Physics-Based Boiling Simulation

V. Mihalef[1], B. Unlusu[2], D. Metaxas[1], M. Sussman[3], M. Y. Hussaini[2]

[1]Department of Computer Science, Rutgers University, USA
[2]School of Computational Science, Florida State University, USA
[3]Department of Mathematics, Florida State University, USA

Abstract

In order to animate complex fluid motion, computer animators have to rely on simulation systems that automatically generate the dynamics in a physics-based manner. We focus in this paper on the phenomenon of boiling, which, due to its complex formulation and physics, has seen very little work done in the graphics field. We propose a new Eulerian method that couples gas and liquid with variable temperature and with a mass transfer mechanism, and we present its application to simulating boiling phenomena. Our philosophy is using physics-based models to obtain visually rich animations that mirror their real life counterparts, including phenomena of increased circulation in the mass of liquid, roiling boil, nucleation seeding on solid boundaries.

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

1. Introduction

In order to simulate boiling accurately one has to pay attention to the specific physical phenomena associated with it, namely the presence of a mixture of liquid and vapor and the mass transfer between them driven by gradients of temperature. One necessarily has to use a Navier-Stokes simulator that includes interfacial advection (we use a level set based one for this purpose), a temperature field providing the heat dynamics and a mass transfer mechanism. In graphics only recently researchers have paid attention to modeling and animating coupled liquid/gas systems (in which both the liquid and the gas have an active contribution to the dynamics), and usually they restrict their attention to air bubbles rising in liquid. [MSKG05] is the first work that did this in a variable temperature setup but their focus on boiling was limited, both in formulation and application. In this work we propose an Eulerian method to achieve the complex goal of simulating boiling in a physics-based setting.

The fluid simulator we implemented is a two-phase flow version of the Coupled Level Set and Volume Of Fluid (CLSVOF) method of [Sus03], augmented with a temperature field and a mass transfer mechanism. The CLSVOF method has been used in graphics in a simpler setting (one phase flow) by [MMS04] to generate breaking ocean waves,

but here we apply the two-phase CLSVOF method to multiphase flows with temperature-controlled mass-transfer. The CLSVOF method combines the tracking properties of a level set, used for generating smooth surfaces, and of a volume-of-fluid function that constrains the level set solver to conserve mass. The coupled level set and volume-of-fluid equations are solved along with the Navier-Stokes equations for velocity and pressure and with an advection-diffusion solver for the temperature. The rate of mass transfer is governed by the following physically based model ([GSR99, WW00, GTA05, ET04a, ET04b]),

$$\dot{m} = -\frac{q^L \cdot n - q^G \cdot n}{L}$$

where q is the heat flux defined by $q = -k\nabla T$, n is the outward facing normal from gas to liquid, L is the latent heat of vaporization, T is the temperature, and k is the coefficient of diffusivity for the temperature. At the gas-liquid interface, we assume the temperature of the gas and the temperature of the liquid both correspond with the saturation temperature T_s. Besides treating mass transfer in multiphase flow, we also consider the interaction of the coupled two-phase flow with variable-temperature solid objects, and show the influence of varying the solid temperature on the boiling phenomenon.

In comparison to the recent work by [MSKG05], our model is based on physical considerations, rather than the treatment chosen by [MSKG05], in which mass transfer only occurs *near* the nucleation site, and depends on the local temperature itself, rather than the jump in the heat flux at the interface. Using a tried and tested physics-based approach as we do (see e.g. Son and Dhir's comparisons with experiments for mass transfer [GSR99], or for multiphase flow without mass transfer, see [JOS05]) will result in animations that conform more closely with "real" life and, therefore, possess more realism.

The following section of the paper reviews previous work related to ours. Our simulation method is presented in section 3, followed by several illustrative animations in section 4.

2. Previous work

In what follows we present a quick overview of work related to ours from computer graphics. Our simulation method is Eulerian: it uses a fixed grid and computes all the unknowns on grid elements (nodes, cell centers, face centers). This is in contrast to the Lagrangian approach, like the one used by [MSKG05], which computes unknowns along motion characteristics, usually particles.

