

# Accepted Manuscript

Small-Scale Surface Details Simulation using Divergence-free SPH

Xiaokun Wang, Xiaojuan Ban, Sinuo Liu, Runzi He, Yuting Xu

PII: S1045-926X(18)30036-3  
DOI: [10.1016/j.jvlc.2018.07.005](https://doi.org/10.1016/j.jvlc.2018.07.005)  
Reference: YJ VLC 848

To appear in: *Journal of Visual Languages and Computing*

Received date: 16 March 2018  
Accepted date: 18 July 2018

Please cite this article as: Xiaokun Wang, Xiaojuan Ban, Sinuo Liu, Runzi He, Yuting Xu, Small-Scale Surface Details Simulation using Divergence-free SPH, *Journal of Visual Languages and Computing* (2018), doi: [10.1016/j.jvlc.2018.07.005](https://doi.org/10.1016/j.jvlc.2018.07.005)



This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

# Small-Scale Surface Details Simulation using Divergence-free SPH

Xiaokun Wang<sup>a,b</sup>, Xiaojuan Ban<sup>a,b,\*</sup>, Sinuo Liu<sup>a,b</sup>, Runzi He<sup>a,b</sup>, Yuting Xu<sup>a,c</sup>

<sup>a</sup>*School of Computer and Communication Engineering, University of Science and Technology Beijing, Beijing 100083, China*

<sup>b</sup>*Beijing Key Laboratory of Knowledge Engineering for Materials Science, Beijing, 100083, China*

<sup>c</sup>*Beijing No.4 High School*

## **Abstract**

To realistic and efficient capture of microscopic features of fluid surface, we proposed a novel method for creating small-scale surface details. In this paper we introduced a surface tension and adhesion model to simulate surface details, which refined the cohesion term and area minimization term. It modified the calculation of surface tension and adhesion and enlarged the support length for cohesion, which makes the microscopic characteristics of surface details more visible. In addition, we integrated this model with a Divergence-free SPH method which fulfills constant density condition and divergence-free condition simultaneously. The experimental results show that our method can well simulate small-scale details of fluid surface in various scenarios meanwhile improves the computational stability and efficiency.

*Keywords:* computer animation fluid simulation Divergence-free SPH  
surface tension

*2018 MSC:* 03-15, 99-00

\*Corresponding author

Email addresses: [wangxiaokun@ustb.edu.cn](mailto:wangxiaokun@ustb.edu.cn) (Xiaokun Wang), [banxj@ustb.edu.cn](mailto:banxj@ustb.edu.cn) (Xiaojuan Ban), [787666673@qq.com](mailto:787666673@qq.com) (Sinuo Liu), [hrz.claire@gmail.com](mailto:hrz.claire@gmail.com) (Runzi He), [xuxiaoyuting@126.com](mailto:xuxiaoyuting@126.com) (Yuting Xu)

## 1. Introduction

Fluid simulation is a popular research topic in computer graphics, virtual reality while has a huge application demand in three-dimensional visualization and human-computer interactions etc. In recent years, mesh-free methods had 5 a rapid development and have become a competitive alternative to mesh-based methods for fluid simulation in computer graphics. Among various mesh-free approaches, Smoothed Particle Hydrodynamics (SPH) is the most popular method for simulating fluid due to computational simplicity and efficiency. For SPH method's particle nature, it is suitable to simulate free surfaces and complex 10 phenomena such as spindrift, explosion. At present, SPH has been successfully used for fluid simulation[1, 2, 3, 4], fluid-solid coupling[5, 6, 7, 8], multiphase fluid[9, 10, 11, 12, 13],etc.

Although SPH has been used to produce animation of various fluid phenomena, simulating small-scale fluid surface details by particle-based method 15 is rarely discussed. From the perspective of simulation domain size, fluid animation includes large scale and small-scale. Large scale simulation usually has significant computational costs, so it always ignores the detail effects of local region for instance the simulation of ocean scene. For this type simulation, acceleration algorithm and efficiency improvement are necessary. While 20 small-scale simulation prevailingly pursues realistic effect of local details such as water droplet which needs to simulate microcosmic details. Using SPH method to simulate small-scale surface details realistically, however, is still a challenging problem. Because the computed density of particles at the fluid-air interface is lower than its real value, which is caused by lacking neighbor particles. It 25 results in negative pressure and causes particle clustering. In addition, the simulation of microscopic characteristics takes a large amount of calculation, and has the problem of time step restriction and numerical instability. For these reasons, we propose a surface small-scale surface details simulation method using Divergence-free SPH (DFSPH) that can obtain a good microcosmic detail effects 30 of free surface and improve the computational efficiency as well as stability.

## 2. Related Work

Since the SPH concept was first introduced to graphics community by the work of Desbrun[14], it has become a hot topic. Monaghan addressed simulating free surface flows with SPH[15], which serves as a basis for fluid simulation.

Mller et al.[1] proposed using gas state equation with surface tension and viscosity forces for fluid simulation, which also bring compressibility issue. Becker and Teschner[2] proposed weakly compressible SPH(WCSPH) employing a stiff equation of state (EOS) to reduce compressibility. It significantly increased realistic effects but the efficiency is limited by time step. To enforce incompressibility efficiently, iterative solvers are proposed including predictive-corrective incompressible SPH (PCISPH)[3], local poisson SPH (LPSPH)[16] and implicit incompressible SPH (IISPH)[4]. Recently, a promising approach for incompressible SPH has been proposed by Bender and Koschier[17], which combines two solvers to enforce low volume compression and ensure divergence-free velocity field.

For surface details simulation, traditional treatment is modeling surface tension, which is first inspired by the continuum surface force (CSF) model[18] where curvature is calculated through a color function[19]. This approach gives an accurate estimation of the effects of surface tension, but the surface's normal vector usually has deviations due to the SPH gradient formula, especially when the surface has sharp corners and few particles which will result in computational error in surface curvature. Adami et al.[20] present a new surface tension model to obtain accurate surface curvature using a reproducing divergence approximation without full support of kernel function. Another treatment of surface tension is microscopic model which imitates molecular attraction by cohesion among particles. Nugent and Posch[21] employed van der Waals equation of state to calculate attraction pressure, and extended the calculation range of attraction to obtain stable droplet effect. Tartakovsky and Meakin[22] proposed a molecular cohesion model to produce surface tension, which controls the attraction and repulsion among particles by cosine function. Becker and Teschner[2] used SPH kernel instead of cosine function to make calculation range of attrac-

tion within smoothing length  $h$ .

In SPH fluid simulation, early methods achieve surface tension effects through minimizing the surface curvature[1, 19]. However, these methods have some problems, for instance the curvature calculation is very sensitive to particle dis-  
65 order and the external force applied to fluid particles is asymmetrical which does not meet momentum conservation. Therefore, researchers used cohesion of adjacent fluid particles to solve these problems[2, 22]. However, only using cohesion can not ensure the surface area minimization and unreal flow effect. Clavet et al.[23] proposed double density relaxation method that can get strong  
70 surface tension effects. But it is not suitable for simulating low viscous liquids. Yu et al.[24] estimated curvature on the surface mesh and applied surface tension to the adjacent fluid particles surrounded by grids. While this method may not be able to detect isolated fluid blocks so that the surface tension effect cannot be exhibited at these areas. Akinci et al.[25] created surface tension  
75 by particles' interaction. Their method can handle large surface tension while maintaining momentum conservation. It generates repulsion for particles too close and prevents particle cluster without additional operation.

