A new SPH density formulation for 3D free-surface flows

Sandra Geara, Sylvain Martin, S. Adami, W. Petry, J. Allenou, B. Stepnik, Olivier Bonnefoy

Version of Record: https://www.sciencedirect.com/science/article/pii/S0045793021003145
Manuscript_1d7a1638c7dcd3232e46002fe49bd3b2

A new SPH density formulation for 3D free-surface flows

S. Geara[a,c,d,*], S. Martin[a], S. Adami[b], W. Petry[c], J. Allenou[d], B. Stepnik[d], O. Bonnefoy[a]

[a] *Mines Saint-Etienne, Univ Lyon, CNRS, UMR 5307 LGF, Centre SPIN, F - 42023 Saint-Etienne, France*
[b] *Chair of Aerodynamics and Fluid Mechanics, Technical University of Munich, 85748 Garching, Germany*
[c] *Research Neutron Source Heinz Maier-Leibnitz (FRM II), Technical University of Munich, 85748 Garching, Germany*
[d] *FRAMATOME, CERCA TM, ZI les Bérauds BP 1114, 26104 Romans-sur-Isère, France*

Abstract

In this paper, a new density formulation for free surface simulations using SPH is presented. This new approach is applicable to surface-tension driven free surface flows with strong topological changes. The density is corrected for each particle by analytically calculating the missing volume of the support domain. This calculation depends on two parameters: the local curvature and the distance of each particle to the free surface. This method was validated and compared with the density evolution method for two test cases: the square droplet and the Rayleigh-Plateau instability. It shows more stable results and a better representation of the free surface.

Keywords:
SPH, Free Surface, Density correction, Curvature, Surface Tension

*Corresponding author
E-mail address : sandra.geara@emse.fr

Nomenclature

a	Acceleration (m/s^2)
C	Correction coefficient
c_s	Speed of sound (m/s)
F^ν	Viscous force (N/m^3)
F^s	Surface tension force (N/m^3)
g	External body acceleration (m/s^2)
h	Smoothing length (m)
k	Curvature (1/m)
m	Mass (kg)
n	Number of time steps
$\vec{n}$	Normal vector
P	Pressure (Pa)
r	Position (m)
R_c	Cut-off distance (m)
t	Time (s)
T	Dimensionless time
v	Velocity (m/s)
V	Volume (m^3)
δ_s	Surface delta function
δt	Time step (s)
γ	EOS coefficient
ν	Dynamic viscosity (Pa.s)
ρ	Density (kg/m^3)
ρ_0	Reference density (kg/m^3)
σ	Surface tension coefficient (N/m)

1. Introduction

Smoothed Particle Hydrodynamics (SPH) is a mesh-free Lagrangian numerical method that was first introduced independently in 1977 by Lucy [1] and Gingold and Monaghan [2] to solve astrophysical problems. Since 1977, the SPH method has been developed and improved significantly to model a wide range of problems, especially in fluid dynamics. This method models a continuous fluid by discretizing it with a series of fluid particles that move through space and time. The continuity of the fluid and its properties are recovered by the spatial convolution of the physical properties of the particles by a smoothing kernel function.

For free surface flows, the standard SPH approximations suffer from the lack of full support. This mainly affects the density estimation and, consequently, potential pressure oscillations. Many techniques were proposed in the literature to overcome this problem.

The first attempt to apply SPH to free surface flow was introduced by Monaghan [3] for the simulation of wave run-up and breaking in shallow water. Monaghan used the continuity equation to calculate the change rate of density as a function of the velocity gradient. He also introduced the XSPH velocity variant algorithm to improve the particle distribution. In XSPH, the velocity of each particle is modified to take into account the average velocity of all nearby particles. A second common approach to smooth out the pressure oscillations is to apply a density filter. Colagrossi and Landrini [4] applied a periodic re-initialization of the density field based on the Moving

Least Squares Approach (MLS) proposed by Belytschko et al. [5] and Dilts [6]. This proposed kernel correction ensures a consistent interpolation of the density field. Bonet and Lok [7] considered this kernel correction unsuitable for explicit schemes. Instead, they proposed a simpler correction of the density field by considering a constant correction instead of a linear one or what is known as the Shepard filter. These density corrections methods should be applied every n time steps, where n is typically chosen between 20 and 50. Bonet and Lok [7] also introduced another correction technique based on the modification of the kernel gradient by a correction matrix. Various schemes for applying the kernel gradiant correction can be found in the literature [8, 9]. A combination of these two techniques, i.e. the constant kernel correction and the kernel gradient correction, is also possible. Furthermore, Molteni and Colagrossi [10] and Ferrari et al. [11] introduced an artificial density diffusion term to smooth out the numerical noise of the pressure field. The intensity of this numerical correction is to be defined according to the problem at hand. More recently, Seo et al. [12] proposed to correct the density of each particle based on shape functions adopted from the interpolation scheme of FEM. Several review papers discussing free-surface flows in SPH have been published, amongst other [13, 14].

All these procedures give good results, some of them are more suitable than the others depending on the application. However, these corrections can be insufficient when dealing with surface tension force. In this case, the force should be applied on surface particles at each time step, thus it is important to estimate the density of these particles accurately.

For applications where the surface tension force is not the dominant force, Calderon et al. [15] developed very recently a geometrical formulation that improves the Shepard correction coefficient. In their work, the dimensionality of the problem is reduced by defining the kernel in such a way that the volume integral is cast into a surface integral. Despite the promising results, this method still contains complex and computationally expensive numerical evaluation [16].

