Eurographics/ ACM SIGGRAPH Symposium on Computer Animation (2007)
D. Metaxas and J. Popovic (Editors)

Real-time Simulations of Bubbles and Foam within a Shallow Water Framework

N. Thürey[2], F. Sadlo[2], S. Schirm[1], M. Müller-Fischer[1] and M. Gross[2]

[1]AGEIA
[2]ETH Zurich, Switzerland

Abstract

Bubbles and foam are important fluid phenomena on scales that we encounter in our lives every day. While different techniques to handle these effects were developed in the past years, they require a full 3D fluid solver with free surfaces and surface tension. We present a shallow water based particle model that is coupled with a smoothed particle hydrodynamics simulation to demonstrate that real-time simulations of bubble and foam effects are possible with high frame rates. A shallow water simulation is used to represent the overall water volume. It is coupled to a particle-based bubble simulation with a flow field of spherical vortices. This bubble simulation is interacting with a smoothed particle hydrodynamics simulation including surface tension to handle foam on the fluid surface. The realism and performance of our approach is demonstrated with several test cases that run with high frame rates on a standard PC.

Categories and Subject Descriptors (according to ACM CCS): I.3.7 [Three-Dimensional Graphics and Realism]: Animation

1. Introduction

Secondary fluid effects such as bubbles are prominent in every day situations such as cooking or filling a cup of coffee. They are, however, more difficult to capture than the overall fluid motion itself, as they obey complex laws governed by surface tension, the behavior of the gas phase, and the properties of very thin fluid sheets. The goal of this paper is to capture these three-dimensional effects within the framework of an essentially two-dimensional shallow water (SW) simulation. We will demonstrate that the complexity of the mechanisms that lead to bubble dynamics and foam formation can be reduced significantly to make interactive simulations with high frame rates possible. As physically based simulations are an important part of modern computer games, our approach makes it possible to include fluids with bubbles and foam into this area. Moreover, we will demonstrate that the algorithm can be used to efficiently simulate these effects for off-line renderings as well.

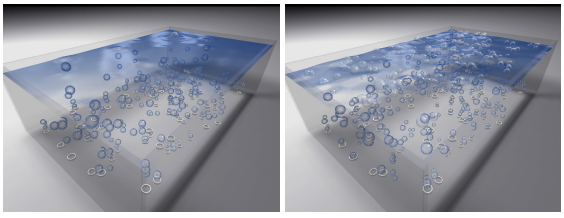

Figure 1: *Two images from a bubble and foam simulation generated with our approach.*

The contributions of this paper are:

- a shallow water based bubble simulation model that introduces spherical vortices to efficiently generate a three-dimensional flow field,
- and a foam simulation based on smoothed particle hydrodynamics (SPH) with surface tension to capture the formation and clustering of foam bubbles.

2. Related Work

The topic of fluid simulations was first brought up in [KM90], where the wave equation was solved to gen-

Symposium on Computer Animation 2007, San Diego, CA, August 03-04, 2007.
© 2007 ACM 978-1-59593-624-4/07/0008 $5.00

erate a height field fluid simulation. Since then, two-dimensional methods have been used, e.g., for generating splashes [OH95] or coupled to terrains and rigid bodies [CdVLHM97]. A different class of methods for wave generation are the spectral approaches, e.g., as described in [HNC02], [Lov03], or in [Tes04]. These approaches are especially suitable for deep water waves with dispersive effects. In the following, we will use a shallow water discretization, as it can be computed efficiently and the equations are conservative. Moreover, wave dispersion is not crucial for the scale of effects we are targeting.

Solving the full three-dimensional equations for fluid flow, the Navier-Stokes (NS) equations, was made popular by Stam [Sta99] together with Foster and Fedkiw [FF01]. As these simulations often require huge amounts of computation, they have recently been combined with two-dimensional techniques to reduce the required computational cost [IGLF06, TRS06]. However, these approaches are not targeted towards real-time applications, but allow the simulation of more detailed fluid effects. In [GH04] the authors generate bubbles in the flow field of such a solver for enhanced realism, while Mihalef et. al present a full two-phase simulation for bubbly flows in [MUM*06]. The handling of thin fluid sheets for foam like structures was addressed in [ZYP06]. Another approach that dealt with rendering and a physically based model for foam simulation was presented in [KVG02]. Moreover, the behavior of bubbles is an important topic in mechanical engineering [CGW78], e.g., to better understand and optimize bubble flow reactors [BDR05]. In the following, we will use a SW simulation for the main fluid volume, as it can efficiently handle the wave propagation on the fluid surface and guarantees the overall conservation of fluid mass. The bubble dynamics will be simplified and reduced to the vortices induced in the flow, a buoyancy force, and effects such as merging.

The vortex filament methods, e.g., as presented in [AN05] and [ANSN06], are based on the vorticity formulation of the NS equations. They have the advantage of defining a vorticity preserving flow field around a set of vortex primitives. We will present a similar method, that, however, directly evaluates an analytical and Navier-Stokes conforming flow field around spherical vortices. It can be evaluated efficiently and has the advantage of yielding a flow field around the spherical vortex region that also correctly captures the fluid dynamics around it.