In addition, adsorption effect is also a characteristic of small-scale surface details. The adhesion makes fluid attracted by other substances. Steele et  
80 al.[26] proposed a Lagrangian method for viscous fluids that achieves the fluid-to-solid adsorption effects. But it has difficulties in simulating high viscous fluids because of using linear density kernel and anti-penetration constraints. Subsequently, Clavet et al.[23] modeled the adhesion of viscoelastic SPH fluids by a distance-based attraction term. Schechter and Bridson[27] realized the  
85 adhesion effect by calculating ghost velocity at each solid particle, which is computed using velocity of solid and tangential component velocity of the nearest fluid particle. Then adhesion is calculated using XSPH method based on artificial viscosity. He et al.[28] simulated adsorption effect using velocity constraint to enforce edge effect of different slip conditions. Akinci et al.[25] implemented  
90 a physically reasonable adhesion effect without additional treatments. In this study, forces are symmetrically applied to the adjacent pairwise fluid particles

as well as fluid and rigid particles to ensure momentum conservation. Recently, Yang et al.[29] proposed a refined surface tension model using pairwise forces, which accurately captures surface tension without extra forces or constraints.

### 95 3. SPH Fluid Simulation

In Lagrangian formalization[1], isothermal Navier-Stokes for incompressible fluids can be expressed as

$$\frac{d\rho_i}{dt} = -\rho_i \nabla \bullet \mathbf{v}_i \quad (1)$$

$$\rho_i \frac{D\mathbf{v}_i}{Dt} = -\nabla p_i + \rho_i g + \mu \nabla^2 \mathbf{v}_i \quad (2)$$

where  $\mathbf{v}_i$  is the velocity,  $\rho_i$  is the density,  $p_i$  is the pressure,  $\mu$  is the viscosity coefficient, and  $g$  represents the external force field. (1) is the mass conservation equation and (2) is the momentum conservation equation.

SPH is a widely used interpolation approach in particle systems. The core concept of SPH is using discrete particles to characterize the successive fields and use integration to approximate the fields. In the standard SPH, a scalar quantity  $A(x_i)$  of particle  $i$  at location  $x_i$  can be interpolated by the sum of quantities from neighbor particles,

$$\langle A(x_i) \rangle = \sum_j m_j \frac{A_j}{\rho_j} W(x_i - x_j, h) \quad (3)$$

where  $m_j$  and  $\rho_j$  represent particle mass and density respectively,  $W(x_i - x_j, h)$  is the smoothing kernel and  $h$  is the smoothing radius.

In SPH fluid simulation, fluid volume is represented as a set of particles. To simulate fluid with particles, the density  $\rho_i$  is interpolated by a weighted sum of the neighbor particle mass  $m_j$ ,

$$\rho_i = \sum_j m_j W(x_i - x_j, h) \quad (4)$$

The pressure  $P_i$  of particle  $i$  is usually represented as a function of density, such as the standard SPH method uses a ideal gas equation  $p_i = k(\rho_i - \rho_0)$ , where

<sup>115</sup>  $\rho_0$  is the rest density,  $k$  is a constant. Becker and Teschner[2] instead gas equation with Tait equation to enforce lower density variations and enhance the efficiency:

$$p_i = \frac{\rho_0 c_S^2}{\gamma} \left( \left( \frac{\rho_i}{\rho_0} \right)^\gamma - 1 \right) \quad (5)$$

where  $\gamma=7$  is stiffness parameters and  $c_S$  is velocity of sound.

Therefore forces between particles including pressure  $\mathbf{f}_i^P$  and viscous force <sup>120</sup>  $\mathbf{f}_i^V$  can be represented as

$$\mathbf{f}_i^P = - \sum_j m_j \left( \frac{P_i}{\rho_i^2} + \frac{P_j}{\rho_j^2} \right) \nabla W_{ij} \quad (6)$$

$$\mathbf{f}_i^V = \mu \sum_j m_j \frac{\mathbf{v}_{ji}}{\rho_j} \nabla^2 W_{ij} \quad (7)$$

In SPH framework, the Navier Stokes equation is discretized on particle locations, which becomes an ordinary differential equation as follow,

$$\rho_i \frac{\partial \mathbf{v}_i}{\partial t} = - \langle \nabla p_i \rangle + \mu \langle \nabla^2 \mathbf{v}_i \rangle + \mathbf{f}_i^{ext} \quad (8)$$

<sup>125</sup> where  $\mathbf{v}_i$  is the velocity,  $\mathbf{f}_i^{ext}$  is the external force,  $\mu$  is the viscosity coefficient,  $\langle \nabla p_i \rangle$  and  $\langle \nabla^2 \mathbf{v}_i \rangle$  are the approximations of pressure gradient and velocity Laplacian.

#### 4. Small-Scale Surface Details Simulation

In particle-based fluid simulation, surface tension is usually modeled based <sup>130</sup> on color field. Each particle has a attribute of color field, and the value is interpolated by SPH formula. For particle  $i$ , its color field  $c_i$  can be written as

$$c_i = \sum_j \frac{m_j c_j}{\rho_j} W_{ij} \quad (9)$$

The surface normal is calculated by  $\mathbf{n} = \nabla c$ , which can make the surface tension perpendicular to the fluid surface and point to the interior of the fluid.

<sup>135</sup> Generally, surface curvature can be measured by the divergence of surface normal vector. It can be expressed as follows:

$$\kappa = \frac{-\nabla^2 c}{|\mathbf{n}|} \quad (10)$$

Thus, surface tension can be constructed by normal vector and surface curvature that is:

$$\mathbf{F}^s = \sigma \kappa \mathbf{n} = -\sigma \nabla^2 c \frac{\mathbf{n}}{|\mathbf{n}|} \quad (11)$$

<sup>140</sup>

Although this way of simulating surface tension is easy, it has some shortcomings. On the one hand, if particles have less neighbors, it will bring calculation deviation because the term of  $-\nabla^2 c$ . Besides, the second derivative of  $-\nabla^2 c$  is sensitive to particles' disorder.

<sup>145</sup> Tartakovsky et al.[22] proposed a cohesion model to control the attraction and repulsion between particles by cosine function. However, this method causes clustering phenomenon. Becker et al. [2] used SPH kernel function instead of cosine function to create attraction, which is suitable for free surface with high curvature. But due to lack repulsion, it brings more severe clustering problem.  
<sup>150</sup> Because these methods do not consider the repulsion or lack enough repulsion, they can neither show surface tension effect nor ensure the surface area minimization. Therefore, Akinci[25] addressed a effective surface tension model using PCISPH. It considers molecular cohesion-repulsion as well as surface area minimum. However, when the surface tension is the main force for fluid particles, time step will be restricted. Besides, neighborhood deficiency at surface is not well solved. For these reasons, we modified Akinci's surface tension and adhesion model and presented a small-scale surface details simulation method that considers the above two aspects.

#### 4.1. Surface Tension

<sup>160</sup> The cohesion model considers the effect of attraction and repulsion that is similar to the intermolecular forces. It makes the particles attract each other

when the distance between them below a certain threshold and repel each other when the distance beyond the threshold. It can be represented as follows:

$$\mathbf{F}_i^c = -\alpha m_i \sum_j m_j \Delta \mathbf{x}_{ij} R(|\Delta \mathbf{x}_{ij}|) \quad (12)$$

Where  $j$  is the neighbor particle of  $i$ ,  $m$  is the mass,  $\mathbf{x}$  is position,  $\alpha$  is a coefficient,  $R$  is the spline function.

