Pairwise Force SPH Model for Real-Time Multi-Interaction Applications

Tao Yang, Ralph R. Martin, Ming C. Lin, *Fellow, IEEE*, Jian Chang, and Shi-Min Hu

Abstract—In this paper, we present a novel *pairwise-force smoothed particle hydrodynamics* (PF-SPH) model to enable simulation of various interactions at interfaces in real time. Realistic capture of interactions at interfaces is a challenging problem for SPH-based simulations, especially for scenarios involving multiple interactions at different interfaces. Our PF-SPH model can readily handle multiple types of interactions simultaneously in a single simulation; its basis is to use a larger support radius than that used in standard SPH. We adopt a novel anisotropic filtering term to further improve the performance of interaction forces. The proposed model is stable; furthermore, it avoids the particle clustering problem which commonly occurs at the free surface. We show how our model can be used to capture various interactions. We also consider the close connection between droplets and bubbles, and show how to animate bubbles rising in liquid as well as bubbles in air. Our method is versatile, physically plausible and easy-to-implement. Examples are provided to demonstrate the capabilities and effectiveness of our approach.

Index Terms—Smoothed particle hydrodynamics (SPH), pairwise force, surface tension, bubble animation, fluid simulation

1 Introduction

In computer graphics, interactions at interfaces between materials in different phases, or immiscible materials in the same phase, have been extensively investigated during the last decade. In this work, we focus on interfaces involving a liquid; thus commonly observed interfaces can be categorized into three classes: between a liquid and a gas, another liquid, or a solid. The gas is mostly considered as air in this paper unless otherwise specified. The interaction between a liquid and air leads to surface tension, which is the main cause of many well known visual effects, including the water crown formed when a droplet falls into a liquid. When two liquids interact, the result of the interaction depends on whether they are miscible or not. In the miscible case, Yang et al. [1] adopted a reactive stress term to describe the effect. In the immiscible case, researchers have mainly focused on high density ratios [2] and interfacial flows [3], [4], [6]. We avoid complex treatments for miscible flows and concentrate on the interactions between different immiscible flows. The interactions between a liquid and a solid are of two main types: fluid-solid coupling and adhesion. Fluid-solid coupling contributes to macroscopic movements while adhesion is caused by molecular forces and results in various wetting effects. In the real world, many scenarios involve interactions at multiple interfaces. For instance, when cracking an egg, there are simultaneous interactions between air, egg white, egg yolk, and the shell. To capture such phenomena, it is crucial to develop a versatile approach that can uniformly handle all types of inter-component interactions.

Interactions arise due to the force between any pair of particles; they can be divided into two types. If the particles belong to two different phases or components, this is an *interfacial interaction*. If the particles are from the same group phase or component, their interaction is called *cohesion*. In the graphics community, air particles are always ignored for particle-based fluid simulations as the interactions between air and a liquid or a solid are relatively insignificant. Both the interfacial interaction and the cohesion need to be considered in most multi-phase real-world scenrios; while only the cohesion is needed when capturing the interaction between a liquid and air, which is actually modeled in terms of a single liquid phase in SPH while it is a two-phase phenomenon.

The flow of multiple phases is commonly described by multi-phase Navier-Stokes equations. Grid-based Eulerian simulators can be used to solve this problem directly, but they require computationally expensive interface tracking techniques [3], [4], [5], [6], [7], [8], [9]. Smoothed particle hydrodynamics (SPH) has gained popularity as a particle-based method due to its mass-conservation and flexibility in handling topological changes. As many works show,

- *T. Yang is with the TNList, Department of Computer Science & Technology, Tsinghua University, Beijing 100084, China. E-mail: yangtao9009@gmail.com.*
- *R.R. Martin is with the School of Computer Science & Informatics, Cardiff University, Cardiff CF10 3XQ, United Kingdom. E-mail: ralph@cs.cf.ac.uk.*
- *M.C. Lin is with the Department of Computer Science, University of North Carolina at Chapel Hill, Chapel Hill, NC 27514. E-mail: lin@cs.unc.edu.*
- *J. Chang is with the National Centre for Computer Animation, Bournemouth University, Poole BH12 5BB, United Kingdom. E-mail: jchang@bournemouth.ac.uk.*
- *S.-M. Hu is with the Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China and the School of Computer Science and Informatics, Cardiff University, Cardiff CF10 3XQ, United Kingdom. E-mail: shimin@tsinghua.edu.cn.*

Manuscript received 12 Mar. 2017; revised 12 May 2017; accepted 14 May 2017. Date of publication 19 May 2017; date of current version 1 Sept. 2017.
(Corresponding author: Tao Yang.)
Recommended for acceptance by J. Lee.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TVCG.2017.2706289

SPH-based simulation is capable of simulating surface tension [10], [11], [12], [13], adhesion [12], [13], [14], and various other couplings [12], [15], [16]. While many works have used SPH, most of them focus on a particular type of interaction, which restricts their application to situations involving multiple complex phenomena. For instance, work addressing surface tension cannot be readily used to handle other kinds of interaction such as adhesion.

From a microscopic point of view, liquids, gases and solids are all composed of molecules. Physically, interaction forces exist between any pair of molecules, which are repulsive at short-range and attractive at long-range. It is crucial to accurately calculate these forces when handling interactions. Since standard SPH method simulates fluids at a macroscopic level, microscopic interaction forces are largely ignored. The *pairwise-force smoothed particle hydrodynamics* (PF-SPH) model was proposed to solve this problem by adding an extra molecular-like interaction force term between pairs of macroscopic SPH particles [17]. Some works (e.g., [11]) have considered simulating surface tension using this model. However, this approach has been criticized its relatively poor performance in computer graphics, and alternatives have been suggested based on surface area minimization [12], or surface energy minization in conjunction with air pressure [13]. The PF-SPH model has been successfully used in the last decade in computational physics to capture surface tension and adhesion [17], [18], [19], [20]. However, this PF-SPH model cannot be readily adopted in computer graphics as it fails to readily obtain plausible results as argued by [12], [13], [21].

The PF-SPH model is known for being easy-to-implement, cluster-preventing, momentum-conserving, and physically-plausible. In this paper, we extend our earlier conference paper with new techniques, and go further to demonstrate how to use the novel PF-SPH model to simulate multi-interaction scenarios. Unlike previous works in computer graphics such as [11], [12], [21], we calculate pairwise forces using a *large* neighborhood of particles in conjunction with an anisotropic filtering term, as explained in Section 3. By doing so, the PF-SPH model can accurately capture surface tension without extra forces or constraints. We go on to show this pairwise view can be readily extended to handle a wide range of interactions. Our model is capable of simulating phenomena involving complex interactions in a versatile way, with the help of the mathematical models derived from the field of computational physics. We also show that the proposed model can further be used to animate bubbles, including bubbles rising in a liquid and bubbles in air.

In the rest of the paper, Section 2 discusses related work, while we describe the details of our modified PF-SPH model in Section 3. Methods for handling various interactions in a unified way are explained in Section 4. We provide illustrative examples in Section 5. Finally, limitations and future work are discussed in Section 6.

2 Related Work

In computer graphics, previous work has considered various interactions including surface tension, adhesion, coupling and interfacial flows, mostly focusing on one or two of these interactions. Most methods rely on grid-based simulators to model interfacial flows [3], [4], [5], [6], surface tension [7], [8], and bubbles [8], [9], [22], [23], [24]. We classify the interactions at interfaces into three most commonly observed categories, i.e., those between a liquid and a gas, or another liquid, or a solid. We provide an overview of the work most closely related to this paper.

2.1 Liquid-Gas Interaction and Bubble Animation

The interaction between a liquid and air has been extensively investigated by previous researchers. Early work focused on the *continuum surface force* (CSF) method [2], [10], [25], which models surface tension based on its effects, aiming to minimize surface curvature. Unfortunately, this model relies on accurate curvature and normal estimates, which are sensitive to particle disorder. Solenthaler and Pajarola [2] adopted a smoothed normal estimate to improve the results. Others [11], [17] instead captured surface tension based on its cause and developed the *pairwise interaction force* method, which works at the molecular level to overcome the limitations of the CSF method. It uses a combination of short-range repulsion and long-range attraction forces to capture the physical nature of surface tension, avoiding erroneous estimates of curvatures and normals. However, the pairwise forces used result in relatively poor performance [12], [13], [21]. Akinci et al. [12] addressed the performance issue in some of previous methods and simulated the surface tension force by combining a cohesion term and a surface area minimization term. This work gives excellent plausible results, but requires well-designed kernel functions, which do not easily extend to handle a variety of interactions at interfaces. Furthermore, surface area minimization is the result of surface tension, not its cause. Most recently, He et al. [13] drew inspiration from the Cahn-Hilliard equation and modeled surface tension by minimizing surface energy in conjunction with air pressure. However, surface energy minimization cannot be accurately implemented using SPH and leads to unsatisfactory results if extra forces are not taken into account. Air pressure is also not the main cause of surface tension.

We also consider bubble animation which requires simulating interaction between a liquid and a gas (or air). Bubbles can be classified into two main categories according to where they are, i.e., bubbles in a liquid, and bubbles in air. The animation of bubbles in liquids using particle-based methods has attracted the attention of many researchers. Thürey et al. [26] simulated bubbles and foam within a shallow water framework. This framework simplifies the implementations due to its simplified two-dimensional representation of a height field, while at the same time limits the applications of their method. Cleary et al. [27] and Ihmsen et al. [28] investigated the complex interactions when air dissolves in the fluid. However, they used discrete spherical particles to represent bubbles or foam, so their methods cannot capture the deformation of bubbles. This issue was solved by incorporating an Eulerian grid-based framework [22]. With the help of a multiphase model, Ihmsen et al. [29] simulated complex bubble flow, including deformation and merging of bubbles. Ren et al. [31] considered bubbles in liquid using a multi-fluid simulation based on a volume fraction representation. Using particle-based frameworks to simulate bubbles in air [21] has not received much attention and is addressed in this paper as well. In this paper, bubbles in air are modeled as the gas inside, the air outside, and a thin film of a liquid, and can be simulated as droplets.

