Journal of Scientific Computing manuscript No.
(will be inserted by the editor)

A Coupled Level Set-Moment of Fluid Method for Incompressible Two-Phase Flows

Matthew Jemison · Eva Loch · Mark Sussman · Mikhail Shashkov · Marco Arienti · Mitsuhiro Ohta · Yaohong Wang

Received: date / Accepted: date

Abstract A coupled level set and moment of fluid method (CLSMOF) is described for computing solutions to incompressible two-phase flows. The local piecewise linear interface reconstruction (the CLSMOF reconstruction) uses information from

Work supported in part by the National Science Foundation under contracts DMS 0713256, DMS 1016381. M. Sussman also acknowledges the support by United Technologies Research Center and Sandia National Labs. M. Arienti acknowledges the support by Sandia National Laboratories via the Early Career LDRD program. Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U. S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000. The work of M. Shashkov was performed under the auspices of the National Nuclear Security Administration of the US Department of Energy at Los Alamos National Laboratory under Contract No. DE-AC52-06NA25396 and partially supported by the DOE Advanced Simulation and Computing (ASC) program and the DOE Office of Science ASCR Program.

Matthew Jemison
Department of Applied & Computational Mathematics, Florida State University
E-mail: mjemison@math.fsu.edu

Eva Loch
Institut für Geometrie und Praktische Mathematik, RWTH Aachen University, Germany
E-mail: eloch@igpm.rwth-aachen.de

Mark Sussman
Department of Applied & Computational Mathematics, Florida State University
E-mail: sussman@math.fsu.edu

Mikhail Shashkov
X-Computational Physics Division, Los Alamos National Laboratory
E-mail: shashkov@lanl.gov

Marco Arienti
Thermal/Fluid Science and Engineering, Sandia National Labs, Livermore
E-mail: marient@sandia.gov

Mitsuhiro Ohta
Division of Applied Sciences, Graduate School of Engineering, Muroran Institute of Technology
E-mail: mohta@mmm.muroran-it.ac.jp

Yaohong Wang
Department of Mathematics, University of California, Santa Barbara
E-mail: ywang@math.ucsb.edu

the level set function, volume of fluid function, and reference centroid, in order to produce a slope and an intercept for the local reconstruction. The level set function is coupled to the volume-of-fluid function and reference centroid by being maintained as the signed distance to the CLSMOF piecewise linear reconstructed interface.

The nonlinear terms in the momentum equations are solved using the sharp interface approach recently developed by Raessi and Pitsch (2009). We have modified the algorithm of Raessi and Pitsch from a staggered grid method to a collocated grid method and we combine their treatment for the nonlinear terms with the variable density, collocated, pressure projection algorithm developed by Kwatra et al (2009). A collocated grid method makes it convenient for using block structured adaptive mesh refinement (AMR) grids. Many 2D and 3D numerical simulations of bubbles, jets, drops, and waves on a block structured adaptive grid are presented in order to demonstrate the capabilities of our new method.

Keywords moment of fluid · volume of fluid · level set · two-phase flow · deforming boundaries

1 Introduction

Recently, the moment of fluid (MOF) method [14,15,3,2,4,1,32] interface capturing method has been developed for computing the motion of deforming boundary problems. The moment of fluid method has been demonstrated to give more accurate results than volume of fluid (VOF) methods, level set methods, or coupled level set and volume of fluid (CLSVOF) methods [42,11] for deforming boundaries that have corners or thin filaments. For example, we illustrate in sections 7.3 and 7.4 below (rotating notched disk and rotating letter "A" respectively), that the error associated with the directional split MOF interface advection scheme on a coarse grid is as small as the errors admitted by the directional split LVIRA [29] or CLSVOF [39] methods on the next finer grid.

In this work, we develop a directional split, coupled level set-moment of fluid (CLSMOF) interface capturing algorithm in order to compute incompressible (2D or 3D) two phase flows that have surface tension. The level set function has two main purposes: (1) the level set slope is used to accelerate to convergence the MOF slope reconstruction step, and (2) the level set function is used in order to determine the location of the interface when implementing the mass weighted nonlinear advection scheme developed by [31] (5.2), the mass weighted interpolation of cell center velocity to face centers [22] (5.12), the "inverse mass weighted" interpolation of cell center pressure to face centers [22] (5.24), and the ghost fluid treatment for surface tension [21] (5.27).

In this paper, we have also hybridized the MOF slope reconstruction with the CLSVOF slope reconstruction in order to develop a CLSMOF slope reconstruction algorithm. The MOF slope reconstruction procedure is defined as follows [14,15,3, 2,4,1]: given a reference volume fraction F^{Ref} and a reference centroid $\boldsymbol{x}^{Ref}$, find the slope $\boldsymbol{n}$ and the intercept b so that the actual volume fraction F^A equals F^{Ref}, and the difference between the actual centroid $\boldsymbol{x}^A$ and the reference centroid is minimized. In other-words,

$$(\boldsymbol{n}, b) = \underset{\boldsymbol{n}, b, F^{Ref}=F^A}{\arg\min} ||\boldsymbol{x}^{Ref} - \boldsymbol{x}^A||. \tag{1.1}$$

The CLSMOF slope reconstruction is a modification of the MOF slope reconstruction (1.1). Suppose Δx is the length of a computational cell in which a slope reconstruction is sought. In smooth regions of a deforming interface where $||\boldsymbol{x}^{Ref} - \boldsymbol{x}^{A}|| > 10^{-8}\Delta x$, the CLSMOF interface reconstruction derives the slope from the level set function. In regions of large curvature, or if $||\boldsymbol{x}^{Ref} - \boldsymbol{x}^{A}|| < 10^{-8}\Delta x$, the MOF reconstruction is used to derive the slope. Details of exactly when we choose the level set function in defining the slope, and when we choose the MOF reconstruction (1.1) in defining the slope is presented in section 4.1 below. We refer the reader to Figure 1 for an illustration of the cell moments $\boldsymbol{x}^{Ref}$ (denoted by the filled circles), cell volume fractions (denoted by V in the figure), and the level set function ϕ which is the signed distance of the cell center (open circles) to the piecewise linear reconstructed interface.

We apply our newly developed CLSMOF slope reconstruction algorithm to solving incompressible two phase flow problems on adaptive grids. All advection terms are discretized in time using directional splitting [34]. We have developed a sharp interface method using the ideas proposed in [31] for treating the nonlinear advective terms when there is a density jump at the gas-liquid interface. The algorithm by [31] was implemented on a staggered grid. We have extended [31] to an unstaggered grid algorithm and implemented a cell centered, variable density projection method developed by [22]. An important contribution from [31] and [22] is that the interpolation of velocity should be mass weighted (5.2), (5.12). Another critical development from [22] was that the interpolation of pressure from cell centers to faces, in order to update the cell center velocity, should be "inverse mass" weighted (5.24). [22] applied their techniques for simulating compressible flows with strong shocks, and we have discovered that their interpolation ideas work well for multiphase flows (with density ratio corresponding to air and water) too.

By storing both the velocity and pressure at the cell centers, it is convenient to reuse existing algorithms for block structured adaptive mesh refinement [5] and it is convenient to reuse existing algorithms for calculating the viscous or viscoelastic force terms [33]. All forces are calculated at the cell centers. The surface tension force is discretized using the "ghost fluid method" [21].

Many of our results in section 7 are computed on a dynamic block structured adaptive grid [5,37,36]. We remark that quadtree adaptive mesh refinement was implemented by [2]. In order to implement CLSMOF on an adaptive grid, we developed routines for transferring fine grid volume fraction and centroid data to an underlying coarse grid, and routines for interpolating the coarse grid volume fraction and centroid data to a finer level in order to prescribe boundary conditions at a coarse-fine border. We simultaneously interpolate both volume fraction and centroid information at the coarse-fine borders. In a given coarse cell, first the MOF interface reconstruction algorithm is carried out in order to define a local interface reconstruction in the coarse cell. Then, the local interface reconstruction is used to define the volume of fluid function and centroid information in the four (in two dimensions) ghost fine cells above the coarse cell. The criterion for adaptivity in most of our calculations in section 7 is that cells in which the volume-of-fluid function F satisfies $0.1 \leq F \leq 0.9$ or in which the level set function ϕ changes sign are tagged for adaptivity. In Figure 16, we also present a result in which the curvature is an additional indicator for refining the mesh.

We remark that the development of numerical methods for multiphase flow is an active field, the following is a representative list of the current state of the art: [11,16,40,31,18,21,30,26,19]. The developments that we report in this paper represent the first application of the MOF concepts to capturing interfaces in incompressible two-phase flows that include three dimensional effects and the effects of surface tension. In Section 7.9 we show an example of the simulation of three dimensional impinging jets. The jets merge, a spray sheet is formed, and then the sheet breaks up into ligaments and drops. The CLSMOF method has better volume conservation throughout the impinging jet simulation than the CLSVOF method.

Outline The remainder of this article is organized as follows. Section 2 gives the governing equations for incompressible two phase flows. Section 3 gives an overview of our CLSMOF algorithm. Section 4 describes our CLSMOF method for local interface reconstruction and directional split interface advection. Section 5 describes our collocated sharp interface method for integrating the solutions to the Navier-Stokes equations for two phase flows. Also in Section 5, we describe the ghost fluid method for surface tension on a collocated grid. Section 6 describes our implementation of the CLSMOF on a dynamic block structured adaptive mesh refinement grid. Our new results are described in Section 7 in which we compare results using our new CLSMOF interface reconstruction algorithm to results using CLSVOF [39], MOF, and LVIRA [29] interface reconstruction. All tests are performed using directional splitting for interface and momentum advection [34]. Finally, Section 8 gives the conclusions.

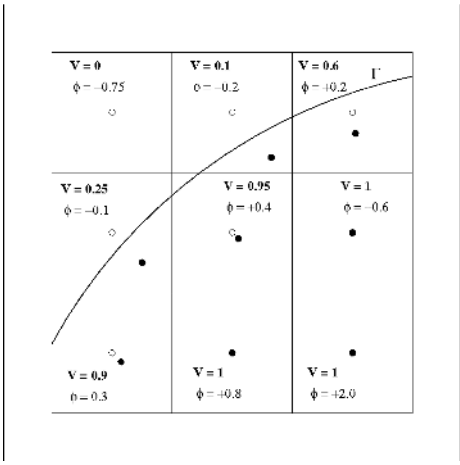


Fig. 1 Illustration of the cell moments $\boldsymbol{x}^{Ref}$ (denoted by the filled circles), cell volume fractions (denoted by V in the figure), and the level set function ϕ which is the signed distance of the cell center (open circles) to the piecewise linear reconstructed interface. The open and filled circles coincide for cells that are not cut by the interface.

2 Governing Equations

The Navier-Stokes equations for incompressible two phase flows are [12]:

$$\rho(\phi)\frac{D\boldsymbol{u}}{Dt} = -\nabla p + \nabla\cdot(2\mu(\phi)D) - \sigma\kappa(\phi)\nabla H(\phi) + \rho(\phi)g\hat{z} \tag{2.1}$$

$$\nabla\cdot\boldsymbol{u} = 0 \tag{2.2}$$

$$\frac{D\phi}{Dt} = 0 \tag{2.3}$$

ϕ is a level set function [28] which is positive in the liquid (dark fluid) and negative in the gas (light fluid). $H(\phi)$ is the Heaviside function,

$$H(\phi) = \begin{cases} 1 & \phi \geq 0 \\ 0 & \phi < 0 \end{cases} \tag{2.4}$$

$\rho(\phi)$ is the density,

$$\rho(\phi) = \rho_L H(\phi) + \rho_G(1 - H(\phi)), \tag{2.5}$$

$\mu(\phi)$ is the viscosity,

$$\mu(\phi) = \mu_L H(\phi) + \mu_G(1 - H(\phi)), \tag{2.6}$$

$\kappa(\phi)$ is the interface curvature,

$$\kappa(\phi) = \nabla\cdot\frac{\nabla\phi}{|\nabla\phi|}, \tag{2.7}$$

$\boldsymbol{u} = (u, v, w)$ is the velocity, D is the rate of deformation tensor,

$$D = \frac{\nabla\boldsymbol{u} + (\nabla\boldsymbol{u})^T}{2}, \tag{2.8}$$

g is a gravitational force constant, and $\hat{z} = (0, 0, 1)$ is a unit vector pointing in the vertical direction.

3 Overview of CLSMOF method for incompressible two-phase flow

At time $t = t^n$ we are given the cell center velocity, $\boldsymbol{u}_{i,j,k}^{n,cell}$, face center (MAC) velocity, $u_{i+1/2,j,k}^{n,MAC}$, $v_{i,j+1/2,k}^{n,MAC}$, $w_{i,j,k+1/2}^{n,MAC}$, level set function, $\phi_{i,j,k}^{n}$, volume-of-fluid function, $F_{i,j,k}^{n}$, and the location of the centroid of dark (liquid) material and light (gas) material in cell (i, j, k), $\boldsymbol{x}_{i,j,k}^{n,dark}$, $\boldsymbol{x}_{i,j,k}^{n,light}$. $\phi \geq 0$ in the liquid and $\phi < 0$ in the gas. In order to integrate the solution to time $t = t^{n+1}$ we perform the following steps:

1. Use a directionally split (Strang-splitting) algorithm in order to advect ϕ, F, $\boldsymbol{u}^{cell}$, $\boldsymbol{x}^{dark}$, $\boldsymbol{x}^{light}$. The new values are denoted by ϕ^{n+1}, F^{n+1}, $\boldsymbol{u}^{cell,advect}$, $\boldsymbol{x}^{n+1,dark}$, and $\boldsymbol{x}^{n+1,light}$. Details are in sections 4 and 5.1.
2. Calculate the effect of the viscous force terms, $\nabla\cdot(2\mu D)$, as in [33] or [38]. Also calculate the gravity force term. The new provisional cell centered velocity is $\boldsymbol{u}^*$.
3. Project the velocity $\boldsymbol{u}^*$ in order to simultaneously determine $\boldsymbol{u}_{i,j,k}^{n+1,cell}$, $u_{i+1/2,j,k}^{n+1,MAC}$, $v_{i,j+1/2,k}^{n+1,MAC}$, and $w_{i,j,k+1/2}^{n+1,MAC}$. Please refer to section 5.2 for details.

4 Directional Split CLSMOF interface reconstruction and advection

4.1 CLSMOF piecewise linear interface reconstruction

In each cell the interface is approximated by a plane in 3D (or a straight line in 2D *X-Y* or *R-Z* coordinate systems). Here, we stick to the 3D case. The 2D cases follow analogously. The interface plane is given by

$$\{\boldsymbol{x} \in \mathbb{R}^3 | \boldsymbol{n} \cdot (\boldsymbol{x} - \boldsymbol{x}_{i,j,k}) + b = 0\}, \tag{4.1}$$

where $\boldsymbol{n}$ is the normal vector, b the intercept and $\boldsymbol{x}_{i,j,k}$ the cell center. The objective of the slope reconstruction algorithm is to find a suitable $\boldsymbol{n}$ and b. The following sections describe the slope reconstruction in case of the MOF method, the CLSVOF method and the CLSMOF method.

4.1.1 MOF reconstruction

The MOF method can be seen as an extended volume of fluid method. In addition to the volume fractions of the dark fluid F_{dark} and the light fluid $F_{\text{light}} = 1 - F_{\text{dark}}$ in each computational cell the centers of mass or centroids $\boldsymbol{x}_{\text{ref,dark}}$ and $\boldsymbol{x}_{\text{ref,light}}$ are given. In each cell the linear approximation to the interface is constructed such that the actual volume of the dark fluid matches its reference volume exactly and the actual centroids $\boldsymbol{x}_A$ of both fluids are the best approximations to the reference centroids [4].

This procedure can be cast as a constrained optimization problem in which one must simultaneously find $\boldsymbol{n}$ and b in order that the volume fraction error satisfies,

$$|F_{\text{ref}}(\boldsymbol{n}, b) - F_A(\boldsymbol{n}, b)| = 0 \tag{4.2}$$

and the centroid error,

$$E_{MOF} = \|\boldsymbol{x}_{\text{ref}} - \boldsymbol{x}_A(\boldsymbol{n}, b)\|_2 \tag{4.3}$$

is minimized. The reference fluid is chosen such that the distance of the reference centroid and the cell center is maximal, i.e.

$$\text{reference fluid} = \begin{cases} \text{dark fluid} & \|\boldsymbol{x}_{\text{ref,dark}} - \boldsymbol{x}_{i,j,k}\| > \|\boldsymbol{x}_{\text{ref,light}} - \boldsymbol{x}_{i,j,k}\| \\ \text{light fluid} & \text{otherwise.} \end{cases} \tag{4.4}$$

Therefore we drop the index *dark* or *light*.

