

A Vortex Particle-on-Mesh Method for Soap Film Simulation

NINGXIAO TAO, Yuanpei College, Peking University, China

LIANGWANG RUAN, School of CS & State Key Laboratory of General Artificial Intelligence, Peking University, China

YITONG DENG, Stanford University, United States of America

BO ZHU, Georgia Institute of Technology, United States of America

BIN WANG*, State Key Laboratory of General Artificial Intelligence, Beijing Institute for General Artificial Intelligence (BIGAI), China

BAOQUAN CHEN*, School of IST & State Key Laboratory of General Artificial Intelligence, Peking University, China

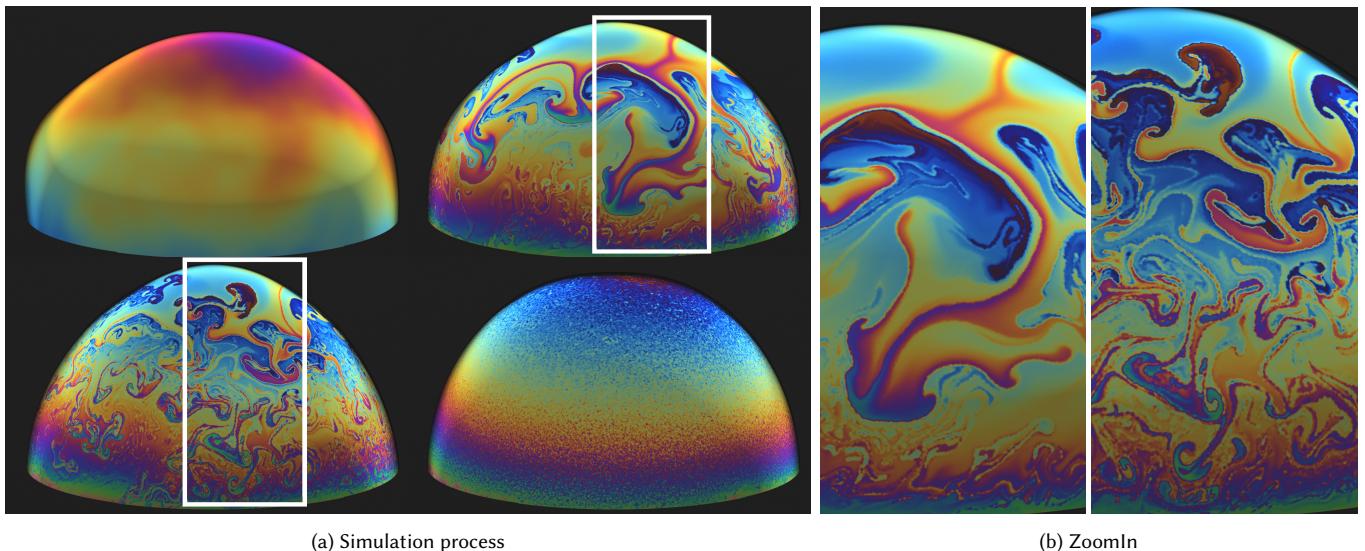


Fig. 1. Deforming Hemisphere. A free-vibrating hemispherical soap bubble (with a radius of 0.5 and the base fixed) is simulated with our method. Left: With the initial perturbation on physical attributes (thickness and surfactant concentration) and geometries, the bubble surface exhibits intricate surface flows during simulation and eventually reach to a equilibrium state, in which 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. The four stages of the simulation are sequentially represented by the four figures arranged as top-left, top-right, bottom-left, and bottom-right. Right: multiscale vortical structures captured at two snapshots during the simulation.

This paper introduces a novel physically-based vortex fluid model for films, aimed at accurately simulating cascading vortical structures on deforming

*corresponding authors

Authors' addresses: Ningxiao Tao, tnx123@stu.pku.edu.cn, Yuanpei College, Peking University, Beijing, China; Liangwang Ruan, ruanliangwang@pku.edu.cn, School of CS & State Key Laboratory of General Artificial Intelligence, Peking University, Beijing, China; Yitong Deng, yitongd@stanford.edu, Stanford University, Stanford, United States of America; Bo Zhu, bo.zhu@gatech.edu, Georgia Institute of Technology, Atlanta, United States of America; Bin Wang, binwangbuua@gmail.com, State Key Laboratory of General Artificial Intelligence, Beijing Institute for General Artificial Intelligence (BIGAI), Beijing, China; Baoquan Chen, baoquan@pku.edu.cn, School of IST & State Key Laboratory of General Artificial Intelligence, Peking University, Beijing, China.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

© 2024 Copyright held by the owner/author(s). Publication rights licensed to ACM. 0730-0301/2024/7-ART53 \$15.00 https://doi.org/10.1145/3658165

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 2012; Xiong et al. 2021], which are mainly designed for volumetric flows and not suited for thin film dynamics. In computer graphics, researchers have developed simulation algorithms based on differential geometry to create circulation-preserving incompressible flow solvers on curved surfaces (e.g., see [Elcott et al. 2007]). However, these approaches are often seen more as a projected 2D Navier-Stokes solver discretized on a manifold surface, rather than as comprehensive 3D thin-film fluid solvers capable of handling the co-evolution of physical ingredients like surfactant, thickness, film deformation, and various external forces. Consequently, most first-principle thin-film solvers (e.g., see [Ishida et al. 2020]) rely on velocity-form models to simulate interactions between the 3D fluid velocities defined on evolving thin films and various physical interactions surrounding them. The inherent inability of this approach to effectively represent, capture, and evolve vortical structures, compared to their vortex-oriented counterparts for volumetric fluid simulations, poses an exciting new challenge in achieving accurate and visually captivating vortex-driven film dynamics.

We propose a physically-based vortex fluid model for thin films, focusing on generating and evolving cascading vortical structures due to interactions between turbulent fluid flow on a deforming film

and physical forces from the three-dimensional world. Central to our method is a novel representation of the film's tangential velocity, which we decompose into *circulation* and *dilatation* components using Helmholtz-Hodge decomposition on 2-manifolds. Our approach employs a hybrid particle-mesh representation to evolve these dynamics. Initially, we employ a set of moving particles in tangential space to transport both circulation and dilatation. Subsequently, we transfer these properties onto a triangle mesh to compute the effects of various forces on soap films. The three-dimensional tangential velocity can be effectively reconstructed from the co-evolved circulation and dilatation components by solving two joint Poisson equations. We further couple the dilatation variable with surfactant and thickness dynamics, along with the film's normal dynamics driven by surface tension, resulting in a fully integrated thin film system. This system is particularly adept at characterizing the interaction between in-plane vortices and out-of-plane physics, such as various forces and solid interactions from 3D environments. A key innovation is our dilatation dynamics model, enabling various vortex-environment interactions and allowing for the simulation of 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.

