A Hybrid Lagrangian-Eulerian Formulation for Bubble Generation and Dynamics

Saket Patkar* Stanford University
Mridul Aanjaneya* Stanford University
Dmitriy Karpman* Stanford University
Ronald Fedkiw* Stanford University, Industrial Light + Magic

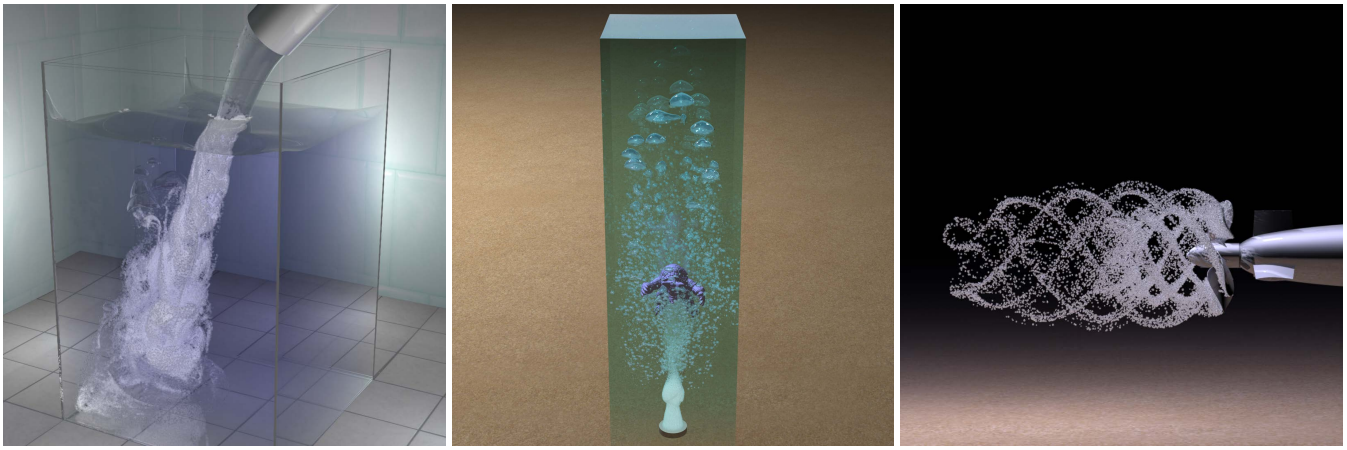

Figure 1: *(Left) a faucet generating bubbles through air entrainment, (Center) a source seeding tiny bubbles which merge and grow as they rise, as well as interact with a moving armadillo illustrating complex object interaction, (Right) a cavitating propeller generates the characteristic helical pattern in its wake.*

Abstract

We present a hybrid Lagrangian-Eulerian framework for simulating both small and large scale bubble dynamics, where the bubbles can grow or shrink in volume as dictated by pressure forces in the surrounding fluid. Small under-resolved bubbles are evolved using Lagrangian particles that are monolithically two-way coupled to the surrounding flow in a manner that closely approximates the analytic bubble oscillation frequency while converging to the analytic volume as predicted by the well-known Rayleigh-Plesset equation. We present a novel scheme for interconverting between these under-resolved Lagrangian bubbles and larger well-resolved bubbles that are modeled with a traditional Eulerian level set approach. We also present a novel seeding mechanism to realistically generate bubbles when simulating fluid structure interaction with complex objects such as ship propellers. Moreover, our framework for bubble generation is general enough to be incorporated into all grid-based as well as particle-based fluid simulation methods.

CR Categories: I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Physically based modeling;

Keywords: bubbles, entrainment, cavitation, water

1 Introduction

Bubbles are ubiquitous in most underwater scenes and embellish the otherwise lifeless water by providing visual cues to the virtually imperceptible velocity field. They also provide a mechanism for sound generation [Zheng and James 2009; Moss et al. 2010]. These sounds are a consequence of the volumetric changes that bubbles undergo during their temporal evolution - volumetric changes which can be substantial when bubbles rise a significant distance, or when fast moving objects such as ship propellers interact with water. In such fluid structure interactions, lower pressure regions are generated near the objects causing some of the water to instantly vaporize through the compressible phenomena of cavitation [Brennen 1995]. Since the density of water is a thousand times larger than that of air, the vaporized water forms bubbles that quickly expand in size becoming visible to the naked eye. Thus, to realistically simulate both bubble sounds and dynamics, it is important to design numerical methods that allow bubbles to change in volume - contrary to the traditional approach of treating the air inside bubbles as incompressible flow, e.g. [Hong and Kim 2003; Hong and Kim 2005; Mihalef et al. 2006; Losasso et al. 2006b; Zheng et al. 2006; Boyd and Bridson 2012].

An early approach to particle-based bubble simulation was proposed by [Greenwood and House 2004] who generated bubbles based on escaped particles from the particle level set method of [Enright et al. 2002] similar to the approach for spray in [Foster and Fedkiw 2001] (see also [Geiger et al. 2006]). Later, a number of authors proposed additional Lagrangian techniques including [Kim et al. 2010; Busaryev et al. 2012; Kim et al. 2012; Ihmsen et al. 2012]. Although [Kim et al. 2010] did propose using a variable density Poisson solver for approximating the average bubble motion, this only gives very limited two-way interactions and ignores changes in the bubble's volume. We instead derive a monolithic (strongly coupled) approach to simulating under-resolved Lagrangian bubbles submerged in an incompressible fluid, motivated by the fully Eulerian level set work of [Aanjaneya et al. 2013]. Our sub-grid bubble model faithfully approximates the analytic bubble oscillation frequency, converges to the analytic volume predicted by the well-known Rayleigh-Plesset equation, and remains stable for large time steps.

Instead of representing each bubble as a single particle, one can use a collection of particles to model a single bubble. SPH-based methods are good candidates for such an approach [Müller et al. 2005; Cleary et al. 2007; Thürey et al. 2007]. This can aid in modeling a bigger range of bubble dynamics obtaining a wider vari-

*e-mail: {patkar,aanjneya,dkarpman,fedkiw}@cs.stanford.edu

SCA 2013, July 19 – 21, 2013, Anaheim, California.

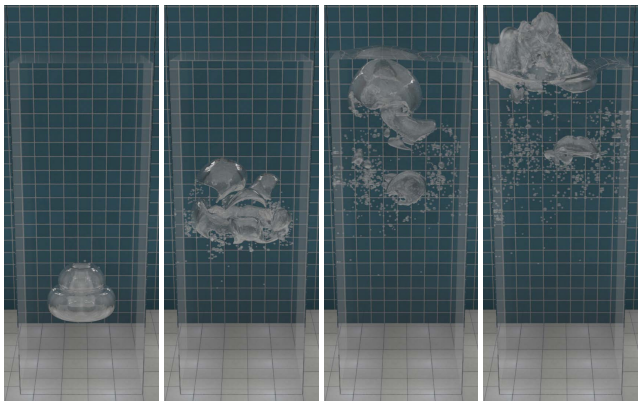

Figure 2: *A single level set bubble rises in a tall domain, undergoing topological deformations and generating smaller level set as well as sub-grid bubbles during its temporal evolution (150×500×150 grid in a 1.5m×5m×1.5m domain); about 1,200 sub-grid bubbles.*

ety of topological shapes that real bubbles exhibit. However, fully Eulerian grid-based methods still seem preferable for these larger bubbles. Moreover, some of the most compelling methods for simulating fluids tie together multiple scales as can be seen in [Losasso et al. 2008; Hong et al. 2008; Lee et al. 2009; Mihalef et al. 2009]. Therefore, we provide a mechanism to model both bubble merging and bubble growth as well as the ability to convert these larger bubbles into an Eulerian level set representation on a background grid.