Due to the parameterization of the normal

$$\boldsymbol{n} = \begin{pmatrix} \sin(\Phi)\cos(\Theta) \\ \sin(\Phi)\sin(\Theta) \\ \cos(\Phi) \end{pmatrix} \tag{4.5}$$

E_{MOF} is a function of the angles Φ and Θ. Thus, minimizing the error E_{MOF} is a non-linear least squares problem for (Φ, Θ), i.e. find (Φ^*, Θ^*) such that

$$E_{MOF}(\Phi^*, \Theta^*) = \|\boldsymbol{f}(\Phi^*, \Theta^*)\|_2 = \min_{(\Phi,\Theta):(4.2)\text{ holds}} \|\boldsymbol{f}(\Phi, \Theta)\|_2, \tag{4.6}$$

where

$$\boldsymbol{f} : \mathbb{R}^2 \to \mathbb{R}^3, \quad \boldsymbol{f}(\Phi, \Theta) = (\boldsymbol{x}_{\text{ref}} - \boldsymbol{x}_A(\Phi, \Theta)). \tag{4.7}$$

Now we solve (4.6) for (Φ^*, Θ^*) numerically by means of the Gauss-Newton algorithm, which is basically given by the following steps.

0. choose initial angles (Φ_0, Θ_0) and initialize $k = 0$. Set $\text{tol} = 10^{-8}\Delta x$, where Δx is the grid size.

while not converged

1. find $b_k(\Phi_k, \Theta_k)$ such that (4.2) holds.
2. find the centroid $\boldsymbol{x}_k(b_k, \Phi_k, \Theta_k)$
3. find the Jacobian matrix J_k of $\boldsymbol{f}$ evaluated at (Φ_k, Θ_k). Define $\boldsymbol{f}_k \equiv \boldsymbol{f}(\Phi_k, \Theta_k)$.
4. stop if one of the following three conditions is fulfilled:
   - $\|J_k^T \cdot \boldsymbol{f}_k\| \leq \text{tol} \cdot 10^{-2}\Delta x$
   - $\|\boldsymbol{f}_k\| < \text{tol}$
   - $k = 11$

   else continue
5. solve the linear least squares problem: find $\boldsymbol{s}_k \in \mathbb{R}^2$ such that

$$\|J_k \boldsymbol{s}_k + \boldsymbol{f}_k\|_2 = \min_{\boldsymbol{s} \in \mathbb{R}^2} \|J_k \boldsymbol{s} + \boldsymbol{f}_k\|_2 \tag{4.8}$$

   by means of the normal equations.
6. update the angles: $(\Phi_{k+1}, \Theta_{k+1}) = (\Phi_k, \Theta_k) + \boldsymbol{s}_k$
7. $k := k + 1$

To find suitable initial angles (Φ_0, Θ_0) an initial normal $\boldsymbol{n}_0$ is chosen as follows. Compute the normal $\boldsymbol{n}_0^1$ using the volume fractions as defined in (4.19) and a second normal

$$\boldsymbol{n}_0^2 = \frac{\boldsymbol{x}_{ref} - \boldsymbol{x}_{i,j,k}}{\|\boldsymbol{x}_{ref} - \boldsymbol{x}_{i,j,k}\|}. \tag{4.9}$$

Define

$$\boldsymbol{n}_0 = \text{argmin}_{\{\boldsymbol{n}_0^1, \boldsymbol{n}_0^2\}} \{E_{MOF}(\boldsymbol{n}_0^1), E_{MOF}(\boldsymbol{n}_0^2)\} \tag{4.10}$$

Then,(4.5) can be used to derive the angles.

In each iteration the interface has to be reconstructed five times in step 3 to evaluate $\boldsymbol{f}_k$ and derive the Jacobian matrix. The partial derivatives $\frac{\partial \boldsymbol{f}}{\partial \Phi}$ and $\frac{\partial \boldsymbol{f}}{\partial \Theta}$ are approximated as

$$\frac{\partial \boldsymbol{f}}{\partial \Phi} \approx (\boldsymbol{f}(\Phi + h) - \boldsymbol{f}(\Phi - h))/(2h) \tag{4.11}$$

$$\frac{\partial \boldsymbol{f}}{\partial \Theta} \approx (\boldsymbol{f}(\Theta + h) - \boldsymbol{f}(\Theta - h))/(2h) \tag{4.12}$$

for an appropriate constant h. Therefore the interface for the truncated angles is needed.

Notes:

– In the original version of MOF described in [4] instead of the Gauss-Newton method an iterative Broyden-Fletcher-Goldfarb-Shanno method was used. More details on this method can be found in appendix C in [4].
– In order to calculate the Jacobian matrix, (4.11) and (4.12), we must perform four additional interface reconstructions per iteration of the Gauss-Newton method. The interface reconstruction algorithm uses Newton's method to find the intercept, given a slope and a volume fraction. The initial starting value for the Newton iteration, when calculating the Jacobian matrix, is the intercept derived from (Φ_k, Θ_k).

4.1.2 CLSVOF reconstruction

The discrete level set function $\Phi^n_{i,j,k}$ and volume fraction $F^n_{i,j,k}$ are given in each cell and located at the cell centers. In addition, the values of the neighboring cells are stored. Thus, for each cell a $3\times3\times3$ stencil of level set values and volume fractions is available. For boundary cells suitable values are derived from the prescribed boundary conditions. We need to reconstruct the interface in cells for which the stencil of level set values has both non-negative and non-positive values or the volume fraction is between 0 and 1. If the stencil of level set values has both non-negative and non-positive values, the normal vector $\boldsymbol{n}$ and the intercept b are computed as the parameters that minimize the error

$$E_{LS}(\boldsymbol{n}, b) = \sqrt{\sum_{i',j',k'=-1}^{1} w_{i',j',k'}(\Phi_{i'+i,j'+j,k'+k} - \Phi^R_{i,j,k}(\boldsymbol{x}_{i'+i,j'+j,k'+k}))^2} \tag{4.13}$$

where

$$\Phi^R_{i,j,k}(\boldsymbol{x}) \equiv \boldsymbol{n}\cdot(\boldsymbol{x} - \boldsymbol{x}_{i,j,k}) + b \tag{4.14}$$

is the piecewise linear reconstructed interface and $w_{i',j',k'}$ is a weight function:

$$w_{i',j',k'} = (\frac{1}{22})^{|i'|+|j'|+|k'|}\left(\delta_{\Delta x}(\Phi_{i+i',j+j',k+k'})\right) \tag{4.15}$$

$$\delta_{\Delta x}(\Phi) = \begin{cases} 10^{-3} & |\Phi| > 2\Delta x \\ 1+\cos(\frac{\pi\Phi}{2\Delta x}) + 10^{-3} & |\Phi| < 2\Delta x \end{cases} \tag{4.16}$$

The 1/22 factor in (4.15) is a quadrature weight since (4.13) is a discrete approximation to

$$\int_{\Omega_{i,j,k}} \delta(\Phi)(\Phi(\boldsymbol{x}) - \Phi^R_{i,j,k}(\boldsymbol{x}))^2 d\boldsymbol{x} \tag{4.17}$$

where $\Omega_{i,j,k}$ represents the rectangular cell (i,j,k).

Afterwards, the intercept b is adjusted such that the volume fraction $F_{i,j,k,\mathrm{dark}}$ is exactly reproduced by the interface plane cutting the cell into two parts.

Remark:

– If the $3 \times 3 \times 3$ stencil of level set values contains only positive or only negative values, then instead of having the CLSVOF slope defined as

$$\boldsymbol{n} = \arg\min_{\boldsymbol{n},b} E_{LS}(\boldsymbol{n}, b), \tag{4.18}$$

we estimate the CLSVOF slope $\boldsymbol{n}$ using the central differences of the volume fractions, i.e.

$$\tilde{\boldsymbol{n}} = \frac{1}{2\Delta x} \begin{pmatrix} F_{i+1,j,k} - F_{i-1,j,k} \\ F_{i,j+1,k} - F_{i,j-1,k} \\ F_{i,j,k+1} - F_{i,j,k-1} \end{pmatrix}, \quad \boldsymbol{n} = \frac{\tilde{\boldsymbol{n}}}{\|\tilde{\boldsymbol{n}}\|} \tag{4.19}$$

4.1.3 CLSMOF Reconstruction

In CLSMOF the level set method is combined with the moment of fluid method. We can predict the error in the level set reconstruction by approximating the curvature:

$$\kappa(\Phi) = \nabla \cdot \frac{\nabla \Phi}{|\nabla \Phi|}. \tag{4.20}$$

The discretization of (4.20) is given in [37] (equations 35 through 38).

For the slope reconstruction both $\kappa(\Phi)$ and E_{MOF} are used to determine whether to use the MOF slope or the CLSVOF slope.
MOF is used to determine $\boldsymbol{n}$ and b if any one of the following equations holds

1.
$$\kappa(\Phi) > \frac{1}{\beta \Delta x} \tag{4.21}$$
2. $E_{MOF} < 10^{-8}\Delta x$
3. $E_{MOF} > \Delta x^3 (1/\Delta x)^2/720 = \Delta x/720$.

Otherwise CLSVOF is used.

Remarks:

– For the first condition (4.21), if β is large, we recover the MOF method. In the results that we report in this paper, we have fixed $\beta = 6$.
– The second condition is essential; the MOF slope should always be selected if the reference centroid equals the actual centroid. It has been shown that if the centroid is known exactly, then the MOF reconstruction is optimal [2,4,1].
– The last condition is due to the influence of the curvature on the distance between the actual centroid and the reference centroid. Referring to Figure 2, suppose one has a cell with lower left corner $(-\Delta x/2, 0)$ and upper right corner $(\Delta x/2, \Delta x)$. Now suppose an interface is defined as $y = \alpha x^2 - \frac{\alpha \Delta x^2}{12} + \frac{\Delta x}{2}$. Then the volume fraction is $1/2$ and the graph is symmetric about $x = 0$ so that the actual centroid will be $(0, \Delta x/4)$. The reference centroid is $(0, \Delta x/4 + \frac{\alpha^2 \Delta x^3}{180})$. The curvature κ at $x = 0$ is 2α. So, now one can write the difference between the actual centroid and the reference centroid in terms of the curvature (for this specific configuration) as $\Delta x^3 \kappa^2/720$.

In our CLSMOF slope reconstruction algorithm, the starting value for the Gauss-Newton algorithm is either (a) the CLSVOF normal $\boldsymbol{n}_0^1$ or (b) the "centroid normal" $\boldsymbol{n}_0^2$ as in (4.9); whichever gives the smaller E_{MOF}:

$$\boldsymbol{n}_0 = \text{argmin}_{\{\boldsymbol{n}_0^1, \boldsymbol{n}_0^2\}} \{E_{MOF}(\boldsymbol{n}_0^1), E_{MOF}(\boldsymbol{n}_0^2)\}. \tag{4.22}$$

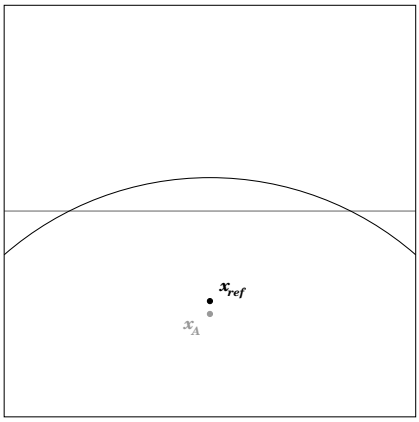


Fig. 2 The reference centroid, $\boldsymbol{x}_{ref}$, does not coincide with the exact centroid, $\boldsymbol{x}_A$, for a parabolic interface cutting the cell. The difference between the two centroids is proportional to the curvature.

4.1.4 Performance of the Gauss-Newton method for MOF and CLSMOF reconstruction

The performance of the MOF slope reconstruction and the CLSMOF slope reconstruction are compared in two test cases: (A) Reversible Single vortex in 2D (section 7.1) and (B) Reversible vortex in 3D (section 7.2).

Table 1 compares, at $t = 0$ and $t = 4$, the number of calls of the Gauss-Newton (GN) algorithm, the total number of iterations and the average number of iterations for the two methods on the basis of the Single Vortex problem (section 7.1) in 2D with a period $T = 8$. The overall CPU time in order to integrate the solution from $t = 0$ to $t = 8$ was 725 seconds for the CLSMOF method and 762 seconds for the MOF method. In order to capture the $t = 4$ statistics in Table 1, we ran the simulation up to $t = 4$ instead of $t = 8$, then restarted the computation at $t = 4$. At $t = 4$, the interface is stretched the most, so the most calls to the Gauss-Newton method will occur at this time. The grid was 128×128. A better performance of the CLSMOF method can be observed concerning the total and, thus, average number of iteration in the Gauss-Newton algorithm. This is expected since the level set function is used to help pick a good starting value for the Gauss-Newton procedure.

Remarks:

– The level set reinitialization step does not necessitate an extra reconstruction of the interface because we store the MOF slope reconstruction information that was used for reinitialization, to be used again at the beginning of the next time step. The level set reinitialization step does not modify the volume fractions or centroids, so that MOF interface reconstruction information before reinitialization does not change after reinitialization.
– We reran the 2d deformation test using MOF slope reconstruction without level set reinitialization, and the overall runtime of the MOF case consumed 713 seconds instead of 762 seconds. This test only proves that MOF is slightly more efficient than CLSMOF for passive advection of an interface. On the other hand, for non-passive advection of a deforming boundary, the reinitialization step/level set function provides an efficient way in order to derive the the departure density $\rho_{i',i}$ (5.2) and the half cell masses (5.15).

Table 2 compares, at $t = 0$ and $t = 1.5$, the number of calls of the Gauss-Newton (GN) algorithm, the total number of iterations and the average number of iterations for the two methods on the basis of the 3D reversible deforming sphere problem (section 7.2) with a period $T = 3.0$. The overall CPU time in order to integrate the solution from $t = 0$ to $t = 3.0$ was 21443 seconds for the CLSMOF method and 23114 seconds for the MOF method. The grid was $64 \times 64 \times 64$. At $t = 1.5$ the sphere experiences the most deformation. As with the 2D deformation case, the number of Gauss Newton iterations to achieve convergence is lower when the level set function is available in order to choose an initial starting slope for the Gauss-Newton procedure.

	MOF		CLSMOF	
	$t = 0$	$t = 4$	$t = 0$	$t = 4$
number of calls of GN algorithm	624	3985	625	3997
total number of iteration	2355	14074	1903	10874
average number of iterations	3.77	3.53	3.04	2.72

Table 1 2D deformation problem: MOF vs. CLSMOF

	MOF		CLSMOF	
	$t = 0$	$t = 1.5$	$t = 0$	$t = 1.5$
number of calls of GN algorithm	10894	40422	10888	40422
total number of iteration	46598	182978	38980	166411
average number of iterations	4.28	4.53	3.58	4.11

Table 2 3D deformation problem: MOF vs. CLSMOF

4.2 Directionally split CLSMOF Interface Advection

We use direction splitting, "Strang splitting" [34], in order to numerically integrate the position of a deforming interface. In each cell whose center is $\boldsymbol{x}_{i,j,k}$, the interface

reconstruction (section 4.1) gives a normal vector and an intercept. The interface within a cell is defined as the set of points, $\boldsymbol{x}$, which satisfy:

$$\phi(\boldsymbol{x}) = \boldsymbol{n} \cdot (\boldsymbol{x} - \boldsymbol{x}_{i,j,k}) + b = 0 \tag{4.23}$$

Using the interface definition (4.23), we can define a Heaviside function to differentiate light and dark regions of the fluid (4.24). Here, $\boldsymbol{n}$ is a unit normal vector pointing to the direction of the dark region.

$$H(\boldsymbol{n} \cdot (\boldsymbol{x} - \boldsymbol{x}_{i,j,k}) + b) = \begin{cases} 1, & \boldsymbol{x} \in \Omega^{dark} \\ 0, & \boldsymbol{x} \in \Omega^{light} \end{cases} \tag{4.24}$$

The following algorithm is carried out in order to integrate the position of an interface in a given x, y, or z direction. We will shorten all cell (i, j, k) subscripts to be just i in the steps (1) through (4) below.

