Semi-Lagrangian Pressure Solver for Accurate, Consistent, and Conservative Volume-of-Fluid Simulations

Julian Fox[1], Mark Owkes[1]

[1]Montana State University, Bozeman, Montana , USA

Abstract
In this work, a novel discretization of the incompressible Navier-Stokes equations for a gas-liquid flow is developed. Simulations of gas-liquid flows are often performed by discretizing time with a predictor $\rightarrow$ pressure $\rightarrow$ corrector approach and the phase interface is represented by a volume of fluid (VOF) method. Recently, unsplit, geometric VOF methods have been developed that use a semi-Lagrangian discretization of the advection term within the predictor step. A disadvantage of the current methods is that an alternative discretization (e.g. finite volume or finite difference) is used for the divergence operator in the pressure equation. Due to the inconsistency in discretizations, a flux-correction to the semi-Lagrangian advection term is required to achieve mass conservation, which increases the computational cost and reduces the accuracy. In this work, we explore the alternative of using a semi-Lagrangian discretization for the divergence operators in *both* the advection term and the pressure equation. The proposed discretization avoids the need to use a flux-correction to the semi-Lagrangian advection term as mass conservation is achieved through consistent discretization. Additionally, avoiding the flux-correction improves the accuracy while reducing the computational cost of the advection term semi-Lagrangian discretization.

Keywords
Volume of fluid, semi-Lagrangian, gas-liquid flows

Introduction

Atomizing sprays are prevalent in daily life, ranging from natural phenomena to industrial processes. Gaining insight into the underlying physics that drives these sprays is difficult, yet offers potential for more efficient and accurate utilization in various industries. Due to the nature of atomization, a rapid transformation of liquid into an opaque cloud, properly visualizing the dynamics present can be complicated. Experimental techniques have been employed to investigate these flows yet require complex techniques and often times only offer a two-dimensional representation of the spray [1, 2]. With the increase in computational power over the last couple of decades it is advantageous to use these resources to investigate the dynamics present in spray-type flows.

One characteristic of simulating atomizing, or gas-liquid, flows is the challenge of resolving the interface between the two phases and addressing the resultant discontinuities along that interface. This challenge has spurred a whole field of research deemed interface tracking and interface capturing. Interface tracking explicitly represents the interface offering an accurate yet computationally expensive solution, while, interface capturing represents the interface implicitly, often with lower computational cost [3, 4]. However, implicit methods sacrifice accuracy in representing the interface. This limitation has prompted much development within the field of interface capturing to improve the accuracy of these methods. One method considered state-of-the-art within interface capturing is the volume-of-fluid (VOF) method.

The volume-of-fluid (VOF) method uses a conserved scalar quantity to represent the ratio of liquid volume to cell volume within a computational cell [5, 6, 7, 8]. This scalar, known as the liquid volume fraction (VF), can be used to implicitly reconstruct the interface at each time step throughout the domain. The VOF method offers conservation of mass in the presence of the large density ratios and/or range of length scales that are present in atomizing sprays [9].

There exist a couple of different advection schemes for transporting the VF through the domain. The earliest methods used a split scheme, where the transport occurs in each dimension in separate steps, requiring re-evaluation of the interface for each dimensional step [6]. Proper implementation of these methods requires the handling of dilation terms to conserve liquid volume, which is equivalent to mass conservation [10].

To avoid the hurdles present in split VOF advection schemes, the unsplit VOF advection scheme was developed. In general unsplit methods differ from split methods by evaluating the transport of each dimension in a single step. Within the category of unsplit VOF methods there exist algebraic and geometric techniques, yet this work will focus on unsplit geometric VOF methods, for a broader perspective on VOF methods see [4]. The premise of unsplit geometric VOF schemes is to handle the transport of any conserved quantity by integrating over volumes formed by characteristics in space-time. Many implementations of this idea have been developed in two and three dimensions, notable examples include [11, 12, 13, 14]. By interpolating cell velocities to the vertex of each cell and projecting them back in time, flux regions can be computed that represent the flow into and out of that cell. We will be focused on developing upon the method proposed by Owkes and Desjardins [15], which we will refer to as the semi-Lagrangian method.

