A Hyperbolic Geometric Flow for Evolving Films and Foams

SADASHIGE ISHIDA, Nikon Corporation and The University of Tokyo
MASAFUMI YAMAMOTO, The University of Tokyo
RYOICHI ANDO, National Institute of Informatics
TOSHIYA HACHISUKA, The University of Tokyo

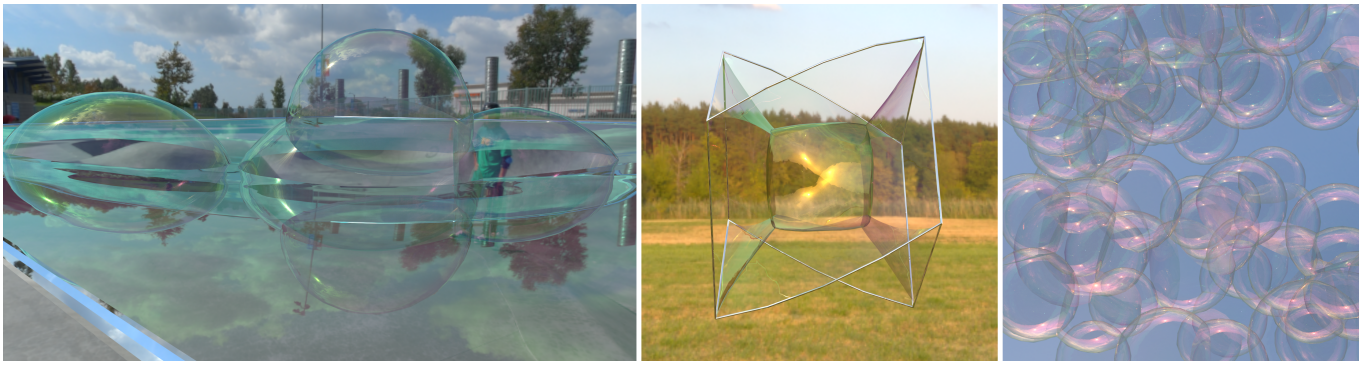

Fig. 1. Various configurations of soap films produced by our method; falling bubbles on a film (left), films spanning a twisted wire (center), and a foam (right). Our geometric formulation robustly handles complex structures of films and foams such as non-manifold surfaces with topology changes.

Simulating the behavior of soap films and foams is a challenging task. A direct numerical simulation of films and foams via the Navier-Stokes equations is still computationally too expensive. We propose an alternative formulation inspired by geometric flow. Our model exploits the fact, according to Plateau's laws, that the steady state of a film is a union of constant mean curvature surfaces and minimal surfaces. Such surfaces are also well known as the steady state solutions of certain curvature flows. We show a link between the Navier-Stokes equations and a recent variant of mean curvature flow, called *hyperbolic mean curvature flow*, under the assumption of constant air pressure per enclosed region. Instead of using hyperbolic mean curvature flow as is, we propose to replace curvature by the gradient of the surface area functional. This formulation enables us to robustly handle non-manifold configurations; such junctions connecting multiple films are intractable with the traditional formulation using curvature. We also add explicit volume preservation to hyperbolic mean curvature flow, which in fact corresponds to the pressure term of the Navier-Stokes equations. Our method is simple, fast, robust, and consistent with Plateau's laws, which are all due to our reformulation of film dynamics as a geometric flow.

CCS Concepts: • **Computing methodologies** → *Animation*; Physical simulation; *Geometry*;

Additional Key Words and Phrases: fluid simulation, soap film, geometric flow, Plateau's problem

This work is supported by Nikon Corporation, under the collaborative research project with the University of Tokyo.

0730-0301/2017/11-ART199 $15.00
https://doi.org/10.1145/3130800.3130835

ACM Reference format:
Sadashige Ishida, Masafumi Yamamoto, Ryoichi Ando, and Toshiya Hachisuka. 2017. A Hyperbolic Geometric Flow for Evolving Films and Foams. *ACM Trans. Graph.* 36, 6, Article 199 (November 2017), 11 pages.
https://doi.org/10.1145/3130800.3130835

1 INTRODUCTION

Ephemeral yet complex shapes of films and foams have fascinated many people over the years. The steady-state shape of soap films has long been a subject of study in differential geometry [Powers et al. 2002; Sullivan and Morgan 1996; Taylor 1976]. The pioneering study by Plateau resulted in a set of laws that describe the steady-state shape called Plateau's laws. Named after him, Plateau's problem [Rado 1930] relates the steady-state shape with area-minimizing surfaces given fixed boundaries. One of the first Fields Medals was awarded to the study [Douglas 1931] of Plateau's problem, and it remains an active topic of study in mathematics even today [Ambrosio 2015; Harrison 2014; Harrison and Pugh 2015].

In addition to mathematical studies, computer simulation of films and foams has also been a challenging problem. A commonly used numerical solver for the Navier-Stokes equations can be computationally very expensive since it involves a simulation of multi-fluid phenomena with extremely thin geometry deforming with surface tension [Saye and Sethian 2013]. Eulerian approaches are particularly not feasible since discretization may overlook such thin features of films. Recent works [Durikovic 2001; Zhu et al. 2014] thus employ Lagrangian approaches and explicitly track surfaces of films. Da et al. [2015] formulated film dynamics as surface-only simulation of films using vortex sheets. Nonetheless, robust and visually pleasing simulation of films is still challenging.

We propose a new formulation of film dynamics that unifies a geometric view of films in mathematics into physics simulation of films. Based on the study in mathematics, we formulate film dynamics as evolving area-minimizing surfaces. We observe that film dynamics is closely related to a recently studied geometric flow called *hyperbolic mean curvature flow* [Dexing et al. 2009; He et al. 2009; LeFloch and Smoczyk 2008]. Hyperbolic mean curvature flow is a variant of classical mean curvature flow. Each point on an evolving surface under hyperbolic mean curvature flow is accelerated in the direction of its mean curvature normal. We modify this hyperbolic mean curvature flow to be volume preserving, and show that it is, in fact, equivalent to a simplified form of the Navier-Stokes equations for film dynamics under some plausible assumptions.

While it is tempting to adopt existing methods for mean curvature flow to solve our model, junctions where multiple films meet, namely Plateau borders, become intractable since mean curvature is undefined. We thus propose to replace mean curvature by the gradient of surface-area functional which is identical to mean curvature in the absence of such borders. While the proof of this equivalence has been known in differential geometry, we are the first to utilize this fact to enable robust handling of otherwise intractable configurations in geometric flow. Since our formulation directly minimizes surface area, the steady-state solutions of our model conform well with Plateau's laws. To summarize, our technical contributions are as follows:

- Introduction of volume preserving hyperbolic mean curvature flow.
- A variational approach for surface area minimization and a multiregional volume preserving technique, allowing us to treat multiple films in a unified manner.
- Fast, simple to implement, and accurate numerical solver for film dynamics which is also consistent with Plateau's laws.

We provide various experiments to analyze the properties of our formulation. Figure 1 shows some of our results. Our work enables an accurate simulation of films and foams that outperforms previous approaches without additional complexity or computational cost.

2 RELATED WORKS

2.1 Animation of Soap Films

For years, visual simulation of films and foams has been extensively studied in computer graphics. For tiny bubbles, researchers have investigated various Eulerian–particle hybrid approaches since they are small enough to be well represented by a collection of particles. The seminal work of Hong et al. [2008] employs the particle level set method and incorporates escaped particles as bubble particles to simulate underwater bubbles. Busaryev et al. [2012] proposed to use a volume-preserving weighted Voronoi diagram to approximate the geometry of foams. Kim et al. [2007] successfully simulated centimeter-scale bubbles and enabled explicit volume control by extending the regional level set method [Zheng et al. 2006]. Unlike these works, we focus on cases where deformation of individual bubble surface is visible.

A more natural representation for thin films would be explicit surface meshes. Da et al. [2015] proposed a novel vortex sheet model for surface-only film simulation, where a scalar circulation quantity is attached to surface meshes to drive the whole motion. In such a surface-based simulation, handling merging, and splitting of film surfaces is a non-trivial issue by itself. Durikovic [2001] introduced a numerical technique to handle such phenomena and Zhu et al. [2014] further improved this tracker, making an efficient representation of geometrically complex simulation possible. Given the success of these methods, we also evolve a mesh-based on surface-only film simulation. We, however, propose a different formulation using geometric flow.

2.2 Geometry and Flows

Geometry of films has been a long-lasting subject of interest among researchers across a broad range of scientific fields including mathematics [Almgren and Taylor 1976] and engineering [Brew and Lewis 2003]. For instance, mathematicians examined the steady states of soap bubbles, which exhibit different visually aesthetic structures depending on their initial configurations [Struwe 2014]. Such structures are driven by the area-minimization due to the surface tension force [Boys 1958], and they are successfully applied for designing architectures [Argyris et al. 1974]. We refer interested readers to a book [Isenberg 1978] for an overview of the dynamics of soap films. Our formulation is inspired by mathematical studies of films and foams, especially those works on Plateau's problem.