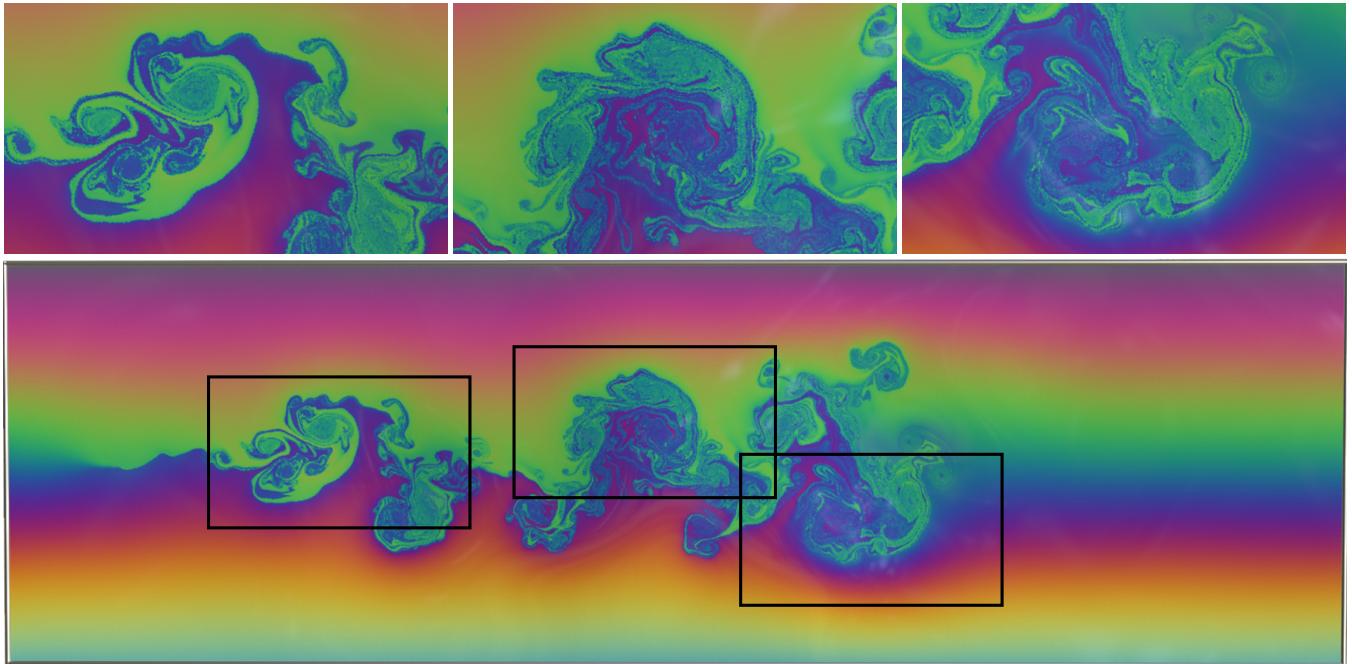


Fig. 2. Stick. A stick sweep across the film and introduces a velocity disturbance at the interface between the fluid and the solid, thereby spreading fascinating turbulence in the area where the stick is drawn (Bottom). Top: The turbulence details in the three black-framed region.

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

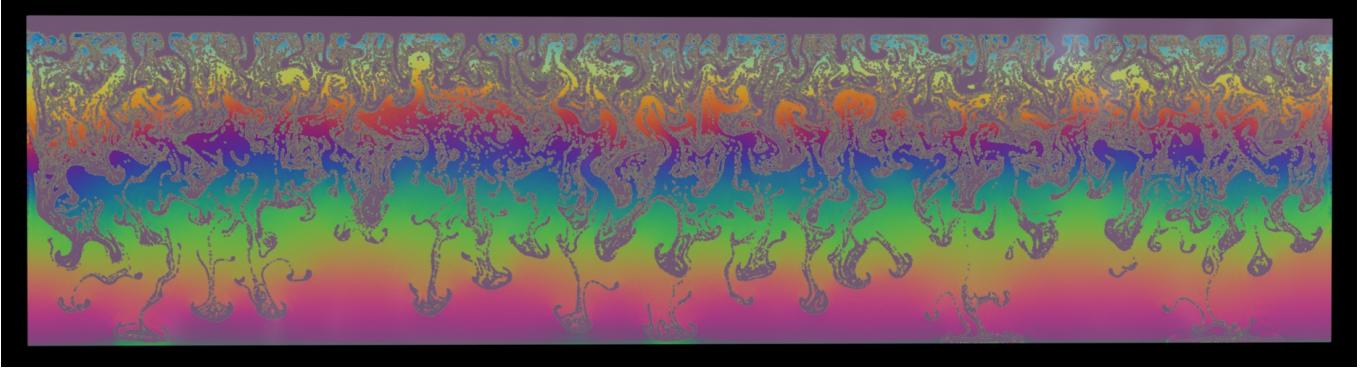


Fig. 3. Rayleigh-Taylor Instability. The dynamic development of finger-like patterns at the interface of two fluid layers with different density and surfactant concentrations, replicating observations from both real-world experiments and simulations.

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.

$$\frac{D\mathbf{u}^T}{Dt} = -\frac{2RT}{\rho\eta}\nabla_s\Gamma + \nu\nabla_s^2\mathbf{u}^T + \frac{1}{\rho}\mathbf{f}_{ext}^T, \quad (1)$$

where $\mathbf{u}^T, \mathbf{f}_{ext}^T$ represents the tangential fluid velocity and external force (only gravity considered), and R, T, ρ, η, Γ and ν 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 $\omega = \nabla \times \mathbf{u}$ 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 ω 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 ω without the need for a manifold curl operator, we employ a third variable which is the circulation Ω [Cottet et al. 2000; Elcott et al. 2007]. The introduction of Ω effectively reconciles this conundrum, because on one hand, Ω relates directly to ω as its integral based on Stokes' Theorem, so modeling the evolution of circulation contributes to effectively maintaining vorticity. On the other hand, Ω relates to velocity \mathbf{u} as its line integral around closed paths, so that the curl operator is not needed. Particularly:

$$\Omega = \oint_{\varphi} \mathbf{u} \cdot d\mathbf{l} = \iint_{S_{\varphi}} (\nabla_s \cdot J\mathbf{u}^T) \cdot dS, \quad (2)$$

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

Having established Ω 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 \mathbf{u}^T . Therefore, the evolution equation for the circulation Ω is derived as

follows:

$$\frac{D\Omega}{Dt} = \oint_{\varphi} \frac{D\mathbf{u}}{Dt} \cdot d\mathbf{l} + \oint_{\varphi} \mathbf{u} \cdot \frac{D(d\mathbf{l})}{Dt} \quad (3)$$

$$= -\frac{2RT}{\rho} \oint_{\varphi} \frac{\nabla_s \Gamma}{\eta} \cdot d\mathbf{l} + \nu \nabla_s^2 \Omega. \quad (4)$$

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 Ω in place of velocity \mathbf{u} allows us to effectively isolate and capture the fluid's rotational motion, but having Ω 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:

$$\mathbf{u}^\top = \nabla_s \Phi + J \nabla_s \Psi + \mathbf{h}, \quad (5)$$

where $\nabla_s \Psi$ and $J \nabla_s \Phi$ are the surface gradient and curl of two scalar potentials, and \mathbf{h} is the harmonic component of the velocity field. In particular, both $\nabla_s \Phi$ and \mathbf{h} are curl-free components which consequently do not contribute to the circulation Ω . As a result, beyond maintaining Ω , additional information is needed for recovering the curl-free components $\nabla_s \Phi$ 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 $\nabla_s \Phi$. 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 $\vartheta = \nabla_s \cdot \mathbf{u}^\top$ 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:

$$\begin{aligned} \frac{\partial \vartheta}{\partial t} &= -\nabla_s \cdot (\mathbf{u} \cdot \nabla_s \mathbf{u}^\top) - \frac{2RT}{\rho} \nabla_s \cdot \left(\frac{\nabla_s \Gamma}{\eta} \right) \\ &\quad + \frac{1}{\rho} \nabla_s \cdot \mathbf{f}_{ext}^\top + \nu \nabla_s^2 \vartheta. \end{aligned} \quad (6)$$

3.3 Velocity Reconstruction

At this point, we already have at our hand $\Omega = \iint_{S_\varphi} (\nabla_s \cdot J \mathbf{u}^\top) \cdot dS$ which corresponds to the rotational component: $J \nabla_s \Psi$, along with $\vartheta = \nabla_s \cdot \mathbf{u}^\top$ which corresponds to the irrotational component: $\nabla_s \Phi$.