An alternative approach for solving the NS equations that does not require a computational grid are smoothed particle hydrodynamics (SPH) [Mon92]. As was demonstrated in [MCG03] and [MSRG05] they can be used to capture a variety of fluid effects, and are interesting for real time environments due to their particle based nature. In [MSRG05] the authors also perform a boiling simulation, but in contrast to our approach, SPH is used to fully simulate both the gas and the liquid phase. This would require huge amounts of particles for scenes such as Figure 1. Furthermore, our approach is aimed at fluids with lower viscosities. We make use of SPH to model the foam structures on the SW surface. In addition, surface tension is used to cause a clustering of the foam bubbles.

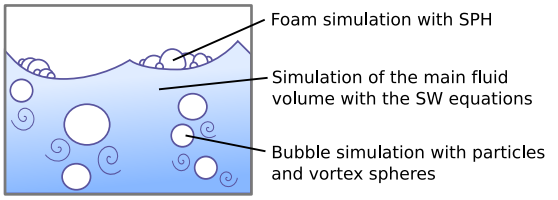

Figure 2: *An overview of the three different coupled simulation techniques.*

3. The Shallow Water Equations

Shallow water (SW) simulations reduce the complexity of the full Navier-Stokes equations to the simplified two-dimensional representation of a height field. Note that, in the following, it will be assumed that the gravity force is acting along the z axis. Thus, the plane for the 2D simulation corresponds to the x-y plane. One simplifying assumption of the SW equations is that the velocity does not vary significantly along the z axis, and that there is a constant pressure gradient from the water surface to the bottom. The only two forces driving the flow are pressure and gravity. It is furthermore assumed that the liquid under consideration has a very low viscosity. Thus, the viscosity term of the NS equations can be neglected.

In the following, $H(\mathbf{x},t)$ is the height of the water above ground, while $\mathbf{v}$ is the horizontal velocity of the fluid, and $\mathbf{g} = (0,0,g)^T$ is the gravitational force perpendicular to the 2D simulation plane. The simulation region consists of N_x grid nodes with size Δx, and N_y grid nodes with size Δy in x and y direction, respectively. The simplified shallow water equations can now be written as

$$\frac{\partial H}{\partial t} = -\mathbf{v}\cdot\nabla H - H\left(\frac{\partial v_x}{\partial x} + \frac{\partial v_y}{\partial y}\right) \quad (1)$$

$$\frac{\partial v_x}{\partial t} = -\mathbf{v}\cdot\nabla v_x - g\frac{\partial H}{\partial x}\ , \text{ and} \quad (2)$$

$$\frac{\partial v_y}{\partial t} = -\mathbf{v}\cdot\nabla v_y - g\frac{\partial H}{\partial y}\ . \quad (3)$$

We use a staggered grid together with a semi-Lagrangian advection step to solve these equations [Sta99]. A good overview of such a SW solver can be found, e.g., in [Bri05]. To simplify the equations, we do not take a changing ground topology into account. Our approach, however, also works for variable ground heights.

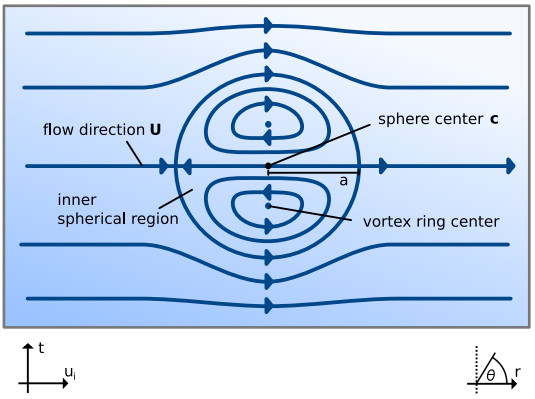

Figure 3: *Streamlines for a spherical Hill vortex.*

4. Simulating Bubbles

The actual flow around bubbles in a liquid with a very low viscosity, such as water, is usually very turbulent and hard to compute. We make a number of simplifications to capture this phenomenon within a real-time simulation. As described above, the main volume of the liquid is simulated with the SW equations of Section 3. The bubbles are simulated as particles that interact with the shallow water surface, and among each other. Here we assume a flow and shape regime (according to [CGW78]) of relatively low bubble velocities, and strong surface tension, as present in water. Therefore, each bubble is treated as a spherical particle, and has the following properties: a position in space $\mathbf{p}_i = (p_{i_x}, p_{i_y}, p_{i_z})$, a velocity $\mathbf{u}_i$, a radius r_i and a volume (or mass) m_i. While the bubble is rising, we update the position with Euler steps. The simulation of foam on the water surface is handled by an SPH simulation, as will be described in Section 5. An overview of the combination of these simulation techniques is shown in Figure 2.

