A NONLINEAR VERTEX-BASED MODEL FOR ANIMATION OF TWO-DIMENSIONAL DRY FOAM

Micky Kelager and Kenny Erleben

eScience Center, Department of Computer Science, University of Copenhagen, Denmark

Abstract: Foam is the natural phenomenon of bubbles that arise due to nucleation of gas in liquids. The current state of art in Computer Graphics rarely includes foam effects on large scales. In this paper we introduce a vertex-based, quasi-static equilibrium model from the field of Computational Physics as a new paradigm for foam effects. Dynamic processes like gas diffusion and bubble collapse are added prior equilibration. Animation-wise the numerical model is well behaved and stable and can converge even if the foam is locally ill-defined. A novel contribution is the Ghost-Bubble method that allows foam simulations with free dynamic boundary conditions. The presented model is interesting and well suited for 2D graphics applications like video games and procedural or animated textures.

1 FOAM ANIMATION IN COMPUTER GRAPHICS

Many methods to simulate liquid fluids have been presented in Computer Graphics (Stam, 1999; Losasso et al., 2004; Selle et al., 2005; Losasso et al., 2006) with a common focus on realism. We believe that the next step in fluids dynamics lies within the subject of liquid foams, or froths. The scientific work on foam dynamics in Computer Graphics is sparse. In our opinion the most promising result to animate beer froth still has issues with the motion and behavior of the foam (Cleary et al., 2007) but the still frames from the foam simulations seem very convincing. We argue that the foam animation problems in Computer Graphics are caused by the fact that foams are treated as fluids and not as real foams. We believe that the theoretic foundation of foam dynamics must be obtained from Computational Physics. In this paper we revisit a vertex-based foam model from the field of Computational Physics and derive a mathematical model with a discretization that is suitable for the purpose of Computer Graphics.

Our focus is directed towards two-dimensional foams. This dimensional restriction is due to that physics literature commonly only agrees on the processes of two-dimensional foams and how they are behaving whereas three-dimensional foams are not yet completely understood. Two-dimensional foams also exist in the real three-dimensional world, e.g. a liquid foam constrained between two glass plates or the single layer of foam resting on a surface. Additionally, in this paper we only focus on the behavior of dry foams as wet foams are not as accurate described by current physical models (Weaire et al., 2003). We handle the dynamics and internal forces of the dry foam along with external interactive contributions, e.g. topological changes, gas diffusion, and bubble collapses, but omit external dynamic forces and collision handling. However, the method is based on a Lagrangian representation and as such additional external couplings and interactions can be handled at vertex level.

This paper introduces the Ghost-Bubble method which is a novel contribution for the two-dimensional vertex-based foam model. The Ghost-Bubble method allows dry foam simulations with free dynamic surfaces and finite boundaries. We derive a nonlinear Newton method for the dry foam model and our discrete model can converge even if the foam is locally ill-defined. The model has less than 1% total error even though we use first order finite difference approximations. The presented physics-based dry foam model is interesting and well suited for two-dimensional graphics applications like video games and procedural or animated textures.

2 PREVIOUS WORK

Computer Graphics has a greater interest in how bubbles and foam simulations are visualized while Computational Physics is more keen on trying to understand, describe, and measure all the different physics processes within a foam.

Analytic and iterative relaxation-based geometrical methods have been used in Computer Graphics to render foam and bubbles. For instance in (Icart and Arquès, 1999) empirical laws were used to construct a bubble-bubble interaction model to describe a single layer of foam bubbles. Constructive solid geometry (CSG) and analytical solutions were used in (Glassner, 2000a; Glassner, 2000b) to visualize small-sized bubble clusters of 2-3 bubbles.

Lagrangian elastic models have also been attempted. One such attempt employs an explicit curvature-driven model where bubble-bubble interactions are modeled using energy potential functions to penalize unwanted bubble configurations (Durikovic, 2001). In this work gas diffusions and Plateau borders are ignored and the computational mesh is fixed. The shown foams consist of no more than 10 bubbles. In (Li and Volkov, 2006) a 2D elastic model is extended with a pressure force model where pressure variation in the external media is included to model buoyancy. In (Iwasaki et al., 2004) a mass-spring system is employed in a curvature-driven method similar to (Durikovic, 2001). Results include visualizations of 1-2 bubbles. The main contribution in this work is a real-time rendering method that takes light interference into account.