The next step is to use both quantities to compute the scalar fields Φ and Ψ by solving two Poisson equations:

$$\begin{cases} \nabla_s^2 \Phi &= \nabla_s \cdot \mathbf{u}^\top \\ \nabla_s^2 \Psi &= -\nabla_s \cdot J \mathbf{u}^\top, \end{cases} \quad (7a)$$

$$(7b)$$

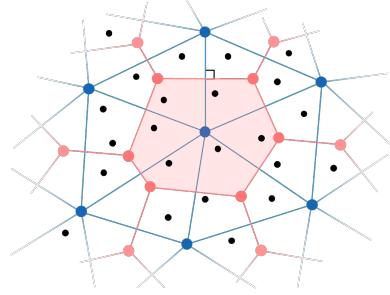


Fig. 4. Our geometry discretization consists of Lagrangian particles \mathcal{P} , a triangle mesh \mathcal{T} and its dual mesh \mathcal{D} .

where ∇_s^2 represents the surface Laplacian operator. The right-hand-side of Eq. (7a) and Eq. (7b) can be simply derived by the relationships $\nabla_s \cdot \mathbf{u}^\top = \vartheta$ and $\nabla_s \cdot J \mathbf{u}^\top = \lim_{S_\varphi \rightarrow 0} \Omega / S_\varphi$. 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 Ψ and Φ 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:

$$\frac{D\mathbf{u}^\perp}{Dt} = \frac{1}{\rho h} (-\Delta p + 2(\sigma_0 - RT\Gamma)H) \mathbf{n} + \frac{\mathbf{f}_{ext}^\perp}{\rho}, \quad (8a)$$

$$\frac{D\Omega}{Dt} = -\frac{2RT}{\rho} \oint_{\varphi} \frac{\nabla_s \Gamma}{\eta} \cdot d\mathbf{l} + \nu \nabla_s^2 \Omega, \quad (8b)$$

$$\left. \begin{aligned} \frac{\partial \vartheta}{\partial t} &= -\nabla_s \cdot (\mathbf{u} \cdot \nabla_s \mathbf{u}^\top) - \frac{2RT}{\rho} \nabla_s \cdot \left(\frac{\nabla_s \Gamma}{\eta} \right) \\ &\quad + \frac{1}{\rho} \nabla_s \cdot \mathbf{f}_{ext}^\top + \nu \nabla_s^2 \vartheta, \end{aligned} \right\} \quad (8c)$$

$$\nabla_s^2 \Phi = \vartheta \quad (8d)$$

$$\nabla_s^2 \Psi = -\nabla_s \cdot J \mathbf{u}^\top, \quad (8e)$$

$$\mathbf{u}^\top = \nabla_s \Phi + J \nabla_s \Psi, \quad (8f)$$

$$\frac{D\Gamma}{Dt} = -\Gamma \vartheta, \quad (8g)$$

$$\frac{D\eta}{Dt} = -\eta \vartheta, \quad (8h)$$

where superscript \perp is used to represent normal components, Δp represents the pressure difference through the film surface, σ_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.

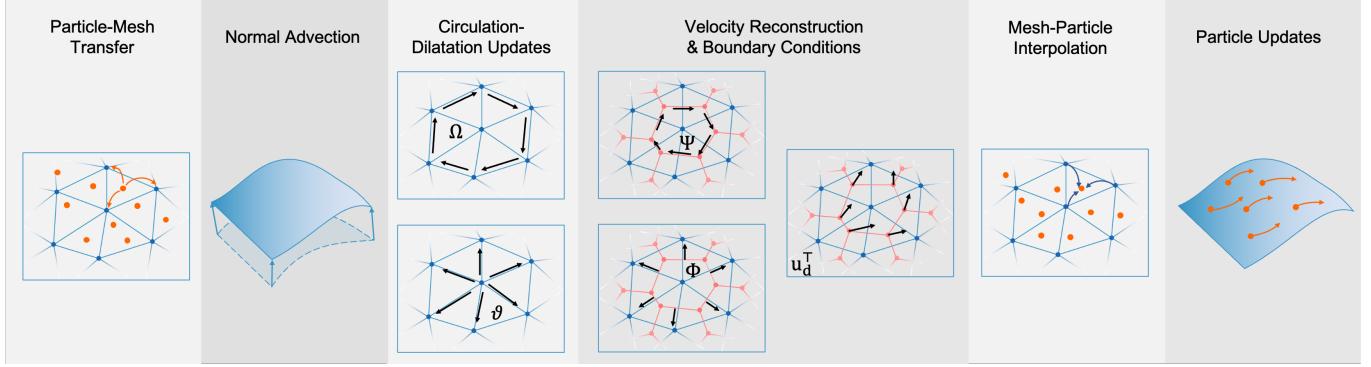


Fig. 5. The computation workflow of a single simulation step in our proposed vortex particle-on-mesh framework.

4.1 Discrete Geometry Representation

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

- The Lagrangian particles \mathcal{P} carry circulations Ω , velocities \mathbf{u} , surfactant concentration Γ and thickness η 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 Ω and dilatation ϑ 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{D} '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 Ω and ϑ 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 Γ_p^n , film thickness η_p^n , circulation Ω_p^n to mesh \mathcal{T} , and tangential velocity $(\mathbf{u}_p^\top)^n$ to dual mesh \mathcal{D} . Calculate the dilatation ϑ_i^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).

Table 1. Main symbols involved throughout the algorithm and their storage locations.

Symbol	Storage Locations			
	Particles	Vertices	Dual Vertices	Edges
Γ	✓	✓		
η	✓	✓		
Ω	✓	✓		
\mathbf{u}^\top	✓			✓
ϑ		✓		
v_{ij}				✓
w_{ij}				✓

- (3) **Circulation-dilatation updates:** Solve Eq. (8b) and Eq. (8c) to update circulation to Ω_i^{\ddagger} and dilatation to ϑ_i^{\ddagger} on mesh \mathcal{T} .
- (4) **Velocity reconstruction & Boundary conditions:** Solve Eq. (8d), Eq. (8e) and Eq. (8f) to update tangential velocity $(\mathbf{u}_d^\top)^{\ddagger}$ of dual mesh \mathcal{D} with boundary conditions, also update circulation to Ω_i^{\ddagger} and dilatation to ϑ_i^{\ddagger} on mesh \mathcal{T} .
- (5) **Mesh-particle interpolation:** Transfer circulation Ω_i^{\ddagger} , dilatation ϑ_i^{\ddagger} on mesh \mathcal{T} , and tangential velocities $(\mathbf{u}_d^\top)^{\ddagger}$ on dual mesh \mathcal{D} to the particles \mathcal{P} to get Ω_p^{\ddagger} , ϑ_p^{\ddagger} and $(\mathbf{u}_p^\top)^{\ddagger}$.
- (6) **Particle updates:** Update surfactant concentration to Γ_p^{n+1} and film thickness η_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:

$$q_i = \sum_{p \in N_r^P(i)} \frac{W(||\mathbf{x}_i - \mathbf{x}_p||, r)}{\sum_{p \in N_r^P(i)} W(||\mathbf{x}_i - \mathbf{x}_p||, r)} q_p, \quad (9)$$

where q_i and q_p denote values of q on mesh vertex i and particle p , r the support radius, $N_r^P(i)$ the particles within radius r of vertex i , and \mathbf{x}_i and \mathbf{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

SPH kernel function:

$$W(d, r) = \begin{cases} \frac{4}{\pi r^2} \left(1 - \frac{d^2}{r^2}\right)^3 & d < r, \\ 0 & \text{otherwise,} \end{cases} \quad (10)$$

defined in accordance with Adams and Wicke [2009]. Both the surfactant concentration Γ^n and thickness η^n are interpolated from \mathcal{P} to \mathcal{T} using Equation 9.

Analogously, a conservative fluid quantity s is distributed from particles \mathcal{P} to vertices of the triangle mesh \mathcal{T} with the following equation:

$$s_i = \sum_{p \in N_r^P(i)} \frac{W(||\mathbf{x}_i - \mathbf{x}_p||, r)}{\sum_{i \in N_r^P(p)} W(||\mathbf{x}_i - \mathbf{x}_p||, r)} s_p, \quad (11)$$

where $N_r^P(p)$ denotes the vertices within radius r from particle p . The circulation Ω^n is distributed from particle to the 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:

$$(\mathbf{u}_d^\top)^n = \sum_{p \in N^P(d)} \frac{w_p}{\sum_{p \in N^P(d)} w_p} (\mathbf{u}_p^\top)^n, \quad (12)$$

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, $N^P(d)$ represents the particles that reside on any dual face containing the dual vertex d .