It is not difficult to find that the surface tension  $\mathbf{F}_i^c$  is determined by function

For comparison, we choose the spline function proposed by Akinci et al.[25], that is

$$R(r) = \frac{32}{\pi h^9} \begin{cases} (h-r)^3 r^3 & 2r > h \wedge r \leq h \\ 2(h-r)^3 r^3 - \frac{h^6}{64} & r > 0 \wedge 2r \leq h \\ 0 & otherwise \end{cases} \quad (13)$$

Where  $r$  is relative displacement,  $h$  is support radius. From Equation(13), it can be found that the force  $\mathbf{F}_i^c$  becomes maximum when the relative displacement is  $h$ . The force  $\mathbf{F}_i^c$  smoothly vanishes to 0 with relative displacement changing to  $h$ . Besides, For particles' relative displacement less than  $\frac{h}{2}$ , the force smoothly decreases to a negative value that results in repulsion forces.

However, Akinci's method can not completely pull the particles together and generates a cobweb-like structures when drop water on a board, as shown in Figure 2 (b) and 3 (b). To solve this problem, we integrated the divergence-free SPH solver which is interpreted in Section 5, and enlarged the support radius of the spline function as Yang et al.[29] did. Generally, the number of neighbor particles is generally about 30 for three-dimensional simulation[30] in computer community. In this paper, we enlarge ratio  $l = \sqrt[3]{70/30} \approx 1.32$  for the number of neighbor particles, thus the spline function can be amended as follow,

$$R(r) = \frac{32}{\pi(lh)^9} \begin{cases} (lh-r)^3 r^3 & \frac{lh}{2} < r \leq lh \\ 2(lh-r)^3 r^3 - \frac{(lh)^6}{64} & 0 < r \leq \frac{lh}{2} \\ 0 & otherwise \end{cases} \quad (14)$$

185 *4.2. Surface Tension Refinement*

The above surface tension model can construct the effect similar to molecular inter-atomic forces. But in order to simulate the microscopic features of fluid surface better, and to show the area minimize effect of fluid, we still need to add the correction term to surface tension. In order to avoid calculating the surface curvature in an explicit way, the normal vector is expressed as follows,

$$\mathbf{n}_i = \kappa \sum_j \frac{m_j}{\rho_j} \nabla W(|\mathbf{x}_i - \mathbf{x}_j|) \quad (15)$$

Where  $\kappa$  is the zoom factor.

The surface tension tends to reduce the curvature, and make different discrete sampling points have consistent direction. Besides, the curvature is proportional to normal difference, and we construct modifications as follows:

$$\mathbf{F}_i^f = -\beta m_i \sum_j (\mathbf{n}_i - \mathbf{n}_j) \quad (16)$$

195 Where  $\beta$  is the correction coefficient. As you can see, the correction term increases with the increase of the curvature. Its value is 0 at the flat areas and the interior of fluid. It avoids the standardization of the normal  $\mathbf{n}_i$  and the explicit calculation of curvature.

Above all, the revised surface tension can be expressed as:

$$\mathbf{F}_i^{cf} = \gamma_{ij} (\mathbf{F}_i^c + \mathbf{F}_i^f) \quad (17)$$

200 Where  $\gamma_{ij}$  is the surface tension control coefficient, and we adopt  $\gamma_{ij}$  represents near the fluid surface, and indicates inside the fluid. Therefore, we can use to enlarge the surface tension of the particles whose neighbors are insufficient.

*4.3. Adhesion*

Different from surface tension, adsorption force is generated by molecular interactions of different materials. In this subsection, adsorption force model is mainly aimed at the effect between fluid and solid. In this paper, we process the fluid-solid coupling simulation in the following steps: 1) sampling rigid

body surface as boundary particles, 2) using the boundary treatment formulas proposed by Akinci et al.[8] to calculate fluid density, in which the boundary particles are considered, 3) deriving each force formula. The fluid-solid coupling method can avoid the "adhesion effect" of fluid particles at the rigid boundary effectively. So this subsection uses an adhesion formula that can be directly applied to the fluid-solid coupling method [25].

$$\mathbf{F}_i^a = -\eta m_i \sum_k \psi_{b_k}(\mathbf{x}_i - \mathbf{x}_k) y(|\mathbf{x}_i - \mathbf{x}_k|) \quad (18)$$

Where  $k$  is boundary particle,  $\mathbf{x}$  is the position of particle,  $\eta$  is adsorption parameter, and  $\psi_{b_k}$  is the volume of boundary particle,  $y$  is the spline function, we construct  $y$  as follows:

$$y(r) = \frac{0.01}{h^5} \begin{cases} -(r - \frac{3}{4}h)^2 + \frac{h^2}{16} & \frac{h}{2} < r \leq h \\ 0 & otherwise \end{cases} \quad (19)$$

Due to that using the boundary treatment methods can solve the adhesion and clustering effect in border of fluid particle effectively, the above type just imposes the adsorption effect on the particles from  $\frac{h}{2}$  to  $h$  to attract each other. You can see that the adsorption is symmetrical, namely  $\mathbf{F}_k^a = -\mathbf{F}_i^a$ .

## 5. SPH Framework using Divergence-free Condition

Our small-scale surface details simulation model can minimize the surface area, prevent clustering, and achieve the surface tension and adhesion effects realistically. Nevertheless, when the surface tension or adhesion is the main force for fluid particles, time step will be restricted, for instance the water droplets scene. Some research has used PCISPH and adaptive time step[25, 29] to improve the simulation efficiency, but there still have problems such as numerical instability. Hence, we proposed a small-scale surface details simulation method using divergence-free SPH to enhance numerical stability and computational efficiency.

According to the continuity equation of Navier-Stokes equations in Lagrangian description[1], incompressible fluids must satisfy the divergence-free condition  $\nabla \cdot \mathbf{v}$  and density condition  $\frac{D\rho}{Dt} = 0$ . Therefore, fluids have divergence-free velocity and constant density is incompressible. However, density deviations caused by numerical time integration and accumulated over the simulation is inevitable. To adjust the density error, another condition  $\rho - \rho_0 = 0$  should be achieved. In other words, to simulate incompressible fluids, the constant density condition  $\rho - \rho_0 = 0$  and divergence-free condition  $\nabla \cdot \mathbf{v}$  must be guaranteed simultaneously. At present, most SPH solvers for incompressible fluids only employ constant density condition while velocity divergence is usually not fulfilled. Obviously, SPH Framework consider both the conditions can be efficient and stable, so we integrate the small-scale surface details model into a SPH solver that fulfills constant density condition and divergence-free condition [17]. It needs to be emphasized that we take surface tension and adhesion to be one of the internal forces, and add them to the correct iterations to guarantee stability and efficiency.

To ensure the condition  $\frac{D\rho_t}{Dt} = 0$  of particle  $i$ , divergence-free SPH[17] correct the divergence error in the neighborhood of particle  $i$  by pressure forces. The pressure force density of particle  $i$  is defined by

$$\mathbf{f}_i^P = -\nabla p_i \quad (20)$$

The pressure gradient  $\nabla p_i$  is calculated by SPH spatial derivative equation of discretizing ideal gas equation:

$$\nabla p_i = \tau_i^P \nabla \rho_i = \tau_i^P \sum_j m_j \nabla W_{ij} \quad (21)$$

where  $\tau_i^P$  is the stiffness parameter.