Fluid-based methods are another paradigm. In (Hong and Kim, 2003) the volume-of-fluid method and the front-tracking method are combined. Interfaces are directly adjusted to avoid numerical diffusion and yield mass conservation. This work addresses bubbles rising in a liquid but does not deal with foams. A continuous multiphase fluid simulator is used in (Zheng et al., 2006) where a novel regional level set method and a semi-implicit surface tension model are introduced. Bubble thickness is explicit modeled by a distribution function and the effect of drainage is included. Gas diffusion is still not modeled nor is it clear how to deal with boundary conditions in the velocity field. Results show 5-10 bubbles.

Lastly bubble-bubble interaction models have also been addressed. (Kück et al., 2002) models bubbles using a mass-less spherical particle system. Springs are added to model the bubble interactions. Bubble creation and film rupture are handled by adding and removing bubbles randomly. Plateau borders are not modeled but instead special ray-tracing shaders are

used to mimic the real foam structure. Results are shown with 700-3000 bubbles. A virtual beer was poured into a glass in (Cleary et al., 2007) using a Lagrangian model of smoothed particle hydrodynamics (SPH) which makes the interaction between the multiphase fluid very convincing. Gas diffusion is handled and bubble collapsing is dealt with similar to (Kück et al., 2002). The foam model omits modeling the foam films and hence the importance of topological changes. Rheology and shearing of the bubbles are handled in a pseudo-physical way by including explicit cohesive forces between the particles. The motion of the dry foam on top of the beer is too rigid and some realism is lost in this account.

In summary the work of foams done in Computer Graphics are concerned with modeling film thickness such that the interference patterns can be shown during rendering. However, many of the dynamic foam processes are ignored, such as gas diffusion, foam rupture, and topological changes. Our method includes all of these. Lately, the work on bubbles and foams in Computer Graphics have begun integrating physically ideas (Cleary et al., 2007; Kim et al., 2007) which allow foams and bubble clusters to behave more convincingly. Common for the two-dimensional foam models in Computational Physics are the requirement of infinite networks which imply periodic boundary conditions in the numerical simulations. Distribution functions and statistics are natural interests within Computational Physics where dynamical effects come secondary (Weaire and Kermode, 1983; Kawasaki et al., 1989; Bolton and Weaire, 1992).

Our work introduces the vertex-based foam model to the field of Computer Graphics. The model includes effects not previously presented in Computer Graphics. Further we extend the model to deal with free moving foam boundaries. We demonstrate examples of dry foam networks up to 10.000 bubbles. As far as we know this is an order of magnitude larger than previous published results on 2D foam simulations in Computer Graphics.

3 A PRIMER ON PHYSICS OF 2D DRY FOAM

We now briefly review the basics of 2D dry foam. A complete source of theory can be found in (Weaire and Hutzler, 1999). A foam is a two-phase system in which bubbles of gas are enclosed by thin layers of a liquid substance. Regularly a foam is disordered which refers to the bubbles within the foam that can vary greatly in size. Ordered foams usually do not occur in real life and are therefore a topic of theoretical

research. Liquid foam has a form that is categorized as cellular due to that bubbles of gas form cells that are surrounded by liquid. In a two-dimensional foam the Plateau borders coincide with the foam films. Thus, a foam can be structured into three elements; bubbles/cells, films/Plateau borders, and Plateau junctions. In terms of mesh topology these elements correspond to faces, edges, and vertices, respectively. This apparently complex two-phase material has a well-defined local structure which allows progress in predicting the dynamic properties of the foam. The different types of foams are generally categorized into either liquid or solid. A liquid foam is a soft, highly deformable, substance that range from wet to dry dependent on how much of its volume that contains liquid. A special variant of liquid foam is froth. We concentrate on liquid foams with the same characteristics as soap foam.

A bubble will not shrink and decrease its surface area to zero due to surface tension γ is balanced against the pressure difference Δp. This relation is known as the law of Laplace-Young which is given by

$$\Delta p = \gamma \left(\frac{1}{R_1} + \frac{1}{R_2} \right) = \frac{2\gamma}{r}, \quad (1)$$

where R_1 and R_2 are the principal radii of curvature for the surface with $\Delta p = (p_i - p_j)$ as the difference between two adjacent cells i and j. In two dimensions $\frac{1}{R_1} + \frac{1}{R_2} = \frac{2}{r}$ where r is the local radius of curvature of a foam film.