Plateau's Problem. The shape of a soap film converges to a local minimum of the area functional. The steady state is conjectured to satisfy the geometric configurations described by empirical rules called *Plateau's laws*. Plateau's problem mathematically formulates this property as a minimization problem of surface area for given boundaries, enclosed volumes, and connection of regions. Since Plateau's problem is an important subject in many fields, it has been studied over 200 years since Lagrange introduced it in 1760. Hutchings [2002] determined the area minimized configuration of two connected bubbles often called "a double bubble", and proved that it follows Plateau's laws. Cases for more than two bubbles are still unsolved.

Plateau's problem is difficult to solve even numerically. Researchers have investigated various methods to handle specific scenarios of respective configurations. Using mean curvature flow, Pinkall [1993] and Polthier [2002] successfully computed discrete minimal surfaces. Variants of a multiple regional level set method [Merriman et al. 1994; Mohammad and Švadlenka 2002; Švadlenka et al. 2014] addressed closed regions without open boundaries. There are some tools [Brakke 1992; Pan et al. 2012] to investigate area-minimized shapes in general settings by heuristically deforming surfaces, however, they do not guarantee convergence to the local minimum. All of the above methods utilize the mean curvature flow or its variations and our method shares the same spirit.

Simply using a curvature flow, however, does not fully solve Plateau's problem in general settings with constraints of enclosed volumes. We analyze the main difficulties as follows. First, curvature flows are applicable only for a single surface since the surface normal or the curvature is not definable for the intersection of surfaces called non-manifold junctions. Second, volume preserving flows can be defined only for a single closed surface. Regional level set

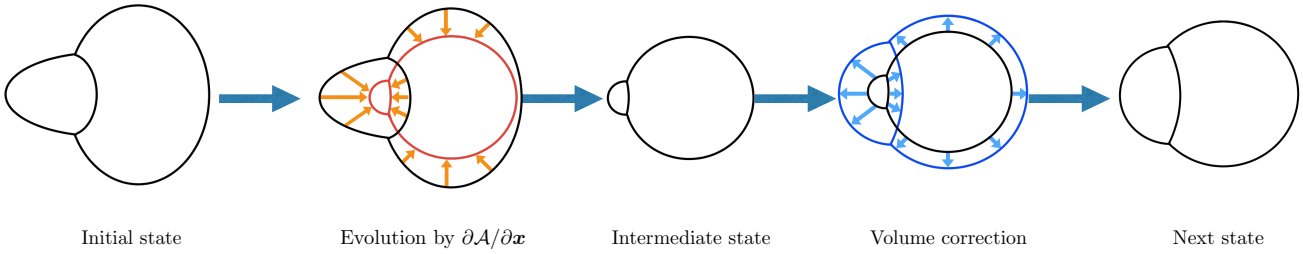

Fig. 2. Overview of a time step. We first evolve the films by applying force on the points toward the negative directions of $\partial\mathcal{A}/\partial\boldsymbol{x}$. The volumes of closed regions may decrease at this point. We next correct the volumes by pushing points toward the normals. The correction amount of each boundary film is related with the pressure difference through the film.

methods may preserve volumes but cannot handle open surfaces. We propose an equation and a computational algorithm that naturally handles non-manifold junctions and preserves the enclosed volumes of multiple regions. Seen as a geometry processing tool, our method numerically solves Plateau's problem for an arbitrary union of closed and open surfaces.

Hyperbolic mean curvature flow. Recently, hyperbolic variant of mean curvature flow named hyperbolic mean curvature flow is introduced in differential geometry [LeFloch and Smoczyk 2008]. It directly models a vibrating membrane whose acceleration is given only by mean curvature. We introduce hyperbolic mean curvature flow to computer graphics to simulate evolving films and foams. Since original hyperbolic mean curvature flow cannot be defined for non-manifold junctions and does not preserve the inner volumes of closed regions, we extend it to resolve these issues.

Curvature flows for fluid dynamics. Recently, curvature flows are used in computer animation to describe phenomena caused by strong surface tension. Thürey et al. [2010] used a volume preserving flow to express complex behaviors of fluid surfaces such as crown splash. Zhang et al. [2012] employed mean curvature flow for a surface-only simulation of small water droplets. Evolving a surface toward the negative gradient of the area functional reduces the surface area, which is equivalent to evolving under mean curvature flow. Misztal et al. [2014] utilized the property for simulating multiphase flow of immiscible fluids. These methods assume that fluid surfaces do not have non-manifold junctions. Our model also accounts for surface tension force via a geometric flow, and it extends to non-manifold geometry.

3 OVERVIEW

Our key idea is to formulate dynamics of soap films using hyperbolic mean curvature flow. Hyperbolic mean curvature flow is defined as

$$\frac{d^2\boldsymbol{x}}{dt^2} = -\beta H(\boldsymbol{x},t)\boldsymbol{n}(\boldsymbol{x},t), \tag{1}$$

where d/dt, $\boldsymbol{x}$, β, H, and $\boldsymbol{n}$ denote total derivative, position, a constant, mean curvature, and the surface normal, respectively. Similar to mean curvature flow, the stationary solutions of this hyperbolic variant are still minimal surfaces. This property is consistent with the fact that the steady state of a soap film without trapped air is a minimal surface. Hyperbolic mean curvature flow is also known to model an elastic membrane moving under the surface tension force which closely resembles soap films [Yau 2000].

Hyperbolic mean curvature flow, however, has been studied mainly in differential geometry as a geometric flow and its application to animation of soap films is unclear. Moreover, Equation (1) is not directly applicable to film dynamics since the mean curvature cannot be defined on Plateau borders.

In order to handle non-manifold surfaces, we propose to replace the mean curvature term with the variational derivative of the area functional with respect to position denoted by $\partial\mathcal{A}(\boldsymbol{x})/\partial\boldsymbol{x}$. This derivative is known to be equivalent to the mean curvature normal of smooth surfaces, and it naturally extends to non-manifold junctions such as Plateau borders. To account for incompressible air trapped by soap films, we include the pressure term $\Delta p\boldsymbol{n}$. Our final model thus becomes

$$\boxed{\frac{d^2\boldsymbol{x}}{dt^2} = -\beta\frac{\partial\mathcal{A}(\boldsymbol{x})}{\partial\boldsymbol{x}} + \Delta p\boldsymbol{n}} \tag{2}$$

where the constant β is equal to twice the surface tension coefficient as we show later. The first term on the right-hand side of Equation (2) no longer poses complications associated with non-manifold structures since area is measurable regardless of the topology. As we show later, our model can, in fact, be derived from the Navier-Stokes equations, under the assumption of constant pressure per bubble and the atmosphere. We note that this assumption is commonly adopted for simulation of bubbles[Busaryev et al. 2012; Patkar et al. 2013; Zhu et al. 2014].

In our algorithm, we first numerically evaluate the area derivative to evolve the surface meshes. Next, we assume that pressure is well approximated by a constant per trapped region [Patkar et al. 2013; Zhu et al. 2014]. Under this assumption, the pressure difference acts only between adjacent regions, conveniently allowing us to perform volume correction by only solving a small and sparse linear system of which the degrees of freedom is the number of the enclosed regions. Finally, we apply a symplectic integrator for time evolution. Figure 2 illustrates the overall steps of our algorithm.

4 GEOMETRIC FLOW OF SOAP FILMS

We now explain our geometric flow for film dynamics. In this paper, a surface is a continuous map $U \to \mathbb{R}^3$ where U is a connected

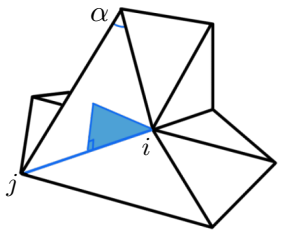

Fig. 3. Illustration of the extended cotangent matrix. We sum up the contribution of all the Voronoi regions of the faces incident to the edges connecting vertices i and j. This figure shows one of the three Voronoi regions.

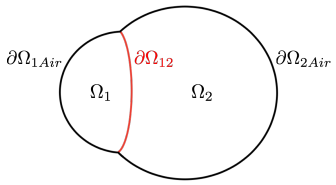

Fig. 4. Regions of a double bubble. We correct the volume losses of the closed domains Ω_1, Ω_2 by moving points of the boundaries $\partial\Omega_{12}, \partial\Omega_{1Air}, \partial\Omega_{2Air}$ toward the normals. We determine the correction amount for each boundary by solving two equations, $\Delta V_i = Area(i, Air)d_i + Area(i, j)(d_i - d_j)$ for $(i, j) = (1, 2)$ and $(i, j) = (2, 1)$.

subset of $\mathbb{R}^2$. Films consist of manifold and non-manifold geometries. A manifold point is a point that belongs only to a single surface. A non-manifold junction is a point that belongs to two or more surfaces. Especially, a triple-junction is an intersection of three surfaces, called a Plateau border for soap films. A quad-junction is an intersection of four surfaces. Almost every quad-junction appearing in soap film dynamics is a point where four Plateau borders meet.

