Particle-Laden Fluid on Flow Maps

ZHIQI LI, Georgia Institute of Technology, USA
DUOWEN CHEN, Georgia Institute of Technology, USA
CANDONG LIN, Georgia Institute of Technology, USA
JINYUAN LIU, Dartmouth College, USA
BO ZHU, Georgia Institute of Technology, USA

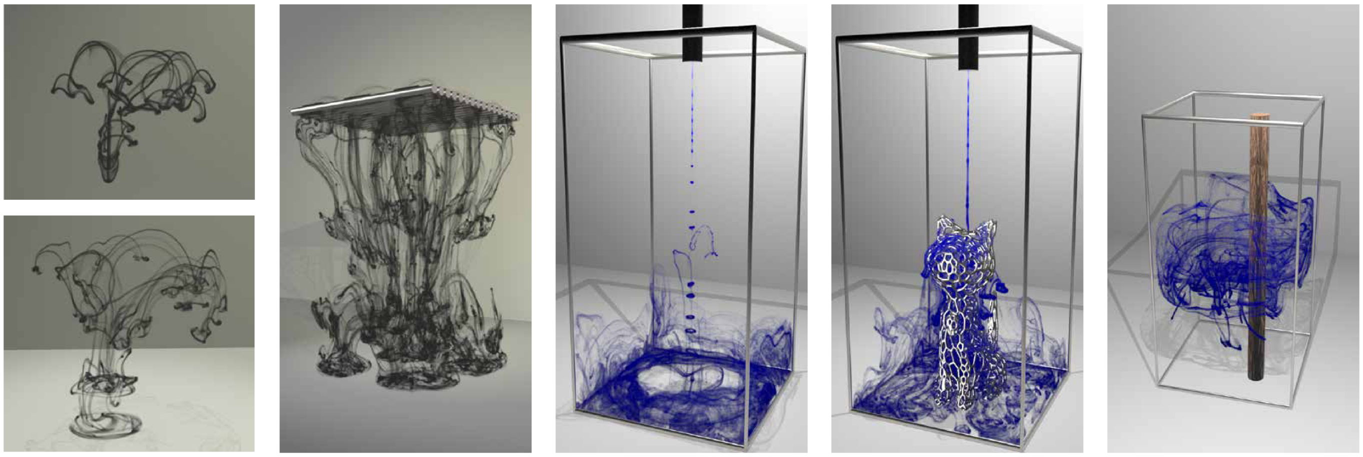

Fig. 1. Examples of ink diffusion phenomena simulated by our method. Five simulation snapshots are shown from left to right: the interaction of two ink drops forming ink rings and filaments; the formation of complex patterns as ink flows down a porous plate; the bulging, breakup, and torus formation of the tail of a dripped ink drop; the interaction between ink and a hollow cat; and the ink diffusion stirred by a chopstick.

We propose a novel framework for simulating ink as a particle-laden flow using particle flow maps. Our method addresses the limitations of existing flow-map techniques, which struggle with dissipative forces like viscosity and drag, thereby extending the application scope from solving the Euler equations to solving the Navier-Stokes equations with accurate viscosity and laden-particle treatment. Our key contribution lies in a coupling mechanism for two particle systems, coupling physical sediment particles and virtual flow-map particles on a background grid by solving a Poisson system. We implemented a novel path integral formula to incorporate viscosity and drag forces into the particle flow map process. Our approach enables state-of-the-art simulation of various particle-laden flow phenomena, exemplified by the bulging and breakup of suspension drop tails, torus formation, torus disintegration, and the coalescence of sedimenting drops. In particular, our method delivered high-fidelity ink diffusion simulations by accurately capturing vortex bulbs, viscous tails, fractal branching, and hierarchical structures.

Authors' Contact Information: Zhiqi Li, zli3167@gatech.edu, Georgia Institute of Technology, USA; Duowen Chen, dchen322@gatech.edu, Georgia Institute of Technology, USA; Candong Lin, lincandong@outlook.com, Georgia Institute of Technology, USA; Jinyuan Liu, jinyuan.liu.gr@dartmouth.edu, Dartmouth College, USA; Bo Zhu, bo.zhu@gatech.edu, Georgia Institute of Technology, USA.

ACM 1557-7368/2024/12-ART266
https://doi.org/10.1145/3687916

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

Additional Key Words and Phrases: Particle Laden Flow, Particle Flow Map, Path Integral, Ink Diffusion

ACM Reference Format:
Zhiqi Li, Duowen Chen, Candong Lin, Jinyuan Liu, and Bo Zhu. 2024. Particle-Laden Fluid on Flow Maps. *ACM Trans. Graph.* 43, 6, Article 266 (December 2024), 12 pages. https://doi.org/10.1145/3687916

1 Introduction

The diffusion process of ink in water exhibits fascinating flow characteristics. Starting from a localized, high-concentration region, the ink diffuses into its fluid environment by forming incipient blobs, filament tails, fractal branches, and chandelier-like structures, exhibiting a tremendous mix of pattern similarities and flow complexities that have attracted experimental observation, mathematical study, and numerical simulation from scientists across different disciplines. Despite the visually complicated flow process during ink diffusion, its underlying governing physics is surprisingly simple: As studied in previous literature on fluid mechanics (e.g., see [Nitsche and Batchelor 1997]), ink diffusion in water can be described as a typical particle-laden flow process, where ink particles are dispersed within the fluid medium and interact with the fluid flow through the drag force produced on each particle's granular shape. The concentrated blobs, elongated tails, and fractal branches form due to the flow instability among vorticity, viscosity, and particle-laden drags.

However, despite the elegance of the physics picture, devising a first-principle method to simulate ink diffusion remains a challenging problem in computational physics and computer graphics. Pioneering efforts to simulate ink diffusion studied the vorticity-viscosity interaction and produced simulation results of one layer of branching with vortex particles in the early 1990s [Nitsche and Batchelor 1997], which, to the best of our knowledge, is the first work simulating ink diffusion from the perspective of particle-laden flow. This work motivated future works in the area (e.g., [Bosse et al. 2005; Machu et al. 2001; Walther and Koumoutsakos 2001]). In computer graphics, researchers have repurposed the standard grid-based smoke simulator with vorticity confinement [Fedkiw et al. 2001] to a particle-laden flow setting for ink diffusion, which reproduced impressive ink branching phenomena [Sagong et al. 2015]. In the recent work of [Padilla et al. 2019], the authors simulated ink chandeliers as vortex filaments with varying thicknesses and achieved notable multi-layer chandelier structures. Despite these inspiring successes, a versatile simulation framework that can naturally handle the vortical development and structural evolution by directly solving particle-laden flow in an arbitrary incompressible flow environment remains to be developed. In particular, the challenge of directly simulating vorticity-viscosity interaction to spontaneously emerge the ink blobs, filaments, branches, and structures by directly solving the Navier-Stokes (NS) equations with sediment particles remains an essential problem to be addressed.

To this end, we propose a novel and versatile framework to simulate ink as a particle-laden flow. Our method is based on the recent line of works on solving incompressible flow with flow maps (e.g., [Deng et al. 2023; Li et al. 2023, 2024; Nabizadeh et al. 2022; Zhou et al. 2024]). Though these flow-map schemes have demonstrated their extraordinary capabilities in preserving accurate vortical structures — which would be particularly beneficial for ink diffusion — they inherently suffer from limitations in tackling dissipative forces such as viscosity and drag. *To date, no flow-map paradigms can accurately solve for viscosity, let alone its interaction with vorticity.* Mathematically, this is due to the lack of a mathematical foundation for calculating viscosity (or any forces other than pressure) along each particle's trajectory.

To complete this missing piece, which will further unleash the flow-map method's notable vorticity-preserving ability in a broader scope of scenarios, in particular, scenarios where vorticity-viscosity interaction plays an important role, we propose a novel particle flow-map simulation method to solve the **full Navier-Stokes equations** with laden-particle interactions. The key technical novelties we delivered include two parts: On the one hand, we have devised two particle systems, one for physical, sediment particles (with mass and momentum) to track mass transport, and the other for the virtual, flow map particles (massless) to evolve vortical structures. Sediment particles and flow-map particles can exchange information via a background grid during the Poisson solve for incompressibility. On the other hand, we devised a novel path integral formula and a flexible flow map adaption strategy to incorporate viscosity and drag forces into the previously purely geometric mapping process. This Euler-to-NS enhancement, in conjunction with its particle-laden coupling, will enable the state-of-the-art simulation of ink diffusion by accurately capturing the flow details during the entire development of ink diffusion, including the vortex bulb, viscous tails, fractal branching, and hierarchical structures.

The main contributions of our approach include:

- A novel path integral form for the Navier-Stokes equations with accurate viscosity and drag treatments;
- A novel mechanism to couple long-range flow maps, short-range forces, and a short-range projection with variable-coefficient Poisson equations;
- A unified Eulerian-Lagrangian particle-laden flow solver that can facilitate the state-of-the-art ink diffusion simulations exhibiting complex vorticity-viscosity interaction.

2 Related Work

Eulerian-Lagrangian Methods. Hybrid methods in fluid simulation merge the strengths of both Lagrangian and Eulerian approaches, resulting in more versatile and robust systems. Since the pivotal work on PIC [Harlow 1962] and FLIP [Brackbill and Ruppel 1986] was introduced to the graphics community by Zhu and Bridson [2005], hybrid Eulerian-Lagrangian representations have become prevalent in fluid simulations [Deng et al. 2022; Raveendran et al. 2011; Zhu et al. 2010]. The Material Point Method (MPM), which extends the concepts of PIC/FLIP, has been employed to simulate a range of continuum behaviors, such as collision and fracture [Stomakhin et al. 2013], viscoplasticity [Yue et al. 2015], magnetization [Sun et al. 2021], solid-fluid interaction [Fang et al. 2020], and sedimentation [Gao et al. 2018]. Ongoing research has improved the accuracy of transfers between particles and grids, addressing issues such as unconserved vorticity [Fu et al. 2017; Jiang et al. 2015], displacement discontinuity [Hu et al. 2018], and volume conservation [Qu et al. 2022].

Particle-Laden Flow. Particle-laden flow refers to the movement of a fluid that contains suspended particles, such as dust in the air [Xiu et al. 2020], sediment in water [Gao et al. 2018], or droplets in a gas [Xu et al. 2023]. The accurate representation of such flows involves modeling the interactions between particles and fluids, including drag [Sagong et al. 2015], dispersion [Xu et al. 2011], and phase transition [Vignesh et al. 2022]. While the particles are typically treated as Lagrangian when coupling with the fluid [Xu and Yu 1997], they can behave like a secondary fluid and be modeled through a two-phase flow solver [Kartushinsky et al. 2016] or a multiscale continuous approach [Idelsohn et al. 2022]. Inks made of fine-scale pigment particles can interact with fluids and exhibit complex diffusion patterns, such as bulging and breakup [Machu et al. 2001], branching [Thomson and Newall 1886], and other topological transitions. Numerous experiments have been conducted to explore these processes through high-speed photography [Thoroddsen et al. 2008] and particle image velocimetry [Zhou et al. 2022], capturing particle speed and size [Wijshoff 2018], viscosity [Krainer et al. 2019], and temperature [Lee et al. 2004]. Ink simulations include vorticity confinement [Sagong et al. 2015], vortex particles [Walther and Koumoutsakos 2001], and vortex filaments [Padilla et al. 2019].

