This paper introduces a novel physically-based vortex fluid model for films, aimed at accurately simulating cascading vortical structures on deforming thin films. Central to our approach is a novel mechanism decomposing the film’s tangential velocity into circulation and dilatation components. These components are then evolved using a hybrid particle-mesh method, enabling the effective reconstruction of three-dimensional tangential velocities and seamlessly integrating surfactant and thickness dynamics into a unified framework. By coupling with its normal component and surface-tension model, our method is particularly adept at depicting complex interactions between in-plane vortices and out-of-plane physical phenomena, such as gravity, surfactant dynamics, and solid boundary, leading to highly realistic simulations of complex thin-film dynamics, achieving an unprecedented level of vortical details and physical realism.

**CCS Concepts:** • Computing methodologies → Physical simulation.

**Additional Key Words and Phrases:** Fluid simulation, thin film simulation, vortex particle method

**ACM Reference Format:**
Ningxiao Tao, Liangwang Ruan, Yitong Deng, Bo Zhu, Bin Wang, and Baoquan Chen. 2024. A Vortex Particle-on-Mesh Method for Soap Film Simulation. ACM Trans. Graph. 43, 4, Article 53 (July 2024), 14 pages. https://doi.org/10.1145/3658165
1 INTRODUCTION

Thin-film fluid simulation has witnessed significant advancements in computer graphics over recent years, with extensive research efforts dedicated to capturing the complex bubble dynamics and its color evolution by establishing thin-film fluid dynamics solvers based on first principles. This involves solving the reduced 3D Navier-Stokes equations discretized on thin-film geometries (e.g., meshes [Ishida et al. 2020] or particles [Deng et al. 2022; Wang et al. 2021]) to capture both in-plane flow details and out-of-plane deformation, along with the topological evolution in 3D space. A particularly captivating aspect of thin-film fluid dynamics is its cascade of vortices on its deforming surface, intertwined with the soap colors created by its evolving thickness, all of which are influenced by and interact with a complex three-dimensional physical environment. These interactions are multifaceted, including forces such as gravity in its simplest form, Marangoni force with surfactant dynamics, breeze force, heat convection, as well as the effects of solid boundaries and moving objects on the thin-film surface. A prominent outcome of these complex interactions is the hierarchical vortex dynamics occurring in the tangential space of an evolving film or bubble. Accurately simulating the interaction mechanisms between physical elements from the three-dimensional world and the vortex dynamics constrained in an embedded codimension-one space, is key to achieving realistic vortex dynamics of thin films.

Despite rapid advances in first-principle soap film solvers, accurately capturing and evolving cascading vortices of varying scales on deforming thin films remains challenging. A major difficulty is the lack of a direct representation to capture the structures of these multi-scale vortices on a codimension-one manifold and a physically accurate model to evolve their dynamics on the deforming thin film by interacting with the 3D environment. Extensive literature exists on vortex methods in computational physics and computer graphics, such as particles [Selle et al. 2005], filaments [Weißmann and Pinkall 2010], sheets [Pfaff et al. 2012], or their hybrids [Barnat and Pollard 2016; Huang et al. 2020] have proposed methods to simulate such complex 3D force-driven thin-film phenomena, such as Newton’s interference fringes and Rayleigh-Taylor instability, all under a tangential vortex dynamics framework, a feat previously unattainable with circulation-preserving models.

The main contributions of our approach are summarized as:

- the first physically-based vortex fluid model for generating and evolving cascading vortical structures on deforming thin films
- a novel representation of film’s tangential velocity based on circulation and dilatation
- a vortex particle-on-mesh method for dynamic interface tracking and the intricate surface flow simulation

2 RELATED WORK

Thin film simulation. The dynamics of thin films (membrane with boundaries and soap bubbles), often relies on a dimension-reduced form of the Navier-Stokes equation [Chomaz 2001; Couder et al. 1989]. This study involves two primary aspects: the evolution of thin film geometries and the interfacial flow within them. In the context of geometric evolution, researchers have developed algorithms for simulating thin films using various representations, such as level-sets [Zheng et al. 2006], meshes [Zhu et al. 2014], particles [Wang et al. 2021], and have even extended it to scenarios involving multiple bubbles with Plateau borders [Da et al. 2015; Ishida et al. 2017; Kim et al. 2007; Qu et al. 2023; Saye and Sethian 2013]. However, to achieve the creation of visually striking iridescent color patterns on thin films, it becomes essential to simulate the thickness evolution driven by interfacial flows. In response, Hill and Henderson [2016]; Huang et al. [2020] have proposed methods to simulate intricate interfacial flows on static spheres. Furthermore, Deng et al. [2022]; Ishida et al. [2020]; Wang et al. [2021] extended their work to simulated tangential flows on deforming film geometries using velocity-based formulations. To improve tangential flow accuracy, we reformulate the Navier-Stokes equation on the manifold using circulation and dilatation as the primary variables.
Vortex particle method. The vortex particle method serves as an effective solution to mitigate the numerical viscosity associated with the traditional semi-Lagrangian advection scheme [Stam 1999]. This method has shown remarkable success in preserving vortex structures within turbulent flows [Park and Kim 2005] and achieving vorticity confinement in Eulerian grid fluids [Pfaff et al. 2009; Selle et al. 2005]. While the vortex particle method is known for its simplicity, it can introduce numerical accuracy challenges attributable to random particle distribution. In particular, within regions featuring intricate vortex structures, the random distribution of particles may struggle to faithfully represent the underlying fluid behavior. To overcome this limitation, various structured vortex geometries have been introduced to provide more precise representations of vorticities. These include vortex filaments [Barnat and Pollard 2012; Ishida et al. 2022; Weißmann and Pinkall 2010], vortex sheets [Brochu et al. 2012; Pfaff et al. 2012], vortex segments [Xiong et al. 2021; Zhang and Bridson 2014], etc. It is important to emphasize that these methods primarily focus on volumetric flows. Da et al. [2015] applied the vortex sheet method to simulate bubbles. However, their method specifically targets the geometric deformation of bubbles and does not address the interfacial flow, which is the primary focus of our research.

Mesh based method. To simulate interfacial flows on triangle meshes, Shi and Yu [2004] extended the stable fluids approach proposed initially by Stam [1999] to accommodate curved surfaces through the incorporation of mesh-based differential operators. Despite its simplicity, this method inherits numerical viscosity issues, particularly in the form of numerical diffusion of vorticity. For a more systematic treatment of integral and differential operations on simplicial manifolds, Discrete Exterior Calculus (DEC) [Crane et al. 2013; Wang et al. 2023] emerges as a robust framework. Based upon DEC, Elcott et al. [2007] developed a circulation-preserving method to simulate fluids on simplicial meshes, which effectively eliminates numerical diffusion of vorticities. Azencot et al. [2014] further improved their method by directly preserving vorticities and avoiding the explicit computation of flow lines. However, it is important to highlight that these methods primarily address the 2D Navier-Stokes equations on the manifold and do not specifically target the evolution of tangential velocity within the thin film. In contrast, Ishida et al. [2020] explored the thickness evolution of bubbles using triangle meshes, but their method is rooted in the velocity formulation akin to that of Shi and Yu [2004], which suffers from significant numerical damping for vorticities.