After obtaining $(\mathbf{u}_d^\top)^n$ on the dual vertices, we compute ϑ_i^n for mesh vertex-i using Stokes' theorem as:

$$\vartheta_i^n = \frac{1}{A} \sum_{e_{jk}} f_{jk}, \quad (13)$$

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 $f_{jk} = -\frac{1}{2}((\mathbf{u}_j^\top)^n + (\mathbf{u}_k^\top)^n) \cdot (J e_{jk})$, where $(\mathbf{u}_j^\top)^n$ and $(\mathbf{u}_k^\top)^n$ 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 Ishida 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 \mathbf{u}_i^\perp and normal direction \mathbf{n}_i on the not-yet-updated mesh \mathcal{T} to obtain \mathbf{u}_p^\perp and \mathbf{n}_p on particles \mathcal{P} using barycentric weights, and move particles by \mathbf{u}_p^\perp . To project the particles \mathcal{P} onto the updated mesh \mathcal{T} , we perform a search in the normal direction \mathbf{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.

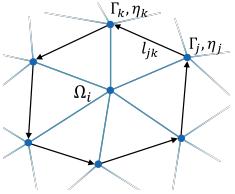
4.5 Circulation-Dilatation Updates

We solve Eq. (8b) and Eq. (8c) by operator-splitting [Stam 1999] to compute the evolution of circulation Ω and dilatation ϑ driven by chemical and external forces. In general, we first apply forces except the viscosity force to update Ω^n and ϑ^n to Ω^* and ϑ^* . Then we apply the viscosity force to update them to Ω^\dagger and ϑ^\dagger . The Ω^{n+1} and ϑ^{n+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 Ω^\dagger and ϑ^\dagger .

Update to Ω^ .* We discretize Eq. (8b) to update Ω as:

$$\frac{\Omega_i^* - \Omega_i^n}{\Delta t} = -\frac{2RT}{\rho} \cdot \frac{A_v}{At} \sum_{e_{jk}} \frac{\Gamma_k^n - \Gamma_j^n}{\frac{1}{2}(\eta_k^n + \eta_j^n)} \cdot l_{jk}. \quad (14)$$

The symbols are depicted on the right.



e_{jk} represents one of the boundary edge 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 Γ_j, Γ_k and η_j, η_k respectively, the Voronoi area associated with vertex i is designated as A_v , while A_t signifies the cumulative area of the triangles encompassed within the 1-ring neighborhood surrounding vertex i .

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

$$\left\{ \begin{array}{l} \frac{\vartheta^* - \vartheta^n}{\Delta t} = -\frac{2RT}{\rho} \nabla_s \cdot \left(\frac{\nabla_s \Gamma^*}{\eta^n} \right) + \frac{1}{\rho} \nabla_s \cdot \mathbf{f}_{ext}^\top, \\ \Gamma^* - \Gamma^n = -\Gamma^n \vartheta^*. \end{array} \right. \quad (15a)$$

$$\left\{ \begin{array}{l} \frac{\vartheta^* - \vartheta^n}{\Delta t} = -\frac{2RT}{\rho} \nabla_s \cdot \left(\frac{\nabla_s \Gamma^*}{\eta^n} \right) + \frac{1}{\rho} \nabla_s \cdot \mathbf{f}_{ext}^\top, \\ \Gamma^* - \Gamma^n = -\Gamma^n \vartheta^*. \end{array} \right. \quad (15b)$$

This relation can be summarized to a single equation for Γ^* :

$$\frac{\Gamma^*}{\Delta t \Gamma^n} - \frac{2RT}{\rho} \Delta t \nabla_s \cdot \left(\frac{\nabla_s \Gamma^*}{\eta^n} \right) = -\vartheta^n + \frac{1}{\Delta t} - \frac{\Delta t}{\rho} \nabla_s \cdot \mathbf{f}_{ext}^\top. \quad (16)$$

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

Update to Ω^\dagger and ϑ^\dagger . Finally, Building upon the approach proposed by Elcott et al. [2007], viscous forces are further incorporated in a semi-implicitly manner as:

$$\left\{ \begin{array}{l} \frac{\Omega^\dagger - \Omega^*}{\Delta t} = \nu \nabla_s^2 \Omega^\dagger, \\ \vartheta^\dagger - \vartheta^* = \nu \nabla_s^2 \vartheta^\dagger, \end{array} \right. \quad (17a)$$

$$\left\{ \begin{array}{l} \frac{\Omega^\dagger - \Omega^*}{\Delta t} = \nu \nabla_s^2 \Omega^\dagger, \\ \vartheta^\dagger - \vartheta^* = \nu \nabla_s^2 \vartheta^\dagger, \end{array} \right. \quad (17b)$$

where ∇_s^2 is the standard cotangent Laplacian matrix on meshes [Meyer et al. 2003].

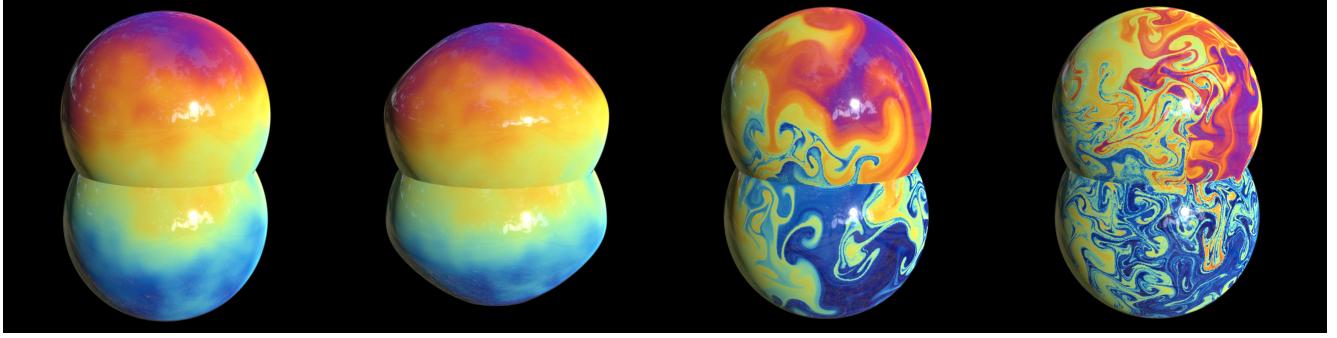
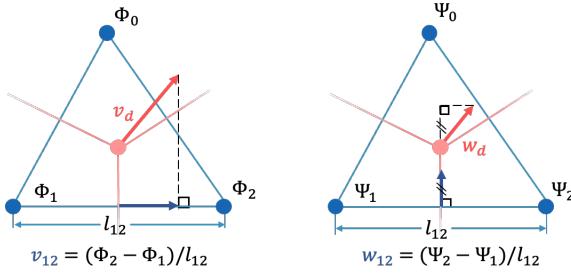


Fig. 6. Double Bubble. The dynamics of double bubble driven by surface tension and Marangoni force.

Fig. 7. Reconstruct velocities on dual mesh. Left: reconstruct v_d from Φ . Right: reconstruct w_d from Ψ .

4.6 Velocity Reconstruction

In this subsection, we initially disregard the boundary conditions, addressing them in the subsequent subsection. To reconstruct velocities on dual mesh \mathcal{D} from Ω^\ddagger and ϑ^\ddagger , we first solve the intermediate fields Ψ and Φ on mesh \mathcal{T} by Eq. (8d) and Eq. (8e). The vorticity $\nabla_s \cdot J\mathbf{u}^\top$ on vertex-i is approximated by Ω_i^\ddagger/S_i , where S_i is the area of the Voronoi cell of vertex-i. ∇_s^2 is discretized by the standard cotangent Laplacian matrix on meshes [Meyer et al. 2003]. After solving for Φ and Ψ , inspired by Elcott et al. [2007], we employ the dual mesh \mathcal{D} for velocity reconstruction using Equation (8f).