For conserve momentum and forces symmetry, the condition  $\mathbf{f}_i^P + \sum_j \mathbf{f}_{i \rightarrow j}^P = 0$  must be satisfied, where  $\mathbf{f}_{i \rightarrow j}^P$  is the force densities acting from particle  $i$  to particles  $j$ . The pressure force densities  $\mathbf{f}_{i \rightarrow j}^P$  is determined with respect to

position :

$$\mathbf{f}_{i \rightarrow j}^p = -\frac{\partial p_i}{\partial \mathbf{x}_j} = \tau_i^p m_j \nabla W_{ij} \quad (22)$$

260 The divergence error employs the SPH formulation of divergence[30]:

$$\frac{D\rho_i}{Dt} = \sum_j m_j (\mathbf{v}_i - \mathbf{v}_j) \nabla W_{ij} \quad (23)$$

Evidently, pressure forces bring the velocity changes  $\Delta \mathbf{v}_i = \Delta t \frac{\mathbf{f}_i^p}{\rho_i}$  and  $\Delta \mathbf{v}_j = \Delta t \frac{\mathbf{f}_{j \rightarrow i}^p}{\rho_j}$ . To correct the deviation, it creates the following equation,

$$-\frac{D\rho_i}{Dt} = \Delta t \sum_j m_j \left( \frac{\mathbf{f}_i^p}{\rho_i} - \frac{\mathbf{f}_{j \rightarrow i}^p}{\rho_j} \right) \nabla W_{ij} \quad (24)$$

265 Then can be solved by substituting Equations (20) and (22) in Equation (24):

$$\tau_i^p = \frac{D\rho_i}{Dt} \frac{1}{\Delta t} \cdot \frac{\rho_i}{\xi_i} \quad (25)$$

where  $\xi_i = \left( \left| \sum_j m_j \nabla W_{ij} \right|^2 + \sum_j |m_j \nabla W_{ij}|^2 \right)$  is determined by particle positions.

270 When calculating pressure forces with  $\tau_i^p$ , the condition  $\frac{D\rho_i}{Dt} = 0$  will be satisfied that means divergence-free velocity field is guaranteed in the neighborhood of particle  $i$ . To achieve a globally divergence-free velocity field, the pressure forces are computed iteratively. The factors  $\xi_i$  can be precomputed before the iterative process since they only depend on the current positions.

The total pressure force  $\mathbf{F}_{i-tot}^p$  for a particle  $i$  is as follows,

$$\begin{aligned} \mathbf{F}_{i-tot}^p &= \frac{m_i}{\rho_i} \mathbf{f}_i^p + \sum_j \frac{m_i}{\rho_i} \mathbf{f}_{i \rightarrow j}^p \\ &= -m_i \sum_j m_j \left( \frac{\tau_i^p}{\rho_i} + \frac{\tau_j^p}{\rho_j} \right) \nabla W_{ij} \end{aligned} \quad (26)$$

275 Therefore, the resultant force of pressure and surface tension and adhesion is

$$\mathbf{F}_{i-tot}^{p,cf,a} = \mathbf{F}_{i-tot}^p + \mathbf{F}_i^{cf} + \mathbf{F}_i^a \quad (27)$$

The divergence-free solver is outlined in line 26-32 of Algorithm 1.

280 The constant density solver iteratively decreases the deviation of  $\rho - \rho_0$ , which works similarly to the divergence-free solver. This is reasonable because the precomputed factors  $\xi_i$  can be reused to reduce the computational costs significantly.

The density error  $\rho_i^* - \rho_0$  can be expressed as

$$\rho_i^* - \rho_i = \Delta t \frac{D\rho_i}{Dt} = \Delta t \sum_j m_j (\mathbf{v}_i^* - \mathbf{v}_j^*) \nabla W_{ij} = \Delta t^2 \sum_j m_j \left( \frac{\mathbf{f}_i^p}{\rho_i} - \frac{\mathbf{f}_{i \rightarrow j}^p}{\rho_i} \right) \nabla W_{ij} \quad (28)$$

285

Similar to compute  $\tau_i^p$ , the stiffness parameter  $\tau_i^p$  used to correct this error can be represented as

$$\tau_i^p = \frac{1}{\Delta t^2} (\rho_i^* - \rho_0) \xi_i \quad (29)$$

290 As a result, the total pressure force  $\bar{\mathbf{F}}_{i-tot}^p$  can be computed like  $\mathbf{F}_{i-tot}^p$ , that is

$$\bar{\mathbf{F}}_{i-tot}^p = -m_i \sum_j m_j \left( \frac{\tau_i^p}{\rho_i} + \frac{\tau_j^p}{\rho_j} \right) \nabla W_{ij} \quad (30)$$

Therefore, the resultant force of pressure and surface tension and adhesion is

$$\bar{\mathbf{F}}_{i-tot}^{p,cf,a} = \bar{\mathbf{F}}_{i-tot}^p + \mathbf{F}_i^{cf} + \mathbf{F}_i^a \quad (31)$$

295

Line12-18 in Algorithm 1 outlines the constant density solver. Note that performing the density stabilization before computing a divergence-free velocity field does not impose any restrictions since both steps are executed in a loop.

## 6. Implementation and Results

We implemented some experiments in this section to demonstrate the effectiveness of our method. The hardware platform is a Intel Xeon E5-2637W

v2 processor with 3.50 GHz, 80 GB memory and a Intel Xeon E5-2687W v4 processor with 3.00 GHz, 72 GB memory. OpenMP was used to parallelize the fluid simulation. For surface construction, we extracted surface meshes using Marching Cubes with anisotropic kernel function. Our method employed spatial background grid to enforce spatial Hash algorithm. In our simulation we used OpenGL 3D graphics library to perform real-time rendering and employed Blender’s Ray-tracking engine Cycles for offline high quality rendering. The commonly used parameters of all the experiments are shown in Table 1.

Table 1: Commonly used settings and parameters of all the experiments.

Item	Value
Simulation domain size	$8m \times 8m \times 8m$
Fluid density	$1000 \text{ kg/m}^3$
Smooth and kernel function	Cubic spline function
Smooth radius	0.2m
Width of the fluid particles	0.1m

Figure 1 illustrates the experiment of cubic water volume deformed to sphere without gravity. The first row is the simulation in particle view, the second row is the results after surface reconstruction, and the third row shows the rendered results. The process of the experiment is as follows: at the beginning the water volume is cubic, then due to the impact of surface tension, the fluid particles are attracted to each other and the water volume deformed to sphere gradually until the force balance. By observations, if only cohesion is applied to fluid particles, particles can attract to each other, but this can not guarantee fluid surface area minimization. As shown in Figure 1(a), the forces among particles are balanced and the water volume is not deformed. But the surface of water volume is uneven, that is, the particles’ location is distributed non-uniform. Since we employ a attractive-repulsive cohesion model with a surface area minimization term to simulate surface tension. It ensures the surface area minimization when the particle is force-balanced. Figure 1(b) is the results of our method. By

comparison, we can find that the droplet achieved spherical using our method and the fluid surface is smoother and more realistic.