Hybrid Lagrangian/Eulerian Methods. The traditional Particle-In-Cell (PIC) methods [Foster and Metaxas 1996; Harlow 1962], which utilize Lagrangian particles and Eulerian grids, have proven effective for realistically simulating fluids and are widely used in computer graphics. However, their significant inherent dissipation makes them less suitable for simulating turbulent flows. To mitigate this issue, the Fluid Implicit Particle (FLIP) method [Brackbill et al. 1988; Zhu and Bridson 2005] was developed, which reduces dissipation by transferring velocity increments during the grid-to-particle process. PIC/FLIP methods [Batty et al. 2007; Fu et al. 2017; Jiang et al. 2015; Qu et al. 2022] commonly employ MAC-Grids, where pressure projection is used to maintain fluid incompressibility, though some studies have opted for an unstructured mesh as the background.
grid. Matsumoto and Kawata [1990] introduced a PIC method on triangular meshes for simulating the movement of charged particles in magnetostatic fields, which offers better management of curved boundaries. Ando et al. [2013] utilized a FLIP method based on tetrahedron mesh discretization to efficiently simulate liquids with high spatial adaptivity. Moreover, Ando et al. [2015] implemented a stream function to enforce incompressibility, without relying on pressure projection. Inspired by these developments, we propose a new PIC/FLIP simulation scheme based on a triangular mesh. This approach involves simulating the evolution of the stream function and dilation to achieve a physically realistic simulation of soap films.

3 PHYSICAL MODEL

In this section, we introduce our thin-film model with a primary focus on preserving vortex structure by enhancing tangential flow evolution. Following this, we integrate these improvements with normal evolution, resulting in a comprehensive dynamic model.

Our approach begins with the 3D Navier-Stokes equations, employing a leading-order approximation under the lubrication assumption, as demonstrated by Huang et al. [2020]. Additionally, we adopt the tangential-normal decoupling method which was proposed by Deng et al. [2022]; Ishida et al. [2020]. While building on these established strategies, the distinctive contribution of this work lies in the treatment of the tangential equation.

\[
\frac{Du^\top}{Dt} = -\frac{2RT\rho\eta}{\rho} \nabla \cdot \Gamma + \nu \nabla^2 u^\top + \frac{1}{\rho} f^\top_{\text{ext}},
\]  

(1)

where \(u^\top, f^\top_{\text{ext}}\) represents the tangential fluid velocity and external force (only gravity considered), and \(R, T, \rho, \eta, \Gamma\) and \(\nu\) denotes the ideal gas constant, temperature, fluid density, film thickness, surfactant concentration and kinematic viscosity respectively.

The accurate solution of Eq. (1) is the key to recreating soap films’ turbulent flows and intricate visual details, and a demanding numerical challenge that have been tackled with grid-based [Huang et al. 2020], mesh-based [Ishida et al. 2020], and particle-based [Deng et al. 2022] methods. Nevertheless, these existing methods are all velocity-based, whose susceptibility to numerical viscosity renders them suboptimal in resolving soap film dynamics which is largely inviscid and vortical due to the thin geometries.

3.1 Circulation

In face of this conundrum, one promising approach is to reformulate the velocity-based Eq. (1) into a vorticity-based form which tracks \(\omega = \nabla \times u\) as its primary variable, as inspired by the rich literature of vortex methods in computer graphics [Azencot et al. 2014; Barnat and Pollard 2012; Elcott et al. 2007; Pfaff et al. 2012; Selle et al. 2005; Weißmann and Pinkall 2010; Xiong et al. 2021]. However, despite tracking the vorticity \(\omega\) can naturally preserve vortex structures and resist numerical diffusion, it is not straightforward to directly apply the curl operator, as defined in three-dimensional space, to our thin-film context.

In order to leverage the advantages in maintaining vorticity \(\omega\) without the need for a manifold curl operator, we employ a third variable which is the circulation \(\Omega\) [Cottet et al. 2000; Elcott et al. 2007]. The introduction of \(\Omega\) effectively reconciles this conundrum, because on one hand, \(\Omega\) relates directly to \(\omega\) as its integral based on Stokes’ Theorem, so modeling the evolution of circulation contributes to effectively maintaining vorticity. On the other hand, \(\Omega\) relates to velocity \(u\) as its line integral around closed paths, so that the curl operator is not needed. Particularly:

\[
\Omega = \oint_{\varphi} u \cdot dl = \iint_{S_{\varphi}} (\nabla_{\perp} \cdot J u^\top) \cdot dS,
\]

(2)

where the operator \(J\) rotates the vector by \(\pi/2\) in the order followed by the crawling line integral and \(S_{\varphi}\) denotes the area enclosed by the closed curve \(\varphi\).

Having established \(\Omega\) as our primary tracked variable, the next step is to derive its governing equations by combining Eq. (1) and Eq. (2). Given that the soap film is significantly thin compared to the other dimensions, it is reasonable to assumed that the closed loop predominantly aligns with the tangential direction of the film and the circulation is primarily determined by the tangential velocity \(u^\top\).

Therefore, the evolution equation for the circulation \(\Omega\) is derived as...
follows:

\[
\frac{D\Omega}{Dt} = \oint_{\partial S} \mathbf{J} \cdot dS + \oint_S \mathbf{u} \cdot \frac{D(dl)}{Dt} = -\frac{2RT}{\rho} \oint_{\partial S} \nabla \Gamma \cdot dS + \nabla^2 \Omega. \tag{4}
\]

The second term of Eq. (3) equals zero according to Cottet et al. [2000]. Notably, in the absence of viscosity and surfactant concentration gradients, Eq. (4) is consistent with Kelvin’s circulation theorem. Additionally, the closed loop integral of gravity in the first term of Eq. (3) equals to 0 as well.  

3.2 Dilatation

Tracking the circulation \( \Omega \) in place of velocity \( \mathbf{u} \) allows us to effectively isolate and capture the fluid’s rotational motion, but having \( \Omega \) alone is not enough for reconstructing the full velocity field \( \mathbf{u} \). To see this, consider the Helmholtz-Hodge decomposition theorem on a 2-manifold embedded in \( \mathbb{R}^3 \) [Azencot et al. 2014; Bhatia et al. 2013; Elcott et al. 2007; Shi and Yu 2004] which states that the tangential velocity field \( \mathbf{u}^\top \) can be expressed as:

\[
\mathbf{u}^\top = \nabla \Psi + J \mathbf{V} + \mathbf{h}, \tag{5}
\]

where \( \nabla \Psi \) and \( J \mathbf{V} \) are the surface gradient and curl of two scalar potentials, and \( \mathbf{h} \) is the harmonic component of the velocity field. In particular, both \( \nabla \Psi \) and \( \mathbf{h} \) are curl-free components which consequently do not contribute to the circulation \( \Omega \). As a result, beyond maintaining \( \Omega \), additional information is needed for recovering the curl-free components \( \nabla \Psi \) and \( \mathbf{h} \). In this paper, under the assumption of a simply-connected domain such as a soap film, we disregard the dynamics of the harmonic component \( \mathbf{h} \). However, Yin et al. [2023] have theoretically discussed that \( \mathbf{h} \) cannot be ignored for surfaces with more complex boundary conditions.