1. In each computational cell i, the local CLSMOF slope reconstruction is found (see Section 4.1.3). The interface within the cell is defined by (4.23).
2. For a given computational cell with center $\boldsymbol{x}_i$ and volume V_i, determine the mapping from cell i's departure region $\Omega_{i,depart}$ (4.25) to cell i (Ω_i). See Figure 3. In the x-direction, we have

$$\Omega_{i,depart} = \{x_{i-\frac{1}{2}} - u_{i-\frac{1}{2}}\Delta t, x_{i+\frac{1}{2}} - u_{i+\frac{1}{2}}\Delta t\} \tag{4.25}$$

$u_{i+1/2}$ and $u_{i-1/2}$ are horizontal velocities on the MAC grid, and the MAC velocities are discretely divergence free. i.e. in 3D:

$$\frac{u_{i+1/2,j,k} - u_{i-1/2,j,k}}{\Delta x} + \frac{v_{i,j+1/2,k} - v_{i,j-1/2,k}}{\Delta y} + \frac{w_{i,j,k+1/2} - w_{i,j,k-1/2}}{\Delta z} = 0. \tag{4.26}$$

The target volume corresponding to $\Omega_{i,depart}$ is the current cell i,

$$\Omega^T_{i,depart} = \Omega_i = \{x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}\}. \tag{4.27}$$

The mapping from the departure volume to the target volume for cell i is:

$$\chi' = \alpha\chi + \beta = \frac{\Delta x(x - (x_{i-1/2} - u_{i-1/2}\Delta t))}{(x_{i+1/2} - u_{i+1/2}\Delta t) - (x_{i-1/2} - u_{i-1/2}\Delta t)} + x_{i-1/2} \tag{4.28}$$

$$\left[x_{i-1/2} - u_{i-1/2}\Delta t, x_{i+1/2} - u_{i+1/2}\Delta t\right] \rightarrow \left[x_{i-1/2}, x_{i+1/2}\right] \tag{4.29}$$

3. Determine the intersection of the departure region with surrounding cells,

$$\Omega_{i',i} = \Omega_{i'+i} \cap \Omega_{i,depart}, i' = -1, 0, 1, \tag{4.30}$$

then map $\Omega_{i',i}$ forward in order to derive $\Omega^T_{i',i}$. (see Figures 4 and 5).

4. Compute new light and dark centroids and mapped volumes/volume fractions (see Figure 4):

$$V_{i,target}^{dark} = \sum_{i'=-1}^{+1} \int_{\Omega_{i',i}^T} H(\frac{\boldsymbol{n}_{i+i'} \cdot (\boldsymbol{x}' - \boldsymbol{x}'_{i+i'})}{\alpha} + b) d\boldsymbol{x}' \tag{4.31}$$

$$V_{i,target}^{total} = \sum_{i'=-1}^{+1} \int_{\Omega_{i',i}^T} 1 d\boldsymbol{x}' = V_i \tag{4.32}$$

$$\boldsymbol{x}_i^{dark} = \frac{\sum_{i'=-1}^{+1} \int_{\Omega_{i',i}^T} \boldsymbol{x}' H \left(\frac{\boldsymbol{n}_{i+i'} \cdot (\boldsymbol{x}' - \boldsymbol{x}'_{i+i'})}{\alpha} + b \right) d\boldsymbol{x}'}{V_{i,target}^{dark}} \tag{4.33}$$

$\boldsymbol{x}_i^{light}$ follows analogously except replace H with $1 - H$. The updated volume fraction is,

$$F_i^* = \frac{V_{i,target}^{dark}}{V_i}. \tag{4.34}$$

Remarks:

– The mapping factor α can be rewritten in a more helpful fashion.

$$\alpha = \frac{1}{1 - \frac{u_{i+1/2} - u_{i-1/2}}{\Delta x} \Delta t} \sim \frac{1}{1 - u_x \Delta t} \tag{4.35}$$

This allows us to make some generalizations about the mapping and the behavior we can expect of volume fractions in the departure region.

$$\begin{cases} \alpha < 1, & \text{compression} \\ \alpha = 1, & \text{translation} \\ \alpha > 1, & \text{expansion} \end{cases}$$

– In a 2D *R-Z* coordinate system, the calculation of the departure region (4.25) is modified in the R direction to be,

$$\Omega_{i,depart} = \{r_{i-\frac{1}{2}} - \tilde{u}_{i-\frac{1}{2}} \Delta t, r_{i+\frac{1}{2}} - \tilde{u}_{i+\frac{1}{2}} \Delta t\}, \tag{4.36}$$

where $\tilde{u}_{i-1/2}$ satisfies,

$$r_{i-1/2} u_{i-1/2} \Delta t = (r_{i-1/2} - \tilde{u}_{i-1/2} \Delta t/2) \tilde{u}_{i-1/2} \Delta t. \tag{4.37}$$

$r_{i-1/2}$ is the R coordinate of the left face of cell i. In a 2D *R-Z* coordinate system, by defining the departure volume in the R direction by (4.36) and (4.37), we guarantee that the corresponding advection method is free stream preserving.

Steps (1) through (4) above are repeated in the x, y, and z directions. F^* in (4.34) is a provisional volume fraction. In order to preserve mass, we must keep track of the total amount of "dark mass" that enters a cell and use that to determine the updated volume fraction. We define $F^{*,(0)} \equiv F^n$, $F^{*,(1)}$ to be the provisional volume fraction after the first sweep, $F^{*,(2)}$ to be the provisional volume fraction after the second sweep, and $F^{*,(3)}$ to be the provisional volume fraction after the last sweep.

The overall algorithm, sweeping in all the directions is as follows:

- Initialize

$$\boldsymbol{M}_i^{dark} = \rho^{dark} F_i^n V_{i,cell} \tag{4.38}$$

- For d = 1, 2, 3 do
  - Reconstruct the interface slope and intercept (4.23).
  - Find the mapping,

$$\chi' = \alpha\chi + \beta, \tag{4.39}$$

    from departure volume χ to target volume χ' (4.28).
  - Identify the intersections of the departure volume with the neighboring cells; advect each intersected region, $\Omega_{i',i}$ forward
  - Compute volume contributions and moments, $V_{i,target}^{dark}$, $\boldsymbol{x}_i^{dark}$, $\boldsymbol{x}_i^{light}$ (see (4.31), (4.33)).
  - Compute the provisional volume fraction $F_i^{*,(d)}$ (4.34).
  - increment the overall dark fluid volume (mass) that enters cell i:

$$\boldsymbol{M}_i^{dark} = \boldsymbol{M}_i^{dark} + \frac{1}{\alpha}\rho^{dark} V_{i,target}^{dark} - \rho^{dark} F^{*,(d-1)} V_{i,cell} \tag{4.40}$$

- enddo
- Mask volume fraction:

$$F_i^{n+1} = \begin{cases} 0 & F_i^{*,(3)} < \epsilon \\ \frac{\boldsymbol{M}^{dark}}{\rho^{dark} V_i} & \epsilon < F_i^{*,(3)} < 1-\epsilon \\ 1 & F_i^{*,(3)} > 1-\epsilon \end{cases} \tag{4.41}$$

We integrate the solution of the level set function ϕ concurrently with the integration of the solution to the volume fractions and centroids. The following steps describe how to integrate the solution of the level set function in a given x, y, or z direction:

1. Use Van-Leer slope limiting [24] in order to locally reconstruct the level set function profile in a given cell i:

$$\phi(x) = \phi_i + \phi'_{i,VL}(x - x_i) \tag{4.42}$$

where

$$\phi'_{i,VL} = \begin{cases} 0, & \left(\frac{\phi_{i+1,j,k}-\phi_{i,j,k}}{\Delta x}\right)\left(\frac{\phi_{i,j,k}-\phi_{i-1,j,k}}{\Delta x}\right) < 0 \\ f^{VL}\left(\frac{\phi_{i+1,j,k}-\phi_{i,j}}{\Delta x}, \frac{\phi_{i,j,k}-\phi_{i-1,j,k}}{\Delta x}\right), & \text{otherwise} \end{cases} \tag{4.43}$$

$$f^{VL}(x,y) = \frac{2xy}{x+y} \tag{4.44}$$

The Van-Leer slope, $\phi'_{i,VL}$, guarantees that the reconstructed profile for ϕ is bounded by the values of ϕ at the cell centers.

2. Find the mapping,

$$\chi' = \alpha\chi + \beta, \tag{4.45}$$

from departure volume χ to target volume χ' (4.28).

3. Find the intersection of departure region $\Omega_{i,depart}$ for cell Ω_i with the cells Ω_i, Ω_{i-1}, and Ω_{i+1} as in (4.46). See Fig. 5.

$$\Omega_{i',i} = \Omega_{i'+i} \cap \Omega_{i,depart}, i' = -1, 0, 1 \tag{4.46}$$

4. Compute the updated level set function ϕ_i^*:

$$\phi_i^* = \frac{\sum_{i'=-1}^{+1} \int_{\Omega_{i',i}} \left(\phi_{i+i'} + \phi'_{i+i',VL}(x - x_{i+i'})\right) d\boldsymbol{x}}{\sum_{i'=-1}^{+1} |\Omega_{i',i}|} \tag{4.47}$$

After sweeping in the x, y, and z directions, the level set function is coupled to the volume-of-fluid function, F^{n+1}, and the moment of fluid function $\boldsymbol{x}^{n+1}$ by replacing the current level set function ϕ^{n+1} with the exact signed distance to the piecewise linear CLSMOF reconstructed interface.

The algorithm for finding the signed distance to the CLSMOF reconstructed interface is the same as that described in [39] except that we do not require the level set function to change sign in an "interface" cell. Instead we only require that $F - 1/2$, where F is the volume of fluid function, changes sign in a 3×3 stencil about the cell in question.

The following algorithm finds the signed normal distance in a band K cells wide about the CLSMOF reconstructed interface.

1. Tag all computational cells (i,j,k)
2. For each computational cell (i,j,k) check to see if

$$\left(F_{i,j,k}^{n+1} - \frac{1}{2}\right)\left(F_{i^*,j^*,k^*}^{n+1} - \frac{1}{2}\right) < 0 \tag{4.48}$$

for some cell $(i^*, j^*, k^*) \equiv (i+i', j+j', k+k')$ such that $|i'| \leq 1$, $|j'| \leq 1$, $|k'| \leq 1$. If there is a (i^*, j^*, k^*) that satisfies (4.48), then perform the following steps:

(a) If

$$0 < F_{i,j,k}^{n+1} < 1 \tag{4.49}$$

find the linear reconstruction

$$\phi_{i,j,k}^{n+1,R}(\boldsymbol{x}) = \boldsymbol{n}_{i,j,k} \cdot (\boldsymbol{x} - \boldsymbol{x}_{i,j,k}) + b_{i,j,k} \tag{4.50}$$

If both (4.49) and (4.48) are satisfied then mark the face centroids and corners of cell (i, j, k) with the sign according to the evaluation of $\phi_{i,j,k}^{n+1,R}(\boldsymbol{x})$ at the cell corners and face centroids. If either (4.49) or (4.48) is not satisfied, then mark the corners and face centroids with the sign of $F_{i,j,k} - 1/2$.

(b) For each cell $(i^*, j^*, k^*) \equiv (i + i', j + j', k + k')$ in the K-band such that $|i'| \leq K$, $|j'| \leq K$, and $|k'| \leq K$ carry out the following steps.

(i) Determine the shortest distance d associated with the cells $\boldsymbol{x}_{i,j,k}$ and $\boldsymbol{x}_{i^*,j^*,k^*}$. One must consider the orthogonal projection of $\boldsymbol{x}_{i^*,j^*,k^*}$ onto the interface in cell (i, j, k) (4.23), if it lies in the cell. If the projection lies outside of the cell, then the intersection of the interface with the cell faces must be considered. Also, one must test corners and cell face centroids. See Fig. 6. For any corner or centroid whose sign is opposite that of (i^*, j^*, k^*), compare the distance from (i^*, j^*, k^*) to the corner/centroid with the current value of d. Take the minimum value.

(ii) Update $\phi^{n+1}_{i^*,j^*,k^*}$ using d:

$$\phi^{n+1}_{i^*,j^*,k^*} = \begin{cases} \mathrm{SGN}(F^{n+1}_{i^*,j^*,k^*} - \frac{1}{2})d & \text{if } d < |\phi^{n+1}_{i^*,j^*,k^*}| \text{ or} \\ & (i^*, j^*, k^*) \text{ is tagged} \\ \phi^{n+1}_{i^*,j^*,k^*} & \text{otherwise} \end{cases} \tag{4.51}$$

(iii) Untag cell (i^*, j^*, k^*).

3. For cells (i, j, k) which are still tagged, we have

$$\phi^{n+1}_{i,j,k} = \begin{cases} \phi_{max}, & F_{i,j,k} \geq 1/2 \\ \phi_{min}, & F_{i,j,k} < 1/2 \end{cases} \tag{4.52}$$

Where ϕ_{max} and ϕ_{min} are the maximum and minimum level set values in the untagged cells.

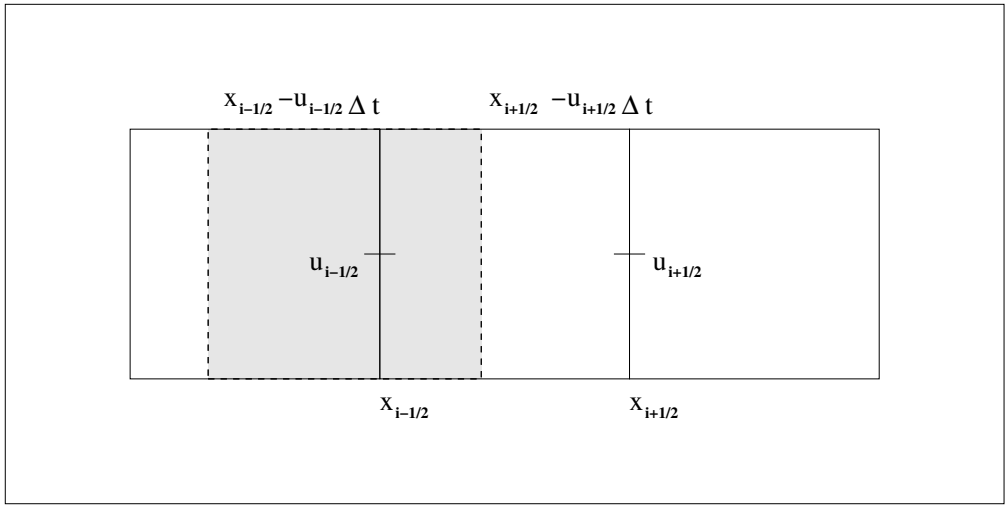


Fig. 3 The intersection of the departure volume with the adjoining cells is found.

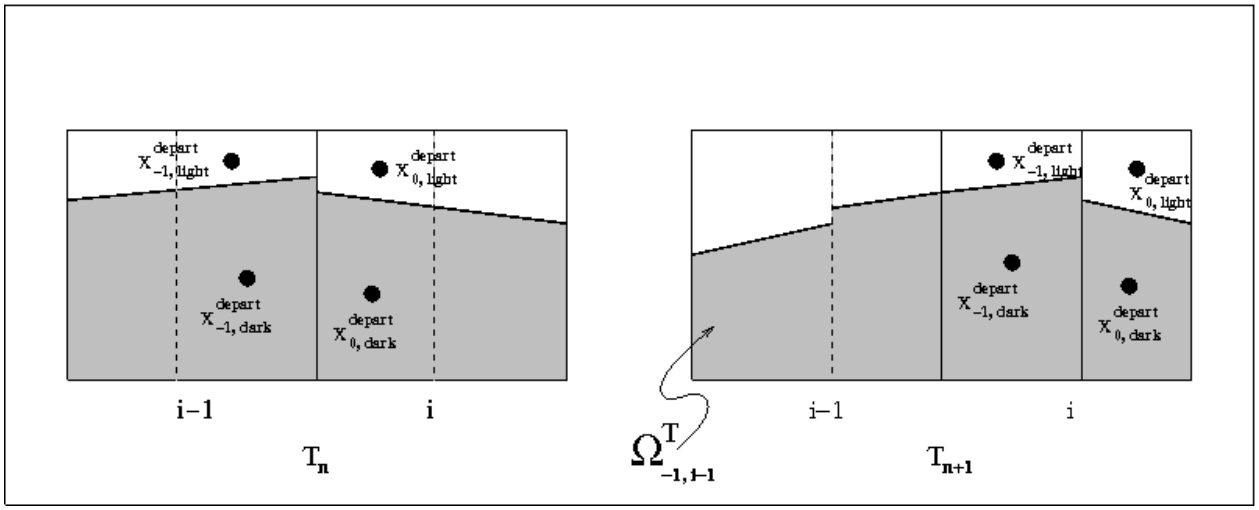


Fig. 4 The intersection of the departure volume (outlined by dashed lines) with the adjoining cells is found and denoted by $\Omega_{i',i}$. The intersected departure regions are mapped forward (see right figure) and then the volume fractions and centroids are calculated in cell i. In this figure, while there can be a contribution from up to three cells, here there is no contribution from the $i' = +1$ cell. The region marked $\Omega^T_{-1,i-1}$ denotes the mapped intersection of the departure region for cell $i-1$ with cell $i-2$

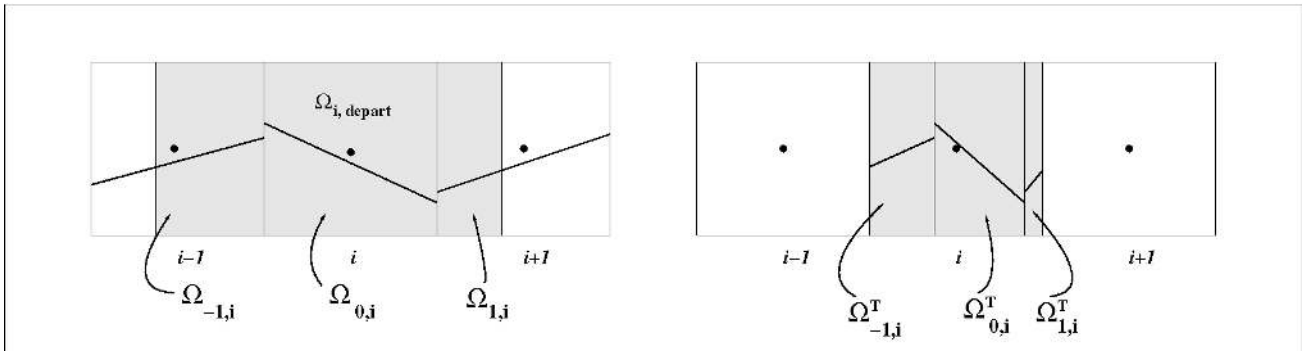


Fig. 5 The shaded departure volume $\Omega_{i,depart}$ is intersected with the cells, then mapped forward into $\Omega^T_{0,i}$, $\Omega^T_{-1,i}$, and $\Omega^T_{1,i}$ in the target cell Ω^T.

