similar results to the case without shifting and the Eulerian methods, while SA2 has a slightly different profile with an increased pressure field. The viscosity was found to have a significant effect on the reflected wave. A sloshing tank case was also tested, showing the importance of the shifting algorithm for the air phase. Close agreement with the experimental results was achieved for the pressure, but changing the shifting algorithm showed small differences.

The new multi-phase shifting algorithm with the free-surface correction in the water was extended to 3D showing a reduction in void formation. The algorithms were tested with a 3D obstacle impact case. Similar to the sloshing tank case, small differences in the pressure values were found, but the height results were improved with the use of the shifting algorithm. The pressure was also affected by the speed of sound used in the equation of state with larger values resulting in higher steady state pressure. Calibrating the numerical parameter of the speed of sound to correspond to the physics of the simulation, achieved here through a still water simulation, is recommended.

A convergence study showed generally closer agreement for the higher resolutions. Differences between SA1 and SA2 were small, with the former showing marginally better agreement for the water height.
Acknowledgements

The authors would like to thank colleagues at the University of Manchester for their assistance and at the Universidade de Vigo for helping in the development of this work.

Funding

The authors gratefully acknowledge the support of the Engineering and Physical Sciences Research Council (EPSRC) [EP/H003045/1] and the Natural Environment Research Council (NERC) [NE/J009202/1] for funding this work.

Notation

\[ b = \text{bi-tangent direction vector (–)} \]
\[ c_s = \text{speed of sound (m s}^{-1}) \]
\[ C = \text{particle concentration (–)} \]
\[ D = \text{particle shifting coefficient (m}^2) \]
\[ f = \text{force per unit mass (m s}^{-2}) \]
\[ g = \text{gravity acceleration (m s}^{-2}) \]
\[ h = \text{smoothing length (m)} \]
\[ h_0 = \text{initial water height (m)} \]
\[ m = \text{mass (kg)} \]
\[ n = \text{normal direction vector (–)} \]
\[ p = \text{pressure (Pa)} \]
\[ r = \text{position vector (m)} \]
\[ s = \text{tangent direction vector (–)} \]
\[ t = \text{time (s)} \]
\[ u = \text{velocity (m s}^{-1}) \]
\[ W = \text{smoothing kernel (m}^{-3}) \]
\[ X = \text{background pressure (Pa)} \]
\[ a = \text{cohesion coefficient (m}^2 \text{ s}^{-2}) \]
\[ a_v = \text{artificial viscosity coefficient (–)} \]
\[ a_n = \text{shifting factor in the normal direction (–)} \]
\[ \beta_n = \text{reference concentration gradient (m}^{-1}) \]
\[ \gamma = \text{isentropic expansion factor (–)} \]
\[ \mu = \text{dynamic viscosity (kg m}^{-1} \text{s}^{-1}) \]
\[ \nu = \text{laminar kinematic viscosity (m}^2 \text{s}^{-1}) \]
\[ \rho = \text{density (kg m}^{-3}) \]

ORCID

Athanasios Mokos ⓒ http://orcid.org/0000-0002-2323-1843
Benedict D. Rogers ⓒ http://orcid.org/0000-0002-3269-7979
Peter K. Stansby ⓒ http://orcid.org/0000-0002-3552-0810