2.2 Liquid-Liquid Interaction

A multi-fluid system is composed of more than one miscible or immiscible fluids. Various particle-based multiple-fluid simulations [1], [2], [32] have assigned different densities and labels to different fluids. Yang et al. [1] proposed a reactive stress term to capture interactions between miscible phases. Solenthaler and Pajarola [2] tried to handle fluids with large density differences more precisely by deriving a modified density calculation method. Müller et al. [15] noted the significance of liquid-liquid interactions and modeled interfacial forces by combining a number of techniques including continuum surface force and diffusion effects. Macklin et al. [33] simulated immiscible fluids with a density ratio using position-based dynamics, capturing the Rayleigh-Taylor instability.

In this paper, we go further to illustrate the varying patterns of interfacial interactions as shown in Fig. 4, which has been largely overlooked by previous multi-fluid simulations.

2.3 Liquid-Solid Interaction

The interactions between a liquid and a solid are of two main kinds: fluid-solid coupling and adhesion. Solenthaler et al. [35] used a unified particle model to handle coupling in various fluid-solid interactions including deformable objects; by doing so, they managed to animate the phase change between a fluid and a solid. Ihmsen et al. [36] focused on rigid-fluid couplings and showed that adaptive timesteps are required for boundary handling in a Predictive-Corrective Incompressible SPH (PCISPH) framework. Akinci et al. [16] gave a versatile two-way fluid-solid coupling approach using SPH and per-particle correction of volumes of boundary particles. Lately, Shao et al. [37] combined PCISPH and geometric lattice shape matching to achieve two-way fluid-solid coupling with large timesteps.

Turning to adhesion, Clavet et al. [34] modeled the adhesion of fluids to solids using a distance-based attractive force. Schechter and Bridson [14] proposed the use of ghost particles for free surface and solid boundary, capturing realistic cohesion of liquids to solids. This work was further extended by He et al. [13]; they calculated the air pressure force without sampling ghost air particles, with a significant reduction in memory and computational costs. By using a well-designed kernel function, Akinci et al. [12] could capture different wetting effects.

3 Pairwise-Force SPH Model

In this section, we first introduce the original pairwise-force SPH model [17] in Section 3.1, and then discuss how to improve it for visual applications based on our earlier conference paper [21] in Section 3.2; the physcial meaning of the pairwise force is also discussed. Finally, we propose an anisotropic filtering term to further refine the improved model in Section 3.3.

3.1 Pairwise Interaction

Physically, neighboring molecules interact with each other, leading to interaction forces, which are the cause of many natural phenomena such as surface tension and adhesion. The basic SPH method simulates fluids at a macroscopic level; therefore, it inevitably ignores inter-molecule forces, lacking the capability to capture various phenomena caused by molecular forces. Since SPH is a particle-based Lagrangian simulator, we may suppose some relationship exists between SPH particles and molecules. Tartakovsky and Meakin [17] proposed the PF-SPH model following this idea. Researchers have attempted to use this model to simulate surface tension [11], [12], adhesion [12] in computer graphics. By adding a molecular-force-like pairwise particle-particle interaction term $\boldsymbol{F}^{int}$ into the conservative SPH approximation of the Navier-Stokes equation, the PF-SPH model states that

$$m_i \frac{D\boldsymbol{u}_i}{Dt} = \boldsymbol{F}_i + \boldsymbol{F}_i^{int}, \tag{1}$$

where $m_i, \boldsymbol{u}_i, \boldsymbol{F}_i^{int}$ are respectively the mass, velocity, and interaction term for particle i. D/Dt is the substantial derivative corresponding to the Eulerian expression $\partial/\partial t + \boldsymbol{u} \cdot \nabla$. The term $\boldsymbol{F}_i$ represents all other forces acting on the ith particle including pressure force, viscosity force, and body forces. $\boldsymbol{F}_i^{int}$ can be broken down as

$$\boldsymbol{F}_i^{int} = \sum_j \boldsymbol{f}_{ji}, \tag{2}$$

where $\boldsymbol{f}_{ji}$ is the pairwise interaction force with which neighboring particle j acts on particle i; note that $\boldsymbol{f}_{ji} = -\boldsymbol{f}_{ij}$. Here,

$$\boldsymbol{f}_{ij} = -c_{ij} m_i m_j f(r_{ij}) \frac{\boldsymbol{r}_{ij}}{r_{ij}}, \tag{3}$$

where $\boldsymbol{r}_{ij} = \boldsymbol{r}_i - \boldsymbol{r}_j$ and $r_{ij} = |\boldsymbol{r}_{ij}|$. $c_{ij}(\geq 0)$ is a user-tuned coefficient that controls the strength of the interaction force. $f(r_{ij})$ is a spline function that determines the behavior of the interaction force.

In fluid mechanics, high pressure regions push on lower pressure regions. Pressure plays a key role to keep the fluid incompressible. In basic SPH, the pressure force $\boldsymbol{F}^p$ can be discretized as follows [38]

$$\boldsymbol{F}_i^p = -\sum_j m_i m_j \left(\frac{p_i + p_j}{\rho_i \rho_j} \right) \frac{dW(r_{ij}, h)}{dr_{ij}} \frac{\boldsymbol{r}_{ij}}{r_{ij}}, \tag{4}$$

where $\boldsymbol{F}_i^p, p_i, \rho_i$ are the pressure force, pressure, and density of particle i respectively. W is a kernel function.

When comparing Eq. (4) with Eqs. (2) and (3), it turns out that the pressure force and the interaction term share a similar formulation. Tartakovsky and Panchenko [38] noticed this connection and called the interaction term in Eq. (1) the *virial pressure*, providing a new view of the pairwise interaction forces.

In computational physics, when each particle has very many neighboring particles in its kernel support domain, the PF-SPH model works well [17], [18], [19]. However, as previously noted, in computer graphics, when using smaller nnumber of neighboring particles, this model does not capture plausible interactions at interfaces, such as surface tension [11], [12]. Akinci et al.[12] demonstrated that it is insufficient to produce realistic surface tension effects by applying pairwise cohesion forces using only a finite support radius. Instead, they simulated surface tension by

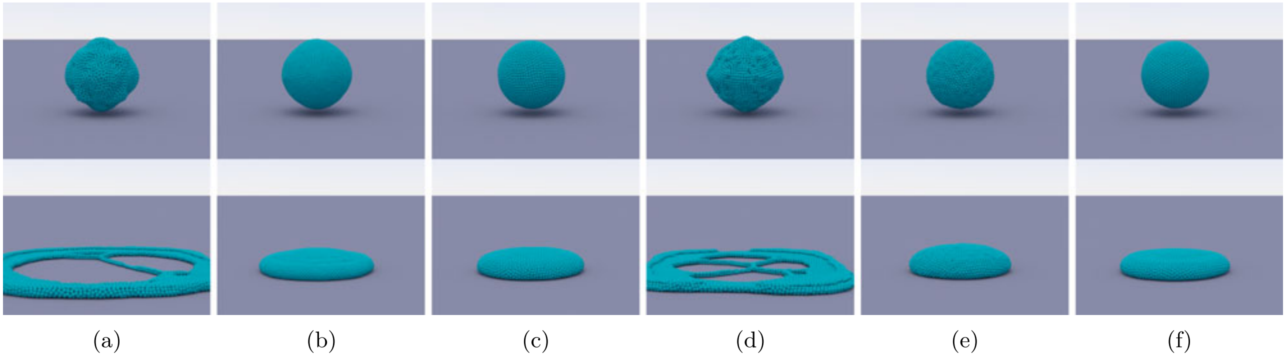

Fig. 1. Results produced by surface tension forces using different modeling approaches; external forces are ignored. Top of each subfigure: configuration reached from a cube. Bottom: Result of dropping that configuration onto a plane. Methods used and performances (#P: Number of particles, FPS: number of frames per second) are: (a) [17] using a small neighborhood; #P: 6k; FPS: 120, (b) [17] using a large neighborhood (denser particles); #P: 13k; FPS: 60, (c) our refinement of [17]; #P: 6k; FPS: 100, (d) [12] without surface area minimization; #P: 6k; FPS: 120, (e) [12] with surface area minimization; #P: 6k; FPS: 105, (f) our refinement of [12] without surface area minimization; #P: 6k; FPS: 100.

combining a pairwise cohesion force and a surface area minimization term; the former acts to alleviate particle clustering. At the microscopic scale, interaction forces are the only forces acting between molecules. Since a free surface does not have liquid molecules on both sides, and the interaction force between liquid and air molecules is much lower than that between liquid molecules, liquid molecules in such regions are pulled back into the liquid. It is this process that is responsible for minimizing the surface area. Thus, unlike Akinci et al., we consider how to produce surface tension using pairwise interaction forces without additional constraints. To do so, an improved PF-SPH model is given next.

3.2 Improved Pairwise Force

When SPH is used in computational physics, the number of neighboring particles in 3D is typically set to 80 to obtain the desired accuracy [19], while in computer graphics, the number is set to about 30 for 3D simulations [39] in an attempt to balance realism and computational cost. Our experiments have shown that doing so compromises the accuracy of the pairwise forces, which in turn leads to implausible results. This is shown in Fig. 1, which presents results produced by surface tension forces using different modeling approaches. The top of each subfigure is the final configuration reached from a cube of particles when external forces are ignored, while the bottom shows the result of dropping that configuration onto a plane. When SPH is used with a small neighborhood, it fails to completely pull the particles together into spheres; the resulting configurations generate cobweb-like series of elongated structures when dropped onto the plane (see Figs. 1a and 1d). In contrast to that, the results are plausible when SPH is used with a larger neighborhood. We obtain almost perfect spheres, without any cobweb-like structures (see Fig. 1b).