4.1. Spherical vortices

For the flow of the fluid around the bubbles, we make use of the concept of Hill's spherical vortex [Hil94]. Such a vortex consists of a flow field that describes the irrotational flow past a sphere, and a closed-streamline vortex inside of this sphere [Ach90]. In contrast to a toroidal vortex with a small core cross section, such as a smoke ring, this vortex is distinguished by the confined vorticity within its spherical region, and the absence of a hole in its middle. The streamlines of a Hill's vortex are shown in Figure 3. Such vortices were experimentally detected around bubbles, an example can be found in, e.g., [Bat67]. This is only an approximation of the flow around a bubble in general, but we will in the following use it to model the flow around the bubbles in our simulation.

Hill's spherical vortex is usually given as a stream function in polar coordinates (r, θ). This stream function Ψ_i is given by:

$$\Psi_i(r,\theta,a) = \begin{cases} \frac{1}{2}\left(r^2 - \frac{a^3}{r}\right) sin^2\theta & \text{for } r \geq a \\ -\frac{3}{4} r^2 \left(1 - \frac{r^2}{a^2}\right) sin^2\theta & \text{for } r < a\ , \end{cases} \tag{4}$$

where a is the radius of the spherical vortex. The velocity field along the isolines of this function could be computed directly with spatial derivatives, but in the following we will use the analytical solution, which is, according to [Saf94]

$$\mathbf{u}_h(\mathbf{x},d) = \begin{cases} \begin{pmatrix} xz \\ yz \\ z^2 + 1 - 2d^2 \end{pmatrix} & \text{for } d < 1, \\ \begin{pmatrix} xz\, r^{-5} \\ yz\, r^{-5} \\ z^2 r^{-5} - 1/3 r^{-3} - 3/2 \end{pmatrix} & \text{for } d \geq 1\ . \end{cases} \tag{5}$$

This equation yields a divergence-free velocity field that, together with an appropriate pressure field, solves the NS equations, and has a zero vorticity outside the spherical region. This analytical solution of the flow field allows an efficient computation of bubble interactions.

The center $\mathbf{c}_i$ of the spherical vortex is computed with the bubble center $\mathbf{p}_i$ as $\mathbf{c}_i = \mathbf{p}_i - (a_i - r_i/2)\mathbf{u}_i/|\mathbf{u}_i|$, where a_i is again radius of the spherical region. Thus, the bubble is placed at the front of the inner spherical region of the vortex. In the following, we choose $a_i = 3r_i$. This parameter can be used to control the extent of the vortices. For $a_i = r_i$, only the flow around the bubble will remain, while larger values will result in trailing vortices behind the bubbles. An example of the flow around a bubble due to a spherical vortex can be found in the accompanying animation.

In the following, $\mathbf{d}_{ij} = \mathbf{p}_j - \mathbf{c}_i$ will denote the distance between the position of bubble j and the center of the spherical vortex around bubble i. $\mathbf{d}_{ij}$ is transformed into the local coordinate system of the spherical vortex (of bubble i), so that the z axis aligns with the bubble velocity $\mathbf{u}_i$. The velocity influence for a bubble j is given by the sum of vorticity forces from all other bubbles i, multiplied by a fall off function f_v, and the velocity of the shallow water simulation $\mathbf{v}$ as:

$$\mathbf{f}_j^v = \left[\mathbf{v}(\mathbf{p}_j) + \sum_i \mathbf{u}_h\left(\frac{\mathbf{d}_{ij}}{a_i}, \frac{|\mathbf{d}_{ij}|}{a_i}\right) f_v(|\mathbf{d}_{ij}|, a_i)\right] \frac{\Delta t}{m_j}\ . \tag{6}$$

The SW velocity in the x-y plane is directly given by the simulation, while the z component can be computed from the heights of the last two time steps as $\mathbf{v}_z = h(t) - h(t - \Delta t)$. For velocity values inside of the fluid volume, we exponentially scale down the velocity values [Ger04]. Once a bubble breaks through the surface and is removed from the simulation, the vortex behind it would dissipate in the fluid over time. This effect is accounted for by letting the vortex sphere of a deleted bubble fade out at its last position, which gives the impression of fluid inertia.

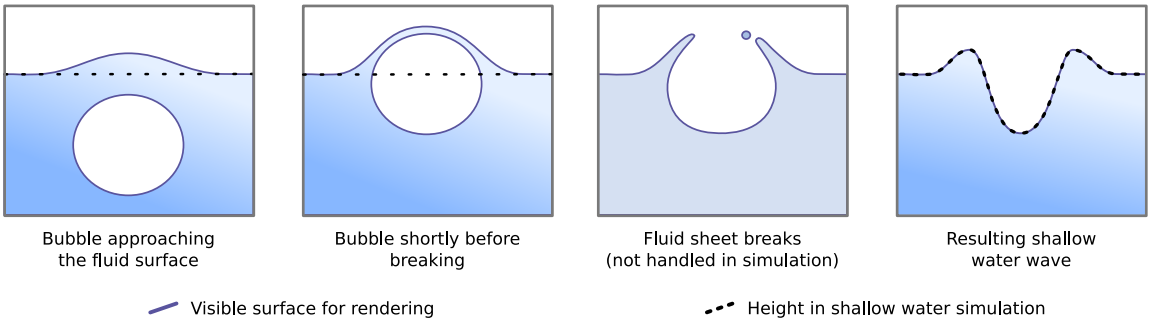

Figure 4: *A bubble approaching the fluid surface and breaking. To the right, the wave initialized for the shallow water simulation is shown.*