It can be computationally expensive to create the flux regions formed by the semi-Lagrangian discretization. Therefore, using the semi-Lagrangian to handle the transport of other conserved quantities would be advantageous, as the flux regions have already been computed to transport the interface. One implementation of this idea was done in [16] which utilized the flux regions to handle momentum transport offering accuracy and conservation of mass and momentum.

While this method is considered state-of-the-art, it still suffers from the need for a flux correction to ensure conservation of mass, which is a common issue for many of the contemporary VOF methods. This lies in the discrepancy between the discretization used to transport the interface and the discretization used to solve for a divergence-free velocity field.

Often these VOF methods to transport the interface are used in conjunction with a predictor $\rightarrow$ pressure $\rightarrow$ corrector approach, or projection method, first developed by Chorin [17]. In essence, the first step in this method is to solve for an intermediate velocity by solving the momentum equation and ignoring the pressure term. From here the pressure can be found by solving a Poisson equation using the intermediate velocity and enforcing the divergence-free constraint. Finally, the intermediate velocity can be corrected by applying the pressure term, creating a divergence-free flow field. Throughout the rest of this paper, we will refer to this method as the predictor-corrector method.

The need for a flux correction in the semi-Lagrangian method arises due to the different discretizations of the divergence operator used in the predictor-corrector method. Up to this point in the literature, the semi-Lagrangian method has only been used to handle interface transport and the calculation of the intermediate velocity, while a different finite volume or finite difference (FV/FD) discretization is used to solve for the pressure term. Consequently, the divergence-free constraint is upheld with respect to the FV/FD discretization and not the semi-Lagrangian discretization that is used to create the intermediate velocity. This requires the flux regions produced by the semi-Lagrangian to need a flux-correction to satisfy the divergence-free constraint.

In this work, the proposed method employs the semi-Lagrangian to transport mass and momentum and additionally to discretize the pressure term. The consistency obtained by using the semi-Lagrangian throughout the predictor-corrector method offers the benefit that the flux regions do *not* require a flux-correction to conserve mass. This implementation offers an increase in computational efficiency and accuracy of the simulation as the flux-corrections are computationally expensive and are non-physical in nature. Furthermore, the proposed method theoretically makes using the semi-Lagrangian method with unstructured meshes or immersed boundary methods less complex. This optimization to the method proposed in [16] has not been explored due to the calculation of the pressure term. Contemporary pressure solvers within the predictor-corrector method solve the Poisson equation, which has been solved with a multitude of numerical methods. Using the semi-Lagrangian to solve for the pressure leads to a Laplace-like equation that has not been identified elsewhere in the literature. Therefore, the main hurdle within this research is developing methods to solve for the pressure within this Laplace-like equation.

Methodology

4.1 Governing Equations

To handle the conservation of mass and momentum for two fluids, the "one-fluid formulation" will be used which is outlined by Tryggvason [18]. This method uses varying fluid properties to account for liquid and gas phases and a delta function to represent the surface tension force along the interface. Introducing varying fluid properties to the incompressible Navier-Stokes equations leads to the following set of equations:

$$\rho_i \frac{\partial \boldsymbol{u}}{\partial t} + \nabla \cdot \left(\rho_i \boldsymbol{u}\boldsymbol{u}^{\mathsf{T}}\right) = -\nabla p + \nabla \cdot \mu_i (\nabla \boldsymbol{u} + (\nabla \boldsymbol{u})^T) + \rho_i \boldsymbol{g} + \sigma \kappa \boldsymbol{n} \delta_S \tag{1}$$

$$\nabla \cdot \boldsymbol{u} = 0 \tag{2}$$

where $\boldsymbol{u}$ is the fluid velocity, t is time, p is pressure, g is gravity, σ is the surface tension coefficient, κ is the local interface curvature, and $\boldsymbol{n}$ is the approximate interface normal vector, and μ_i and ρ_i are the varying fluid properties for viscosity and density, respectively. We now have a form of the incompressible Navier-Stokes equations that define the whole flow field for two phases using one set of equations, defined as the "one-fluid formulation".