Our system pipeline uses the particle level set method of [Enright et al. 2002] for tracking the interface and incompressible flow for the liquid. Small under-resolved bubbles are tracked using Lagrangian particles that are monolithically two-way coupled to the surrounding fluid. Various bubble-bubble interactions are also simulated and overlapping bubbles are merged together. Once these bubbles grow large enough to be resolved on the Eulerian grid, they are converted to their corresponding level set representation and subsequently solved for using a fully coupled monolithic Poisson solve for modeling the compressibility of these bubbles as proposed in [Aanjaneya et al. 2013]. Additionally, we convert back to the Lagrangian bubble representation whenever the level set loses air mass due to under-resolved features on the Eulerian grid. A vorticity-based seeding mechanism is also proposed to realistically model bubble generation for simulating air entrainment and complex fluid-structure interactions.

In summary, our main contributions are: 1) a monolithically two-way coupled sub-grid model for under-resolved bubbles that converges to the correct volume and matches the analytic oscillation frequency as predicted by the Rayleigh-Plesset equation; 2) an interconversion mechanism to seamlessly transition between sub-grid and level set bubbles; 3) a novel method to render the sub-grid bubbles as time-evolving level set functions to achieve visually pleasing results, and 4) a novel vorticity-based seeding mechanism for simulating bubble generation through cavitation and air-entrainment.

2 Semi-Implicit Compressible Flow

To understand the monolithically coupled Poisson solve for bubbles as proposed in [Aanjaneya et al. 2013], we first review the semi-implicit formulation for compressible flow proposed in [Kwatra et al. 2009]. The multi-dimensional Euler equations are given by,

$$\begin{pmatrix} \rho \\ \rho\vec{u} \\ E \end{pmatrix}_t + \begin{pmatrix} \nabla\cdot\rho\vec{u} \\ \nabla\cdot(\rho\vec{u})\vec{u} \\ \nabla\cdot(E\vec{u}) \end{pmatrix} + \begin{pmatrix} 0 \\ \nabla p \\ \nabla\cdot(p\vec{u}) \end{pmatrix} = \vec{0} \tag{1}$$

where ρ is the density, $\rho\vec{u}$ is the momentum, E is the total energy per unit volume and p is the pressure. For a system with internal energy per unit mass given by e, the total energy E can be written as $E = \rho e + \rho\|\vec{u}\|^2/2$. After providing an equation of state that defines p as a function of ρ and e, the above set of equations form a closed system. The flux terms in equation (1) have been expressed as a sum of advection and non-advection components. Note that the more prohibitive time step restriction of $|u| \pm c$ required by a fully explicit scheme is avoided by the explicit advection step which imposes a restriction only based on $|u|$. We denote the post-advected quantities as $\rho^\star$, $\rho\vec{u}^\star$ and $E^\star$. Note that pressure does not affect the continuity equation, so $\rho^{n+1} = \rho^\star$.

The time t^{n+1} velocities can be found by solving for the pressure component of the flux vector implicitly using a Poisson equation very similar to that for incompressible flow. Using a forward Euler discretization in time on the first two rows of equation (1) gives

$$\vec{u}^{n+1} - \vec{u}^\star = \Delta t\frac{\nabla p}{\rho^{n+1}}. \tag{2}$$

Unlike incompressible flow where one would set $\nabla\cdot\vec{u}^{n+1} = 0$, for compressible flow one can get an estimate of $\nabla\cdot\vec{u}^{n+1}$ using the pressure evolution equation [Fedkiw et al. 2002],

$$p_t + \vec{u}\cdot\nabla p = -\rho c^2\nabla\cdot\vec{u}. \tag{3}$$

Discretizing $p_t + \vec{u}\cdot\nabla p$ explicitly using a forward Euler scheme and computing the post-advected pressure $p^\star = p(\rho, e)$, as proposed in [Grétarsson and Fedkiw 2013] gives

$$\nabla\cdot\vec{u}^{n+1} = \frac{p^\star - p^{n+1}}{\Delta t\rho c^2}. \tag{4}$$

where $\nabla\cdot\vec{u}$ is fixed at time t^{n+1}. By taking the divergence of equation (2) and substituting the value of $\nabla\cdot\vec{u}^{n+1}$ from equation (4), the following implicit system for pressure is obtained after some rearrangement,

$$\frac{p^{n+1}}{\rho^n(c^2)^n} - \Delta t^2\nabla\cdot\left(\frac{\nabla p^{n+1}}{\rho^{n+1}}\right) = \frac{p^\star}{\rho^n(c^2)^n} - \Delta t\nabla\cdot\vec{u}^\star. \tag{5}$$

Discretizing the gradient and divergence operators in equation (5) to G and $-G^T$ gives

$$\left[\frac{I}{\rho^n(c^2)^n\Delta t^2} + G^T\frac{1}{\hat{\rho}^{n+1}}G\right]\tilde{p}^{n+1} = \frac{\tilde{p}^\star}{\rho^n(c^2)^n\Delta t^2} + G^T\hat{u}^\star, \tag{6}$$

where $\hat{\rho}$ and $\hat{u}$ denote density and velocity values interpolated to cell faces, and $\tilde{p}$, $\tilde{p}^\star$ denote pressure values scaled by Δt. Note that equation (6) reduces to the standard Poisson equation for incompressible flow in the limit as $c \to \infty$.

The values of p^{n+1} and $(p\vec{u})^{n+1}$ at cell faces are computed using a density-weighted average of pressure from the cell centers and setting $\hat{u}^{n+1}_{i+1/2} = \hat{u}^\star_{i+1/2} - \Delta t(\nabla p^{n+1}/\hat{\rho}^{n+1}_{i+1/2})$. These values are used to update the time t^{n+1} momentum and energy as,

$$(\rho\vec{u})^{n+1} = (\rho\vec{u})^\star - \Delta t\frac{p^{n+1}_{i+1/2} - p^{n+1}_{i-1/2}}{\Delta x} \tag{7}$$

and

$$E^{n+1} = E^\star - \Delta t \frac{(p\hat{u})^{n+1}_{i+1/2} - (p\hat{u})^{n+1}_{i-1/2}}{\Delta x} \tag{8}$$

3 Simplifications to the air flow

Equation (6) can be used to simulate fully non-linear compressible flow with shocks and rarefactions. As suggested in [Aanjaneya et al. 2013], a number of simplifications can be made for bubbles. First, an isothermal assumption can be made by choosing a simplified equation of state $p = B\rho$, decoupling the energy equation and making the first two rows of equation (1) form a closed system. This simplifies $c^2 = B$ and $p^\star = B\rho^\star = B\rho^{n+1}$, and equation (6) can be rewritten as

$$\left[\frac{I}{\rho^n B\Delta t^2} + G^T \frac{1}{\hat{\rho}^{n+1}} G\right] \tilde{p}^{n+1} = \frac{1}{\Delta t^2} + G^T \hat{u}^\star, \tag{9}$$

The spatial variations in the bubble density can also be removed by uniformly redistributing the density field to be constant inside each bubble. Furthermore, one can assume that the pressure inside the bubble is spatially constant as well, although time-varying. To get an equation for this single bubble pressure degree of freedom, one can sum up the equations corresponding to all cells that belong to a single bubble. This is equivalent to collapsing all the rows and columns that belong to a bubble into a single (albeit a long) row and column in the Poisson matrix. Summing equation (9) over all grid cells Ω that belong to a bubble gives

$$\left[\frac{N}{\rho^n B\Delta t^2} + \sum_\Omega G^T \frac{1}{\hat{\rho}^{n+1}} G\right] \tilde{p}^{n+1} = \frac{N}{\Delta t^2} + \sum_\Omega G^T \hat{u}^\star, \tag{10}$$

for a single bubble occupying N grid cells. By using Green's theorem, the second terms on both sides can be converted from a volume sum to a surface sum which is equal to the average of the quantity over the boundary multiplied by the surface area of the boundary. For a well-resolved Eulerian level set bubble occupying N grid cells, this gives the following equation

$$\frac{N}{\Delta t^2 B\rho_b} p^{n+1} - \overline{\left(\frac{\nabla p^{n+1}}{\rho}\right)} \frac{\mathcal{P}^n}{V_c} = \frac{N}{\Delta t^2} - \frac{\bar{u}^n \mathcal{P}^n}{V_c \Delta t} \tag{11}$$

where Δt is the size of the time step, V_c is the volume of a grid cell, $\bar{u}^n$ is the average radial velocity of the bubble, $\mathcal{P}^n$ is the surface area of the bubble, and $\overline{\left(\frac{\nabla p}{\rho}\right)}$ is the average density-weighted pressure gradient across the bubble-water interface. Surface tension can be included as a jump condition as shown in [Aanjaneya et al. 2013].