3D Navier-Stokes solvers were introduced to animation by Foster and Metaxas [FM96], who modified the classic marker and cell method [HW65] to obtain realistic fluid behavior in an efficient manner. Stam [Sta99], decoupled the numerical scheme used by Foster and Metaxas from its dependence on the Courant-Friedrichs-Levy (CFL) number by introducing semi-Lagrangian techniques for stably computing the advection part of the Navier-Stokes equations. In order to improve the representation of the liquid surface, Foster and Fedkiw introduced in [FF01] a hybrid liquid model, combining implicit surfaces and particles, while Enright et al. [EMF02] improved the hybrid model using the particle level set method (PLS) and obtained very realistic animations of complex water surfaces. Level set based solvers like the aforementioned one are widely used lately due to their capabilities for capturing and rendering the interface as a smooth implicit surface, and also for the theoretical and implementation ease with which they can deal with topology changes. [HK05] is such a work in which the authors use the PLS as a surface capturing device in a two-phase flow setting and generate impressive bubbles at high resolution. [GH04] also uses to maximum advantage the PLS to model air-bubble entrainment by following out-of-water particles inside the water flow and attaching spheres to them. Even though their air model is not physics-based it does a good job animation-wise. While not focusing specifically on boiling, [LSSF06] simulate multiple interacting fluids and report adding a plethora of extra capabilities like air-coupling, temperature fields and surface reactions. They show an example of air bubble generation inside liquid as a consequence of a surface reaction, but they do not include a physics-based model for mass transfer in their work.

One of the main strengths of the PLS method is that it improves dramatically the mass conservation properties of the level set method. Other methods that achieve the same goal are the CLSVOF method used in graphics by [MMS04] for single-phase flow, and the CIP method used by [TFK*03] for one phase flow and by [SSK05] for two-phase flow. The CIP method advects along with a color field its gradient and obtains as a consequence a sharp interface between air and water. For a good review of CIP see [YXU01]. The CLSVOF method was devised by Sussman [SP00] and it addressed formally the loss of mass of the level set method by coupling it with a volume of fluid method. Volume of fluid methods are arguably the most widely used Eulerian methods in computational fluid dynamics, due to their unconditional mass preservation properties. In graphics [HK03] used such a method for generating bubbles in liquid. Two of the problems encountered by these methods, namely the generation of spurious volume fractions ("flotsam and jetsam") and the recovery of a smooth surface from the volume fraction field are addressed and solved by CLSVOF. As a final word on Eulerian methods we remark that all of them (PLS, CIP, CLSVOF, etc.) converge to the solution of the Navier-Stokes equations once the grid is refined. Thus, one could argue that any comparison between them boils down ultimately to their efficiency and flexibility when one performs mesh refinement. Before we move on to discuss the Lagrangian methods we could mention the Eulerian coupled map lattice method (CML) used specifically for boiling simulations of a single liquid by [Yan92,Har02]. Their work has many limitations as they are not concerned with modeling the water vapor bubbles, mass-transfer issues or interface advection, but rather heat diffusion and transfer, plus thermal buoyancy.

Another class of Navier-Stokes solvers consists of the Lagrangian methods, which are mainly implementations and extensions of the smooth particle hydrodynamics method (SPH) proposed by [Mon92]. One of the papers that is an exception to the rule is the multiple liquids interaction work of [PTB*03]. The main paper that has relevance for us is [MSKG05], in which the authors attach a temperature property to particles and use it to generate several interesting animations of a lava lamp and of boiling water flow. We should also note that the first paper in computer graphics to attach a temperature property to particles is [TPF89], in which they used a spring-mass system for modeling heating and melting of deformable objects. They also modeled phase-change between solids and fluids by varying locally the spring constant or (in the limit) by removing the spring. A newer piece of work of relevance to us is [KAG*05], in which the authors propose a comprehensive particle-based model for animating both solids and fluid as well as phase-transitions.

Next we present the details of our physics-based numeri-

cal approach to temperature driven mass-transfer in boiling. We also study some techniques for creating a "roiling" boiling pot of liquid and for interacting with heated solid objects.