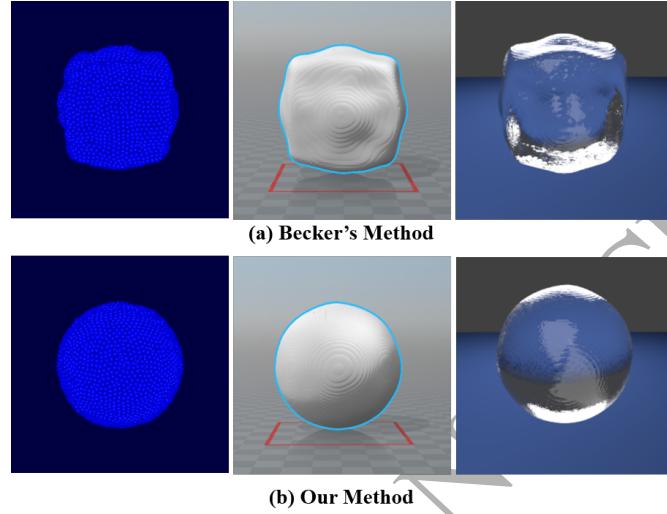


Figure 1: The scene of cubic water volume transformed into sphere.

325     Figure 2 shows the particle view of cubic water droplet falls to a board. The experimental process is as follows: Initially, cubic droplet is static in the air and the droplet is only affected by the surface tension without any external force, so cubic droplet can deform into sphere. For a moment, gravity is added, and then the droplet starts to fall freely. When the droplet contacts with the board,  
 330     it tumbles around the board until it stops. Figure 2 (a) is the experimental results without surface tension. As shown, at beginning the cubic droplet did not change to a spherical water column. When the droplet collides with the board, fluid particles move around independently and rest uniformly on the board finally. Figure 2 (b) is the results of Akinci's method. When the droplet moves  
 335     on the board, it appears a cobweb-like structure due to numerical instability. Figure 2 (c) shows the experimental effect of our method. In this experimental scenario, our method is more stable in particle motion and produces stronger surface tension among particles compared with Akinci's method. Because the stronger constraint, the cobweb-like structure did not appear during the simu-

<sup>340</sup> lation in our method. Obviously, our method ensures the stability of particle motion in intense collision when simulates fluids with large surface area. At the same time, it can minimize the surface area and make the fluid surface more smooth.

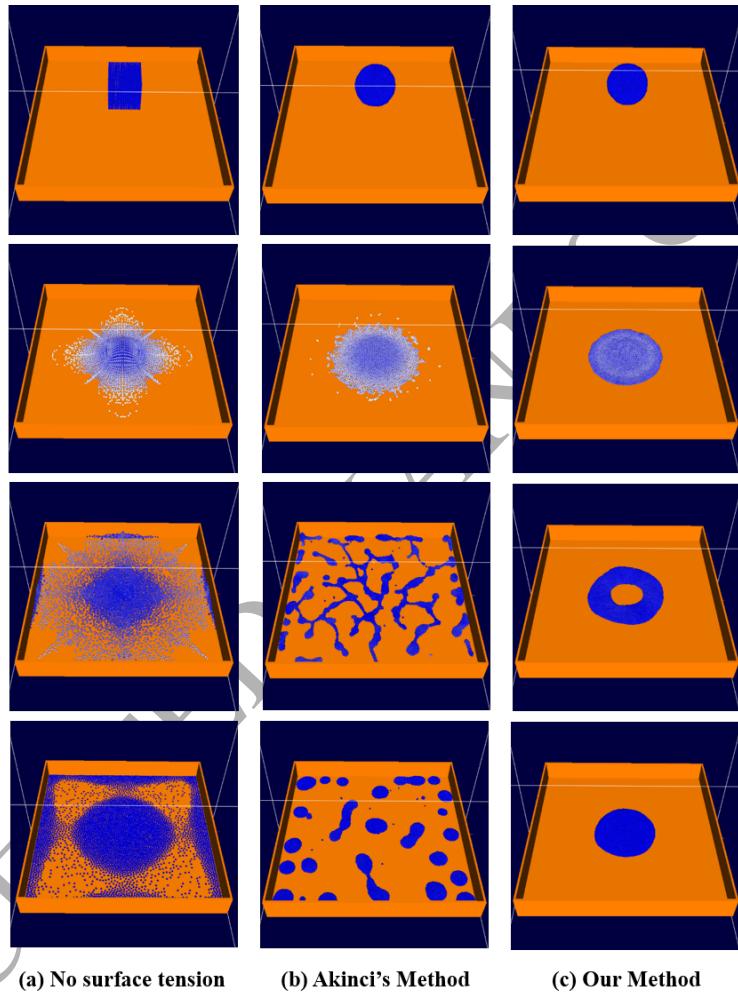


Figure 2: Particle view of cubic droplet falls to a board.

<sup>345</sup> Figure 3 exhibits the rendering results of cubic droplet falls to a board, which is corresponding to Figure 2. From Figure 3(a), we find that the fluid particles are distributed uniformly without surface tension, especially at the edges of the

board, which is obviously not consistent with the real scenario. This problem has been significantly improved by adding surface tension, as shown in Figure 3(b) and Figure 3(c). However, comparing Figure 3(b) with 3(c), we find that our  
350 method is more stable in dealing with violent collisions and does not appear the cobweb-like structure that similar to Figure 3(b). Meanwhile, due to stronger constraint force among particles, our method is more suitable for simulating droplet with large surface area and without the problem appeared in Figure 3(b).

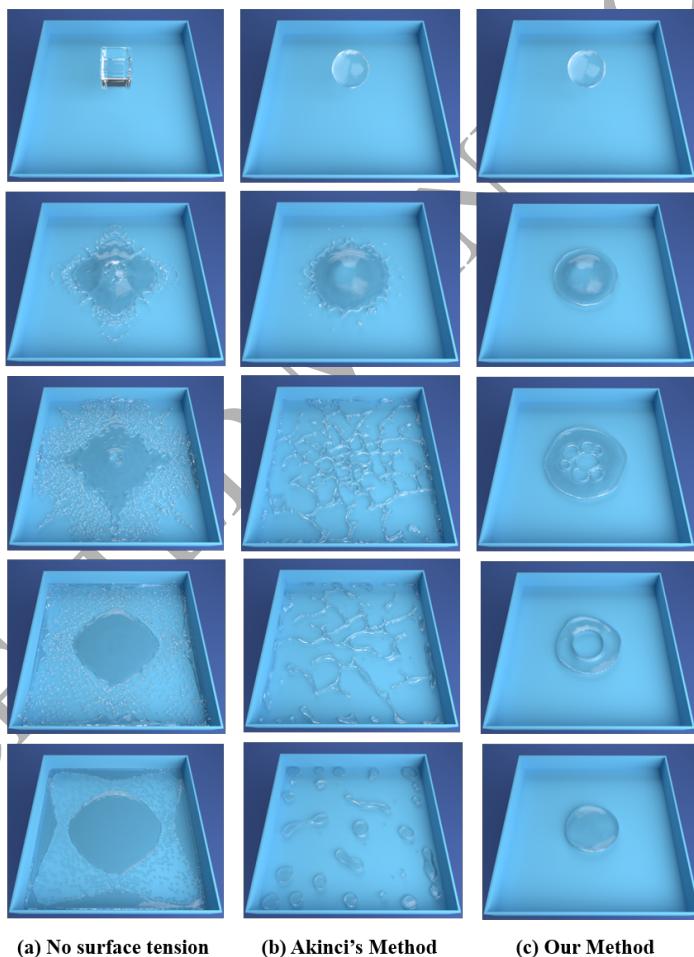


Figure 3: Rendering results of cubic droplet falls to a board.

