ILASS-Americas 22nd Annual Conference on Liquid Atomization and Spray Systems, Cincinnati, OH, May 2010

A level-set based methodology for modeling interfacial flows characterized by large density ratios

M. Raessi*and H. Pitsch
Center for Turbulence Research
Stanford University, Stanford, CA 94305

Abstract

We introduce a numerical methodology for modeling interfacial flows characterized by large density ratios. The method uses the temporal evolution of a signed distance function, commonly employed in the level set method to represent fluid interfaces. In this method, the conservative form of the momentum equation is solved to advect momentum. To calculate the density term in momentum flux at any point, we use the signed distance function at that point at two subsequent time steps. For consistency, we use the same density for mass flux, and thereby establish a tight coupling between mass and momentum transport. We present a set of results in which the density ratio ranges from 650 to 10000. The methodology is validated using theoretical and experimental test cases. Finally, to demonstrate the capability of the methodology in handling flows with large interface deformations, we present simulations of liquid sheet breakup in shear air.

---

*Corresponding Author: mraessi@stanford.edu

Introduction

Many important industrial and research applications involve interfacial flows – flows consisting of two or more immiscible fluids with distinct interfaces. These applications include sprays and atomizer, drop-on-demand generators used, for example, in ink-jet printing and DNA sampling, casting, coating, and oil recovery processes. One common characteristic among the interfacial flows encountered in these applications is the large density jump across fluid interfaces. The large density ratio, which is typically around 1000 in these applications, is indeed a challenge for many interfacial flow solvers. A literature survey suggests that there are several incompressible interfacial flow solvers that, due to numerical stability issues, are limited by some density ratio that is significantly smaller than a realistic value. However, using a realistic density ratio in simulations of a problem is generally essential in order to perform predictive simulations that truly represent the problem of interest.

In the category of incompressible interfacial flow solvers that employ the volume-of-fluid method [1, 2] for modeling interface kinematics, Rudman [3] and Bussmann et al. [4] suggest two approaches to modeling flows with large density ratios. The key points shared among their approaches are two-fold: (i) momentum is transported by solving the conservative form of the equation, and (ii) the mass flux information is used to compute the density term in the momentum flux, thereby achieving consistent mass and momentum advection schemes.

The importance of solving the conservative form of the equation for momentum transport (point (i)) was also recently stressed by Park et al. [5]. They observed that solving the non-conservative form of the equation yields incorrect velocity fields and fluid volume fractions in a two-phase flow with large density gradients, while solving the conservative form works significantly better.

To further explain point (ii), we consider the example depicted in Fig. 1, which shows two adjacent cells in a staggered arrangement of variables, where the scalars (e.g. density) are defined at numerical cell centers i and $i+1$, and the velocities are on the faces $(i+1/2)$. Cell i contains a portion of the interface separating fluids 1 and 2. To compute momentum transfer across the interface and update face velocity u_f, densities at surfaces i and $i+1$ are required to compute momentum fluxes. In the approaches suggested in [3, 4], to calculate the density term in the momentum flux across surface i, area (in 2-D, or volume in 3-D) A_1 is assigned the density of fluid 1 and A_2 the density of fluid 2. The density term in the momentum flux at face i, denoted by $\tilde{\rho}_i$, is then

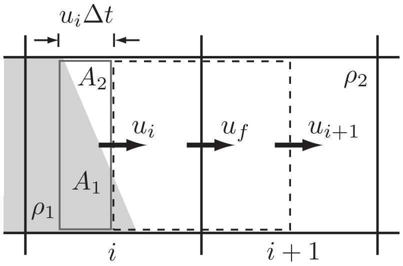

Figure 1. Advection of mass and momentum in two-fluid flows.

$$\tilde{\rho}_i = \frac{\rho_1 A_1 + \rho_2 A_2}{A_1 + A_2}, \tag{1}$$

which is the average density of fluids crossing surface i. The density term at surface $i+1$ is simply ρ_2.

Rudman [3] offers a technique for finite-difference, staggered flow solvers, where mass is advected on a grid twice as fine as the flow solver grid. Bussmann et al. [4] present a technique for finite-volume flow solvers with collocated arrangement of variables in which mass and momentum control volumes coincide. However, because in both approaches [3, 4], the volume of fluid (VOF) method was used for modeling interface kinematics, areas like A_1 and A_2 of the above example are readily available. In fact, the VOF method requires these areas to calculate mass fluxes and advect fluid volumes.