Flow Map Methods. Flow map, a geometric representation of spatial time slice, was widely used to reduce diffusion errors presented in interpolations and advections. Such a method was first introduced in [Wiggert and Wylie 1976] in computational physics and later brought to the graphics community in [Hachisuka 2005]. The idea of mapping between simulation time slices connected by tracing virtual particles between frames was explored in [Qu et al. 2019; Sato et al. 2018, 2017], but the mapping happened between velocity fields. From the introduction of covector fluids [Nabizadeh et al. 2022] to the graphics community, the benefit of flow map aiding the advection of covectors proved its superiority in vorticity concentrated phenomena. Flow map methods for covectors were improved by [Deng et al. 2023] based on a neural buffer to facilitate a bidirectional map, which has been further improved with particles [Li et al. 2024; Sancho et al. 2024; Zhou et al. 2024] to reduce memory consumption.

3 Physical Model

Naming Convention. We distinguish fluid and sediment using superscripts f and s. We denote sediment particles with subscript p. We use $\mathbf{u}$ and $\mathbf{v}$ to denote fluid and sediment velocities.

3.1 Fluid

Based on [Gao et al. 2018; Nielsen and Østerby 2013; Sun and Xiao 2016], the fluid momentum and mass conservation equations are

$$\begin{cases} \frac{\partial(\epsilon^f \rho^f \mathbf{u})}{\partial t} + \nabla \cdot (\epsilon^f \rho^f \mathbf{u} \otimes \mathbf{u}) = -\epsilon^f \nabla p^f + \epsilon^f \rho^f \mathbf{g} + \mu \epsilon^f \Delta \mathbf{u} + \mathbf{f}^f_{\text{drag}}, \\ \frac{\partial(\epsilon^f \rho^f)}{\partial t} + \nabla \cdot (\epsilon^f \rho^f \mathbf{u}) = 0, \end{cases} \tag{1}$$

where ϵ^f, ρ^f, μ, $\mathbf{u}$, p^f, $\mathbf{g}$ and $\mathbf{f}^f_{\text{drag}}$ are the fluid volume fraction, fluid intrinsic density, fluid viscosity coefficient, fluid velocity, fluid pressure, the gravitational constant and the fluid drag force density respectively. Assuming constant fluid density ρ^f and combined with the mass conservation equation, the fluid momentum equation can be reformulated as

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} = -\frac{1}{\rho^f} \nabla p^f + \mathbf{g} + \frac{\mu}{\rho^f} \Delta \mathbf{u} + \frac{1}{\rho^f \epsilon^f} \mathbf{f}^f_{\text{drag}}. \tag{2}$$

Assuming constant sediment density ρ^s and combined with mass equation for sediment $\frac{\partial(\epsilon^s \rho^s)}{\partial t} + \nabla \cdot (\epsilon^s \rho^s \mathbf{v}) = 0$ and with relationship of volume fraction $\epsilon^f + \epsilon^s = 1$, the fluid incompressibility condition is obtained as

$$\nabla \cdot (\epsilon^f \mathbf{u} + \epsilon^s \mathbf{v}) = 0, \tag{3}$$

where ϵ^s, ρ^s, and $\mathbf{v}$ are the sediment volume fraction, sediment intrinsic density, and sediment velocity respectively.

3.2 Sediment

Similar to [Sun and Xiao 2016], the sediment is described by a collection of sediment particles following Newton's Law. For each particle p, it holds

$$\frac{d\mathbf{v}_p}{dt} = \left(\frac{\rho^s - \rho^f}{\rho^s} \right) \mathbf{g} + \frac{1}{m_p} \mathbf{f}_{\text{drag},p}, \tag{4}$$

where $\mathbf{v}_p$ and $m_p = \rho^s V_p$ denote velocity and mass of sediment particles respectively. Here, $V_p = \frac{4}{3}\pi r_p^3$ and r_p is the radius of a sediment particle. For viscous laden flow like ink, we use Stokes' Law to calculate the drag force $\mathbf{f}_{\text{drag},p}$, given by:

$$\mathbf{f}_{\text{drag},p} = 6\pi\mu r_p (\mathbf{u}(\mathbf{x}_p, t) - \mathbf{v}_p), \tag{5}$$

where $\mathbf{x}_p$ denote the position of particle p. By Newton's third law, for a fluid microelement occupying the space $\mathcal{V}$, the fluid drag force density can be calculated as

$$\mathbf{f}^f_{\text{drag}} = -\frac{1}{|\mathcal{V}|} \sum_{\mathbf{x}_p \in \mathcal{V}} \mathbf{f}_{\text{drag},p}, \tag{6}$$

where $|\mathcal{V}|$ denote the volume of space $\mathcal{V}$.

Table 1. Summary of important notations used in the paper.

Notation	Type	Definition
$*^f$	all	Fluid property
$*^s$	all	Sediment particle property
$*_p$	all	property on sediment particle p
$*_q$	all	property on fluid particle q
$*_i$	all	fluid or sediment property on the grid i
r	scalar	Current time
s	scalar	Initial time
s'	scalar	One time step before the current time
$\Phi_{a\to b}$	vector	Forward flow map from time a to b
$\Psi_{b\to a}$	vector	Backward flow map from time b to a
$\mathcal{F}_{a\to b}$	matrix	Jacobian of forward flow map $\Phi_{a\to b}$
$\mathcal{T}_{b\to a}$	matrix	Jacobian of backward flow map $\Psi_{b\to a}$
$\Gamma_{a\to b}$	vector	Covector-based integration of fluid force from time a to b
$\mathbf{u}^M_{a\to b}$	vector	**Mapped** velocity at time b by covector flow map from time a
$\mathbf{u}^A_{a\to b}$	vector	**Advected** velocity at time b by particle velocity from time a

4 Particle Flow Map for Laden Flow

4.1 Flow Map Theory

Flow Map. During the motion of fluid with velocity $\mathbf{u}_t(\mathbf{x}) = \mathbf{u}(\mathbf{x}, t)$, $\forall t > 0$, consider an arbitrary fluid particle moves from the position $\mathbf{x}_s \in \Omega_s$ at the initial time s to the position $\mathbf{x}_r \in \Omega_r$ at the current time r, where Ω_s and Ω_r represent the domain at times s and r respectively. The positions $\mathbf{x}_s$ and $\mathbf{x}_r$ can be associated using the forward flow map $\Phi_{s\to r} : \Omega_s \to \Omega_r$ and the backward flow map $\Psi_{r\to s} : \Omega_r \to \Omega_s$, satisfying $\Phi_{s\to r}(\mathbf{x}_s) = \mathbf{x}_r$ and $\Psi_{r\to s}(\mathbf{x}_r) = \mathbf{x}_s$. The Jacobian matrices of the flow maps $\Phi_{s\to r}$ and $\Psi_{r\to s}$ are denoted as $\mathcal{F}_{s\to r}(\mathbf{x}_s) = \frac{\partial \Phi_{s\to r}(\mathbf{x}_s)}{\partial \mathbf{x}_s}$ and $\mathcal{T}_{r\to s} = \frac{\partial \Psi_{r\to s}(\mathbf{x}_r)}{\partial \mathbf{x}_r}$ respectively. According to [Deng et al. 2023], for any time

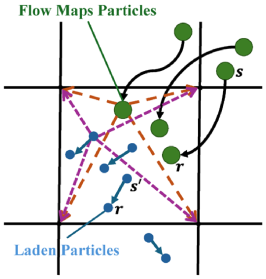

Fig. 2. Flow Maps are tracked by fluid particles q and long-range mapped velocity is calculated, while sediment particles p advect short-range velocity step-by-step. They interact via the background grid with interpolation and Poisson solve.

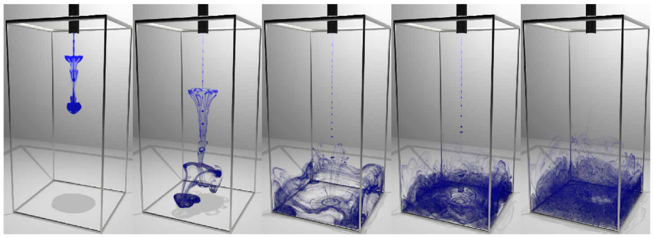

Fig. 3. Dripping. The long tail of a descending ink will bulge and break up into many suspension drops, which form tori afterward

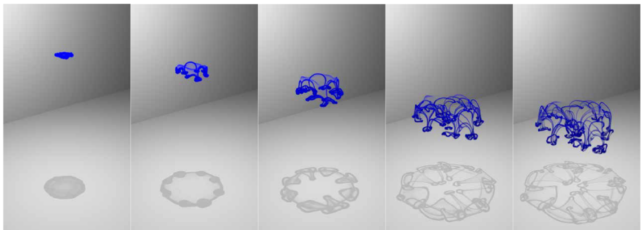

Fig. 4. Ink Torus Breakup: Under drag force and viscous force, a ink torus disintegrates into several blobs, which deform into tori and disintegrate again, resulting in a cascade process of blob deformations and breakups.

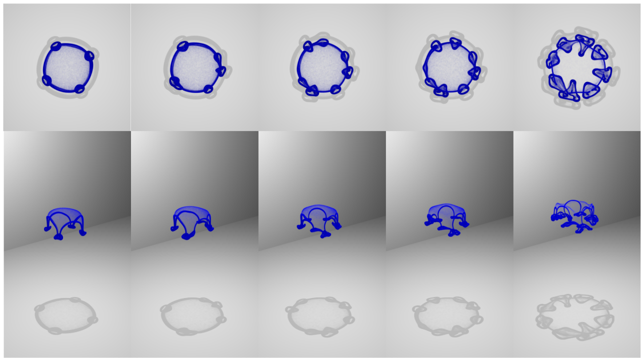

Fig. 5. Ink Torus Breakup Under Re = 16.5, 18, 19.5, 21.0, 30.0: number of blobs disintegrated from an ink torus increase with Reynolds Number, under interaction of viscous force and vorticity.

t, $\mathcal{F}_{s\to t}$ and $\mathcal{T}_{t\to s}$ satisfy the evolution equations:

$$\begin{aligned}\frac{D\mathcal{F}_{s\to t}}{Dt} &= \nabla\mathbf{u}(\mathbf{x})\mathcal{F}_{s\to t},\\ \frac{D\mathcal{T}_{t\to s}}{Dt} &= -\mathcal{T}_{t\to s}\nabla\mathbf{u}(\mathbf{x}),\end{aligned} \tag{7}$$

where $\mathcal{F}_{s\to t}$ and $\mathcal{T}_{t\to s}$ are abbreviations for $\mathcal{F}_{s\to t}(\Psi_{t\to s}(\mathbf{x}))$ and $\mathcal{T}_{t\to s}(\mathbf{x})$, respectively. The $\Psi_{r\to s}$, as the mapping $\Psi_{r\to s} : \Omega_r \to \Omega_s$, induces a pullback of scalar fields $(\Psi_{r\to s})^* : \Omega_s^* \to \Omega_r^*$ and a pullback of covector fields $(\Psi_{r\to s})^* : \mathfrak{X}^*(\Omega_s) \to \mathfrak{X}^*(\Omega_r)$ where Ω_r^* is the space of scalar field on Ω_r and $\mathfrak{X}^*(\Omega_r)$ is the space of covector field on Ω_r (the pullback of $\Phi_{s\to t}$ is similar, see [Crane et al. 2013]).