5 Fluid Algorithm

5.1 Momentum Advection

We follow the ideas described in [31] for discretizing the non-linear advection terms for multiphase flows. Concurrently with directionally split CLSMOF advection (Section 4.2), we discretize the non-linear advective terms as follows.

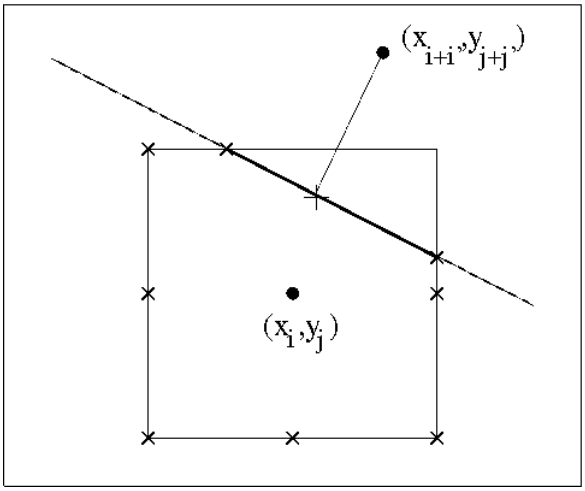


Fig. 6 When finding the nearest point in cell (i, j) to point $(i+i', j+j')$, one needs to test the points marked with an 'x': face centroids, corners, the orthogonal projection onto the interface, and the intersection of the interface with the cell faces.

1. In each cell, the velocity has the reconstruction as in (5.1), where $u'_{i,VL}$ is determined from Van Leer slope limiting. In addition, if $\phi_i \phi_{i\pm1} \leq 0$, or if $0 < F_i < 1$, then the slope $u'_{i,VL}$ is set to zero.

$$u = u_i + u'_{i,VL}(x - x_i) \tag{5.1}$$

2. For a given cell Ω_i, find the intersection of the departure region for that cell, $\Omega_{i,depart}$, with all surrounding cells, as in Figure 3.
3. Denote the intersection of the departure region of cell i with neighboring cell $i + i'$ as $\Omega_{i',i}$. See Figure 5.
4. Find the $1D$ fraction of dark material in $\Omega_{i',i}$ using linear interpolation of the levelset function (see Fig. 7), then find the density in each $\Omega_{i',i}$. Call this $\rho_{i',i}$, the departure density.
5. Compute the new velocity as a result of advection,

$$u^{n+1,advect} = \frac{\sum_{i'=-1}^{+1} \int_{\Omega_{i',i}} \rho_{i',i}(u_{i+i'} + u'_{i+i',VL}(x - x_{i+i'}))d\boldsymbol{x}}{\sum_{i'=-1}^{+1} |\Omega_{i',i}| \rho_{i',i}} \tag{5.2}$$

Note: The mass in the denominator of (5.2) will unlikely be the mass derived from the updated level set function ϕ_i^* (4.47), therefore momentum is not discretely conserved by our method.

5.2 Pressure Projection

We approximate solutions to the following two equations, (5.3) and (5.4), in order to derive the updated velocity $\boldsymbol{u}$. In our implementation, we maintain both a MAC velocity and a cell centered velocity. The MAC velocity is used to determine

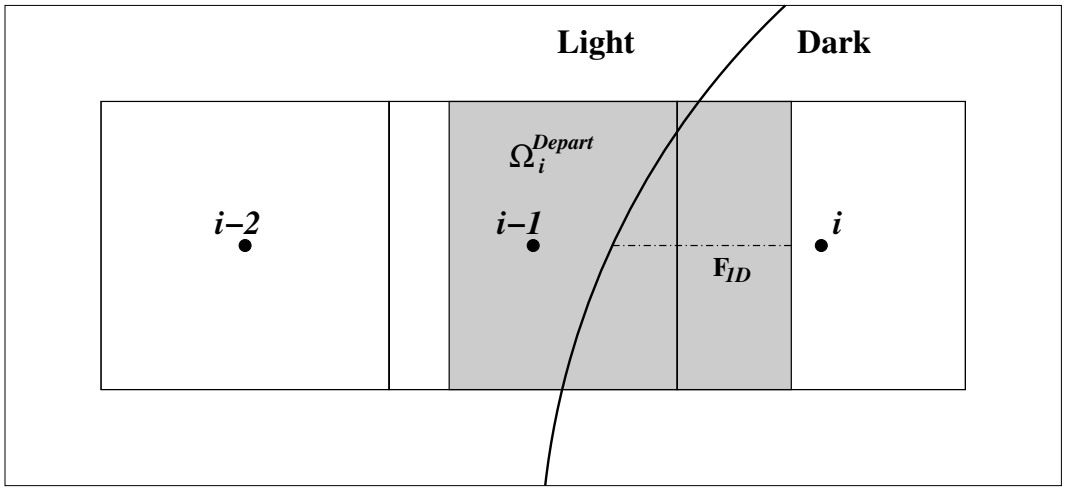


Fig. 7 The departure region, shaded, is cut by the interface. The one-dimensional dark volume fraction F_{1D} (shown as a dotted line) in the departure cell is found by linear interpolation of the level set function. The root of the linear interpolant of ϕ is the location of the $1D$ reconstructed interface.

the departure regions (4.25) for transporting the interface and velocity field. The MAC velocity will be discretely divergence free and the cell centered velocity will be "approximately" divergence free.

$$\nabla \cdot \frac{\nabla p}{\rho} = \nabla \cdot \boldsymbol{u}^* \tag{5.3}$$

$$\boldsymbol{u} = \boldsymbol{u}^* - \frac{\nabla p}{\rho} \tag{5.4}$$

The labeling convention, in defining ρ and the surface tension force, is that 'dark' refers to liquid regions and 'light' refers to gas regions. A level set function $\phi(\boldsymbol{x})$ is defined such that

$$\phi(\boldsymbol{x}) \begin{cases} \geq 0, & \boldsymbol{x} \in \Omega_{dark} \\ < 0, & \boldsymbol{x} \in \Omega_{light} \end{cases}$$

The MAC pressure gradient is discretized in the x-direction as in (5.5).

$$\frac{\partial p/\partial x}{\rho} = \frac{p_{i+1,j} - p_{i,j}}{\Delta x} \cdot \frac{1}{\rho_{i+\frac{1}{2},j}} \tag{5.5}$$

The MAC density $\rho_{i+\frac{1}{2},j}$ is derived from the level-set function ϕ according to (5.6 - 5.11). See Figure 9.

$$\phi_j^+ = \max(\phi_j, 0) \tag{5.6}$$

$$\phi_{i+1/2} = \frac{\phi_i + \phi_{i+1}}{2} \tag{5.7}$$

$$F_{1D,i}^{Right} = \begin{cases} 1, & \phi_i \geq 0, \phi_{i+1/2} \geq 0 \\ 0, & \phi_i < 0, \phi_{i+1/2} < 0 \\ \dfrac{\phi_i^+ + \phi_{i+1/2}^+}{|\phi_i| + |\phi_{i+1/2}|}, & otherwise \end{cases} \tag{5.8}$$

$$F_{1D,i}^{Left} = \begin{cases} 1, & \phi_i \geq 0, \phi_{i-1/2} \geq 0 \\ 0, & \phi_i < 0, \phi_{i-1/2} < 0 \\ \dfrac{\phi_i^+ + \phi_{i-1/2}^+}{|\phi_i| + |\phi_{i-1/2}|}, & otherwise \end{cases} \tag{5.9}$$

$$F_{1D,i+1/2} = \frac{F_{1D,i}^{Right} + F_{1D,i+1}^{Left}}{2} \tag{5.10}$$

$$\rho_{i+\frac{1}{2},j} = \rho_{Dark} F_{1D,i+1/2} + \rho_{Light}(1 - F_{1D,i+1/2}) \tag{5.11}$$

The right-hand side of equation (5.3) is discretized as

$$\nabla \cdot \boldsymbol{u}^* \sim \frac{u_{i+1/2,j}^* - u_{i-1/2,j}^*}{\Delta x} + \frac{v_{i,j+1/2}^* - v_{i,j-1/2}^*}{\Delta y} + \frac{w_{i,j,k+1/2}^* - w_{i,j,k-1/2}^*}{\Delta z}$$

where

$$u_{i+\frac{1}{2}}^* = \frac{M_{i,R} u_i + M_{i+1,L} u_{i+1}}{M_{i,R} + M_{i+1,L}} \tag{5.12}$$

The half cell masses $M_{i,R}$ and $M_{i+1,L}$ are defined by (5.13 - 5.15).

$$\rho_{i,Left} = F_{1D,i}^{Left} \rho_{Dark} + (1 - F_{1D,i}^{Left}) \rho_{Light} \tag{5.13}$$

$$\rho_{i,Right} = F_{1D,i}^{Right} \rho_{Dark} + (1 - F_{1D,i}^{Right}) \rho_{Light} \tag{5.14}$$

$$M_{i,Left} = \rho_{i,Left} \frac{\Delta x}{2} \qquad M_{i,Right} = \rho_{i,Right} \frac{\Delta x}{2} \tag{5.15}$$

After solving (5.3), the discretely divergence-free MAC velocity (used for computing the departure volume) is defined in Eqn. (5.16 - 5.18) following [21,40]:

$$u_{i+1/2,j,k} = u_{i+1/2,j,k}^* - \frac{p_{i+1,j,k} - p_{i,j,k}}{\rho_{i+1/2,j,k} \Delta x} \tag{5.16}$$

$$v_{i,j+1/2,k} = v_{i,j+1/2,k}^* - \frac{p_{i,j+1,k} - p_{i,j,k}}{\rho_{i,j+1/2,k} \Delta y} \tag{5.17}$$

$$w_{i,j,k+1/2} = w_{i,j,k+1/2}^* - \frac{p_{i,j,k+1} - p_{i,j,k}}{\rho_{i,j,k+1/2} \Delta z} \tag{5.18}$$

The "approximately" divergence free cell centered velocity is defined in the following equations (5.19-5.21) following [22]:

$$u_{i,j,k} = u_{i,j,k}^* - \frac{p_{i+1/2,j,k} - p_{i-1/2,j,k}}{M_{i,Left} + M_{i,Right}} \tag{5.19}$$

$$v_{i,j,k} = v^*_{i,j,k} - \frac{p_{i,j+1/2,k} - p_{i,j-1/2,k}}{M_{j,Back} + M_{j,Front}} \tag{5.20}$$

$$w_{i,j,k} = w^*_{i,j,k} - \frac{p_{i,j,k+1/2} - p_{i,j,k-1/2}}{M_{k,Bot} + M_{k,Top}} \tag{5.21}$$

We follow the work of [22] in interpolating the pressure from the cell centers, $p_{i,j,k}$, to the cell faces, $p_{i+1/2,j,k}$, $p_{i,j+1/2,k}$, and $p_{i,j,k+1/2}$. As in [22], we enforce that $\boldsymbol{u} - \boldsymbol{u}^*$ is continuous across cell faces. In otherwords, if one has,

$$u_{i,R} = u^*_{i,R} - \frac{p_{i+1/2} - p_i}{M_{i,Right}} \qquad u_{i+1,L} = u^*_{i+1,L} - \frac{p_{i+1} - p_{i+1/2}}{M_{i+1,Left}}, \tag{5.22}$$

then we enforce that the sides of a cell face remain in constant contact to arrive at (5.23):

$$\frac{p_{i+\frac{1}{2}} - p_i}{M_{i,Right}} = \frac{p_{i+1} - p_{i+\frac{1}{2}}}{M_{i+1,Left}} \tag{5.23}$$

Equation (5.23) is solved for pressure at the face:

$$p_{i+1/2} = \frac{\frac{p_{i+1}}{M_{i+1,Left}} + \frac{p_i}{M_{i,Right}}}{\frac{1}{M_{i+1,Left}} + \frac{1}{M_{i,Right}}} \tag{5.24}$$

Once this process has been carried out to find the pressure on all of the faces, we obtain an expression for the pressure gradient at the center of the cell. For example,

$$u_i = u^*_i - \frac{p_{i+\frac{1}{2}} - p_{i-\frac{1}{2}}}{M_{i,Left} + M_{i,Right}} \tag{5.25}$$

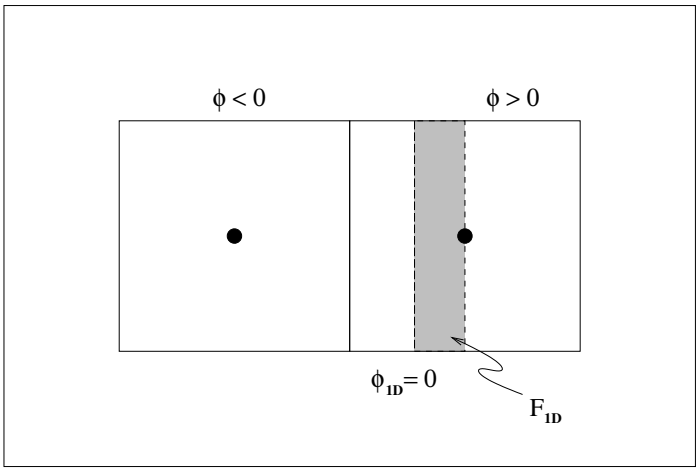


Fig. 8 Given a change in sign of the level set function across adjoining cells, we can use a linear interpolation in 1D to reconstruct the boundary and volume fraction F_{1D} in the face control volume.

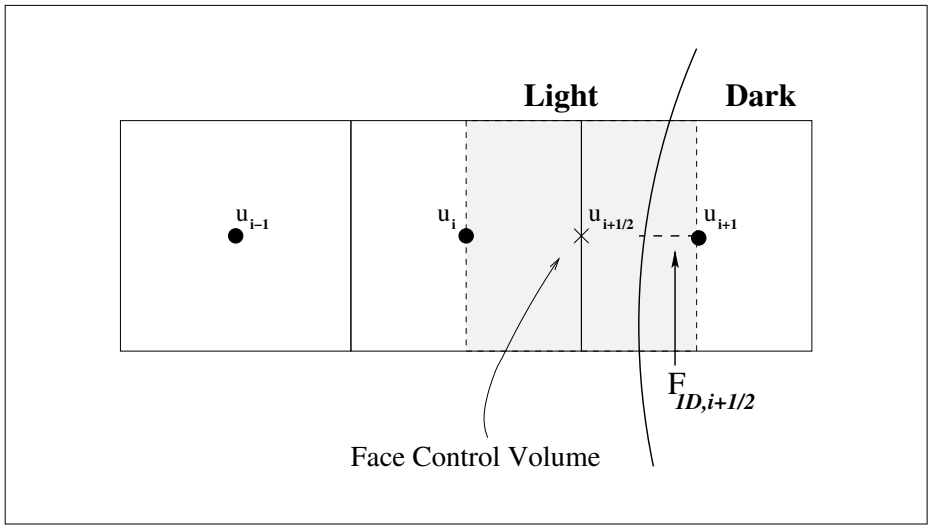


Fig. 9 A face control volume can be defined between two grid cells such that the center is the face centroid of the common side of the grid cells. $F_{1D,i+1/2}$ is the fraction of the face control volume occupied by the dark fluid, as found by the $1D$ linear interpolation of the level set function.