In this work, we propose a new density evaluation method. The idea behind this method is to calculate analytically a coefficient that represents the weight of the missing particles from the support domain and use it to correct the density summation for particles near the free surface. This new correction factor depends on the distance to the surface as well as on the curvature of the free surface. A variant of this method has originally been proposed by Herant [17] and also used by Vanhala and Cameron [18] in order to implement boundary conditions in cases where boundaries are only used for confinement. To our best knowledge, this method has never been extensively described in the literature and never been used to simulate free surface flows with surface tension.

2. SPH governing equations and numerical model

In the Lagrangian description, the Navier-Stokes equations read as

$$\frac{d\rho}{dt} = -\rho \nabla \cdot \vec{v} \tag{1}$$

and

$$\frac{d\vec{v}}{dt} = \vec{g} + \frac{1}{\rho}[-\nabla P + F^{\nu} + F^{s}] \tag{2}$$

where P and $\vec{g}$ are the material pressure and body force, respectively. F^{ν} denotes the viscous force and F^{s} is the surface tension force. In weakly compressible SPH, the pressure is related to the density by means of an Equation-Of-State (EOS)

$$P = \frac{\rho_0 c_s^2}{\gamma}\left[\left(\frac{\rho}{\rho_0}\right)^{\gamma} - 1\right] \tag{3}$$

The exponent γ is usually taken equal to 7 for water. The artificial sound of speed c_s is estimated based on a scale analysis of the Navier-Stokes equation presented by Morris et al. [19].

The SPH method is based on the discretization of the domain with a set of particles used to interpolate continuous field functions. The contribution or weight of each particle is determined by a kernel function. The value of any field function f at a position r can be estimated according to the following summation form

$$f(\vec{r}) \approx \sum_{j}^{N} \frac{m_j}{\rho_j} f(\vec{r}_j) W(|\vec{r} - \vec{r}_j|, h), \tag{4}$$

where m_j and $\vec{r}_j$ are the mass and position of particle j, respectively. W represents the weighting kernel function with the smoothing length h.

Here, we use the cubic spike kernel function with a compact support of $3h$. We prefer this kernel for its stability against compression [20] and it has the advantage of being a one piece function which simplifies the analytical calculation of the correction factor C, defined later in Section 3.

$$W_{ij} = \begin{cases} \alpha(1 - \frac{r}{3h})^3 & \text{if } \frac{r}{h} \leq 3 \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

with

$$\alpha = \begin{cases} \frac{2}{3h} & \text{dim=1} \\ \frac{9}{10\pi h^2} & \text{dim=2} \\ \frac{15}{27\pi h^3} & \text{dim=3} \end{cases} \tag{6}$$

Various SPH formulations can be obtained depending on the assumptions and purpose of the simulation [21]. In our study, we used the formulation proposed by Adami [22]. The momentum equation is written as

$$\frac{d\vec{v}_i}{dt} = \frac{1}{m_i} \sum_j -(V_i^2 + V_j^2) \left[\tilde{p}_{ij} \vec{\nabla} W_{ij} + \frac{2\nu_i \nu_j}{\nu_i + \nu_j} \frac{\vec{v}_{ij}}{r_{ij}} \frac{\partial W_{ij}}{\partial r_{ij}} \right] + \frac{\vec{F}_i^{(s)}}{m_i} \tag{7}$$

where V and ν are the volume and dynamic viscosity of each particle, respectively. $\vec{v}_{ij} = \vec{v}_i - \vec{v}_j$ is the relative velocity between particles i and j and $\vec{r}_{ij} = |\vec{r}_i - \vec{r}_j|$ is the distance between the two particles. $\vec{F}_i^{(s)}$ is the surface tension force of particle i, and $\tilde{p}_{ij}$ is the density-weighted inter-particle

averaged pressure

$$\tilde{p}_{ij} = \frac{\rho_i p_j + \rho_j p_i}{\rho_i + \rho_j} \tag{8}$$

In the present work, the Continuum Surface Force (CSF) approach is used to model the surface tension force. This approach was initially proposed by Brackbill [23] and then extended by Morris et al. [24] to be applied in the framework of SPH. The surface tension force is expressed as a volumetric force applied only on particles close to the interface

$$\vec{F}^{(s)} = \sigma k \vec{n} \delta_s \tag{9}$$

Here, δ_s is the surface-delta function used to smooth the surface tension force over a band of particles near the free surface, $\vec{n}$ is the normal vector and k is the curvature. In general and for this work, it is taken equal to the norm of the color gradient factor.

Due to the lack of full support near the free-surface, additional correction should be applied to accurately estimate the normal direction and the curvature of the interface. In this work, the correction matrix for the kernel gradient [7] is used only for the calculation of the local normal vector and the curvature is estimated following Sirotkin et al. [20]. The full details for the surface tension force can be seen in [20].

For single phase simulation, the color function of all liquid particles in

the bulk is equal to $c_j^0 = 1$. The smoothing of the color function gives

$$c_i = \sum_j \frac{m_j}{\rho_j} c_j^0 W_{ij} \tag{10}$$

The value of the smoothed color function c_i is theoretically equal to 1 for particles in the bulk with a full kernel support. Contrary, close to the free surface the number of neighbouring particles decreases and thus the value of c_i will also decrease. In general, a threshold value of 0.9 is defined for detecting surface particles. However with this method voids inside the fluid phase can be detected as free-surface, to avoid this problem a more accurate surface tracking algorithms was used. The cover-vector technique was proposed by Barecasco et al. [25]. Each particle i is represented by a sphere and has a cover vector defined by : $\vec{b}_i = \sum_j \frac{\vec{r}_{ij}}{||\vec{r}_{ij}||}$. For detecting surface particles, a cone of angle θ_i (threshold angle) is considered around each b_i. If at least one of the neighbouring particles j is inside the cone, then particle i is not considered as a surface particle, otherwise particle i belongs to the free surface. The value of θ_i plays an important role in boundary particle detection, it is usually chosen equal to $\frac{\pi}{3}$.