The PF-SPH model in computational physics achieves desired results. However, since much more neighboring particles are taken into account, it requires more computational cost (see Fig. 1b), which makes it less attractive for real-time simulations in computer graphics. Our observation presents that the accuracy of the pairwise force is compromised if insufficient neighboring particles are used. To overcome this problem, we attempt to increase the number of neighboring particles considered by enlarging the support radius when calculating pairwise forces. And as our implementation demonstrated, we achieve plausible results with relatively little extra computational load by doing so (see Figs. 1c and 1f).

The required enlargement ratio k for the original radius of the neighborhood h can be estimated to

$$k = (80/30)^{\frac{1}{3}} \approx 1.4. \tag{5}$$

Tartakovsky and Meakin [17] proposed a cosine function to generate pairwise interaction forces. After applying our refinement, the original spline function is modified to be

$$f(r_{ij}) = \begin{cases} \cos\left(\frac{3\pi}{2kh} r_{ij}\right), & r_{ij} \leq kh \\ 0, & r_{ij} > kh. \end{cases} \tag{6}$$

The idea of using a different support radius is not completely new. For instance, He et al. [13] adopted a two-scale pressure approximation to robustly estimate internal pressures to capture thin features. Physically, surface tension arises due to interaction between molecules, following the Lennard-Jones potential. For molecules inside the fluid, the attractions cancel one another out. For those near the surface, the asymmetrical distribution of neighbors leads to a non-zero net force towards the fluid domain. The attraction actually exists between any pair of molecules. By adopting an enlarged support radius, more neighboring particles are considered when calculating interaction forces. This brings us closer to the nature of surface tension and leads to more realistic net force differences between interior molecules and those near the free surface. Enlarging the support radius results in additional attractive forces, causing particle clustering in SPH. However, the nature of surface tension is to pull molecules together. The balance between surface tension and particle clustering is a trade-off. In our simulation, the pairwise forces we adopt do not vanish for close neighbors (see Fig. 2), which helps to prevent particle clustering. Experimentally, we have observed that the enlargement ratio k used in our examples is capable of capturing accurate surface tension while simultaneously avoiding particle clustering.

Tensile instability which leads to particle clustering has been extensively investigated in SPH [14], [16], [42]. This issue arises due to neighborhood deficiency at the free surface. Using an enlarged support radius to capture interactions means a larger number of neighboring particles is considered. However, our method cannot fix the particle

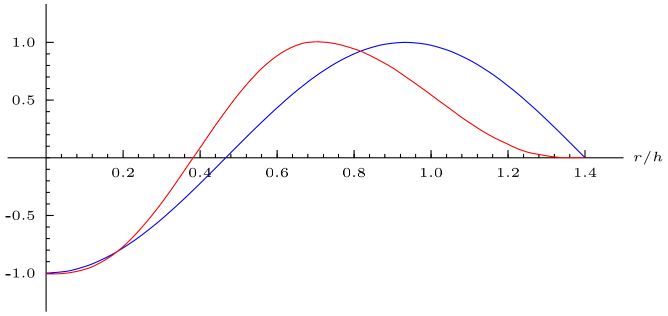

Fig. 2. Dimensionless interaction force coefficients for surface tension, applying our refinement to the models of: Tartakovsky and Meakin's (blue) and Akinci et al. (red). The coefficients are scaled to start at the same position.

deficiency issue, partly because the enlarged support radius only works for calculating pairwise interaction forces.

3.3 Anisotropic Filtering

The model discussed in Section 3.2 works quite well as illustrated in our conference paper. However, it still suffers from calculating inaccurate surface tension force (see Fig. 3). To cope with this issue, we extend to refine the improved PF-SPH model with an anisotropic filtering term inspried by the work of Yu and Turk [40], Ando et al. [30] and He et al. [13]. For the ith particle, we compute its anisotropic covariance C_i as follows:

$$C_i = \frac{\sum_j (\boldsymbol{x}_j - \boldsymbol{x}_i)(\boldsymbol{x}_j - \boldsymbol{x}_i)^T W(r_{ij}, h)}{\sum_j W(r_{ij}, h)}, \tag{7}$$

where $\boldsymbol{x}_i$ is the position of the ith particle. $(\boldsymbol{x}_j - \boldsymbol{x}_i)^T$ represents the transpose of vector $\boldsymbol{x}_j - \boldsymbol{x}_i$.

The Singular Value Decomposition (SVD) of the associated C_i yields the directions of stretch or compression for deforming kernel W, and gives the eigenvectors and eigenvalues

$$C_i = \boldsymbol{R}\boldsymbol{\Sigma}\boldsymbol{R}^T, \boldsymbol{\Sigma} = diag(\epsilon_1, \epsilon_2, \epsilon_3), \tag{8}$$

where R denotes the eigenvector matrices. $\epsilon_1 \geq \epsilon_2 \geq \epsilon_3$ are the eigenvalues. To avoid singular matrices, we set $\epsilon_\kappa = \max(\epsilon_\kappa, \epsilon_{min})$, where $\kappa \in \{1, 2, 3\}$, and ϵ_{min} is user-defined minimum value of the eigenvalue. We use $\epsilon_{min} = 0.5$ in our implementations. A modified covariance matrix $\widetilde{C}_i$ is expressed as

$$\widetilde{C}_i = \boldsymbol{R}\widetilde{\boldsymbol{\Sigma}}\boldsymbol{R}^T, \tag{9}$$

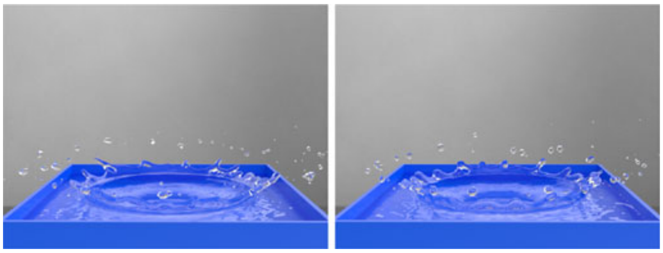

Fig. 3. Comparison of a selected frame from the water crown experiment using our earlier conference paper (left) and current model (right). The splashes look sharp and edgy using the method proposed by our earlier conference paper; while spherical droplets are captured using our current model.

where $\widetilde{\boldsymbol{\Sigma}}$ is the diagonal matrix with modified eigenvalues. In our experiments, $\boldsymbol{G}_i$ wich is the inversion of $\widetilde{C}_i$ is needed, and it is given by

$$\boldsymbol{G}_i = \boldsymbol{R}\widetilde{\boldsymbol{\Sigma}}^{-1}\boldsymbol{R}^T. \tag{10}$$

We then propose an anisotropic filtering term $\boldsymbol{T}_i$ as

$$\boldsymbol{T}_i = (1-\eta)\boldsymbol{I} + \eta\frac{\boldsymbol{G}_i}{||\boldsymbol{G}_i||}. \tag{11}$$

where $\boldsymbol{I}$ is the identity matrix. η is a user-defined coefficient to determine the degree of anisotropy. In our implementations, we vary η according to k: to achieve comparable results, we compensate a small reduction in k by using a slightly larger η. We use $\eta = 0.2$ in our experiments.

With the help of the anisotropic filtering term, the pairwise interaction force $\boldsymbol{F}_i^{int}$ in Eqs. (1) and (2) for particle i can be redefined as

$$\boldsymbol{F}_i^{int} := \boldsymbol{T}_i\boldsymbol{F}_i^{int}. \tag{12}$$

The anisotropic filtering term is used to scale the interaction forces experienced by a given particle from its neighbors. If the neighborhood is symmetric, the anisotropic filtering term is an identity matrix, scaling the interaction forces of all particles within the support domain uniformly. If the neighborhood is asymmetric, the anisotropic filtering term scales the interaction forces of different particles within the domain differently. See Fig. 5.

In our experiments, Eq. (12) further improves the performance of our PF-SPH model (see Fig. 3). Since we adopt a large support radius when calculating interaction forces, Eq. (7) should sum over all the particles within the large neighboring domain.

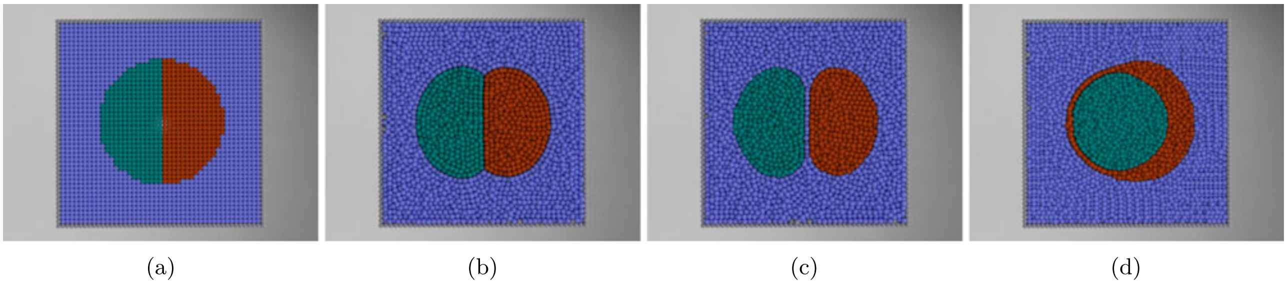
(a) (b) (c) (d)

Fig. 4. PF-SPH model for three fluids: 1 (blue), 2 (red), and 3 (green) shown on a cross-sectional plane. (a) initial distribution; (b) the three fluids stay in contact with each other in equilibrium using interaction coefficients: $c_{11} = 6 \times 10^3, c_{22} = 1 \times 10^4, c_{33} = 1.2 \times 10^4, c_{12} = c_{23} = 0, c_{13} = 2 \times 10^3$, (c) fluid 2 separates from fluid 3 using interaction coefficients: $c_{11} = 1 \times 10^4, c_{22} = 8 \times 10^3, c_{33} = 1.4 \times 10^4, c_{12} = c_{23} = 6 \times 10^3, c_{13} = 0$, (d) fluid 3 becomes separated from fluid 3 by fluid 2, using interaction coefficients: $c_{11} = 6 \times 10^3, c_{22} = 1.4 \times 10^4, c_{33} = 1.4 \times 10^4, c_{12} = c_{13} = 6 \times 10^3, c_{23} = 0$.