A constant mean curvature (CMC) surface is a surface with the same mean curvature everywhere. A minimal surface is a special case of CMC surfaces with zero mean curvature. A minimal surface locally minimizes its area, that is, any infinitesimal smooth change of the surface increases the area. According to Plateau's laws, the steady state of a soap film is a union of CMC and minimal surfaces. For example, a double bubble with different pressures consists of three CMC surfaces. The steady state of a film between two rings is a minimal surface also known as a catenoid. In general, a film portion with different air pressure on each side becomes a CMC surface and equal pressure on both sides leads to a minimal surface. Our model leverages this geometric relationship between soap films and CMC/minimal surfaces for animation of films.

4.1 Variational Derivative of the Area Functional

It is well known that mean curvature is undefined on discrete surfaces since differentiation needs a continuous setting. Discrete differential geometry circumvents this limitation by discretizing mean curvature normal $H\boldsymbol{n}$ as differentiation of the area functional, based on the fact that $H\boldsymbol{n} = \partial\mathcal{A}/\partial\boldsymbol{x}$ on a smooth surface (see Appendix A). We found that this equality $H\boldsymbol{n} = \partial\mathcal{A}/\partial\boldsymbol{x}$ naturally extends to even non-manifold junctions, making our geometric flow definable regardless of configurations.

Both $H\boldsymbol{n}$ and $\partial\mathcal{A}/\partial\boldsymbol{x}$ have the following properties in common: for an infinitesimal smooth change of the surface, the change along its direction gives the maximal area change, and the magnitude indicates how different the current shape is from the area-minimizing configuration. By acting force on each point toward the negative direction of $\partial\mathcal{A}/\partial\boldsymbol{x}$, we can locally minimize the surface area. Therefore, if Plateau's laws as conjectures describing the geometric conditions of the local minima of the area functional are true, our formulation is automatically consistent with Plateau's laws.

For a polygon mesh, we discretize $\partial\mathcal{A}/\partial\boldsymbol{x}$ as follows. It is known that mean curvature normal is discretized as $H_N = M^{-1}LX$ where matrices H_N and X encode mean curvature normals and positions of the vertices, M is the mass matrix, and L is the cotangent matrix [Desbrun et al. 1999; Jacobson et al. 2013; Meyer et al. 2002]. Note that L is undefined for non-manifold vertices. Replacing L by our extended cotangent matrix L', we can define $\partial\mathcal{A}/\partial\boldsymbol{x} = M^{-1}L'X$. Each entry of L' is defined by,

$$L'_{ij} = \begin{cases} \sum_f \frac{1}{4A_{voronoi}} \cot(\alpha) & \text{for } j \in N(i) \\ -\sum_{k\neq i} L_{ik} & \text{for } j = i \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

where $N(i)$ is the set of neighborhood vertices of i-th vertex and f loops over faces incident to the edge connecting $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$, and α is the opposite angle of the edge in f, and $A_{voronoi}$ is the area of Voronoi region (Figure 3). It is a natural extension since L'_{ij} coincides with L_{ij} if the edge connecting vertices i and j is manifold. Our extended cotangent matrix is sparse similarly to the original definition.

4.2 Volume Preservation

Since hyperbolic mean curvature flow is not a volume-preserving flow, volume loss occurs at each time step, which is incompatible with soap film dynamics. We thus introduce the volume preserving term $\Delta p\boldsymbol{n}$ which, as we show later, is equivalent to the pressure difference through the film with the assumption of incompressibility and constant pressure per region.

Instead of explicitly solving for the pressure term, we correct each enclosed volume by extending Müller's technique [2009] for multiple regions. In the original technique, the volume loss ΔV of an enclosed region is corrected by moving each point on a surface along its normal direction. The amount of correction per point is given as a constant $d = \Delta V/A$ where A is the area of the region.

We extend this technique to handle multiple connected regions by simultaneously preserving all enclosed volumes. In our setting, the amount of correction for each boundary denoted by Δd relates to the pressure difference of the incident regions. We first solve a linear system $\boldsymbol{A}\boldsymbol{d} = \Delta\boldsymbol{V}$ where $\boldsymbol{d}$ is a vector of coefficients related to the inner pressures, and $\Delta\boldsymbol{V} = (V_0^{init} - V_0, \ldots, V_K^{init} - V_K)$ is a *vector* of volume losses of K enclosed regions. The matrix $\boldsymbol{A}$ is

$$A_{ij} = \begin{cases} -Area(i,j) & \text{(regions } i \text{ and } j \text{ are adjacent)} \\ Area(i, air) + \sum_k Area(i,k) & (j = i) \\ 0 & \text{(otherwise)} \end{cases} \tag{4}$$

where $Area(i,j)$ is the area of the surface of the boundary domain between i-th and j-th closed regions, and $Area(i,air)$ is the area of the boundary domain between i-th closed region and the external fluid such as air. Each area is computed as $Area(i,j) = \int_{\partial\Omega_{ij}} ds$ for each boundary surface $\partial\Omega_{ij}$. Note that $\boldsymbol{A}$ is always invertible.

Figure 4 shows an example of the boundary domains for two connected bubbles. The amount of correction at each point is determined by $\Delta d = d(region_l) - d(region_s)$ where $region_l$ and $region_s$ are two regions that the point belongs to. For non-manifold junctions, we take the average of $\Delta d\boldsymbol{n}$ over its neighborhoods. For regions of external air, we use $d(air) = 0$. Appendix B gives a proof that this extension corrects all the enclosed volumes.

The volume correction happens after moving vertices by the acceleration $d^2\boldsymbol{x}/dt^2 = -\beta\,\partial\mathcal{A}/\partial\boldsymbol{x}$. This operation is analogous to computing the pressure term $\Delta p\boldsymbol{n}$ with the first-order approximation for time.

4.3 Connection to Physics

Although Equation (2) appears to lack any physical interpretation, it can be derived from the inviscid Navier-Stokes equations (Euler equations) assuming that the film is infinitesimally thin. The Euler equations with the surface tension force are

$$\frac{D\boldsymbol{u}}{Dt} = -\frac{\sigma H\delta(\boldsymbol{x})}{\rho}\boldsymbol{n} + \frac{1}{\rho}\nabla p, \tag{5}$$

where D/Dt, $\boldsymbol{u}$, ρ, σ, H, $\boldsymbol{n}$, and p denote material derivative, the velocity, fluid density, the surface tension strength, mean curvature, surface normal, and pressure, respectively. The term $\delta(\boldsymbol{x})$ is the Dirac delta function that acts only on the liquid interfaces, essentially encoding the pressure discontinuity.

Integrating both sides of Equation (5) across a thin film gives

$$\int_{\Omega_R} \frac{D\boldsymbol{u}}{Dt} dV = \int_{\Omega_R} \Big(-\frac{\sigma H\delta(\boldsymbol{x})}{\rho}\boldsymbol{n} + \frac{1}{\rho}\nabla p\Big) dV, \tag{6}$$

where Ω_R denotes a small domain crossing the interfaces. The left-hand side of the equation simply encodes the temporal change of the velocity of the film crossing with the domain. Working with the Dirac function on the right-hand side yields

$$\int_{\Omega_R} \Big(-\frac{\sigma H\delta(\boldsymbol{x})}{\rho}\boldsymbol{n} + \frac{1}{\rho}\nabla p\Big) dV = -\frac{2\sigma H\boldsymbol{n}}{\rho} + \int_{\Omega_R} \frac{1}{\rho}\nabla p dV. \tag{7}$$

We have doubled the mean curvature term since the Dirac function exists twice (e.g., $\delta(\frac{1}{2}\varepsilon)$ and $\delta(-\frac{1}{2}\varepsilon)$) where ε denotes the thickness of the film) on air-liquid and liquid-air interfaces.

Assuming constant pressure for air, meaning that the pressure gradient only exists within the film, the gradient theorem applied to the second term of the above equation yields

$$\int_{\Omega_R} \frac{1}{\rho}\nabla p dV = \frac{1}{\rho}\oint_{\partial\Omega_R} p\boldsymbol{n} dS = \frac{p_{\text{front}} - p_{\text{back}}}{\rho}\boldsymbol{n} = \frac{1}{\rho}\Delta p\boldsymbol{n}, \tag{8}$$

where $\partial\Omega_R$, p_{front}, p_{back}, and $\boldsymbol{n}$ denote an infinitesimal cylinder across the boundary surface, the pressure at the front side, the pressure at the back side, and the surface normal of the film, respectively.

In the limit of infinitesimal Ω_R, the integration domain Ω_R approaches a point, allowing us to drop the integral from the left-hand side of Equation (6). By using the relation $D\boldsymbol{u}/Dt = d^2\boldsymbol{x}/dt^2$, letting $\rho = 1$, and defining $\beta = 2\sigma$ for simplicity, we arrive at Equation (2).