An equally common group of interfacial flow solvers utilizes the level set (LS) method [6, 7] for modeling interface kinematics. In the LS method, because mass fluxes are not explicitly calculated, areas A_1 and A_2 of the above example are not readily available. Hu et al. [8] suggested an approach to modeling flows with large density gradients in the LS context, but their method is suitable for flow solvers that solve multiple sets of flow equations (each phase is solved individually).

In this work, our focus is on flow solvers that use a single set of flow equations for all phases. Our objective is to develop a numerical technique for modeling interfacial flows characterized by large density ratios in the context of the LS method. In this paper, we will first present the governing equations of interfacial flows, with a brief discussion on the numerical methods used in our flow solver (NGA) [9, 10, 11]. Then, details of the new technique for handling flows with large density ratios and its implementation are presented. Finally, a set of results are presented that demonstrate the accuracy and efficacy of the new technique.

Governing equations

The governing equations for an interfacial flow of Newtonian, immiscible, and incompressible fluids are conservation of mass and momentum

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \boldsymbol{U}) = 0 \tag{2}$$

$$\frac{\partial (\rho \boldsymbol{U})}{\partial t} + \nabla \cdot (\rho \boldsymbol{U}\boldsymbol{U}) = -\nabla p + \nabla \cdot \tau + \boldsymbol{F}_B, \tag{3}$$

where $\boldsymbol{U}$ denotes velocity, p pressure, τ the shear stress tensor $\tau = \mu(\nabla \boldsymbol{U} + \nabla \boldsymbol{U}^T)$, and $\boldsymbol{F}_B$ any body forces.

The following conditions apply across fluid interface: for viscous fluids with no mass interchange between two different phases of a single substance, the velocity components normal and tangential to fluid interfaces are equal in both fluids [12]. That is,

$$[\boldsymbol{U}] \equiv \boldsymbol{U}^{(1)} - \boldsymbol{U}^{(2)} = 0. \tag{4}$$

where superscripts (1) and (2) denote fluids 1 and 2.

Furthermore, the following jump condition for the stress tensor exists across fluid interfaces [13]:

$$[(-pI + \tau) \cdot \hat{n}] = \sigma \kappa \hat{n}, \tag{5}$$

where $\hat{n}$ is the unit normal vector to the interface, σ is the surface tension coefficient (assumed constant) and κ is interface curvature.

Using the level set method to model the interface kinematics, we solve the following transport equation for a scalar indicator function $G = G(\boldsymbol{x}, t)$:

$$\frac{\partial G}{\partial t} + \boldsymbol{U} \cdot \nabla G = 0. \tag{6}$$

Although G can be any smooth function, the most commonly used one is a signed distance function. The interface is represented by an isosurface $G_0 = 0$. We solve a single set of Eqs. (2) and (3) in all fluids; the fluid properties (assumed constant in each fluid) are determined from G at any point.

Numerical method

Before presenting the flow solver and the details of the new method for modeling flows with large density ratios, it is worth noting that a common approach to modeling incompressible interfacial flows is to solve Eq. (3) in the following non-conservative form

$$\frac{\partial \boldsymbol{U}}{\partial t} + \boldsymbol{U} \cdot \nabla \boldsymbol{U} = \frac{1}{\rho}\left(-\nabla p + \nabla \cdot \tau + \boldsymbol{F}_B\right). \tag{7}$$

As discussed in [4], the above non-conservative formulation is likely fine at low density ratios where the jump in fluid densities has less importance in the change of momentum in interface cells than local variations of velocity. However, as shown below, in flows with high density ratios (~ 1000), solving the non-conservative form in the momentum transport step results in large errors. Therefore, in our new method, we solve the conservative form of the equation to transport momentum. Details of the new method is presented next.

Flow solver

The new method for handling flows with large density ratios was implemented in the framework of NGA, which is an in-house code described extensively in [9, 10, 11]. NGA is a structured, finite-difference flow solver with staggered arrangement of variables. It employs the projection method to solve the governing equations (2) and (3). NGA is a parallel code designed for direct numerical simulation and large-eddy simulation of turbulent reactive flows.