5.3 Surface Tension

We use the ghost fluid method [21] for discretizing the surface tension force. Curvature κ is estimated by using the height function technique as in [13], [35], or [20].

Suppose the i^{th} cell is a dark cell. We use the pressure jump across the interface to rewrite the 'light' pressure in terms of the 'dark' pressure using the Ghost Fluid Method [17, 21]. See Figure 10. For example, (5.25) is recast as follows if a "light" cell is nearby:

$$u_i = u_i^* - \frac{p_{i+1/2}^{Dark} - p_{i-1/2}^{Dark}}{M_{i,Left} + M_{i,Right}} \tag{5.26}$$

Referring to Figure 10, since both p_i and p_{i+1} are liquid pressures, $p_{i+1/2}^{Dark} = p_{i+1/2}$, where $p_{i+1/2}$ is defined by (5.24). Since p_{i-1} is a gas pressure, we have:

$$p_{i-1/2}^{Dark} = \frac{\frac{p_i}{M_{i,Left}} + \frac{p_{i-1} - \sigma\kappa_{i-1/2}}{M_{i-1,Right}}}{\frac{1}{M_{i,Left}} + \frac{1}{M_{i-1,Right}}} \tag{5.27}$$

6 Dynamic Block Structured Adaptive Mesh Refinement

For all of our test computations, the underlying computational domain is established using dynamic block structured adaptive mesh refinement [37, 36] (AMR). A computational domain that is organized using AMR is made up of a hierarchy

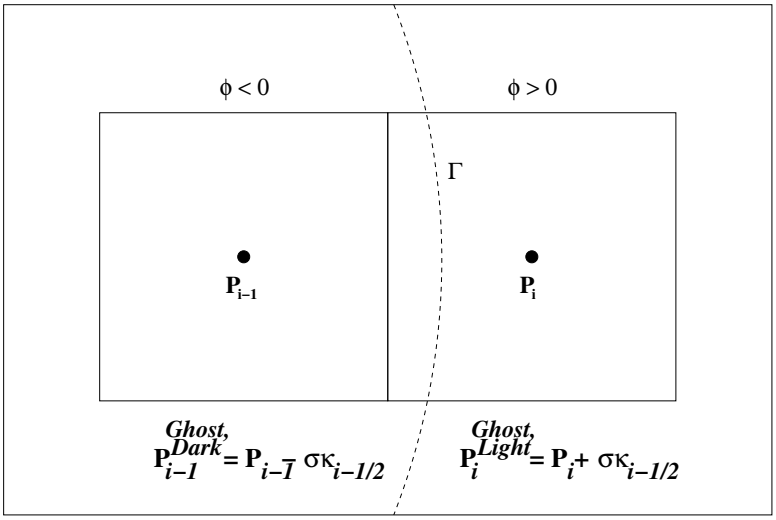


Fig. 10 The i^{th} cell rests in the dark region, and the interface cuts in between the i and $i-1$ cells. The pressure gradient is computed as in (5.26) in terms of the ghost dark pressure and accounting for the jump condition across interface Γ.

of adaptive levels $\ell = 0, \ldots, \ell^{max}$ with each level being the union of disjoint rectangular grids. Level $\ell = 0$ is the coarsest level and the mesh size on each finer level is half the mesh size of the preceding level, $\Delta x^{\ell+1} = \Delta x^{\ell}/2$.

In [37,36], the boundary conditions for the level set function ϕ at the boundary between a coarse level ℓ and a fine level $\ell+1$ was treated using linear interpolation. In other words, conservative Van-Leer slope limited linear interpolation on the coarse level ℓ was used to initialize "ghost" fine level $\ell+1$ cells and the average of the level set values on fine grid cells that cover a given coarse grid cell, $\phi^{\ell+1}$, was used to overwrite the coarse level set function, ϕ^{ℓ}.

In this work, we use the same grid transfer operations for the level set function as in [37], and in addition, we have developed routines for transferring volume fraction and centroid information from the coarse grid to the fine grid and vice versa. Referring to Figure 11, in order to interpolate the volume fraction and centroid information from the underlying coarse grid (open circles) to the ghost fine grid points (green cells), we first find the piecewise linear MOF reconstructed interface in the coarse cells below the "ghost fine cells." Then the ghost fine grid volume fraction and centroid are derived from the coarse MOF reconstructed interface. We synchronize the coarse level volume fraction and centroid information with the fine level volume fraction and centroid information by overwriting the coarse grid volume fraction (open circles in blue region) with the average of the overlying fine grid volume fraction values and by overwriting the coarse grid centroid with the fine grid volume fraction weighted average of the overlying fine grid centroids.

The criteria for adaptivity on a given level ℓ is to tag a grid cell $\boldsymbol{x}^{\ell}_{i,j,k}$ for adaptivity if the level set function $\phi^{\ell}_{i,j,k}$ changes sign and if the radius of curvature satisfies $\kappa^{\ell}_{i,j,k} > 1/R^{\ell,cut}$.

After every user defined number of time steps, the adaptive mesh is reconstructed as follows:

- A coarse level, $\ell = 0$, is established which covers the whole computational domain.
- Coarse level cells $\boldsymbol{x}^{\ell}_{i,j,k}$ are tagged based on user defined criteria.
- Cells within an "errorbuf" radius of a tagged cell are also tagged.
- Cells tagged on level ℓ are grouped together according to the Berger and Rigoustsos [9] clustering algorithm in order to define a new adaptive level $\ell + 1$.
- The process is repeated until level ℓ^{max} is established.

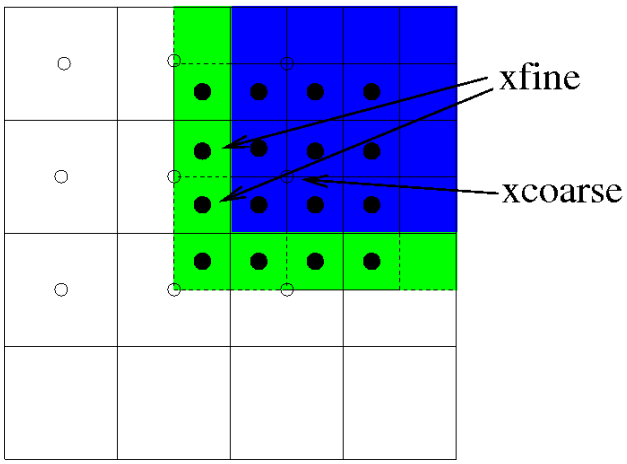


Fig. 11 For the levelset function, the "ghost points" on a fine level (green region) are initialized using conservative, Van-Leer limited, linear interpolation from the coarser level. For example, the points labeled as "xfine" are initialized from the level set values defined at the open circles. Coarse grid level set values located at points that are covered by a finer level (blue region) are synchronized to the fine level by overwriting coarse grid level set values with the average of overlying fine grid level set values. In order to initialize ghost volume fractions and centroids in the green cells, we first find the piecewise linear MOF reconstructed interface on the underlying coarse level cells, then derive the fine grid volume fractions and centroids from the reconstructed interface.

7 Numerical Results

In the results that follow, where an exact solution is available, we compare the computed solution to the exact solution using the symmetric difference error. We define Ω_C and Ω_E to be the computed and exact regions respectively of 'dark fluid' in the domain. The symmetric difference error is then,

$$E_{sym} = |\Omega_C \cup \Omega_E - \Omega_C \cap \Omega_E|. \tag{7.1}$$

(7.1) can also be defined as,

$$E_{sym} = \sum_{i,j,k} \int_{\Omega_{i,j,k}} |H(\boldsymbol{n} \cdot (\boldsymbol{x} - \boldsymbol{x}_{i,j,k}) + b) - H(\phi_E(\boldsymbol{x}))| d\boldsymbol{x}. \tag{7.2}$$

H is the Heaviside function (2.4), $\boldsymbol{n}$ and b are derived from the interface reconstruction (4.1), $\boldsymbol{x}_{i,j,k}$ is the center of the cell $\Omega_{i,j,k}$, and the zero level set of $\phi_E(\boldsymbol{x})$ is the exact interface location.

We compute the symmetric difference error by approximating the integral in (7.2) using adaptive quadrature.

The time step Δt is the minimum of the following quantities:

$$\Delta t_{advect} = \text{cfl}\frac{\Delta x}{\max|u|} \tag{7.3}$$

$$\Delta t_{tension} = \text{cfl}\Delta x\sqrt{\frac{\Delta x\rho^{dark}}{8\pi\sigma}} \tag{7.4}$$

$$\Delta t_{gravity} = \text{cfl}\sqrt{2\Delta x\pi/g} \tag{7.5}$$

We fix "cfl" always to be 1/2 in our simulations.

In the examples that follow, we will often compare the following four different slope reconstruction techniques:

- CLSVOF slope reconstruction (section 4.1.2)
- MOF slope reconstruction (section 4.1.1)
- CLSMOF slope reconstruction (section 4.1.3)
- LVIRA slope reconstruction (see [29])

In order to emphasize that we use directional splitting [34] for interface and momentum advection, we refer to each of the four different techniques as DS-CLSVOF, DS-MOF, DS-CLSMOF, and DS-LVIRA, respectively.

7.1 Single Vortex

In this test, a circle of radius $R_0 = 0.15$ and center $(0.5, 0.75)$ is placed inside a unit sized box. The velocity field is given by the stream function [8]:

$$\Psi(x,y,t) = \frac{1}{\pi}\sin^2(\pi x)\sin^2(\pi y)\cos(\frac{\pi t}{T}), \tag{7.6}$$

where T is the period of a reversing vortical flow.

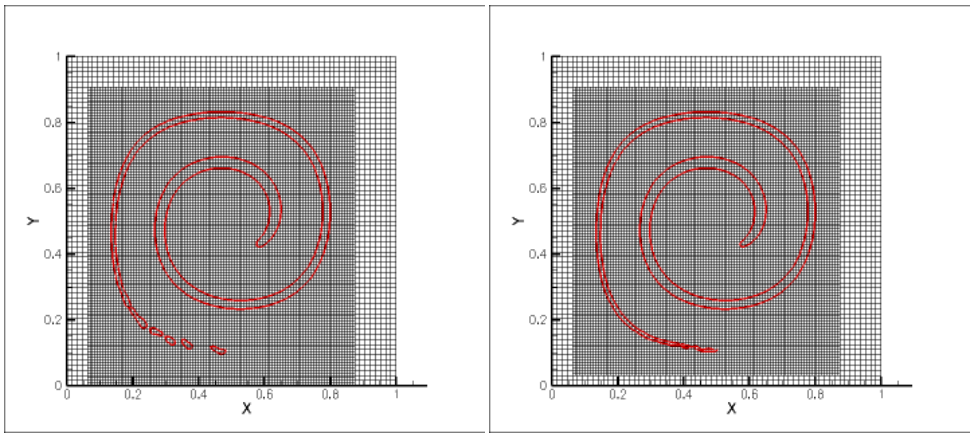
The resulting velocity field first stretches the circle into an ever thinner filament that wraps around the center of the box, then after time $t = T/2$, slowly reverses and pulls the filament back into the initial circular shape at time $t = T$.

For this test we prescribe the velocity at the cell faces in terms of finite differences of the exact stream function Ψ (7.6). This is important in order to insure that the face velocity is discretely divergence free.

The velocity depends on time, so we must reverse the order of operations within each time step in order for the directionally split method to be formally second order accurate. So for each time step, the velocity is specified at $t^{n+1/2}$ and we sweep in the x, y, y, x directions respectively, each with a time step of $\Delta t/2$.

In table 3, we compare the error at $t = T$, E_{sym} (7.1), using the 4 different slope reconstruction techniques: DS-CLSVOF, DS-MOF, DS-CLSMOF, and DS-LVIRA.

The results using the DS-CLSMOF method, for $T = 8$ and an effective fine grid resolution of 128x128 is displayed in Figure 12 at time $t = 4.0$. Results are compared with the DS-CLSVOF method.



(a) DS-CLSVOF (b) DS-CLSMOF

Fig. 12 Deformation problem with period $T = 8$. Results at $t = 4.0$. The corresponding piecewise linear reconstructed interface is plotted. Base grid 64x64. one level of AMR; effective fine grid resolution 128x128.

Size	T	DS-CLSMOF	DS-MOF	DS-CLSVOF	DS-LVIRA
64	1/2	$1.15E-4$	$1.43E-4$	$1.11E-4$	$1.33E-4$
128	1/2	$3.22E-5$ (1.8)	$4.14E-5$ (1.8)	$3.30E-5$	$3.46E-5$
256	1/2	$9.22E-6$ (1.8)	$1.19E-5$ (1.8)	$9.28E-6$	$9.02E-6$
32	8	$2.92E-2$	$3.45E-2$	$5.45E-2$	$6.61E-2$
64	8	$5.51E-3$ (2.4)	$1.00E-2$ (1.8)	$1.05E-2$	$1.08E-2$
128	8	$1.37E-3$ (2.0)	$1.11E-3$ (3.2)	$1.74E-3$	$1.42E-3$

Table 3 Symmetric Difference Error from Section 7.1, for the reversible $2D$ vortex. Errors are taken at the end time $t = T$. The order of accuracy for the DS-CLSMOF and DS-MOF methods are reported in parenthesis.

7.2 Reversible Vortex - 3D

In this test problem, a sphere with radius 0.15 and center $(0.35, 0.35, 0.35)$ is placed in the following flow field:

$$u = 2\cos(\pi t/3)\sin^2(\pi x)\sin(2\pi y)\sin(2\pi z) \tag{7.7}$$

$$v = -\cos(\pi t/3)\sin^2(\pi y)\sin(2\pi x)\sin(2\pi z) \tag{7.8}$$

$$w = -\cos(\pi t/3)\sin^2(\pi z)\sin(2\pi x)\sin(2\pi y) \tag{7.9}$$

The initial sphere undergoes severe deformation for $0 < t < 3/2$. For $3/2 < t < 3$, the flow is "reversed" and the final expected shape is a sphere again.

As in section 7.1, we prescribe the velocity at the cell faces in terms of finite differences of the following function:

$$f(x, y, z) = \frac{1}{\pi^2}\sin^2(\pi x)\sin^2(\pi y)\sin^2(\pi z) \tag{7.10}$$

$$u = 2\cos(\pi t/3)\frac{\partial^2 f}{\partial y \partial z} \tag{7.11}$$

$$v = -\cos(\pi t/3)\frac{\partial^2 f}{\partial x \partial z} \tag{7.12}$$

$$w = -\cos(\pi t/3)\frac{\partial^2 f}{\partial x \partial y} \tag{7.13}$$

The velocity depends on time, so we must reverse the order of operations within each time step in order for the directionally split method to be formally second order accurate. So for each time step, the velocity is specified at $t^{n+1/2}$ and we sweep in the x, y, z, z,y, x directions respectively, each with a time step of $\Delta t/2$.

The results on a $64 \times 64 \times 64$ grid using the DS-CLSMOF method versus the DS-CLSVOF method are compared in Figure 13 at time $t = 1.5$ and compared in Figure 14 at time $t = 3.0$. We note that while the difference in the symmetric-difference error of the two methods is small, i.e. the volume of material that is "displaced" for the DS-CLSVOF method is not that much smaller than for the DS-CLSMOF method at $t = 3.0$, the *surface area* of displaced material for the DS-CLSVOF method is considerably larger than for the DS-CLSMOF method.

The symmetric difference error for the reversible 3D flow are shown in Table 4.

Size	DS-CLSMOF	DS-MOF	DS-CLSVOF	DS-LVIRA
32	$4.81E-3$	$5.72E-3$	$6.92E-3$	$7.08E-3$
64	$1.99E-3$ (1.3)	$2.02E-3$ (1.5)	$2.13E-3$	$3.01E-3$

Table 4 Symmetric Difference Error from Section 7.2, for the reversible $3D$ vortex. Errors are taken at the end time $t = 3$. The order of accuracy for the DS-CLSMOF and DS-MOF methods are reported in parenthesis.