Plateau added a set of additional rules to the law of Laplace-Young. These rules are vital for foam equilibrium. For a two-dimensional dry foam the law of Plateau states: *The films can intersect only three at a time and must do so with an angle of $\phi = 120°$.*

During equilibration and evolution of a dry foam the network will undergo different structural changes. These changes can be categorized into topological processes entitled T1 and T2. It has been observed that other topological restructurings in a foam are combinations of one or more T1 and T2 processes. In the two-dimensional T1 process the topological
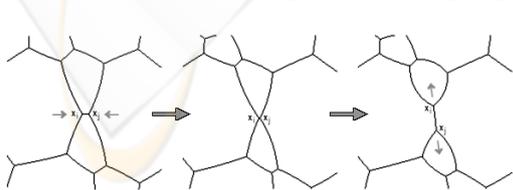
Figure 1: The T1 process on a 2D dry foam. An approaching unstable four-fold vertex will be split into two stable three-fold vertices.

change consists of exchanging, or flipping, the films

in question as depicted in Figure 1. When two three-fold vertices are moving towards each other the distance between the vertices becomes zero and the foam structure will be in an unstable configuration. If the distance between the vertices becomes close to zero, or less than some threshold T_{dist}, then a T1 process takes place,

$$\vec{d} \cdot \vec{d} < T_{\text{dist}}^2 \quad \wedge \quad T_{\text{dist}} > 0, \quad (2)$$

where $\vec{d} = (\vec{x}_i - \vec{x}_j)$ is the displacement vector between the two three-fold vertices. Generally, the length of the film is larger than $\| \vec{d} \|$ as the film is curved due to the law of Laplace-Young. We will argue that the difference in lengths can be neglected as T_{dist} can be set sufficiently close to zero.

When the T1 condition (2) is satisfied the flipping strategy we choose is to place the film vertices perpendicular to the line segment described by $\vec{d}$. If we assume that the two vertices meet Plateau's $120°$-rule then due to the geometrical properties of the T1 operation the vertices will be forced to move apart.
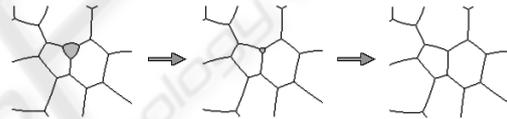
Figure 2: The T2 process for a 2D dry foam. When a three-sided cell approaches zero area due to gas diffusion it will perish.

The T2 topological change consists of removing a three-sided cell, as depicted in Figure 2, and occurs due to gas diffusion. In a T2 process a foam cell will collapse into a single vertex. The position of the resulting vertex is set to the average of the three cell vertices. The T2 process is performed when the cell area is less or equal to some area threshold T_{area},

$$n_i = 3 \quad \wedge \quad A_i \leq T_{\text{area}} \quad \wedge \quad T_{\text{area}} \geq 0, \quad (3)$$

where n_i designates the number of sides of the ith cell and A_i is the current area of the cell. Cells with more than three sides can also collapse. Here a number of T1 operations flip the smaller films in the cell until a three-sided cell is formed.

Gas diffusion is driven by the pressure differences between adjacent cells across their common boundaries. Some cells will grow at the expense of others but in the end all cells will perish. This evolution of a foam is called coarsening. By far the most common applied model for coarsening in an ideal two-dimensional dry foam is known as von Neumann's law,

$$\frac{dA_i}{dt} = \frac{2\pi}{3} \gamma \kappa (n_i - 6), \quad (4)$$

where κ is a diffusion constant. This law shows that cells with more than six sides steadily enlarge while cells with less than six sides steadily decrease in area.

4 THE VERTEX-BASED DRY FOAM MODEL

In this section we construct the mathematical equilibrium model for a two-dimensional dry foam based on the laws of foam physics. We assume that viscous forces and inertial effects are negligible. The enclosed gas is treated as incompressible such that the foam will minimize the surface energy at all times (Weaire and Kermode, 1983). For now we assume a foam with periodic boundaries which implies that each vertex has three films and three adjacent cells. Let $\mathcal{V}$, $\mathcal{E}$, $\mathcal{F}$ and $\mathcal{A}$ be index sets of vertices, edges, faces, and vertex angles, respectively. The dry foam mesh consists of faces that correspond to bubble cells. The fth cell is labeled with its current area A_f and pressure p_f, while junctions have positions where the vth vertex has coordinates $\vec{v}_v = \begin{bmatrix} x_v & y_v \end{bmatrix}^T$. We measure the angle of intersection between two films, i.e. edges, meeting at a junction, and denote these vertex angles by ϕ_a for $a \in \mathcal{A}$. From geometry and the law of Laplace-Young it follows that the cell areas and angles of film intersections can be written as functions of vertex positions and cell pressures.