ALGORITHM 1: Time integration

(1) Move constrained vertices.
(2) Set intermediate velocities of free particles.
$\boldsymbol{u}' = \boldsymbol{u} + \Delta t(-\beta\,\partial\mathcal{A}/\partial\boldsymbol{x} + \boldsymbol{f}_{ext})$.
(3) Evolve free particles to intermediate positions.
$\boldsymbol{x}' = \boldsymbol{x} + \Delta t\boldsymbol{u}'$.
(4) Update positions by performing volume preservation,
$\boldsymbol{x} = \boldsymbol{x}' + \Delta d\boldsymbol{n}$.
(5) Update velocities.
$\boldsymbol{u} = \boldsymbol{u}' + (\Delta d/\Delta t)\boldsymbol{n}$.
(6) Update the polygon mesh including topology change by LosTopos.
(7) Determine the velocities of newly generated vertices by barycentric interpolation.

5 IMPLEMENTATION

5.1 Spatial Discretization

We use a triangle mesh to discretize films as in the previous work [Da et al. 2015]. To handle topology changes, we use LosTopos package [Da et al. 2014], which is currently the only surface tracker that can handle multi-regional surfaces with non-manifold junctions. Since LosTopos requires regions to be closed and any triangle to be incident to two different regions, we use ghost vertices and triangles that are out of the simulation. For a scene with separate open surfaces, we index $0, 1, 2, \ldots, M$ for open regions and $M+1, M+2, \ldots$ for closed regions. We re-order the orientation of triangles such that their normal vectors are always pointing out towards the smaller number assigned to each region of air.

5.2 Time Integration

One time step of our time integration scheme is illustrated in Algorithm 1. We take special care to compute the volume correction vectors $\Delta d\boldsymbol{n}$ for non-manifold junctions where normals are undefined. For triple-junctions, we average $\Delta d\boldsymbol{n}$ of the incident manifold vertices. For quad-junctions, we average $\Delta d\boldsymbol{n}$ of the incident triple-junctions. For junctions more than quad, we handle them in the same way as quad-junctions. We, however, have never observed such a junction in our simulation.

Overall, the total computational complexity is $O(N_T) + O(N_R) + O(N_V)$, where N_V, N_T, and N_R are the numbers of vertices, triangles, and regions respectively. Each notation is defined as follows: $O(N_T)$ refers to the cost of computing the volumes and the surface areas, $O(N_R)$ cost for solving a sparse linear system, and $O(N_V)$ for correcting vertices' positions. We found that $O(N_R)$ scales close to proportional to the number of non-zero entries in the matrix.

We apply the first-order symplectic Euler method. We perform the integration by computing the intermediate velocity $\boldsymbol{u}'$ and use it to integrate the intermediate position $\boldsymbol{x}'$. Once we have computed $\boldsymbol{x}'$, we perform volume correction.

6 RESULTS

We performed all of our examples on a Mac Pro with 3.5GHz Intel Xeon E5. We organize our representative examples by topology change, volume control, scenarios with external forces, computational timings, surface area transition, convergence accuracy to Plateau's laws, and a comparison to a real footage.

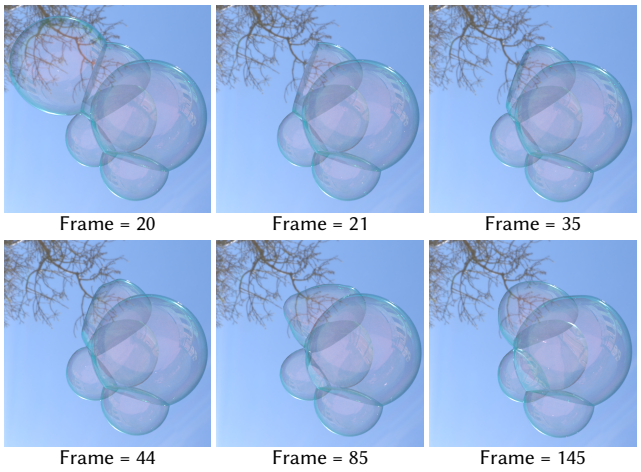

Fig. 5. Foam of six bubbles. The top left image is a foam at equilibrium. After a bubble bursts at Frame 21, the foam gradually settles down to the next equilibrium state while oscillating.

A foam setup of Figure 5 highlights the ability of our method to reproduce visually convincing soap films and foams with high accuracy. The simulation took 149 msec per timestep on average. For all the experiments, we employed constant time steps ranging from 1 to 10 msec depending on the minimal volume of bubbles. The volume change of regions was less than 0.1% over the course of simulation. To visualize films, we extended the "thindielectric" BSDF in Mitsuba renderer [Jakob 2010] to achieve the effect of thin film interference. Figure 6 shows an example of a film strained by two rings that is stretching and pinches off as it contacts in the center. With the aid of LosTopos surface tracker, our algorithm naturally handles such topology changes.

Volume Control. Figure 7 shows an example of explicit volume control due to the inflation and the deflation of a bubble. In this example, the surfaces of films locally bounce for a while after the volume control and gradually returns to the equilibrium state of a spherical shape. We believe that such an example can be applied to reproduce the other interesting effects of injecting or removing air inside bubbles through a straw. We perform the volume control by simply changing the volumes of the target bubbles at the beginning of each time step.

External Forces. We point out that external forces such as gravity forces are straightforward to incorporate in our model by simply adding them to $d^2\boldsymbol{x}/dt^2$ with $(-\beta\,\partial\mathcal{A}/\partial\boldsymbol{x})$. In either case of volume control (Figure 7) and external forces (Figure 9), our operation does not violate Plateau's laws. Figure 8 shows an example of our method coupled with a fluid simulator via external forces. For simplicity, we employed Stable Fluids [Stam 1999], but our method is not limited to any fluid solver. At the beginning of each time step, we first advance the fluid solver and compute velocity and pressure on grids. We then obtain the velocity $\boldsymbol{v}$ and the pressure p of the air at an arbitrary position $\boldsymbol{x}$ through the trilinear interpolation. When computing the intermediate velocity $\boldsymbol{u}'$, we simply add the

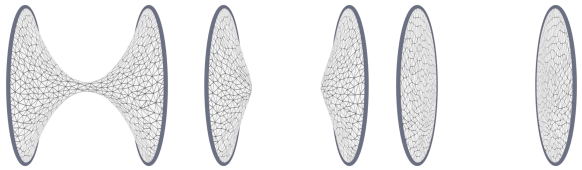

Fig. 6. Wireframe view of a strained film attached to two rings being stretched one another. From left to right: The film splits into two when the thickness exceeds a critical point. Surface tension forces induce acceleration and they drive the film to split apart and the films converge to two flat discs.

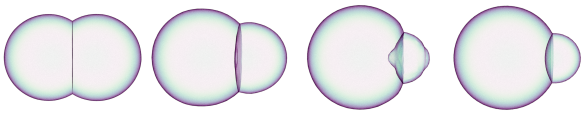

Fig. 7. Transformation of film surfaces caused by a volume change. Left: A double bubble at equilibrium. Center left: Volume changing by inflating the left region and deflating the right region. Center right: Each film component becomes aspherical immediately after the volume change for a short while. Right: The steady state after volume change. The boundary membrane is pushed to the larger region since the smaller region has the greater pressure, hence Plateau's laws still hold.

pressure of the air by

$$\boldsymbol{u}' = \boldsymbol{u} + \Delta t \left(-\beta \frac{\partial \mathcal{A}}{\partial \boldsymbol{x}} + p(\boldsymbol{v}\cdot\boldsymbol{n})\boldsymbol{n}\right). \tag{9}$$

The film vertices receiving forces due to the wind are imposed to have the negative inner product $\boldsymbol{v}\cdot\boldsymbol{n}$. Figure 8 shows an example of a bubble blown by the wind spouted out from the bottom.

In this example, we only accounted for one-way coupling from the air to film. We believe that we can further two-way couple air and film by replacing pressure on the grids inside bubbles with ones computed during the volume correction phase.

6.1 Timings

We analyzed timings through a range of experiments and observed that our method yields significant performance improvement over previous methods. Figure 10 illustrates a comparison with the method of Da et al. [2015] for the same setups. Each timing includes mesh processing by LosTopos in common. In this experiment, our results were from 3 to 21 times faster than Da et al. [2015]. We, however, emphasize that our governing equation and the scheme are quite different from Da et al. [2015]. This comparison thereby serves as a supplemental. In this experiment, we accelerated the method of Da et al. [2015] with the fast multipole method (FMM) to improve the runtime cost down to $O(N_V)$ [Da et al. 2015].

Table 1 outlines timings of our individual simulation step. Currently, the surface evolution and the topological operations via LosTopos surface tracker dominate the cost of the calculations. However, our method is not tied with any specific surface tracker, hence we view this issue as orthogonal to our main contributions. We note

Frame = 0 Frame = 80 Frame = 160 Frame = 240

Fig. 8. A bubble blown up by the wind. The pressure of the air simulated by a fluid solver affects the bubble as an external force.