3. Simulation Method

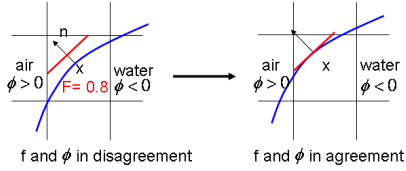### 3.1. Boundary conditions for nucleation sites

Figure 1: *The level set is updated so that the new liquid volume matches the volume fraction F obtained from the volume-of-fluid computation; the update is done by moving the piecewise linear reconstruction along the normal provided by the level set so that the correct volume is computed*

During boiling, bubbles form in the small cavities on the solid surface and grow until the buoyancy forces balance the surface tension forces. During the growth process, due to the non-uniform change in the shape of a bubble, first, a vortex forms in the liquid. As many bubbles form and rise to the surface, the liquid pushed away by the rising bubbles that flows beneath the bubbles, to fill up the emptied space, induces circulation in the pot.

In our simulations, the bubbles are seeded at an arbitrary frequency on some selected nucleation sites on the solid wall. In the partial nucleate boiling regime, a few locations are activated and bubble merging is not allowed. The number of active locations and nucleation rate are increased to simulate the fully-developed nucleate boiling regime. Rising bubbles get affected by the flow field already established due to the bubble flow from the neighboring cavitation sites. This, in turn, enhances the formation of liquid circulation. The wall effects are included in our simulations by using no-slip boundary conditions, although they have a tendency to suppress circulation of liquid close to the walls. In order to enhance circulation effects, periodic boundary conditions will be implemented in the future.

3.2. Our Fluid Solver

There are two components to our fluid solver: (1) interface capturing and (2) Navier-Stokes solver. The air/water surface is "captured" using the Coupled Level Set and Volume-Of-Fluid method (CLSVOF) [SP00]. The level set function is used to follow the interface as it is coupled with the Navier-Stokes solver. In other words, only interface cells in which the level set function changes sign are used to (a) create a signed distance function and (b) denote where the

jump in phase properties (e.g. density and viscosity) occurs. The volume fractions characteristic to the volume-of-fluid method are used to explicitly build-in mass conservation. The coupling between the level set function and the volume-of-fluid function occurs primarily during the level set re-initialization step in which the level set function is made to be the exact distance from the volume-of-fluid reconstructed interface (in cells in which the level set function changes sign), and the volume-of-fluid reconstructed interface gets the normal from the level set function (fig.1). The resulting "coupled" level set function preserves mass to a fraction of a percent in all the standard tests.

The two-phase Navier-Stokes solver that we implement in this work is a second order extension of the single-phase method developed in [Sus03] in which the gas was treated as a void. In other words, in [Sus03], the liquid was incompressible and the gas pressure was assumed spatially constant. A drawback to a single-phase implementation is that it cannot treat the entrainment of bubbles or the effect of wind on water. In [SHS*04] the authors developed a two-phase method which exhibits the same accuracy as single-phase approaches, but treats the gas as incompressible, and in this paper we follow the same approach. This approach supersedes existing first order two-phase approaches - "continuum" methods that give the phase interface a thickness, like [SSO94]. It is worth to make a note of comparison of CLSVOF with the usual surface tracker used by the graphics community, the particle level set. While the PLS favors the formation of thin sheets of liquid, CLSVOF displays more numerical surface tension, in that, if a sheet of liquid gets too thin and is close to disappearing, the method favors the formation of droplets of liquid - or bubbles of air, in order to preserve mass. In the limit, as the domain is refined, both methods *converge* to the same solution. Thus we believe that in simulations in which one prefers the formation of many bubbles or droplets, CLSVOF may be a strong candidate to choose over PLS.

The governing equations consist of the level set equation, volume-of-fluid equation, Navier-Stokes equation, temperature equation and continuity condition:

$$\dot{m} = -\frac{q^L \cdot n - q^G \cdot n}{L}$$

$$q = -k\nabla T,$$

$$n = \frac{\nabla \phi}{|\nabla \phi|}$$

$$\frac{D\phi}{Dt} = -\frac{\dot{m}}{\rho^L}|\nabla \phi| \quad \frac{DF}{Dt} = -\frac{\dot{m}}{\rho^L}|\nabla F|$$

$$\rho\frac{D\mathbf{u}}{Dt} = -\nabla p^{live} + \nabla \cdot (\mu(\nabla\mathbf{u} + \nabla\mathbf{u}^T)) - \gamma\kappa\nabla H - (\rho^L - \rho^G)g(z - z_{surf})\nabla H$$
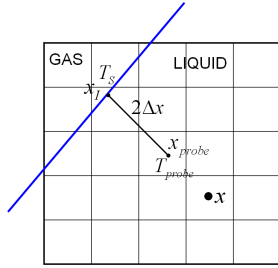
Figure 2: *Illustration of normal probe calculation for calculating the normal temperature gradient in the liquid.*