Once the surface particles are detected, the normal vector n_i can be expressed as the gradient of the color function : $\vec{n}_i = \frac{\nabla c_i}{|\nabla c_i|}$. To overcome the problem of lacking full support for free-surface flows, the correction matrix to adjust the kernel gradient [7], can be used.

For each particle i, the correction matrix is defined as

$$L_i = \sum_j V_j \nabla W_{ij} \otimes \vec{r_{ij}} \tag{11}$$

and the corrected kernel gradient is expressed as follows

$$\nabla \tilde{W}_{ij} = L_i^{-1} \nabla W_{ij} \tag{12}$$

In SPH, the local curvature for each particle is calculated as the divergence of the normalised normal vector as follows

$$k_i = \sum_j (\vec{n_j} - \vec{n_i}).\vec{\nabla} \tilde{W}_{ij} \tag{13}$$

Furstenau et al. [26] found that equation 13 overestimates the curvature by almost a factor of 2 in 3D simulations. To overcome this problem, they calculated the mean curvature based on a local coordinate system. First, the global curvature tensor is calculated by

$$\Xi_{ij} = \sum_j min(R_i, R_j)(\vec{n_j} - \vec{n_i}) \otimes \nabla \tilde{W}_{ij} \tag{14}$$

Then, this global curvature tensor is rotated into the local coordinate system as follows : $\Xi_{kl} = T_{ki}^T \Xi_{ij} T_{jl}$. The metric T is calculated as the scalar product of the basis vectors $T_{ij} = G_i.L_j$, where G is the global basis vectors which form a unity matrix and $L = (\vec{n}, t^1, t^2)$, with t^1 and t^2 being two tangent vectors. After the transformation, the 3D matrix is reduced to a 2D matrix

by deleting the row and column related to the normal vectors, in this case the first row and column. Then, the eigenvalues of the 2x2 matrix are calculated. They are considered as the principal curvature k_1 and k_2. The mean curvature is finally obtained by $k = \frac{1}{2}(k_1 + k_2)$.

The time integration scheme used in this work is the kick-drift-kick scheme also known as the velocity Verlet algorithm used by Monaghan [21]. It starts with the prediction of the intermediate velocity

$$\vec{v}^{(t+\frac{1}{2}dt)} = \vec{v}^{(t)} + \frac{1}{2}\delta t \vec{a}^{(t)}. \tag{15}$$

Then, the position is updated by

$$\vec{r}^{(t+dt)} = \vec{r}^{(t)} + \delta t \vec{v}^{(t+\frac{1}{2}dt)}. \tag{16}$$

The new density and forces are calculated at this new position, the acceleration is deduced from Newton's second law of motion. Finally the velocity is updated by

$$\vec{v}^{(t+dt)} = \vec{v}^{(t+\frac{1}{2}dt)} + \frac{1}{2}\delta t \vec{a}^{(t+dt)}. \tag{17}$$

For stability reasons the time step δt should be limited. To satisfy all the conditions, the minimum of these four time steps is used [24]

- CFL condition

$$\delta t \leq 0.25 \ \frac{h}{c_s + v_{ref}} \tag{18}$$

- Surface tension condition

$$\delta t \leq 0.25 \left[\frac{\rho h^3}{2\pi\sigma}\right]^{1/2} \tag{19}$$

- Viscous diffusion condition

$$\delta t \leq 0.125 \ \frac{\rho h^2}{\nu} \tag{20}$$

- Body force condition

$$\delta t \leq 0.25 \left[\frac{h}{g}\right]^{1/2} \tag{21}$$

3. New density formulation

The novelty in this work is a new version of the density summation taking into account the proximity of a free surface by algebraic correction .

From the different density correction techniques cited above, the Shepard density filter is the most common one. In this case the density will be

calculated using the density evolution formula

$$\frac{d\rho_i}{dt} = \sum_j m_j \vec{v}_{ij} \cdot \vec{\nabla} W_{ij} \tag{22}$$

Then, the density is corrected every n time steps by the Shepard summation as follows

$$\rho_i = m_i \frac{\sum_j W_{ij}}{\sum_j V_j W_{ij}} \tag{23}$$

Note, equation 23 can be written in the form of a corrective factor C_i, which will depend in this work on the distance of each particle to the free surface and the local curvature.

$$\rho_i = C_i \; m_i \sum_j W_{ij} \tag{24}$$

If we consider only the geometric aspect, then the correction coefficient C can be expressed as follows

$$C_i \approx \frac{V_{\text{SD}}}{V_{SD} - V_{\text{empty}}} \tag{25}$$

where V_{SD} and V_{empty} are the volume of the support domain of particle i and the empty volume represented by the missing particles at the free surface, respectively. Figure 1 represents the support domain area and empty volume of a particle at a distance d from the free surface. The support domain is represented by the circle of radius R_c and the empty volume is represented by the hatched moon. The curvature of the free surface is equal to k.

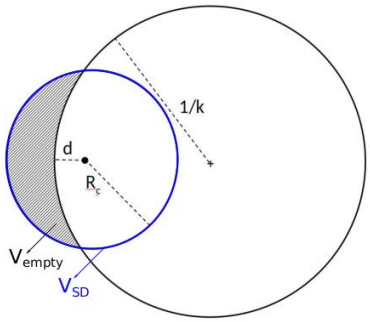


Figure 1: Representation of the empty volume (hatched moon) for a particle at a distance d from the free surface.