7.3 Notched Disk: rigid body motion

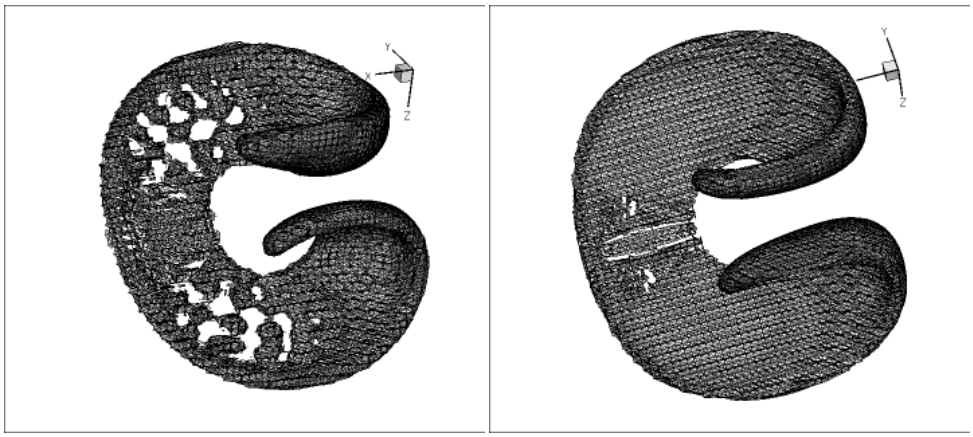
In this section, we test the DS-CLSMOF reconstruction algorithm for the problem of the rigid body rotation of Zalesak's disk in a constant vorticity velocity field [43]. The computational domain is $[0, 100] \times [0, 100]$. The disk is a slotted circle centered at $(50, 75)$ with a radius of 15, and a slot length of 25 and width of 5. The constant vorticity velocity field is given by:

$$u = -(\pi/314)(y - 50)$$

$$v = (\pi/314)(x - 50)$$

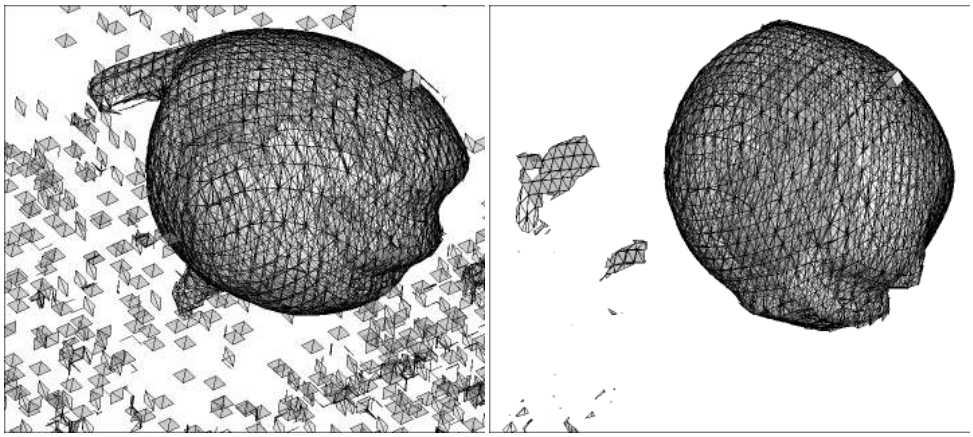
so that the disk completes one revolution every 628 time units.

In table 5 we report the symmetric difference error at $t = 628$ for the four different interface reconstruction methods (DS-CLSMOF, DS-MOF,DS-CLSVOF, and DS-LVIRA). In Figure 15, we compare the shape after one full rotation of the notched disk between DS-CLSMOF and DS-CLSVOF interface reconstruction. The effective fine grid resolution in Figure 15 is 100×100. In Figure 16 we compare



(a) DS-CLSVOF (b) DS-CLSMOF

Fig. 13 3D Deformation problem with period $T = 3$. Results at $t = 1.5$. The corresponding piecewise planar reconstructed interface is plotted. 64x64x64 grid.



(a) DS-CLSVOF (b) DS-CLSMOF

Fig. 14 3D Deformation problem with period $T = 3$. Results at $t = 3.0$. The corresponding piecewise planar reconstructed interface is plotted. 64x64x64 grid. Both methods have entrained "light" material, the DS-CLSMOF method has considerably less "flotsam" than the DS-CLSVOF method.

the shape after one full revolution between DS-CLSMOF and DS-CLSVOF when the effective fine grid resolution is 200×200. The criterion for adaptivity for the finest level of Figure 16, (Case C), was that $\kappa(\phi) > 1/(2\Delta x)$ where $\kappa(\phi)$ is the curvature. The corners in Figure 16 (Case C), are preserved almost just as well

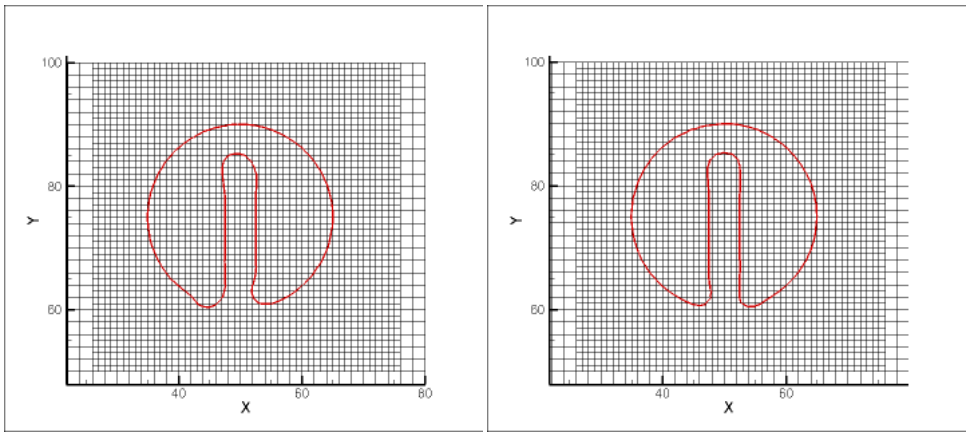
N	DS-CLSMOF	DS-MOF	DS-CLSVOF	DS-LVIRA
100	7.76	6.11	12.6	14.7
200	3.07 (1.3)	2.22 (1.5)	5.22	6.18
400	1.16 (1.4)	0.749 (1.6)	2.00	2.51

Table 5 Symmetric Difference Error from Section 7.3, the rotating, notched disk. measurement taken at the final time step, $t = 628$. The order of accuracy for the DS-CLSMOF and DS-MOF methods are reported in parenthesis.

as when the interface is wholly contained on the finest adaptive level (Case B). It took 1.9 times longer for the result in (B) to finish than in (C).

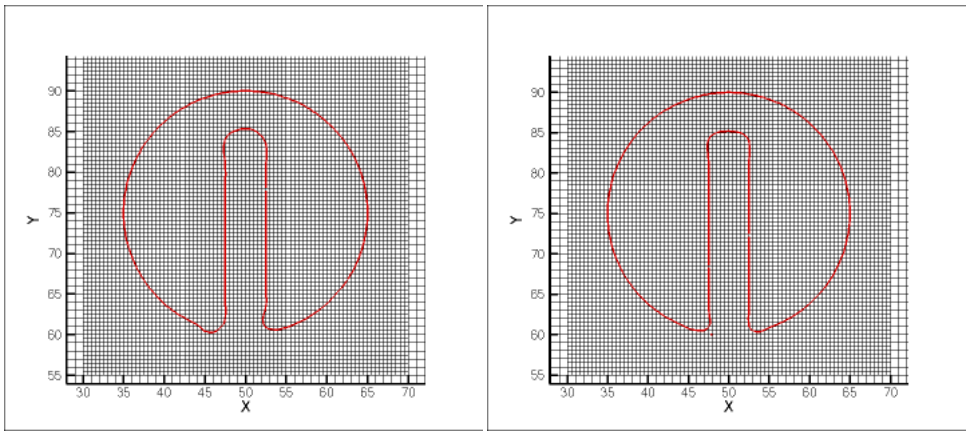
In Table 5, we see that the DS-MOF method has the smallest error, followed by the DS-CLSMOF method. The reason that DS-MOF is better than DS-CLSMOF is that the interface near the corners (but not within a grid cell of the corners) will have a curvature that does not trip the condition (4.21) for selecting the MOF slope. One possible remedy for this problem would be to increase β in (4.21), but then there would be a penalty in smooth regions (see table 3, $T = 1/2$, the error using either DS-CLSMOF or DS-CLSVOF is smaller than the error using DS-MOF) of the interface where the MOF slope would be selected instead of the CLSVOF slope. Another possible remedy, that should be tried in the future, is to replace condition (4.21) with the following condition for choosing the MOF slope:

$$\max_{-1<i'<1,-1<j'<1,-1<k'<1} |(\kappa(\Phi))_{i+i',j+j',k+k'}| > \frac{1}{\beta \Delta x}. \tag{7.14}$$



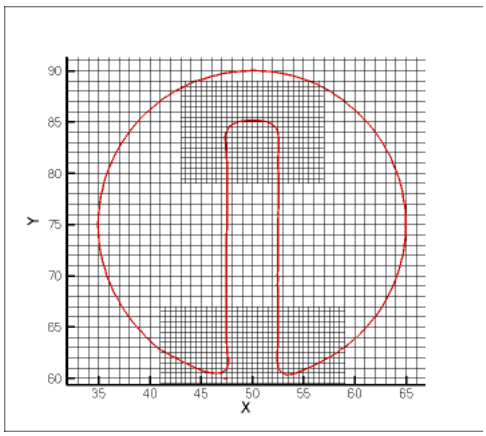
(a) DS-CLSVOF (b) DS-CLSMOF

Fig. 15 $N = 100$ results after one revolution for Zalesak's problem. The corresponding piecewise linear interface reconstruction is plotted at $t = 628$.



(a) DS-CLSVOF

(b) DS-CLSMOF



(c) DS-CLSMOF

Fig. 16 Results after one revolution ($t = 628$) for Zalesak's problem. The effective fine grid resolution is 200×200 for all 3 results. The coarse grid resolution is 50×50 and there are 2 levels of AMR. The criterion for the finest level of figure (c) is that $\kappa(\phi) > 1/(2\Delta x)$.

7.4 Letter "A": rigid body motion

The shape of the letter "A" is generated via the composition of level set functions and is passively advected. The test of a "rotating A" was also computed in [14]. As with our other passive advection tests, we shall compare the following 4 slope reconstruction algorithms: DS-MOF, DS-CLSMOF, DS-CLSVOF, and DS-LVIRA.

N	DS-CLSMOF	DS-MOF	DS-CLSVOF	DS-LVIRA
100	15.6	13.0	30.5	35.9
200	6.72 (1.2)	4.92 (1.4)	12.0	14.2
400	2.58 (1.4)	1.73 (1.5)	4.59	5.60

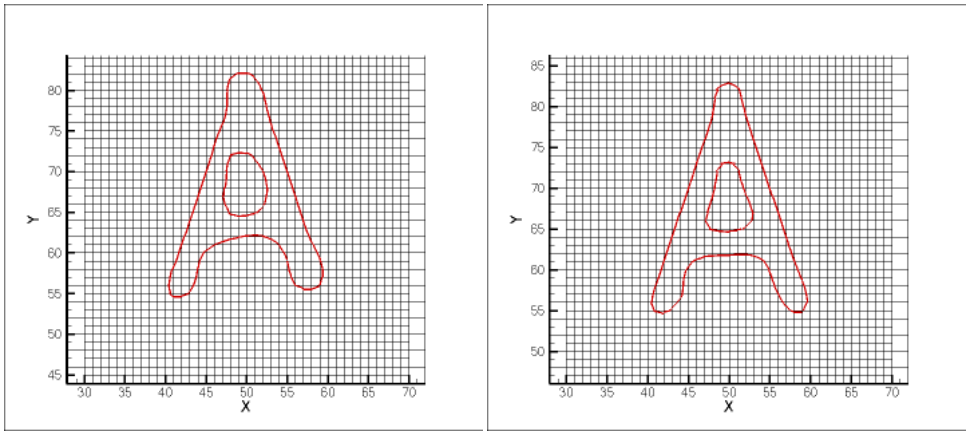
Table 6 Symmetric Difference Error from Section 7.4, the rotating "A." Measurement taken at the final time step, $t = 628$. The order of accuracy for the DS-CLSMOF and DS-MOF methods are reported in parenthesis.

The computational domain is $[0, 100] \times [0, 100]$. The constant vorticity velocity field is given by:

$$u = -(\pi/314)(y - 50)$$
$$v = (\pi/314)(x - 50)$$

so that the "A" shape completes one revolution every 628 time units. Table 6 reports the symmetric difference errors for the different slope reconstruction methods. In Figure 17, we compare the shape after one full rotation of the "A"; DS-CLSMOF versus DS-CLSVOF interface reconstruction.

Again, as in section 7.3, we see that the DS-MOF method has the smallest error, followed by the DS-CLSMOF method.



(a) DS-CLSVOF (b) DS-CLSMOF

Fig. 17 $N = 100$ results after one revolution for the rotating letter "A" problem. The corresponding piecewise linear interface reconstruction is plotted at $t = 628$.

7.5 Comparison with experiments for a rising gas bubble in liquid

We compare the computed rising bubble shape and velocity with experimental data reported in [10]. The effective fine grid resolution is 256x256 and the dimensionless domain size (with respect to the bubble diameter) is $0 \leq r \leq 4$ and

$0 \leq z \leq 4$. We use a 3D axisymmetric coordinate system. We prescribe symmetric boundary conditions at the left side of the computational domain, free slip boundary conditions at the right side, inflow boundary conditions at the top and outflow conditions at the bottom. The inflow velocity is a prescribed constant chosen so that the bubble would stay in the computational domain. See Table 7 for a list of the conditions we tested along with a comparison of the computed Reynolds number Re_{cal} (after a steady state has been reached) with the experimental observed Reynolds number Re_{exp}. In Figure 18 we compare the steady bubble shape with experimental data for conditions 1,3,4 and 6. We find from Table 7 that for all of the 6 different conditions we tried, we capture the correct steady bubble rising speed to within a half of a percent. In Figure 18, we illustrate 4 out of the 6 cases. In all of the cases, the aspect ratio of the computed bubble, when juxtaposed with the corresponding experimental photograph, "looks" the same.

	E_0	M	Re_{exp}	Re_{cal}
condition 1	116	41.1	7.16	7.16
condition 2	116	5.51	13.3	13.3
condition 3	116	1.31	20.4	20.3
condition 4	116	0.103	42.2	42.2
condition 5	115	$4.63E-3$	94.0	94.1
condition 6	243	266	7.77	7.77

Table 7 Comparison of computed Reynolds number $Re_{cal} = \rho_L V_{cal} d/\mu_L$ with the experimentally observed ([10]) Reynolds number $Re_{exp} = \rho_L V_{exp} d/\mu_L$. $E_0 = \frac{(\rho_L - \rho_G) g d^2}{\sigma}$ and $M = \frac{g \mu_L^4 (\rho_L - \rho_G)}{\rho_L^2 \sigma^3}$ represent the Eötvös and Morton numbers respectively.

7.6 Surface tension driven vibrations of a drop in zero gravity

In this section, we perform a grid refinement study for the problem of surface tension driven vibrations of a drop in zero gravity. We compare the DS-CLSMOF method to the DS-MOF method.

We have previously reported results for this problem, using DS-CLSVOF, in [40] and using a high order level set method in [42]. For this problem, the density ratio is 1000:1 and the viscosity ratio is 1000:1.

According to the linearized results derived in [23] (1932, §275), the position of the drop interface is

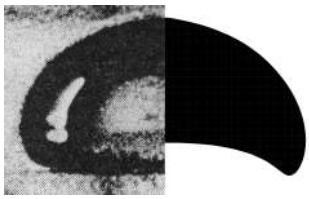
$$R(\theta, t) = a + \epsilon P_n(\cos(\theta)) \sin(\omega_n t + \pi/2),$$

where

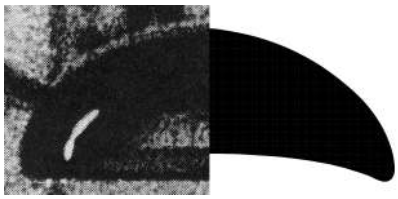
$$\omega_n^2 = \sigma \frac{n(n-1)(n+1)(n+2)}{a^3(\rho_l(n+1) + \rho_g n)}$$

and P_n is the Legendre polynomial of order n. θ runs between 0 and 2π, where $\theta = 0$ corresponds to $r = 0$ and $z = a$. If viscosity is present, Lamb (1932, §355) found that the amplitude is proportional to $e^{-t/\tau}$, where

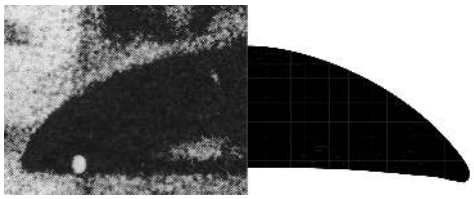
$$\tau = \frac{a^2 \rho_L}{\mu_L (2n+1)(n-1)}.$$



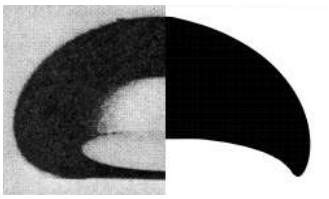
(a) Condition 1



(b) Condition 3



(c) Condition 4



(d) Condition 6

Fig. 18 Comparison of computed bubble steady shape (right side of each juxtaposed figure) with the experimentally observed bubble shape (left side of each juxtaposed figure - photo taken from [10]).