We illustrate the reconstruction process with a single triangle example as shown in Fig. 7. For $\nabla_s \Phi$, it's straightforward to obtain the velocity along each edge as $v_{ij} = (\Phi_j - \Phi_i)/l_{ij}$, where l_{ij} is the edge length between vertex-i and vertex-j. The velocities on the three edges of the triangle impose three constraints on the velocity v_d at the vertex of dual mesh: the parallel component of v_d along each edge should be equal to v_{ij} . Note that $\sum v_{ij}l_{ij} = \sum \Phi_j - \Phi_i = 0$ always holds for the velocities along the three edges, leaving us with only two independent constraints that uniquely define the velocity v_d at the vertex of the dual mesh, which corresponds to the $\nabla_s \Phi$:

$$\mathbf{v}_d = \frac{1}{2A} J(v_{02}l_{02}\mathbf{e}_{01} + v_{01}l_{01}\mathbf{e}_{20}), \quad (18)$$

where $\mathbf{e}_{ij} = \mathbf{x}_j - \mathbf{x}_i$ is the edge vector from vertex-i to vertex-j, A is the area of the triangle, and J is the rotation operation.

$J\nabla_s \Psi$ is computed similarly, with the only difference being the inclusion of an extra rotation J . This yields \mathbf{w}_d , representing $J\nabla_s \Psi$

at the vertex of the dual mesh, and can be expressed as:

$$\mathbf{w}_d = -\frac{1}{2A} (w_{02}l_{02}\mathbf{e}_{01} + w_{01}l_{01}\mathbf{e}_{20}), \quad (19)$$

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

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 $\mathbf{n} \cdot \nabla_s \Phi = 0$ and $\mathbf{n} \cdot J\nabla_s \Psi = 0$ 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 Ω and dilatation ϑ . Therefore, instead of directly modifying \mathbf{u}^\top , we adjust the velocities v_{ij} and w_{ij} on edges in Eq. (18) and Eq. (19). Specifically, after the viscosity update, we set v_{ij} and w_{ij} inside the solid to be the projection of the solid velocity along the edge direction, which accordingly changes the tangential velocity to $(\mathbf{u}_d^\top)^\ddagger$. Moreover, we update the circulation and dilatation to Ω^\ddagger and ϑ^\ddagger by:

$$\left\{ \begin{array}{l} \Omega_i^\ddagger = \sum_j w_{ij}l'_{ij}, \\ \vartheta_i^\ddagger = \frac{1}{A} \sum_j v_{ij}l'_{ij}, \end{array} \right. \quad (20a)$$

$$\left\{ \begin{array}{l} \Omega_i^\ddagger = \sum_j w_{ij}l'_{ij}, \\ \vartheta_i^\ddagger = \frac{1}{A} \sum_j v_{ij}l'_{ij}, \end{array} \right. \quad (20b)$$

where A is the Voronoi cell area of vertex-i, l'_{ij} denotes the length of e_{ij} 's dual edge.

4.8 Mesh-Particle Interpolation

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

$$\delta\Omega_p = \sum_{p \in N_r^P(i)} \frac{W(||\mathbf{x}_i - \mathbf{x}_p||, r)}{\sum_{p \in N_r^P(i)} W(||\mathbf{x}_i - \mathbf{x}_p||, r)} \cdot \delta\Omega_i, \quad (21)$$

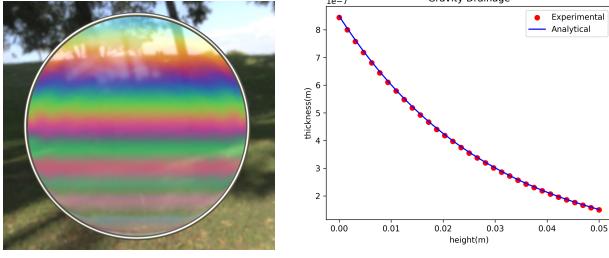


Fig. 8. Gravity Drainage. Left: Our method can accurately simulate the changes in film thickness caused by gravity drainage and generates Newton's interference fringes. Right: The steady-state thickness profile of our simulation result (red dot) is highly consistent with the analytical solution (blue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

where the definition of $W(||\mathbf{x}_i - \mathbf{x}_p||, r)$ and $N_r^p(i)$ retains the same as in the particle-mesh interpolation step. The dilatation ϑ_p^\pm on particles are interpolated using the barycentric weights on mesh \mathcal{T} . For tangential velocities \mathbf{u}^\top , we perform interpolation from the dual mesh \mathcal{D} to the particles' tangential velocities $(\mathbf{u}_p^\top)^\pm$ using the barycentric weights on the Voronoi cell, as introduced in Section 4.3.

4.9 Particle Updates

We update the surfactant concentration Γ and film thickness η on particles using explicit Euler integration of Eq. (8g) and Eq. (8h):

$$\left\{ \begin{array}{l} \Gamma^{n+1} = -\Gamma^n \vartheta_p^\pm \Delta t + \Gamma^n, \\ \eta^{n+1} = -\eta^n \vartheta_p^\pm \Delta t + \eta^n. \end{array} \right. \quad (22a)$$

$$\left\{ \begin{array}{l} \Gamma^{n+1} = -\Gamma^n \vartheta_p^\pm \Delta t + \Gamma^n, \\ \eta^{n+1} = -\eta^n \vartheta_p^\pm \Delta t + \eta^n. \end{array} \right. \quad (22b)$$

The particle positions are updated by tracing trajectories on the underlining mesh [Shi and Yu 2004], and it is enhanced by the classical Runge-Kutta method (RK4) to further reduce numerical dissipation. The final circulation Ω_p^{n+1} and velocities \mathbf{u}_p^{n+1} are set with Ω_p^\pm and \mathbf{u}_p^\pm . We avoid the computation of ϑ_i^{n+1} , because the ϑ_i^{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}_{non} and a set of vortex particles \mathcal{P}_{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}_{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 $\mathcal{T}_{\text{non}}^i$, on which a well-defined dual mesh $\mathcal{D}_{\text{non}}^i$ is established. Similarly, vortex particles are assigned to specific manifolds, which organizes them into distinct groups $\mathcal{P}_{\text{non}}^i$ 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

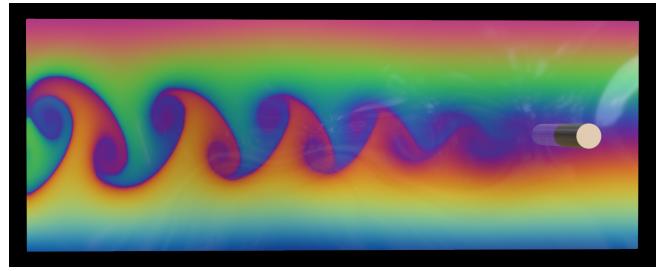


Fig. 9. Von Kármán Vortex Street. After the steady incoming flow passes through the solid, recurrent swirling vortices are generated behind the solid using our method.

geometries and can be directly applied to our mesh \mathcal{T}_{non} . Afterward, we perform the remeshing and project the particles \mathcal{P}_{non} onto the mesh \mathcal{T}_{non} as described in Section 4.4.

- (2) **Tangential Evolution:** For each manifold i , we integrate time as outlined in Section ?? (excluding the normal evolution described in Section 4.4) using $\mathcal{T}_{\text{non}}^i$, $\mathcal{D}_{\text{non}}^i$ and $\mathcal{P}_{\text{non}}^i$.
- (3) **Material Exchange:** When particles from manifold i reach a facet t 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:

$$C = \psi \cdot \left(1 - \min \left(1, \frac{\Gamma_P^j}{\Gamma_P^i} \right) \right). \quad (23)$$

In this equation, ψ is the transport strength parameter, and Γ_P^i and Γ_P^j 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 $\eta(z) = \eta_0 e^{-\frac{\rho g \eta_0 z}{2(\sigma_0 - \sigma)}}$, where η_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

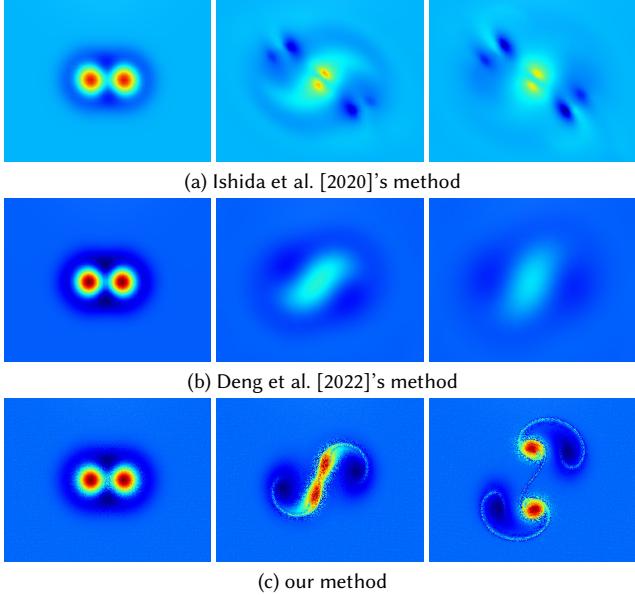


Fig. 10. Taylor vortices. The initial distance between the two vortices satisfies the separation condition, so they should gradually separate during the rotation process. From top to bottom are results from: [Ishida et al. 2020], [Deng et al. 2022] and ours, respectively. Only our approach successfully achieves their separation.