4 Sub-grid bubbles

We use the equation of state $P_b = B\rho_b$ for the sub-grid bubbles with the constant B chosen such that a density $\rho_b = 1.226$ kg/m^3 gives a pressure $P_b = 101{,}325$ Pa. Since small bubbles tend to remain spherical due to surface tension effects we assume that the sub-grid bubbles are spherical in shape with radius r, have a single radial velocity degree of freedom v_r, and a single pressure degree of freedom P_b which is coupled to all the surrounding fluid degrees of freedom in a monolithic fashion.

When solving for the bubble volumes, monolithic approaches such as [Aanjaneya et al. 2013] are preferable to partitioned approaches because they do not require additional relaxation techniques for stability and robustness (see e.g. [Zheng et al. 2006; Kim et al. 2007]). Therefore for stability reasons, we follow an approach similar to [Aanjaneya et al. 2013]. We would like to use a similar equation for the sub-grid bubbles as well so that they have the same qualitative behavior as the level set bubbles and seamlessly convert into them when they grow large enough. A brute force approach for achieving this by creating a mesh for each sub-grid bubble would result in increased complexity and poor conditioning due to small control volumes. Instead, we make some approximations noting that our resulting scheme gives adequate results as illustrated in Figure 3.

First, we substitute $\bar{u}^n = v_r^n$ and $N = V_b^n / V_c$, where V_b^n is the volume of the bubble, and rewrite equation (11) as,

$$\frac{V_b^n}{V_c \Delta t^2 B\rho_b} P_b - \overline{\left(\frac{\nabla p}{\rho}\right)} \frac{\mathcal{P}^n}{V_c} = \frac{V_b^n}{V_c \Delta t^2} - \frac{v_r^n \mathcal{P}^n}{V_c \Delta t} \tag{12}$$

Notice as $\Delta t \to 0$, the first term on each side of the equation must balance indicating that the bubble pressure equals the equation of state pressure. Moreover, when the bubble pressure is identical to the equation of state pressure these terms cancel, and in order to remain at equilibrium with $v_r^n = 0$ the term $\overline{\left(\frac{\nabla p}{\rho}\right)} \frac{\mathcal{P}^n}{V_c}$ must also vanish. This means that the bubble pressure tries to match the average external pressure from the fluid when it is near radial equilibrium (n.b. equation (14)). Note that $\overline{\left(\frac{\nabla p}{\rho}\right)}$ is an area-weighted average where the weights are computed based on the fraction of the bubble's surface area visible to a neighboring fluid cell and that cell's pressure degree of freedom p_i. We *estimate* these weights w_i as the weights each of the neighboring eight cells would have in a tri-linear interpolation formula for the location of the center of a bubble. Then

$$\overline{\left(\frac{\nabla p}{\rho}\right)} \approx \sum_{i=1}^{8} w_i \frac{(p_i - P_b)}{\Delta x_i \rho} \approx \sum_{i=1}^{8} w_i \frac{(p_i - P_b)}{\Delta x \rho_b} \tag{13}$$

where Δx_i is the distance between the sub-grid bubble center and the center of the ith incompressible cell. We have found that we can make further approximations replacing Δx_i by a characteristic length Δx and replacing ρ by the bubble density ρ_b as seen in the rightmost term in equation (13). Here Δx is chosen as the length of a grid cell in the case of our uniform grid. Although these approximations might appear aggressive, they allow us to treat the sub-grid bubbles as point particles while keeping the equations well-defined even for degenerate cases where the sub-grid bubbles overlap each other or encompass a fluid degree of freedom. Note that

$$\sum_{i=1}^{8} w_i \frac{(p_i - P_b)}{\Delta x \rho_b} = \frac{p_{\text{avg}} - P_b}{\Delta x \rho_b} \tag{14}$$

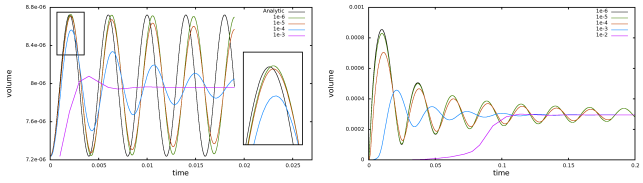

Figure 3: *Using equations (15) and (22), we solve the oscillating bubble problem of [Aanjaneya et al. 2013] for a sub-grid bubble (radius = .3Δx) on a 25^3 grid in a 1m^3 domain with (Left) an initial density of 1.1 kg/m^3, where the bubble starts with an initial volume of 7.238 cm^3, converges to the predicted volume of 7.962 cm^3, and closely approximates the analytic bubble oscillation frequency as the size of the time step is refined, and (Right) an initial density of 1,100 kg/m^3, where the bubble starts with an initial volume of .268 cm^3 and expands three orders of magnitude, remaining stable even when it grows beyond its incompressible neighbors. Note that the bubble remains stable at all time steps in both cases.*