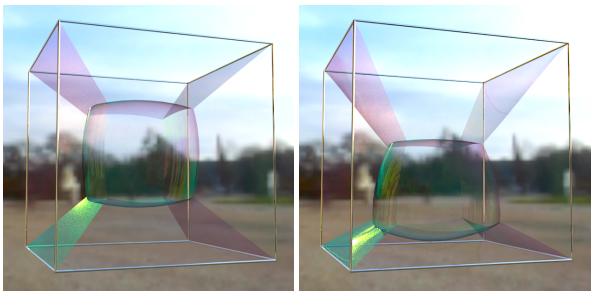

Fig. 9. Steady state of a film spanning a cubic frame. Left: Without external forces. Right: Under an extremely strong gravity (exaggerated twenty times).

that "Others" are mostly spent for conversion operations that bridge different data types of LosTopos engine and Eigen Library [Guennebaud et al. 2010] and thus can be sped up with further low-level optimization.

Figure 11 compares the transition of our computational complexity of two different scenarios. The top row examines the rise of complexity due to the increase of vertices by increasing the mesh resolution. In the bottom row, we increased the number of regions of cubes in each dimension. The computational cost of our method scales closely to linear for both cases.

6.2 Analytical Solution and Numerical Convergence

We measured the time transition of surface area for two examples: a double bubble and a catenoid (See Figure 13 top), and observed that the numerical solution slowly converges to the analytical solution as expected. Figure 13 shows our results and the counterpart analytical surface areas overlaid with striped lines.

The analytical surface area of the steady state of a double bubble is $A_{min} = 27\pi\ \sqrt[2/3]{V/9\pi}$ for the volume of each representative region V. We set up a single bubble vertically split at the center with $V = 2.0612$ and $A_{min} = 14.803$. The resulting converged area of our simulation was 14.813.

Our next example is a catenoid, of which the closed-form solution is given by

$$(x, y, z) = (a\cosh(v/a)\cos u, a\cosh(v/a)\sin u, v)\,,$$

where $a, b > 0$, $u \in [0, 2\pi)$, and $v \in [-b, b]$ denote arbitrary scalar parameters. The analytical solution of the surface area is given by $\pi a^2\,(\sinh(2b/a) + 2b/a)$. We set up a cylinder shaped surface attached to the two solid rings on its edges with parameters set $a = 0.39565, b = 0.28$, and $A_{min} = 1.64895$, respectively. The resulting converged area of our simulation was 1.6511. We also observed that the surface areas of both examples approach the analytical solutions as we increase the mesh resolution.

6.3 Plateau's Laws

Plateau's laws state that three film surfaces meet at a triple-junction at $\arccos(-1/2) = 120°$ and four triple-junctions meet at a quad-junction at $\arccos(-1/3) \approx 109.47°$ [Ball 2009]. These geometric conditions are essential for visually pleasing simulation of films. Table 2 provides the list of errors of angles of triple/quad junctions. We also provide the errors of the same experiments using the method of Brakke [1992] and Da et al. [2015] for reference. Note that discretization of the surface area for each vertex in Equation (10) of Da et al. [2015] is not uniquely determined at non-manifold junctions. The errors in T/Q (Da1) column are examples when each of the areas is simply discretized using the incident triangles per region. Fei [2017] pointed out that discretization as one-third of the total area of the incident triangles divided by the number of incident regions gives superior results because it eliminates mesh-dependence and thereby corrects force imbalances arising from asymmetries in the incident triangulations [Da et al. 2017]. Applying this discretization, results improved to as in T/Q (Da2) column. We observed that all the triple/quad-junction errors of our examples, including scenes with external forces, were less than 2.0° and approached zero as the mesh resolution increases.

6.4 Comparison with a Real Footage

Figure 12 shows a comparison with a real experiment recorded using a high-speed camera [Pucci et al. 2015]. In this experiment, an immersed soap bubble is slowly merging to a static soap film at the bottom. We observe that the result of our model resembles the real footage for a long duration. The subtle difference is due to the loss of kinematic energy caused by the way we use the surface tracker as we discuss later.

7 DISCUSSION

Computational complexity. As we noted before, the computational complexity of our algorithm is $O(N_T) + O(N_V) + O(N_R)$. This order can be approximated simply by $O(N_V)$ since N_T is linear to N_V in most cases, and the computational cost of volume correction is much less than the surface evolution by $\partial\mathcal{A}/\partial\boldsymbol{x}$ even with a large number of regions. While another formulation by Da et al. [2015] can be made to achieve the same linear computational complexity using the fast multipole method, the computational complexity of our formulation is intrinsically linear without any additional numerical technique. In our current implementation, the computation time is dominated by LosTopos. Since our model is independent of the surface tracker, we expect that a more efficient surface tracker can potentially accelerate the overall computation significantly. We, however, found that LosTopos is currently the best option for our purpose.

Target	Vertices	Faces	Regions	TPF (Ours)	TPF (Da)	Ratio
Single cubic bubble	1003	2002	2	24.70	179.72	7.3
Double bubble	1792	3626	3	68.76	526.83	7.7
125 latticed bubbles	6702	16192	126	160.22	3402.61	21.2
Film spanning a frame	2679	5742	8	431.647	1323.91	3.1
Catenoid	1346	2688	2	69.17	221.25	3.2
Six bubbles	4882	9972	7	149.04	2570.16	17.2

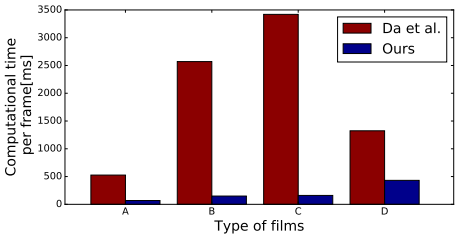

Fig. 10. Comparison of timings per frame for various film examples to Da et al. [2015]. Vertices, Faces, and Regions refer to the number of the initial vertices, faces, and regions, respectively. TPF is the computational time per frame (ms), and Ratio is the ratio of computational timings TPF (Da) / TPF (Ours). The figure on the right side illustrates the timings of the results in the list. Films A, B, C, and D are a double bubble, a foam of six bubbles, 125 lattice-configured bubbles, and a film spanning a frame respectively.

Table 1. Timings per frame by simulation component (ms). The list is ordered as follows: the total computational time, the evolution by $\partial \mathcal{A}/\partial \boldsymbol{x}$, volume correction, mesh processing by LosTopos, and remaining operations.

Target	Total	$\partial \mathcal{A}/\partial \boldsymbol{x}$	VC	LosTopos	Others
Single cubic bubble	24.70	1.01	0.01	20.98	2.71
125 latticed bubbles	160.22	2.63	1.08	148.39	8.12
Six bubbles	149.04	4.04	1.02	135.94	8.05

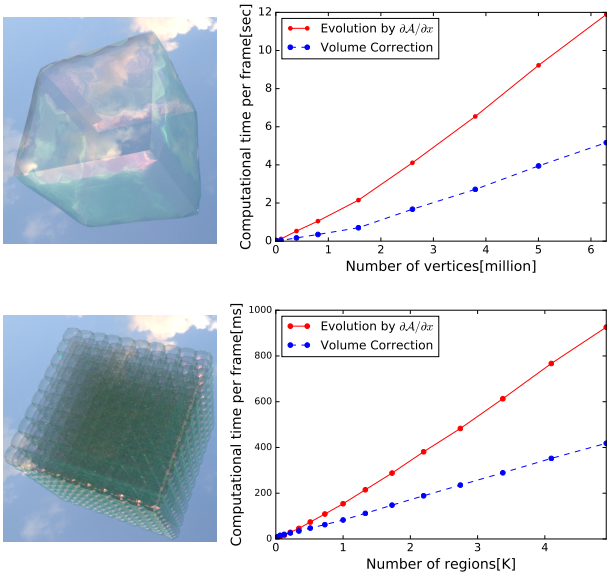

Fig. 11. Complexity analysis by different discrete variables. Up: Timings due to the increase of the number of vertices on surfaces (maximal number 6291458). Bottom: Timings due to the increase of number of regions (maximal number 4913). Notice that both draw closely linear curves.

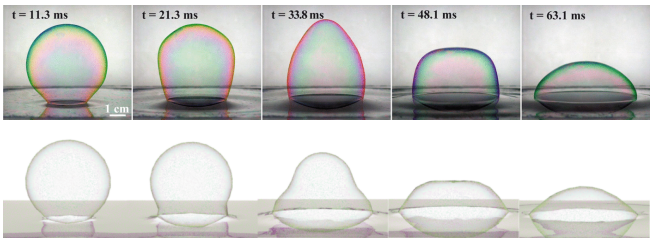

Fig. 12. Time sequence of a soap bubble being trapped in a planar film. Up: Experiment by Pucci et al. [2015]. Bottom: Our simulation.