Particle Flow Map. To solve incompressible and inviscid fluid, Li et al. [2024] and Zhou et al. [2024] use particles as flow maps to calculate the path integral form (see [Li et al. 2024] for details):

$$\mathbf{u}_q(r) = \mathcal{T}_{r\to s,q}^T\mathbf{u}_q(s) - \nabla\Lambda_{r\to s,q}, \tag{8}$$

where the subscript q denotes quantities carried by the particle q, $\Lambda_{r\to s,q} = \int_s^r \lambda\left(\Phi_{s\to\tau}(\Psi_{r\to s}(x_q(r))), \tau\right) d\tau$, $\lambda = p - \frac{1}{2}|u|^2$ and $x_q(r)$ is the position of q at time r. Based on Equation 8, the map-projection process is used to solve the incompressible Euler equation:

(1) (**Long-Range Mapping**) Calculate the long-range mapped velocity on particles as: $\mathbf{u}_{s\to r,q}^M = \mathcal{T}_{r\to s,q}{}^T\mathbf{u}_{s,q}$;
(2) (**Long-Range Projection**) Solve the Poisson equation regarding Λ_s^r: $\nabla\cdot\nabla\Lambda_s^r = \nabla\cdot\mathbf{u}_{s\to r}^M$ on the grid to obtain Λ_s^r. Then carry out a projection as $\mathbf{u}_r = \mathbf{u}_{s\to r}^M - \nabla\Lambda_s^r$ to ensure $\mathbf{u}_r$ satisfies $\nabla\cdot\mathbf{u}_r = 0$.

In this scheme, particles are used to track flow maps and compute mapped velocity, while the grid or other discrete structures can be used to perform projection.

4.2 Adapting Particle Flow Map to Laden Flow

When applying the particle flow map method to laden flow, we encountered the following challenges:

(1) The integral form Equation 8 is derived from the Euler equation without considering dissipative forces, which makes the particle flow map method unable to account for forces other than fluid pressure.
(2) The particle flow map method cannot directly interact with sediment particles because it operates on a long-range mapped velocity $\mathbf{u}_{s\to r}^M$. In contrast, the velocity of sediment particles is described by step-by-step updates of short-range advected velocity $\mathbf{v}_p$. Although $\mathbf{u}_r$ can interact with $\mathbf{v}_p$, the interactions between the short-range fluid velocity $\mathbf{u}_r$ and the sediment particles' velocity $\mathbf{v}_p$ are short-range. They cannot be reflected in the long range.
(3) In the particle flow map method, the pressure integral $\Lambda_{s\to r}$ is solved by directly projecting the mapped velocity $\mathbf{u}_{s\to r}^M$. This approach is not applicable in sediment flow because the velocity in sediment flow needs to satisfy the incompressible constraint Equation 3, which includes information about the sediment and the fluid's volume fraction. Leveraging this short-range information to enforce incompressibility with a Poisson equation for the long-range pressure $\Lambda_{s\to r}$ is difficult.

4.3 Integration Form and Path Integral

To address the first challenge, we propose a new integral formulation based on Equation 2 capable of handling forces other than pressure. For simplicity, we denote any force other than pressure as γ and rewrite Equation 2 as

$$\frac{\partial\mathbf{u}}{\partial t} + (\mathbf{u}\cdot\nabla)\mathbf{u} = \gamma - \frac{1}{\rho^f}\nabla p. \tag{9}$$

By utilizing covector forms, we derived the integral form (see Appendix A for detailed derivation):

$$\mathbf{u}(\mathbf{x}, r) = \underbrace{\mathcal{T}_{r\to s}^T(\mathbf{x})\mathbf{u}_s(\Psi_{r\to s}(\mathbf{x}), s)}_{\text{Mapping}} + \underbrace{\mathcal{T}_{r\to s}^T(\mathbf{x})\Gamma_{s\to r}(\mathbf{x})}_{\text{Force Path Integral}}, \tag{10}$$

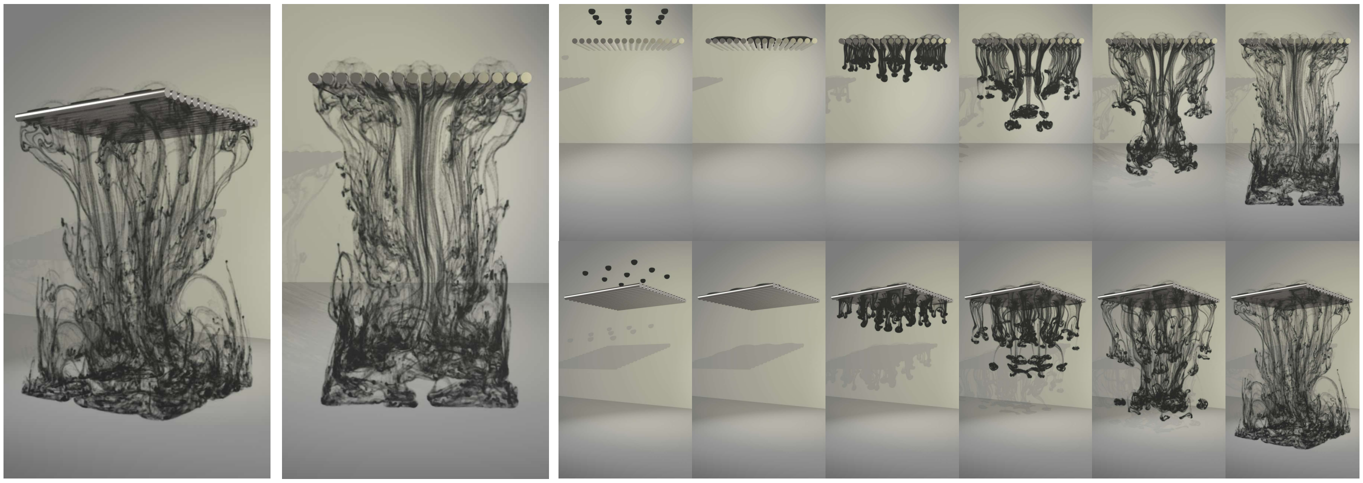

Fig. 6. Nine Ink Drops Passing Porous Obstacle. Nine ink drops drip down from the gaps between cylinders, turning into many small falling drops.

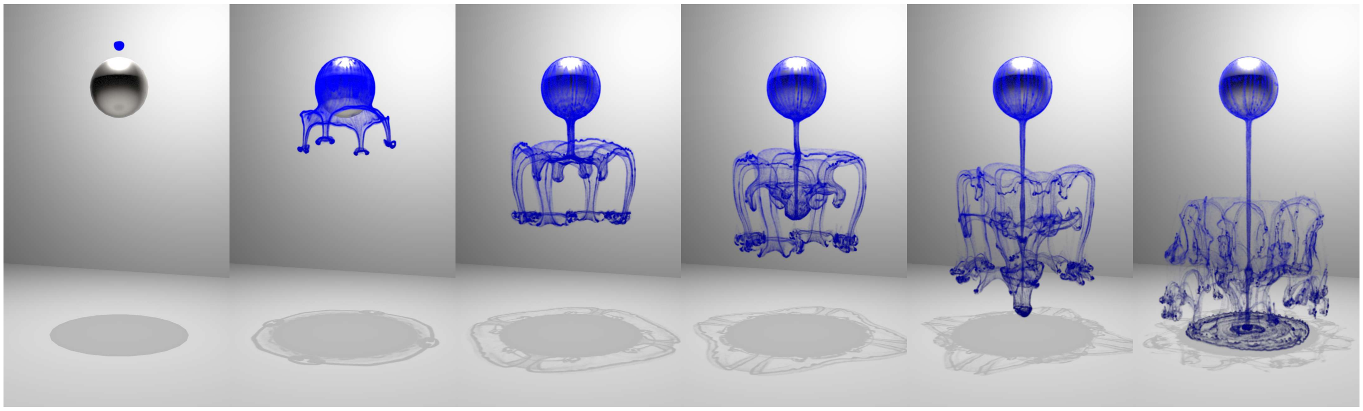

Fig. 7. One Ink Drop Passing Sphere Obstacle. One ink drop drops on a sphere, some parts form small drops flowing along the surface while some parts form a gathered drop at the sphere's bottom.

where $\Gamma_{s\to r}(\mathbf{x})$ is defined as

$$\Gamma_{s\to r}(\mathbf{x}) = \int_s^r \mathcal{F}^T_{s\to\tau}\Psi_{r\to s}(\mathbf{x})\left(\gamma - \frac{1}{\rho^f}\nabla\lambda\right)(\Phi_{s\to\tau}(\Psi_{r\to s}(\mathbf{x})),\tau)d\tau, \tag{11}$$

where $\lambda = p - \frac{1}{2}\rho^f|\mathbf{u}|^2$ is the Lagrangian pressure with density ρ^f.

Note that the integral $\Gamma_{s\to r}$ can be reformulated into a path integral form along a Lagrangian trajectory. For a fluid particle q with position $x_q(t)$ at time t, the force path integral term in Equation 10 is the path integral of $\mathcal{F}^T_{s\to t}\left(\gamma - \frac{1}{\rho^f}\nabla\lambda\right)$ along its trajectory from time s to r, which is denoted as

$$\Gamma_{s\to r,q} = \int_s^r \mathcal{F}^T_{s\to\tau,q}\left(\gamma - \frac{1}{\rho^f}\nabla\lambda\right)(\mathbf{x}_q(\tau),\tau)d\tau, \tag{12}$$

where $\mathcal{F}^T_{s\to\tau,q} = \mathcal{F}^T_{s\to\tau}(\mathbf{x}_q(s))$ represents the Jacobian of the forward flow map from s to τ along the particle trajectory, which could be carried on particles. Then Equation 10 can be reformulated as the path integral form as

$$\boxed{\mathbf{u}_{r,q} = \underbrace{\mathcal{T}_{r\to s,q}{}^T\mathbf{u}_{s,q}}_{\text{Mapping}} + \underbrace{\mathcal{T}^T_{r\to s,q}\Gamma_{s\to r,q}}_{\text{Force Path Integral}}.} \tag{13}$$

4.4 Mapped Velocity Conversion

To address the second and third challenges, we adopt a similar idea of Long-Range Mapping Classical Projection (LMCP) proposed in [Li et al. 2024]. We design a method based on Equation 13 to convert long-range mapped velocity $\mathbf{u}^M_{s\to r}$ into advected velocity $\mathbf{u}^A_{s'\to r}$, where $\mathbf{u}^A_{s'\to r}$ is the classical advected velocity advected by $\frac{D\mathbf{u}}{Dt} = 0$ one time step from s'. s' is one time step before r. $\mathbf{u}^A_{s'\to r}$ is used to interact with sediment particle and interaction is accumulated to long range by updating the path integrator, which allows us to address the second challenge. Simultaneously, by performing projection on $\mathbf{u}^A_{s'\to r}$, we resolve the third challenge.