$$\frac{DT^L}{Dt} = \frac{1}{\rho^L c_v^L} \nabla k^L \nabla T^L$$

$$\frac{DT^G}{Dt} = \frac{1}{\rho^G c_v^G} \nabla k^G \nabla T^G$$

$$\nabla \cdot \mathbf{u} = \dot{m} \left(\frac{1}{\rho^G} - \frac{1}{\rho^L} \right) |\nabla H|$$

ϕ is the level set function which is positive in the liquid and negative in the gas. F is the volume-of-fluid function which satisfies $F = 0$ in computational cells containing only gas, $F = 1$ in computational cells containing only liquid and $0 < F < 1$ otherwise. The density ρ and Heaviside function H satisfy

$$\rho = \begin{cases} \rho^L & \phi \geq 0 \\ \rho^G & \phi < 0 \end{cases} \qquad H = \begin{cases} 1 & \phi \geq 0 \\ 0 & \phi < 0 \end{cases}$$

p^{live} is the "live" pressure, which is the total pressure offset by the hydrostatic pressure (z_{surf} is the initial surface water level),

$$p^{live} = p - \rho g(z - z_{surf})$$

From now on, when we write or talk about the pressure p we mean p^{live}. $\frac{D}{Dt}$ is the material (advective) derivative,

$$\frac{D}{Dt} \equiv \frac{\partial}{\partial t} + \mathbf{u} \cdot \nabla$$

All the simulations we present in this work assume outflow conditions at the top boundary, no-slip boundary conditions on the left and right walls and slip boundary conditions for the front and back walls. The temperature is fixed at the bottom wall to the wall temperature, and the temperature is fixed to the initial liquid temperature on the side walls.

3.3. Outline of numerical method for gas-liquid coupling

At the beginning of each time step, we are given the liquid (gas) velocity $\mathbf{u}^L(\mathbf{u}^G)$, the liquid (gas, solid) temperature T^L (T^G, T^S), the gas/liquid level set function ϕ, the solid/fluid level set function ψ, and the gas/liquid volume-of-fluid function F. The face centered density is defined in terms of the "height fraction" θ,

$$\rho = \rho^S \theta_{solid} + (1 - \theta_{solid})(\rho^L \theta + \rho^G (1 - \theta)$$

The height fraction θ for the liquid level set gives the one dimensional fraction of water between adjacent cells (the solid height fraction is analogously defined). For example, in 2D the height fraction would be

$$\theta_{i+1/2,j} = \frac{\phi_{i+1,j}^+ + \phi_{i,j}^+}{|\phi_{i+1,j}| + |\phi_{i,j}|}$$

where $\phi^+ = max(\phi, 0)$. With these conventions, for each time step, one does the following:

1. calculate $\dot{m}(x) = -\frac{q^L \cdot n - q^G \cdot n}{L}$. The normal temperature gradient, e.g. $\nabla T^L \cdot n$ is approximated by finding the normal probe $T_{probe}^L(x_I + 2\Delta x n)$ using linear interpolation, and setting

$$\nabla T^L \cdot n \approx \frac{T_{probe}^L - T_s}{2\Delta x}.$$

x_I represents the closest point on the zero level set to the cell center x. Please refer to figure 2.

2. update ϕ and F due to mass transfer:

$$\phi^* = \phi^n - \Delta t \frac{\dot{m}}{\rho^L} |\nabla \phi|$$

$$F^* = F^n + F(\phi^*) - F(\phi^n)$$

3. calculate provisional advective states for cell centered variables,

$$\frac{D\mathbf{u}^L}{Dt} = 0 \quad \frac{D\mathbf{u}^G}{Dt} = 0 \quad \frac{D\phi}{Dt} = 0 \quad \frac{DF}{Dt} = 0$$

$$\frac{DT^L}{Dt} = 0 \quad \frac{DT^G}{Dt} = 0$$

4. temperature diffusion,

$$\rho^L c_v^L \frac{T^{n+1,(L)} - T^{*,(L)}}{\Delta t} = \nabla \cdot k \nabla T^{n+1,(L)}$$

$$\rho^G c_v^G \frac{T^{n+1,(G)} - T^{*,(G)}}{\Delta t} = \nabla \cdot k \nabla T^{n+1,(G)}$$

The temperature along the gas/liquid boundary $\phi = 0$ is fixed at the saturation temperature T_s. The discretization of a Poisson equation in irregular geometries is described in detail by Gibou et al [GFCK02].