Within each computational cell of the mesh, we need to define a liquid volume fraction (VF) that represents the ratio of liquid to gas for that cell. Starting with a heaviside function, or liquid distribution function:

$$f(\boldsymbol{x}, t) = \begin{cases} 0 & \text{if } \boldsymbol{x} \text{ is in the gas at time t,} \\ 1 & \text{if } \boldsymbol{x} \text{ is in the liquid at time t,} \end{cases} \tag{3}$$

that offers a continuous representation of the liquid and gas within the domain. From here we need to discretize the liquid distribution function with regard to the computational mesh. This is defined as the volume integral of the heaviside function within the cell volume leading to:

$$\alpha_p = \frac{1}{V_p} \int_{V_p} f(\boldsymbol{x}, t) dV \tag{4}$$

where α_p is the VF at the pth computational cell, and V_p is the volume of the cell.

This liquid volume fraction is defined to represent interface location and fluid properties, where $\alpha_p = 1$ defines a cell containing only the liquid phase and $\alpha_p = 0$ defines a cell containing only the gaseous phase. From here we can use the VF to define our mixture density and viscosity to handle the jump discontinuities that exist for these properties along the interface. The mixture density and viscosity at the pth cell are defined as:

$$\rho_p = \rho_l \alpha_p - \rho_g (1 - \alpha_p) \tag{5}$$

$$\mu_p = \mu_l \alpha_p - \mu_g (1 - \alpha_p) \tag{6}$$

where the subscript $i = g$ or l for gas and liquid variables, respectively.

The advection of a conserved quantity, f, is defined as:

$$\frac{\partial f}{\partial t} + \nabla \cdot (f\boldsymbol{u}) = 0 \tag{7}$$

which comes from the material derivative of a conserved quantity for incompressible flows.

4.2 Semi-Lagrangian Discretization

As mentioned before, the semi-Lagrangian scheme uses cell vertex velocities to determine the flux of a conserved quantity through a cell over a time step. The scheme was developed with velocities stored along cell faces and any scalars such as pressure stored at cell centers. These face velocities are then interpolated to the cell vertices and projected back in time. Consequently, in two dimensions, a face will have two vertices, and thus, two velocities; whereas in three dimensions, a face will have four vertices, and accordingly, four velocities. Using these vertex velocities a flux region can be created which represents the flux coming into or out of that face of the computational cell. Taking into account all the flux regions for a given cell, the total flux through that cell can be determined.

One unique advantage of the semi-Lagrangian scheme is its ability to handle the discontinuities present in gas-liquid flows. For example, when handling transport for a cell containing an interface the corresponding flux region is recursively cut by the computational mesh. Once sliced by the mesh the flux volume is similarly cut by the phase interface until regions local to a single cell and phase are created. From here the conserved quantity to be transported can be integrated within the cut flux volume effectively dealing with the discontinuities present along the interface. Therefore, the semi-Lagrangian scheme allows for the evaluation of the advection of any conserved quantity through a cell for a given time step even in the presence of interface discontinuities.

For an in-depth derivation of the semi-Lagrangian see [15], in short, the discretization develops a relationship between the material evolution of a conserved quantity applied to a fixed control volume and the advection of the conserved quantity within a computational cell due to flux regions. The material evolution of a conserved scalar quantity $f(\boldsymbol{x}, t)$ within a solenoidal velocity field takes the form:

$$\frac{\partial f}{\partial t} + \nabla \cdot (f\boldsymbol{u}) = 0 \tag{8}$$

where x is the spatial coordinate, $\boldsymbol{u}$ is the velocity field, and t is time. If we integrate over a discrete time step and a fixed control volume, and then apply Gauss's theorem we can get a relationship between the change in the conserved scalar f within the control volume and the flux through the surface of the control volume (CV) described by:

$$\int_{CV} f(\boldsymbol{x}, t^{n+1}) - f(\boldsymbol{x}, t^{n}) dV + \int_{t^n}^{t^{n+1}} \oint_{CS} f\boldsymbol{u} \cdot \boldsymbol{n}_{CV} dS dt = 0 \tag{9}$$