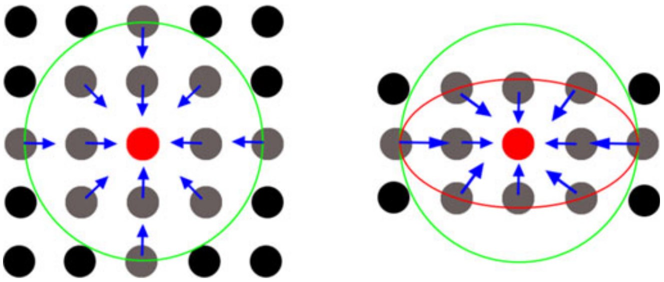

Fig. 5. 2D illustration of how anisotropic filtering works. Left: Isotropic case. Right: Anisotropic case. Green boundaries show the searching domains of the red particles; gray particles are the neighbors. In the anisotropic case, the red oval is plotted by scaling the neighborhood with the covariance matrix, and is used for ease of understanding the anisotropy. Lengths of blue arrows indicate scaling ratios applied to the interaction forces by the anisotropic filtering term. In the isotropic case, the anisotropic filtering term is an identity matrix, leading to uniform interaction forces for all particles within the neighborhood. in the anisotropic case, anisotropic filtering scales interaction forces of different particles according to the anisotropy.

We test our novel PF-SPH model to simulate surface tension using the cosine function. The results are now an almost perfect sphere when starting from a cube and no cobweb-like structures are observed when the configuration drops onto a plane as shown in Fig. 1c.

To further examine the utility of enlarging the neighborhood in conjunction with the anisotropic filtering, we have tested it using the pairwise force model proposed by Akinci et al. [12]. Again, we obtain an almost perfect sphere, without any cobweb-like structures (see Fig. 1f). The results obtained in this way are as plausible as those acheived by Akinci et al. with the help of an extra surface area minimization constraint (see Fig. 1e).

These experiments show that our novel PF-SPH model is effective and results in more plausible surface tension. The non-spherical configuration and the cobweb-like structures are both caused by inaccurate surface tension forces. As demonstrated in Figs. 1a and 1d, when SPH is used with a small neighborhood, the original pairwise forces proposed by Tartakovsky and Meakin [17] and Akinci et al. [12] both result in such unrealistic structures. With our refinement, such structures disappear.

4 Multi-Interaction Applications

We introduced our refined PF-SPH model in Section 3, and showed how to apply it to capture the interaction between a liquid and air, i.e., surface tension. Since this approach is based on particle interactions without resorting to artificial considerations including surface area minimization or air pressure, it has the further advantage of allowing us to extend the pairwise view to handle other types of interactions, including fluid-solid and fluid-fluid interactions. In this section, we demonstrate how to use the PF-SPH model to handle phenomena involving multiple interactions in a unified manner. This is one of the biggest advantages of using pairwise interaction forces; it has so far been overlooked in computer graphics. Compared to our conference paper, we go further to show how to handle multiple interfacial flows, and how to capture the intricate deformation details of rising bubbles in liquid. An empirical model is also introduced to simulate various wetting effects when the gravity force is ignored.

When simulating multi-interaction scenarios, a number of interaction coefficients must be determined. Many parameters must be set to achieve the desired results, a problem which we now turn to. Later, we also show how to animate bubbles either in air or in liquids.

4.1 Interaction Coefficients

In fluid simulations, commonly observed interactions at interfaces can be categorized into three kinds: those between a fluid and air, those between immiscible fluids, and those between a fluid and a solid. We thus discuss how to set interaction coefficients to capture these phenomena respectively.

The parameter c_{ij} used in Eq. (3) determines the strength of the interaction force between particle i and particle j. This value varies depending on the materials involved, and may be written as $c_{\alpha\beta}$, where α, β respectively represent the particular phase or component that particles i, j respectively belong to. If α, β belong to the same phase or component, $c_{\alpha\beta}$ is the cohesion coefficient; otherwise, it is the interaction coefficient at the interface.

4.1.1 Surface Tension

To model surface tension, air and fluid both matter. Physically, air is also composed of molecules, so three different interaction forces between molecules must be considered when modeling surface tension: air-air, air-fluid, and fluid-fluid. Since the interaction between fluid and air is insignificant, we may idealize the coefficient as 0. While surface tension is actually a two-phase phenomenon, it can thus be modeled in terms of a single fluid phase. When simulating surface tension, we thus only need to set the cohesion coefficient for the fluid.

4.1.2 Interfacial Flows

When multiple immiscible fluids interact in a single scenario, the results vary according to the parameter values used in the PF-SPH model. In this section, we take a three-fluid system as an example for simplicity, but the general approach here can be applied to fluid systems with a larger number of fluids.

For a three-fluid system (fluids 1, 2, 3), a surface tension parameter: $\sigma_{12}, \sigma_{13}, \sigma_{23}$ is specified for each pair of fluids. Tartakovsky and Panchenko [20] showed that the relationship between the parameters in the pairwise forces and the surface tension parameters is

$$\sigma_{\alpha\beta} = \lambda(\bar{c}_{\alpha\alpha} + \bar{c}_{\beta\beta} - 2\bar{c}_{\alpha\beta}), \tag{13}$$

where λ is a constant coefficient. α, β are indexes of fluid phases. $\bar{c}_{\alpha\beta}$ is given by

$$\bar{c}_{\alpha\beta} = n_\alpha n_\beta c_{\alpha\beta}, \tag{14}$$

where n_α is the number density [2] of phase α. Note that $c_{\alpha\beta} = c_{\beta\alpha}$

In the PF-SPH model for this three-fluid system, six parameters should be specified, i.e., $c_{11}, c_{22}, c_{33}, c_{12}, c_{13}$, and c_{23}. A commonly used assumption is that

$$c_{\alpha\beta} \ll c_{\alpha\alpha}, \quad \alpha \neq \beta, \quad \alpha, \beta \in \{1, 2, 3\}. \tag{15}$$

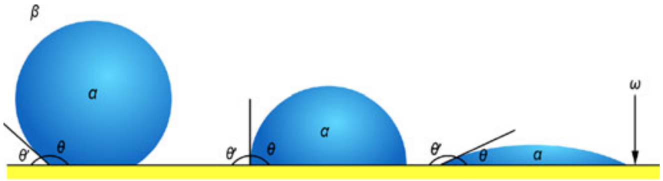

Fig. 6. Different wetting effects and contact angles.

The above assumption may not work in every case; we only consider it a start point of simplfying coefficient tunning.

Substituting Eq. (15) into Eq. (13) gives

$$\sigma_{\alpha\beta} \approx \lambda n_{eq}^2 (c_{\alpha\alpha} + c_{\beta\beta}), \tag{16}$$

where we assume $n_{eq} := n_\alpha = n_\beta$.

If the three fluids stay in contact with each other at equilibrium, the surface tension parameters should satisfy the following conditions:

$$\sigma_{\alpha\beta} < \sigma_{\alpha\gamma} + \sigma_{\beta\gamma}, \quad \alpha, \beta, \gamma \in \{1, 2, 3\}, \tag{17}$$

where $\alpha \neq \beta \neq \gamma$, and $\sigma_{\alpha\beta} = \sigma_{\beta\alpha}$.

It can be seen that, for any choice of $c_{\alpha\alpha}$, $\alpha \in \{1, 2, 3\}$, the surface tension parameters defined according to Eq. (16) satisfy these conditions, i.e., the three fluids stay in contact with each other at equilibrium as illustrated in Fig. 4b. For simulations which should prevent all three fluids from staying in contact with each other, the conditions in Eq. (17) must be violated; thus the three-fluid contact line (point in 2D simulation) cannot be formed. In Fig. 4c, we set $c_{12} = c_{13} = O(c_{\alpha\alpha})$ and $c_{23} \ll c_{\alpha\alpha}$. Fluids 2 and 3 draw apart from each other, and in Fig. 4d, we set $c_{12} = c_{23} = O(c_{\alpha\alpha})$ and $c_{13} \ll c_{\alpha\alpha}$. Fluid 2 encloses fluid 3, separating the latter from fluid 1.

4.1.3 Adhesion

Bandara et al. [18] focused on the adhesion of fluids to solids using the PF-SPH model. Considering classical theories of surface tension, they determined the relationship between contact angle and various coefficients to be

$$\cos\theta = \frac{\bar{c}_{\alpha\alpha} - \bar{c}_{\beta\beta} + 2\bar{c}_{\omega\alpha} - 2\bar{c}_{\omega\beta}}{\bar{c}_{\alpha\alpha} + \bar{c}_{\beta\beta} - 2\bar{c}_{\alpha\beta}}, \tag{18}$$

where α, β are fluid phases and ω is the solid phase. Here θ is the static contact angle between fluid α and the solid phase ω in the presence of fluid β, as illustrated in Fig. 6; note the order of α and β matters.

A typical adhesion involves interactions between a fluid, air, and a solid. Since air particles are usually ignored in SPH, there are more coefficients in Eq. (18) than needed. For instance, when simulating the wetting effects of a fluid α on a solid ω, we only need to know the coefficients of $c_{\alpha\alpha}$ and $c_{\omega\alpha}$. However, Eq. (18) requires coefficients associated with the air phase β, which makes Eq. (18) impractical for artistic control.