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 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 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.

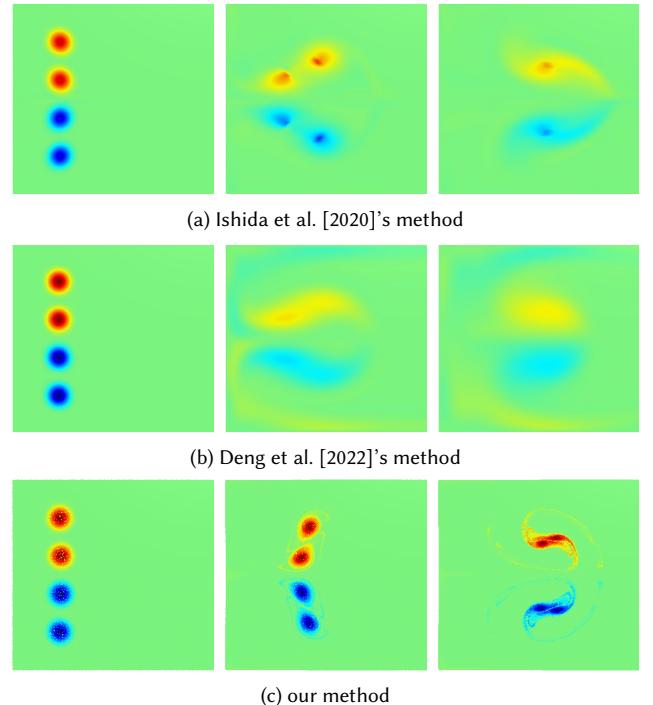


Fig. 11. Leapfrog. Four equal-strength point vortices are placed at the left-hand side of the fluid domain. The upper two and the lower two have opposite rotating directions. From top to bottom are results from: [Ishida et al. 2020], [Deng et al. 2022] and ours, respectively. Our approach demonstrates a remarkable ability to preserve these vortices for longer durations.

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 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

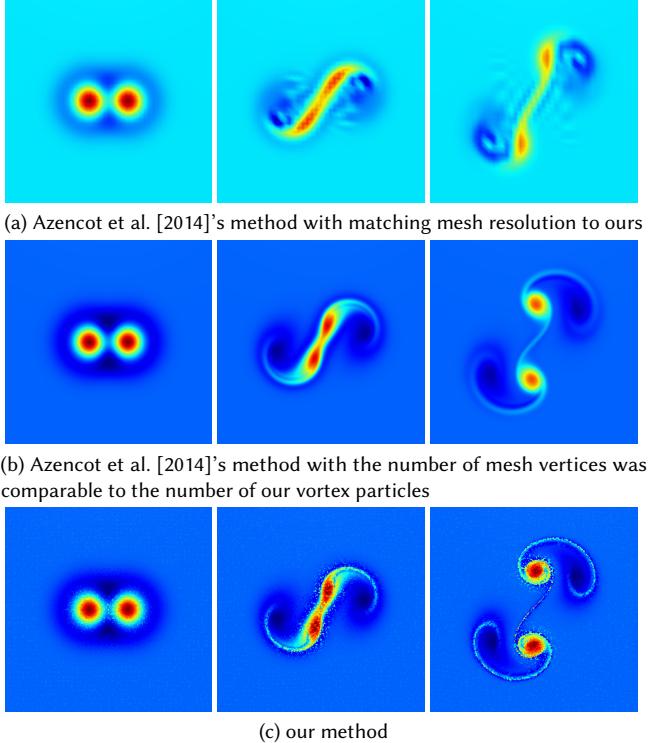


Fig. 12. Comparison with Vortex Method. The top result is simulated using the method proposed by Azencot et al. [2014], which is based on a triangular mesh with 4225 vertices and 8192 faces. The middle result also utilizes the method proposed by Azencot et al. [2014], but with a finer triangular mesh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

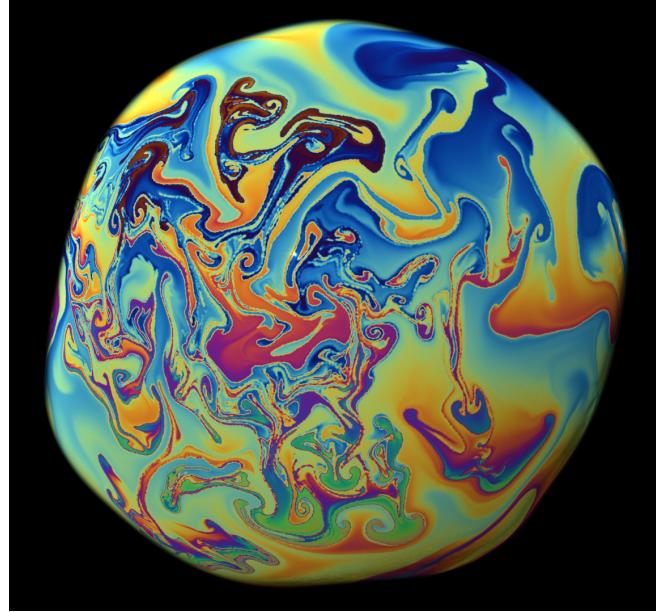


Fig. 13. Giant Bubble. A spherical soap bubble with a radius of 0.5m undergoing free vibrations in response to its initial deformation is simulated with our method. Due to the initial perturbation on physical attributes, the bubble surface exhibits intricate surface flows, driven by the interplay between 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

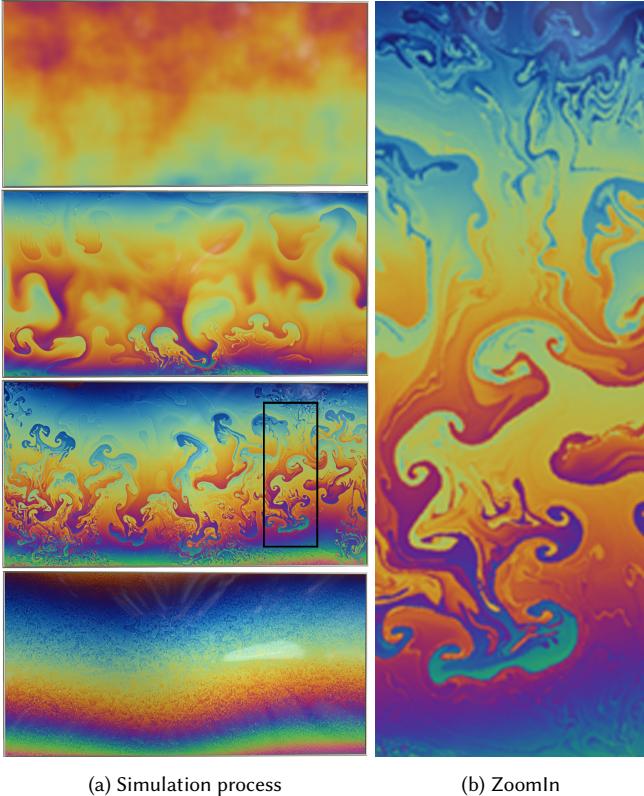


Fig. 14. Deforming Rectangle. A free-vibrating rectangular soap film (with a uniform surfactant concentration and a smooth thickness transition from $\eta \approx 400$ nm at the top to $\eta \approx 200$ nm at the bottom) is simulated with our method. Left: from top to bottom are 4 stages of the simulation. Initially, the soap film exhibits Perlin noise in thickness and surfactant concentration. As the simulation progresses, numerous vortex structures develop on the surface. Eventually, a stable equilibrium state is reached. Right: multiscale vortices captured at one snapshot of the simulation.

influence of gravity, and the lighter material rise further upward, the disturbance rapidly spreads throughout the space. To replicate the real experiment setup by Shabalina et al. [2019], during the simulation, we maintain a constant thickness for the upper layer, while allowing the thickness of the lower layer to evolve dynamically. As illustrated in Fig. 3, the finger-like transformation of the interface in our simulation is consistent with the observations in real world, as documented by Shabalina et al. [2019].