Now the only unmodeled component lies in the divergent field \( \nabla \mathbf{V} \). We note that such a component is non-zero despite how we model incompressible fluid, since it behaves like a compressible medium when observed from the tangential domain due to the ability to thicken and thin. Our approach to calculate the divergent component involves dealing with the velocity divergence, represented as dilatation \( \partial = \nabla \cdot \mathbf{u}^\top \) on a surface. We treat this quantity as a separately monitored variable that evolves over time. We derive its governing equation by computing the divergence of Eq. (1), which results in:

\[
\frac{d\partial}{dt} = -\nabla \cdot (\mathbf{u} \cdot \nabla \mathbf{u}^\top) - \frac{2RT}{\rho} \nabla \cdot (\frac{\nabla \Gamma}{\eta}) + \frac{1}{\rho} \nabla \cdot \mathbf{f}_{ext}^\top + \nabla^2 \partial. \tag{6}
\]

3.3 Velocity Reconstruction

At this point, we already have at our hand \( \Omega = \int_{\partial S_c} (\nabla \cdot (\mathbf{u} \cdot \mathbf{u}^\top)) \cdot dS \) which corresponds to the rotational component: \( \nabla \mathbf{V}, \) along with \( \partial = \nabla \cdot \mathbf{u}^\top \) which corresponds to the irrotational component: \( \nabla \Phi. \)

The next step is to use both quantities to compute the scalar fields \( \Phi \) and \( \Psi \) by solving two Poisson equations:

\[
\begin{align*}
\nabla^2 \Phi &= \nabla \cdot \mathbf{u}^\top, \tag{7a} \\
\nabla^2 \Psi &= -\nabla \cdot J \mathbf{u}^\top, \tag{7b}
\end{align*}
\]

Fig. 4. Our geometry discretization consists of Lagrangian particles \( P \), a triangle mesh \( T \) and its dual mesh \( \mathcal{D} \).

where \( \nabla^2 \) represents the surface Laplacian operator. The right-hand-side of Eq. (7a) and Eq. (7b) can be simply derived by the relationships \( \nabla \cdot \mathbf{u}^\top = \partial \) and \( \nabla \cdot J \mathbf{u}^\top = \text{lim}_{l \to 0} \Omega / \partial. \) It should be noted that Eq. (7b) corresponds to the stream function formulation used by Ando et al. [2015]; Azencot et al. [2014]; Elcott et al. [2007]. Solving for \( \Phi \) and \( \Psi \) in this way allows us to ultimately reconstruct the full tangential velocity field according to Eq. (5).

3.4 Soap Film Dynamics Model

In summary, through exploiting the vortex formulation of the Navier-Stokes equations on manifold, we propose a novel soap film dynamics model as:

\[
\begin{align*}
\frac{D\mathbf{u}}{Dt} &= \frac{1}{\rho} \left( -\Delta \mathbf{u} + 2 \left( \alpha_0 - RT \right) \mathbf{H} \right) \mathbf{n} + \mathbf{f}_{ext}^\top, \tag{8a} \\
\frac{D\Omega}{Dt} &= -\frac{2RT}{\rho} \oint_{\partial S} \nabla \Gamma \cdot dS + \nabla^2 \Omega, \tag{8b} \\
\frac{\partial \partial}{\partial t} &= -\nabla \cdot (\mathbf{u} \cdot \nabla \mathbf{u}^\top) - \frac{2RT}{\rho} \nabla \cdot \left( \frac{\nabla \Gamma}{\eta} \right) + \frac{1}{\rho} \nabla \cdot \mathbf{f}_{ext}^\top + \nabla^2 \partial, \tag{8c} \\
\nabla^2 \Phi &= \partial, \tag{8d} \\
\nabla^2 \Psi &= -\nabla \cdot J \mathbf{u}^\top, \tag{8e} \\
\mathbf{u}^\top &= \nabla \cdot \mathbf{V} + J \mathbf{V}, \tag{8f} \\
\frac{\partial l}{\partial t} &= -\Gamma \partial, \tag{8g} \\
\frac{\partial l}{\partial t} &= -\eta \partial, \tag{8h}
\end{align*}
\]

where superscript \( \perp \) is used to represent normal components, \( \Delta \rho \) represents the pressure difference through the film surface, \( \alpha_0 \) signifies the surface tension of pure water, \( H \) and \( \mathbf{n} \) are the mean curvature and normal vector on the interfaces. Within our model, the evolution of the normal components (Eq. (8a)), the surfactant concentration (Eq. (8g)) and the film thickness (Eq. (8h)) continue to follow the original forms established by Deng et al. [2022]; Ishida et al. [2020]. Our model stands out from previous works due to the inclusion of the vortex-based evolution equation for tangential velocity (Eq. (8b) and the incorporation of the dilation (Eq. (8c))

4 NUMERICAL ALGORITHMS

We propose a vortex particle-on-mesh method for dynamic interface tracking and the intricate surface flow simulation.
4.1 Discrete Geometry Representation
Our geometry discretization consists of a set of Lagrangian particles \( \mathcal{P} \), a triangle mesh \( \mathcal{T} \) and dual mesh \( \mathcal{D} \), as depicted in Fig. 4.

- The Lagrangian particles \( \mathcal{P} \) carry circulations \( \Omega \), velocities \( \mathbf{u} \), surfactant concentration \( \Gamma \) and thickness \( \eta \) advected by the local fluid velocities, transporting matters and momentum on the thin film.
- The triangle mesh \( \mathcal{T} \) is responsible for tracking the underlying geometry, and providing a stable computational stencil for updating the circulation \( \Omega \) and dilatation \( \vartheta \) by chemical and external forces, which we will store on the its vertices.
- The dual mesh \( \mathcal{D} \) is the Voronoi dual of the triangle mesh \( \mathcal{T} \). Geometrically, \( \mathcal{T} \)'s vertices are the circumcenters of \( \mathcal{T} \)'s triangles, and its elements are the Voronoi regions of \( \mathcal{T} \)'s vertices. Dynamically, \( \mathcal{D} \) is in charge of storing the velocity variable \( \mathbf{u}^\top \). Such an assignment arise from the fact that we store \( \Omega \) and \( \vartheta \) on the vertices of mesh \( \mathcal{T} \), and these scalar fields naturally reconstruct the vector field \( \mathbf{u}^\top \) on its the vertices of its dual \( \mathcal{D} \), as detailed in the theory of discrete exterior calculus [Crane et al. 2013; Elcott et al. 2007].