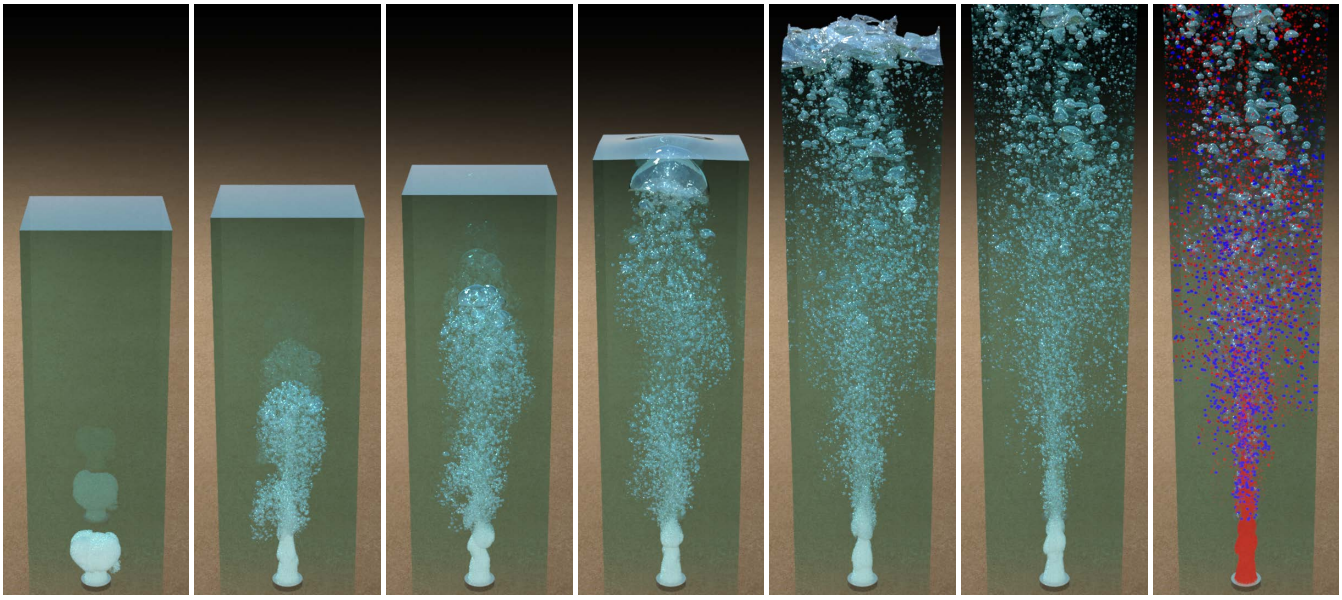

Figure 4: *A cylindrical source inside a tall domain seeding tiny bubbles which grow as they rise and merge together due to attraction forces, ultimately forming large sub-grid and level set bubbles as they approach the surface (128×640×128 grid in a 50m×200m×50m domain); about 500,000 sub-grid bubbles. (Far right) shows the sub-grid bubbles in red and blue, where red depicts the smaller spherical ones (Figure 6 left), and blue depicts those rendered using the time evolving level set dictionary (Figure 6 right). Note that the domain size is very large to accentuate the volume growth and the bubbles expand up to 10 times in volume.*

where p_{avg} is the incompressible flow pressure linearly interpolated to the bubble's center. This provides some intuition as to why these approximations work. The term on the right is a reasonable approximation to the gradient between the incompressible flow pressure and the bubble's pressure using the same characteristic length scale.

In summary, we use the following equation when solving for the pressure of a sub-grid bubble,

$$\frac{V_b^n}{V_c \Delta t^2 B \rho_b} P_b - \sum_{i=1}^{8} w_i \frac{(p_i - P_b)\mathcal{P}^n}{\Delta x \rho_b V_c} = \frac{V_b^n}{V_c \Delta t^2} - \frac{v_r^n \mathcal{P}^n}{V_c \Delta t} \quad (15)$$

Even with these approximations, we converge to the analytic bubble volume at sub-grid resolutions and the simulation remains stable with large time steps. The denominator ρ in equation (13) controls the bubble's oscillation frequency. For any given radius, it can be tuned such that the model closely matches the analytic bubble oscillation frequency. Most of our sub-grid bubbles are seeded with radii in the interval $(.2\Delta x, .3\Delta x)$, and setting $\rho = \rho_b$ works quite well in this case as shown in Figure 3(left). For more accuracy such as when simulating fluid sounds, one could choose a better value for ρ or even make it a function of the bubble's radius. We leave this as future work since all our examples use large time steps and only rely on the sub-grid bubbles converging to the right volume while remaining stable.

Note that we treat all sub-grid bubbles independently of each other when coupling to the external fluid pressures. In addition, we set weights to zero when a neighboring fluid degree of freedom is inside a kinematic object or is subject to a free surface pressure boundary condition. This means that both kinematic objects and free air cannot see the pressure from the sub-grid bubble which is fine. When a neighboring fluid degree of freedom is inside a level set bubble, the level set pressure degree of freedom can be coupled to the sub-grid bubble pressures, except that this increases the density of the Poisson equation matrix repeatedly for every sub-grid bubble adjacent to a single level set region - which we have observed increases the number of iterations required by PCG for convergence. Therefore, we set weights to zero in this case as well.

As shown in [Aanjaneya et al. 2013] and [Kwatra et al. 2009], the divergence of the velocity field at time t^{n+1} is given by the second and fourth terms in equations (11) and (12), and thus equation (15). Since the volume-weighted divergence of a sub-grid spherical bubble is given by $v_r \mathcal{P}$, we can write

$$v_r^{n+1}\mathcal{P}^{n+1} = V_c D^{n+1} = v_r^n \mathcal{P}^n - \Delta t \mathcal{P}^n \sum_{i=1}^{8} w_i \frac{(p_i - P_b)}{\Delta x \rho_b} \quad (16)$$

After solving a monolithically coupled Poisson equation for all the fluid and bubble pressures, the right hand side of equation (16) is known. Using the definition of surface area $\mathcal{P}^{n+1} = 4\pi(r^{n+1})^2$ and writing $v_r^{n+1} = dr^{n+1}/dt$, we analytically integrate equation (16) from time t^n to t^{n+1} to obtain r^{n+1}. Once r^{n+1} is determined, we use a backward Euler discretization of $r^{n+1} - r^n = \Delta t v_r^{n+1}$ for computing v_r^{n+1}.

5 Coupling to Incompressible Flow

Consider an incompressible fluid containing many sub-grid bubbles with the inviscid Navier-Stokes equations given by

$$\vec{u}_t + (\vec{u} \cdot \nabla)\vec{u} + \frac{\nabla p}{\bar{\rho}} = \vec{g} \quad (17)$$

where $\bar{\rho}$ is the average density, $\vec{u}$ is the velocity, and $\vec{g}$ is the net body force acting on the fluid. We discretize these equations on a MAC grid where we first explicitly update

$$\frac{\vec{u}^\star - \vec{u}^n}{\Delta t} + (\vec{u} \cdot \nabla)\vec{u} = \vec{g} \quad (18)$$

with a semi-Lagrangian MacCormack method [Selle et al. 2008], and then solve for the pressure via

$$\nabla \cdot \frac{\nabla p}{\bar{\rho}} = \frac{\nabla \cdot \vec{u}^\star}{\Delta t} - \frac{\nabla \cdot \vec{u}^{n+1}}{\Delta t} \quad (19)$$

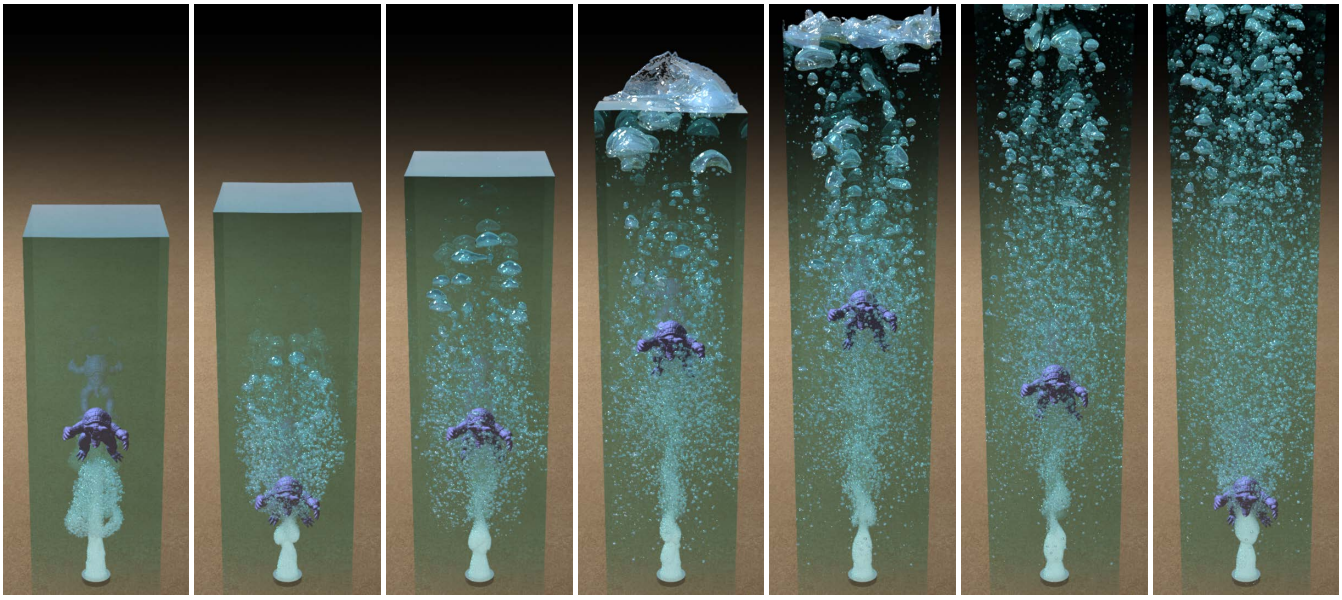

Figure 5: *Same as Figure 4 except an armadillo moving up and down illustrating complex object interaction.*

in order to update the intermediate velocity $\vec{u}^\star$ as follows

$$\frac{\vec{u}^{n+1} - \vec{u}^\star}{\Delta t} + \frac{\nabla p}{\bar{\rho}} = 0 \tag{20}$$

We generally follow [Enright et al. 2002] using the semi-Lagrangian level set advection of [Enright et al. 2005], a fast marching method reinitialization of [Losasso et al. 2006a], and a second order cut-cell pressure discretization of [Enright et al. 2003].