We compute the evolution of a drop with $a = 1$, $g = 0$, $\mu_L = 1/50$, $\mu_L/\mu_G = 1000$, $\sigma = 1/2$, $\rho_L = 1$ and $\rho_L/\rho_G = 1000$. The initial interface is given by $R(\theta, 0)$, with $\epsilon = .05$ and $n = 2$. With these parameters we find $\omega_2 = 2.0$ and $\tau = 10.0$. The fluid domain is $\Omega = \{(r, z)|0 \leq r \leq 1.5 \text{ and } 0 \leq z \leq 1.5\}$ and we compute on grid sizes ranging from 32×32 to 128×128. The time step for each respective grid size ranges from 0.0007 to 0.000175. Symmetric boundary conditions are imposed at $r = 0$ and $z = 0$.

In Table 8, we compare the relative error between succeeding resolutions for the minor amplitude $R_{\Delta x}(0, t)$ of the droplet when using the DS-CLSMOF method versus the DS-MOF method.

The average error $E^{avg}_{amplitude}$ is given by

$$E^{avg}_{amplitude} \equiv \int_0^{3.5} |R_{\Delta x}(0, t) - R_{2\Delta x}(0, t)|dt,$$

and the maximum amplitude error $E^{max}_{Amplitude}$ is given by

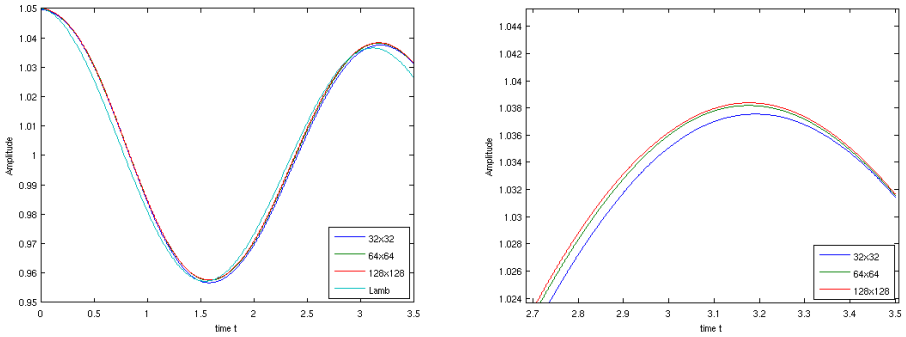
$$E^{max}_{Amplitude} \equiv \max_{0 \leq t \leq 3.5} |R_{\Delta x}(0, t) - R_{2\Delta x}(0, t)|.$$

In Figure 19, we plot the minor amplitude versus time for the three different grid resolutions using the DS-CLSMOF method.

From Table 8, we see that both DS-CLSMOF and DS-MOF methods converge under grid refinement. The relative error between the $\Delta r = 3/64$ and $\Delta r = 3/128$ meshes is larger for the DS-MOF method than the DS-CLSMOF method. But the DS-MOF method becomes slightly more accurate than the DS-CLSMOF method when looking at the relative error between the $\Delta r = 3/128$ and $\Delta r = 3/256$ meshes.

	DS-CLSMOF		DS-MOF	
Δr	$E^{avg}_{Amplitude}$	$E^{max}_{amplitude}$	$E^{avg}_{Amplitude}$	$E^{max}_{amplitude}$
3/128	0.00072	0.00148	0.00112	0.00328
3/256	0.00029 (1.3)	0.00084 (0.8)	0.00026 (2.1)	0.00082 (2.0)

Table 8 Convergence study; comparison between the DS-CLSMOF and DS-MOF methods for the vibration of a drop in zero gravity $\sigma = 1/2$, $\mu_L = 1/50$, $\mu_L/\mu_G = 1000$, $\rho_L/\rho_G = 1000$ and $\alpha = 2$. The order of accuracy for the DS-CLSMOF and DS-MOF methods are reported in parenthesis.



(a) $0 \leq t \leq 3.5$ (b) $2.7 \leq t \leq 3.5$

Fig. 19 Amplitude versus time for the simulation (DS-CLSMOF) of surface tension driven zero gravity droplet oscillations The linearized prediction from [23] is included in the plot for $0 \leq t \leq 3.5$. Density ratio is 1000:1. Viscosity ratio is 1000:1. Three different grid resolutions tested: 32×32, 64×64, 128×128. Δt equals 0.0007, 0.00035, and 0.000175 respectively.

7.7 Liquid-gas jet co-flowing flow

We compute the growth-rate of instability of a co-flowing jet using our DS-CLSMOF method and compare with a linear stability analysis (LSA) algorithm developed in [41]. The LSA algorithm is a general viscous, two-phase, three dimensional, algorithm. A coaxial flow, which satisfies exactly the Navier-Stokes equations, are given by [25]

$$W_1 = -1 + \frac{Nr^2}{N-(1-l^2)}\{1 - \frac{1-Q}{4N}R[2\ln l + (1-l^2)]\}$$

$$W_2 = -\frac{l^2-r^2}{N-(1-l^2)}\{1 - \frac{1-Q}{4N}R[2\ln l + (1-l^2)]\} + \frac{1-Q}{4N}R[l^2 - r^2 - 2ln(\frac{l}{r})]$$

$$U_1 = U_2 = V_1 = V_2 = 0$$

in which

$$N = \frac{\mu_2}{\mu_1}, \quad l = \frac{R_2}{R_1}, \quad Q = \frac{\rho_2}{\rho_1}$$

$$Re = \frac{\rho_1 W_0 R_1}{\mu_1}, \quad Fr = \frac{W_0}{gR_1}, \quad R = \frac{Re}{Fr}$$

where the subscript 1 and 2 stand for the liquid and gas phase streamwise velocity respectively, and g is the gravitational acceleration in the negative z direction.

The non-dimensional constants are defined as following:

$$N = 0.0018, \quad Q = 0.0013, \quad Re = 50, \quad We = 10$$

$$l = 10, Fr^{-1} = 0$$

As in [41,42], we will use the perturbed velocity derived from the LSA eigenvectors in order to prescribe an initial perturbed velocity for the DS-CLSMOF algorithm. The initial disturbance frequency is prescribed to be $\alpha = 0.5$. Figure 20 plots the amplitude versus time of the $\alpha = 0.5$ mode. The DS-CLSMOF method was used in computing the results. The coarsest simulation, 64×128, is computed on an AMR grid in which the coarse grid has dimensions 32×64 and there is one additional level of adaptivity. The finest simulation, 512×1024, is computed on an AMR grid in which the coarse grid has dimensions 32×64 and there are four additional levels of adaptivity.

The results here are converging faster than the results reported in [42]. We attribute this improvement to the new sharp interface treatment that we have implemented that follows the concepts proposed by [31] for treating the nonlinear advective terms. For early time, the growth rate is converging to the growth rate predicted by linear stability analysis. See Table 9. The computed growth rate was derived from the interface data, $r = f(z)$, by finding the Fourier expansion of $f(z)$ and identifying the amplitude of the $\alpha = 0.5$ mode, $A(t)$. The computed growth rate was then,

$$\omega = \frac{\ln A(t) - \ln A(0)}{t}. \tag{7.15}$$

t was chosen as the closest available time to 0.5.

fine grid resolution	ω
64x128	0.010
128x256	0.049
256x512	0.069
512x1024	0.078
LSA	0.088

Table 9 Growth rate of the $\alpha = 0.5$ mode predicted by the DS-CLSMOF method for the co-flowing jet problem. Growth rate measured for $0 < t < 1/2$. The Density ratio is $\rho^G/\rho^L = 0.0018$ and the viscosity ratio is $\mu^G/\mu^L = 0.0013$. Re= 50 and We= 10.

7.8 Collapsing Water Column - 2D

In this section, we compute the collapse of a 2D water column using our new DS-CLSMOF method. For reference, this problem has been previously simulated in [31] and experimental data has been reported in [27]. Density of the fluids are taken as $\rho_w = 1000$ kg/m^3 and $\rho_a = 1.226$ kg/m^3. The viscosity of the fluids are

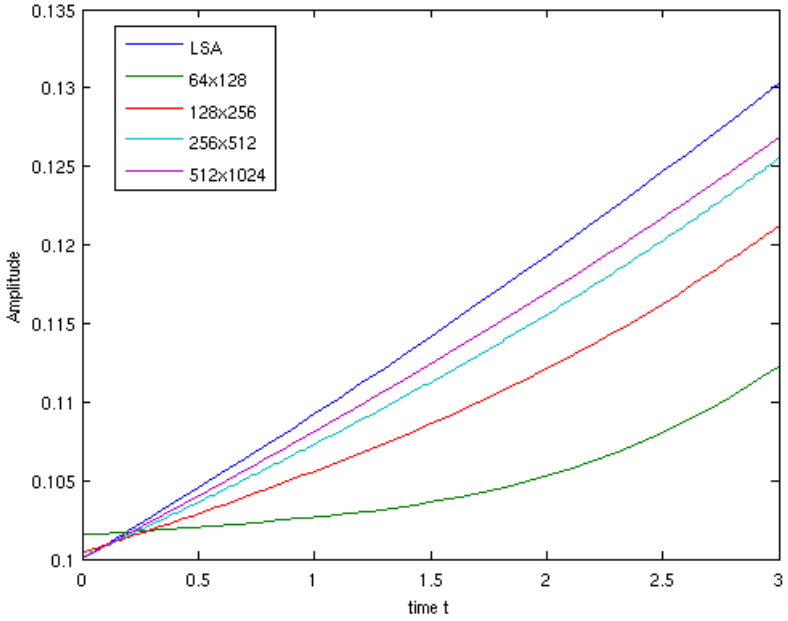


Fig. 20 Amplitude of the $\alpha = 0.5$ mode versus time predicted by the DS-CLSMOF method for the co-flowing jet problem. The coarsest simulation, 64×128, is computed on an AMR grid in which the coarse grid has dimensions 32×64 and there is one additional level of adaptivity. The finest simulation, 512×1024, is computed on an AMR grid in which the coarse grid has dimensions 32×64 and there are four additional levels of adaptivity. The Density ratio is $\rho^G/\rho^L = 0.0018$ and the viscosity ratio is $\mu^G/\mu^L = 0.0013$. Re= 50 and We= 10.

$\mu_w = 1.137 \times 10^{-3}$ kg/ms and $\mu_a = 1.78 \times^{-5}$ kg/ms. The surface tension coefficient is $\sigma = 0.0728$ N/m.

The initial height and width of the water column, $a = 5.715$ cm, are taken as the characteristic length of the system. The non-dimensional quantities are defined as $\mathrm{Fr} = u/\sqrt{ag} = 1$, $\mathrm{We} = \rho u^2 a/\sigma = 440$, and $\mathrm{Re} = \rho u a/\mu = 37635$. The computational domain has dimensions 7x7/4 in terms of the characteristic length scale. No-slip boundary conditions are prescribed on all 4 walls with a 90 degree contact angle. In Figure 22, we plot the dimensionless position of the front of the collapsing water column versus time. We compare our data to experiments. In Figure 21 we plot the interface profile at dimensionless time $t = 4$ for three different effective fine grid resolutions: 128x32, 256x64, and 512x128.

7.9 Impinging Jets

A case of liquid jet impingement is shown in Figure 24. Following the experiments reported in [6], the two identical water jets (density, kinematic viscosity and surface tension equal to 998 kg/m3, 0.076 N/m, and 1.00x10-6 m2/s, respectively) are inclined by an angle of 60 and injected in air at ambient conditions: the liquid-gas density ratio is therefore 832 and the dynamic viscosity ratio is 50. In the

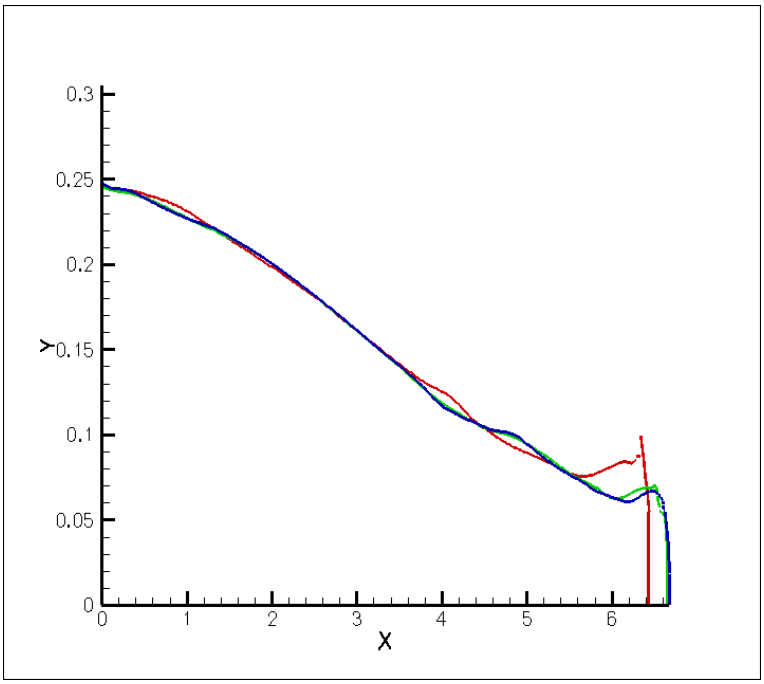


Fig. 21 Interface profile at dimensionless time $t = 4$ for a 2D dam break problem. The DS-CLSMOF piecewise linear reconstructed interface is plotted. The density ratio is 1.226/1000.0 and the viscosity ratio is 0.0178/1.137. The red, green, blue interfaces correspond to an effective fine grid resolution of 128x32, 256x64, and 512x128 respectively.

simplified setting of this simulation, the pre-impingement length (25.4 mm in the experiment) is shortened to 4 mm and the boundary velocity profile is that of a plug flow. The initial diameter of the jets is d0 = 0.635 mm. The computational domain of 16 x 12 x 24 mm with four levels of refinement, giving a final mesh density of 0.03125 mm in the space occupied by the liquid phase. A previous grid convergence study suggests that an additional local refinement by two is necessary to capture the correct spray size distribution downstream of the injection: the main purpose of Figure 23 is to illustrate the difference in terms of mass conservation between the DS-CLSMOF and the DS-CLSVOF methodology.

7.10 Six holed injector

Figure 25 displays a simulation snapshot of liquid injection in a Bosch six-hole vertical diesel nozzle manufactured at a 20:1 scale to allow flow measurements inside the sac volume and the injection holes ([7]); the nominal hole size of the model is 3.52 mm, which corresponds to a hole size of about 0.176 mm in the real injector. The conical element (the needle) is placed at 6 mm above its seat, at the maximum lift of the second stage of a real size two-stage injector. The working fluid has a density and kinematic viscosity of 0.893 kg/m3 and 1.6410-6 m2/s, respectively. The surface tension coefficient is 0.024 N/m. The computational domain of 100 x

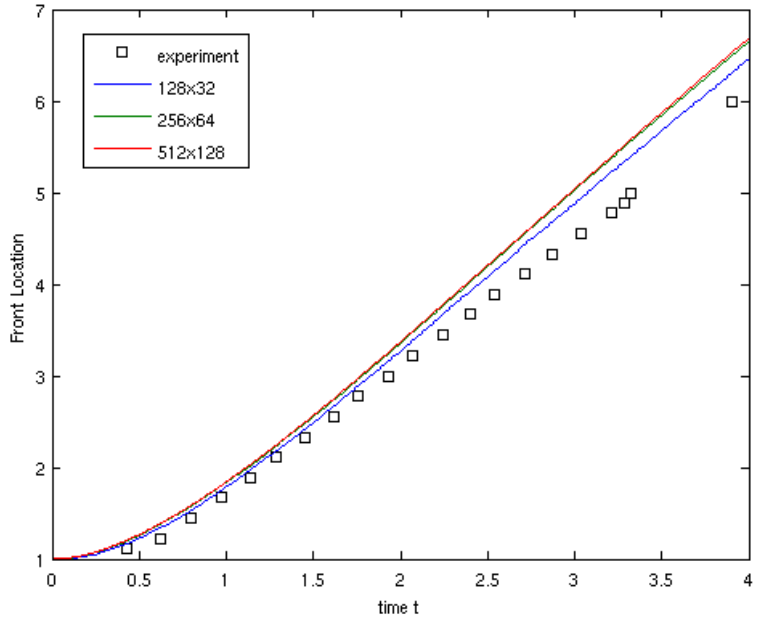


Fig. 22 Dimensionless front position versus time for a collapsing 2D water column. The experimental data reported in [27] is included in the plot.