For f_v, we use a simple linear fall off function and restrict the force to act within a finite radius of $3a$:

$$f_v(r,a) = \begin{cases} 1 & \text{for } r < 2a \\ 3 - r/a & \text{for } 2a \leq r < 3a \\ 0 & \text{for } r \geq 3a \ . \end{cases} \tag{7}$$

Hence, the correct Navier-Stokes solution of Equation (5) is retained within a radius of $2a$. Further away, we blend the already small values of $\mathbf{u}_h$ to zero.

4.2. Shallow Water Coupling

The coupling of the bubbles and the shallow water simulation has no effect on the latter before the bubbles break through the water surface. While a bubble is fully submerged, the flow around it is smooth enough to only lead to a displacement of the fluid surface, due to our assumptions of the flow properties. This displacement is computed using the normalized spline kernel $W(h,d)$, with width h that is evaluated at distance d. According to [MKN*04], it is computed as:

$$W(h,d) = \begin{cases} \frac{315(h-d)^3}{64\pi h^9} \text{ for } d \leq h \\ 0 \text{ otherwise} \ . \end{cases} \tag{8}$$

The displacement itself $d_i(\mathbf{x})$, which depends on the distance of the bubble to the fluid surface is then computed with

$$\begin{aligned} d_i(\mathbf{x}) &= \frac{m_i}{1 + (H(\mathbf{x}) - p_{i_z})} W(h, |\mathbf{p}_i - \mathbf{x}|) \ , \text{ with} \\ h &= 2\, r_i \Big(1 + H(\mathbf{x}) - p_{i_z} \Big) \ . \end{aligned} \tag{9}$$

Note that d_i is only used as an offset for the rendering of the water surface, but does not influence the SW simulation itself. However, the integral of d_i is equal to the volume of the bubble (for $H(\mathbf{x}) = p_{i_z}$), hence it correctly represents the volume of the gas inside of it.

Once the bubble breaks through the surface, meaning that $(H(\mathbf{x}) + d_i(\mathbf{x}) - p_{i_z}) < 0$, we modify the shallow water surface to represent a circular wave around the bubble. We thus add

$$d_i(\mathbf{x})' = W(h, |\mathbf{p}_i - \mathbf{x}|) - W(h/2, |\mathbf{p}_i - \mathbf{x}|) \tag{10}$$

to the fluid heights $H(\mathbf{x})$ of the SW simulation, as shown in Figure 4. This results in a surface wave within the shallow water simulation according to the size of bubble. Note that Equation (10) preserves the mass of the SW simulation. Thus, it corresponds to a case where the gas of the bubbles is, e.g., streamed into the fluid with a nozzle below it. To account for a loss of fluid mass due to a phase change, e.g., when heating the liquid, the mass of the SW simulation could be explicitly and uniformly reduced whenever a bubble is created. However, we will in the following keep the overall fluid mass constant.

For the integration of the bubble velocities, we compute the forces acting upon the bubble in the fluid. In our model, we only incorporate buoyancy and velocity forces:

$$\mathbf{u}_i(t + \Delta t) = \mathbf{u}_i(t) + \Delta t (\mathbf{f}_i^b + \mathbf{f}_i^v)/m_i \ . \tag{11}$$

The buoyancy force is given by the volume of the fluid displaced by the bubble and the gravity $\mathbf{g}$ as $\mathbf{f}_i^b = -\mathbf{g}\gamma_b m_i$. Here γ_b is a parameter to set the density difference between the gas and the fluid, for simplicity, we use $\gamma_b = 1$. For more viscous fluids, a drag force counteracting the current bubble velocity could also be added at this point.

4.3. Coalescence and Surface Animation

The algorithm described so far can already produce realistic motions of interacting bubbles. However, another important effect is the coalescence of bubbles. If the distance between two bubbles i and j is less than the sum of their radii, they are removed from the simulation, and a new bubble with a volume m_n that is equal to the sum of the two touching bubbles $(m_i + m_j)$ is introduced. The other properties of the bubble are now interpolated according to this volume. The velocity,

for example, is initialized as

$$\mathbf{u}_m = \frac{\mathbf{u}_i m_i}{m_n} + \frac{\mathbf{u}_i m_j}{m_n} \; . \tag{12}$$

The radius of the new bubble on the other hand is computed from its volume with $r_n = 3/4\, m_n^{1/3}$. This also leads to a corresponding change of the radius of the bubble's spherical vortex.