Figure 3: *Circulation increase in a boiling flow: (a) early "straight" pattern (b) late development mixing pattern*

5. velocity diffusion,

$$\rho \frac{u^{n+1} - u^*}{\Delta t} = \nabla \cdot \mu \nabla u^{n+1} + (\mu u_x^{n+1})_x + (\mu v_x^*)_y + (\mu w_x^*)_z$$

$$\rho \frac{v^{n+1} - v^*}{\Delta t} = \nabla \cdot \mu \nabla v^{n+1} + (\mu v_y^{n+1})_y + (\mu u_y^*)_x + (\mu w_y^*)_z$$

$$\rho \frac{w^{n+1} - w^*}{\Delta t} = \nabla \cdot \mu \nabla w^{n+1} + (\mu w_z^{n+1})_z + (\mu u_z^*)_x + (\mu v_z^*)_y$$

The face centered viscosity is defined as,

$$\mu = \frac{1}{\frac{\theta}{\mu_L} + \frac{1-\theta}{\mu_G}}$$

The velocity is defined as,

$$u^* = \begin{cases} u^{*,(L)} & \phi \geq 0 \\ u^{*,(G)} & \phi < 0 \end{cases}$$

Here, we have adopted the discretization described by Li et al. [LRR00] for treating velocity diffusion.

6. update of gravity terms and viscous terms at the cell faces,

$$\mathbf{v}^\mathbf{P} = \mathbf{u}^{\mathbf{advect,P}} + \Delta t \left(\frac{-(\rho^\mathbf{L} - \rho^\mathbf{G}) g(z - z_{surf}) \nabla H}{\rho} + \left(\frac{(\nabla \cdot \mu(\nabla \mathbf{u} + \nabla \mathbf{u}^T))}{\rho} \right)_{face} \right)$$

where $\mathbf{P}$ is taken first to be $\mathbf{L}$, then $\mathbf{G}$. The viscous force terms are interpolated from cell centers to cell faces using density weighted averaging.

7. pressure correction step.

$$\mathbf{v}_{i+1/2,j} = \begin{cases} \mathbf{v}^\mathbf{L}_{i+1/2,j} & \phi_{i+1,j} \geq 0 \text{ or } \phi_{i,j} \geq 0 \\ \mathbf{v}^\mathbf{G}_{i+1/2,j} & \text{otherwise} \end{cases}$$

$$\nabla \cdot \frac{\nabla p}{\rho} = \nabla \cdot \mathbf{v} - \dot{m} \left(\frac{1}{\rho^G} - \frac{1}{\rho^L} \right) |\nabla H|$$

The face centered Heaviside function is defined as,

$$H_{i+1/2,j} = \begin{cases} 1 & \phi_{i+1,j} \geq 0 \text{ or } \phi_{i,j} \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

8. update the density and height fractions using the current values of the level sets, then update the global velocity

$$\mathbf{u^{n+1}}_{i+1/2,j} = \begin{cases} \mathbf{v^L}_{i+1/2,j} - \frac{\nabla p}{\rho} & \phi_{i+1,j} \geq 0 \text{ or } \phi_{i,j} \geq 0 \\ \mathbf{v^G}_{i+1/2,j} - \frac{\nabla p}{\rho} & \text{otherwise} \end{cases}$$

9. set the liquid and gas velocity to be the global velocity $\mathbf{u^{n+1}}$ (they will pick up their correct component by doing this) and extrapolate the liquid velocity (this can be done as in [Sus03] or as in [EMF02]).