References

Batchelor, G. K. (1967). An introduction to fluid dynamics. Cambridge: Cambridge University Press.
Botia-Vera, E., Souto-Iglesias, A., Bulian, G., & Lobovsky, L. (2010). Three SPH novel benchmark test cases for free surface flows. Paper presented at the 5th ERCOFTAC SPHERIC Workshop, Manchester, UK.
Colagrossi, A., & Landrini, M. (2003). Numerical simulation of interfacial flows by smoothed particle hydrodynamics. Journal of Computational Physics, 191(2), 448–475. doi:10.1016/S0021-9991(03)00324-3
Colicchio, G., Landrini, M., & Chaplin, J. R. (2005). Level-set computations of free surface rotational flows. Journal of Fluids Engineering-Transactions of the Asme, 127(6), 1111–1121. doi:10.1115/1.2062707
Crespo, A. C., Dominguez, J. M., Barreiro, A., Gomez-Gesteira, M., & Rogers, B. D. (2011). GPUs, a new tool of acceleration in CFD: Efficiency and reliability on smoothed particle hydrodynamics methods. Plos One, 6(6), e20685. doi:10.1371/journal.pone.0020685
Crespo, A. J. C., Dominguez, J. M., Rogers, B. D., Gomez-Gesteira, M., Longshaw, S., Canelas, R., . . . Garcia-Feal, O. (2015). DualSPHysics: Open-source parallel CFD solver based on Smoothed Particle Hydrodynamics (SPH). Computer Physics Communications, 187(0), 204–216. doi:10.1016/j.cpc.2014.10.004
Crespo, A. J. C., Gomez-Gesteira, M., & Dalrymple, R. A. (2007). Boundary conditions generated by dynamic particles in SPH methods. Cmc-Computers Materials & Continua, 5(3), 173–184.
Dominguez, J. M., Crespo, A. J. C., & Gomez-Gesteira, M. (2013). Optimization strategies for CPU and GPU implementations of a smoothed particle hydrodynamics method. Computer Physics Communications, 184(3), 617–627. doi:10.1016/j.cpc.2012.10.015
Dominguez, J. M., Crespo, A. J. C., Gomez-Gesteira, M., & Rogers, B. D. (2013). Simulating more than 1 billion SPH particles using GPU hardware acceleration. Paper presented at the 8th International SPHERIC Workshop, Trondheim, Norway.
Ferrand, M., Laurence, D. R., Rogers, B. D., Violeau, D., & Kassiotis, C. (2013). Unified semi-analytical wall boundary conditions for inviscid, laminar or turbulent flows in the meshless SPH method. International Journal for Numerical Methods in Fluids, 71(4), 446–472. doi:10.1002/Fld.3666
Fourtakas, G., Vaylonas, R., & Rogers, B. D. (2015). On the approximate zeroth and first-order consistency in the presence of 2-D irregular boundaries in SPH obtained by the virtual boundary particle methods. International Journal for Numerical Methods in Fluids, 78(8), 475–501. doi:10.1002/Fld.4026
Ghaitanellis, A., Violeau, D., Leroy, A., Joly, A., & Ferrand, M. (2015). Application of the unified semi-analytical wall boundary conditions to multi-phase SPH. Paper presented at the 10th International SPHERIC Workshop, Parma, Italy.
Gingold, R. A., & Monaghan, J. J. (1977). Smoothed particle hydrodynamics: Theory and application to non-spherical stars. Monthly Notices of the Royal Astronomical Society, 181(2), 375–389.
Greco, M., Landrini, M., & Faltings, O. M. (2004). Impact flows and loads on ship-deck structures. Journal of Fluids and Structures, 19(3), 251–275. doi:10.1016/j.jfluidstructs.2003.12.009

Grenier, N., Antuono, M., Colagrossi, A., Le Touzé, D., & Alessandri, B. (2009). An Hamiltonian interface SPH formulation for multi-fluid and free surface flows. Journal of Computational Physics, 228(22), 8380–8393. doi:10.1016/j.jcp.2009.08.009

Hérault, A., Bilotta, G., & Dalrymple, R. A. (2010). SPH on GPU with CUDA. Journal of Hydraulic Research, 48, 74–79. doi:10.1080/00221686.2010.9641247

Hu, X. Y., & Adams, N. A. (2006). A multi-phase SPH method for macroscopic and mesoscopic flows. Journal of Computational Physics, 213(2), 844–861. doi:10.1016/j.jcp.2005.09.001

Kleefsmann, K. M. T., Fekken, G., Veldman, A. E. P., Iwanowski, B., & Buchner, B. (2005). A Volume-of-Fluid based simulation method for wave impact problems. Journal of Computational Physics, 206(1), 363–393. doi:10.1016/j.jcp.2004.12.007

Koshizuka, S., & Oka, Y. (1996). Moving-particle semi-implicit method for fragmentation of incompressible fluid. Nuclear Science and Engineering, 123(3), 421–434.

Lee, E.-S., Moulinec, C., Xu, R., Violeau, D., Laurence, D., & Stansby, P. (2008). Comparisons of weakly compressible and truly incompressible algorithms for the SPH mesh free particle method. Journal of Computational Physics, 227(18), 8417–8436. doi:10.1016/j.jcp.2008.06.005

Lee, E.-S., Violeau, D., Issa, R., & Ploix, S. (2010). Application of weakly compressible and truly incompressible SPH to 3-D water collapse in waterworks. Journal of Hydraulic Research, 48, 50–60. doi:10.1080/00221686.2010.9641245

Leonardi, M., Manenti, S., & Sibila, S. (2011). 2D and 3D sloshing with SPH. Paper presented at the 7th International SPHERIC Workshop, Prato, Italy.

Lind, S. J., Stansby, P. K., & Rogers, B. D. (2016). Incompressible–compressible flows with a transient discontinuous interface using smoothed particle hydrodynamics (SPH). Journal of Computational Physics, 309, 129–147. doi:10.1016/j.jcp.2015.12.005

Lind, S. J., Xu, R., Stansby, P. K., & Rogers, B. D. (2012). Incompressible smoothed particle hydrodynamics for free-surface flows: A generalised diffusion-based algorithm for stability and validations for impulsive flows and propagating waves. Journal of Computational Physics, 231(4), 1499–1523. doi:10.1016/j.jcp.2011.10.027

Mokus, A., Rogers, B. D., Stansby, P. K., & Dominguez, J. M. (2015). Multi-phase SPH modelling of violent hydrodynamics on GPUs. Computer Physics Communications, 196, 304–316. doi:10.1016/j.cpc.2015.06.020