Although a spherical bubble shape is assumed for the simulation, the bubble surfaces are animated for rendering. As a bubble also represents a free surface, it likewise exhibits wave propagation once the surface is perturbed. This is important, e.g., when two bubbles merge. To achieve this effect, a sinusoidal offset is added along the direction of the bubble movement, with a frequency given by the bubble velocity and size. We make the assumption that larger and faster bubbles experience stronger perturbations from the surrounding fluid. When two bubbles merge, the frequency and amplitude of this displacement is also temporarily increased.

4.4. Limitations

As will be shown in Section 6, the approach described so far yields realistic simulations of bubbly flows, yet, it has some inherent limitations. Obviously, due to the coupling between velocity forces and bubble positions, inertia effects of the fluid cannot be really handled. Hence, interesting flow structures will only be visible around bubbles, and as long as bubbles are in the fluid. Without bubbles, only the SW simulation is left for the fluid volume, which cannot capture any three-dimensional flow effects. Moreover, the resulting velocity field is not divergence free. However, this should not be a problem for targeted applications such as computer games, as the overall mass conservation of the fluid volume is still guaranteed by our approach.

5. Simulating Foam

Foam structures on a fluid surface are caused by bubbles that exhibit delayed breaking due to surface tension effects. The surface tension also leads to the typical clustering of the foam bubbles among each other, and to hydrophilic walls. For the simulation of such a foam we use an SPH simulation with a surface tension algorithm and forces applied from the shallow water simulation.

5.1. Smoothed Particle Hydrodynamics

SPH algorithms numerically solve the Navier-Stokes equations with a Lagrangian approach. The volume of the fluid is represented by particles, and a spherical kernel function is used to evaluate macroscopic fluid quantities such as pressure and velocity. In the following, we use the SPH algorithm as described in [MKN*04]. Thus, a fluid property A is

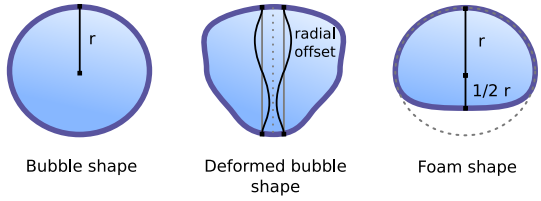

Figure 5: *The shape of a bubble beneath the surface to the left, and a foam bubble to the right.*

evaluated at a point $\mathbf{x}$ in space from the per particle properties A_i with

$$A(x) = \sum_i A_i v_i W(\mathbf{x} - \mathbf{x}_i, h) \; , \tag{13}$$

where the particle volume v_i is given by the initial sampling. These interpolations are used to compute pressure and viscosity forces acting upon the particles.

5.2. Foam Model

For the foam simulation, we do not represent the actual fluid in the thin sheets of a foam with the SPH simulation, but the overall volume of the foam. Each bubble in the foam is represented with an SPH particle. The surface tension in the SPH simulation, which acts on the scale of the SPH particles, is used to model the agglomeration of bubbles in a foam.

The particle size s_i of the SPH simulation is initialized with the average size of the bubbles that are generated within the bubble simulation of Section 4. We generate foam particles with a user defined probability p_f from bubbles. Instead of performing the wave initialization with Equation (10) for a bubble deletion, we initialize a foam particle at the position of the deleted bubble. Although the size of the SPH particles is fixed to s_i, each particles stores a virtual size r_i that is inherited from the original bubble. This size value is used for rendering of the foam, and generating the surface forces. For a typical water based foam, the foam bubbles on the surface are not spherical, but have a more hemispherical shape, as shown in Figure 5. Once a foam bubble is deleted (after a randomized life time), a surface wave is initialized as for a breaking bubble with Equation (10).

The typical clustering of foam bubbles is achieved by adding surface tension to the SPH simulation. The surface tension within SPH is computed by evaluating a color field function c_s on the particles to generate normals, as described in [MKN*04]. The divergence of these normals is used to compute the curvature and the resulting strength of the surface tension force:

$$\mathbf{f}_t = -\gamma_{st} \nabla^2 c_s \frac{\mathbf{n}}{|\mathbf{n}|} \; . \tag{14}$$

Here γ_{st} is the parameter to control the strength of the applied force. A detailed description of this approach for com-

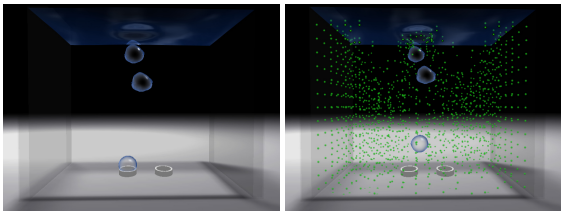

Figure 6: *Vortex interactions between two bubbles. The picture to the right shows the flow movement around the bubbles with marker particles.*

Table 1: *Simulation settings (SW simulation resolution, maximal number of bubble particles, maximal number of foam SPH particles) and performance measurements (lowest frame rate) for the different test cases.*

	SWS	Bubbles	Foam	FPS
Figure 7	40x66	81	133	161.2
Figure 8	50x50	142	131	144.1
Figure 11	50x80	501	466	34.3
Figure 12	50x80	654	1129	18.9

puting surface tension forces within an SPH simulation can be found in [MKN*04].