We lump the divergences of all the sub-grid bubbles into a column vector $\vec{D}^{n+1}$ so that they can affect the divergences of the individual fluid cells via $W\vec{D}^{n+1}$, where W is a weight matrix with W_{jk} corresponding to the fraction of the divergence of bubble k that is added to the divergence of cell j. For simplicity of exposition, consider a single bubble where we only consider one column of the weight matrix which we index solely by the cell, i.e., for example W_j. Then, equation (19) can be written as

$$-\nabla \cdot \frac{\nabla p}{\bar{\rho}} = -\frac{\nabla \cdot \vec{u}^\star}{\Delta t} + \frac{W\vec{D}^{n+1}}{\Delta t} \tag{21}$$

With the aid of equation (16), we discretize equation (21) for cell j with faces f as follows

$$-\sum_{f=1}^{6} \frac{p_f - p_j}{\bar{\rho}\Delta x^2} + W_j \sum_{i=1}^{8} w_i \frac{(p_i - P_b)\mathcal{P}^n}{\Delta x \rho_b V_c} = -\frac{\nabla \cdot \vec{u}^\star}{\Delta t} + W_j \frac{v_r^n \mathcal{P}^n}{V_c \Delta t} \tag{22}$$

where p_f refers to the pressure on the other side of the face f. For multiple bubbles, the second and fourth terms in equation (22) must be summed over all the influencing bubbles k with W_j replaced by W_{jk}.

The weight matrix W can be chosen such that the resulting system of equations (15) and (22) is symmetric positive definite allowing for the use of fast solvers such as preconditioned conjugate gradient. In order to obtain symmetry, the coefficient of P_b in the second term in equation (22) must be the same as the coefficient of p_j in equation (15), and the coefficient of p_i (when $i \neq j$) must be the same as that for p_j in the corresponding equation for cell i. The first condition means that $w_j = W_j \sum_{i=1}^{8} w_i$ or $W_j = w_j / \sum_{i=1}^{8} w_i$. Note that this relation automatically satisfies the second condition for symmetry as well. Typically, since w_i represents the interpolation weight, $\sum_{i=1}^{8} w_i = 1$ and we are simply using the interpolation weights once again to define W. However, as pointed out in Section 4, objects, level set bubbles and the free surface are not directly coupled to the sub-grid bubble's pressure discretization yielding weight values of zero and $\sum_{i=1}^{8} w_i \neq 1$. Technically, this means that our sub-grid bubbles are not directly coupled to objects, level set bubbles, or free surface pressure boundary conditions but are always assumed to be submerged in the neighboring fluid degrees of freedom that happen to be present.

Finally, after solving for the pressure and updating the fluid velocities in the usual manner, the translational velocity of the bubble is set to be the interpolated average fluid velocity $\vec{u}$ at the center of the bubble. One could make the bubble motion more lively by applying additional forces such as buoyancy, vorticity confinement or a random perturbation as done in [Kim et al. 2010]. We use buoyancy in some of our examples since the lift from variable density Poisson solve might not be enough if the bubble is too small compared to a grid cell. For greater accuracy, this new velocity can then be subtracted from the average velocities to conserve the fluid momentum, although this step is not essential since the bubble momenta is very small.

6 Bubble-Bubble Interactions

When two sub-grid bubbles overlap, we merge them into a single bubble adding their masses and volumes. The radial velocity of the bubble is chosen such that the net divergence is equal to the sum of the divergences of the original bubbles. Additionally, for increased realism, similar to [Hong et al. 2008] we apply an attraction force which is of the form $f_{\text{attract}} = Km_1m_2/r^2$, where K is a constant, m_1, m_2 are the masses of the two bubbles and r is the distance between them. When a sub-grid bubble grows large enough such that its radius covers more than two grid cells, we convert it to a level set function as proposed in [Song et al. 2005]. This is accomplished by rasterizing the sub-grid bubble onto the grid and adding its mass to the level set region by computing the appropriate density. Note that the divergence remains continuous during this process since the background fluid velocity already contains the bubble's divergence.

Also, when a sub-grid bubble enters a level set bubble we delete the sub-grid bubble and add its mass to the mass of the level set region by modifying the density field.

If a level set bubble becomes smaller than a grid cell, it can lose mass because of numerical errors during advection. However, the bubble mass cannot disappear because it is advected conservatively using the method of [Lentine et al. 2011]. This stray mass was distributed to the nearby bubbles in [Aanjaneya et al. 2013]. However, such a scheme can sometimes move the bubble mass too far away in a non-physical manner. Instead, we propose to track this stray mass using sub-grid bubbles as shown in Figure 2. To achieve this, we first run a greedy condensation procedure on the stray density field by moving it in the direction of the gradient vectors for a few iterations. Then for every cell with density above some threshold we seed a sub-grid bubble with the appropriate mass. To correctly choose its volume, we set the steady state pressure $p = \rho_I g h$ (where ρ_I is the density of the incompressible fluid and h is the depth of the sub-grid bubble from the water surface) to be equal to the equation of state pressure $P_b = B\rho_b = BM_b/V_b$ and solve for V_b. Note that we do not use the incompressible pressure for computing the bubble's volume because it can oscillate wildly and even go negative at times during the course of the simulation due to small numerical errors in the velocity field - this is because of the well-known fact that the fluid pressure in incompressible flow is more of a Lagrange multiplier (see [Majda 2001]) than an actual pressure. Finally note that even if our initial volume estimate has some errors, the monolithic coupling keeps the scheme stable and the bubble readily changes volume to an appropriate value.

In summary, starting from initial data containing level set and sub-grid bubbles and a volumetric field that represents the bubble mass, one loop of our pipeline runs as follows: the level set function is advanced using the particle level set method of [Enright et al. 2002] and the bubble mass is advected using the unconditionally stable, fully conservative, semi-Lagrangian advection scheme of [Lentine et al. 2011]. After advection, the bubble mass and the level set function might be inconsistent as they are advanced using different advection schemes and also because some small level set bubbles might have disappeared due to numerical errors during advection. To make them consistent, the mass surrounding a bubble is uniformly redistributed such that the bubble density is spatially constant inside the bubble. Next, the sub-grid bubbles are advected forward after applying forces such as buoyancy and attraction. Subsequently, inter-conversions are handled by merging overlapping bubbles, converting stray density to sub-grid bubbles and converting large sub-grid bubbles to their corresponding level set representation. Finally, a coupled system is solved where equation (11) is written per level set bubble, equation (15) per sub-grid bubble and the standard incompressible flow equations with the modification for sub-grid bubbles, i.e., equation (22) in the rest of the fluid to get a pressure at every grid cell center as well as at every sub-grid bubble. This pressure is then used to update the fluid velocities and the radial velocities of the sub-grid bubbles. To update the air velocities, we perform a second projection step using fluid velocities at the bubble-water interface as Neumann boundary conditions, as described in [Aanjaneya et al. 2013].