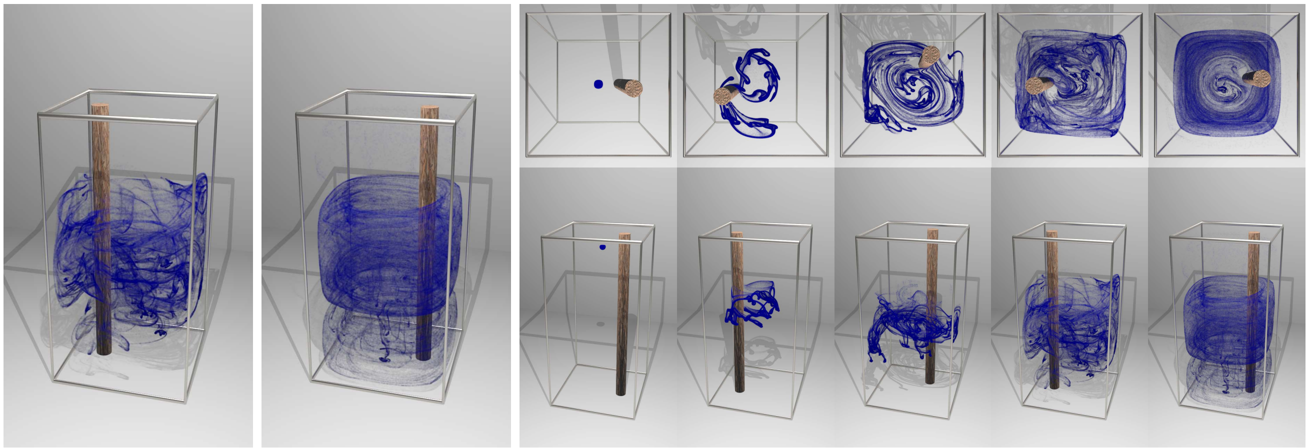

Fig. 8. Chopstick Stirring Ink. The stirred ink forms many layers and pulls out many fine threads, gradually spreading throughout the entire domain.

To present the formula for converting $\mathbf{u}^M_{s\to r,q}$ to $\mathbf{u}^A_{s'\to r,q}$, first, decompose $\mathcal{T}^T_{r\to s}\Gamma_{s\to r,q}$ as

$$\begin{aligned}\mathcal{T}^T_{r\to s}\Gamma_{s\to r,q} =&\mathcal{T}^T_{r\to s}\Gamma_{s\to s',q}\\ &+\mathcal{T}^T_{r\to s}\int_{s'}^{r}\mathcal{F}^T_{s\to\tau,q}\left(\gamma-\frac{1}{\rho^f}\nabla\lambda\right)(\mathbf{x}_q(\tau),\tau)d\tau.\end{aligned} \tag{14}$$

Here, the second part $\mathcal{T}^T_{r\to s}\int_{s'}^{r}\mathcal{F}^T_{s\to\tau,q}\left(\gamma-\frac{1}{\rho^f}\nabla\lambda\right)(\mathbf{x}_q(\tau),\tau)d\tau$ on the right-hand side of the above equation is equal to $\left(\gamma-\frac{1}{\rho^f}\nabla p+\frac{1}{2}\nabla|\mathbf{u}_{s'}|^2\right)(\mathbf{x}_q(r))\Delta t$ when using the Euler scheme to compute with the identity $\mathcal{T}^T_{r\to s}\mathcal{F}^T_{s\to r}=I$. Utilizing the relationship $\mathbf{u}_{r,q}=\mathbf{u}^M_{s\to r,q}+\mathcal{T}^T_{r\to s,q}\Gamma_{s\to r,q}=\mathbf{u}^A_{s'\to r,q}+\left(\gamma\Delta t-\frac{1}{\rho^f}\nabla p\right)(\mathbf{x}_q(r))$, we can derive the formula for converting $\mathbf{u}^M_{s'\to r,q}$ to $\mathbf{u}^A_{s'\to r,q}$ as

$$\boxed{\mathbf{u}^A_{s'\to r,q}=\mathbf{u}^M_{s\to r,q}+\mathcal{T}^T_{r\to s}\Gamma_{s\to s',q}+\frac{1}{2}(\nabla|\mathbf{u}_{s'}|^2)(\mathbf{x}_q(r))\Delta t.} \tag{15}$$

In the following discussion, we use $\mathbf{u}^A_{s'\to r}$ to represent the field indicated by $\mathbf{u}^A_{s'\to r,q}$ on all particles.

4.5 Force Effect

After converting $\mathbf{u}^M_{s\to r,q}$ to $\mathbf{u}^A_{s'\to r}$, we can incorporate forces γ and pressure p in the fluid into $\mathbf{u}^A_{s'\to r}$ in a conventional manner. Once γ is computed, it is applied to $\mathbf{u}^A_{s'\to r}$ yielding $\mathbf{u}^*=\mathbf{u}^A_{s'\to r}+\gamma\Delta t$. As $\mathbf{u}^*$ is derived from the classical advected velocity $\mathbf{u}^A_{s'\to r}$, the pressure p can be directly obtained from the variable coefficient Poisson equation Equation 18 derived from Equation 3. The projected velocity $\mathbf{u}_r$ is then computed as $\mathbf{u}_r=\mathbf{u}^*-\frac{1}{\rho_f}\nabla p$. Through this process, we address the third challenge.

Subsequently, it is necessary to accumulate the force γ and pressure $-\frac{1}{\rho_f}\nabla p$ into the path integrator $\Gamma_{s\to r,q}$. With $\Delta\Gamma=\left(\gamma-\frac{1}{\rho^f}\nabla p\right)$, $\Gamma_{s\to r,q}$ is updated by its definition Equation 11 as

$$\Gamma_{s\to r,q}=\Gamma_{s\to s',q}+\mathcal{F}^T_{s\to r_q}\left(\Delta\Gamma_q+\frac{1}{2}(\nabla|\mathbf{u}_{s'}|^2)(x_q(r))\right)\Delta t. \tag{16}$$

The updated $\Gamma_{s\to r,q}$ is used in the conversion between the mapped velocity and the advected velocity in the next step.

To be specific, we will substitute γ with specific force and delve into detailed discussions regarding the drag force $\frac{1}{\rho^f\epsilon^f}\mathbf{f}^f_{\text{drag}}$, viscous force $\frac{\mu}{\rho^f}\Delta\mathbf{u}^A_{s'\to r}$, and gravity $\mathbf{g}$ appearing in Equation 2. In the complete computation of laden flow, the effects of these three forces will be considered simultaneously according to the following discussion and summed up.

Drag Force. First, let's consider when γ is the drag force. The drag force originates from the interaction between sediment particles and fluid, and since both the advected velocity $\mathbf{u}^A_{s'\to s}$ and $\mathbf{v}_q$ are short-range, they can interact with each other. Utilizing the formula in Equation 5, we compute the drag force $\mathbf{f}_{\text{drag},p}$ acting on sediment particles. Then, employing $\mathbf{f}_{\text{drag},p}$ and Equation 6, we calculate $\mathbf{f}^f_{\text{drag}}$, and accumulate it onto $\mathbf{u}^A_{s'\to s}$ to obtain $\mathbf{u}^*$. After projection of $\mathbf{u}^*$, $\mathbf{f}^f_{\text{drag}}$ needs to be accumulated onto long-range using $\Delta\Gamma=\left(\frac{1}{\rho^f\epsilon^f}\mathbf{f}^f_{\text{drag}}-\frac{1}{\rho^f}\nabla p\right)$ and Equation 16. Through this process, we address the second challenge. The numerical computation process for $\mathbf{f}_{\text{drag},p}$ and $\mathbf{f}^f_{\text{drag}}$ is detailed in subsection 5.2.

Viscous Force and Gravity. When γ is the viscous force, we only need to compute $\Delta\mathbf{u}_{s'\to r}$. Then, we accumulate $\frac{\mu}{\rho^f}\Delta\mathbf{u}_{s'\to r}$ onto $\mathbf{u}^A_{s'\to r}$ to obtain $\mathbf{u}^*$. After projection, we can accumulate $\frac{\mu}{\rho^f}\Delta\mathbf{u}_{s'\to r}$ onto $\Gamma_{s\to r,q}$ using Equation 16. Gravity, on the other hand, can be directly accumulated onto $\mathbf{u}^A_{s'\to r}$ to obtain $\mathbf{u}^*$ and subsequently accumulated into $\Gamma_{s\to r,q}$ after projection.

4.6 Conclusion of Algorithm

Based on Equation 15 and subsection 4.5, we obtain our time-split scheme as:

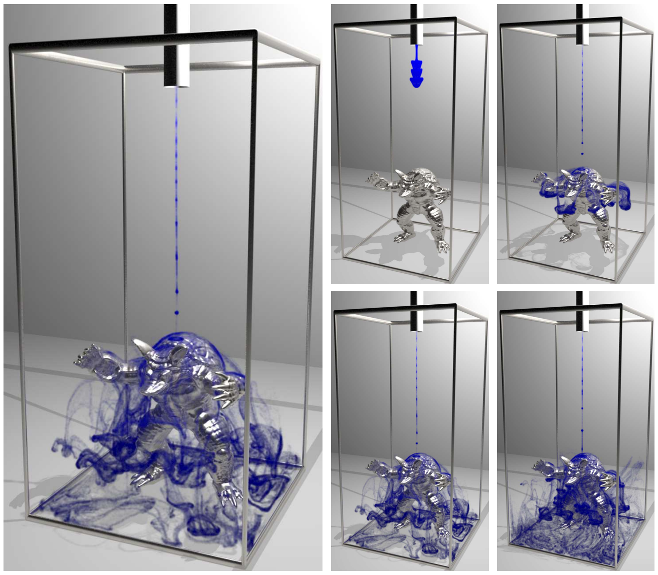

Fig. 9. Ink dropping onto an armadillo: once collides the ink splinters into numerous drops and coats the armadillo; descending drop tails bulge and break up into many small suspension drops.

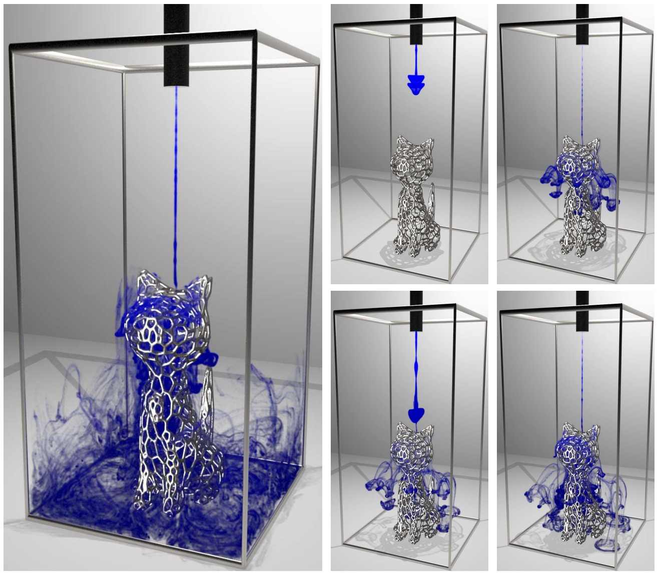

Fig. 10. Ink dropping onto a hollow cat: the ink interacts with the hollow cat and forms complex patterns as the ink passes through the hollow cat, illustrating the robustness of our method.