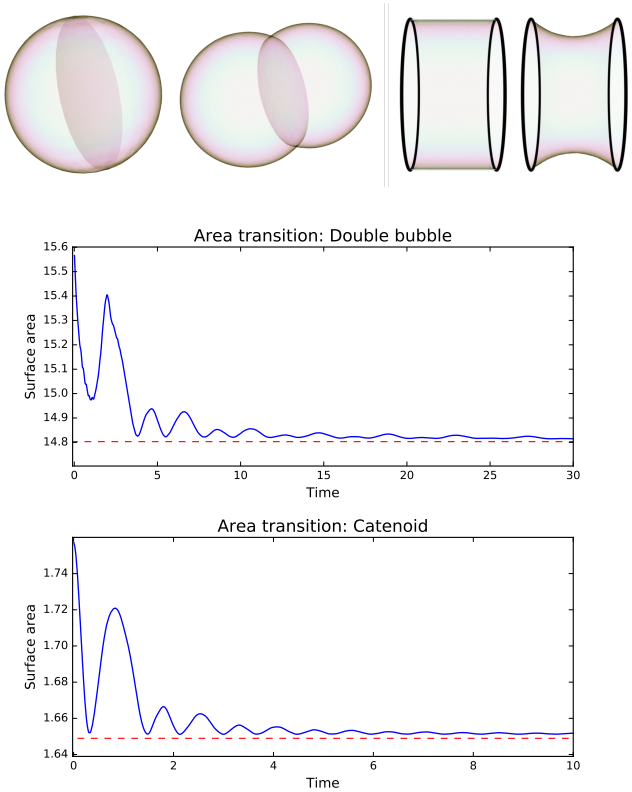

Fig. 13. Top: Initial states of a double bubble and a catenoid and their steady states. Middle and bottom: Surface area transition of films. Striped lines show their corresponding analytical solutions.

Table 2. List of errors of the angles on non-manifold junctions from Plateau's laws. In the list, T denotes the root mean square (RMS) of the differences between $\arccos(-1/2)$ and the dihedral angles of the faces incident to Plateau borders, and Q denotes the RMS of the differences between $\arccos(-1/3)$ and the angles of the Plateau borders intersecting at quad-junctions.

Target	T/Q (Ours)	T/Q (Brakke)	T/Q (Da1)	T/Q (Da2)
Double bubble	1.53°	1.40°	14.67°	3.62°
Film spanning a frame	0.60°/0.67°	1.80°/2.27°	4.04°/5.77°	1.84°/3.68°

Alternative Numerical Solvers. To compute intermediate quantities, we can apply an implicit scheme instead of symplectic Euler [Desbrun et al. 1999]. We observed that the implicit scheme allows us to increase the time step by a factor of ten at most, but it was not necessarily numerically more stable in our implementation due to some additional parameters.

Another alternative approach would be to compute the next position and the velocity by computing $\beta\, \partial \mathcal{A}/\partial \boldsymbol{x}$ and $\Delta p \boldsymbol{n}$ in a strongly coupled manner without computing intermediate quantities. Since our model is not analytically differentiable, the Newton-Raphson method is not applicable. Hence, the bisection method is an option for seeking the value Δp that preserves the enclosed volumes. We, however, expect that such a scheme would increase the computational complexity to as much as $O(2^{N_R})$, and we view this as too expensive for a large number of bubbles.

Connection to physics. We showed a theoretical connection between our governing equation and the Navies-Stokes equations. This is valid only for manifold surfaces at this moment. We believe that generalizing quantities such as surface normal for junctions extends it to non-manifold cases.

Different scenarios. Da et al. [2015] demonstrated scenarios where a double bubble is pulled apart and bubbles are generated from a sweeping wire ring. We note that our method can also handle such settings without difficulties. When bubbles are split, we just need to reassign the region labels of the triangles consisting of the boundary surface. In our implementation, this process is automatically done by LosTopos. For the latter scenario, we add a region for the newly generated bubble and compute its initial volume.

Limitations. In comparison to a mesh based simulation of fluid surfaces that takes thickness of films into account [Zhu et al. 2014], we regard the film to be infinitesimally thin. Our model thus cannot simulate the moment of burst of bubbles or temporal color change of liquid surface. We expect that incorporation of liquid simulation on deformable surfaces would enable simulation of such phenomena using our model.

When a surface tracker generates a new vertex by edge collapse or split, we interpolate velocities of neighboring vertices by barycentric interpolation. Alternatives for more accurate interpolations are higher-order approximations such as spline interpolation and Hermite interpolation. We found that, however, a small amount of energy loss occurs with any of the higher-order alternatives mentioned above , because a naive interpolation of vectors with different directions decreases its magnitude, similarly to triangle inequality for two vectors. We have not found an effective solution which can preserve local energy around a newly generated vertex. Energy preserving interpolation of a vector field on a triangle mesh is an interesting topic for future work.

Our current implementation is not unconditionally stable. We speculate that the maximum timestep size is proportional to the maximal edge length and the inverse of the maximal value of $\partial \mathcal{A}/\partial \boldsymbol{x}$. We, however, have not analyzed the CFL condition or relevant condition for stability since such an analysis depends on several complex factors such as discretization and the stability of a surface tracker. We also observed that LosTopos becomes unstable if there are significant changes in a triangle mesh. We thus add some damping to avoid a sudden change for a large time step. A different discretization step with an improved surface tracker can potentially resolve this stability issue. This issue is independent of our theoretical formulation of film dynamics as a geometric flow.

8 CONCLUSIONS

We introduced an extension of hyperbolic mean curvature flow to simulate dynamics of films and foams. Unlike previous works, we start from the observations made by Plateau to construct a geometric flow which precisely captures the properties of the steady-state shapes of soap films. We generalized a geometric flow to consistently handle an arbitrary combination of open boundaries, enclosed volumes, and connectivity of regions, that are crucial for robust simulation of films and foams. We showed that the resulting geometric flow is closely related with the Navier-Stokes equations. The main advantages of our model are the computational speed and the convergence of the films following Plateau's laws. Our method also enables explicit volume control and the straightforward addition of external force, allowing for new physics phenomena such as floating bubbles blown by the wind and then slowly fall due to the gravity.

For Plateau's problem, our model provides a unified solution as an area-minimizer without heuristic operations. Moreover, it solves a generalized version of Plateau's problem in the presence of external forces. We believe that it can open up lots of opportunities across different fields, similarly to the fact that another generalization of the Plateau's problem, called Plateau-Kirchhoff problem [Giomi and Mahadevan 2012] led to numerous follow-up works [Giusteri et al. 2017; Perez et al. 2017]. Since our model can also be seen as a general geometric flow, future work includes applications to problems in geometry processing such as finding CMC surfaces for architectural design.

ACKNOWLEDGMENTS

The authors would like to thank Sigurd Ofstad for discussion and experiments during the development of the algorithm, Hisanari Otsu for customization of Mitsuba renderer, Christopher Batty, Fang Da, and Raymond Yun Fei for the discussion on the previous work, and Jamorn Sriwasansak for the advice on the Thai language in the supplementary video. This work is supported in part by grants from Nikon Corporation.

A MEAN CURVATURE AND VARIATIONAL DERIVATIVE OF THE AREA FUNCTIONAL

A handful of books on Riemannian geometry and mean curvature flow [Berger and Gostiaux 2012; López 2013; Mantegazza 2011] provide proofs of the equivalence between mean curvature normal and the variational derivative of the area functional. However, their proofs are targeted for a Riemannian manifold with any dimension and not immediately clear how it relates to our setups. To this end, we provide a sketch of a proof for a special case of surfaces in three-dimensional Euclidean space. We first fix our definitions and notations. The area functional for a surface $S : U \to \mathbb{R}^3$ is defined

as

$$\mathcal{A}(S) = \int_\Sigma ds = \int_U |S_u \times S_v| du dv \tag{10}$$

where $\Sigma = S(U)$. Using a continuous map $V : U \to \mathbb{R}^3$, we can define a family of surfaces $S^\epsilon = S + \epsilon V$ parametrized by a parameter $\epsilon \in (-E, E)$ for a small positive number E. For each ϵ, we write $L^2(\Sigma_\epsilon)$ for the space consisting of square-integrable functions on $\Sigma_\epsilon = S^\epsilon(U)$. The variational derivative of the area functional on S is then defined by

$$\frac{\partial \mathcal{A}}{\partial \boldsymbol{x}} \equiv \arg\max_{\|V\|=k} \frac{d}{d\epsilon}\Big|_{\epsilon=0} \mathcal{A}(S^\epsilon) \tag{11}$$

where $\|V\|$ is the total speed defined by $(\int_\Sigma |V|^2 ds)^{1/2}$ and k is a real constant. To uniquely define $\partial\mathcal{A}/\partial\boldsymbol{x}$, we here set $k = \|H\|$ with a norm in $L^2(\Sigma)$ defined by $\|H\| = (\int_\Sigma |H|^2 ds)^{1/2}$.