To make the foam float and spread out on the SW surface, we add forces along the gravity direction to each particle during the SPH update step. The force $\mathbf{f}_s$ for a SPH particle at position $\mathbf{x}_i$ is calculated to move the particle to the z-position of the SW height field during the next SPH update step:

$$\mathbf{f}_s = \frac{\mathbf{g}}{|\mathbf{g}|}\left(H(\mathbf{x}) - (\mathbf{x}_z + \frac{r_i}{2})\right)/\Delta t \qquad (15)$$

Note that these forces are not a hard constraint, but together with the SPH forces allow the particles to move into an equilibrium state between SW surface, SPH and surface tension forces.

6. Results

The following simulations were run on a standard PC with a Core 2 Duo CPU (2.4 GHz) and a Geforce 7900 GPU (with a single thread). Details of the different resolutions and settings can be found in Table 1. The frame rate measurements, which are minimum over the course of the simulation, include the rendering, which took on average $1/3$ of the overall computation time.

A simple test case to demonstrate the interactions of two bubbles due to the spherical vortices can be seen in Figure 6. Varying the flow rate of the nozzles at the bottom can lead to very different flow behavior - from orderly rising bubbles, to chaotic interactions. A larger test case, with more than one hundred bubbles and more than thousand foam particles can be seen in Figure 7. Such a simulation runs with more than 18.9 frames per second, and exhibits a high amount of detail and interesting flow effects. In this case the computations are clearly dominated by the SPH simulation due to the high number of foam particles.

A different test case is shown in Figure 8, where a user controlled heated source is moved at the bottom of a basin of fluid, with a minimal frame rate of 144. Depending on the speed of the movement, the bubbles either quickly merge to larger ones, or rise as clusters of many small ones. Several other examples and variations of these test cases can be found in the accompanying animation.

7. Conclusions

We have presented an algorithm to perform simulations of bubbles and foam in real-time. The high performance of our approach is achieved by coupling a shallow water simulation to particle based bubble and foam simulations. A flow field with vortices is efficiently generated around the bubbles by the use of spherical vortices. Per particle forces from the SW simulation lead to the formation and clustering of foam within the SPH simulation. This makes complex scenes with several hundred bubble and foam particles possible, which still run with high frame rates on a standard PC.

As future work, we would like to enhance the shaders for rendering the foam. These could also compute the intersections of the foam bubbles, and generate fluid sheets between them, as suggested in [KVG02]. Another extension would be to handle thick layers of foam, by applying the forces from Equation (15) only to a layer of particles near the SW surface. Moreover, it would be possible to create a force field around hydrophilic objects in the fluid, to make the foam particles cluster around it.

The animation of the bubbles could be enhanced by allowing for stronger deformations of the spherical shapes to animate large or merging bubbles. Moreover, this could be used to select different shape regimes from [CGW78] for the simulation, e.g., leading to cap-shaped instead of spherical bubbles.

8. Acknowledgements

Thanks to AGEIA for funding this research. Moreover, thanks to Anders Stenberg, Hermes Lanker and Philipp Hatt for help with the implementation, as well as to Ronny Peikert and Mark Pauly for helpful comments.

References

[Ach90] ACHESON D. J.: *Elementary Fluid Dynamics*. Oxford University Press, 1990.

[AN05] ANGELIDIS A., NEYRET F.: Simulation of smoke based on vortex filament primitives. *SCA '05: Proceedings of the 2005 ACM SIGGRAPH/Eurographics symposium on Computer animation* (2005), 87–96.