(1) (**Long-Range Mapping**) Calculate the long-range mapped velocity as: $\mathbf{u}^M_{s\to r,q} = \mathcal{T}_{r\to s,q}{}^T\mathbf{u}_{s,q}$;

(2) ($\mathbf{u}^M_{s\to r,q}$ **to** $\mathbf{u}^A_{s'\to r,q}$) Calculate $\frac{1}{2}(\nabla|\mathbf{u}_{s'}|^2)(\mathbf{x}_q(r))$ and convert $\mathbf{u}^M_{s\to r,q}$ to $\mathbf{u}^A_{s'\to r,q}$ based on Equation 15;

(3) (**Exert Force**) Update sediment particle dynamics and calculate force like viscous force $\frac{\mu}{\rho^f}\Delta\mathbf{u}^A_{s'\to r}$ and drag force $\frac{1}{\rho^f\epsilon^f}\mathbf{f}^f_{\text{drag}}$ on the grid, as discussion in subsection 4.5. Then calculate $\mathbf{u}^*$ as

$$\mathbf{u}^* = \mathbf{u}^A_{s'\to r} + \left(\mathbf{g} + \frac{\mu}{\rho^f}\Delta\mathbf{u}^A_{s'\to r} + \frac{1}{\rho^f\epsilon^f}\mathbf{f}^f_{\text{drag}}\right)\Delta t; \tag{17}$$

(4) (**Classical Projection**) Solve a variable coefficient Poisson equation derived from Equation 3 [Gao et al. 2018]:

$$-\frac{\Delta t}{\rho^f}\nabla\cdot(\epsilon^f\nabla p) = -\nabla\cdot(\epsilon^s\mathbf{v}) - \nabla\cdot(\epsilon^f\mathbf{u}^*); \tag{18}$$

(5) (**Update**) Project $\mathbf{u}^*$ by $\mathbf{u}_r = \mathbf{u}^* - \frac{1}{\rho^f}\nabla p$ and update $\Gamma_{s\to r,q}$ as Equation 16.

In the above scheme, detailed calculations regarding $\mathcal{T}_{s\to r,q}$, $\mathcal{F}_{s\to r,q}$, ϵ^f, $\frac{1}{2}(\nabla|\mathbf{u}_{s'}|^2)(\mathbf{x}_q(r))$, etc. are discussed in section 5.

5 Numerical Implementation

5.1 Fluid Calculation

We employ the hybrid grid-particle method to perform numerical computations for the time-split scheme detailed in subsection 4.6. Particles are used to track flow maps and compute the mapped velocity, and the grid is used to perform gradient and other differential operations. We use subscripts q and i to distinguish between quantities on particles and quantities on the grid, respectively. Each fluid particle carries particle initial velocity $\mathbf{u}_{s,q}$, Jacobian of backward flow map $\mathcal{T}_{s\to r}$, Jacobian of forward flow map $\mathcal{F}_{s\to r}$ and the force path integral $\Gamma_{s\to r,q}$. $\mathbf{x}_i$ and $\mathbf{x}_q$ denote positions of grid and particles respectively and in the processes of Particle-to-Grid Transfer and Grid-to-Particle Transfer, we use the quadratic kernel $w(\cdot)$ from [Jiang et al. 2016] with denoting $w_{iq} = w(\mathbf{x}_q - \mathbf{x}_i)$.

(1) Advection of Flow Maps. In [Deng et al. 2023], $\mathbf{x}_{r,q}$, $\mathcal{T}_{r\to s,q}$ and $\mathcal{F}_{s\to r,q}$ are advected using the Runge-Kutta 4th order method to solve $\frac{D\mathbf{x}_{r,q}}{Dt} = \mathbf{u}$ and Equation 7. The velocity used to evolve $\mathbf{x}_{r,q}$, $\mathcal{T}_{r\to s,q}$ and $\mathcal{F}_{s\to r,q}$ can either be the velocity $\mathbf{u}_{s',i}$ from the previous step or the midpoint time velocity $\mathbf{u}^{\text{mid}}_r$ calculated according to Algorithm 2 in [Deng et al. 2023]. Following [Deng et al. 2023], we use the midpoint time velocity to evolve $\mathbf{x}_{r,q}$, $\mathcal{T}_{r\to s,q}$ and $\mathcal{F}_{s\to r,q}$.

(2) Calculation of $\frac{1}{2}(\nabla|\mathbf{u}_{s'}|^2)(\mathbf{x}_q(r))$. The conversion from long-range mapped velocity to advected velocity occurs on the particles. However, the particles do not carry the quantity $\frac{1}{2}(\nabla|\mathbf{u}_{s'}|^2)(\mathbf{x}_q(r))$ in Equation 15. Therefore, it is necessary to compute the gradient $\frac{1}{2}(\nabla|\mathbf{u}_{s'}|^2)(\mathbf{x}_q(r))$ at the particle positions through interpolation, using the final velocity $\mathbf{u}_{s'}$ from the previous step on the grid.

$$\frac{1}{2}(\nabla|\mathbf{u}_{s'}|^2)(\mathbf{x}_q(r)) = \sum_i \frac{1}{2}|\mathbf{u}_{s',i}|^2\nabla w_{iq}. \tag{19}$$

(3) Particle-to-Grid Transfer. After calculating the mapped velocity on particles $\mathbf{u}^M_{s\to r,q} = \mathcal{T}^T_{r\to s,q}\mathbf{u}_{s,q}$, the mapped velocity is converted to short-range advected velocity on particles based on Equation 15. Then, the short-range advected velocity will be interpolated to the grid by APIC, similar to [Sancho et al. 2024] as

$$\mathbf{u}^A_{s'\to r,i} = \sum_q w_{iq}(\mathbf{u}^A_{s\to r,q} + A_q(\mathbf{x}_i - \mathbf{x}_q))/\left(\sum_q w_{iq}\right), \tag{20}$$

where affine matrix A_q is calculated during Grid-to-Particle Transfer.

(4) Volume Fraction and Sediment Velocity. When solving the Poisson Equation 18 and computing the drag force, the velocity field of sediment and the volume fractions of sediment and fluid on the grid are required. The velocity field and volume fractions of sediment on the grid are given by:

$$\epsilon_i^s = \frac{1}{\Delta x^3}\sum_p w_{ip}\frac{m_p}{\rho^s}, \quad \mathbf{v}_i = \sum_p w_{ip}m_p\mathbf{v}_p \Big/ \left(\sum_p m_p w_{ip}\right), \tag{21}$$

where Δx represents the size of a grid cell. Here, the particle-to-grid process of velocity is consistent with the particle-to-grid process of velocity in MPM [Jiang et al. 2016]. Then volume fraction of fluid is calculated as $\epsilon_i^f = 1 - \epsilon_i^s$.

(5) Force Calculation on Grid. On the grid, we need to compute the viscous force and drag force. The viscous force calculated by the finite difference using the six-point stencil on the grid is given by:

$$\left[\frac{\mu}{\rho^f}\Delta\mathbf{u}\right]_i = \frac{\mu\sum_{j\in N_i}(\mathbf{u}^A_{s'\to r,j}-\mathbf{u}^A_{s'\to r,i})}{\rho^f|N_i|\Delta\mathbf{x}^2}, \tag{22}$$

where N_i represents the neighboring cells of i, in three-dimensional space, $|N_i| = 6$. While the drag force is first computed for sediment particles $\mathbf{f}_{\text{drag},p}$ as subsection 5.2, then utilizing Equation 6, it is computed on the grid through the following process:

$$\mathbf{f}^f_{\text{drag},i} = -\frac{1}{\rho^f\Delta x^3}\sum_p \mathbf{f}_{\text{drag},p}w_{ip}. \tag{23}$$

After calculating the viscous force and drag force, $\mathbf{u}^*$ is calculated by exerting these forces on $\mathbf{u}^A$:

$$\mathbf{u}_i^* = \mathbf{u}^A_{s'\to r,i} + \left(\left[\frac{\mu}{\rho^f}\Delta\mathbf{u}\right]_i + \frac{1}{\rho^f\epsilon^f}\mathbf{f}^f_{\text{drag},i}\right)\Delta t. \tag{24}$$

(6) Pressure Projection. Similar to [Gao et al. 2018], on the grid, we solve the discretized variable coefficient Poisson equation Equation 18 with a multigrid preconditioner. After solving the Poisson equation, $\mathbf{u}_r$ is calculated as $\mathbf{u}_{r,i} = \mathbf{u}^A_{s'\to r,i} - \frac{1}{\rho^f}\nabla p_i$.

(7) Grid-to-Particle Transfer. During the Grid-to-Particle transfer process, p_i and γ_i need to be interpolated to particles and then used to update $\Gamma_{s\to r,q}$ by Equation 16 as:

$$\Gamma_{s\to r,q} = \Gamma_{s\to s',q} + \sum_i w_{iq}\gamma_i - \frac{1}{\rho^f}\sum_i p_i\nabla w_{iq} + \frac{1}{2}(\nabla|\mathbf{u}_{s'}|^2)(\mathbf{x}_q(r)). \tag{25}$$

The affine matrix in APIC should be calculated for the Particle-to-Grid process next step:

$$A_q = \sum_i \mathbf{u}_{r,i}\nabla w_{iq}. \tag{26}$$

5.2 Sediment Calculation

Implicit Scheme. Following [Sagong et al. 2015], to improve stability, we use implicit Euler to compute Equation 4 for updating particle velocities, where the numerical calculation form of Equation 4 and Equation 5 is written as:

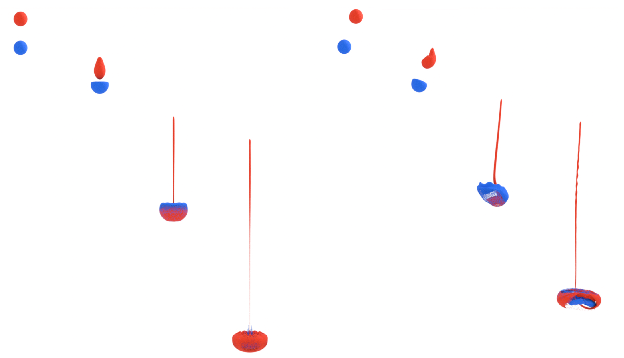

Fig. 11. Pass-Through of Two Drops. When two vertically aligned sediment drops fall simultaneously (left), the lower drop flattens into an oblate shape and the upper drop elongates into a prolate shape. The upper drop then catches up with and passes through the lower drop, and then two drops begin to mix. Our result is consistent with Figure 14 in the real experiments from [Machu et al. 2001]. When two horizontally offset drops fall (right), the upper drop moves into vertical alignment with the lower drop, and then, similar to vertically aligned drops, catches up, penetrates, and mixes with the lower drop, which is consistent with Figure 16 in the real experiments from [Machu et al. 2001].