To make Eq. (18) more practical as a way to estimate the coefficients that actually control the simulation, some approximations are needed. Suppose α, β, and ω represent a fluid, air, and a solid, respectively. Physical observations make it clear that the interactions between air and a fluid or a solid, i.e., $c_{\alpha\beta}$, $c_{\omega\alpha}$, are significantly smaller than other interactions, so we may set $c_{\alpha\beta} = c_{\omega\beta} = 0$. While we can ignore interactions between air and liquid, interactions between particles in the same phase cannot be ignored, including those between air particles. The ignored air particles can in some ways be treated as ghost particles [14], so it is plausible to assume $n_\alpha = n_\beta$ and $c_{\alpha\alpha} = c_{\beta\beta}$. This simplifies Eq. (18) to

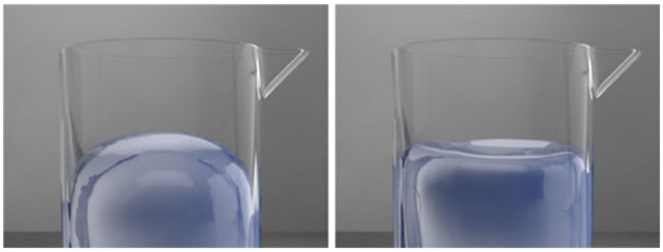

Fig. 7. Liquids form menisci in a graduated cylinder. Left: Convex surface due to the obtuse contact angle of mercury with glass. Right: Concave surface due to the acute contact angle of water with glass.

$$\cos\theta \approx \bar{c}_{\omega\alpha}/\bar{c}_{\alpha\alpha}. \tag{19}$$

To verify the validity of Eq. (19), we choose α, β in the opposite way, i.e., letting α represent air and β a fluid. Then following the idea above, Eq. (18) gives

$$\cos\theta' \approx -\bar{c}_{\omega\beta}/\bar{c}_{\beta\beta}, \tag{20}$$

where θ' represents the contact angle of air α with solid ω in the presence of a fluid β. Since $\theta, \theta' \in [0, \pi]$, combining Eqs. (19) and (20), we see that $\theta + \theta' = \pi$, as illustrated in Fig. 6.

However, in our model, all interaction coefficients are non-negative. Thus the simplified form of Eq. (18) given in Eqs. (19) and (20) cannot capture wetting effects with obtuse or acute contact angles: if $c_{\omega\alpha}$ is set to 0, the contact angle given by Eq. (19) is 90 degree. In our earlier conference paper, the interaction coefficients can be negative and the gravity force is considered; Eq. (19) works. But now, it is problematic. To address this issue, we note the gravity force can be ignored in small scenarios where the surface tension force domainates [12], such as a droplet on a plane. In this case, if $c_{\omega\alpha} = 0$, the actual contact angle should be 180 degree. Therefore, based on Eq. (19), we can improve upon this by setting

$$\cos\theta \approx 2\bar{c}_{\omega\alpha}/\bar{c}_{\alpha\alpha} - 1. \tag{21}$$

While Eq. (21) is not entirely physically meaningful, it works well in our experiments. We take an example of simulating strawberry sauce on a ball as considered later in Fig. 14 to demonstrate how to set coefficients using Eq. (21). Since the contact angle of strawberry sauce on the ball in air is close to 0, we set $\bar{c}_{\omega\alpha} = \bar{c}_{\alpha\alpha}$ according to Eq. (21) where α, β, ω represent strawberry sauce, air, and the ball, respectively. Furthermore, if the strawberry sauce and the ball are composed of particles with the same number densities, then $c_{\omega\alpha} = c_{\alpha\alpha}$. The values of the coefficients depend on the properties of the materials as well as the simulators used.

Adhesion is the main cause for many real-world visual phenomena, such as different wetting effects shown in Fig. 15, and the convex or concave meniscus at a liquid surface in a container as shown in Fig. 7. Our method can achieve comparable results using alternative models for contact angles. With the help of Eqs. (18) and (21), our PF-SPH model provides an easy-to-handle way to capture those challenging scenarios.

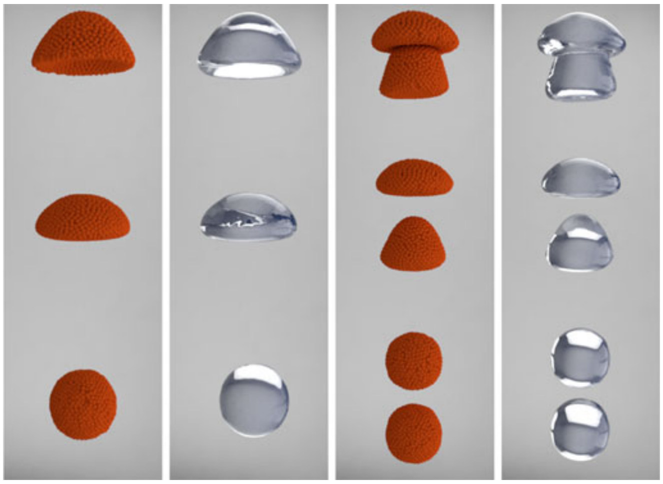

Fig. 8. Bubbles rising in liquids. Bottom to top: Three stages of a bubble rising. Left, center left: Particle view and rendered view of a single bubble rising; note how the bubble deforms as it rises. Center right, right: Particle view and rendered view of two bubbles coalescing as they rise.

4.1.4 Discussion

We have demonstrated how to use parameters in the PF-SPH model to achieve desired results for different types of interactions including surface tension, interfacial flows, and adhesion. When considering interactions at multiple interfaces, it is quite straightforward to sequentially determine appropriate coefficients for all interfaces. For instance, in the example of cracking an egg shown in Fig. 12, we need to specify five interaction coefficients: $c_{\alpha\alpha}, c_{\beta\beta}, c_{\omega\alpha}, c_{\omega\beta}$ and $c_{\alpha\beta}$, where α, β, and ω represent egg white, egg yolk, and the plane respectively (The interactions between the shell and egg white or egg yolk are considered as fluid-solid couplings). We first determine $c_{\alpha\alpha}, c_{\omega\alpha}$ and $c_{\beta\beta}, c_{\omega\beta}$ to achieve the desired wetting effects of egg white and egg yolk respectively using Eq. (21). We then select coefficients for the interactions between egg white, egg yolk, and air as discussed in Section 4.1.2.

One of the advantages of using pairwise interaction forces is that a uniform approach can be used without needing to design new kernel functions for different interfaces, e.g. as necessary in the approach of Akinci et al. [12]. By setting appropriate values for various coefficients according to the interaction being modeled, our approach is capable of simulating complex phenomena with multiple interactions at different interfaces.

4.2 Bubble Animation

Bubble simulation is an interesting and challenging topic. The shapes of bubbles are dominated by surface tension as demonstrated by many particle- and grid-based methods [8], [22], [24]. Droplets are also dominated by surface tension forces when external forces are absent. Both bubbles and droplets are volume-conserving at a constant temperature. Based on these considerations, we observe that it is possible to animate bubbles either in liquids (e.g., in champagne) or in air (e.g., soap bubbles) by treating them as droplets. A bubble in liquid is full of gas while a droplet is full of liquid; thus simply considering the particles composed of a droplet to be the air particles is sufficient to capture the bubbles in liquid as demonstrated in Section 4.2.1. Turning to bubbles in air, it would be more challenging due to a thin film of liquid on surface. We address this problem by adopting a two-phase model as discussed in Section 4.2.2.

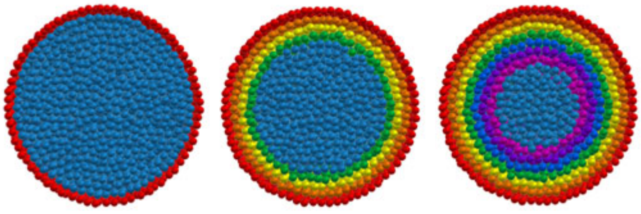

Fig. 9. Results of surface particle detection. Surface particles can be accurately detected layer by layer (from left to right).

4.2.1 Bubbles in Liquid

SPH approaches have been used to animate the movements of bubbles in liquids [15], [22], [27], [29]; little has been done in computer graphics to capture the intricate details of rising bubbles using SPH solvers. Using the close connection between bubbles and droplets, we solve this problem for the first time in computer graphics using our PF-SPH model.

One of the problematic issues in doing so lies in the high density ratio between the liquid phase and the gas phase. Using the definition of the particle mass, the density ρ of particle i in standard SPH is given by

$$\rho_i = \sum_j m_j W(r_{ij}, h). \tag{22}$$

This equation sums over the masses of neighboring particles. The main disadvantage of this formation is the difficulty of representing sharp density discontinuities at interfaces. To overcome this problem, we use the revised formation proposed by Solenthaler and Pajarola [2]

$$\rho_i = m_i \sum_j W(r_{ij}, h). \tag{23}$$

In this way, the density of a particle is determined only by the spatial distribution of its neighboring particles, but not their masses. Therefore, particles located near an interface but belonging to different fluids may interact without having their density affected by the other phase.

Interpenetration between particles is another problem that occurs when there is a high density ratio between the two phases. To prevent this interpenetration and to maintain the sharpness of interface, an additional term which introduces an extra repulsive force between two phases is used in the work of Hu and Adams [44]. Our model however does not need this term, as the pairwise force we use is short-range repulsive.

For typical applications, this density ratio is close to 10^3. A high density ratio leads to numerical issues, so very small timesteps are needed, making it impossible for real-time simulations. Fortunately, Hua and Lou [43] pointed out that density ratio affects the velocity of a rising bubble more than its terminal shape; the latter mainly depends on the surface tension. Since we focus on the deformation during bubble rising, we set the density ratio to be 10 in our experiments to ensure real-time performance. We demonstrate the capability of our model to capture rising bubbles in Figs. 8 and 17.