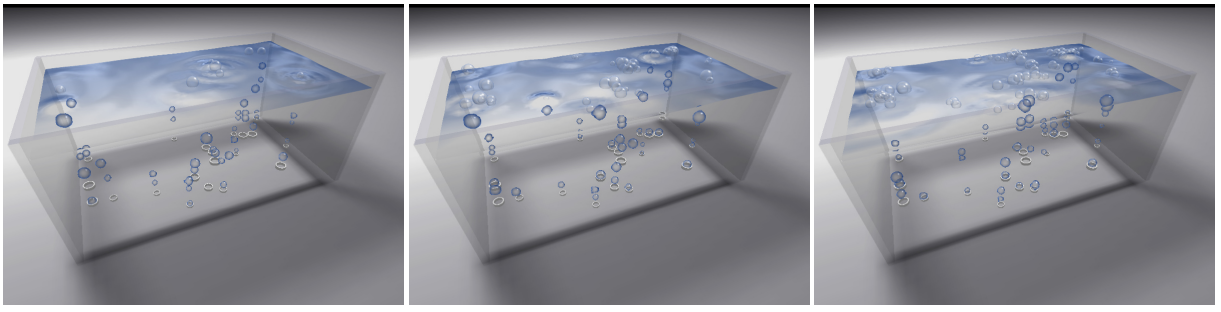

Figure 7: *Several images from a larger bubble simulation.*

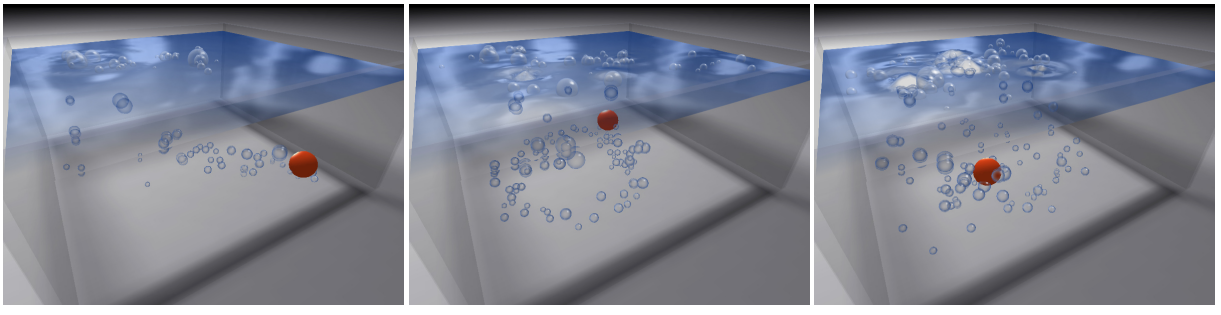

Figure 8: *A user interactively controls the heated source at the bottom of the basin to generate bubbles.*

[ANSN06] ANGELIDIS A., NEYRET F., SINGH K., NOWROUZEZAHRAI D.: A controllable, fast and stable basis for vortex based smoke simulation. *SCA '06: Proceedings of the 2006 ACM SIGGRAPH/Eurographics symposium on Computer animation* (2006), 25–32.

[Bat67] BATCHELOR G. K.: *An Introduction to Fluid Dynamics*. Cambridge Mathematical Library, 1967.

[BDR05] BUWA V. V., DEO D. S., RANADE V. V.: Eulerian-Lagrangian Simulations of Unsteady Gas-Liquid Flows in Bubble Columns. *Int. J. Multiphase Flow* (2005).

[Bri05] BRIDSON R.: Shallow water discretization, Lecture notes Animation Physics, 2005.

[CdVLHM97] CHEN J. X., DA VITORIA LOBO N., HUGHES C. E., MOSHELL J. M.: Real-time fluid simulation in a dynamic virtual environment. *IEEE Comput. Graph. Appl.* (1997).

[CGW78] CLIFT R., GRACE J. R., WEBER M. E.: *Bubbles, Drops and Particles*. Academic Press, London, 1978.

[FF01] FOSTER N., FEDKIW R.: Practical animation of liquids. In *Proc. of ACM SIGGRPAH* (2001), pp. 23–30.

[Ger04] GERSTNER F. V.: *Theory of waves*. Abhandlungen der Königlichen böhmischen Gesellschaft der Wissenschaften zu Prag, 1804.

[GH04] GREENWOOD S. T., HOUSE D. H.: Better with bubbles: enhancing the visual realism of simulated fluid. *SCA '04: Proceedings of the 2004 ACM SIGGRAPH/Eurographics symposium on Computer animation* (2004), 287–296.

[Hil94] HILL M. J. M.: On a spherical vortex. *Philos. Trans. Roy. Soc. London A 185* (1894), 213–245.

[HNC02] HINSINGER D., NEYRET F., CANI M.-P.: Interactive Animation of Ocean Waves. *Proc. of the 2002 ACM SIGGRAPH/Eurographics Symposium on Computer animation* (July 2002).

[IGLF06] IRVING G., GUENDELMAN E., LOSASSO F., FEDKIW R.: Efficient Simulation of Large Bodies of Water by Coupling Two and Three Dimensional Techniques. *ACM Trans. Graph. 25* (2006).

[KM90] KASS M., MILLER G.: Rapid, Stable Fluid Dynamics for Computer Graphics. *ACM Trans. Graph. 24*, 4 (1990), 49–55.

[KTH*05] KÖRNER C., THIES M., HOFMANN T., THÜREY N., RÜDE U.: Lattice Boltzmann Model for Free Surface Flow for Modeling Foaming. *Journal of Statistical Physics 121 [1-2]* (October 2005), 179–196.

[KVG02] KÜCK H., VOGELGSANG C., GREINER G.: Simulation and rendering of liquid foams. *Proc. Graphics Interface '02* (2002), 81–88.