100 x 40 mm is sufficiently large to include both the flow inside the nozzle and several orifice diameters of diesel jet in standard air. Starting with a coarse mesh of 80 x 80 x 32, the three levels of adaptive refinement used in the simulation provide a minimum mesh distance of 0.15625 mm. This grid density is reached at the interior walls and at the free surface of the jets. The injector wall geometry is modeled by a triangular mesh of 28814 elements: the minimum node spacing of 0.4 mm is reached at the six orifices and smoothly increases everywhere else. Variations from jet to jet in the simulation can be attributed to small differences in the orifice geometries.

8 Conclusions

A CLSMOF interface reconstruction technique was presented for simulating deforming interfaces. We have applied our new CLSMOF algorithm for solving incompressible two phase flows in 2D and 3D. The advection terms are discretized in time using directional splitting. The momentum equations are discretized using a novel cell centered projection method for two phase flows in which we have combined the ideas of Raessi and Pitsch (2009) and of Kwatra et al (2009) in which mass weighting is used when interpolating the velocity and "inverse mass weighting" is used when interpolating pressure from the cell centers to the cell faces. For many test problems that we have tried, not heretofore tried with the MOF method, the CLSMOF method and CLSVOF method will have comparable

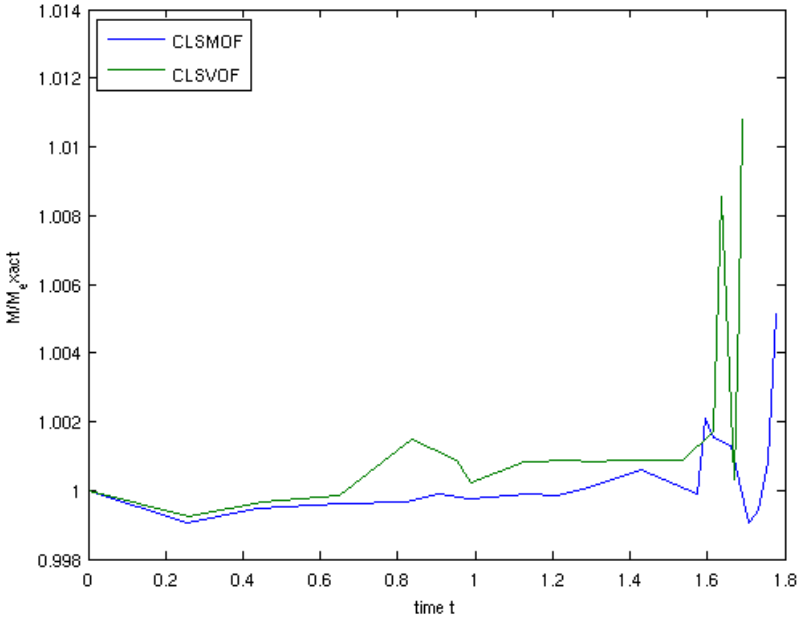
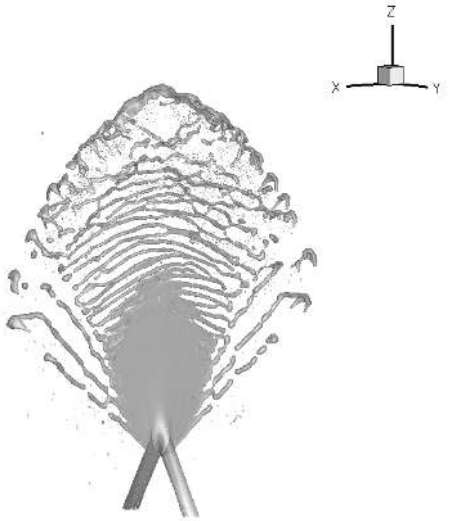


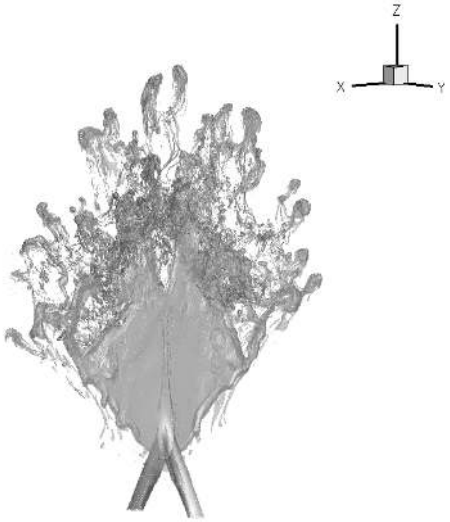
Fig. 23 $M(t)/M_{expect}(t)$ for the simulation of two impinging jets. DS-CLSMOF mass fluctuates less about the expected value than DS-CLSVOF.

accuracy (e.g. rising bubble problem, co-flowing jet problem, zero gravity droplet oscillations). On the other hand, for 3D problems with deforming, stretching, disintegrating, interfaces, we show that the CLSMOF method preserves the fidelity of the interface much better than the CLSVOF or VOF (LVIRA reconstruction) methods (e.g. 3D reversible vortex problem, impinging jets, flow through a six holed diesel injector).

We find that for the single vortex problem, our directionally split MOF method (or CLSMOF method) under performs by about 30 percent compared to the unsplit MOF method [2]. One improvement for the future would be to extend our directionally split CLSMOF method to an unsplit method. Another possibility for improvement is to derive the density directly from the volume-of-fluid function (from the CLSMOF piecewise linear reconstructed interface) instead of from the level set function, so that even when a jet breaks up into tiny droplets, the droplets will still have an effect on the velocity field. In Figure 16, (Case B), we used curvature, derived from the level set function, as a criterion for adaptivity. In the future, we will investigate using the discrepancy between the reference centroid and actual centroid (4.3) [2] as a criterion for adaptivity. Finally, we wish to further develop the CLSMOF or MOF reconstruction algorithm in order to solve problems with 3 or more materials with deforming boundaries as well as be able to handle the scenario when both sides of a filament are wholly contained within a computational cell.



(a) DS-CLSVOF



(b) DS-CLSMOF

Fig. 24 Interface profile for impinging jets. Effective fine grid resolution: 256x192x384.

References

1. Ahn, H., Shashkov, M.: Geometric algorithms for 3d interface reconstruction. In: IMR'07, pp. 405–422 (2007)

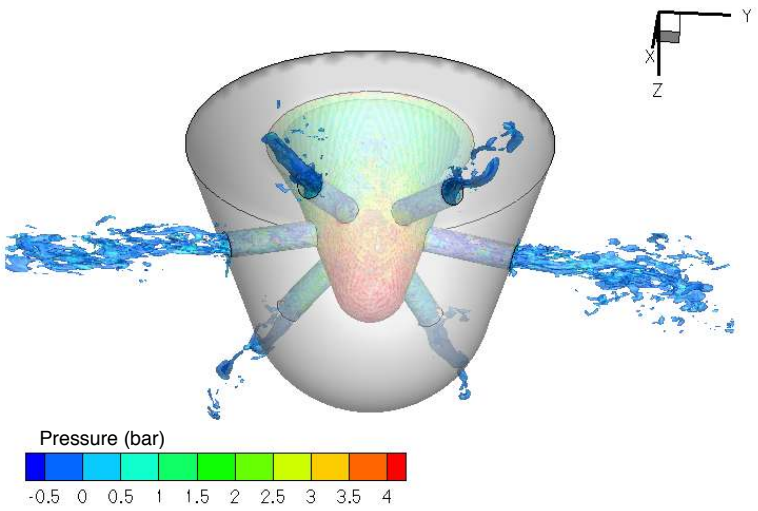


Fig. 25 Snapshot of a simulation of liquid injection in a Bosch six-hole vertical diesel nozzle. Effective fine grid resolution 640x640x256.

2. Ahn, H., Shashkov, M.: Adaptive moment-of-fluid method. J. Comput. Phys. **228(8)**, 2792–2821 (2009)
3. Ahn, H., Shashkov, M., Christon, M.: The moment-of-fluid method in action. Communications in Numerical Methods in Engineering **25(10)**, 1009–1018 (2009)
4. Ahn, H.T., Shashkov, M.: Multi-material interface reconstruction on generalized polyhedral meshes. J. Comput. Phys. **226**, 2096–2132 (2007). DOI 10.1016/j.jcp.2007.06.033. URL http://dl.acm.org/citation.cfm?id=1290206.1290465
5. Almgren, A.S., Bell, J.B., Colella, P., Howell, L.H., Welcome, M.: A conservative adaptive projection method for the variable density incompressible Navier-Stokes equations. J. Comput. Phys. **142**, 1–46 (1998)
6. Anderson, W., Ryan, H., Santoro, J., Hewitt, R.: Combustion instability mechanism in liquid rocket engines using impinging jet injectors. In: 31st AIAA ASME SAE ASEE Joint Propulsion Conference, AIAA 95-2357 (1995)
7. Arcoumanis, C., Gavaises, J., Nouri, E.A.W., Horrocks, R.: Analysis of the flow in the nozzle of a vertical multi hole diesel engine injector. Tech. Rep. SAE Paper 980811 (1998)
8. Bell, J.B., Colella, P., Glaz, H.M.: A second-order projection method for the incompressible Navier-Stokes equations. J. Comput. Phys. **85**, 257–283 (1989)
9. Berger, M.J., Rigoustsos, I.: An algorithm for point clustering and grid generation. Tech. Rep. NYU-501, New York University-CIMS (1991)
10. Bhaga, D., Weber, M.: Bubbles in viscous liquids: Shapes, wakes and velocities. J. Fluid Mech. **105**, 61–85 (1981)
11. Cervone, A., Manservisi, S., Scardovelli, R., Zaleski, S.: A geometrical predictor-corrector advection scheme and its application to the volume fraction function. J. Comput. Phys. **228(2)**, 406–419 (2009)
12. Chang, Y., Hou, T., Merriman, B., Osher, S.: Eulerian capturing methods based on a level set formulation for incompressible fluid interfaces. J. Comput. Phys. **124**, 449–464 (1996)
13. Cummins, S., Francois, M., Kothe, D.: Estimating curvature from volume fractions. Computers and Structures **83**, 425–434 (2005)

14. Dyadechko, V., Shashkov, M.: Moment-of-fluid interface reconstruction. Technical Report LA-UR 07-1537, Los Alamos National Laboratory (2007)
15. Dyadechko, V., Shashkov, M.: Reconstruction of multi-material interfaces from moment data. J. Comput. Phys. **227(11)**, 5361–5384 (2008)
16. Enright, D., Nguyen, D., Gibou, F., Fedkiw, R.: Using the particle level set method and a second order accurate pressure boundary condition for free surface flows. In: M. Kawahashi, A. Ogut, Y. Tsuji (eds.) Proc. of the 4th ASME-JSME Joint Fluids Eng. Conf., FEDSM2003-45144. Honolulu, HI (2003)
17. Fedkiw, R., Aslam, T., Xu, S.: The ghost fluid method for deflagration and detonation discontinuities. Technical Report CAM Report 98-36, University of California, Los Angeles (1998)
18. Francois, M., Cummins, S., Dendy, E., Kothe, D., Sicilian, J., Williams, M.: A balanced-force algorithm for continuous and sharp interfacial surface tension models within a volume tracking framework. J. Comput. Phys. **213(1)**, 141–173 (2006)
19. Gross, S., Reusken, A.: An extended pressure finite element space for two-phase incompressible flows with surface tension. Journal of Computational Physics **224(1)**, 40–58 (2007)
20. Helmsen, J., Colella, P., Puckett, E.: Non-convex profile evolution in two dimensions using volume of fluids. LBNL technical report LBNL-40693, Lawrence Berkeley National Laboratory (1997)
21. Kang, M., Fedkiw, R., Liu, X.D.: A boundary condition capturing method for multiphase incompressible flow. J. Sci. Comput. **15**, 323–360 (2000)
22. Kwatra, N., Su, J., Grétarsson, J.T., Fedkiw, R.: A method for avoiding the acoustic time step restriction in compressible flow. J. Comput. Phys. **228**, 4146–4161 (2009). DOI 10.1016/j.jcp.2009.02.027. URL http://dl.acm.org/citation.cfm?id=1528939.1529237
23. Lamb, H.: Hydrodynamics. Dover Publications, New York (1932)
24. van Leer, B.: Towards the ultimate conservative difference scheme. V. a second-order sequel to Godunov's method. J. Comput. Phys. **32**, 101–136 (1979)
25. Lin, S., Chen, J.: Role played by the interfacial shear in the instability mechanism of a viscous liquid jet surrounded by a viscous gas in a pipe. J. Fluid Mech. **376**, 37–51 (1998)
26. Marchandise, E., Remacle, J.: A stabilized finite element method using a discontinuous level set approach for solving two phase incompressible flows. Journal of Computational Physics **219(2)**, 780–800 (2006)
27. Martin, J., Moyce, W.J.: An experimental study of the collapse of liquid columns on a rigid horizontal plane. Phil. Trans. R. Soc. Lond. A **244**, 312–324 (1952)
28. Osher, S., Sethian, J.A.: Fronts propagating with curvature-dependent speed: Algorithms based on hamilton-jacobi formulations. J. Comput. Phys. **79**(1), 12–49 (1988)
29. Pilliod, J., Puckett, E.: Second-order accurate volume-of-fluid algorithms for tracking material interfaces. Journal of Computational Physics **199(2)**, 465–502 (2004)
30. Popinet, S.: An accurate adaptive solver for surface-tension-driven interfacial flows. J. Comput. Phys. **228**, 5838–5866 (2009)
31. Raessi, M., Pitsch, H.: Modeling interfacial flows characterized by large density ratios with the level set method. Annual research brief, Center for Turbulence Research (2009)
32. Schofield, S., Christon, M.: Effects of element order and interface reconstruction in fem/volume-of-fluid incompressible flow simulation. International Journal for Numerical Methods in Fluids p. `DOI:10.1002/fld.3657` (2012)
33. Stewart, P., Lay, N., Sussman, M., Ohta, M.: An improved sharp interface method for viscoelastic and viscous two-phase flows. Journal of Scientific Computing **35(1)**, 43–61 (2008)
34. Strang, G.: On the construction and comparison of difference schemes. SIAM J. Numer. Anal. **5**, 506–517 (1968)
35. Sussman, M.: A second order coupled levelset and volume of fluid method for computing growth and collapse of vapor bubbles. Journal of Computational Physics **187**, 110–136 (2003)
36. Sussman, M.: A parallelized, adaptive algorithm for multiphase flows in general geometries. Computers and Structures **83**, 435–444 (2005)
37. Sussman, M., Almgren, A., Bell, J., Colella, P., Howell, L., Welcome, M.: An adaptive level set approach for incompressible two-phase flows. J. Comput. Phys. **148**, 81–124 (1999)
38. Sussman, M., Ohta, M.: A stable and efficient method for treating surface tension in incompressible two-phase flow. SIAM J. Sci. Comput. **31**(4), 2447–2471 (2009)

39. Sussman, M., Puckett, E.: A coupled level set and volume of fluid method for computing 3D and axisymmetric incompressible two-phase flows. J. Comp. Phys. **162**, 301–337 (2000)
40. Sussman, M., Smith, K., Hussaini, M., Ohta, M., Zhi-Wei, R.: A sharp interface method for incompressible two-phase flows. J. Comp. Phys. **221(2)**, 469–505 (2007)
41. Wang, Y.: Numerical methods for two-phase jet flow. Ph.D. thesis, Dept. of Mathematics, Florida State University (2010)
42. Wang, Y., Simakhina, S., Sussman, M.: A hybrid level set-volume constraint method for incompressible two phase flows. Journal of Computational Physics (in review) (2012)
43. Zalesak, S.T.: Fully multidimensional flux-corrected transport algorithms for fluids. J. Comput. Phys. **31**, 335–362 (1979)