To facilitate clarity, we adopt the following notations: subscript \( p \) for attributes store on Lagrangian particles \( \mathcal{P} \), subscript \( i \) for attributes store on the vertices of triangle mesh \( \mathcal{T} \), and subscript \( d \) for attributes store on the vertices of \( \mathcal{D} \).

4.2 Time Integration
In each timestep, our algorithm solve Eq. (8a) to Eq. (8h) sequentially through the following 6 steps, as illustrated in Fig. 5. We will explain each of these steps in detail in the following subsections, where the superscripts \( n \) and \( n+1 \) are exclusively employed to denote the time step index.

1. **Particle-mesh interpolation**: From particles, transfer surfactant concentration \( \Gamma_p^n \), film thickness \( \eta_p^n \), circulation \( \Omega_p^n \) to mesh \( \mathcal{T} \), and tangential velocity \( (\mathbf{u}_p^n)^\top \) to dual mesh \( \mathcal{D} \). Calculate the dilatation \( \vartheta_p^n \) on \( \mathcal{T} \).
2. **Normal advection**: Evolve mesh \( \mathcal{T} \), dual mesh \( \mathcal{D} \) and particles \( \mathcal{P} \) in the normal direction by Eq. (8a).
3. **Circulation-dilatation updates**: Solve Eq. (8b) and Eq. (8c) to update circulation to \( \Omega_i^{n+1} \) and dilatation to \( \vartheta_i^{n+1} \) on mesh \( \mathcal{T} \).
4. **Velocity reconstruction & Boundary conditions**: Solve Eq. (8d), Eq. (8e) and Eq. (8f) to update tangential velocity \( (\mathbf{u}_i^{n+1})^\top \) of dual mesh \( \mathcal{D} \) with boundary conditions, also update circulation to \( \Omega_i^{n+1} \) and dilatation to \( \vartheta_i^{n+1} \) on mesh \( \mathcal{T} \).
5. **Mesh-particle interpolation**: Transfer circulation \( \Omega_p^n \), dilatation \( \vartheta_p^n \) on mesh \( \mathcal{T} \), and tangential velocities \( (\mathbf{u}_p^n)^\top \) on dual mesh \( \mathcal{D} \) to the particles \( \mathcal{P} \) to get \( \Omega_p^{n+1} \), \( \vartheta_p^{n+1} \) and \( (\mathbf{u}_p^{n+1})^\top \).
6. **Particle updates**: Update surfactant concentration to \( \Gamma_p^{n+1} \) and film thickness \( \eta_p^{n+1} \) on particles \( \mathcal{P} \) by Eq. (8g) and Eq. (8h), and advect their positions.

4.3 Particle-Mesh Transfer
For a generic fluid variable \( q \), we interpolate \( q \) from particles \( \mathcal{P} \) at a mesh vertex \( i \in \mathcal{T} \) with the following equation:

\[
q_i = \sum_{p \in N_p^i} \frac{W(||x_i - x_p||, r_p)}{\sum_{p' \in N_p^i} W(||x_i - x_p'|| r_p)} q_p,
\]

where \( q_i \) and \( q_p \) denote values of \( q \) on mesh vertex \( i \) and particle \( p \), \( r \) the support radius, \( N_p^i \) the particles within radius \( r \) of vertex \( i \), and \( x_i \) and \( x_p \) the vertex and particle positions respectively. Taking into account the smoothing effect of surface tension on soap films, which results in minimal curvature, we employ the 3D Euclidean distance metric for computational simplicity. The function \( W \) is an

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Storage Locations</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \Gamma )</td>
<td>✓</td>
</tr>
<tr>
<td>( \eta )</td>
<td>✓</td>
</tr>
<tr>
<td>( \Omega )</td>
<td>✓</td>
</tr>
<tr>
<td>( \mathbf{u}^\top )</td>
<td>✓</td>
</tr>
<tr>
<td>( \vartheta )</td>
<td>✓</td>
</tr>
<tr>
<td>( v_{ij} )</td>
<td>✓</td>
</tr>
<tr>
<td>( w_{ij} )</td>
<td>✓</td>
</tr>
</tbody>
</table>

Fig. 5. The computation workflow of a single simulation step in our proposed vortex particle-on-mesh framework.
We advance mesh \( \mathcal{P} \) to \( \mathcal{T} \) following the equation:

\[
s_i = \sum_{p \in N_i^e(p)} \frac{W(||x_i - x_p||, r)}{\sum_{l \in \mathcal{N}_i(l)} W(||x_i - x_l||, r)} s_p,
\]

where \( N_i^e(p) \) denotes the vertices within radius \( r \) from particle \( p \). The circulation \( \Omega^e \) is distributed from particle to mesh using Equation 11. For both types of abovementioned transfers, we define \( r \) as three times the length of the longest edge in the mesh at initialization.

For velocity transfer between particles \( \mathcal{P} \) and vertices of the dual mesh \( \mathcal{D} \), the following equation is implemented:

\[
(u_d^e)^n = \sum_{p \in N\mathcal{D}(d)} \frac{w_p}{\sum_{p \in N\mathcal{D}(d)} w_p} (u_p^n)^n,
\]

where \( w_p \) represents the barycentric coordinate of particle \( p \) within its respective Voronoi cell, computed using the method proposed by Warren et al. [2007]. Additionally, \( N\mathcal{D}(d) \) represents the particles that reside on any dual face containing the dual vertex \( d \).

After obtaining \((u_d^e)^n\) on the dual vertices, we compute \( \vartheta^p_i \) for mesh vertex-i using Stokes’ theorem as:

\[
\vartheta^p_i = \frac{1}{A} \sum_j f_{jk},
\]

where \( e_{jk} \) iterates through all the boundary edge of the mesh vertex’s Voronoi cell, \( A \) denotes the cell’s total area, and \( f_{jk} \) denotes the flux through edge \( e_{jk} \). We calculate \( f_{jk} \) as \( f_{jk} = -\frac{1}{2}((u_d^e)^p + (u_d^e)^n)^e \).

\( (e_{jk}) \), where \((u_d^e)^p \) and \((u_d^e)^n \) denote the tangential velocity of the two ending points of \( e_{jk} \), and the negative sign indicates that the outward flux is considered positive. Following the right-hand rule, the traversal direction of the boundary edges aligns with the normal direction of the mesh vertex.

### 4.4 Normal Advection

We advance mesh \( \mathcal{T} \) by solving Eq. (8a) in an explicit Eulerian integration style as proposed by Ihida et al. [2020], followed by a remeshing step [Brochu and Bridson 2009] to fix ill-conditioned triangles. We reconstruct dual mesh \( \mathcal{D} \) on updated \( \mathcal{T} \). To update particles \( \mathcal{P} \) in the normal direction, we first interpolate the normal velocities \( u_d^n \) and normal direction \( n_i \) on the not-yet-updated mesh \( \mathcal{T} \) to obtain \( u_p^n \) and \( n_p \) on particles \( \mathcal{P} \) using barycentric weights, and move particles by \( u_d^n \). To project the particles \( \mathcal{P} \) onto the updated mesh \( \mathcal{T} \), we perform a search in the normal direction \( n_p \) for each particle \( p \) to find the intersection with \( \mathcal{T} \), which is robust in the context of the smoothing soap film surfaces.