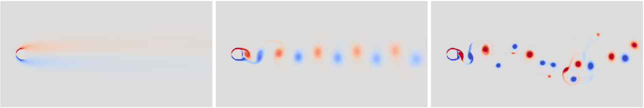

Fig. 12. Kármán vortex shedding: Re=25, 250, and 2500.

$$\begin{aligned} \frac{\mathbf{v}_p(r)-\mathbf{v}_p(s')}{\Delta t} &= \left(\frac{\rho^s-\rho^f}{\rho^s}\right)g + \frac{6\pi\mu r_p}{m_p}(\mathbf{u}_r(\mathbf{x}_p)-\mathbf{v}_p(r)), \\ \mathbf{f}_{\text{drag},p} &= \frac{6\pi\mu r_p}{m_p}(\mathbf{u}_r(\mathbf{x}_p)-\mathbf{v}_p(r)), \end{aligned} \tag{27}$$

where $\mathbf{u}_r(\mathbf{x}_p)$ is computed by interpolating the fluid velocity on the grid $\mathbf{u}_r(\mathbf{x}_p) = \sum_i w_{ip}\mathbf{u}_i$. After updating the particle velocities, the particle positions are updated as: $\mathbf{x}_p(r) = \mathbf{x}_p(s') + \mathbf{v}_p(s')\Delta t$.

Particle Cluster. When facing a physical scenario that includes a large number of sediment particles, computing the state of each particle using Equation 27 is computationally too expensive. To address this issue, following [Bosse et al. 2005], we introduce particle clusters. Each particle cluster represents a group of N sediment particles as a whole, assuming that the motion of these N particles is identical and described by the motion of the cluster. A particle cluster's motion follows Equation 27. When using clusters, the only modifications needed are to change Equation 23 to:

$$\mathbf{f}^f_{\text{drag},i} = -\frac{N}{\Delta x^3}\sum_p \mathbf{f}_{\text{drag},p}w_{ip}, \tag{28}$$

and change sediment volume fraction calculation in Equation 21 to

$$\epsilon_i^s = \frac{N}{\Delta x^3}\sum_p w_{ip}\frac{m_p}{\rho_s}. \tag{29}$$

Boundary Treatment. We represent solid obstacles using a level set function $L(\mathbf{x})$. After updating the position of any particle q, we check $L(\mathbf{x}_q)$. If $L(\mathbf{x}_q) < 0$, we move $\mathbf{x}_q$ along $\nabla L(\mathbf{x}_q)$ until $L(\mathbf{x}_q) \geq 0$, and update the particle velocity to $\mathbf{v}_q - (\mathbf{v}_q \cdot \nabla L(\mathbf{x}_q))\nabla L(\mathbf{x}_q)$.

6 Time Integration

We summarize our time integration scheme in Algorithm 1.

Algorithm 1 Laden Flow on Particle Flow Map

```
Initialize: $\mathbf{u}_i$, $\mathbf{v}_p$ to initial velocity; $\mathcal{T}_{s\to s}$, $\mathcal{F}_{s\to s}$ to $\boldsymbol{I}$
for $k$ in total steps do
    $j \leftarrow k \pmod{n^L}$;
    if $j = 0$ then
        Set initial time $s$ to now;
        Uniformly distribute fluid particles;
        Reinitialize $\mathcal{T}_{s\to s}$,$\mathcal{F}_{s\to s}$ to $\boldsymbol{I}$;
        Reinitialize velocity $\mathbf{u}_{s,q}$ by interpolating from $\mathbf{u}_{s,i}$;
    Compute $\Delta t$ with $\mathbf{u}_i$, $\mathbf{v}_p$ and the CFL number;
    Estimate midpoint velocity $\boldsymbol{u}_i^{\text{mid}}$;
    Advect $\boldsymbol{x}_q$, $\mathcal{T}_{r\to s}$, $\mathcal{F}_{s\to r}$ with $\boldsymbol{u}_i^{\text{mid}}$ and $\Delta t$;
    Calculate mapped velocity $\mathbf{u}^M_{s\to r,q}$ and convert to short-range advected velocity $\mathbf{u}^A_{s'\to r,q}$;   ▷ eq. 15
    Compute $\frac{1}{2}(\nabla|\mathbf{u}_{s'}|^2)(\mathbf{x}_q(r))$ by interpolation;   ▷ eq. 19
    Compute $\mathbf{u}^A_{s'\to r,i}$ by transfer $\mathbf{u}^A_{s'\to r,q}$ to grid;   ▷ eq. 20
    Compute volume fraction $\epsilon^s_i$,$\epsilon^f_i$ and sediment velocity $\mathbf{u}^s_i$ on the grid;   ▷ eq. 21
    Compute viscous force;   ▷ eq. 22.
    Compute drag force and update sediment state;   ▷ eq. 27, 23.
    Compute $\mathbf{u}^*_i$;   ▷ eq. 17.
    Compute pressure $p_i$ by solving Poisson equation and get final velocity $\mathbf{u}_{r,i}$ by projection;   ▷ eq. 18.
    Update path integration $\Gamma_{s\to r,q}$ in Grid-to Particle Transfer and calculate the affine matrix $A_q$;   ▷ eq. 25,26.
```

7 Results and Discussion

Validation. We first compare our simulation results with real-world phenomena to verify the accuracy of our method. **(1) Kármán Vortex Shedding.** We validate the correctness of adding viscosity as described in subsection 4.5 through 2D Kármán vortex shedding experiments. As illustrated in Fig. 12, the phenomena of Kármán vortex shedding are obtained through our simulations at different Reynolds numbers $Re = 25, 250$, and 2500. Our simulation revealed three distinct flow patterns at different Reynolds numbers: a laminar wake without a vortex street at low Re (25), a periodic vortex street at moderate Re (250), and turbulent mixing at high Re (2500). These results are consistent with the physical experiments reported in [Blevins 1977].

(2) Pass-through for Vertically Aligned Drop. Fig. 11 shows the phenomenon called pass-through when two vertically aligned sediment drops fall under low Reynolds number $Re = 3$, which is consistent with Figure 14 in the real experiments from [Machu et al. 2001]. **(3) Pass-through for Horizontally Offset Drop.** Fig. 11 shows the pass-through phenomenon when two horizontally offset drops fall, which is consistent with Figure 16 in the real experiments from [Machu et al. 2001]. **(4) Ink Torus Breakup Under Different Reynolds Number.** When a sediment drop falls, it forms an ink torus, which disintegrates into several blobs consequently. According to [Bosse et al. 2005], the number of blobs from the torus increases with the Reynolds Number. We use $Re =$ 16.5, 18, 19.5, 21.0, 30.0 and observe that the number of blobs gradually increases from 4 to 8, as shown in Fig. 5.

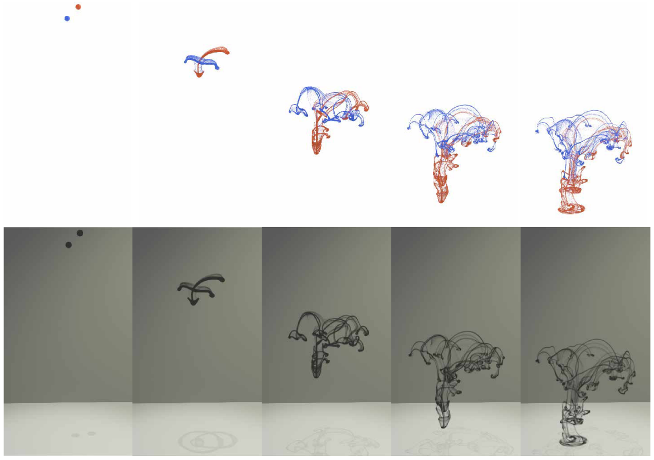

Fig. 13. Two Drop Interaction. At a high Reynolds Number, two drops interact and form a complex pattern.

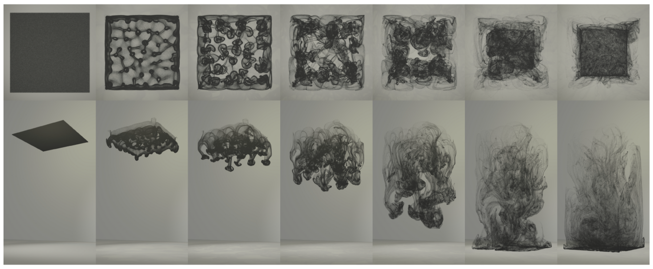

Fig. 14. RT Instability. The initially thin layer of ink at the top of the water tank gradually becomes unstable. Through the process of coalescence, it forms many small ink drops. These aggregated ink drops begin to fall and take on the classic figure shape of RT instability.

Comparison. Next, we demonstrate the effectiveness of our method by performing three experiments against four laden flow solvers, which use Semi-Lagrangian [Stam 1999], APIC [Jiang et al. 2015], Bimocq[Qu et al. 2019], and the basic flow-map method [Sato et al. 2017] to compute the fluid portion, respectively. The laden flow solver with the Semi-Lagrangian method of solving fluid on grids is based on [Sagong et al. 2015]. **(1) Ink Torus Breakup Comparison.** Fig. 16 compares the disintegration of the merged torus formed by the descent of two ink droplets, computed using different methods at $Re = 15$. **(2) Two Ink Drops Oblique Collision.** Fig. 15(above) compares the reconnection of ink torus when two ink tori collide with three methods. **(3) Three Ink Drops Oblique Collision.** Fig. 15 (below) compares the reconnection of ink torus when three ink tori collide with three methods.

Table 2. The catalog of all our simulation examples. #Fluid is the number of fluid particles per cell and #Sediment is the total number of sediment particle clusters

Name	Figure	Resolution	#Fluid	#Sediment	Time (sec /substep)	Memory Cost (GB)
Kármán Vortex Shedding	Figure 12	512 × 256	16	N/A	0.91	0.43
Ink Torus Breakup	Figure 4	128 × 256 × 128	8	1.0×10^6	1.84	11.49
Ink Torus Breakup(Under Different Re)	Figure 5	128 × 256 × 128	8	1.0×10^6	1.84	11.49
Pass-Through of Two Drops (Vertically Aligned)	Figure 11(left)	128 × 256 × 128	8	1.0×10^6	1.83	11.49
Pass-Through of Two Drops (Horizontally Offset)	Figure 11(right)	128 × 256 × 128	8	1.0×10^6	1.81	11.49
Two Ink Drops Oblique Collision Comparison	Figure 15(above)	128 × 256 × 128	8	1.0×10^6	1.85	11.49
Three Ink Drops Oblique Collision Comparison	Figure 15(below)	128 × 256 × 128	8	1.5×10^6	1.83	11.52
Ink Torus Breakup Comparison	Figure 16	128 × 256 × 128	8	1.0×10^6	1.84	11.49
Two Drop Interaction	Figure 13	128 × 256 × 128	8	1.0×10^6	1.81	11.49
Dripping	Figure 3	128 × 256 × 128	8	2.0×10^6	1.85	11.55
RT Instability	Figure 14	128 × 256 × 128	8	4.0×10^6	1.94	11.66
One Ink Drop Passing Sphere Obstacle	Figure 7	128 × 256 × 128	8	1.0×10^6	1.99	11.49
Chopstick Stirring Ink	Figure 8	128 × 256 × 128	8	1.0×10^6	2.06	11.49
Nine Ink Drops Passing Porous Obstacle	Figure 6	128 × 256 × 128	8	3.0×10^6	2.32	11.49
Ink Dropping onto an Armadillo	Figure 9	128 × 256 × 128	8	4.0×10^6	2.21	11.66
Ink Dropping onto a Hollow Cat	Figure 10	128 × 256 × 128	8	2.0×10^6	2.06	11.55