At this point, the surface of the CV (cell faces) needs to be partitioned into sub-surfaces ∂CS_i. Each subsurface will get a flux volume, $\Omega_i(t)$, which has a bounding surface, $\omega_i(t)$, where positive and negative values represent flux coming into and out of the cell, respectively. Again, this time integrating over the sub-surfaces and applying Gauss' theorem and Leibniz's rule we can recast Eq. 8. This equation gives a relationship between the change in our conserved quantity within the CV, or cell face, and the fluxes coming into and out of the cell faces of the CV. This equation when made compact takes the form:

$$f^{n+1} = f^{n} - \sum_{s=1}^{N_s} \int_{\Omega_s} f(x, t^n) dV \tag{10}$$

where N_s is the number of faces for the CV, and Ω_i is the flux volume at each face i. Now we have a relationship between the transport of a conserved quantity through a cell and the flux of that quantity through each face of the cell.

Essentially, the Semi-Lagrangian discretization allows for a novel way to deal with divergence operators within our governing equations. In the next section, Sec. 4.3, we will expand on how the Semi-Lagrangian can be used to discretize divergence operators elsewhere in the computational algorithm.

4.3 Predictor-Corrector method/Projection method

Up to this point in the literature, a few different unsplit geometric VOF methods have been developed that offer conservation of mass and momentum yet require some form of flux-correction [14]. This requirement is caused by an inconsistency within the discretization techniques used to handle the transport of conserved quantities. The crux of this issue lies within how the predictor-corrector method is discretized, specifically, how the divergence operator is discretized.

In this work, the predictor-corrector method is used to integrate over time and employs the projection method developed by Chorin [17] to decouple the velocity and pressure fields present in the conservation of momentum equation. Together these methods allow for iterative solutions to the incompressible Navier-Stokes equations defined in Sec. 4.1. To illustrate the predictor-corrector method and eventually the consistent discretization proposed in this work, we start by discretizing time with the simplest Euler step. In practice, more accurate, iterative, methods are used but the key parts are equivalent. Discretizing the momentum equation with Euler's method leads to:

$$\frac{\boldsymbol{u}^{n+1}-\boldsymbol{u}^{n}}{\Delta t}+\nabla\cdot\left(\boldsymbol{u}\boldsymbol{u}^{\mathsf{T}}\right)=-\frac{1}{\rho_i}\nabla p+\frac{\mu_i}{\rho_i}(\nabla\boldsymbol{u}+(\nabla\boldsymbol{u})^{T})+\boldsymbol{g}+\frac{\sigma\kappa\delta}{\rho_i} \tag{11}$$

where the superscript indicates the time-step ($t^{n+1} = t^n + \Delta t$). At this point the pressure and velocity fields are still coupled, therefore an operator splitting technique that allows for the decoupling of these two terms is applied resulting in:

$$\frac{\boldsymbol{u}^{*}-\boldsymbol{u}^{n}}{\Delta t}+\nabla\cdot\left(\boldsymbol{u}\boldsymbol{u}^{\mathsf{T}}\right)=\frac{\mu_i}{\rho_i}(\nabla\boldsymbol{u}+(\nabla\boldsymbol{u})^{T})+\boldsymbol{g}+\frac{\sigma\kappa\delta}{\rho_i} \tag{12}$$

$$\frac{\boldsymbol{u}^{n+1}-\boldsymbol{u}^{*}}{\Delta t}=-\frac{1}{\rho_i}\nabla p^{n+1} \tag{13}$$

where u^* is the intermediate velocity and u^{n+1} is the velocity at the next time step. Eq. 12 is considered the *predictor* step where an intermediate velocity is calculated that neglects the pressure term and the second equation, Eq. 13, is known as the *corrector* step. The pressure in Eq. 13 is found such that u^{n+1} satisfies the continuity equation (Eq. 2). By solving this equation in this step-wise manner an intermediate velocity is calculated and projected onto the divergence-free subspace. This projection method results from ideas present in the Hodge decomposition which says that any vector field on a simply connected domain can be decomposed into divergence-free and curl-free components, see Bloomquist [19] for a more in-depth discussion on the use of the Hodge decomposition in this application. Solving Eq. 13 for u^{n+1} and taking the divergence, and ensuring that u^{n+1} is divergence-free leads to:

$$\nabla\cdot\left(\boldsymbol{u}^{*}-\frac{\Delta t}{\rho}\nabla p^{n+1}\right)=0 \tag{14}$$

In summary, the predictor-corrector method solves the continuity and momentum equations by first solving Eq. 12 for $\boldsymbol{u}^*$, then Eq. 14 is solved for p^{n+1}, and finally Eq. 13 is used to compute u^{n+1} completing the time step. Note that any scalars, including the color function, are transported at the same time Eq. 12 is solved using consistent fluxes between mass and momentum.