Figure 4 illustrates a scene of pouring water on a solid object. By contrast, it  
 355 can be found that the particles do not attract with each other and are uniformly  
 dispersed for no surface tension, as shown in Figure 4(a). On the contrary,  
 particles are not in a discrete state but interact with each other when appending  
 surface tension shown in Figure 4(b) and 4(c), which is more approximate to  
 real scene. However, comparing Figure 4 (b) with 4 (c), when dealing with  
 360 the violent collision between fluids and the solid object, the stability of fluid  
 movement using our method is much better. The fluids are not break apart  
 violently and not split into smaller droplets, which means splash is not drastic  
 employing our method.

Figure 5 is the rendering results of pouring water on a solid object, which is  
 365 corresponding to Figure 4. From this picture, we can see that the fluid has an  
 obvious sense of granule without surface tension, especially at the edges. While  
 particles are no longer scattered after adding surface tension. Nonetheless, when  
 using Akinci's method, the poured water has problems of violent splashing and  
 splitting, as shown in Figure 5(b). In contrary, constraints between particles is  
 370 stronger and simulation stability is better adopting our method. At the same  
 time, our method is more effective to minimize the surface area, so the fluid has  
 a larger adsorptive area.

Table 2 is the running time comparison of cubic droplet falls to a board  
 and pouring water on a solid object. As shown, under the same scenario and  
 375 parameter settings, our method has an obvious improvement in efficiency, which  
 gains about 3 times speedup.

Table 2: Algorithm running time comparison of different experiment.

Experiment	Method	Fluid particles	Solid particles	Operation time(s)	Speedup ratio
Cubic droplet falls to a board	Akinci's method	6321	15001	914	-
	Our method	6321	15001	336	2.72
Pouring water on a solid object	Akinci's method	132751	28707	1211	-
	Our method	132751	28707	401	3.02

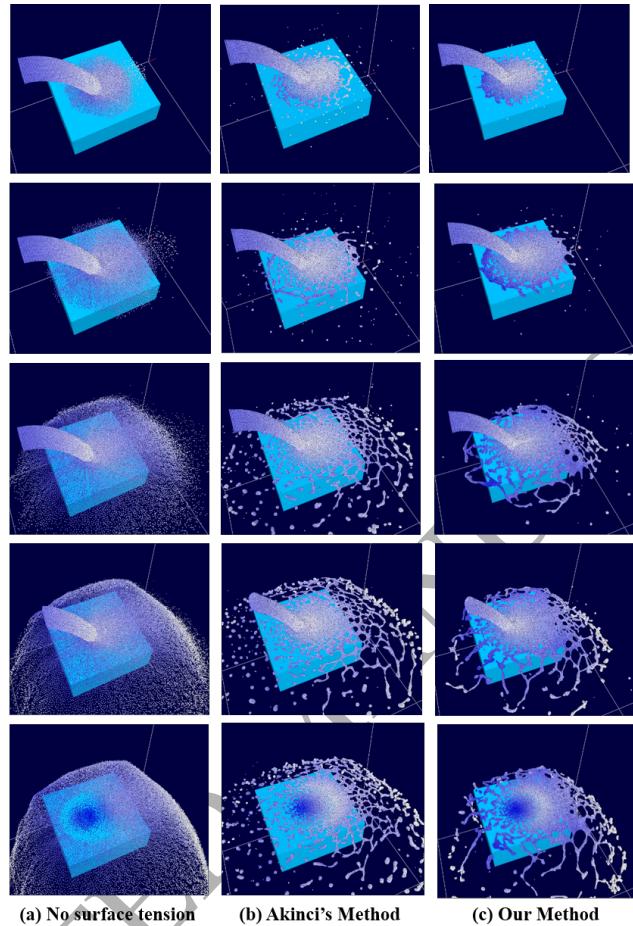


Figure 4: Particle view of pouring water on a solid object.

Figure 6 exhibits the average iterations of our method and Akinci's method in different time steps. Because PCISPH compresses the pressure based on a global time-dependent stiffness, the stability and performance of PCISPH are completely relied on the time step. As a result, if the time step is big, it will lead to over-adjustment. While small time step will have a negative impact on convergence speed. However, under the same time step, DFSPH is more stable and efficient, the average iterations of our method is less. In addition, compared with the 1% density error threshold of PCISPH, our method can

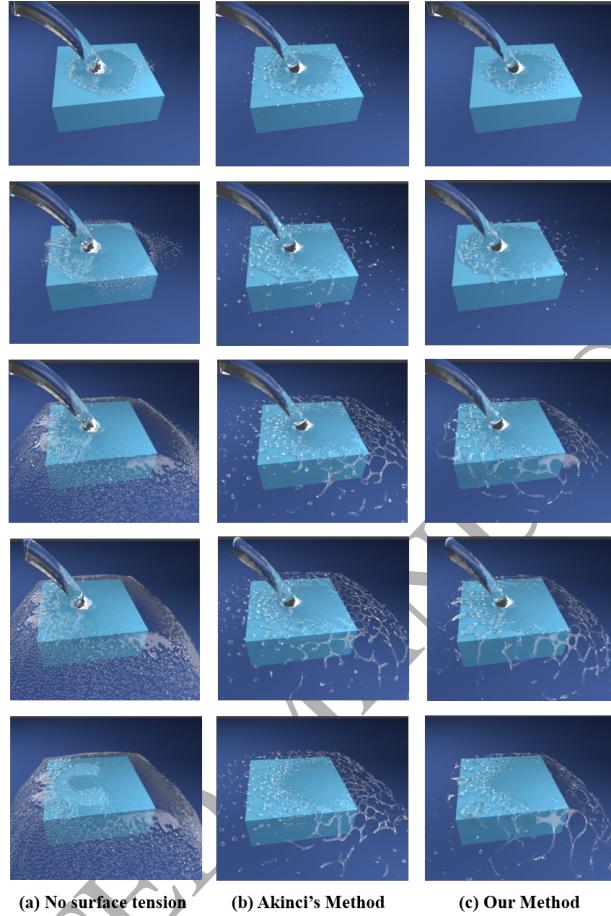


Figure 5: Rendering results of pouring water on a solid object.

385 enforce the density error within 0.1% which ensures numerical stability during  
 iterative process. Therefore, our method can work stably at large time step,  
 while PCISPH would cause numerical instability problem. In summary, our  
 method has obvious advantages in operation efficiency.

390 Figure 7 shows the adhesion effect of fluid to solid using our method. This  
 experiment using the same scene as Figure 5. The adhesion stiffness is controlled  
 by coefficient  $\gamma$ . As shown in Figure 6, with the increase of adhesion coefficient  
 ( $\gamma$  is 1, 5, 10 from left to right), the adhesion effect is gradually increased. In

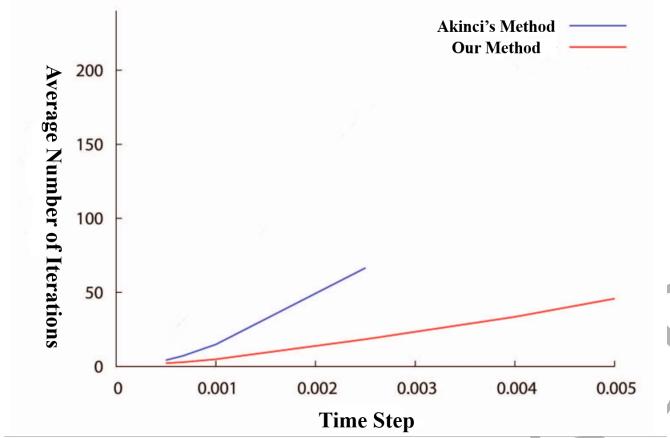


Figure 6: The average iterations in different time steps.

consequence, the fluid appears visibly adsorptive phenomenon in the moving process (as the third picture in Figure 7). This also proves the validity of the adhesion model.