7 Time-evolving proxy geometry

Although sub-grid bubbles are monolithically coupled to the surrounding fluid, rendering them as oscillating spheres next to fully deforming level set bubbles can look visually disturbing. To avoid this, we render them as time-evolving level set functions which have been pre-computed offline. This was achieved by maintaining a dictionary of level sets acquired from a rising bubble simulated on a coarse grid. During the rendering process these level sets are substituted within the bounding boxes of the sub-grid bubbles and intersected with the rays, as shown in Figure 6. We used time as key for choosing a level set from the dictionary, looping back after a certain period of time and scaled these level sets based on the bubble radii. Bubble shapes can also be handcrafted or created via superposition of certain basis functions [Moss et al. 2010] for use during the rendering process. Using this method for rendering a simulation with hundreds of thousands of sub-grid bubbles can be computationally quite expensive. However, we found that using this method on the largest few thousand bubbles added negligible computational overhead while drastically increasing the visual realism, see Figure 4(far right). Note that in some cases we have rendered the sub-grid bubbles larger than their actual radii to increase the visual expression.

Simulation Loop

```
while time < endTime do
    Compute the time step size δt.
    Advance the level set.
    Advance bubble mass using [Lentine et al. 2011].
    Redistribute the bubble mass.
    Advance sub-grid bubbles after applying forces.
    Handle bubble-bubble interactions.
        a) Merge overlapping bubbles.
        b) Convert stray density to sub-grid bubbles.
        c) Convert large sub-grid bubbles to level set bubbles.
    Convect both water and air velocities.
    Solve the coupled system for the pressure.
    Update water velocities using the pressure.
    Update radial velocities for all sub-grid bubbles.
    Solve another projection to update velocities inside level set
    bubbles using water velocities as Neumann conditions.
    time += δt.
end while
```

8 Solid object interaction

As described in [Aanjaneya et al. 2013], when advecting the fluid velocities, the object velocity is set as a Dirichlet boundary condition at cell faces that lie inside the object. For level set advection, objects are treated as water. i.e., the level set function ϕ is initialized to be the value as if no objects were present and subsequently also updated at grid cells that lie inside the object. For advecting the air mass, the forward and backward advection rays are clamped when they hit an object, as described in [Guendelman et al. 2005; Lentine et al. 2011]. The surface point is used for computing the

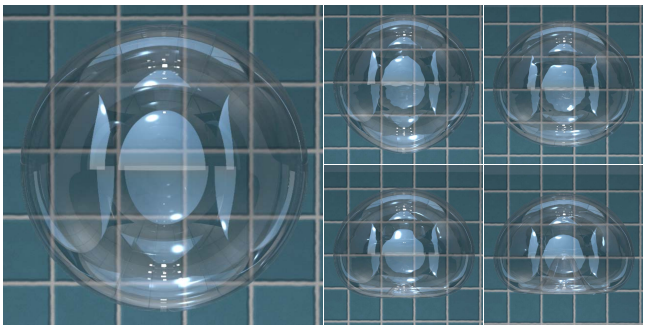

Figure 6: *To preserve visual realism, we render a sub-grid bubble as a time-evolving level set function by maintaining a dictionary of level sets acquired from a coarse simulation and intersecting rays with them during the rendering process. Shown in the figure is a sub-grid bubble rising on a 6×18×6 grid rendered (Left) as a sphere, and (Right) at different points in time using our level set dictionary.*

Figure 7: *A fast moving hydrofoil generates the typical Von Karman vortex street in its wake through cavitation. The vortex street is generated because of the two-dimensional cross-sectional nature of the hydrofoil (1024×128×128 grid in a 50m×200m×50m domain); about 600,000 sub-grid bubbles.*

interpolation weights, where weights coming from cells inside the object are discarded and the remaining weights are rescaled to sum to 1. Similarly, when advecting the sub-grid bubbles we clamp the advection rays to the object surface. During the pressure projection step, we set Neumann boundary conditions at cell faces that lie inside thick objects, while for thin shells we use visibility information as described in [Guendelman et al. 2005].

As discussed in Section 4, if a sub-grid bubble has a neighbor that lies inside an object then the weight w_i is set to zero. This means that sub-grid bubble pressures do not directly couple with solid objects. Although this is fine for kinematically coupled objects, the method should be modified in order to properly handle interactions with two-way coupled objects, increasing the overall complexity. We leave this as interesting future work. Figure 5 shows a kinematically coupled armadillo moving inside the underwater bubble simulation of Figure 4. Note that the larger level set bubbles form earlier in this example because of the merging of the sub-grid bubbles due to collisions with the armadillo.

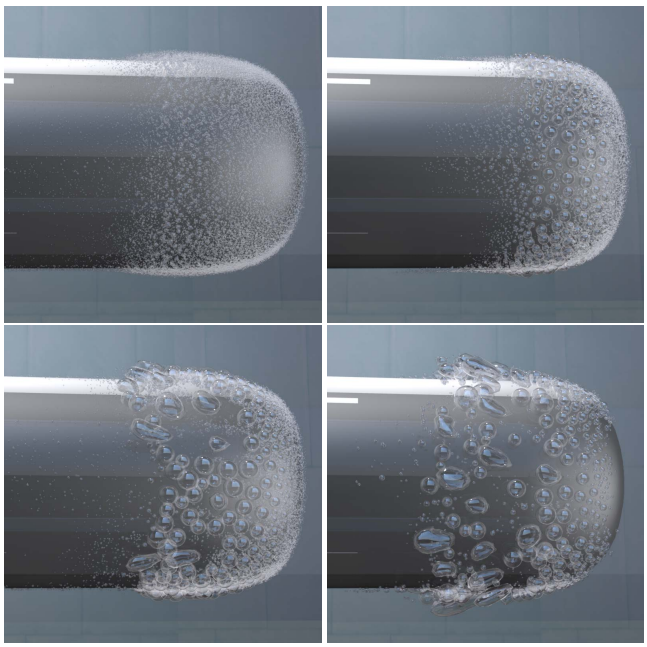

Figure 8: *Headforms with varying surface roughness to illustrate different nuclei densities in water ranging from a large number of small bubbles to a few large bubbles (*$512 \times 256 \times 256$ *grid in a* $1m \times .5m \times .5m$ *domain). Note that our results are qualitatively similar to the experimental results on bubble cavitation shown in Figure B.3 in [van Terwisga].*

9 Bubble seeding

When considering fluid structure interactions with fast moving objects such as ship propellers, lower pressure regions are generated near the object and some of the water instantly vaporizes through cavitation [Brennen 1995] and forms bubbles. Since the density of water is a thousand times larger than that of air, these bubbles quickly expand in size becoming visible. The problem of modeling bubble generation has been addressed by various authors for phenomena such as boiling [Mihalef et al. 2006; Kim and Carlson 2007] or air entrainment [Greenwood and House 2004; Hong et al. 2008; Mihalef et al. 2009]. While the former schemes predict bubble seeding locations using temperature and objects, the latter set of schemes use the escaped level set particles. As a result, all these schemes are unsuitable for modeling cavitation. Note that it is extremely difficult to vaporize pure water due to strong cohesion forces between the water molecules. Thus, the major mechanism for cavitation is through nuclei that are very tiny bubbles (of the order of microns) already present in water, or which are generated near rough surfaces. When these bubbles enter lower pressure regions, they quickly grow in size becoming visible to the naked eye.