4.3.1 Traditional Inconsistent Discretization

Current discretizations (see e.g., [15]) of the predictor corrector method use a semi-Lagrangian method for advection terms ($\nabla \cdot (\boldsymbol{u}\boldsymbol{u}^\mathsf{T})$) and FV/FD discretizaton of Eq. 14. The FV/FD discretization is linear and allows for the distribution of the divergence operator to the velocity term and pressure leading to the traditional pressure Poisson equation $\nabla^2 p^{n+1} = \frac{\rho}{\Delta t}\nabla \cdot \boldsymbol{u}^*$. As referenced earlier, using the semi-lagrangian for advection terms and to discretize Eq. 12 but not Eq. 14 leads to the requirement of a flux-correction.

4.3.2 Proposed Consistent Discretization

In this work the divergence operators in both the advection term within Eq. 12 and Eq. 14 are discretized with a semi-Lagrangian formulation. However, the semi-Lagrangian operator is non-linear, thus the divergence operator in Eq. 14 can not be distributed. Therefore, Eq. 14 must be solved as is and can not be simplified to a Poisson equation. In its current implementation, the proposed methodology uses a discrete Newton method, described by Ortega [20], to solve Eq. 14.

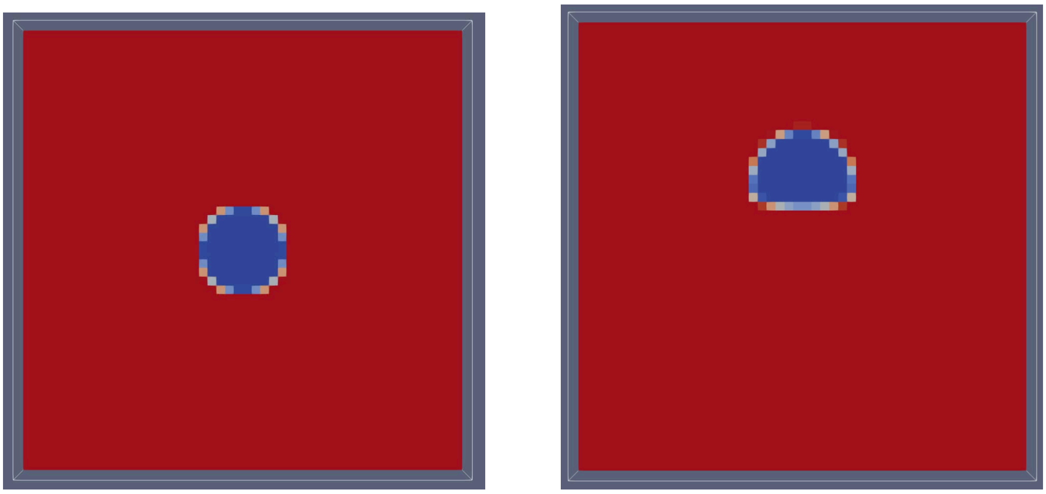
The proposed method alleviates the need for a flux-correction by using a consistent discretization for the divergence operators used in the advection terms and the pressure equation. Removing the flux-correction will reduce the computational cost of evaluating the advection terms and avoid the non-physical effects of adding the correction.

Results and Discussions

In this section, we present the initial findings of our computational algorithm, offering a preliminary glimpse into its potential efficacy and performance. The test consists of a rising bubble discretized on a 50 by 50 mesh and serves as a proof-of-concept of the proposed methodology. Fig. 1 showcases the simulation results, a coarse rising bubble can be seen as well as the expected deformation as the bubble is rising.