In fact, the impact of missing neighbours on the density calculation depends on their location. The impact of removing a distant neighbour is much smaller than the impact of a closer one. Therefore, the weight of each missing particle, represented by the kernel function, should be taken into account when expressing C_i. V_{empty} is modified to $\int_{V_{\text{empty}}} W(r)dV$ and V_{SD} is modified to $\int_{V_{\text{SD}}} W(r)dV$ which is the integral of the kernel function over all the support domain. Equation 25 will be then modified to

$$C_i = \frac{1}{1 - \frac{\int_{V_{\text{empty}}} W(r)dV}{\int_{V_{\text{SD}}} W(r)dV}} \tag{26}$$

Because the kernel functions are normalized, the integral of W over all the support domain should theoretically be equal to 1. The integral over the empty volume can be expressed as

$$\int_{V_{\text{empty}}} W(r)dV = \int_{d}^{R_c} W(r)2\pi(r - I)dr \tag{27}$$

where I represents the distance between the center and the intersection of

the two spheres. By writing I as a function of $1/k$, r and d, we obtain

$$I = -\frac{r^2 + (\frac{1}{k} - d)^2 - (\frac{1}{k})^2}{2(\frac{1}{k} - d)} \tag{28}$$

Finally, by substituting equation 28 in equation 27, we obtain

$$\int_{V_{\text{empty}}} W(r)dV = \int_d^{R_c} W(r)2\pi r \left[r + \frac{r^2 + (\frac{1}{k} - d)^2 - (\frac{1}{k})^2}{2(\frac{1}{k} - d)} \right] dr \tag{29}$$

For 3D simulations, the integral over the empty volume can be expressed as a function of $\delta = \frac{d}{h}$ and $\varrho = \frac{1}{|k|h}$. d represents the distance between particle i and the nearest surface particle and k is the curvature of this surface particle. The kernel function used in this case is the Spike with $R_c = 3h$ (Equation 5). Two cases must be considered.

For positive curvatures,

$$\int_{V_{\text{empty}}} W(r)dV = \frac{(\delta - 3)^5(-4\delta^2 + 3\delta + 28\delta\varrho + 42\varrho + 27)}{20412(\delta - \varrho)}, \tag{30}$$

and for negative curvatures

$$\int_{V_{\text{empty}}} W(r)dV = \frac{(\delta - 3)^5(-4\delta^2 + 3\delta - 28\delta\varrho - 42\varrho + 27)}{20412(\delta + \varrho)}. \tag{31}$$

The same scheme can be used in 2D simulations by integrating over a surface instead of a volume. However, the analytical solution for surface integrals is surprisingly not as simple as for 3D cases, thus numerical in-

tegration is necessary in these cases. The same problem was reported by Hérant [17] although he used another kernel function for the integration. We limit our study here to 3D simulations.

It should be noted that when δ tends to or is bigger than ϱ, equation 30 can diverge. Because the second derivative of the color function is very sensitive to the particle distribution, the curvature k is limited to $\frac{DIM-1}{3h}$. In 3D, $k_{max} = \frac{2}{3h}$. Plus, the maximum value of δ is defined by the radius of the support domain, which is in this case equal to 3. This means that :

$$0 < \delta < 3$$
$$\frac{2}{3} < \varrho < \infty$$

To avoid this problem, the correction coefficient C_i is only calculated for values of ϱ bigger than the maximum value of δ. In this case, for values of ϱ below 3, the standard Shepard summation (Equation 23) is used for the density calculation. The implementation of this correction technique is detailed in algorithm 1.

Algorithm 1 Density Computation

At t=0 isnitial calculation of ρ_i with equation 23
for all particles i **do**
  Define surface particles
  Calculate curvature k_i
end for
for all particles i **do**
  if particle i is a surface particle **then**
    Set $\delta = \frac{1}{2}$
    Set $\varrho = \frac{1}{|k_i|h}$
  else
    Find the nearest surface particle j
    Compute the distance $d = d_{ij} + \frac{h}{2}$
    Set $\delta = \frac{d}{h}$
    Set $\varrho = \frac{1}{|k_j|h}$
  end if
  if $\varrho > 3$ **then**
    Compute C_i with equation 26 and ρ_i with equation 24
  else
    Compute ρ_i with equation 23
  end if
end for

4. Numerical validation

4.1. Cubic droplet

As first test case, we analyse is the transformation of a cubic droplet into a spherical droplet. The main purpose of this example is to test the new correction coefficient for estimating the density at the free surface. For this example, an initial cube of $l_x = l_y = l_z = 0.6$ with a total number of 3375 particles is considered. Dimensionless properties were assigned to the fluid phase, see table 1. The spherical droplet formed after t=2 is presented in figure 2 for three different methods of density evaluation.