Proof of $\partial\mathcal{A}/\partial\boldsymbol{x} = H\boldsymbol{n}$. Let S, V, S^ϵ be as above. We can decompose V into the normal and the tangential part as $V = V^\perp \boldsymbol{n} + V^T \boldsymbol{t}$. The first variation formula states

$$\frac{d}{d\epsilon}\Big|_{\epsilon=0} \int_{\Sigma_\epsilon} ds^\epsilon = \int_\Sigma V^\perp H ds \tag{12}$$

where Σ_ϵ and ds^ϵ are the domain of integration and the area element for S^ϵ. Note that the operation $\mathcal{F} : g, h \to \int_\Sigma g(s)h(s)ds$ is an inner product of $L^2(\Sigma)$. Therefore we can apply the Cauchy-Schwarz inequality to the right-hand side of Equation (12), and obtain,

$$\langle H, V^\perp \rangle \le \|H\| \|V^\perp\|. \tag{13}$$

Equality holds *if and only if* $V^\perp = C_H H$ for a constant C_H. Therefore, $V' = C_H H\boldsymbol{n}$ maximizes the first variation among continuous maps $V : U \to \mathbb{R}^3$ with $\|V\| = C$ for a positive constant C. By taking $C = \|H\|$, we have $C_H = 1$ and obtain

$$\arg\max_{\|V\|=\|H\|} \frac{d}{d\epsilon}\Big|_{\epsilon=0} \mathcal{A}(S^\epsilon) = H\boldsymbol{n}. \tag{14}$$

□

B PROOF OF VOLUME PRESERVATION

We here prove that moving each point by $\Delta d\boldsymbol{n}$ with Δd as in Section 4.2 corrects all the enclosed volumes for first-order approximation. This means,

$$\frac{1}{3}\int_{\partial\Omega_i'} (\boldsymbol{x} + \Delta d\boldsymbol{n}) \cdot \boldsymbol{n}' ds' \approx V_i + \Delta V_i \tag{15}$$

where $\partial\Omega' = S'(U)$, $\boldsymbol{n}' = S_u' \times S_v' / |S_u' \times S_v'|$, and $ds' = |S_u' \times S_v'| du dv$ are the domain of integration, the unit normal, and the area element for the variational surface $S' \equiv S + \Delta d\boldsymbol{n}$.

Proof. We first show that the left-hand side of Equation (15) is approximated to $V_i + \int_{\partial\Omega_i} \Delta d\, ds$. We note that non-manifold junctions do not contribute to the integral since the set of non-manifold junctions is of measure-zero with respect to the area element ds. We have,

$$\begin{aligned} S_u' \times S_v' &= (S_u + \Delta d\boldsymbol{n}_u) \times (S_v + \Delta d\boldsymbol{n}_v) \\ &= S_u \times S_v + \Delta d\boldsymbol{n}_u \times S_v + \Delta d S_u \times \boldsymbol{n}_v + (\Delta d)^2 \boldsymbol{n}_u \times \boldsymbol{n}_v \\ &= \Big(1 + \Delta d(A + D) + (\Delta d)^2 (AD - BC)\Big) S_u \times S_v. \end{aligned} \tag{16}$$

In the above, we employed Weingarten equations [Kreyszig 1991] $\boldsymbol{n}_u = AS_u + BS_v$ and $\boldsymbol{n}_v = CS_u + DS_v$ where the coefficients are given by

$$\begin{pmatrix} A \\ B \\ C \\ D \end{pmatrix} = \frac{1}{|S_u \times S_v|^2} \begin{pmatrix} (S_u \cdot S_v)(S_{uv} \cdot \boldsymbol{n}) - (S_v \cdot S_v)(S_{uu} \cdot \boldsymbol{n}) \\ (S_u \cdot S_v)(S_{uu} \cdot \boldsymbol{n}) - (S_u \cdot S_u)(S_{uv} \cdot \boldsymbol{n}) \\ (S_u \cdot S_v)(S_{vv} \cdot \boldsymbol{n}) - (S_v \cdot S_v)(S_{uv} \cdot \boldsymbol{n}) \\ (S_u \cdot S_v)(S_{uv} \cdot \boldsymbol{n}) - (S_u \cdot S_u)(S_{vv} \cdot \boldsymbol{n}) \end{pmatrix}.$$

Hence, it holds $ds' = \Big(1 + \Delta d(A + D) + (\Delta d)^2 (AD - BC)\Big) ds$ and $\boldsymbol{n}' = \boldsymbol{n}$. Using these relations, we have

$$\begin{aligned} &\frac{1}{3}\int_{\partial\Omega_i'} (\boldsymbol{x} + \Delta d\boldsymbol{n}) \cdot \boldsymbol{n}' ds' \\ = \;& \frac{1}{3}\int_{\partial\Omega_i} (\boldsymbol{x} \cdot \boldsymbol{n} + \Delta d) \Big(1 + \Delta d(A + D) + (\Delta d)^2 (AD - BC)\Big) ds. \end{aligned} \tag{17}$$

Regarding $(\Delta d)^2 = 0$, the above equation is approximated to

$$\begin{aligned} &\frac{1}{3}\int_{\partial\Omega_i} \boldsymbol{x} \cdot \boldsymbol{n} ds + \frac{1}{3}\int_{\partial\Omega_i} \Delta d\,(1 + (A + D)\boldsymbol{x} \cdot \boldsymbol{n})\, ds \\ = \;& V_i + \int_{\partial\Omega_i} \Delta d\, ds. \end{aligned} \tag{18}$$

In the above, we used the relations that $A + D = 2H$ with mean curvature H, and $\int_{\partial\Omega_i} H\boldsymbol{x} \cdot \boldsymbol{n} ds = \int_{\partial\Omega_i} ds$ under the assumption that $\partial\Omega_i$ is a closed surface [Kreyszig 1991].

We finally claim $\int_{\partial\Omega_i} \Delta d\, ds = \Delta V_i$ by the definition of Δd. It holds

$$\begin{aligned} \int_{\partial\Omega_i} \Delta d ds &= \int_{\partial\Omega_{i,air}} d_i ds + \sum_{j\neq i} \int_{\partial\Omega_{i,j}} (d_i - d_j) ds \\ &= d_i Area(i, air) + \sum_{j\neq i} (d_i - d_j) Area(i, j) \\ &= d_i A_{ii} + \sum_{j\neq i} d_j A_{ij} = \sum_j d_j A_{ij}. \end{aligned} \tag{19}$$

Since $\boldsymbol{Ad} = \Delta\boldsymbol{V}$, we have $d_j = (A^{-1}\Delta V)_j = \sum_k A^{-1}_{jk} \Delta V_k$. Substituting this into Equation (19), we obtain

$$\begin{aligned} \sum_j d_j A_{ij} &= \sum_j A_{ij} \sum_k A^{-1}_{jk} \Delta V_k \\ &= \sum_k \Delta V_k \sum_j A_{ij} A^{-1}_{jk} = \sum_k \Delta V_k \delta_{ik} = \Delta V_i. \end{aligned} \tag{20}$$

Hence, the enclosed volume of region Ω_i is corrected for first-order approximation. □

REFERENCES

Frederick J Almgren and Jean E Taylor. 1976. The geometry of soap films and soap bubbles. *Scientific American* 235 (1976), 82–93.

Luigi Ambrosio. 2015. Regularity theory for mass-minimizing currents (after Almgren-De Lellis-Spadaro). *Calculus of Variations and Geometric Measure Theory* (2015), 1–23.

John H Argyris, T Angelopoulos, and Bruno Bichat. 1974. A general method for the shape finding of lightweight tension structures. *Computer Methods in Applied Mechanics and Engineering* 3, 1 (1974), 135–149.

Philip Ball. 2009. Shapes: Nature's Patterns: A Tapestry in Three Parts (Natures Patterns). In *Shapes: Nature's Patterns: A Tapestry in Three Parts (Natures Patterns)*. Oxford University Press, Oxford,UK, 97–98.

Marcel Berger and Bernard Gostiaux. 2012. *Differential Geometry: Manifolds, Curves, and Surfaces: Manifolds, Curves, and Surfaces*. Vol. 115. Springer Science & Business Media, New York, NY, USA.