Not only does the initial result indicate the proposed methodology functions as expected, but it also illustrates the difference in computational cost when compared to an algorithm similar to what was proposed in [15]. The main difference between these two algorithms was how the pressure term was calculated. In the proposed methodology the flux-correction is avoided by the use of the semi-Lagrangian for the divergence operator in the pressure calculation in conjunction with semi-Lagrangian advection of conserved quantities. On the other hand, the contemporary method uses a FV/FD method to handle the pressure calculation while dealing with advection of conserved quantities using the semi-Lagrangian. The downside of the contemporary method has to do with the inconsistent discretizations of the divergence operators present in the pressure calculation and advection steps. It was observed that the computational cost of the proposed methodology, in its current implementation, was about 10% more computationally expensive than [15].

The main avenue for increasing the computational efficiency of the algorithm deals with the pressure calculation. The discretized Newton method requires a rather expensive Jacobian to be created at each time step.



(a) 50^2 mesh, t=0.0 **(b)** 50^2 mesh, t=0.5

Figure 1: Proof-of-concept results from a 50^2 simulation of a rising bubble.

Conclusions

In this paper, we have developed a novel computational algorithm that offers a consistent discretization to handle interface advection and the predictor-corrector method. This work builds off ideas present in [16] where the semi-Lagrangian method was employed to handle advection of scalars (such as VF) and momentum. One potential downside of that implementation is the computationally expensive flux-corrections required to conserve mass and momentum. The proposed computational algorithm not only employs the semi-Lagrangian method to handle advection of scalars and momentum, but also to solve for the pressure term within the predictor-corrector method. Using the semi-Lagrangian discretization to handle the pressure calculation avoids the need for a computationally expensive flux-correction. As flux-corrections are non-physical avoiding that step would improve the accuracy of the simulation. Additionally, alleviating the need for flux-corrections allows for less complicated implementations of unstructured meshes and immersed boundary methods. When using the semi-Lagrangian discretization for an unstructured mesh many cases cause non-intersecting fluxes or overlapping fluxes after flux-corrections have been applied leading to conservation errors [14].

The main contribution to the community from this work is that we have developed a computational algorithm that offers a consistent divergence discretization using the semi-Lagrangian method and shown that it works. Not only have we tentatively demonstrated the validity of the method but also described its potential for increased accuracy and computational efficiency when compared to other contemporary unsplit geometric VOF methods. Future work will focus on the development of a more computationally efficient method to solve for the pressure term, while more thoroughly validating the method's accuracy.

Acknowledgement

This material is based upon work supported by the National Science Foundation under Grant No. 1749779

References

[1] Benjamin R. Halls et al. "High-speed, two-dimensional synchrotron white-beam x-ray radiography of spray breakup and atomization". In: *Optics Express* 25 (2 Jan. 2017), p. 1605. ISSN: 10944087. DOI: 10.1364/oe.25.001605.

[2] Megan Paciaroni et al. "Ballistic Imaging of Atomizing Spray Liquid Core 51 Single-Shot Two-Dimensional Ballistic Imaging of the Liquid Core in an Atomizing Spray". In: *Atomization and Sprays* 16 (2006), pp. 51–69.

[3] S. McKee et al. "The MAC method". In: *Computers and Fluids* 37 (8 Sept. 2008), pp. 907–930. ISSN: 00457930. DOI: 10.1016/j.compfluid.2007.10.006.

[4] S Mirjalili, S S Jain A N, and D M S Dodd. "Interface-capturing methods for two-phase flows: An overview and recent developments". In: (2017).

[5] James E Pilliod and Elbridge G Puckett. "Second-Order Accurate Volume-of-Fluid Algorithms for Tracking Material Interfaces". In: (1997).

[6] C. W. Hirt and B. D. Nichols. "Volume of fluid (VOF) method for the dynamics of free boundaries". In: *Journal of Computational Physics* 39 (1 Jan. 1981), pp. 201–225. ISSN: 0021-9991. DOI: 10.1016/0021-9991(81)90145-5.

[7] Carlos M. Lemos. "Higher-order schemes for free surface flows with arbitrary configurations". In: *International Journal for Numerical Methods in Fluids* 23 (6 Sept. 1996), pp. 545–566. ISSN: 02712091. DOI: 10.1002/(sici)1097-0363(19960930)23:6<545::aid-fld440>3.0.co;2-r.