Although lower pressure regions might appear to be good candidates for seeding bubbles, this idea does not work well in practice because the incompressible pressure behaves like a Lagrange multiplier (see [Majda 2001]), as mentioned in Section 6. The incompressible flow velocities, on the other hand, are much more reliable. We observed that the vorticity of the velocity field is a very good predictor for cavitating regions and thus, we determine such regions by thresholding the vorticity magnitude. Note that one should be careful when computing vorticity at fluid cells bordering objects because the vorticity magnitude can be erroneously high due to Neumann boundary conditions. To avoid this, we first extrapolate the fluid velocity inside objects and then compute vorticity. We seed bubbles with small radii, zero radial velocity, and use the steady state pressure for setting their mass, as described in Section 6. Once seeded, these bubbles stably grow to their correct volume in a few time steps because of the monolithic coupling scheme. Figure 1(far right) shows the characteristic helical pattern generated by a cavitating propeller simulated on a $512 \times 256 \times 256$ grid in a $2m \times 1m \times 1m$ domain, and Figure 7 shows a cavitating hydrofoil generating the typical Von Karman vortex street.

A nucleus in a lower pressure region keeps growing until it becomes large enough to affect the surrounding pressure. The number of cavitating bubbles and their size is determined by the nuclei density in water, a high nuclei density implying that there are many cavitation sites in lower pressure regions and so each nuclei can only grow by a small amount before it starts affecting the surrounding pressure - ultimately manifesting as a mist of small bubbles. Solid objects affect the nuclei density in proportion to their surface roughness, rougher surfaces generating more nuclei. Our method allows us to emulate different nuclei densities by varying the magnitude of the

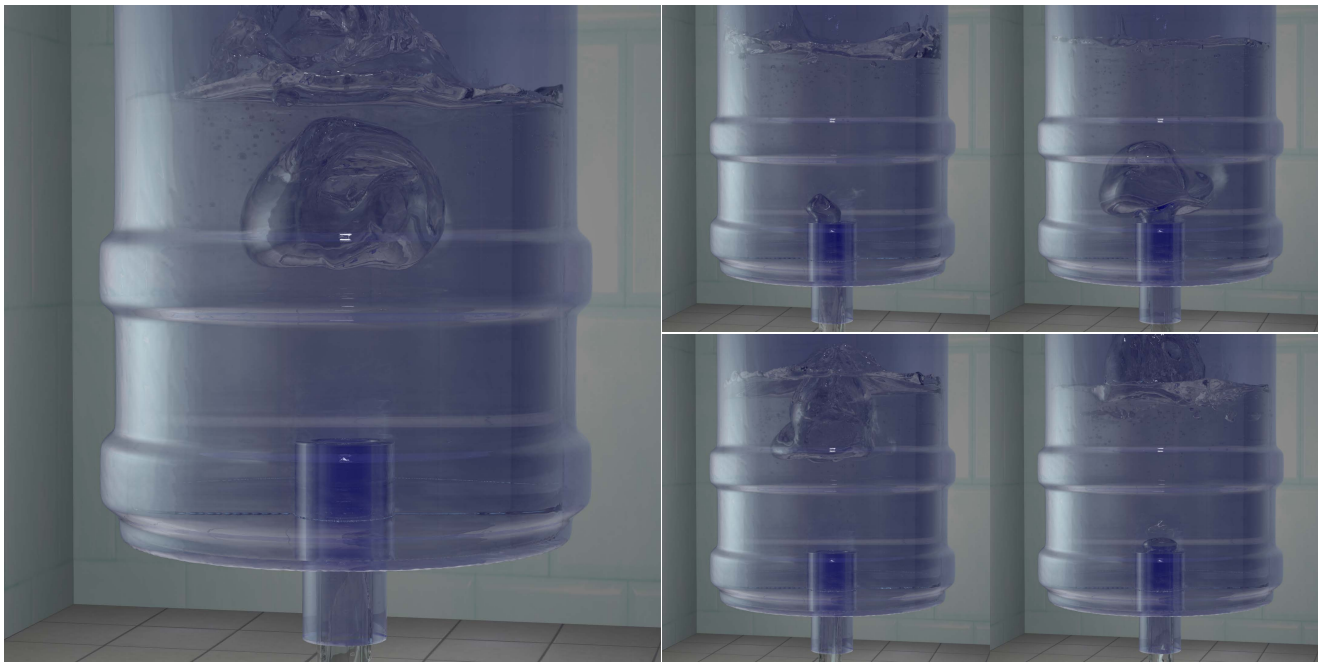

Figure 9: *A fully simulated water dispenser (200×300×200 grid in a 1m×1.5m×1m domain). As water exits the spout, the air pressure above the free surface decreases and some air gets entrained from below to compensate for this pressure drop forming bubbles.*

attraction forces between the sub-grid bubbles, a small magnitude implying a higher nuclei density as less bubbles merge together and vice-versa (see Figure 8).

Although initially designed for simulating cavitation, we found that our vorticity-based seeding mechanism works well for simulating air entrainment as well because of high vorticity at the boundary between the faster impinging jet and the slower surrounding flow (see [Ihmsen et al. 2012]). Figure 10 shows a faucet pouring water into a container entraining bubbles. Figure 9 shows a fully simulated water dispenser where the free air above the water surface expands whenever water pours out decreasing the air pressure. To balance this pressure drop, some air gets entrained from below forming bubbles, and the process continues.

10 Performance

The constant pressure solver proposed by [Aanjaneya et al. 2013] is about 6 times faster in the projection step than a standard two-phase incompressible flow solver. Adding the sub-grid bubbles has a negligible effect on the simulation time, since the new degrees of freedom added (approximately 500,000 sub-grid bubbles) are still only a small fraction of the grid-based degrees of freedom (approximately 15 million). Memory-wise we need to store an additional velocity field for the air phase and a density field for tracking the bubble mass. Computation-wise we incur an additional cost for advecting the bubble density and air velocities as well as for the second projection step, but these costs are insignificant compared to the speedup achieved by the coupled projection step.

11 Conclusion

We proposed a novel method for simulating bubbles at sub-grid resolution using Lagrangian particles that are monolithically coupled to the surrounding fluid. We showed that despite the aggressive approximations made in Section 4, our sub-grid model still closely approximates the analytic bubble oscillation frequency and converges to the analytic volume as predicted by the Rayleigh-Plesset equation while remaining stable even for large time steps. Moreover, our method adds negligible computational overhead when simulating about 500,000 Lagrangian bubbles on a grid with roughly 15 million cells. Currently, rendering is the biggest limitation of our system since we had hundreds of thousands of transparent/reflective bubbles which were difficult to ray trace as the numbers of transmitted/reflected rays were exponential in the ray depth. This is a well-known problem as mentioned in [Jakob and Marschner 2012],

Figure 10: *Faucet pouring water showing air entrainment (200×400×200 grid in a 1m×2m×1m domain); about 300,000 sub-grid bubbles. Note that the size of the sub-grid bubbles was accentuated to highlight the complex bubble interactions and the dynamic flow field.*

and we would like to explore better methods to render them faster. In addition, sub-grid bubbles are only coupled to the surrounding water and not to each other, to level set bubbles or to objects, all of which we would like to consider in future work.

12 Acknowledgements

Research was supported in part by ONR N00014-09-1-0101, ONR N-00014-11-1-0027, ONR N00014-11-1-0707, ARL AHPCRC W911NF-07-0027, and the Intel Science and Technology Center for Visual Computing. Computing resources were provided in part by ONR N00014-05-1-0479. We would like to thank Jure Leskovec and Christos Kozyrakis for additional computing resources as well as Andrej Krevl and Jacob Leverich for helping us use those resources.

References

AANJANEYA, M., PATKAR, S., AND FEDKIW, R. 2013. A monolithic mass tracking formulation for bubbles in incompressible flow. *Journal of Computational Physics 247*, 17–61.

BOYD, L., AND BRIDSON, R. 2012. Multiflip for energetic two-phase fluid simulation. *ACM Trans. Graph. 31*, 2, 16:1–16:12.