### 4.5 Circulation-Dilatation Updates

We solve Eq. (8b) and Eq. (8c) by operator-splitting [Stam 1999] to compute the evolution of circulation \( \Omega \) and dilatation \( \vartheta \) driven by chemical and external forces. In general, we first apply forces except the viscosity force to update \( \Omega^e \) and \( \vartheta^p \) to \( \Omega^e \) and \( \vartheta^p \). Then we apply the viscosity force to update them to \( \Omega^f \) and \( \vartheta^f \). The \( \Omega^{e+1} \) and \( \vartheta^{p+1} \) for the next step requires the boundary handling and advection, which are described in detail in the latter subsections. Here we only focus on the process before \( \Omega^f \) and \( \vartheta^f \).

**Update to \( \Omega^e \).** We discretize Eq. (8b) to update \( \Omega^e \) as:

\[
\frac{\Omega^e_i - \Omega_i^n}{\Delta t} = -\frac{2RT}{\rho} \frac{A_i}{A_T} \sum_j \frac{\Gamma^n_j - \Gamma_j}{\eta^n_j + \eta^n_{ij}} \cdot l_{jk}.
\]

The symbols are depicted on the right. \( e_{jk} \) represents one of the boundary edges in the 1-ring neighborhood of the vertex-i, with a length of \( l_{jk} \). The surfactant concentration and film thickness on the ending points of \( e_{jk} \) are denoted as \( \Gamma_j, \Gamma_k, \eta_j, \eta_k \), respectively, the Voronoi area associated with vertex-i is designated as \( A_i \), while \( A_T \) signifies the cumulative area of the triangles encompassed within the 1-ring neighborhood surrounding vertex-i.

**Update to \( \vartheta^p \).** Following Deng et al. [2022]; Huang et al. [2020], we update \( \vartheta \) by semi-implicitly integrating the surfactant concentration \( \Gamma \) on vertices of mesh \( \mathcal{P} \). The relation between the surfactant concentration \( \Gamma \) and dilatation \( \vartheta \) can be deduced from Eq. (8c) and Eq. (8g):

\[
\begin{align}
\frac{\vartheta^p - \vartheta^n}{\Delta t} &= -\frac{2RT}{\rho} \frac{\Gamma^n}{\eta^n} + \frac{1}{\rho} \nabla \cdot f_{ext}, \quad (15a) \\
\frac{\Gamma^n - \Gamma}{\Delta t} &= -\vartheta^n \cdot \Gamma^e. \quad (15b)
\end{align}
\]

This relation can be summarized to a single equation for \( \Gamma^e \):

\[
\frac{\Gamma^e - \Gamma^n}{\Delta t} = -\frac{2RT}{\rho} \frac{\nabla \cdot (\nabla \Gamma^e)}{\eta^n} = -\vartheta^n + \frac{1}{\Delta t} \nabla \cdot f_{ext}. \quad (16)
\]

The left-hand-side of Eq. (16) turns to be a sparse linear system which is symmetric positive definite for \( \Gamma^e \) after discretizing the \( \nabla \) operator as proposed by Crane et al. [2013]. Once \( \Gamma^e \) is solved, we evaluate the \( \vartheta^n \) for each vertex of mesh \( \mathcal{T} \) by Eq. (15b).

**Update to \( \Omega^f \) and \( \vartheta^f \).** Finally, Building upon the approach proposed by Elcott et al. [2007], viscous forces are further incorporated in a semi-implicitly manner as:

\[
\begin{align}
\frac{\Omega^f - \Omega^e}{\Delta t} &= \nu \nabla^2 \Omega^f, \quad (17a) \\
\frac{\vartheta^f - \vartheta^e}{\Delta t} &= \nu \nabla^2 \vartheta^f. \quad (17b)
\end{align}
\]

where \( \nabla^2 \) is the standard cotangent Laplacian matrix on meshes [Meyer et al. 2003].
where $e$ in the area of the triangle, and each edge should be equal to three edges of the triangle impose three constraints on the velocity along each edge as $\nabla \cdot \bm{s} = 0$.

In this subsection, we initially disregard the boundary conditions, and we transfer the updated circulation $\Omega_0$ at the vertex of the dual mesh, and can be expressed as:

$$\mathbf{w}_d = -\frac{1}{2A} (w_{02}l_{02}\mathbf{e}_{01} + w_{01}l_{01}\mathbf{e}_{20}),$$

where $w_{ij} = (\Psi_j - \Psi_i)/l_{ij}$. The final tangential velocity $(\mathbf{u}_d^*)^\top$ on dual mesh is computed as $(\mathbf{u}_d^*)^\top = \mathbf{v}_d + \mathbf{w}_d$.

4.7 Boundary Conditions

We incorporate two boundary conditions in our algorithm. The first one is the non-penetration Neumann boundary condition, which can be achieved by setting $\mathbf{n} \cdot \nabla \Phi = 0$ and $\mathbf{n} \cdot \mathbf{fV}_s \Psi = 0$ when solving Equations (8d) and (8e).

The second condition is the no-slip Dirichlet boundary condition, where we employ the Immersed Boundary Method [Morency et al. 2012], i.e. setting the tangential velocities inside the solid to be the same as the solid velocities. Note that the no-slip boundary condition affects not only the tangential velocities, but also the circulation $\Omega$ and dilatation $\vartheta$. Therefore, instead of directly modifying $\mathbf{u}^*$, we adjust the velocities $\mathbf{v}_{ij}$ and $\mathbf{w}_{ij}$ on edges in Eq. (18) and Eq. (19). Specifically, after the viscosity update, we set $\mathbf{v}_{ij}$ and $\mathbf{w}_{ij}$ inside the solid to be the projection of the solid velocity along the edge direction, which accordingly changes the tangential velocity to $(\mathbf{u}_d^*)^\top$. Moreover, we update the circulation and dilatation to $\Omega^\Sigma$ and $\vartheta^\Sigma$ by:

$$\begin{align*}
\Omega_i^\Sigma &= \sum_j w_{ij}l_{ij}^*, \\
\vartheta_i^\Sigma &= \frac{1}{2} \sum_j \mathbf{v}_{ij}l_{ij}^*
\end{align*}$$

where $A$ is the Voronoi cell area of vertex-$i$, $l_{ij}^*$ denotes the length of $\mathbf{e}_{ij}$'s dual edge.