Table 1: Properties of the fluid phase

ρ	ν	σ
1	0.2	1

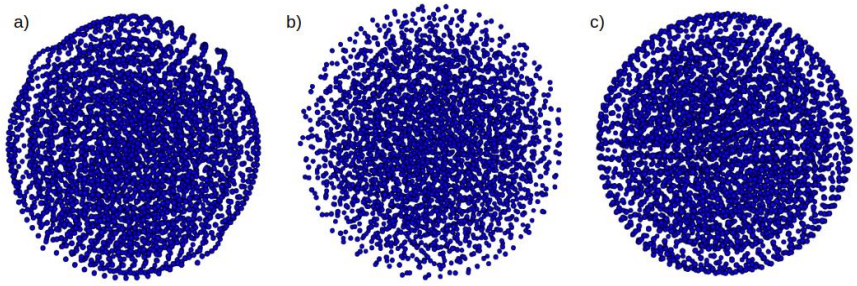
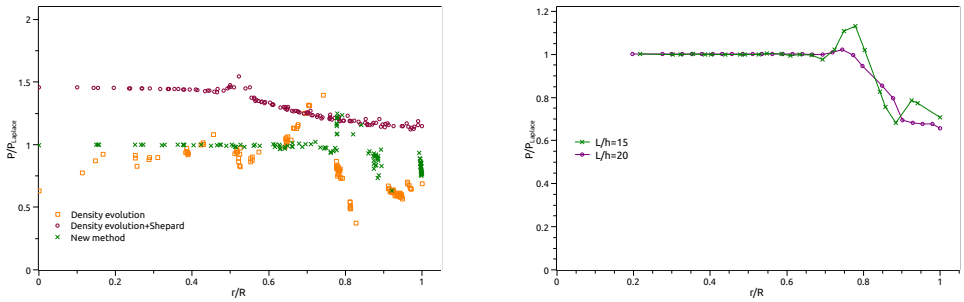


Figure 2: Particle position at t=2: a) Density evolution (equation 22) b) Density evolution with Shepard filter (n=30) and c) New correction based on curvature.

The Young-Laplace law gives the theoretical pressure inside the droplet at equilibrium

$$P = \frac{2\sigma}{R_{droplet}} = \frac{2\sigma}{L}\left(\frac{4\pi}{3}\right)^{1/3}. \tag{32}$$



(a) Comparison of the different methods

(b) New density method with different resolutions

Figure 3: Pressure inside the droplet as a function of the radius

Figure 3 presents the pressure inside the droplets as a function of the radius (all particles are considered). With the new method, the homogeneity of the density field is strongly improved. However, we can still see the effect of the initial square shape on the pressure profile, where we see a pressure variation in the regions around the corners which is mainly a geometric effect due to the initial particle stacking. By comparing the results of the three methods, it can be found that the new method outperforms the existing approaches for particle distribution at the interface. Figure 4 shows the variation of the average pressure inside the droplet as a function of time. The problem with applying the Shepard filter every n time steps is that the density of the surface particles can change instantaneously, which results in strong pressure fluctuations. As a consequence, the drop is unstable and keeps expanding. Accordingly, the pressure inside the droplet keeps increasing, see the corresponding curve (Evolution + Shepard) in figure 4. It should be noted that this new calculation method does not add much computational cost for the simulations because the curvature is already calculated for the surface tension force. It was found that the computational cost is increased

by around 5% compared to the density evolution and by 2% compared to the density evolution with the Shepard summation.

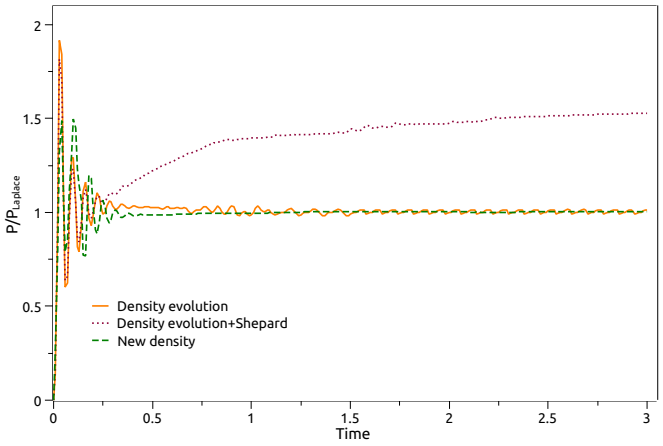


Figure 4: Average pressure inside the droplet as a function of time

Table 2 shows the percentiles of the number of particles as a function of pressure, for a better comparison between the density calculation methods. It can be seen that with the new method, the pressure variation is smaller, with almost all the particles having a pressure variation within 30% of the theoretical Laplace pressure.

Table 2: Fraction in (%) of particles having a pressure within a certain range around $P_{Laplace}$

Pressure / Method	P ±1%	P ±10%	P ±30%	P ±50%
Density evolution	1.066	25	60	100
New method (L/h = 15)	20	54	91	100

4.2. Rayleigh Plateau instability

We consider the Rayleigh-Plateau instability for validation of our method in dynamic situations. The test case simulation was first presented by Dai and Schmidt [27] using a new moving mesh algorithm and then reproduced by Olejnik and Szewc [28] using SPH with two phases. For this simulation, we consider a fully periodic cubic ($L \times L \times L$) domain containing in its center a liquid column of length L and a radius $r = L/10$. An initial perturbation is imposed by a sinusoidal velocity field applied to the liquid column in its longitudinal direction

$$u_x = u_0 \sin(\frac{2\pi x}{L}). \tag{33}$$

Because the particles are placed on a Cartesian grid and do not form a perfect cylinder, we let the system first relax by imposing a high viscosity. Then, we switch to the real properties of water before applying the velocity perturbation. The dimensionless numbers describing this case are the Reynolds number (Re=18) and the Weber number (We=1.4) calculated based on u_0. The dimensionless time is defined by

$$T = \frac{t}{\sqrt{\frac{\rho r_0^D}{\sigma}}}. \tag{34}$$

The simulations have been performed with a total number of fluid particles equal to 18000, which corresponds to $h = L/82$. Figure 5 shows several particle snapshots at different times for the classical and the new approach. Clearly, the new method shows more stable results compared to the classic density evolution with the density filter. It should be noted that the two pic-

tures at the last row correspond to the particles positions after the break-up. With this new density formulation, only 0.005% of the total particle number disintegrate (as compared to 1.2% for the standard method of the density evolution with the Shepard filter). Because the break-up does not occur at the same moment for both cases, the time T is not the same. The break-up time is defined as the time when the ligament diameter between droplets is composed of less than two particles.