Stick. A soap film starts with a linear thickness gradient and uniform surfactant. Without considering the effect of gravity, as there is no concentration gradient on the film, the Marangoni effect does not manifest, resulting in the absence of tangential fluid flow. However, as depicted in Fig. 2, moving a stick horizontally across the film introduces a velocity disturbance at the fluid-solid interface, leading to captivating turbulent vortices within the region where the stick traverses.

Water Strider. A bipedal miniature robot, resembling a water strider, glides smoothly on a thin soap film, as demonstrated in Fig. 15a. We initialized the thickness and surfactant concentration of

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).

Figure	Scene	$T(s)$	V	F	N
10a	Taylor vortex [†]	96	4225	8192	-
10b	Taylor vortex [‡]	195	-	-	4096E+84100L
10c	Taylor vortex [*]	325	4225	8192	81920
11a	Leapfrog [†]	226	8385	16384	-
11b	Leapfrog [‡]	756	-	-	8192E+168490L
11c	Leapfrog [*]	763	8385	16384	163840

[†] simulated using method proposed by Ishida et al. [2020].

[‡] 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.

Fig.	Method	$T(s)$	V	F	N
12a	[Azencot et al. 2014]	0.19	4225	8192	-
12b	[Azencot et al. 2014]	16.6	66049	131072	-
12c	Ours	0.65	4225	8192	81920

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,

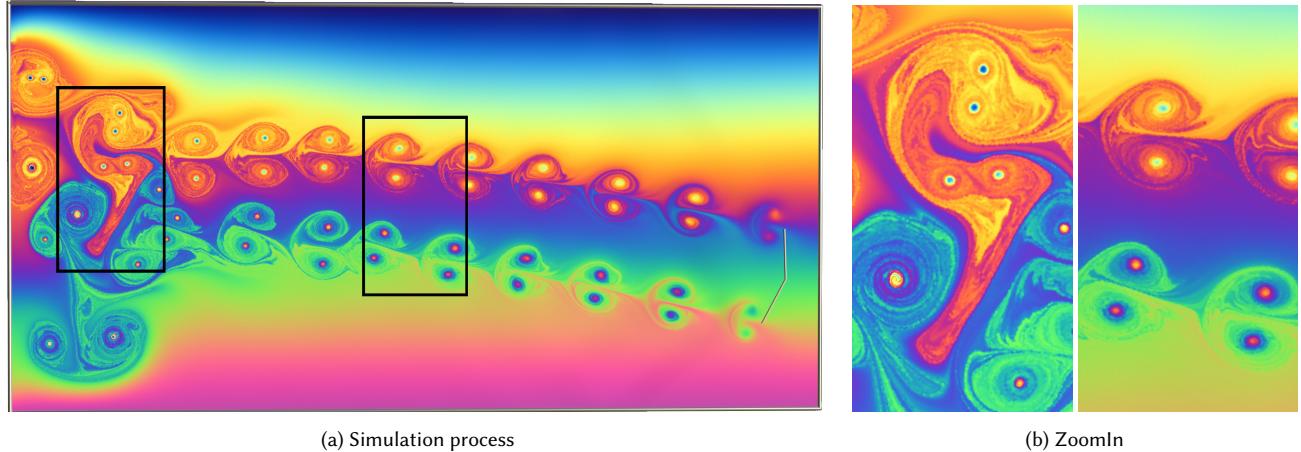


Fig. 15. Water Strider. Left: A bipedal miniature robot periodically steps on the soap film, triggering intricate swirling patterns on liquid film surface. Right: The fine strucutre of the vortex flow in the black-framed regions.

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.

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

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 light-weight 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 (CJG/ZD2021048092601003). The Houdini Education license is credited for the video generations.