4. Animations

In order to demonstrate the power and wide applicability of our framework, we generated several animations using an Athlon 64 workstation with 2.4GHz and 3GB RAM and rendering in [Vd]. In all of our animations the setup was such that the bottom side of the computational domain was set up to a medium-high temperature (500 K), the vapor was set to the same temperature, and the initial liquid temperature 300 K was chosen rather close to the boiling point 373 K. We used no-slip boundary conditions on the side walls. The saturation threshold temperature was set to 405 K. The hydrostatic pressure pushed the bubbles to the surface, and their dynamics was modulated by the temperature gradients. Often the simulations evolved to very energetic boiling, as can be seen in the videos. Another effect visible in the videos is the decrease in the liquid-level during boiling. This is due to the fact that bubbles burst when they reach the liquid surface and mass is transferred to the air.

Circulation We worked inside a thin vertical domain discretized on a $128 \times 8 \times 128$ grid. We set up 6 nucleation sites which generated new bubbles at certain moments in time, chosen by us, given the requirements of the dynamics of the simulation. In particular, as time went by, we increased the generation rate of the bubbles with the specific purpose of increasing the circulation. There are no special velocity impositions at these nucleation sites, and the bubbles rise to the surface naturally, mainly due to hydrostatic forces. The
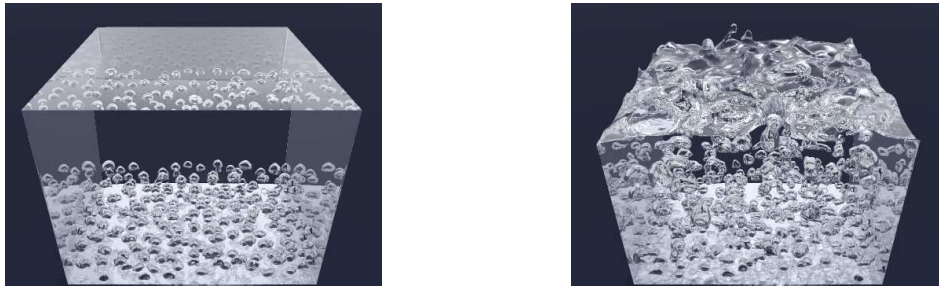
Figure 4: *Roiling boil: (a) early random pattern (b) late development*

presence of the temperature induced gradients brings on extra dynamics, and, for careful choices of the viscosity coefficients, one can get increased circulation inside the domain with very interesting velocity patterns formation (fig.3), as can be seen in the accompanying video. One of the most important roles of the viscosity is to carry along vorticity inside the computational domain, thus increasing the circulation. What we noticed verifies the theory: if we set up a small liquid viscosity and left the bubbles evolve, they would shoot straight up the domain, due to the weak "coupling" effect of the viscosity. On the other hand, if the viscosity was set up to be too large, then it would dampen the motion enough so that, again, the bubbles would shoot straight up (albeit at a rather slow pace). For a range of coefficients in between these extreme values (in the results we show our liquid viscosity was 0.5) one gets various amounts of circulation. Our code used about one minute to generate a frame and the time step used was 0.008 seconds.

Roiling-boil For this animation (fig.4) we worked on a cubic domain discretized with a rather coarse $64 \times 64 \times 64$ grid. The nucleation sites were chosen randomly, close to the bottom of the domain, at every few time steps. The liquid viscosity was 0.1 and vapor viscosity 0.005. The boiling became very energetic, with lots of circulation, what one would call a roiling boil. There are numerous changes of topology taking place during the boil, and one can appreciate how well the level set handles them.

Interactions with heated objects We also performed interactions with static objects, and we present here results obtained from interaction with a sphere whose temperature varies in time (fig.5). For the first part of the simulation the (red) sphere has the same temperature as the hot bottom of the domain (500 K), while for the last part the (green) sphere is set to 300 K. The initially high temperature of the sphere forces the vapor bubbles to increase their own temperature and consequently grow in volume as they travel to the surface. As a note, in the "circulation" animations discussed above, the bubbles' rate of growth on their way up is smaller, as expected. Another interesting effect is that on the high-

temperature sphere appear extra nucleation zones that extend along its surface. This sets up a strong boiling dynamics that diminishes immediately after switching the temperature to 300 K (the color changes to green).

5. Conclusion

We introduced to graphics a new physics-based method for modeling and simulating boiling-type phenomena. We used a two-phase flow formulation of the CLSVOF method augmented with a temperature field and a physics based mass transfer mechanism, and witnessed the extra visual complexity introduced by this extended formulation. While the space limitation of a conference paper doesn't allow for an in-depth study of the dependence of the boiling dynamics on the various parameters available for "tweaking" (boundary conditions or various geometric and physical parameters) we have on our agenda carrying out such a task and reporting on it in a future journal paper. Incorporating two-way interactions with heated deformable objects is another avenue of future work that will increase the applicability and realism of our work.