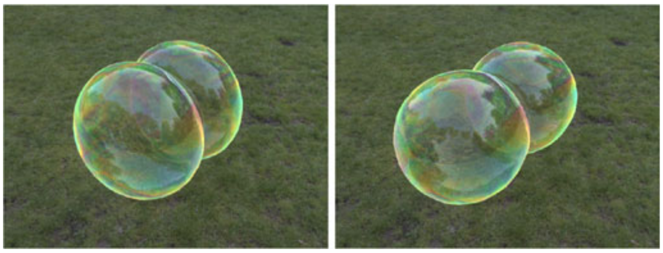

Fig. 10. A double-bubble. Two air bubbles collide with each other, presenting wobbling behaviors.

4.2.2 Bubbles in Air

Unlike bubbles in liquids, little has been done to simulate bubbles in air using the SPH method. The main difficulty lies in the thin structure of such bubbles. He et al. [13] considered how to robustly simulate thin features using SPH, but their method is incapable of handling bubbles in air. We solve this problem by taking a different approach.

A straightforward way to simulate air bubbles is to simply treat them as droplets. However, we find that this approach makes the resulting bubbles look too rigid due to the different materials contained inside a droplet (liquid) and a bubble (gas). In other words, a liquid droplet contains only a single liquid phase while a bubble in air has two different phases: a liquid phase on the surface and a gas phase inside. These two phases function differently. The gas phase dominates shape changes and conserves the volume of the bubble, while the liquid phase is responsible for various interactions. To simulate air bubbles less rigidly, we instead adopt a two-phase model using the volume fraction [15] to represent the spatial distribution of each phase. Particles in the outermost layer of the droplet are composed of two phases, corresponding to a thin film in a liquid phase and an inner gas phase, while in the interior of the bubble, the liquid volume fraction is considered to be 0. To simplify the implementation, we assume that the thin film of liquid is even over the whole bubble surface, so the volume fraction of every surface particle is the same. As the surface area changes along with the bubble shape, the volume fraction changes accordingly to conserve both liquid and interior gas volumes. A larger surface area leads to a thinner film and more surface particles, requiring a smaller volume fraction for the liquid phase in each surface particle. To determine the interaction forces between particles composed of two phases, the following interaction coefficient is used:

$$\tilde{c}_{ij} = abc_{\alpha\alpha} + (a(1-b) + (1-a)b)c_{\alpha\beta} + (1-a)(1-b)c_{\beta\beta}, \tag{24}$$

where particles i and j each contain both phases α and β, and a, and b are the volume fractions of phase α in particles i and j respectively (see Fig. 11).

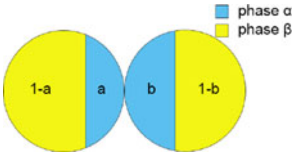

Fig. 11. Interaction between two-phase particles.

To dynamically obtain the correct volume fraction, the surface particles are detected at each step using the method of Barecasco et al. [21], [45] (see Fig. 9). When simulating bubbles as droplets, the surface particles are composed of a liquid phase and an inner-gas phase, while the others are considered as inner-gas particles. The liquid phase fraction of a surface particle actually varies from particle to particle, and from time to time; in our implementations, we set it to the same value for each surface particle for simplicity.

5 Results

We now give various further illustrations to demonstrate the range of simulations which our method can handle. See the supplementary material, which can be found on the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TVCG.2017.2706289, for associated videos. We have implemented the proposed method using the PCISPH framework, allowing the use of large time steps. We take the interaction forces to be one kind of internal force, and add them to the pressure correction iterations to ensure stable simulation: our experiments show that our model remains stable for long simulation times.

Cracking an Egg. Fig. 12 illustrates that our method is capable of simulating complex phenomena with multiple types of interface interactions. A number of coefficients associated with interfaces between egg white, egg yolk, air and solid were sequentially set using Eqs. (18) and (21). The egg white and egg yolk present different wetting effects. Letting α, β and ω represent egg white, egg yolk, and solid respectively, we used: $c_{\alpha\alpha} = 8.0 \times 10^3, c_{\beta\beta} = 1.6 \times 10^4, c_{\omega\alpha} = 6.0 \times 10^3, c_{\omega\beta} = 2.0 \times 10^3$ and $c_{\alpha\beta} = 1.0 \times 10^3$.

Water Crown. Our surface tension model can be used to capture the details of water splashing, generating a crown and the 'i' pattern typically captured by photography. See Fig. 13. Our refined method achieves similar results to those of Akinci et al.[12] without the need for an extra surface area minimization term. We set the interaction coefficient to 1.4×10^4.

Strawberry Sauce on a Ball. A stream of strawberry sauce flows over a ball. See Fig. 14. This example shows that our approach can model the adhesion force between a fluid and a solid by setting appropriate coefficients for solid and fluid phases without requiring extra 'ghost' air particles or

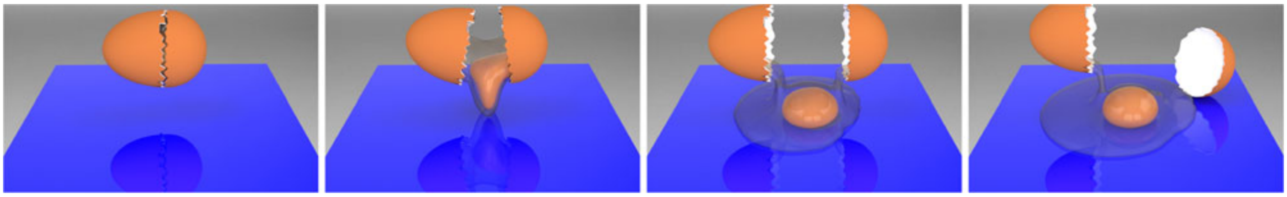

Fig. 12. A simulated egg white and egg yolk flow out of the broken shell of a cracked egg, onto a plane, using our versatile approach to handling interactions at interfaces.

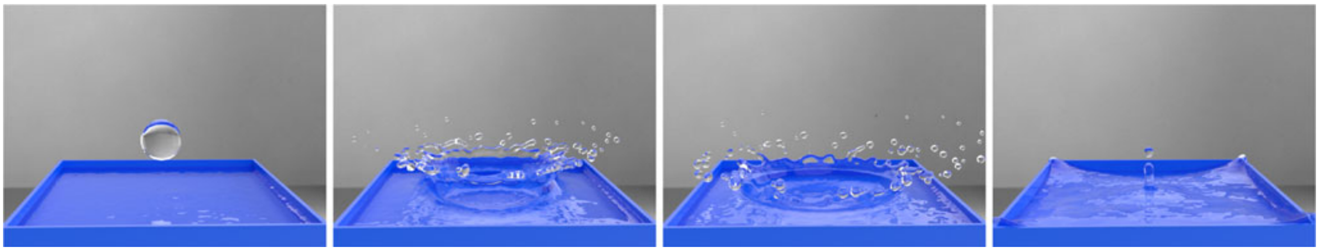

Fig. 13. Water crown. A droplet drops into a pool of liquid. Our model produces a plausible water crown and an 'i' pattern. Water in the pool is not completely still or smooth, leading to asymmetric splashes.

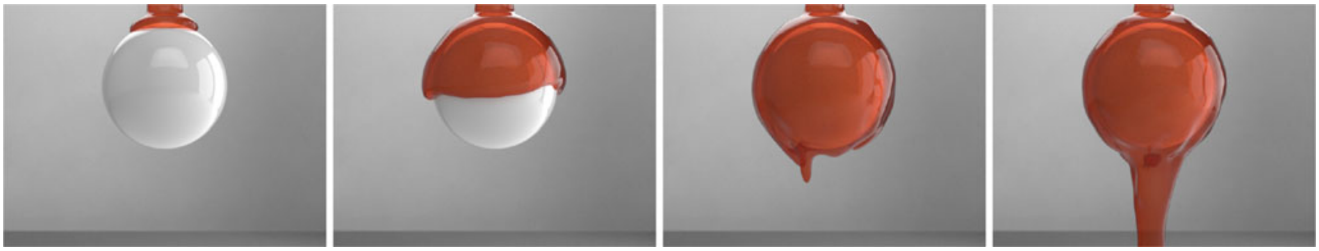

Fig. 14. Strawberry sauce on a ball. A stream of strawberry sauce flows over a ball, sticking to it by the adhesion force between fluid and solid. The stream pouring downward is not an entirely symmetric column, resulting in slightly asymmetric simulation.

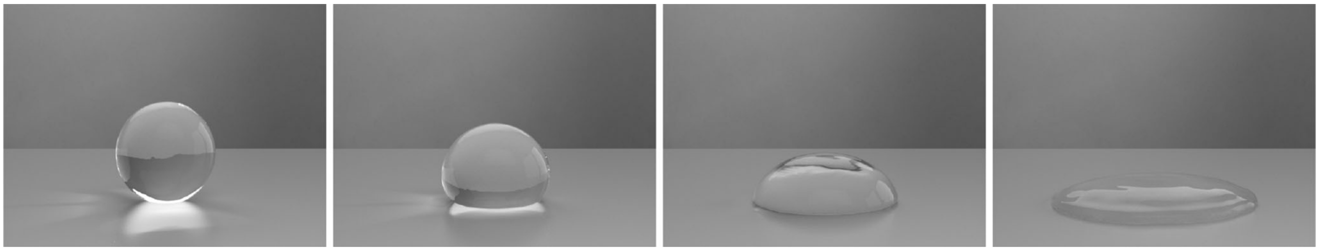

Fig. 15. Wetting effects. Our method is capable of simulating different wetting effects in a simple way.