The rule of Plateau results in a constraint for each vertex angle

$$\frac{2\pi}{3} - \phi_a = 0 \quad \forall a \in \mathcal{A}. \tag{5}$$

The cell areas may be controlled by a similar constraint

$$A_f^{\text{target}} - A_f = 0 \quad \forall f \in \mathcal{F}. \tag{6}$$

The target area A_f^{target} is coupled with von Neumann's law to yield the equilibrium area during coarsening of the dry foam. Next we agglomerate vertex coordinates and cell pressures into one state vector

$$\vec{x} = \begin{bmatrix} \vec{v}_1^T & \cdots & \vec{v}_{\|\mathcal{V}\|}^T & p_1 & \cdots & p_{\|\mathcal{F}\|} \end{bmatrix}^T. \tag{7}$$

The cell area and vertex angle constraints can similar be agglomerated into a system of nonlinear equations

$$\vec{F}(\vec{x}) = \begin{bmatrix} A_1^{\text{target}} - A_1(\vec{x}) \\ \vdots \\ A_{\|\mathcal{F}\|}^{\text{target}} - A_{\|\mathcal{F}\|}(\vec{x}) \\ \frac{2\pi}{3} - \phi_1(\vec{x}) \\ \vdots \\ \frac{2\pi}{3} - \phi_{\|\mathcal{A}\|}(\vec{x}) \end{bmatrix} = \vec{0}. \tag{8}$$

A solution $\vec{x}^*$ for equilibrium must fulfill $\vec{F}(\vec{x}^*) = 0$. Our idealized mathematical model may be summarized into,

$$\text{ideal model} \Leftrightarrow \begin{cases} \text{von Neumann's Law} \\ \text{Topological T1/T2 processes} \\ \text{State changes shear/collapse} \\ \vec{F}(\vec{x}) = \vec{0} \end{cases}. \tag{9}$$

This model consists of a coupling of highly nonlinear and non-smooth sub processes.

4.1 Discretization of the Mathematical Model

As the first step we apply a linearization to the whole model. Essentially, this means that we solve the four sub processes in an iterative manner until the system converges. To determine the cell target areas for the current iteration we apply an Euler discretization of von Neumann's Law. Secondly, we apply the topological processes and state changes as discrete events, and finally we solve $\vec{F}(\vec{x}) = \vec{0}$.

We apply a nonlinear Newton method for $\vec{F}(\vec{x}) = \vec{0}$ (Nocedal and Wright, 2006). Given the current iterate $\vec{x}^k$ for some iteration k for which $F(\vec{x}^k) \neq \vec{0}$, we seek to find $\Delta \vec{x}^k$ such that $F(\vec{x}^k + \Delta \vec{x}^k) = \vec{0}$. Taking a first order Taylor series approximation gives

$$\vec{0} = F(\vec{x}^k + \Delta \vec{x}^k) \approx F(\vec{x}^k) + \frac{dF(\vec{x}^k)}{d\vec{x}} \Delta \vec{x}^k. \tag{10}$$

Each iteration of the Newton method solves $J\vec{\Delta}x^k = \vec{b}$ where $J = \frac{dF(\vec{x}^k)}{d\vec{x}}$ and $\vec{b} = -F(\vec{x}^k)$. The Newton update $\vec{x}^{k+1} = \vec{x}^k + \Delta \vec{x}^k$ is done once a solution for the step-direction $\vec{\Delta}x^k$ is found. There are many practical difficulties in a straightforward implementation. An analytical solution to the Jacobian J is not feasible in practice due to the nonlinear implicit coupling between the state variables. Further, non-singularity for J is not guaranteed which implies that an inexact Newton approach must be taken.