[8] Ruben Scardovelli and Stéphane Zaleski. "Direct numerical simulation of free-surface and interfacial flow". In: *Annual Review of Fluid Mechanics* 31 (1999), pp. 567–603. ISSN: 00664189. DOI: 10.1146/annurev.fluid.31.1.567.

[9] Vincent Le Chenadec and Heinz Pitsch. "A monotonicity preserving conservative sharp interface flow solver for high density ratio two-phase flows". In: *Journal of Computational Physics* 249 (2013), pp. 185–203. ISSN: 0021-9991. DOI: https://doi.org/10.1016/j.jcp.2013.04.027. URL: https://www.sciencedirect.com/science/article/pii/S0021999113002921.

[10] G. D. Weymouth and Dick K.P. Yue. "Conservative Volume-of-Fluid method for free-surface simulations on Cartesian-grids". In: *Journal of Computational Physics* 229 (8 Apr. 2010), pp. 2853–2865. ISSN: 0021-9991. DOI: 10.1016/J.JCP.2009.12.018.

[11] Vincent Le Chenadec and Heinz Pitsch. "A 3D Unsplit Forward/Backward Volume-of-Fluid Approach and Coupling to the Level Set Method". In: *Journal of Computational Physics* 233 (1 Jan. 2013), pp. 10–33. ISSN: 0021-9991. DOI: 10.1016/J.JCP.2012.07.019.

[12] Joaquin López et al. "A volume of fluid method based on multidimensional advection and spline interface reconstruction". In: *Journal of Computational Physics* 195 (2 Apr. 2004), pp. 718–742. ISSN: 0021-9991. DOI: 10.1016/J.JCP.2003.10.030.

[13] J. Hernández et al. "A new volume of fluid method in three dimensions - Part I: Multidimensional advection method with face-matched flux polyhedra". In: *International Journal for Numerical Methods in Fluids* 58 (8 Nov. 2008), pp. 897–921. ISSN: 02712091. DOI: 10.1002/fld.1776.

[14] Christopher B. Ivey and Parviz Moin. "Conservative and bounded volume-of-fluid advection on unstructured grids". In: *Journal of Computational Physics* 350 (Dec. 2017), pp. 387–419. ISSN: 10902716. DOI: 10.1016/j.jcp.2017.08.054.

[15] Mark Owkes and Olivier Desjardins. "A computational framework for conservative, three-dimensional, unsplit, geometric transport with application to the volume-of-fluid (VOF) method". In: *Journal of Computational Physics* 270 (Aug. 2014), pp. 587–612. ISSN: 10902716. DOI: 10.1016/j.jcp.2014.04.022.

[16] Mark Owkes and Olivier Desjardins. "A mass and momentum conserving unsplit semi-Lagrangian framework for simulating multiphase flows". In: *Journal of Computational Physics* 332 (Mar. 2017), pp. 21–46. ISSN: 10902716. DOI: 10.1016/j.jcp.2016.11.046.

[17] Alexandre Joel Chorin. “A Numerical Method for Solving Incompressible Viscous Flow Problems*”. In: *Journal of Computational Physics* 135 (1997), pp. 118–125.

[18] Gretar Tryggvason et al. “Direct numerical simulations of gas/liquid multiphase flows”. In: *Fluid Dynamics Research* 38 (9 Sept. 2006), pp. 660–681. ISSN: 0169-5983. DOI: 10.1016/J.FLUIDDYN.2005.08.006.

[19] Matthew Blomquist et al. “Stable nodal projection method on octree grids”. In: *Journal of Computational Physics* 499 (Feb. 2024). ISSN: 10902716. DOI: 10.1016/j.jcp.2023.112695.

[20] J. M. Ortega and W. C. Rheinboldt. *Iterative Solution of Nonlinear Equations in Several Variables*. Society for Industrial and Applied Mathematics, 2000. DOI: 10.1137/1.9780898719468. eprint: https://epubs.siam.org/doi/pdf/10.1137/1.9780898719468. URL: https://epubs.siam.org/doi/abs/10.1137/1.9780898719468.