Examples. In our experiment, the fluid part is calculated using a $128 \times 128 \times 256$ grid, with each grid cell containing 8 fluid particles. For the sediment part, we use 1,000,000 to 4,000,000 particle clusters. $CFL = 0.5$ is used for calculate the Δt. We use Taichi [Hu et al. 2019] for our implementation, and experiments are run on Tesla A100 GPUs and each substep takes approximately 2.0 seconds. The details of memory cost and runtime are presented in Table 1. **(1) Two Drop Interaction.** In Fig. 13, at $Re = 30$, two horizontally offset drops interact as they fall, forming a complex pattern composed of ink rings and filaments. **(2) Rayleigh-Taylor (RT) Instability.** Fig. 14 shows a thin layer of ink turns unstable by gravity and forms small ink drops, leading to the classic figure shape of RT instability. **(3) Chopstick Stirring Ink.** As shown in Fig. 8, under the action of stirring, the ink initially dropped into the water tank gradually becomes thin and forms many layers, pulling out many fine threads. As the stirring time increases, the ink gradually spreads throughout the entire water tank. **(4) One Ink Drop Passing Sphere Obstacle.** In Fig.7, as an ink drop descends past a spherical obstacle, it flows over the surface of the sphere, and part of the ink coalesces into small drops near the bottom of the ink and then separates, while the remaining ink continues to flow along the surface of the sphere and gathers at the bottom. Under the influence of gravity, these gathered ink drops then fall downward. During falling, the aggregated ink continuously forms torus shapes and disintegrates into smaller blobs. **(5) Nine Ink Drops Passing Porous Obstacle.** As shown in Fig.6, when 9 drops of ink fall through a porous obstacle composed of several cylindrical structures, the ink drips down from the gaps between cylinders, forming many falling drops. These falling drops interact, creating a complex pattern. **(6) Dripping.** At the top of the water tank, we intermittently drip ink into the tank. As shown in Fig. 3 the ink gradually falls and impacts at the bottom of the tank. During the ink falling, a long tail forms between it and the dropper. This tail will bulge and break up, forming many small suspension drops, which is consistent with the real photographs in Figure 6 of [Machu et al. 2001]. As the small suspension drops continue to fall, they develop into tori, which is consistent with Figure 9. in [Rogers 1858]. **(7) Ink Dripping onto an Obstacle.** In Fig 9 and Fig. 10, we dripped ink onto complex solids to demonstrate the phenomena of ink and complex solid interaction, proving our method's robustness.

Discussion. Our particle-laden flow map model extends the simulation scope of traditional impulse/covector methods (e.g., [Deng et al. 2023; Feng et al. 2022; Nabizadeh et al. 2022] to the realm of viscosity-driven and particle-laden fluid phenomena. This scope extension is due to our method's new capability of tackling forces such as fluid viscosity and laden particle drag, which play an important role in producing physically-based vortical flow structures. Traditional impulse/covector methods such as [Deng et al. 2023; Nabizadeh et al. 2022] solve the Euler equation for inviscid incompressible flow that does not consider viscosity. Moreover, the force that sediment particles exert on the fluid cannot be incorporated into the traditional impulse/covector fluid models, limiting their modeling of interactions between sediment particles and their surrounding fluid. Last, the fluid portion of particle-laden fluid flow requires solving a variable-coefficient Poisson equation, which makes it difficult for solvers relying on solving a constant-coefficient Poisson system (e.g., the advection-reflection method [Zehnder et al. 2018]) to apply.

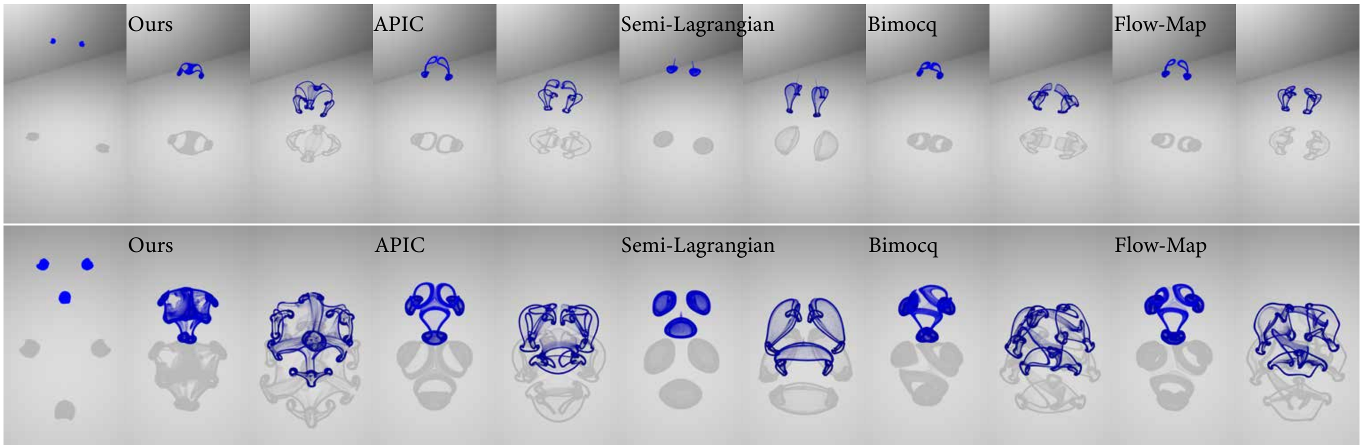

Fig. 15. Ink Drops Oblique Collision Comparison. When two Ink Drops collide obliquely (above), the tori formed by the two drops exhibits ring reconnection similar to vortex ring collisions. Compared to APIC, Semi-Lagrangian, Bimocq and basic flow-map methods, our method better maintains the vortex results, achieving ink ring reconnection uniquely. When three Ink Drops collide obliquely (below), only our method achieves ink ring reconnection compared to APIC, Semi-Lagrangian, Bimocq and basic flow-map methods, forming a new small ink ring at the center.

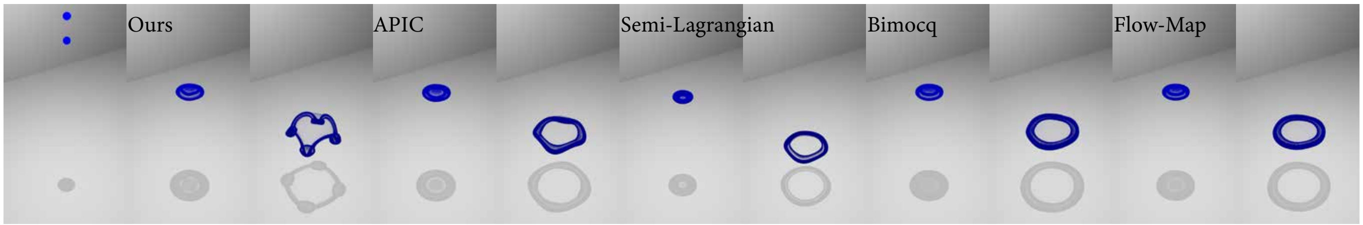

Fig. 16. Ink Torus Breakup Comparison. At $Re = 15$, when two vertically aligned drops fall, after they undergo the process of pass-through, mixing and forming a torus, the torus forming by two drops disintegrates into more rings as they continue to fall. Due to our method's lower numerical dissipation, the formed torus disintegrates into more rings more effectively.

8 Limitations and Future Work

In summary, we have enhanced the flow map method to accommodate forces beyond pressure and introduced a novel framework for simulating ink as a particle-laden flow using particle flow maps. The most significant limitation of our framework is that it only considers the interaction forces between the fluid and sediment, neglecting interactions between sediment particles. This limitation prevents the handling of phenomena such as sediment accumulation. We plan to incorporate interactions between sediment particles into our framework to enable the simulation of a broader range of laden flow phenomena. We also consider implementing adaptive particles to achieve larger-scale laden flow effects such as sand storms.

Acknowledgments

We express our gratitude to the anonymous reviewers for their insightful feedback. Georgia Tech authors also acknowledge NSF IIS #2433322, ECCS #2318814, CAREER #2433307, IIS #2106733, OISE #2433313, and CNS #1919647 for funding support. We credit the Houdini education license for video animations.

References

Robert D Blevins. 1977. Flow-induced vibration. *New York* (1977).

Thorsten Bosse, Leonhard Kleiser, Carlos Härtel, and Eckart Meiburg. 2005. Numerical simulation of finite Reynolds number suspension drops settling under gravity. *Physics of Fluids* 17, 3 (2005).

Jeremiah U Brackbill and Hans M Ruppel. 1986. FLIP: A method for adaptively zoned, particle-in-cell calculations of fluid flows in two dimensions. *Journal of Computational physics* 65, 2 (1986), 314–343.

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, Article 7, 126 pages. https://doi.org/10.1145/2504435.2504442

Yitong Deng, Mengdi Wang, Xiangxin Kong, Shiying Xiong, Zangyueyang Xian, and Bo Zhu. 2022. A moving eulerian-lagrangian particle method for thin film and foam simulation. *ACM Transactions on Graphics (TOG)* 41, 4 (2022), 1–17.

Yitong Deng, Hong-Xing Yu, Diyang Zhang, Jiajun Wu, and Bo Zhu. 2023. Fluid Simulation on Neural Flow Maps. *ACM Transactions on Graphics (TOG)* 42, 6 (2023), 1–21.

Yu Fang, Ziyin Qu, Minchen Li, Xinxin Zhang, Yixin Zhu, Mridul Aanjaneya, and Chenfanfu Jiang. 2020. IQ-MPM: an interface quadrature material point method for non-sticky strongly two-way coupled nonlinear solids and fluids. *ACM Transactions on Graphics (TOG)* 39, 4 (2020), 51–1.

Ronald Fedkiw, Jos Stam, and Henrik Wann Jensen. 2001. Visual simulation of smoke. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*. 15–22.

Fan Feng, Jinyuan Liu, Shiying Xiong, Shuqi Yang, Yaorui Zhang, and Bo Zhu. 2022. Impulse Fluid Simulation. *IEEE Transactions on Visualization and Computer Graphics* (2022).

Chuyuan Fu, Qi Guo, Theodore Gast, Chenfanfu Jiang, and Joseph Teran. 2017. A polynomial particle-in-cell method. *ACM Transactions on Graphics (TOG)* 36, 6 (2017), 1–12.

Ming Gao, Andre Pradhana, Xuchen Han, Qi Guo, Grant Kot, Eftychios Sifakis, and Chenfanfu Jiang. 2018. Animating fluid sediment mixture in particle-laden flows.

ACM Transactions on Graphics (TOG) 37, 4 (2018), 1–11.

Toshiya Hachisuka. 2005. Combined Lagrangian-Eulerian approach for accurate advection. In *ACM SIGGRAPH 2005 Posters*. 114–es.

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).

Yuanming Hu, Yu Fang, Ziheng Ge, Ziyin Qu, Yixin Zhu, Andre Pradhana, and Chenfanfu Jiang. 2018. A moving least squares material point method with displacement discontinuity and two-way rigid body coupling. *ACM Transactions on Graphics (TOG)* 37, 4 (2018), 1–14.