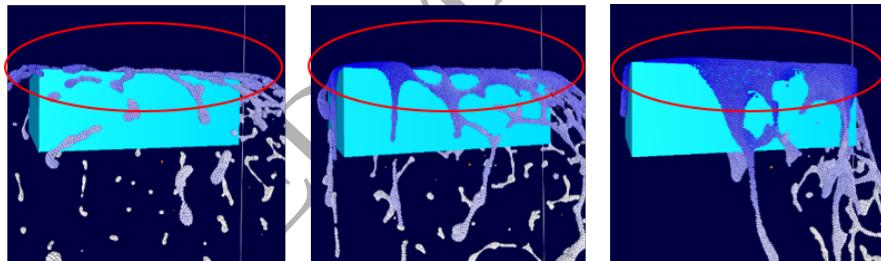


Figure 7: Experiment of adhesion effect (adhesion coefficient  $\gamma$  is 1,5,10 respectively)

395

Figure 8 shows the effect of droplets under different wet conditions ( $\delta \leq 1, \gamma \geq 0.001$ ) using our method. The upper and lower rows are particle view and rendering view respectively. When adjusting coefficients of our surface tension and adhesion model, we can simulate the droplet shape from completely wet condition (the first column) to thoroughly non-wetting condition in the same scene (the fourth column).

400

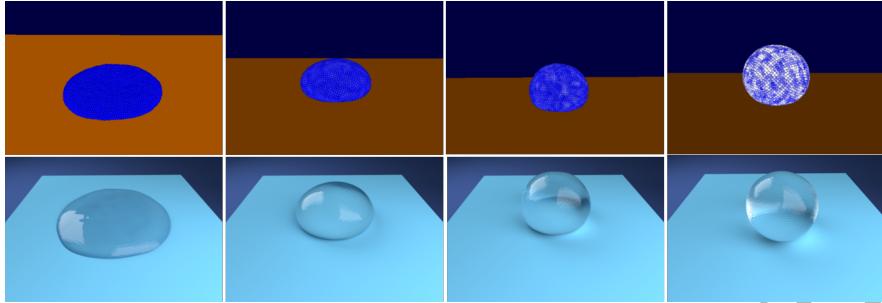


Figure 8: Simulation of droplets under different wet conditions( $\delta < 1, \gamma \geq 0.001$ ).

## 7. Conclusion

We have proposed a new method to create small-scale surface details for fluid simulation. It refined the cohesion term and area minimization term and enlarged the support length for cohesion, which makes the microscopic characteristics of surface details more visible. Besides, our method combined the small-scale surface details model with the DFSPH to improve the computational stability and efficiency. The experimental results show that our method can capture the microscopic features of fluid surface more vividly compare to previous methods, for instance the area minimization effect and adsorption effect of solid boundary. Moreover, our method can keep a good performance in large and complex scene because adding the divergence-free solver. However, there still has problem in simulating multiphase fluids and viscous fluids when surface tension and adhesion is the dominant force acting on particles. So future work would be extending it to multiphase fluids and viscous fluids.

## Acknowledgment

This work was supported by the National Natural Science Foundation of China (under Grant Nos.61702036 and 61572075) and Fundamental Research Funds for the Central Universities of China (under Grant No.FRF-TP-17-012A1) and China Postdoctoral Science Foundation (under Grant No.2017M620619).

## References

### References

- [1] M. Müller, D. Charypar, M. Gross, Particle-based fluid simulation for interactive applications, Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation (2003) 154–159.  
 URL <http://dl.acm.org/citation.cfm?id=846276.846298>
- [2] M. Becker, M. Teschner, Weakly compressible sph for free surface flows, Proceedings of the 2007 ACM SIGGRAPH/Eurographics Symposium on Computer Animation (2007) 209–217.  
 URL <http://dl.acm.org/citation.cfm?id=1272690.1272719>
- [3] B. Solenthaler, R. Pajarola, Predictive-corrective incompressible sph, Acm Transactions on Graphics 28 (3) (2009) 1–6. doi:10.1145/1576246.1531346.  
 URL <https://dl.acm.org/citation.cfm?id=1531346>
- [4] M. Ihmsen, J. Cornelis, B. Solenthaler, C. Horvath, M. Teschner, Implicit incompressible sph, IEEE Transactions on Visualization & Computer Graphics 20 (3) (2014) 426–435. doi:10.1109/TVCG.2013.105.
- [5] M. Müller, S. Schirrm, M. Teschner, B. Heidelberger, M. Gross, Interaction of fluids with deformable solids, Computer Animation & Virtual Worlds 15 (3-4) (2010) 159–171. doi:10.1002/cav.v15:3/4.
- [6] T. Lenaerts, B. Adams, P. Dutr, Porous flow in particle-based fluid simulations, Acm Transactions on Graphics 27 (3) (2008) 49. doi:10.1145/1360612.1360648.
- [7] M. Becker, H. Tessendorf, M. Teschner, Direct forcing for lagrangian rigid-fluid coupling, IEEE Transactions on Visualization & Computer Graphics 15 (3) (2008) 493–503. doi:10.1109/TVCG.2008.107.

- [8] N. Akinci, M. Ihmsen, G. Akinci, B. Solenthaler, M. Teschner, Versatile rigid-fluid coupling for incompressible sph, ACM Trans. Graph. 31 (4) (2012) 62:1–62:8. doi:10.1145/2185520.2185558.  
450 URL <http://doi.acm.org/10.1145/2185520.2185558>
- [9] M. Mller, R. Keiser, A. Nealen, M. Pauly, M. Gross, M. Alexa, Point based animation of elastic, plastic and melting objects, in: ACM Siggraph/eurographics Symposium on Computer Animation, 2004, pp. 141–151. doi:10.1145/1028523.1028542.  
455
- [10] R. Keiser, B. Adams, D. Gasser, P. Bazzi, P. Dutre, M. Gross, A unified lagrangian approach to solid-fluid animation, in: Point-Based Graphics, 2005. Eurographics/IEEE VGTC Symposium Proceedings, 2005, pp. 125–148. doi:10.1109/PBG.2005.194073.
- [11] B. Solenthaler, J. Schlfli, R. Pajarola, A unified particle model for fluid-solid interactions., Computer Animation & Virtual Worlds 18 (1) (2010) 69–82. doi:10.1002/cav.v18:1.  
460
- [12] B. Ren, C. Li, X. Yan, M. C. Lin, J. Bonet, S. M. Hu, Multiple-fluid sph simulation using a mixture model, Acm Transactions on Graphics 33 (5) (2014) 1–11. doi:10.1145/2645703.  
465
- [13] T. Yang, M. C. Lin, R. R. Martin, J. Chang, S. M. Hu, Versatile interactions at interfaces for sph-based simulations, ACM Siggraph/eurographics Symposium on Computer Animation (2016) 57–66doi:10.2312/sca20161223.
- [14] M. Desbrun, M. P. Gascuel, Smoothed particles: A new paradigm for animating highly deformable bodies, Computer Animation and Simulation (1996) 61–76doi:10.1007/978-3-7091-7486-9\_5.  
470
- [15] J. J. Monaghan, Simulating free surface flows with sph, Journal of Computational Physics 110 (2) (1994) 399–406. doi:10.1006/jcph.1994.1034.