In our new method, unlike [10, 11], we solve Eq. (3) in the conservative form when advecting momentum, i.e. as the first step in the time advancement, the following equation is solved to transport momentum,

$$\frac{\partial \rho \boldsymbol{U}}{\partial t} = -\nabla \cdot (\rho \boldsymbol{U}\boldsymbol{U}), \tag{8}$$

That is, at the momentum transport step in the projection method, we evaluate an interim velocity $\boldsymbol{U}^*$ via[1],

$$\frac{\rho^{n+1}\boldsymbol{U}^* - (\rho \boldsymbol{U})^n}{\Delta t} = -\nabla \cdot (\rho \boldsymbol{U}\boldsymbol{U})^n, \tag{9}$$

Following the momentum transport step, we obtain $\boldsymbol{U}$ from $\rho \boldsymbol{U}$ and return to the non-conservative formulation. We then follow the methods described in [10] to implement the remaining terms in Eq. (7). These methods include the Ghost-Fluid method (GFM) [10, 14], which is employed to account for density jumps in the pressure term as well as the surface tension force. Note that the only difference between the new method and [10] lies in the treatment of the momentum transport.

Calculating the momentum flux

To solve Eq. (8) for momentum transport, we need the divergence of the momentum flux $\rho \boldsymbol{U}\boldsymbol{U}$ (i.e. RHS of Eq. (8)) at cell faces where velocity is defined (see Fig. 2). For that, we compute the momentum flux $\rho \boldsymbol{U}\boldsymbol{U}$ at cell centers (points i and $i+1$ in Fig. 2). We evaluate $\rho \boldsymbol{U}\boldsymbol{U}$ term by term as follows:

---

[1] First-order time discretization is for demonstration only; in practice the temporal discretization is second-order accurate or higher.

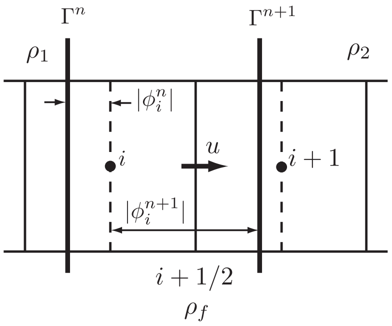

Figure 2. Calculation of flux density across surface i from the signed distance function ϕ at two consecutive time levels n and $n+1$.

The velocities in $\rho\boldsymbol{UU}$ is computed by a central interpolation scheme which gives the velocity at cell centers. To compute the density term in the momentum flux, which is denoted by $\tilde{\rho}$ and referred to as "flux density", we introduce a new technique in the level set context. This technique, which is based on the signed distance function ϕ, is explained by the following 1-D example:

Consider an interface at time step n, denoted by Γ^n, in a staggered arrangement of variables, as shown in Fig. 2. The distance function for node i at time n is shown as ϕ_i^n. As the interface moves to its new location Γ^{n+1}, the flux across the surface at i (shown by dashed line) consists of some amount of fluid 2 and more of fluid 1. To obtain $\tilde{\rho}$ at i from time step n to $n+1$, we can establish the following formulations in 1-D (note that this formulation is valid anywhere in the domain not just near fluid interfaces):

Assuming $\phi > 0$ in fluid 1, if $\phi_i^n \phi_i^{n+1} > 0$ or $\phi_i^n = 0$ on the surface, the flux density is

$$\tilde{\rho}_i = \begin{cases} \rho_1 \; ; & \phi_i^{n+1} \geq 0 \\ \\ \rho_2 \; ; & \text{otherwise} \end{cases} . \tag{10}$$

else,

$$\tilde{\rho}_i = \begin{cases} \frac{|\phi_i^n|\rho_2 + |\phi_i^{n+1}|\rho_1}{|\phi_i^n| + |\phi_i^{n+1}|} \; ; & \phi_i^n < 0 \\ \\ \frac{|\phi_i^n|\rho_1 + |\phi_i^{n+1}|\rho_2}{|\phi_i^n| + |\phi_i^{n+1}|} \; ; & \text{otherwise} \end{cases} . \tag{11}$$

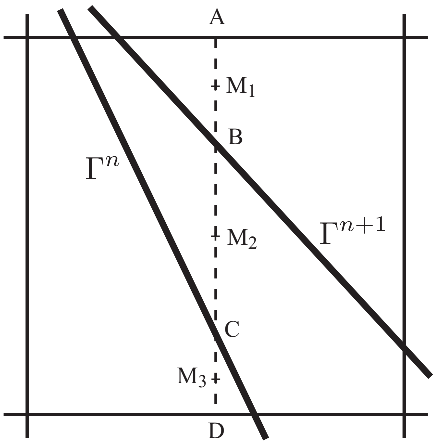

Figure 3. An interface Γ at time steps n and $n+1$ in a 2-D numerical cell.