4.8 Mesh-Particle Interpolation

We transfer the updated circulation $\Omega^\Sigma$, dilatation $\vartheta^\Sigma$ and tangential velocities $(\mathbf{u}_d^*)^\top$ to particles. To minimize numerical dissipation, we choose to transfer the change of circulation $\delta \Omega = \Omega^\Sigma - \Omega^\pi$ from mesh to particles as FLIP [Zhu and Bridson 2005]:

$$\delta \Omega_p = \frac{1}{\sum_{p \in N_p(i)} W(||\mathbf{x}_i - \mathbf{x}_p||, r)} \cdot \delta \Omega_i,$$

where $W(||\mathbf{x}_i - \mathbf{x}_p||, r)$ is the weight function.
We update the surfactant concentration $\Gamma$ with $\Omega$, where the definition of $\vartheta$ is the transport strength parameter, and $\nu \cdot (\vartheta^{p+1})$ is the dilatation. The final circulation $\Gamma^{n+1}$ is constructed from the mesh velocities by Eq. 13 in the subsequent timestep, rather than being transferred from particles. This completes our time integration.

5 NON-MANIFOLD

To model non-manifold geometries, we utilize a triangular mesh $\mathcal{T}_{\text{non}}$ and a set of vortex particles $\mathcal{P}_{\text{non}}$ for discretization, adopting a method similar to that proposed by Deng et al. [2022]. In this model, a non-manifold foam is viewed as a collection of manifold components, each triangular facet within $\mathcal{T}_{\text{non}}$ is labeled to show its association with a particular manifold. Most facets are linked to a single manifold, but those on shared surfaces carry markings that indicate their connection to adjacent manifolds. These collectively marked facets for the same manifold $i$ constitute the mesh $\mathcal{T}_{i_{\text{non}}}$ on which a well-defined dual mesh $\mathcal{D}_{i_{\text{non}}}$ is established. Similarly, vortex particles are assigned to specific manifolds, which organizes them into distinct groups $\mathcal{P}_{i_{\text{non}}}$ for each manifold $i$. Below, we outline the simulation process for a non-manifold over a time step:

1. **Normal Evolution:** The method proposed by Ishida et al. [2020] is applicable to the normal evolution of non-manifold geometries and can be directly applied to our mesh $\mathcal{T}_{\text{non}}$. Afterward, we perform the remeshing and project the particles $\mathcal{P}_{\text{non}}$ onto the mesh $\mathcal{T}_{\text{non}}$ as described in Section 4.4.
2. **Tangential Evolution:** For each manifold $i$, we integrate time as outlined in Section 4.4 (excluding the normal evolution described in Section 4.4) using $\mathcal{T}^{i}_{\text{non}}$, $\mathcal{D}^{i}_{\text{non}}$, and $\mathcal{P}^{i}_{\text{non}}$.
3. **Material Exchange:** When particles from manifold $i$ reach a facet $j$ located on a shared surface between manifolds $i$ and $j$, there is a possibility they could be assignable to manifold $j$. This procedure aims to promote the movement of material from regions with high surfactant concentrations to those with lower concentrations. As a result, we have introduced a probabilistic approach for the migration of vortex particles, inspired by Deng et al. [2022], detailed as follows:

$$C = \psi \cdot \left(1 - \min \left(1, \frac{1}{\Gamma^{i}_{\text{non}}}, \frac{1}{\Gamma^{j}_{\text{non}}} \right). \right.$$ (23)

In this equation, $\psi$ is the transport strength parameter, and $\Gamma^{i}_{\text{non}}$ and $\Gamma^{j}_{\text{non}}$ denote the surfactant concentrations at the locations of the particles on manifolds $i$ and $j$, respectively. The probability of a particle being reassigned to manifold $j$ is calculated using $\min(1, C)$.

6 RESULTS

6.1 Validation

**Gravity Drainage.** This phenomenon underscores the significance of considering velocity divergence for the accurate simulation of soap films, which is also the key to distinguishing our method from other 2D incompressible Navier-Stokes solvers [Azencot et al. 2014; Elcott et al. 2007; Shi and Yu 2004]. When the film is positioned vertically, the fluid gradually flows downwards due to its own weight, resulting in an increased thickness at the bottom. It’s conceivable that the divergence of the velocity field is negative at the top and positive at the bottom. As a result, the surfactant gradually accumulates at the bottom, giving rise to an upward Marangoni force to counteract gravity, and eventually achieving equilibrium in this process. The steady-state thickness profile, derived by Couder et al. [1989], takes the form $\eta(z) = \eta_0 e^{-(z-z_0)/\delta} a^2$, where $\eta_0$ denotes the film thickness when it is laid flat. By setting $\eta_0 = 400$ nm, we confirm the alignment of our simulation result with the analytical solution, as illustrated on the right side of Fig. 8. Its exponential.

![Fig. 8. Gravity Drainage. Left: Our method can accurately simulate the changes in film thickness caused by gravity drainage and generates Newton’s interference fringes. Right: The steady-state thickness profile of our simulation result (red dot) is highly consistent with the analytical solution (blue curve) proposed by Couder et al. [1989].](image1)

![Fig. 9. Von Kármán Vortex Street. After the steady incoming flow passes through the solid, recurrent swirling vortices are generated behind the solid using our method.](image2)
variation in thickness produces characteristic Newton’s interference fringes, depicted as color stripes on the left side of Fig. 8. In contrast, two-dimensional incompressible fluid solvers only permit a divergence-free velocity field on their surfaces [Azencot et al. 2014; Elcott et al. 2007; Shi and Yu 2004], which hinders the evolution of mass and surfactant concentration, making it impossible to achieve physically realistic results.

Von Kármán vortex street. As depicted in Fig. 9, a stick with a radius of $0.15 \text{ m}$ is vertically inserted into one end of a soap film. The film is subjected to open boundary conditions, and a steady flow (from right to left with a velocity of $2.5 \text{ m/s}$) maintains on the film. The recurrent formation of swirling vortices behind the stick, recognized as the von Kármán vortex street, verified the no-slip boundary condition at the fluid-structure coupling boundary are correctly handled in our method.

6.2 Comparison

6.2.1 Comparison with soap film simulation approaches. We conduct a qualitative comparison between the existing particle-based [Deng et al. 2022] and mesh-based [Ishida et al. 2020] approaches to soap film simulations, focusing on the preservation of tangential vortex structures. This comparison is conducted through two commonly used benchmark tests. To ensure a fair comparison, both our method and mesh-based method [Ishida et al. 2020] are performed using the same underlying mesh. As for the pure particle-based method [Deng et al. 2022], since the role of its E and L particles are equivalent to the mesh vertices and particles in ours, we adjust the number of E and L particles to be approximately equal to the number of mesh vertices and particles in our method, respectively, the computational details can be found in Table 2.

Taylor Vortex. In this experiment, we place two vortices in close proximity using the configuration outlined by McKenzie [2007]. The subsequent separation or merging of these vortices during the simulation depends on their initial distance, which we specified as $0.81 \text{ m}$, slightly exceeding the critical separation distance. As depicted in Fig. 10, both Deng et al. [2022] and Ishida et al. [2020] struggle to effectively separate the two vortices, whereas our approach successfully achieves their separation. The specific simulation configuration for the three compared methods comprises approximately 4k mesh vertices (E particles) and 8w vortex particles (L particles).