BRENNEN, C. E. 1995. *Cavitation and Bubble Dynamics*. Oxford University Press, USA.

BUSARYEV, O., DEY, T. K., WANG, H., AND REN, Z. 2012. Animating bubble interactions in a liquid foam. *ACM Trans. Graph. 31*, 4, 63:1–63:8.

CLEARY, P. W., PYO, S. H., PRAKASH, M., AND KOO, B. K. 2007. Bubbling and frothing liquids. *ACM Trans. Graph. 26*, 3.

ENRIGHT, D., MARSCHNER, S., AND FEDKIW, R. 2002. Animation and rendering of complex water surfaces. *ACM Trans. Graph. (SIGGRAPH Proc.) 21*, 3, 736–744.

ENRIGHT, D., NGUYEN, D., GIBOU, F., AND FEDKIW, R. 2003. Using the particle level set method and a second order accurate pressure boundary condition for free surface flows. In *Proc. 4th ASME-JSME Joint Fluids Eng. Conf., number FEDSM2003–45144. ASME*.

ENRIGHT, D., LOSASSO, F., AND FEDKIW, R. 2005. A fast and accurate semi-Lagrangian particle level set method. *Computers and Structures 83*, 479–490.

FEDKIW, R., LIU, X.-D., AND OSHER, S. 2002. A general technique for eliminating spurious oscillations in conservative schemes for multiphase and multispecies euler equations. *Int. J. Nonlinear Sci. and Numer. Sim. 3*, 99–106.

FOSTER, N., AND FEDKIW, R. 2001. Practical animation of liquids. In *Proc. of ACM SIGGRAPH 2001*, 23–30.

GEIGER, W., LEO, M., RASMUSSEN, N., LOSASSO, F., AND FEDKIW, R. 2006. So real it'll make you wet. In *SIGGRAPH 2006 Sketches & Applications*, ACM Press.

GREENWOOD, S. T., AND HOUSE, D. H. 2004. Better with bubbles: enhancing the visual realism of simulated fluid. In *Proc. of the 2004 ACM SIGGRAPH/Eurographics Symp. on Comput. Anim.*, 287–296.

GRÉTARSSON, J., AND FEDKIW, R. 2013. Fully conservative, robust treatment of thin shell fluid-structure interactions in compressible flows. *Journal of Computational Physics 245*, 160–204.

GUENDELMAN, E., SELLE, A., LOSASSO, F., AND FEDKIW, R. 2005. Coupling water and smoke to thin deformable and rigid shells. *ACM Trans. Graph. (SIGGRAPH Proc.) 24*, 3, 973–981.

HONG, J.-M., AND KIM, C.-H. 2003. Animation of bubbles in liquid. *Comput. Graph. Forum (Eurographics Proc.) 22*, 3, 253–262.

HONG, J.-M., AND KIM, C.-H. 2005. Discontinuous fluids. *ACM Trans. Graph. (SIGGRAPH Proc.) 24*, 3, 915–920.

HONG, J.-M., LEE, H.-Y., YOON, J.-C., AND KIM, C.-H. 2008. Bubbles alive. *ACM Trans. Graph. 27*, 3, 48:1–48:4.

IHMSEN, M., AKINCI, N., AKINCI, G., AND TESCHNER, M. 2012. Unified spray, foam and air bubbles for particle-based fluids. *Vis. Comput. 28*, 6-8, 669–677.

JAKOB, W., AND MARSCHNER, S. 2012. Manifold exploration: a markov chain monte carlo technique for rendering scenes with difficult specular transport. *ACM Trans. Graph.*, 58:1–58:13.

KIM, T., AND CARLSON, M. 2007. A simple boiling module. In *ACM SIGGRAPH/Eurographics Symp. on Comput. Anim.*, 27–34.

KIM, B., LIU, Y., LLAMAS, I., JIAO, X., AND ROSSIGNAC, J. 2007. Simulation of bubbles in foam with the volume control method. *ACM Trans. Graph. 26*, 3.

KIM, D., SONG, O.-Y., AND KO, H.-S. 2010. A practical simulation of dispersed bubble flow. *ACM Trans. Graph. 29*, 70:1–70:5.

KIM, P.-R., LEE, H.-Y., KIM, J.-H., AND KIM, C.-H. 2012. Controlling shapes of air bubbles in a multi-phase fluid simulation. *Vis. Comput. 28*, 6-8, 597–602.

KWATRA, N., SU, J., GRÉTARSSON, J., AND FEDKIW, R. 2009. A method for avoiding the acoustic time step restriction in compressible flow. *J. Comput. Phys. 228*, 11, 4146–4161.

LEE, H.-Y., HONG, J.-M., AND KIM, C.-H. 2009. Interchangeable sph and level set method in multiphase fluids. *Vis. Comput. 25*, 5-7, 713–718.

LENTINE, M., GRÉTARSSON, J., AND FEDKIW, R. 2011. An unconditionally stable fully conservative semi-lagrangian method. *J. Comput. Phys. 230*, 2857–2879.

LOSASSO, F., FEDKIW, R., AND OSHER, S. 2006. Spatially adaptive techniques for level set methods and incompressible flow. *Computers and Fluids 35*, 995–1010.

LOSASSO, F., SHINAR, T., SELLE, A., AND FEDKIW, R. 2006. Multiple interacting liquids. *ACM Trans. Graph. (SIGGRAPH Proc.) 25*, 3, 812–819.

LOSASSO, F., TALTON, J., KWATRA, N., AND FEDKIW, R. 2008. Two-way coupled sph and particle level set fluid simulation. *IEEE TVCG 14*, 4, 797–804.

MAJDA, A. J. 2001. *Vorticity and Incompressible Flow*. Cambridge Univ Pr.

MIHALEF, V., UNLUSU, B., METAXAS, D., SUSSMAN, M., AND HUSSAINI, M. 2006. Physics based boiling simulation. In *SCA '06: Proc. of the 2006 ACM SIGGRAPH/Eurographics Symp. on Comput. Anim.*, 317–324.

MIHALEF, V., METAXAS, D. N., AND SUSSMAN, M. 2009. Simulation of two-phase flow with sub-scale droplet and bubble effects. *Comput. Graph. Forum*.

MOSS, W., YEH, H., HONG, J.-M., LIN, M. C., AND MANOCHA, D. 2010. Sounding liquids: Automatic sound synthesis from fluid simulation. *ACM TOG 29*, 3, 21:1–21:13.

MÜLLER, M., SOLENTHALER, B., KEISER, R., AND GROSS, M. 2005. Particle-based fluid-fluid interaction. In *Proc. of the 2005 ACM SIGGRAPH/Eurographics Symp. on Comput. Anim.*, 237–244.

SELLE, A., FEDKIW, R., KIM, B., LIU, Y., AND ROSSIGNAC, J. 2008. An Unconditionally Stable MacCormack Method. *J. Sci. Comp. 35*, 2, 350–371.

SONG, O.-Y., SHIN, H., AND KO, H.-S. 2005. Stable but nondissipative water. *ACM Trans. Graph.*, 81–97.

THÜREY, N., SADLO, F., SCHIRM, S., MÜLLER-FISCHER, M., AND GROSS, M. 2007. Real-time simulations of bubbles and foam within a shallow water framework. In *SCA '07: Proc. of 2007 ACM SIGGRAPH/Eurographics symposium on Computer animation*, 191–198.

VAN TERWISGA, T. J. C. Cavitation on ship propellers. `http://ocw.tudelft.nl/courses/marine-technology/cavitation-on-ship-propellers/`.

ZHENG, C., AND JAMES, D. L. 2009. Harmonic fluids. *ACM Trans. Graph. (SIGGRAPH Proc.) 28*, 3, 37:1–37:12.

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