We can clearly see from figure 5 that the break-up with the new method is much smoother. For the first case the unstable flying particles are seen at the moment of the break-up. In fact, the density evolution with no additional correction can accumulate errors with time. However, the additional main issue of this method, for this specific test case with fragmentation, is that the density could change drastically when a bulk particle becomes a surface particle, thus creating instabilities.

The correction coefficient for the Shepard technique was compared with the new correction coefficient. Figure 6 presents these two correction coefficients for surface particles as a function of particle position along the column axis. The main difference is that this coefficient is more dispersed in the case of the Shepard correction compared to the new technique presented in this article.

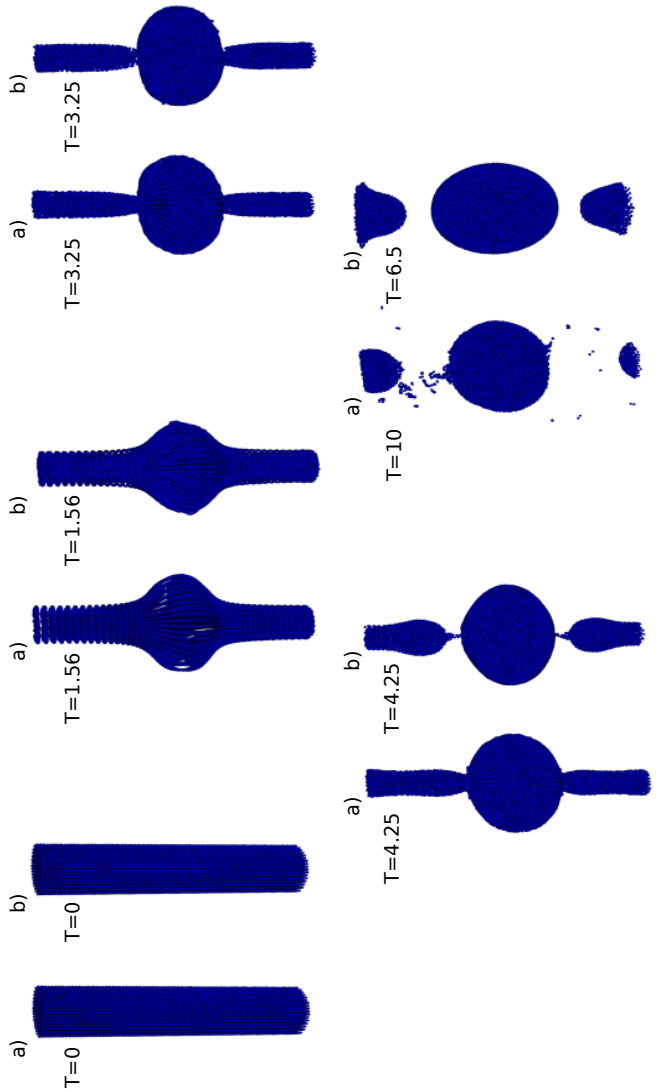


Figure 5: Particle position at different time: a) Density evolution and Shepard filter (n=30) and b) New correction based on curvature.

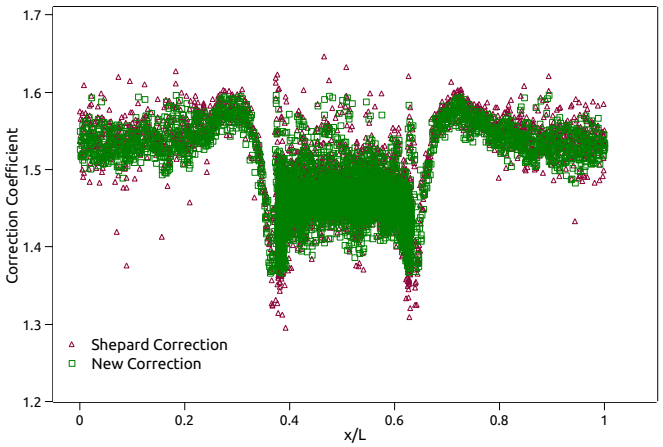


Figure 6: Correction coefficients as a function of the position of the particles along the jet axis

To compare both methods, the SPH simulations have been performed for different resolutions. Table 3 presents the number of flying particles for each case. The result shows that the new method presents stable results even for simulations with a lower resolution.

Table 3: Number of "flying" particles for each method as a function of the system resolution

Method \ Resolution (L/h)	55	66	82
Density evolution + Shepard summation	30 (0.6 %)	96 (1 %)	217 (1.2%)
New method	4 (0.08 %)	1 (0.01 %)	1 (0.005 %)

For a quantitative comparison, the relative disturbance size ($\frac{r_{max}(t)-r_0}{r_0}$) as a function of time is presented in figure 7 for the highest resolution. The results show that the growth dynamics for the two methods agree well with the reference data presented by Dai and Schmidt, even though the break-up time is not the same. Furthermore, the droplet diameter can be estimated

from the volume of the liquid within the disturbance wavelength λ [29].

$$d^3_{droplet} = 6\lambda R^2_{jet} \tag{35}$$

For both cases, the droplet diameter after the break-up agrees well with the theoretical one, with an error around 5%.