REFERENCES

- Bart Adams and Martin Wicke. 2009. Meshless Approximation Methods and Applications in Physics Based Modeling and Animation. In *Eurographics (Tutorials)*. 213–239.
- Ryoichi Ando, Nils Thuerey, and Chris Wojtan. 2015. A stream function solver for liquid simulations. *ACM Trans. Graph.* 34, 4 (2015), 1–9.
- Ryoichi Ando, Nils Thürey, and Chris Wojtan. 2013. Highly adaptive liquid simulations on tetrahedral meshes. *ACM Transactions on Graphics (TOG)* 32, 4 (2013), 1–10.
- Omri Azencot, Steffen Weißmann, Maks Ovsjanikov, Max Wardetzky, and Mirela Ben-Chen. 2014. Functional Fluids on Surfaces. *Computer Graphics Forum* 33, 5 (2014), 237–246.
- Alfred Barnat and Nancy Pollard. 2012. Smoke Sheets for Graph-Structured Vortex Filaments. In *Proceedings of the 2012 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (Lausanne, Switzerland) (SCA '12). ACM, 10 pages.
- Christopher Batty, Florence Bertails, and Robert Bridson. 2007. A fast variational framework for accurate solid-fluid coupling. *ACM Trans. Graph.* 26, 3 (2007), 100–es.
- Harsh Bhatia, Gregory Norgard, Valerio Pascucci, and Peer-Timo Bremer. 2013. The Helmholtz-Hodge Decomposition—A Survey. *IEEE Transactions on Visualization and Computer Graphics* 19, 8 (2013), 1386–1404.
- Jeremiah U Brackbill, Douglas B Kothe, and Hans M Ruppel. 1988. FLIP: a low-dissipation, particle-in-cell method for fluid flow. *Computer Physics Communications* 48, 1 (1988), 25–38.
- Tyson Brochu and Robert Bridson. 2009. Robust Topological Operations for Dynamic Explicit Surfaces. *SIAM Journal on Scientific Computing* 31, 4 (2009), 2472–2493.
- Tyson Brochu, Todd Keeler, and Robert Bridson. 2012. Linear-time smoke animation with vortex sheet meshes. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (Lausanne, Switzerland) (SCA '12). Eurographics Association, Goslar, DEU, 87–95.
- Jean-Marc Chomaz. 2001. The dynamics of a viscous soap film with soluble surfactant. *Journal of Fluid Mechanics* 442 (2001), 387–409.
- Georges-Henri Cottet, Petros D Kouroussakos, et al. 2000. *Vortex methods: theory and practice*. Vol. 8. Cambridge university press Cambridge.
- Y Couder, JM Chomaz, and Marc Rabaud. 1989. On the hydrodynamics of soap films. *Physica D: Nonlinear Phenomena* 37, 1–3 (1989), 384–405.
- Keenan Crane, Fernando de Goes, Mathieu Desbrun, and Peter Schröder. 2013. Digital geometry processing with discrete exterior calculus. In *ACM SIGGRAPH 2013 Courses* (Anaheim, California) (SIGGRAPH '13). Association for Computing Machinery, New York, NY, USA, 126 pages.
- 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. Graph.* 34, 4 (Jul 2015), 9 pages.
- Y. Deng, M. Wang, X. Kong, S. Xiong, Z. Xian, and B. Zhu. 2022. A Moving Eulerian-Lagrangian Particle Method for Thin Film and Foam Simulation. *ACM Trans. Graph.* 41, 4 (2022).
- Sharif Elcott, Yiyi Tong, Eva Kanso, Peter Schröder, and Mathieu Desbrun. 2007. Stable, circulation-preserving, simplicial fluids. *ACM Trans. Graph.* 26, 1 (2007), 4–es.
- Nick Foster and Dimitri Metaxas. 1996. Realistic animation of liquids. *Graphical models and image processing* 58, 5 (1996), 471–483.
- Chuyuan Fu, Qi Guo, Theodore Gast, Chenfanfu Jiang, and Joseph Teran. 2017. A polynomial particle-in-cell method. *ACM Trans. Graph.* 36, 6 (2017), 1–12.
- GameDev.net. 2013. Thin Film Interference for Computer Graphics. <https://www.gamedev.net/tutorials/programming/graphics/thin-film-interference-for-computer-graphics-r2962/>. Accessed: Date Accessed.
- Gaël Guennebaud, Benoit Jacob, et al. 2010. Eigen v3. <http://eigen.tuxfamily.org>.
- Francis H Harlow. 1962. *The particle-in-cell method for numerical solution of problems in fluid dynamics*. Technical Report. Los Alamos National Lab.(LANL), Los Alamos, NM (United States).
- David J. Hill and Ronald D. Henderson. 2016. Efficient Fluid Simulation on the Surface of a Sphere. 35, 2 (apr 2016), 9 pages.
- Weizhen Huang, Julian Iseringhausen, Tom Kneiphof, Ziyin Qu, Chenfanfu Jiang, and Matthias B Hullin. 2020. Chemomechanical simulation of soap film flow on spherical bubbles. *ACM Trans. Graph.* 39, 4 (2020), 41–1.
- Sadashige Ishida, Peter Synak, Fumiya Narita, Toshiya Hachisuka, and Chris Wojtan. 2020. A model for soap film dynamics with evolving thickness. *ACM Trans. Graph.* 39, 4 (2020), 31–1.
- Sadashige Ishida, Chris Wojtan, and Albert Chern. 2022. Hidden Degrees of Freedom in Implicit Vortex Filaments. *ACM Trans. Graph.* 41, 6 (nov 2022), 14 pages.
- Sadashige Ishida, Masafumi Yamamoto, Ryoichi Ando, and Toshiya Hachisuka. 2017. A hyperbolic geometric flow for evolving films and foams. *ACM Trans. Graph.* 36, 6 (2017), 1–11.
- Chenfanfu Jiang, Craig Schroeder, Andrew Selle, Joseph Teran, and Alexey Stomakhin. 2015. The affine particle-in-cell method. *ACM Trans. Graph.* 34, 4 (2015), 1–10.
- Byungmoon Kim, Yingjie Liu, Ignacio Llamas, Xiangmin Jiao, and Jarek Rossignac. 2007. Simulation of bubbles in foam with the volume control method. In *ACM SIGGRAPH 2007 Papers* (San Diego, California) (SIGGRAPH '07). Association for Computing Machinery, New York, NY, USA, 98–es.
- Masami Matsumoto and Shigeo Kawata. 1990. TRIPIC: Triangular-mesh particle-in-cell code. *J. Comput. Phys.* 87, 2 (1990), 488–493.
- Alexander George McKenzie. 2007. *HOLA: a high-order Lie advection of discrete differential forms with applications in Fluid Dynamics*. Ph. D. Dissertation. California Institute of Technology.
- Mark Meyer, Mathieu Desbrun, Peter Schröder, and Alan H. Barr. 2003. Discrete Differential-Geometry Operators for Triangulated 2-Manifolds. In *Visualization and Mathematics III*, Hans-Christian Hege and Konrad Polthier (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 35–57.
- François Morency, Hélène Beaugendre, and Fédérico Gallizio. 2012. Aerodynamic force evaluation for ice shedding phenomenon using vortex in cell scheme, penalisation and level set approaches. *International Journal of Computational Fluid Dynamics* 26, 9–10 (2012), 435–450.
- Sang Il Park and Myoung Jun Kim. 2005. Vortex fluid for gaseous phenomena. In *Proceedings of the 2005 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (Los Angeles, California) (SCA '05). Association for Computing Machinery, New York, NY, USA, 261–270.
- Tobias Pfaff, Nils Thuerey, and Markus Gross. 2012. Lagrangian vortex sheets for animating fluids. *ACM Trans. Graph.* 31, 4 (Jul 2012), 8 pages.
- Tobias Pfaff, Nils Thürey, Andrew Selle, and Markus Gross. 2009. Synthetic turbulence using artificial boundary layers. *ACM Trans. Graph.* 28, 5 (dec 2009), 1–10.
- Ziyin Qu, Minchen Li, Fernando De Goes, and Chenfanfu Jiang. 2022. The power particle-in-cell method. *ACM Transactions on Graphics* 41, 4 (2022).
- Ziyin Qu, Minchen Li, Yin Yang, Chenfanfu Jiang, and Fernando De Goes. 2023. Power Plastics: A Hybrid Lagrangian/Eulerian Solver for Mesoscale Inelastic Flows. *ACM Trans. Graph.* 42, 6 (dec 2023), 11 pages.
- 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. <https://doi.org/10.1126/science.1230623>
- Andrew Selle, Nick Rasmussen, and Ronald Fedkiw. 2005. A vortex particle method for smoke, water and explosions. In *ACM SIGGRAPH 2005 Papers* (Los Angeles, California) (SIGGRAPH '05). Association for Computing Machinery, New York, NY, USA, 910–914.
- Evgenia Shabalina, Antoine Bérut, Mathilde Cavelier, Arnaud Saint-Jalmes, and Isabelle Cantat. 2019. Rayleigh-Taylor-like instability in a foam film. *Phys. Rev. Fluids* 4 (Dec 2019), 124001. Issue 12.
- Lin Shi and Yizhou Yu. 2004. Inviscid and incompressible fluid simulation on triangle meshes. *Computer Animation and Virtual Worlds* 15, 3–4 (2004), 173–181.
- SideFX. 2023. Houdini (Version 19.5). <https://www.sidefx.com>. [Computer software].
- 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., USA, 121–128.
- Mengdi Wang, Yitong Deng, Xiangxin Kong, Aditya H. Prasad, Shiying Xiong, and Bo Zhu. 2021. Thin-film smoothed particle hydrodynamics fluid. *ACM Trans. Graph.* 40, 4 (Jul 2021), 16 pages.
- Stephanie Wang, Mohammad Sina Nabizadeh, and Albert Chern. 2023. Exterior Calculus in Graphics: Course Notes for a SIGGRAPH 2023 Course. In *ACM SIGGRAPH 2023 Courses* (, Los Angeles, California) (SIGGRAPH '23). Association for Computing Machinery, New York, NY, USA, 126 pages.
- Joe Warren, Scott Schaefer, Anil N Hirani, and Mathieu Desbrun. 2007. Barycentric coordinates for convex sets. *Advances in computational mathematics* 27 (2007), 319–338.
- Steffen Weißmann and Ulrich Pinkall. 2010. Filament-based smoke with vortex shedding and variational reconnection. *ACM Trans. Graph.* 29, 4 (Jul 2010), 12 pages.
- S. Xiong, R. Tao, Y. Zhang, F. Feng, and B. Zhu. 2021. Incompressible flow simulation on vortex segment clouds. *ACM Trans. Graph.* 40, 4 (2021).
- Hang Yin, Mohammad Sina Nabizadeh, Baichuan Wu, Stephanie Wang, and Albert Chern. 2023. Fluid Cohomology. *ACM Trans. Graph.* 42, 4 (Jul 2023), 25 pages.
- Xinxin Zhang and Robert Bridson. 2014. A PPPM fast summation method for fluids and beyond. *ACM Trans. Graph.* 33, 6 (Nov 2014), 11 pages.
- 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* (Vienna, Austria) (SCA '06). Eurographics Association, Goslar, DEU, 325–333.
- Bo Zhu, Ed Quigley, Matthew Cong, Justin Solomon, and Ronald Fedkiw. 2014. Codimensional surface tension flow on simplicial complexes. *ACM Trans. Graph.* 33, 4 (Jul 2014), 11 pages.
- Yongning Zhu and Robert Bridson. 2005. Animating sand as a fluid. *ACM Trans. Graph.* 24, 3 (Jul 2005), 965–972.