Yuanming Hu, Tzu-Mao Li, Luke Anderson, Jonathan Ragan-Kelley, and Frédo Durand. 2019. Taichi: a language for high-performance computation on spatially sparse data structures. *ACM Transactions on Graphics (TOG)* 38, 6 (2019), 201.

Sergio R Idelsohn, Juan M Gimenez, Rainald Löhner, and Eugenio Oñate. 2022. A multiscale approach for the study of particle-laden flows using a continuous model. *Computer Methods in Applied Mechanics and Engineering* 401 (2022), 115174.

Chenfanfu Jiang, Craig Schroeder, Andrew Selle, Joseph Teran, and Alexey Stomakhin. 2015. The affine particle-in-cell method. *ACM Transactions on Graphics (TOG)* 34, 4 (2015), 1–10.

Chenfanfu Jiang, Craig Schroeder, Joseph Teran, Alexey Stomakhin, and Andrew Selle. 2016. The material point method for simulating continuum materials. In *Acm siggraph 2016 courses*. 1–52.

A Kartushinsky, S Tisler, JL Goes Oliveira, and CWM Van der Geld. 2016. Eulerian-Eulerian modelling of particle-laden two-phase flow. *Powder Technology* 301 (2016), 999–1007.

Sarah Krainer, Chris Smit, and Ulrich Hirn. 2019. The effect of viscosity and surface tension on inkjet printed picoliter dots. *RSC advances* 9, 54 (2019), 31708–31719.

Sanboh Lee, HY Lee, IF Lee, and CY Tseng. 2004. Ink diffusion in water. *European journal of physics* 25, 2 (2004), 331.

Xingqiao Li, Xingyu Ni, Bo Zhu, Bin Wang, and Baoquan Chen. 2023. GARM-LS: A Gradient-Augmented Reference-Map Method for Level-Set Fluid Simulation. *ACM Transactions on Graphics (TOG)* 42, 6 (2023), 1–20.

Zhiqi Li, Barnabás Börcsök, Duowen Chen, Yutong Sun, Bo Zhu, and Greg Turk. 2024. Lagrangian Covector Fluid with Free Surface. In *ACM SIGGRAPH 2024 Conference Papers*. 1–10.

Gunther Machu, Walter Meile, Ludwig C Nitsche, and UWE Schaflinger. 2001. Coalescence, torus formation and breakup of sedimenting drops: experiments and computer simulations. *Journal of Fluid Mechanics* 447 (2001), 299–336.

Mohammad Sina Nabizadeh, Stephanie Wang, Ravi Ramamoorthi, and Albert Chern. 2022. Covector fluids. *ACM Transactions on Graphics (TOG)* 41, 4 (2022), 1–16.

Michael B Nielsen and Ole Østerby. 2013. A two-continua approach to Eulerian simulation of water spray. *ACM Transactions on Graphics (TOG)* 32, 4 (2013), 1–10.

JM Nitsche and GK Batchelor. 1997. Break-up of a falling drop containing dispersed particles. *Journal of Fluid Mechanics* 340 (1997), 161–175.

Marcel Padilla, Albert Chern, Felix Knöppel, Ulrich Pinkall, and Peter Schröder. 2019. On bubble rings and ink chandeliers. *ACM Transactions on Graphics (TOG)* 38, 4 (2019), 1–14.

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, Xinxin Zhang, Ming Gao, Chenfanfu Jiang, and Baoquan Chen. 2019. Efficient and conservative fluids using bidirectional mapping. *ACM Transactions on Graphics (TOG)* 38, 4 (2019), 1–12.

Karthik Raveendran, Chris Wojtan, and Greg Turk. 2011. Hybrid smoothed particle hydrodynamics. In *Proceedings of the 2011 ACM SIGGRAPH/Eurographics symposium on computer animation*. 33–42.

William B Rogers. 1858. ART. XXXIII.–On the Formation of Rotating Rings by Air and Liquids under certain conditions of discharge. *American Journal of Science and Arts (1820-1879)* 26, 77 (1858), 246.

Donghoon Sagong, Nahyup Kang, Junyong Noh, Xiaogang Jin, and Joseph S Shin. 2015. Simulating Drops Settling in a Still Liquid. *IEEE Computer Graphics and Applications* 35, 1 (2015), 46–55.

Sergio Sancho, Jingwei Tang, Christopher Batty, and Vinicius C Azevedo. 2024. The Impulse Particle-In-Cell Method. In *Computer Graphics Forum*. Wiley Online Library, e15022.

Takahiro Sato, Christopher Batty, Takeo Igarashi, and Ryoichi Ando. 2018. Spatially adaptive long-term semi-Lagrangian method for accurate velocity advection. *Computational Visual Media* 4, 3 (2018), 6.

Takahiro Sato, Takeo Igarashi, Christopher Batty, and Ryoichi Ando. 2017. A long-term semi-lagrangian method for accurate velocity advection. In *SIGGRAPH Asia 2017 Technical Briefs*. 1–4.

Jos Stam. 1999. Stable fluids. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*. 121–128.

Alexey Stomakhin, Craig Schroeder, Lawrence Chai, Joseph Teran, and Andrew Selle. 2013. A material point method for snow simulation. *ACM Transactions on Graphics (TOG)* 32, 4 (2013), 1–10.

Rui Sun and Heng Xiao. 2016. SediFoam: A general-purpose, open-source CFD–DEM solver for particle-laden flow with emphasis on sediment transport. *Computers & Geosciences* 89 (2016), 207–219.

Yuchen Sun, Xingyu Ni, Bo Zhu, Bin Wang, and Baoquan Chen. 2021. A material point method for nonlinearly magnetized materials. *ACM Transactions on Graphics (TOG)* 40, 6 (2021), 1–13.

Joseph John Thomson and Hugh Frank Newall. 1886. V. On the formation of vortex rings by drops falling into liquids, and some allied phenomena. *Proceedings of the royal society of London* 39, 239-241 (1886), 417–436.

Sigurdur T Thoroddsen, Takeharu Goji Etoh, and Kohsei Takehara. 2008. High-speed imaging of drops and bubbles. *Annu. Rev. Fluid Mech.* 40 (2008), 257–285.

K Shri Vignesh, Shruti Tandon, Praveen Kasthuri, and RI Sujith. 2022. Cluster formation in particle-laden flows is a continuous phase transition. *arXiv preprint arXiv:2201.04313* (2022).

Jens Honoré Walther and P Koumoutsakos. 2001. Three-dimensional vortex methods for particle-laden flows with two-way coupling. *J. Comput. Phys.* 167, 1 (2001), 39–71.

DC Wiggert and EB Wylie. 1976. Numerical predictions of two-dimensional transient groundwater flow by the method of characteristics. *Water Resources Research* 12, 5 (1976), 971–977.

Herman Wijshoff. 2018. Drop dynamics in the inkjet printing process. *Current opinion in colloid & interface science* 36 (2018), 20–27.

Zihao Xiu, Wen Nie, Jiayi Yan, Dawei Chen, Peng Cai, Qiang Liu, Tao Du, and Bo Yang. 2020. Numerical simulation study on dust pollution characteristics and optimal dust control air flow rates during coal mine production. *Journal of Cleaner Production* 248 (2020), 119197.

BH Xu and AB Yu. 1997. Numerical simulation of the gas-solid flow in a fluidized bed by combining discrete particle method with computational fluid dynamics. *Chemical Engineering Science* 52, 16 (1997), 2785–2809.

Shibiao Xu, Xing Mei, Weiming Dong, Zhiyi Zhang, and Xiaopeng Zhang. 2011. Interactive visual simulation of dynamic ink diffusion effects. In *Proceedings of the 10th International Conference on Virtual Reality Continuum and Its Applications in Industry*. 109–116.

Zhikun Xu, Tianyou Wang, and Zhizhao Che. 2023. Breakup of particle-laden droplets in airflow. *Journal of Fluid Mechanics* 974 (2023), A42.

Yonghao Yue, Breannan Smith, Christopher Batty, Changxi Zheng, and Eitan Grinspun. 2015. Continuum foam: A material point method for shear-dependent flows. *ACM Transactions on Graphics (TOG)* 34, 5 (2015), 1–20.

Jonas Zehnder, Rahul Narain, and Bernhard Thomaszewski. 2018. An advection-reflection solver for detail-preserving fluid simulation. *ACM Transactions on Graphics (TOG)* 37, 4 (2018), 1–8.

Gang Zhou, Junpeng Wang, Ruixin Song, Cuicui Xu, and Pengfei Wang. 2022. Experimental study on droplet breakup and droplet particles diffusion of a pressure nozzle based on PIV. *Chemical Engineering Science* 258 (2022), 117737.

Junwei Zhou, Duowen Chen, Molin Deng, Yitong Deng, Yuchen Sun, Sinan Wang, Shiying Xiong, and Bo Zhu. 2024. Eulerian-Lagrangian Fluid Simulation on Particle Flow Maps. *ACM Transactions on Graphics (TOG)* 43, 4 (2024), 1–20.

Bo Zhu, Xubo Yang, and Ye Fan. 2010. Creating and Preserving Vortical Details in Sph Fluid. In *Computer Graphics Forum*, Vol. 29. Wiley Online Library, 2207–2214.

Yongning Zhu and Robert Bridson. 2005. Animating sand as a fluid. *ACM Transactions on Graphics (TOG)* 24, 3 (2005), 965–972.

A Derivation of Eq.10

Similar to [Nabizadeh et al. 2022], we reformulated Equation 9 as $\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u}\cdot\nabla)\mathbf{u} + \nabla\mathbf{u}^T\cdot\mathbf{u} = \gamma - \frac{1}{\rho^f}\nabla\left(p + \frac{1}{2}\rho^f|\mathbf{u}|^2\right)$ and expressed it in covector form using the Lie derivative

$$\left(\frac{\partial}{\partial t} + L_{\mathbf{u}}\right)\mathbf{u}^\flat = \gamma^\flat - d\lambda \tag{30}$$

By integrating this equation of Lie derivative form, we obtain

$$\begin{aligned}\mathbf{u}_r^\flat &= \Psi_{r\longrightarrow s}^*\mathbf{u}_s^\flat + \int_s^t (\Phi_{s\longrightarrow\tau}\circ\Psi_{r\longrightarrow s})^*(\gamma_\tau^\flat - d\lambda_\tau)d\tau \\ &= \Psi_{r\longrightarrow s}^*\mathbf{u}_s^\flat + \Phi_{s\longrightarrow\tau}^*\int_s^t \Psi_{r\longrightarrow s}^*(\gamma_\tau^\flat - d\lambda_\tau)d\tau\end{aligned} \tag{31}$$

where $\Psi_{r\longrightarrow s}^*$ and $\Phi_{s\longrightarrow\tau}^*$ are the pullbacks of the covector induced by $\Psi_{r\longrightarrow s}$ and $\Phi_{s\longrightarrow\tau}$, respectively. Convert the above expression back to vector form, and note that $\Psi_{r\longrightarrow s}^*\mathbf{v}^\flat$ and $\Phi_{s\longrightarrow\tau}^*\mathbf{v}^\flat$ corresponds to $\nabla\Psi_{r\longrightarrow s}^T\mathbf{v}$ and $\nabla\Phi_{s\longrightarrow\tau}^T\mathbf{v}$ respectively for arbitrary vector field $\mathbf{v}$.