Monaghan, J. J. (2000). SPH without a tensile instability. Journal of Computational Physics, 159(2), 290–311. doi:10.1006/jcph.2000.6439

Monaghan, J. J. (2005). Smoothed particle hydrodynamics. Reports on Progress in Physics, 68(8), 1703–1759. doi:10.1088/0034-4885/68/8/101

Monaghan, J. J., & Kos, A. (1999). Solitary waves on a Cretean beach. Journal of Waterway Port Coastal and Ocean Engineering-Asce, 125(3), 145–155. doi:10.1061/(Asce)0733-950x(1999)125:3(145)

Monaghan, J. J., & Pongracic, H. (1985). Artificial viscosity for particle methods. Applied Numerical Mathematics, 1(3), 187–194. doi:10.1016/0168-9274(85)90015-7

Nugent, S., & Posch, H. A. (2000). Liquid drops and surface tension with smoothed particle applied mechanics. Physical Review E, Statistical Physics, Plasmas, Fluids, and Related Interdisciplinary Topics, 62(4 Pt A), 4968–4975. doi:10.1103/PhysRevE.62.4968

Peregrine, D. H., & Thais, L. (1996). The effect of entrained air in violent water wave impacts. Journal of Fluid Mechanics, 325, 377–397. doi:10.1017/S0022112096008166

Rogers, B. D., Leduc, J., Marongiu, J. C., & Leboeuf, F. (2009). Comparison and evaluation of multi-phase and surface tension models. Paper presented at the 4th SPHERIC Workshop, Nantes, France.

Roozey, E. A., Wei, W. Y., Irish, J. L., Weiss, R., Dalrymple, R. A., Herault, A., & Bilotta, G. (2011). Testing accuracy and convergence of GPUSPH for free-surface flows.

Shadloo, M. S., Zainali, A., & Yildiz, M. (2013). Simulation of single mode Rayleigh-Taylor instability by SPH method. Computational Mechanics, 51(5), 699–715. doi:10.1007/s00466-012-0746-2

Shadloo, M. S., Zainali, A., Yildiz, M., & Suleman, A. (2012). A robust weakly compressible SPH method and its comparison with an incompressible SPH. International Journal for Numerical Methods in Engineering, 89(8), 939–956. doi:10.1002/nme.3267

Shepard, D. (1968). A two dimensional interpolation function for irregularly spaced data. Paper presented at the 23rd ACM National Conference, Las Vegas, NV.

Skillen, A., Lind, S., Stansby, P. K., & Rogers, B. D. (2013). Incompressible Smoothed Particle Hydrodynamics (SPH) with reduced temporal noise and generalised Fickian smoothing applied to body-water slam and efficient wave-body interaction. Computer Methods in Applied Mechanics and Engineering, 265, 163–173.

Souto-Iglesias, A., Botía-Vera, E., Martín, A., & Perez-Arribas, F. (2011). A set of canonical problems in sloshing. Part 0: Experimental setup and data processing. Ocean Engineering, 38(16), 1823–1830. doi:10.1016/j.oceaneng.2011.09.008

Tsuruta, N., Khayyer, A., & Gotoh, H. (2015). Space potential particles to enhance the stability of projection-based particle methods. International Journal of Computational Fluid Dynamics, 29(1), 100–119. doi:10.1080/10618562.2015.1006130
Vacondio, R., Rogers, B. D., & Stansby, P. K. (2012). Smoothed Particle Hydrodynamics: Approximate zero-consistent 2-D boundary conditions and still shallow-water tests. *International Journal for Numerical Methods in Fluids*, 69(1), 226–253. doi:10.1002/Fld.2559

Vacondio, R., Rogers, B. D., Stansby, P. K., Mignosa, P., & Feldman, J. (2013). Variable resolution for SPH: A dynamic particle coalescing and splitting scheme. *Computer Methods in Applied Mechanics and Engineering*, 256, 132–148. doi:10.1016/j.cma.2012.12.014

Verlet, L. (1967). Computer “Experiments” on classical fluids. I. Thermodynamical properties of Lennard-Jones molecules. *Physical Review*, 159(1), 98–103. doi:10.1103/PhysRev.159.98

Violeau, D., & Rogers, B. D. (2016). Smoothed particle hydrodynamics (SPH) for free-surface flows: Past, present and future. *Journal of Hydraulic Research*, 54(1), 1–26. doi:10.1080/00221686.2015.1119209

Xu, R., Stansby, P., & Laurence, D. (2009). Accuracy and stability in incompressible SPH (ISPH) based on the projection method and a new approach. *Journal of Computational Physics*, 228(18), 6703–6725. doi:10.1016/j.jcp.2009.05.032

Zainali, A., Tofighi, N., Shadloo, M. S., & Yildiz, M. (2013). Numerical investigation of Newtonian and non-Newtonian multiphase flows using ISPH method. *Computer Methods in Applied Mechanics and Engineering*, 254, 99–113. doi:10.1016/j.cma.2012.10.005