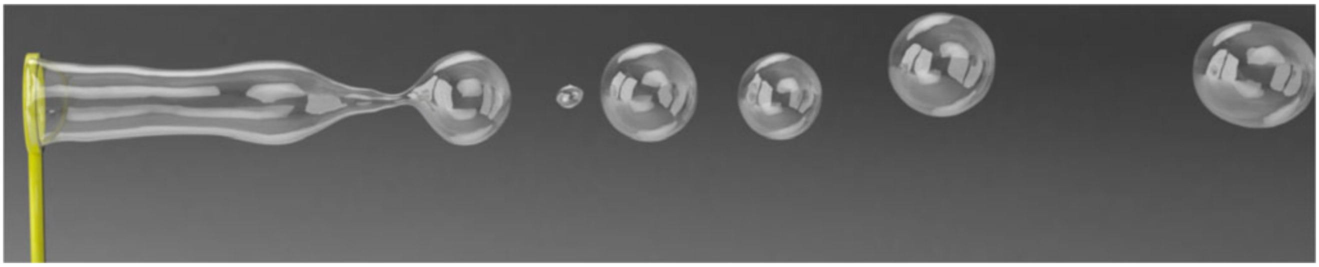

Fig. 16. Blowing bubbles. Our method is capable of animating bubbles in the air as droplets. There is a close connection between blowing bubbles and water jets. Using this observation, we provide the first particle-based approach to simulate blowing bubbles.

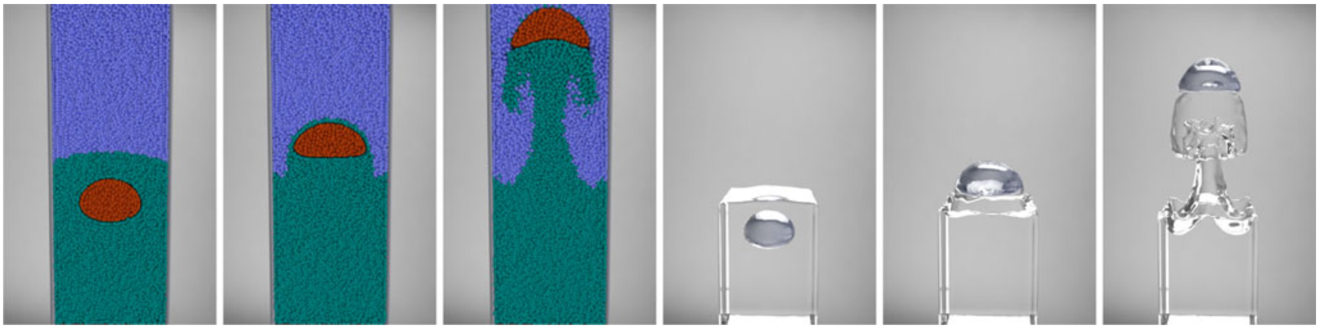

Fig. 17. Particle views, and rendered views, of a single bubble rising through a fluid-fluid interface. The rising bubble changes the interface shape by pulling the bottom fluid upward, and results in a mushroom-shaped column of the lower fluid being dragged upwards by the bubble.

TABLE 1
Performance. #P: Number of Particles. #S: Number of Steps per Frame. T: Integration Time to Compute Each Frame. FPS: Number of frames per second.

Example	#P	#S	T	FPS
Egg cracking	180 k	3	34 ms	29
Water splashing	232 k	2	55 ms	18
Strawberry sauce	10 k-40 k	1	16 ms	63
Wetting effects	9 k	2	13 ms	77
Double bubbles	6 k	2	12 ms	83
Blowing bubbles	0 k-12 k	2	36 ms	28
Rising Bubble	295 k	4	46 ms	22

artificial air pressure forces. Here, $c_{\omega\alpha}$ and $c_{\alpha\alpha}$ are both set to 1.2×10^4, where α and ω represent the fluid and solid phases respectively.

Wetting Effects. Our PF-SPH model provides a simple way to capture wetting effects with various contact angles. In this example, the cohesion cofficient is set to 1.4×10^4; to obtain different wetting effects, we may vary the interaction coefficient between the fluid and the solid according to Eqs. (18) and (21). See Fig. 15.

Double Bubbles. Fig. 10 shows our method for simulating bubbles in air as droplets. Two bubbles collide with each other in air, presenting wobbling behaviors.

Blowing Bubbles. There is a strong connection between commonly observed water jets and blowing bubbles; our approach is the first particle-based approach in graphics to be able to simulate blowing bubbles. See Fig. 16.

Rising Bubbles. Fig. 17 shows how our model can be used to capture the intricate details of a single bubble rising in liquid. The rising bubble breaks the interface between two fluids, and then starts to change the shape of the interface by pulling the bottom fluid upward. A mushroom-shaped column of the lower fluid can be seen below the bubble. Fig. 8 further demonstrates how our model is capable of simulating the coalescence of two rising bubbles.

We have implemented our algorithm using CUDA on an NVIDIA GeForce GTX1080 GPU using single precision, as it is adequate for graphical simulations and reduces memory requirements. Performance for these examples is given in Table 1. The particle numbers shown include both fluid and boundary particles. Times presented are averages. We achieve real-time performance in all cases.

6 Conclusion

We have shown how a refined pairwise force SPH model can handle multiple interactions in a versatile way. Our method succeeds by adopting a larger support radius than previous works in conjunction with the anisotropic filtering term. It results in more accurate simulations of various interactions including surface tension, interfacial flows, and adhesion, without requiring additional constraints, such as curvature minimization, surface area minimization, energy minimization, or air pressure control. For the first time in computer graphics, it is possible to simulate bubbles in air as droplets using SPH, providing an alternative method of bubble simulation. Our approach can uniformly handle multiple types of interactions, and is versatile, physically meaningful, and easy to implement. It also preserves particle clustering. Previous surface tension methods suffer from stiffer timestep restrictions; we achieve more stable simulations by considering more neighboring particles.

We use a fixed number of neighbouring particles, as this leads to simplicity of implementation, convenience in neighborhood search, and consistency in numerical evaluation. It would be interesting to try varying this value to provide error control.

Although our approach is capable of capturing adhesion forces and various pinning effects (see the supplementary video, available online), it cannot handle abrupt, turbulent fluid-solid couplings with large time steps; additional boundary pressure terms are needed in those cases.

While we have proposed a two-phase model to simulate bubbles in air, the bubbles still behave somewhat more rigidly than bubbles simulated by mesh-based methods like those in [8], [24]. Furthermore, our particle-based method is incapable of capturing certain interesting phenomena, such as thin film structures spanning a circular wire. Resolving these issues is another direction for future research.

Finally, we note that while increasing the support radius works well in simulating surface tension and adhesion, it cannot fix the particle deficiency issue which causes tensile instability in SPH. Some form of normalization is required.

Acknowledgments

This work was supported in part by the Natural Science Foundation of China (Project Number 61521002), Research Grant of Beijing Higher Institution Engineering Research Center, EU FP7 project Dr. Inventor (FP7-ICT-611383) and FP7 Marie Curie IRSES project AniNex (612627), the U.S. National Science Foundation, and an EPSRC, UK travel grant.

References

[1] T. Yang, J. Chang, B. Ren, M. C. Lin, J. J. Zhang, and S.-M. Hu, "Fast multiple-fluid simulation using Helmholtz free energy," *ACM Trans. Graph.*, vol. 34, no. 6, pp. 201:1–201:11, 2015.

[2] B. Solenthaler and R. Pajarola, "Density contrast SPH interfaces," in *Proc. ACM SIGGRAPH/Eurographics Symp. Comput. Animation*, 2008, pp. 211–218.

[3] F. Losasso, T. Shinar, A. Selle, and R. Fedkiw, "Multiple interacting liquids," *ACM Trans. Graph.*, vol. 25, no. 3, pp. 812–819, Jul. 2006.

[4] B. Kim, "Multi-phase fluid simulations using regional level sets," *ACM Trans. Graph.*, vol. 29, no. 6, pp. 175:1–175:8, Dec. 2010.

[5] M. K. Misztal, et al., "Multiphase flow of immiscible fluids on unstructured moving meshes," in *Proc. ACM SIGGRAPH/Eurographics Symp. Comput. Animation*, 2012, pp. 97–106.

[6] F. Da, C. Batty, and E. Grinspun, "Multimaterial mesh-based surface tracking," *ACM Trans. Graph.*, vol. 33, no. 4, pp. 112:1–112:11, 2014.

[7] N. Thuerey, C. Wojtan, M. Gross, and G. Turk, "A multiscale approach to mesh-based surface tension flows," *ACM Trans. Graph.*, vol. 29, no. 4, pp. 48:1–48:10, 2010.

[8] F. Da, C. Batty, C. Wojtan, and E. Grinspun, "Double bubbles sans toil and trouble: Discrete circulation-preserving vortex sheets for soap films and foams," *ACM Trans. Graph.*, vol. 34, no. 4, pp. 149:1–149:9, 2015.

[9] B. Kim, Y. Liu, I. Llamas, X. Jiao, and J. Rossignac, "Simulation of bubbles in foam with the volume control method," *ACM Trans. Graph.*, vol. 26, no. 3, Jul. 2007, Art. no. 98.

[10] M. Müller, D. Charypar, and M. Gross, "Particle-based fluid simulation for interactive application," in *Proc. ACM SIGGRAPH/Eurographics Symp. Comput. Animation*, 2003, pp. 154–159.

[11] M. Becker and M. Teschner, "Weakly compressible SPH for free surface flows," in *Proc. ACM SIGGRAPH/Eurographics Symp. Comput. Animation*, 2007, pp. 209–217.

[12] N. Akinci, G. Akinci, and M. Teschner, "Versatile surface tension and adhesion for SPH fluids," *ACM Trans. Graph.*, vol. 32, no. 6, pp. 182:1–182:8, 2013.