Leapfrog. In the classic 2D leapfrog experiment depicted in Fig. 11, we place four point vortices centered at coordinates $x = -1.5$ and $y = [-0.6, 0.6, -0.2, 0.2]$. These vortices are of equal strength, with the upper two being positive and the lower two being negative. Our approach demonstrates a remarkable ability to preserve these vortices for longer durations when compared to both [Ishida et al. 2020] and [Deng et al. 2022]. The specific simulation configuration for the three compared methods consist of approximately 8k mesh vertices (E particles) and 16w vortex particles (L particles).

6.2.2 Comparison with Vortex Method. We also performed a comparative evaluation of our method against the established state-of-the-art vorticity-streamfunction solver for triangular meshes, as proposed by Azencot et al. [2014], in tangential flow simulations. We implemented this method in C++ and evaluated its performance in two scenarios: first, where the mesh resolution was equivalent to
A Vortex Particle-on-Mesh Method for Soap Film Simulation

5.1 Comparison with Vortex Method

(a) Azencot et al. [2014]'s method with matching mesh resolution to ours
(b) Azencot et al. [2014]'s method with the number of mesh vertices was comparable to the number of our vortex particles
(c) our method

Fig. 12. Comparison with Vortex Method. The top result is simulated using the method proposed by Azencot et al. [2014], which is based on a triangular mesh with 4225 vertices and 8192 faces. The middle result also utilizes the method proposed by Azencot et al. [2014], but with a finer triangular mesh with 66049 vertices and 131072 faces. The bottom result is simulated using our method, operating on a triangular mesh with 4225 vertices, 8192 faces, and 81920 vortex particles.

ours, and second, where the number of mesh vertices was comparable to the number of our vortex particles. Both tests were performed during simulations of the Taylor vortex phenomenon. As depicted in Fig. 12, in the first scenario, our method delivered a superior visual representation. Although the visual results were comparable in the second scenario, our method demonstrated greater computational efficiency, as indicated in Table 3. These outcomes underscore the effectiveness and efficiency of our hybrid particle-mesh technique for simulating vortices.

6.3 Examples

The comprehensive specifications for all the examples simulated by our proposed system, along with the details of the computational resources employed, can be found in Table 4. The photorealistic rendering is accomplished within Houdini [SideFX 2023], with color derived from thin film interference [GameDev.net 2013].

Deforming Hemisphere. A hemispherical soap bubble with a radius of 0.5 m undergoes free vibrations in response to its initial deformation. The initial deformation, surface thickness and surfactant concentration all conform to a 1-octave Perlin noise distribution, reflected in its smooth shape and color change (see upper-left of Fig. 1a). Concurrently driven by the gravity and Marangoni force, the bubble surface exhibits rich surface flows, as demonstrated in Fig. 1b. Additionally, the soap bubble gradually thickens from top to bottom, with the upper end displaying a golden hue and the lower end exhibiting a green shade, as shown in lower-right of Fig. 1a.

Giant Bubble. Following a initial configuration akin to the hemispherical bubble example, our method replicates intricate surface flow patterns over a giant spherical bubble as shown in Fig. 13.

Deforming Rectangle. A rectangular soap film (4 m by 2 m) maintains a uniform surfactant concentration and a smooth thickness transition from $\eta \approx 400$ nm at the top to $\eta \approx 200$ nm at the bottom. Initially, both the soap film's thickness and surfactant concentration feature Perlin noise perturbations (top of Fig. 14a). As the simulation progresses, under the interplay of gravity and Marangoni force, a multitude of vortex structures (see the middle two figures of Fig. 14a) evolve on the film's surface due to the interplay between gravity and the Marangoni force. Ultimately, the film reaches a stable equilibrium state, with the thickness gradually increasing from top to bottom, as is shown in the lower part of Fig. 14a.

Rayleigh-Taylor Instability. In the initial configuration, a thin film is divided into two layers, with the upper half having a greater thickness and lower surfactant concentration, while the lower half having a thinner layer and higher surfactant concentration. At the boundary between these two layers, a fragile unstable equilibrium is established due to the interplay of the gravity and Marangoni forces. During the simulation, this local equilibrium is disrupted by the randomness in particle distribution, causing a parcel of denser fluid to move downward. As the denser material descends under the
while allowing the thickness of the lower layer to evolve dynamically. As illustrated in Fig. 3, the finger-like transformation of the interface leading to captivating turbulent vortices within the region where the film introduces a velocity disturbance at the fluid-solid interface, as documented by Shabalina et al. [2019].

Table 2. Simulation parameters for the comparisons with Deng et al. [2022]: Ishida et al. [2020]: the total computation time \( T(s) \) for simulating a five-second animation using the respective method, the number of vertices \( V \), the number of faces \( F \), the number of vortex particles \( N \) (for [Deng et al. 2022] we record the number of E particles and L particles instead).

<table>
<thead>
<tr>
<th>Figure</th>
<th>Scene</th>
<th>( T(s) )</th>
<th>( V )</th>
<th>( F )</th>
<th>( N )</th>
</tr>
</thead>
<tbody>
<tr>
<td>10a</td>
<td>Taylor vortex( ^\d )</td>
<td>96</td>
<td>4225</td>
<td>8192</td>
<td>-</td>
</tr>
<tr>
<td>10b</td>
<td>Taylor vortex( ^\d )</td>
<td>195</td>
<td>-</td>
<td>-</td>
<td>4096E+84100L</td>
</tr>
<tr>
<td>10c</td>
<td>Taylor vortex*</td>
<td>325</td>
<td>4225</td>
<td>8192</td>
<td>81920</td>
</tr>
<tr>
<td>11a</td>
<td>Leapfrog( ^\d )</td>
<td>226</td>
<td>8385</td>
<td>16384</td>
<td>-</td>
</tr>
<tr>
<td>11b</td>
<td>Leapfrog( ^\d )</td>
<td>756</td>
<td>-</td>
<td>-</td>
<td>8192E+168490L</td>
</tr>
<tr>
<td>11c</td>
<td>Leapfrog*</td>
<td>763</td>
<td>8385</td>
<td>16384</td>
<td>163840</td>
</tr>
</tbody>
</table>

\( ^\d \) simulated using method proposed by Ishida et al. [2020]. \( ^\d \) simulated using method proposed by Deng et al. [2022]. * simulated using our method.

Table 3. Simulation parameters and time statistics for comparisons with Azencot et al. [2014]: the computation time for each step \( T(s) \), the number of vertices \( V \), the number of faces \( F \), the number of vortex particles \( N \). The time step size has been uniformly set at \( h = 0.01 \) s.