To remedy the problems above we apply a blocked Gauss-Seidel approach to solve for the step-direction and performing the Newton update. Blocks are created from the state variables that are affected by the vertices of the foam. A block corresponds to the x and y coordinates of one vertex and the cell pressure values p_i, p_j, and p_k of the adjacent three cells that are shared by the vertex. A similar approach is used in (Kermode and Weaire, 1990). The Newton system for a single block can now be written as

$$\begin{bmatrix} \frac{\partial A_i}{\partial p_i} & \frac{\partial A_i}{\partial p_j} & \frac{\partial A_i}{\partial p_k} & \frac{\partial A_i}{\partial x} & \frac{\partial A_i}{\partial y} \\ \frac{\partial A_j}{\partial p_i} & \frac{\partial A_j}{\partial p_j} & \frac{\partial A_j}{\partial p_k} & \frac{\partial A_j}{\partial x} & \frac{\partial A_j}{\partial y} \\ \frac{\partial A_k}{\partial p_i} & \frac{\partial A_k}{\partial p_j} & \frac{\partial A_k}{\partial p_k} & \frac{\partial A_k}{\partial x} & \frac{\partial A_k}{\partial y} \\ \frac{\partial \phi_i}{\partial p_i} & \frac{\partial \phi_i}{\partial p_j} & \frac{\partial \phi_i}{\partial p_k} & \frac{\partial \phi_i}{\partial x} & \frac{\partial \phi_i}{\partial y} \\ \frac{\partial \phi_j}{\partial p_i} & \frac{\partial \phi_j}{\partial p_j} & \frac{\partial \phi_j}{\partial p_k} & \frac{\partial \phi_j}{\partial x} & \frac{\partial \phi_j}{\partial y} \end{bmatrix} \begin{bmatrix} \Delta p_i \\ \Delta p_j \\ \Delta p_k \\ \Delta x \\ \Delta y \end{bmatrix} = \begin{bmatrix} \Delta A_i \\ \Delta A_j \\ \Delta A_k \\ \Delta \phi_i \\ \Delta \phi_j \end{bmatrix}. \tag{11}$$

The right-hand side of (11) consists of the area and angle constraints

$$\Delta A_f = A_f^{\text{target}} - A_f \quad f \in \{i,j,k\} \subset \mathcal{F}, \tag{12a}$$

$$\Delta \phi_a = \frac{2\pi}{2} - \phi_a \quad a \in \{i,j\} \subset \mathcal{A}. \tag{12b}$$

The third angle ϕ_k is not part of the Jacobian because it is directly given by

$$\phi_k = 2\pi - (\phi_i + \phi_j). \tag{13}$$

The partial derivatives of the blocked Jacobian matrix are computed numerically by central difference approximations. To ensure the central difference approximations are sufficiently accurate we apply a step size-halving strategy until the absolute differences lie within a given threshold. To avoid problems with possible singularities the resulting linear system is solved using the method of Singular Value Decomposition (SVD). The topological processes are dealt with directly in the foam mesh as they correspond to edge flips and face collapses. Von Neumann's Law is discretized as

$$A_f^{target} \leftarrow A_f^{target} + \frac{2\pi}{3}\gamma\kappa(n_f - 6), \quad \forall f \in \mathcal{F}. \tag{14}$$

Since we are dealing with a quasi-static simulation we have chosen a time step size of unity.

4.2 The Ghost-Bubble Method

A foam with a freely movable surface can not have periodic boundaries. A new dynamic boundary condition must be applied to the vertex model to allow for a free foam surface.

We introduce the *Ghost-Bubble method* which adds one virtual bubble to the foam network. To make the ghost bubble move with the foam itself we add it on the back of the foam mesh. This is illustrated in Figure 3. We seek to model the ghost bubble as a natural element of the real world, i.e. a *world bubble*, that defines the atmosphere that surrounds the foam. When gas diffuses through boundary films into the atmosphere the change in volume of the world is insignificant compared to the volume of the atmosphere.
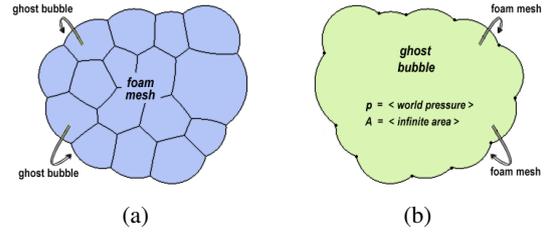
Figure 3: The ghost bubble. (a) The **front** of the foam mesh defines the foam network. (b) The ghost bubble is defined on the **back** of the foam mesh where it reuses all existing boundary vertices of the foam. It acts as a *mirrored* bubble that contains all bubbles in the foam network.

From this argument the ghost bubble should retain a constant area. The same argument holds for a constant ghost bubble pressure. This results in the following two rules; the ghost bubble must have constant, infinite area and constant, unit pressure.