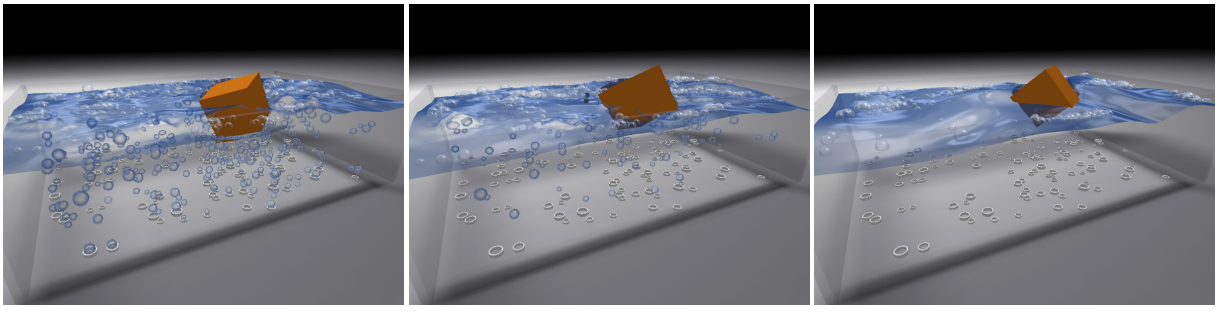

Figure 9: *A user interactively moves the obstacle, pushing away the foam and bubbles.*

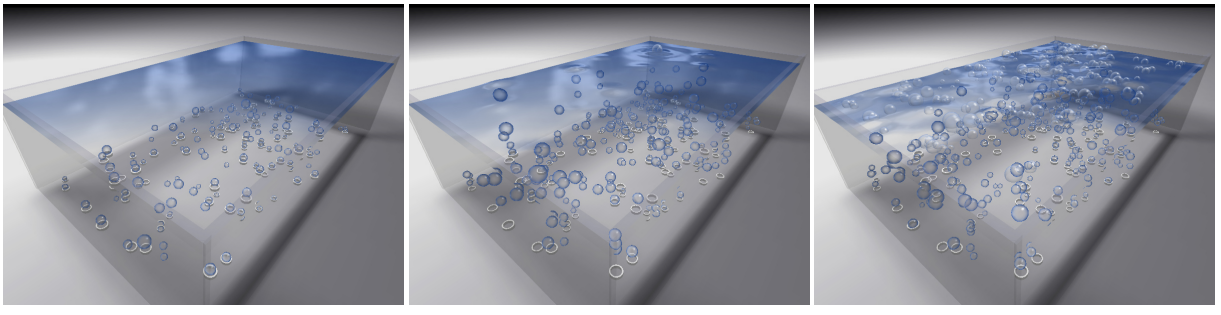

Figure 10: *Two images from a bubble and foam simulation generated with our approach. This simulation makes use of more than 650 interacting bubble particles, and more than 1100 foam SPH particles.*

[Lov03] LOVISCACH J.: Complex Water Effects at Interactive Frame Rates. *Journal of WSCG 11* (2003), 298–305.

[MCG03] MÜLLER M., CHARYPAR D., GROSS M.: Particle-based fluid simulation for interactive applications. *Proc. of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer animation* (2003), 154–159.

[MKN*04] MÜLLER M., KEISER R., NEALEN A., PAULY M., GROSS M., ALEXA M.: Point based animation of elastic, plastic and melting objects. In *Proc. of the 2004 ACM SIGGRAPH/EUROGRAPHICS Symposium on Computer Animation* (Aug 2004).

[Mon92] MONAGHAN J.: Smoothed particle hydrodynamics. *Annu. Rev. Astron. Phys. 30* (1992), 543–574.

[MSRG05] MÜLLER M., SOLENTHALER B., RICHARD, GROSS M.: Particle-based fluid-fluid interaction. *Proc. of the 2005 ACM Siggraph/Eurographics Symposium on Computer Animation* (2005).

[MUM*06] MIHALEF V., UNLUSU B., METAXAS D., SUSSMAN M., HUSSAINI M. Y.: Physics based boiling simulation. *Proceedings of the 2006 ACM SIGGRAPH/Eurographics Symposium on Computer animation* (2006), 317–324.

[OH95] O'BRIEN J. F., HODGINS J. K.: Dynamic simulation of splashing fluids. p. 198.

[Saf94] SAFFMAN P. G.: *Vortex Dynamics*. Cambridge Monographs on Mechanics, 1994.

[Sta99] STAM J.: Stable Fluids. *Proc. of ACM SIGGRAPH* (1999), 121–128.

[Tes04] TESSENDORF J.: Simulating Ocean Surfaces. *SIGGRAPH 2004 Course Notes 31* (2004).

[TRS06] THÜREY N., RÜDE U., STAMMINGER M.: Animation of Open Water Phenomena with coupled Shallow Water and Free Surface Simulations. *Proc. of the 2006 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2006).

[ZYP06] ZHENG W., YONG J.-H., PAUL J.-C.: Simulation of bubbles. *SCA '06: Proceedings of the 2006 ACM SIGGRAPH/Eurographics symposium on Computer animation* (2006), 325–333.