- [16] X. He, N. Liu, S. Li, H. Wang, G. Wang, Local poisson sph for viscous incompressible fluids, in: Computer Graphics Forum, 2012, pp. 1948–1958.  
doi:10.1111/j.1467-8659.2012.03074.x.
- [17] J. Bender, K. Dan, Divergence-free smoothed particle hydrodynamics, in: ACM SIGGRAPH / Eurographics Symposium on Computer Animation, 2015, pp. 147–155. doi:10.1145/2786784.2786796.
- [18] J. U. Brackbill, D. B. Kothe, C. Zemach, A continuum method for modeling surface tension, Academic Press Professional, Inc., 1992. doi:10.1016/0021-9991(92)90240-Y.
- [19] J. P. Morris, Simulating surface tension with smoothed particle hydrodynamics, International Journal for Numerical Methods in Fluids 33 (3) (2000) 333–353. doi:10.1002/1097-0363(20000615)33:3<333::AID-FLD11>3.0.CO;2-7.
- [20] S. Adami, X. Y. Hu, N. A. Adams, A new surface-tension formulation for multi-phase sph using a reproducing divergence approximation, Journal of Computational Physics 229 (13) (2010) 5011–5021. doi:10.1016/j.jcp.2010.03.022.
- [21] S. Nugent, H. A. Posch, Liquid drops and surface tension with smoothed particle applied mechanics, Physical Review E 62 (4 Pt A) (2000) 4968–4975. doi:10.1103/PhysRevE.62.4968.
- [22] A. Tartakovsky, P. Meakin, Modeling of surface tension and contact angles with smoothed particle hydrodynamics, Physical Review E Statistical Nonlinear & Soft Matter Physics 72 (2) (2005) 026301. doi:10.1103/PhysRevE.72.026301.
- [23] S. Clavet, P. Beaudoin, P. Poulin, Particle-based viscoelastic fluid simulation, in: ACM Siggraph/eurographics Symposium on Computer Animation, SCA 2005, Los Angeles, Ca, Usa, July, 2005, pp. 219–228. doi:10.1145/1073368.1073400.

- [24] J. Yu, C. Wojtan, G. Turk, C. Yap, Explicit mesh surfaces for particle based fluids, Computer Graphics Forum 31 (2pt4) (2012) 815824. doi: 10.1111/j.1467-8659.2012.03062.x.
- 505 [25] N. Akinci, G. Akinci, M. Teschner, Versatile surface tension and adhesion for sph fluids, Acm Transactions on Graphics 32 (6) (2013) 1–8. doi: 10.1145/2508363.2508395.
- 510 [26] K. Steele, D. Cline, P. K. Egbert, J. Dinerstein, Modeling and rendering viscous liquids: Research articles, Computer Animation & Virtual Worlds 15 (3-4) (2004) 183–192. doi:10.1002/cav.v15:3/4.
- [27] R. Bridson, R. Bridson, Ghost SPH for animating water, ACM, 2012. doi: 10.1145/2185520.2185557.
- 515 [28] X. He, N. Liu, G. Wang, F. Zhang, S. Li, S. Shao, H. Wang, Staggered meshless solid-fluid coupling, Acm Transactions on Graphics 31 (6) (2012) 1–12. doi:10.1145/2366145.2366168.
- [29] T. Yang, M. C. Lin, R. R. Martin, J. Chang, S. M. Hu, Versatile interactions at interfaces for sph-based simulations, in: ACM Siggraph/eurographics Symposium on Computer Animation, 2016, pp. 57–66. doi:10.2312/sca.20161223.
- 520 [30] M. Ihmsen, J. Orthmann, B. Solenthaler, A. Kolb, M. Teschner, Sph fluids in computer graphics, Eurographics Association (2014) 21–42doi: 10.2312/egst.20141034.

**Algorithm 1** Small-Scale Surface Details Simulation Based on DFSPH

---

```

1: while animating do
2:   for each particle  $i$  do
3:     search neighbor particles  $N_i(t)$  of  $i$ 
4:   end for
5:   for each particle  $i$  do
6:     calculate the density  $\rho_i(t)$ 
7:     calculate the factor  $\xi_i(t)$ 
8:   end for
9:   for each particle  $i$  do
10:    calculate non-pressure force  $\mathbf{F}_i^{adv}(t)$ 
11:   end for
12:   for each particle  $i$  do
13:      $\mathbf{v}_i^* = \mathbf{v}_i + \Delta t \frac{\mathbf{F}_i^{adv}}{m_i}$ 
14:   end for
15: end while
16: while  $(\rho_{avg} - \rho_0 > \eta^\rho) \vee (\text{iter} < 2)$  do
17:   for each particle  $i$  do
18:     calculate  $\rho_i^*$ 
19:   end for
20:   for each particle  $i$  do
21:      $\tau_i^\rho = \frac{\rho_i^* - \rho_0}{\Delta t^2} \xi_i, \tau_j^\rho = \frac{\rho_j^* - \rho_0}{\Delta t^2} \xi_j$ 
22:     calculate surface tension  $\mathbf{F}_i^{cf}$ , adhesion  $\mathbf{F}_i^a$ 
23:      $\mathbf{v}_i^* := \mathbf{v}_i^* + \Delta t \frac{\mathbf{F}_{i-tot}^{p,cf,a}}{m_i}$ 
24:   end for
25: end while
26: for each particle  $i$  do
27:    $\mathbf{x}_i(t + \Delta t) = \mathbf{x}_i(t) + \Delta t \mathbf{v}_i^*$ 
28: end for
29: for each particle  $i$  do
30:   search neighbor particles  $N_i(t + \Delta t)$  of  $I$ 
31: end for
32: for each particle  $i$  do
33:   calculate the density  $\rho_i(t + \Delta t)$ 
34:   calculate the factor  $\xi_i(t + \Delta t)$ 
35: end for
36: while  $\left( \left( \frac{D\rho}{Dt} \right)_{avg} > \eta^d \right) \vee (\text{iter} < 1)$  do
37:   for each particle  $i$  do
38:      $\frac{D\rho_i}{Dt} = -\rho_i \nabla \cdot \mathbf{v}_i^*$ 
39:   end for
40:   for each particle  $i$  do
41:      $\tau_i^p = \frac{1}{\Delta t} \frac{D\rho_i}{Dt} \xi_i, \tau_j^p = \frac{1}{\Delta t} \frac{D\rho_j}{Dt} \xi_j$ 
42:     calculate surface tension  $\mathbf{F}_i^{cf}$ , adhesion  $\mathbf{F}_i^a$ 
43:      $\mathbf{v}_i^* := \mathbf{v}_i^* + \Delta t \frac{\mathbf{F}_{i-tot}^{p,cf,a}}{m_i}$ 
44:   end for
45: end while
46: for each particle  $i$  do
47:    $\mathbf{v}_i(t + \Delta t) = \mathbf{v}_i^*$ 
48: end for

```

---