CV Boys. 1958. Soap-Bubbles, their colors and forces which mold them (1890). (1958).
Kenneth A. Brakke. 1992. The Surface Evolver. *Experimental Mathematics* 1, 2 (1992), 141–165.
JS Brew and WJ Lewis. 2003. Computational form-finding of tension membrane structures-Non-finite element approaches: Part 1. Use of cubic splines in finding minimal surface membranes. *International journal for numerical methods in engineering* 56, 5 (2003), 651–668.
Oleksiy Busaryev, Tamal K. Dey, Huamin Wang, and Zhong Ren. 2012. Animating bubble interactions in a liquid foam. *ACM Trans. Graph.* 31, 4 (2012), 63.
Fang Da, Christopher Batty, and Eitan Grinspun. 2014. Multimaterial Mesh-Based Surface Tracking. *ACM Trans. on Graphics (SIGGRAPH 2014)* 33 (2014), 1–11.
Fang Da, Christopher Batty, Chris Wojtan, and Eitan Grinspun. 2015. Double Bubbles Sans Toil and Trouble: Discrete Circulation-Preserving Vortex Sheets for Soap Films and Foams. *ACM Trans. on Graphics (SIGGRAPH 2015)* 34, 4 (2015), 149:1–149:9.
Fang Da, Christopher Batty, Chris Wojtan, and Eitan Grinspun. 2017. Project page of "Double Bubbles Sans Toil and Trouble: Discrete Circulation-Preserving Vortex Sheets for Soap Films and Foams". (2017). http://www.cs.columbia.edu/cg/doublebubbles/
Fang Da, Christopher Batty, Chris Wojtan, and Eitan Grinspun. 2015. Use of Fast Multipole to Accelerate Discrete Circulation-Preserving Vortex Sheets for Soap Films and Foams. (2015), 2.
Mathieu Desbrun, Mark Meyer, Peter Schröder, and Alan H. Barr. 1999. Implicit Fairing of Irregular Meshes Using Diffusion and Curvature Flow. In *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '99)*. ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, 317–324.
Kong Dexing, Liu Kefeng, and Wang Zenggui. 2009. Hyperbolic mean curvature flow: evolution of plane curves. *Acta Mathematica Scientia* 29, 3 (2009), 493–514.
Jesse Douglas. 1931. Solution of the problem of Plateau. *Trans. Amer. Math. Soc.* 33, 1 (1931), 263–321.
Roman Durikovic. 2001. Animation of Soap Bubble Dynamics, Cluster Formation and Collision. *Comput. Graph. Forum* 20, 3 (2001), 67–76.
Raymond Yun Fei. 2017. Personal communication. (Aug. 2017).
Luca Giomi and Lakshminarayanan Mahadevan. 2012. Minimal surfaces bounded by elastic lines. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 468, 2143 (mar 2012), 1851–1864.
Giulio G Giusteri, Luca Lussardi, and Eliot Fried. 2017. Solution of the Kirchhoff–Plateau problem. *Journal of Nonlinear Science* 27, 3 (2017), 1043–1063.
Gaël Guennebaud, Benoît Jacob, et al. 2010. Eigen v3. http://eigen.tuxfamily.org. (2010).
Jenny Harrison. 2014. Soap film solutions to Plateau's problem. *Journal of Geometric Analysis* 24, 1 (2014), 271–297.
Jenny. Harrison and Harrison Pugh. 2015. Plateau's Problem: What's Next. *ArXiv e-prints* (Sept. 2015), 27. arXiv:1509.03797
Chun-Lei He, De-Xing Kong, and Kefeng Liu. 2009. Hyperbolic mean curvature flow. *Journal of Differential Equations* 246, 1 (2009), 373–390.
Jeong-Mo Hong, Ho-Young Lee, Jong-Chul Yoon, and Chang-Hun Kim. 2008. Bubbles alive. *ACM Trans. Graph.* 27, 3 (2008), 48:1–48:4.
Michael Hutchings, Frank Morgan, Manuel Ritoré, and Antonio Ros. 2002. Proof of the double bubble conjecture. *Annals of Mathematics* 155, 2 (2002), 459–489.
Cyril Isenberg. 1978. *The science of soap films and soap bubbles*. Courier Corporation, North Chelmsford, Massachusetts, USA.
Alec Jacobson, Daniele Panozzo, et al. 2013. libigl: A simple C++ geometry processing library. (2013). http://igl.ethz.ch/projects/libigl/.
Wenzel Jakob. 2010. Mitsuba renderer. (2010). http://www.mitsuba-renderer.org.
Byungmoon Kim, Yingjie Liu, Ignacio Llamas, Xiangmin Jiao, and Jarek Rossignac. 2007. Simulation of Bubbles in Foam With The Volume Control Method. *ACM Trans. on Graphics (SIGGRAPH 2007)* 26, 3 (2007), 10.
Erwin Kreyszig. 1991. *Differential geometry*. Dover Publications, New York.
Philippe G LeFloch and Knut Smoczyk. 2008. The hyperbolic mean curvature flow. *Journal de mathématiques pures et appliquées* 90, 6 (2008), 591–614.
Rafael López. 2013. *Constant mean curvature surfaces with boundary*. Springer Science & Business Media, New York, NY, USA.
Carlo Mantegazza. 2011. *Lecture notes on mean curvature flow*. Vol. 290. Springer Science & Business Media, New York, NY, USA.
Barry Merriman, James K. Bence, and Stanley J. Osher. 1994. Motion of Multiple Junctions: A Level Set Approach. *J. Comput. Phys.* 112 (1994), 334–363.
Mark Meyer, Mathieu Desbrun, Peter Schröder, and Alan H. Barr. 2002. Discrete Differential-Geometry Operators for Triangulated 2-Manifolds. (2002).
Marek Krzysztof Misztal, Kenny Erleben, Adam Bargteil, Jens Fursund, Brian Bunch Christensen, Jakob Andreas Bærentzen, and Robert Bridson. 2014. Multiphase flow of immiscible fluids on unstructured moving meshes. *IEEE transactions on visualization and computer graphics* 20, 1 (2014), 4–16.
Rhudaina Z. Mohammad and Karel Švadlenka. 2002. Proof of the double bubble conjecture. *Ann. Math* (2002), 0406017.
Matthias Müller. 2009. Fast and robust tracking of fluid surfaces. In *Proceedings of the 2009 ACM SIGGRAPH/Eurographics Symposium on Computer Animation (SCA '09)*. ACM, New York, NY, USA, 237–245.
Hao Pan, Yi-King Choi, Yang Liu, Wenchao Hu, Qiang Du, Konrad Polthier, Caiming Zhang, and Wenping Wang. 2012. Robust Modeling of Constant Mean Curvature Surfaces. *ACM Trans. Graph.* 31, 4, Article 85 (July 2012), 11 pages.
Saket Patkar, Mridul Aanjaneya, Dmitriy Karpman, and Ronald Fedkiw. 2013. A hybrid Lagrangian-Eulerian formulation for bubble generation and dynamics. In *The ACM SIGGRAPH / Eurographics Symposium on Computer Animation, SCA '13, Anaheim, CA, USA, July 19-21, 2013*. ACM Press, New York, NY, USA, 105–114.
Jesus Perez, Miguel A. Otaduy, and Bernhard Thomaszewski. 2017. Computational Design and Automated Fabrication of Kirchhoff-Plateau Surfaces. *ACM Trans. on Graphics (Proc. of ACM SIGGRAPH)* 36, 4 (2017), 62.1–62.12.
Ulrich Pinkall and Konrad Polthier. 1993. Computing Discrete Minimal Surfaces and Their Conjugates. *Experimental Mathematics* 2 (1993), 15–36.
Konrad Polthier and Wayne Rossman. 2002. Discrete Constant Mean Curvature Surfaces And Their Index. In *Visualization and Mathematics*. Springer Verlag, 47–77.
Thomas R Powers, Greg Huber, and Raymond E Goldstein. 2002. Fluid-membrane tethers: minimal surfaces and elastic boundary layers. *Physical Review E* 65, 4 (2002), 041901.
Giuseppe Pucci, Daniel M Harris, and John WM Bush. 2015. Partial coalescence of soap bubbles. *Physics of Fluids* 27, 6 (2015), 061704.
Tibor Rado. 1930. On Plateau's Problem. *The Annals of Mathematics* 31, 3 (jul 1930), 457.
Robert I. Saye and James A. Sethian. 2013. Multiscale Modeling of Membrane Rearrangement, Drainage, and Rupture in Evolving Foams. *Science* 340, 6133 (2013), 720–724.
Jos Stam. 1999. Stable Fluids. In *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '99)*. ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, 121–128.
Michael Struwe. 2014. *Plateau's Problem and the Calculus of Variations.(MN-35)*. Princeton University Press, Princeton, New Jersey, USA.
John M Sullivan and Frank Morgan. 1996. Open problems in soap bubble geometry. *International Journal of Mathematics* 7, 06 (1996), 833–842.
Jean E Taylor. 1976. The structure of singularities in soap-bubble-like and soap-film-like minimal surfaces. *Annals of Mathematics* 103, 3 (1976), 489–539.
Nils Thürey, Chris Wojtan, Markus Gross, and Greg Turk. 2010. A multiscale approach to mesh-based surface tension flows. *ACM Transactions on Graphics* 29, 4 (2010), 10.
Karel Švadlenka, Elliott Ginder, and Seiro Omata. 2014. A variational method for multiphase volume-preserving interface motions. *J. Comput. Appl. Math.* 257 (2014), 157–179.
Shing-Tung Yau. 2000. Review of geometry and analysis. *Mathematics: frontiers and perspectives* (2000), 353–401.
Yizhong Zhang, Huamin Wang, Shuai Wang, Yiying Tong, and Kun Zhou. 2012. A deformable surface model for real-time water drop animation. *IEEE Transactions on Visualization and Computer Graphics* 18, 8 (2012), 1281–1289.
Wen Zheng, Jun-Hai Yong, and Jean-Claude Paul. 2006. Simulation of Bubbles. In *Proceedings of the 2006 ACM SIGGRAPH/Eurographics Symposium on Computer Animation (SCA '06)*. Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, 325–333.
Bo Zhu, Ed Quigley, Matthew Cong, Justin Solomon, and Ronald Fedkiw. 2014. Codimensional surface tension flow on simplicial complexes. *ACM Transactions on Graphics (TOG)* 33, 4 (2014), 111.