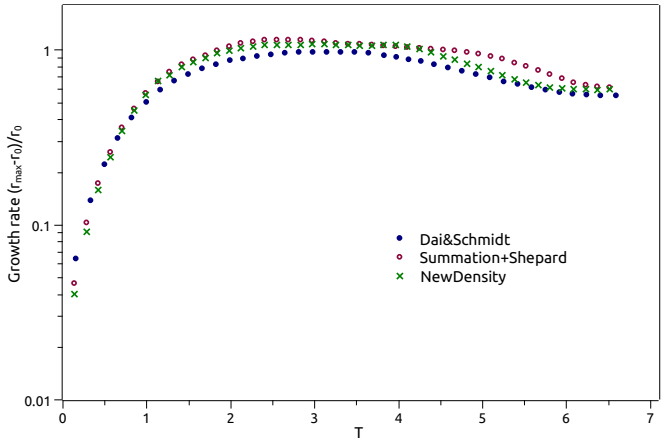


Figure 7: Disturbance growth process in time: Comparing the growth rate as a function of time with the reference data from Dai and Schmidt

This new method proved its efficiency for simulating free surface flows undergoing strong topological variations under the effect of surface tension force. We found the method to work best when the interface is well resolved with smoothed shapes. In fact, the density calculation presented above depends significantly on the curvature and the distance of each particle from the free surface. Thus, these two parameters must be calculated with great precision in order to have the desired output result.

5. Conclusion

This paper presents a new method for estimating the density for free surface simulations dominated by surface-tension force in SPH. The main idea is to correct the density by calculating the missing volume of the support domain of each particle taking into account the local curvature and the distance from the free surface.

This technique was successfully applied to 3D test cases with free-surface flows and surface-tension effects: Square Droplet and Rayleigh-Plateau instability. The results for the square droplet show that the current method can accurately predict the density of free surface particles. Numerical results of the Rayleigh-Plateau instability show that the stability of the break-up of a liquid column is improved with this correction technique. This method can be easily extended to simulate other situations with jet break-up and liquid atomization.

The new correction factors is strongly dependent to an accurate curvature prediction, which still gives room for future improvement especially in terms of robustness.

References

[1] L. B. Lucy, A numerical approach to the testing of the fission hypothesis, The Astronomical Journal 82 (1977) 1013. doi:10.1086/112164.
URL http://adsabs.harvard.edu/cgi-bin/bib_query?1977AJ...82.1013L

[2] R. A. Gingold, J. J. Monaghan, Smoothed particle hydrodynamics: theory and application to non-spherical stars, Monthly Notices of the Royal Astronomical Society 181 (3) (1977) 375–389. doi:10.1093/mnras/181.3.375.
URL https://academic.oup.com/mnras/article-lookup/doi/10.1093/mnras/181.3.375

[3] J. Monaghan, Simulating Free Surface Flows with SPH, Journal of Computational Physics 110 (2) (1994) 399–406. doi:10.1006/jcph.1994.1034.
URL https://linkinghub.elsevier.com/retrieve/pii/S0021999184710345

[4] A. Colagrossi, M. Landrini, Numerical simulation of interfacial flows by smoothed particle hydrodynamics, Journal of Computational Physics 191 (2) (2003) 448–475. doi:10.1016/S0021-9991(03)00324-3.
URL https://linkinghub.elsevier.com/retrieve/pii/S0021999103003243

[5] T. Belytschko, Y. Krongauz, J. Dolbow, C. Gerlach, On the completeness of meshfree particle methods, International Jour-

nal for Numerical Methods in Engineering 43 (5) (1998) 785–819. doi:10.1002/(SICI)1097-0207(19981115)43:5<785::AID-NME420>3.0.CO;2-9.
URL https://onlinelibrary.wiley.com/doi/10.1002/(SICI)1097-0207(19981115)43:5<785::AID-NME420>3.0.CO;2-9

[6] G. A. Dilts, Moving least-squares particle hydrodynamics II: conservation and boundaries, International Journal for Numerical Methods in Engineering 48 (10) (2000) 1503–1524. doi:10.1002/1097-0207(20000810)48:10<1503::AID-NME832>3.0.CO;2-D.
URL http://doi.wiley.com/10.1002/1097-0207%2820000810%2948%3A10%3C1503%3A%3AAID-NME832%3E3.0.CO%3B2-D

[7] J. Bonet, T.-S. Lok, Variational and momentum preservation aspects of Smooth Particle Hydrodynamic formulations, Computer Methods in Applied Mechanics and Engineering 180 (1-2) (1999) 97–115. doi:10.1016/S0045-7825(99)00051-1.
URL https://linkinghub.elsevier.com/retrieve/pii/S0045782599000511

[8] J. Shao, H. Li, G. Liu, M. Liu, An improved SPH method for modeling liquid sloshing dynamics, Computers & Structures 100-101 (2012) 18–26. doi:10.1016/j.compstruc.2012.02.005.
URL https://linkinghub.elsevier.com/retrieve/pii/S0045794912000429

[9] Z. Zhang, M. Liu, Smoothed particle hydrodynamics with kernel gradient correction for modeling high velocity impact in two- and three-

dimensional spaces, Engineering Analysis with Boundary Elements 83 (2017) 141–157. doi:10.1016/j.enganabound.2017.07.015.
URL https://linkinghub.elsevier.com/retrieve/pii/S0955799717301261

[10] D. Molteni, A. Colagrossi, A simple procedure to improve the pressure evaluation in hydrodynamic context using the SPH, Computer Physics Communications 180 (6) (2009) 861–872. doi:10.1016/j.cpc.2008.12.004.
URL https://linkinghub.elsevier.com/retrieve/pii/S0010465508004219

[11] A. Ferrari, M. Dumbser, E. F. Toro, A. Armanini, A new 3D parallel SPH scheme for free surface flows, Computers & Fluids 38 (6) (2009) 1203–1217. doi:10.1016/j.compfluid.2008.11.012.
URL https://linkinghub.elsevier.com/retrieve/pii/S0045793008002284

[12] H.-D. Seo, H.-J. Park, J.-I. Kim, P. S. Lee, The particle-attached element interpolation for density correction in smoothed particle hydrodynamics, Advances in Engineering Software 154 (2021) 102972. doi:10.1016/j.advengsoft.2021.102972.
URL https://linkinghub.elsevier.com/retrieve/pii/S0965997821000016

[13] M. Gomez-Gesteira, B. D. Rogers, R. A. Dalrymple, A. J. Crespo, State-of-the-art of classical SPH for free-surface flows, Journal of Hydraulic Research 48 (sup1) (2010) 6–27. doi:10.1080/00221686.2010.