A permutation of indices is performed prior setting up (11) for a boundary block such that the ghost bubble corresponds to the cell with index k. From our rules we have $\Delta A_k = \Delta p_k = 0$ and the changes to the Newton subsystem become,

$$\begin{bmatrix} \frac{\partial A_i}{\partial p_i} & \frac{\partial A_i}{\partial p_j} & 0 & \frac{\partial A_i}{\partial x} & \frac{\partial A_i}{\partial y} \\ \frac{\partial A_j}{\partial p_i} & \frac{\partial A_j}{\partial p_j} & 0 & \frac{\partial A_j}{\partial x} & \frac{\partial A_j}{\partial y} \\ 0 & 0 & 0 & 0 & 0 \\ \frac{\partial \phi_i}{\partial p_i} & \frac{\partial \phi_i}{\partial p_j} & 0 & \frac{\partial \phi_i}{\partial x} & \frac{\partial \phi_i}{\partial y} \\ \frac{\partial \phi_j}{\partial p_i} & \frac{\partial \phi_j}{\partial p_j} & 0 & \frac{\partial \phi_j}{\partial x} & \frac{\partial \phi_j}{\partial y} \end{bmatrix} \begin{bmatrix} \Delta p_i \\ \Delta p_j \\ 0 \\ \Delta x \\ \Delta y \end{bmatrix} = \begin{bmatrix} \Delta A_i \\ \Delta A_j \\ 0 \\ \Delta \phi_i \\ \Delta \phi_j \end{bmatrix}. \tag{15}$$

5 EXPERIMENTS AND RESULTS

We post-process Voronoi diagrams from random generated point sites to create initial networks for our foam solver. This is similar to (Weaire and Kermode, 1984) and illustrated in Figure 4.
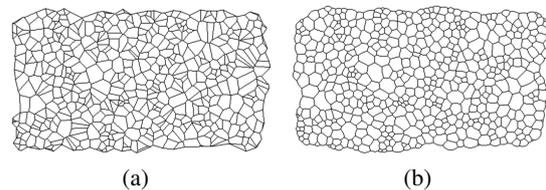
Figure 4: In (a) a Voronoi diagram of 500 cells is generated from random point sites and (b) shows the corresponding equilibrated 2D dry foam network with an area error of 0.001% and an angle error 0.01%.

Our experiments indicate that good threshold val-

ues for the topological conditions can be chosen as

$$0.01 \leq \{T_{\text{dist}}, T_{\text{area}}\} \leq 0.25. \qquad (16)$$

We use the surface tension of $\gamma \approx 0.00025 \frac{N}{m}$. We have evaluated the presented foam model by running hundreds of simulations with different numbers of cells and area sizes without any technical instabilities. We have varied κ from $[10^{-3}..10^3]$ and when a high value of κ is employed the foam is collapsing more rapidly similar to plain water foam.

We have measured the computation times from different foam configurations and plotted them in Figure 5. The plot depicts that the computation time appears to scale linear in the number of cells. Figures 4(b), 6, and 7 illustrate the free surface bound-
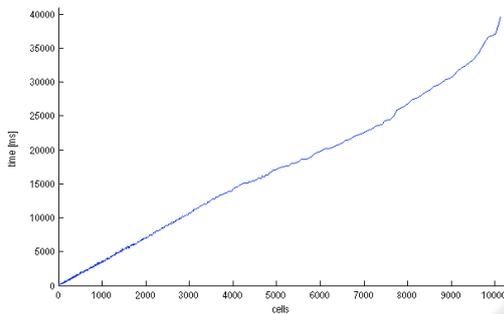
Figure 5: Time performance plot for a foam network evolving over time. The initial number of cells is 10k with a frame time of 39.5s. The last frame contains 11 cells and took 84ms. The simulation ran on a standard single core 2GHz laptop.

aries that is a result of the Ghost-Bubble method.

Figure 4 establishes that the dry foam model can equilibrate a newly created foam. Four frames from a complete foam evolution with initially $10,000$ cells are shown in Figure 6.

To illustrate the foam bubble collapse mechanism a random foam mesh of 66 cells have been used. The dynamic gas diffusion process has been deactivated to explicitly take control over which bubbles that should disappear. T2 operations are forced upon the selected bubbles. The result of a sequence of bubble collapses is shown in Figure 7 where cells of 4, 5, and 6 edges have been targeting for deletion.