<table>
<thead>
<tr>
<th>Fig.</th>
<th>Method</th>
<th>( T(s) )</th>
<th>( V )</th>
<th>( F )</th>
<th>( N )</th>
</tr>
</thead>
<tbody>
<tr>
<td>12a</td>
<td>[Azencot et al. 2014]</td>
<td>0.19</td>
<td>4225</td>
<td>8192</td>
<td>-</td>
</tr>
<tr>
<td>12b</td>
<td>[Azencot et al. 2014]</td>
<td>16.6</td>
<td>66049</td>
<td>131072</td>
<td>-</td>
</tr>
<tr>
<td>12c</td>
<td>Ours</td>
<td>0.65</td>
<td>4225</td>
<td>8192</td>
<td>81920</td>
</tr>
</tbody>
</table>

The soap layer using a method akin to the previous Stick experiment. The robot is kinematically controlled to propel itself forward, with its two legs periodically stepping on the film. Upon contact, the robot’s legs create instantaneous velocity perturbations in both the normal and tangential directions on the surrounding fluid film. This leads to the emergence of intriguing swirling patterns (see Fig. 15b) on the surface of the liquid film.

**Double Bubble.** Fig. 6 illustrates two bubbles, each with a radius of approximately 0.8 m, arranged to form a double bubble with a common surface. These bubbles start with velocities directed towards each other. The double bubble should have a uniform surfactant concentration, with its thickness gradually decreasing from about 500 nm in the upper region to about 100 nm at the bottom. Initially, both the thickness and the surfactant concentration are disturbed by Perlin noise. As the simulation progresses, driven by the Marangoni force, there is an exchange of material between the bubbles, creating complex surface flows as shown in Fig. 6.

**7 CONCLUSION, LIMITATION, AND FUTURE WORK**

We presented a physically-based vortex fluid model for soap films, adeptly addressing the intricate challenge of simulating visually complicated vortical structures on dynamically evolving thin films and their interactions with the 3D physical environment. By decomposing the film’s tangential velocity into circulation and dilatation components and advancing their evolution through a hybrid particle-mesh approach, our model bridges the gap between the previous thin-film models and vortex methods. Our framework facilitates a comprehensive integration of surfactant and thickness dynamics,
A Vortex Particle-on-Mesh Method for Soap Film Simulation

(a) Simulation process
(b) ZoomIn

Fig. 15. Water Strider. Left: A bipedal miniature robot periodically steps on the soap film, triggering intricate swirling patterns on liquid film surface. Right: The fine structure of the vortex flow in the black-framed regions.

Table 4. Simulation parameters for the examples: the number of vertices $V$, the number of faces $F$, the number of vortex particles $N$. We solve all of our linear system using ICPCG solver from Eigen library [Guennebaud et al. 2010]. Simulation time per step for all the examples was recorded on a desktop equipped with an AMD Ryzen™ 7950X processor.

<table>
<thead>
<tr>
<th>Figure</th>
<th>Scene Description</th>
<th>Time Step(s)</th>
<th>V</th>
<th>F</th>
<th>N</th>
<th>Simulation Time/Step(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>Gravity Drainage</td>
<td>1e-2</td>
<td>1261</td>
<td>2400</td>
<td>24000</td>
<td>0.16</td>
</tr>
<tr>
<td>9</td>
<td>Von Kármán vortex street</td>
<td>1e-2</td>
<td>24929</td>
<td>49152</td>
<td>491520</td>
<td>3.55</td>
</tr>
<tr>
<td>13</td>
<td>Giant Bubble</td>
<td>1e-2</td>
<td>163842</td>
<td>327680</td>
<td>3276800</td>
<td>77.52</td>
</tr>
<tr>
<td>14</td>
<td>Deforming Rectangle</td>
<td>1e-2</td>
<td>131841</td>
<td>262144</td>
<td>2621440</td>
<td>66.30</td>
</tr>
<tr>
<td>1</td>
<td>Deforming Hemisphere</td>
<td>1e-2</td>
<td>82177</td>
<td>163584</td>
<td>1635840</td>
<td>17.43</td>
</tr>
<tr>
<td>3</td>
<td>Rayleigh-Taylor Instability</td>
<td>1e-3</td>
<td>66177</td>
<td>131072</td>
<td>1310720</td>
<td>9.6</td>
</tr>
<tr>
<td>2</td>
<td>Stick</td>
<td>1e-2</td>
<td>123713</td>
<td>245760</td>
<td>2457600</td>
<td>37.27</td>
</tr>
<tr>
<td>15</td>
<td>Water Strider</td>
<td>1e-2</td>
<td>131841</td>
<td>262144</td>
<td>2621440</td>
<td>71.40</td>
</tr>
<tr>
<td>6</td>
<td>Double Bubble</td>
<td>1e-2</td>
<td>30632</td>
<td>62796</td>
<td>2752080</td>
<td>89.28</td>
</tr>
</tbody>
</table>

ensuring a high-fidelity Eulerian-Lagrangian representation capable of tackling the various interactions between the codimension-zero physical forces and codimension-one flow dynamics. In sum, our work marks a noticeable step toward the accurate simulation of thin film dynamics, offering state-of-the-art vortical detail and physical accuracy, and sets a new standard for realism in thin-film fluid simulation within computer graphics.

Our approach encounters several limitations. First, our current framework struggles with managing the topological evolution of soap film, a task that was successfully achieved by previous methods [Deng et al. 2022; Saye and Sethian 2013]. In order to enable more flexible topological evolution, the background mesh could be substituted with more adaptable geometric representations, such as point-set surfaces. This substitution necessitates a re-derivation of the circulation and dilatation evolution models for these novel data structures. Second, our current implementation of solid boundaries employs kinematic conditions only. Advancing this to incorporate two-way coupling models between thin-film vortices and lightweight objects is an exciting prospect, which would particularly focus on addressing surface-tension-dominated contact problems on thin-film surfaces. Third, maintaining our method’s numerical stability requires high mesh quality and a large number of particles to ensure accurate surface tension calculations and prevent thickness at mesh vertices from dropping to zero after particle-mesh interpolation. Finally, our method is limited to simply-connected regions with minimal curvature, such as bubbles, due to its reliance on 3D Euclidean distance and zero harmonic component. Future work could incorporate dynamics of harmonic components in non-simply-connected domains [Yin et al. 2023] and precise distance calculation on curved surface to broaden simulation capabilities.

In summary, we anticipate that vortex methods for thin films will open new frontiers in both film simulation and solid-fluid interaction research. Addressing these computational challenges promises to unlock a wide range of novel fluid phenomena that will greatly enrich the domain of graphical simulations.

ACKNOWLEDGMENTS

We thank the anonymous reviewers for their constructive comments. Authors from Peking University acknowledge the funding support by National Key R&D Program of China (2022ZD0160801) and Shenzhen Collaborative Innovation Program (CJGJZD202104048092601003). The Houdini Education license is credited for the video generations.
REFERENCES


ACM Trans. Graph., Vol. 43, No. 4, Article 53. Publication date: July 2024.