[13] X. He, H. Wang, F. Zhang, H. Wang, G. Wang, and K. Zhou, "Robust simulation of small-scale thin features in SPH-based free surface flows," *ACM Trans. Graph.*, vol. 34, no. 1, pp. 7:1–7:9, 2014.

[14] H. Schechter and R. Bridson, "Ghost SPH for animating water," *ACM Trans. Graph.*, vol. 31, no. 4, pp. 61:1–61:8, 2012.

[15] M. Müller, B. Solenthaler, R. Keiser, and M. Gross, "Particle-based fluid-fluid interaction," in *Proc. ACM SIGGRAPH/Eurographics Symp. Comput. Animation*, 2005, pp. 237–244.

[16] N. Akinci, M. Ihmsen, G. Akinci, B. Solenthaler, and M. Teschner, "Versatile rigid-fluid coupling for incompressible SPH," *ACM Trans. Graph.*, vol. 31, no. 4, pp. 62:1–62:8, 2012.

[17] A. Tartakovsky and P. Meakin, "Modeling of surface tension and contact angles with smoothed particle hydrodynamics," *Phys. Rev. E*, vol. 72, no. 2, 2005, Art. no. 026301.

[18] U. C. Bandara, A. M. Tartakovsky, M. Oostrom, B. J. Palmer, J. Grate, and C. Zhang, "Smoothed particle hydrodynamics pore-scale simulations of unstable immiscible flow in porous media," *Advances Water Resources*, vol. 62, pp. 356–369, 2013.

[19] A. M. Tartakovsky, N. Trask, K. Pan, B. Jones, W. Pan, and J. R. Williams, "Smoothed particle hydrodynamics and its applications for multiphase flow and reactive transport in porous media," *Comput. Geosciences*, vol. 20, no. 4, pp. 807–834, 2016.

[20] A. M. Tartakovsky and A. Panchenko, "Pairwise force smoothed particle hydrodynamics model for multiphase flow: Surface tension and contact line dynamics," *J. Comput. Phys.*, vol. 305, pp. 1119–1146, 2016.

[21] T. Yang, M. C. Lin, R. R. Martin, J. Chang, and S.-M. Hu, "Versatile interactions at interfaces for SPH-based simulations," in *Proc. ACM SIGGRAPH/Eurographics Symp. Comput. Animation*, 2016, pp. 57–66.

[22] J.-M. Hong, H.-Y. Lee, J.-C. Yoon, and C.-H. Kim, "Bubbles alive," *ACM Trans. Graph.*, vol. 27, no. 3, pp. 48:1–48:4, Aug. 2008.

[23] O. Busaryev, T. K. Dey, H. Wang, and Z. Ren, "Animating bubble interactions in a liquid foam," *ACM Trans. Graph.*, vol. 31, no. 4, pp. 63:1–63:8, Jul. 2012.

[24] B. Zhu, E. Quigley, M. Cong, J. Solomon, and R. Fedkiw, "Codimensional surface tension flow on simplicial complexes," *ACM Trans. Graph.*, vol. 33, no. 4, pp. 111:1–111:11, 2014.

[25] J. P. Morris, "Simulating surface tension with smoothed particle hydrodynamics," *Int. J. Numerical Methods Fluids*, vol. 33, no. 3, pp. 333–353, 2000.

[26] N. Thürey, F. Sadlo, S. Schirm, M. Müller-Fischer, and M. Gross, "Real-time simulations of bubbles and foam within a shallow water framework," in *Proc. ACM SIGGRAPH/Eurographics Symp. Comput. Animation*, 2007, pp. 191–198.

[27] P. W. Cleary, S. H. Pyo, M. Prakash, and B. K. Koo, "Bubbling and frothing liquids," *ACM Trans. Graph.*, vol. 26, no. 3, Art. no. 97, 2008.

[28] M. Ihmsen, N. Akinci, G. Akinci, and M. Teschner, "Unified spray, foam and bubbles for particle-based fluids," *Visual Comput.*, vol. 28, no. 6–8, pp. 669–677, 2012.

[29] M. Ihmsen, J. Bader, G. Akinci, and M. Teschner, "Animation of air bubbles with SPH," in *Proc. Int. Conf. Comput. Graph. Theory Appl.*, Mar. 5–7, 2011, pp. 225–234.

[30] R. Ando, N. Thürey, and R. Tsuruno, "Preserving fluid sheets with adaptively sampled anisotropic particles," *IEEE Trans. Vis. Comput. Graph.*, vol. 18, no. 8, pp. 1202–1214, Aug. 2012.

[31] B. Ren, Y. Jiang, C. Li, and M. C. Lin, "A simple approach for bubble modelling from multiphase fluid simulation," *Comput. Visual Media*, vol. 2, no. 1, pp. 171–181, 2015.

[32] S. Liu, Q. Liu, and Q. Peng, "Realistic simulation of mixing fluids," *Visual Comput.*, vol. 27, no. 3, pp. 241–248, 2011.

[33] M. Macklin, M. Müller, N. Chentanez, and T. Y. Kim, "Unified particle physics for real-time applications," *ACM Trans. Graph.*, vol. 33, no. 4, pp. 153:1–153:12, 2014.

[34] S. Clavet, P. Beaudoin, and P. Poulin, "Particle-based viscoelastic fluid simulation," in *Proc. ACM SIGGRAPH/Eurographics Symp. Comput. Animation*, 2005, pp. 219–228.

[35] B. Solenthaler, J. Schläfli, and R. Pajarola, "A unified particle model for fluid-solid interactions," *Comput. Animation Virtual Worlds*, vol. 18, no. 1, pp. 69–82, 2007.

[36] M. Ihmsen, N. Akinci, M. Gissler, and M. Teschner, "Boundary handling and adaptive time-stepping for PCISPH," in *Proc. Workshop Virtual Reality Interactions Phys. Simul.*, 2010, pp. 79–88.

[37] X. Shao, Z. Zhou, N. Magnenat-Thalmann, and W. Wu, "Stable and fast fluid-solid coupling for incompressible SPH," *Comput. Graph. Forum*, vol. 34, no. 1, pp. 191–204, 2015.

[38] A. M. Tartakovsky and A. Panchenko, "Pairwise force smoothed particle hydrodynamics model for multiphase flow: Surface tension and contact line dynamics," *J. Comput. Phys.*, vol. 305, pp. 1119–1146, 2016.

[39] M. Ihmsen, J. Orthmann, B. Solenthaler, A. Kolb, and M. Teschner, "SPH fluids in computer graphics," in *Proc. Eurographics-State Art Reports*, 2014, pp. 21–42.

[40] J. Yu and G. Turk, "Reconstructing surface of particle-based fluids using anisotropic kernels," in *Proc. ACM SIGGRAPH/Eurographics Symp. Comput. Animation*, 2010, pp. 217–225.

[41] M. Liu, P. Meakin, and H. Huang, "Dissipative particle dynamics with attractive and repulsive particle-particle interactions," *Phys. Fluids*, vol. 18, 2008, Art. no. 017101.

[42] J. J. Monaghan, "SPH without a tensile instability," *J. Comput. Phys.*, vol. 159, no. 2, pp. 290–311, 2000.

[43] J. Hua and J. Lou, "Numerical simulation of bubble rising in viscous liquid," *J Comput. Phys.*, vol. 222, no. 2, pp. 769–795, 2007.

[44] X. Y. Hu and N. A. Adams, "A multi-phase SPH method for macroscopic and mesoscopic flows," *J. Comput. Phys.*, vol. 213, no. 2, pp. 844–861, 2006.

[45] A. Barecasco, H. Terissa, and C. F. Naa, "Simple free-surface detection in two and three-dimensional SPH solver," in *Proc. Int. Symp. Comput. Sci.*, 2013.

Tao Yang received the BS degree from Tsinghua University, in 2013. He is currently working toward the PhD degree at Tsinghua University. His research interests include physically based simulation and rendering.

Ralph R. Martin recently retired after a long career in visual computing, and is now an emeritus professor of Cardiff University. He is currently enjoying life in the Welsh countryside.

Ming C. Lin received the BS, MS, and PhD degrees in electrical engineering and computer science from the University of California, Berkeley. She is currently the John R. & Louise S. Parker Distinguished professor of computer science with the University of North Carolina (UNC) at Chapel Hill, and an honorary chair professor (Yangtze Scholar) with Tsinghua University, China. She has received several honors and awards, including the IEEE, VGTC, VR Technical Achievement Award 2010 and 10 best paper awards. Her research interests include computer graphics, robotics, and human-computer interaction, with focuses on physically based modelling, sound rendering, haptics, algorithmic robotics, virtual environments, interactive techniques, geometric computing, and distributed interactive simulation. She has (co-) authored more than 250 refereed scientific publications, co-edited/authored four books. She is a former editor-in-chief of the *IEEE Transactions on Visualization and Computer Graphics* during 2011-2014, a member of the IEEE CS Board of Governors, and a member of the CRA-W Board of Directors. She is a fellow of the ACM and the IEEE.

Jian Chang received the PhD degree in computer graphics from Bournemouth University, in 2007. He is currently an associate professor in computer animation with Bournemouth University, United Kingdom. His research has focused on a number of topics relating to deformation and physically-based animation, geometric modelling, virtual surgery, character rigging and skinning (key techniques to animate a virtual agent) to address industry needs, and create impacts.

Shi-Min Hu received the PhD degree from Zhejiang University, in 1996. He is currently a professor in the Department of Computer Science and Technology, Tsinghua University, Beijing. His research interests include digital geometry processing, video processing, rendering, computer animation, and computer-aided geometric design. He has published more than 100 papers in journals and refereed conference. He is editor-in-chief of the *Computational Visual Media*, and on editorial board of several journals, including the *IEEE Transactions on Visualization and Computer Graphics* and the *Computer Aided Design and Computer & Graphics*.