References

[EMF02] ENRIGHT D., MARSCHNER S., FEDKIW R.: Animation and rendering of complex water surfaces. *ACM TOG 21*, 3 (2002), 736–744.

[ET04a] ESMAELI A., TRYGGVASON G.: Computations of film boiling. part i: numerical method. *Intl. J. Heat and Mass Trans. 47* (2004), 5451–5461.

[ET04b] ESMAELI A., TRYGGVASON G.: Computations of film boiling. part ii: multi-mode film boiling. *Intl. J. Heat and Mass Trans. 47* (2004), 5463–5476.

[FF01] FOSTER N., FEDKIW R.: Practical animation of liquids. *Proceedings of SIGGRAPH 2001*, 23–30 (2001).

[FM96] FOSTER N., METAXAS D.: Realistic animation of liquids. *Graphical Models and Image Processing 58* (1996), 471–483.
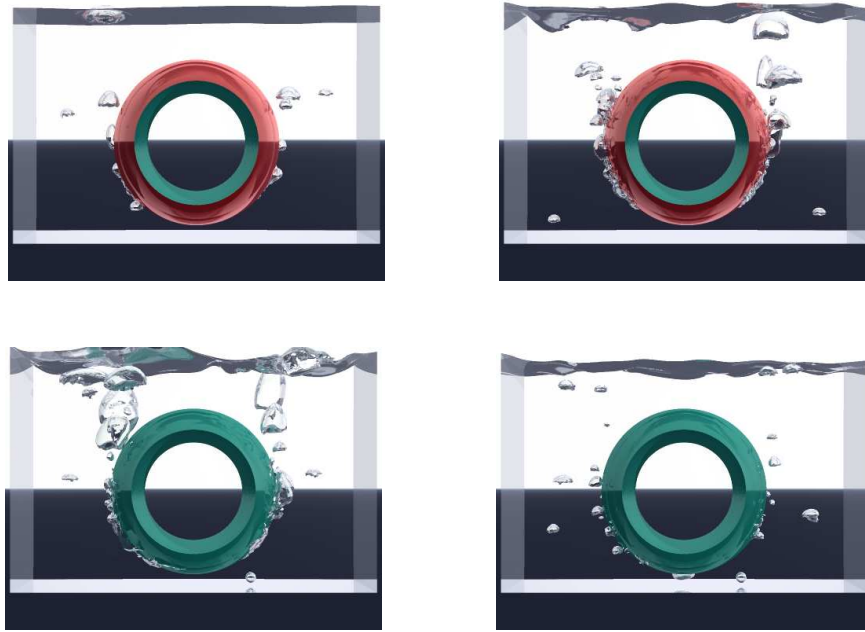
Figure 5: *Interaction with heated objects: from left to right and top to bottom, the simulation at times 2.0, 2.3, 2.5 and 2.7. The sphere temperature is 500 K before t=2.5 and 300 K afterwards. The third picture shows the moment when the temperature switch is performed.*

[GFCK02] GIBOU F., FEDKIW R., CHENG L.-T., KANG M.: A second order accurate symmetric discretization of the poisson equation on irregular domains. *Journal of Computational Physics 176* (2002), 205–227.

[GH04] GREENWOOD S. T., HOUSE D. H.: Better with bubbles: enhancing the visual realism of simulated fluid. In *Proceedings of the 2004 Symposium of Computer Animation* (2004), ACM SIGGRAPH/Eurographics, ACM Press, pp. 287–296.

[GSR99] G. SON V. K. D., RAMANUJAPU N.: Dynamics and heat transfer associated with a single bubble during nucleate boiling on a horizontal surface. *J. Heat Trans. 121* (1999), 623–631.

[GTA05] G. TOMAR G. BISWAS A. S., AGRAWAL A.: Numerical simulation of bubble growth in film boiling using a coupled level-set and volume of fluid method. *Phys. of Fluids* (2005).

[Har02] HARRIS M. J.: Implementation of a CML boiling simulation using graphics hardware. *UNC Chapel Hill, Dept of Computer Science Report* (2002).