The foam mesh can be rendered using flat shaded polygons. Figure 8 illustrates a large example of how the pressure differences are distributed throughout a foam. Notice that cells with similar pressures tend to group together and form walls that almost makes the image look like a labyrinth.
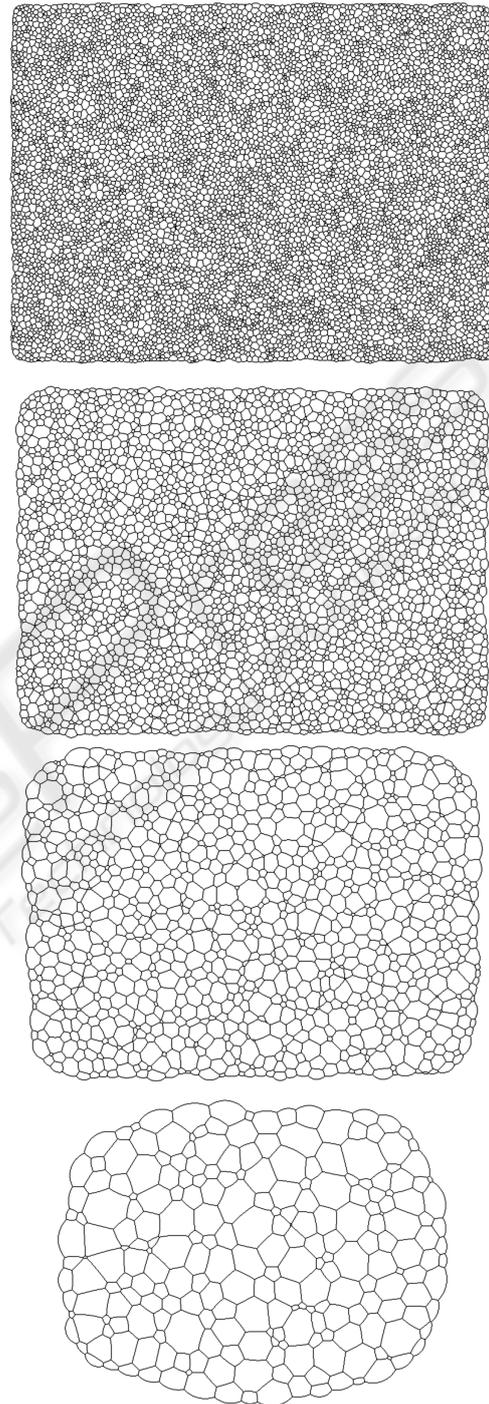
Figure 6: A foam is evolving with initially $10,000$ cells. The time step is squared between each shown frame.
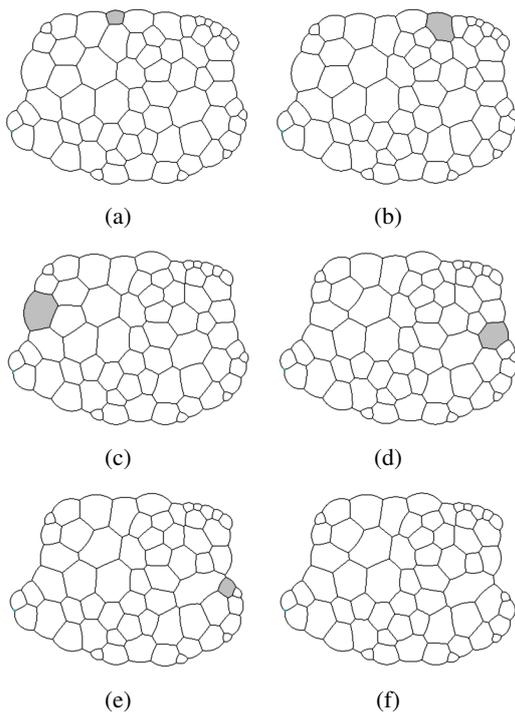
Figure 7: Forced bubble collapse is performed on the dynamic foam boundary. Between each sub figure the marked cell is next to be collapsed.

Figure 8: The cell pressures are visualized in flat shaded blue tones. The colors have been exaggerated to make the difference more visible.

6 CONCLUSIONS AND FUTURE WORK

We have introduced a new paradigm for foam simulation from the field of Computational Physics. Further, we have presented a novel method to deal with free moving boundaries in quasi-static simulation of 2D dry foam. Our discrete model has been developed with focus on stability and robustness that are needed in the field of Computer Graphics. We believe that the presented physics-based dry foam model is interesting and well suited for 2D graphics applications like video games and procedural or animated textures.