9641242.
URL https://www.tandfonline.com/doi/full/10.1080/00221686.2010.9641242

[14] D. Violeau, B. D. Rogers, Smoothed particle hydrodynamics (SPH) for free-surface flows: past, present and future, Journal of Hydraulic Research 54 (1) (2016) 1–26. doi:10.1080/00221686.2015.1119209.
URL https://www.tandfonline.com/doi/full/10.1080/00221686.2015.1119209

[15] J. Calderon-Sanchez, J. Cercos-Pita, D. Duque, A geometric formulation of the Shepard renormalization factor, Computers & Fluids 183 (2019) 16–27. doi:10.1016/j.compfluid.2019.02.020.
URL https://linkinghub.elsevier.com/retrieve/pii/S0045793019300416

[16] W. Kostorz, A. Esmail-Yakas, A semi-analytical boundary integral method for radial functions with application to Smoothed Particle Hydrodynamics, Journal of Computational Physics 417 (2020) 109565. doi:10.1016/j.jcp.2020.109565.
URL https://linkinghub.elsevier.com/retrieve/pii/S0021999120303399

[17] M. Herant, Dirty tricks for sph, Memorie della Societa Astronomica Italiana 65 (1994) 1013.

[18] H. A. T. Vanhala, A. G. W. Cameron, Numerical Simulations of Triggered Star Formation. I. Collapse of Dense Molecular Cloud Cores, The

Astrophysical Journal 508 (1) (1998) 291–307. doi:10.1086/306396.
URL https://iopscience.iop.org/article/10.1086/306396

[19] J. P. Morris, P. J. Fox, Y. Zhu, Modeling Low Reynolds Number Incompressible Flows Using SPH, Journal of Computational Physics 136 (1) (1997) 214–226. doi:10.1006/jcph.1997.5776.
URL https://linkinghub.elsevier.com/retrieve/pii/S0021999197957764

[20] F. V. Sirotkin, J. J. Yoh, A new particle method for simulating breakup of liquid jets, Journal of Computational Physics 231 (4) (2012) 1650–1674. doi:10.1016/j.jcp.2011.10.020.
URL https://linkinghub.elsevier.com/retrieve/pii/S0021999111006206

[21] J. J. Monaghan, Smoothed particle hydrodynamics, Reports on Progress in Physics 68 (8) (2005) 1703–1759. doi:10.1088/0034-4885/68/8/R01.
URL http://stacks.iop.org/0034-4885/68/i=8/a=R01?key=crossref.c562820df517a049ca7f11d0aefe49b4

[22] S. Adami, X. Hu, N. 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.
URL https://linkinghub.elsevier.com/retrieve/pii/S0021999110001324

[23] J. Brackbill, D. Kothe, C. Zemach, A continuum method for modeling surface tension, Journal of Computational Physics 100 (2) (1992) 335–354. doi:10.1016/0021-9991(92)90240-Y.
URL https://linkinghub.elsevier.com/retrieve/pii/002199919290240Y

[24] 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.
URL http://doi.wiley.com/10.1002/1097-0363%2820000615%2933%3A3%3C333%3A%3AAID-FLD11%3E3.0.CO%3B2-7

[25] A. Barecasco, H. Terissa, C. F. Naa, Simple free-surface detection in two and three-dimensional SPH solver, arXiv:1309.4290 [physics]ArXiv: 1309.4290 (Sep. 2013).
URL http://arxiv.org/abs/1309.4290

[26] J.-P. Fürstenau, C. Weißenfels, P. Wriggers, Free surface tension in incompressible Smoothed Particle Hydrodynamcis (ISPH), Computational Mechanics 65 (2) (2020) 487–502. doi:10.1007/s00466-019-01780-6.
URL http://link.springer.com/10.1007/s00466-019-01780-6

[27] M. Dai, D. P. Schmidt, Adaptive tetrahedral meshing in free-surface flow, Journal of Computational Physics 208 (1) (2005) 228–252. doi:10.1016/j.jcp.2005.02.012.

URL https://linkinghub.elsevier.com/retrieve/pii/S0021999105001075

[28] M. Olejnik, K. Szewc, Smoothed particle hydrodynamics modelling of the Rayleigh-Plateau instability, Journal of Theoretical and Applied Mechanics (2018) 675doi:10.15632/jtam-pl.56.3.675.
URL http://ptmts.org.pl/jtam/index.php/jtam/article/view/4306

[29] N. Ashgriz (Ed.), Handbook of Atomization and Sprays, Springer US, Boston, MA, 2011. doi:10.1007/978-1-4419-7264-4.
URL http://link.springer.com/10.1007/978-1-4419-7264-4

A novel density calculation method is developed for free surface flows in SPH

A geometric based coefficient is proposed for correcting the density near the free surface

The present method is found to increase the stability for surface tension driven simulations