[HK03] HONG J.-M., KIM C.-H.: Animation of bubbles in liquid. *Computer Graphics Forum 22*, 3 (2003), 253–262.

[HK05] HONG J.-M., KIM C.-H.: Discontinuous fluids. *ACM Transactions on Graphics 24*, 3 (2005), 915–920.

[HW65] HARLOW F. H., WELCH J. E.: Numerical calculation of time-dependent viscous incompressible flow of fluid with a free surface. *Physics of Fluids 8* (1965), 212–218.

[JOS05] JIMENEZ E., OHTA M., SUSSMAN M.: A computational study of bubble motion in newtonian and viscoelastic fluids. *Fluid Dynamics and Materials Processing 1(2)* (2005), 97–107.

[KAG*05] KEISER R., ADAMS B., GASSER D., BAZZI P., DUTRE P., GROSS M.: A unified Lagrangian approach to solid-fluid animation. *Proceedings of Eurographics Symposium on Point-Based Graphics* (2005).

[LRR00] LI J., RENARDY Y., RENARDY M.: Numerical simulation of breakup of a viscous drop in simple shear flow through a volume-of-fluid method. *Physics of Fluids 12(2)* (2000), 269–282.

[LSSF06] LOSASSO F., SHINAR T., SELLE A., FEDKIW R.: Multiple interacting liquids. *ACM Transactions on Graphics* (2006).

[MMS04] MIHALEF V., METAXAS D., SUSSMAN M.: Animation and control of breaking waves. In *Proceedings of the 2004 Symposium of Computer Animation* (2004),

ACM SIGGRAPH/Eurographics, ACM Press, pp. 315–324.

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

[MSKG05] MULLER M., SOLENTHALER B., KEISER R., GROSS M.: Particle-based fluid-fluid interaction. In *Proceedings of the 2005 Symposium of Computer Animation* (2005), ACM SIGGRAPH/Eurographics, ACM Press.

[PTB*03] PREMOZE S., TASDIZEN T., BIGLER J., LEFOHN A., WHITAKER R. T.: Particle-based simulation of fluids. *Eurographics 22*, 3 (2003), 401–410.

[SHS*04] SUSSMAN M., HUSSAINI M., SMITH K., ZHI-WEI R., MIHALEF V.: A second order adaptive sharp interface method for incompressible multiphase flow. In *Proceedings of the 3rd international conference on Computational Fluid Dynamics* (Toronto, Canada, 2004).

[SP00] SUSSMAN M., PUCKETT E. G.: A coupled level set and volume of fluid method for computing 3D and axisymmetric incompressible two-phase flows. *Journal of Computational Physics 162* (2000), 301–337.

[SSK05] SONG O.-Y., SHIN H., KO H.-S.: Stable but non-dissipative water. *ACM Transactions on Graphics 24*, 1 (2005), 81–97.

[SSO94] SUSSMAN M., SMEREKA P., OSHER S.: A level set approach for computing solutions to incompressible two-phase flow. *Journal of Computational Physics 114* (1994), 146–159.

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

[Sus03] SUSSMAN M.: A second order coupled level set and volume-of-fluid method for computing growth and collapse of vapor bubbles. *Journal of Computational Physics 187* (2003), 110–136.

[TFK*03] TAKAHASHI T., FUJII H., KUNIMATSU A., HIWADA K., SAITO T., TANAKA K., UEKI H.: Realistic animation of fluid with splash and foam. In *Proceedings of Eurographics* (2003), pp. 391–399.

[TPF89] TERZOPOULOS D., PLATT J., FLEISHER K.: Heating and melting deformable models (from goop to glop). *Proceedings of Graphics Interface* (1989), 219–226.

[Vd] VUE-D'ESPRIT: http://www.e-onsoftware.com.

[WW00] WELCH S. W. J., WILSON J.: A volume of fluid based method for fluid flows with phase change. *J. Comp. Phys. 160* (2000), 662–682.

[Yan92] YANAGITA T.: Phenomenology of boiling: a coupled map lattice model. *Chaos 2* (1992), 343–350.

[YXU01] YABE T., XIAO F., UTSUMI T.: The constrained interpolation profile method for multiphase analysis. *Journal of Computational Physics 169* (2001), 556–593.