For future work we will add dynamic forces to the foam model. From a performance point of view we could replace the sequential blocked Gauss-Seidel variant of the nonlinear Newton solver with a parallel blocked red-black Gauss-Seidel variant. We may also exploit that a dual mesh could encode the geometric invariants of the rules of Plateau. This work may be extended to 3D where tetrahedra would correspond to film junctions. However, it is not clear how to deal with the topological processes in 3D, nor is it clear how to reconstruct the curved surfaces of the film geometry in 3D.

REFERENCES

Bolton, F. and Weaire, D. (1992). The effects of plateau borders in the two-dimensional soap froth. II. general simulation and analysis of rigidity loss transition. *Philosophical Magazine Part B*, 65(3):473–487.

Cleary, P. W., Pyo, S. H., Prakash, M., and Koo, B. K. (2007). Bubbling and frothing liquids. *ACM Transactions on Graphics*, 26(3):97–1–97–6.

Durikovic, R. (2001). Animation of soap bubble dynamics, cluster formation and collision. *Comput. Graph. Forum*, 20(3).

Glassner, A. (2000a). Soap bubbles: Part 1. *IEEE Comput. Graph. Appl.*, 20(5):76–84.

Glassner, A. (2000b). Soap bubbles: Part 2. *IEEE Comput. Graph. Appl.*, 20(6):99–109.

Hong, J.-M. and Kim, C.-H. (2003). Animation of bubbles in liquid. *Comput. Graph. Forum*, 22(3):253–262.

Icart, I. and Arquès, D. (1999). An approach to geometrical and optical simulation of soap froth. *Computers & Graphics*, 23(3):405–418.

Iwasaki, K., Matsuzawa, K., and Nishita, T. (2004). Real-time rendering of soap bubbles taking into account light interference. In *CGI '04: Proceedings of the*

Computer Graphics International (CGI'04), pages 344–348, Washington, DC, USA. IEEE Computer Society.

Kawasaki, K., Nagai, T., and Nakashima, K. (1989). Vertex models for two-dimensional grain growth. *Philosophical Magazine Part B*, 60(3):399–421.

Kermode, J. P. and Weaire, D. (1990). 2D-FROTH: a program for the investigation of 2-dimensional froths. *Computer Physics Communications*, 60(1):75–109.

Kim, B., Liu, Y., Llamas, I., Jiao, X., and Rossignac, J. (2007). Simulation of bubbles in foam with the volume control method. In *SIGGRAPH '07: ACM SIGGRAPH 2007 papers*, page 98, New York, NY, USA. ACM.

Kück, H., Vogelgsang, C., and Greiner, G. (2002). Simulation and Rendering of Liquid Foams. In *Proc. Graphics Interface*, pages 81–88.

Li, L. and Volkov, V. (2006). Inflatable models. *J. Comput. Sci. Technol.*, 21(2):154–158.

Losasso, F., Gibou, F., and Fedkiw, R. (2004). Simulating water and smoke with an octree data structure. *SIGGRAPH 2004, ACM TOG*, 23:457–462.

Losasso, F., Shinar, T., Selle, A., and Fedkiw, R. (2006). Multiple interacting liquids. *SIGGRAPH 2006, ACM TOG*, 25:812–819.

Nocedal, J. and Wright, S. J. (2006). *Numerical Optimization*. Springer, second edition.

Selle, A., Rasmussen, N., and Fedkiw, R. (2005). A vortex particle method for smoke, water and explosions. *SIGGRAPH 2005, ACM TOG*, 24:910–914.

Stam, J. (1999). Stable fluids. In *SIGGRAPH 1999 Conference Proceedings*, pages 121–128.

Weaire, D. and Hutzler, S. (1999). *The Physics of Foams*. Oxford University Press, New York.

Weaire, D., Hutzler, S., Cox, S., Kern, N., Alonso, M. D., and Drenckhan, W. (2003). The fluid dynamics of foams. *Journal of Physics: Condensed Matter*, 15(1):S65–S73.

Weaire, D. and Kermode, J. P. (1983). Computer simulation of a two-dimensional soap froth - i. method and motivation. *Philosophical Magazine Part B*, 48(3):245–259.

Weaire, D. and Kermode, J. P. (1984). Computer simulation of a two-dimensional soap froth - ii. analysis of results. *Philosophical Magazine Part B*, 50(3):379–395.

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*, pages 325–333, Aire-la-Ville, Switzerland, Switzerland. Eurographics Association.