For extending these formulations to 2-D, consider an interface at time steps n and $n+1$ in a numerical cell shown in Fig. 3. The total flux density across the face AD (see Fig. 3) is calculated by splitting AD into three segments: AB, BC and CD, which are found from the intersections between AD and Γ^n or Γ^{n+1}. Then, the above 1-D calculation is performed for the midpoint of each line segment, denoted by M_1, M_2 and M_3. The total flux density across face AD is then

$$\tilde{\rho}_{AD} = \frac{\tilde{\rho}_{M_1}\overline{AB} + \tilde{\rho}_{M_2}\overline{BC} + \tilde{\rho}_{M_3}\overline{CD}}{\overline{AD}} . \tag{12}$$

It is easy to note that if an interface moves normal to a surface, the above method yields a time-averaged density along that surface, which is, in fact, a very good estimation. The intersection of an interface and cell boundaries are found by the marching cube algorithm.

Computing $\boldsymbol{U}$ from $\rho\boldsymbol{U}$

To obtain $\boldsymbol{U}$ from $\rho\boldsymbol{U}$ following the momentum transport step, densities at cell faces (where $\boldsymbol{U}$ is located; see Fig. 2) are required; we denote density on cell faces by ρ_f. To calculate ρ_f^{n+1}, we solve the following conservation of mass equation

$$\frac{\partial \rho_f}{\partial t} = -\nabla \cdot (\rho \mathbf{U}), \tag{13}$$

where the initial condition is computed based on the level set field at time n. The crucial point in solving Eq. (13) is to use the same flux densities $\tilde{\rho}$ as those used in Eq. (8). This ensures consistent mass and momentum transports.

An alternative to the above approach is to use the level set function at time $n+1$ to calculate ρ_f^{n+1} but that, in general, does not guarantee consistent mass and momentum transport simply because the advections of momentum and the level set function are not coupled. Small discrepancies between the transport of momentum and the level set function cause mismatch between mass and momentum and result in non-physical velocities, especially evident at large density ratios.

For the example shown in Fig. 2, in which we assume a constant velocity u here, the flux density across surfaces i and $i+1$ are

$$\tilde{\rho}_i = \frac{|\phi_i^n|\rho_2 + |\phi_i^{n+1}|\rho_1}{|\phi_i^n| + |\phi_i^{n+1}|},$$

and $\tilde{\rho}_{i+1} = \rho_2$. ρ_f on face $i+1/2$ can be calculated from Eq. (13) as follows [2]

$$\frac{\rho_f^{n+1} - \rho_f^n}{\Delta t} = -\frac{u}{\Delta x}(\rho_2 - \tilde{\rho}_i) = -\frac{u}{\Delta x}\frac{|\phi_i^{n+1}|(\rho_2 - \rho_1)}{|\phi_i^n| + |\phi_i^{n+1}|}, \tag{14}$$

where Δx denotes the distance between points i and $i+1$ and Δt is the time step between times n and $n+1$. Since $u\Delta t = |\phi_i^n| + |\phi_i^{n+1}|$,

$$\rho_f^{n+1} - \rho_f^n = \frac{|\phi_i^{n+1}|}{\Delta x}(\rho_1 - \rho_2) \tag{15}$$

And, because $\rho_f^n = \rho_2$, we obtain

$$\rho_f^{n+1} = \frac{|\phi_i^{n+1}|\rho_1 + (\Delta x - |\phi_i^{n+1}|)\rho_2}{\Delta x} \tag{16}$$

which is a convex combination of fluids densities ρ_1 and ρ_2.

Evolution of fluids interfaces

To transport the interface, Eq. (6) is solved using a spectrally refined interface approach [11]: a set of quadrature points are used to define G on subgrid levels with a polynomial reconstruction. A semi-Lagrangian technique is then used to transport G. Interfacial quantities $\hat{n}$ and κ are computed from G by

$$\hat{n} = \frac{\nabla G}{|\nabla G|} \tag{17}$$

and

$$\kappa = -\nabla \cdot \hat{n}. \tag{18}$$

Results

The remainder of this paper presents the results of the new method. First, we study the convergence rate of the new algorithm for calculating flux densities. Then, we assess the performance of the new method (in which Eq. (8) is solved for momentum transport) in solving actual flow problems characterized by large density ratios, as well as comparisons between the results of the new method and the non-conservative formulation (Eq. (7)). Note that the only difference between the two formulations lies in the momentum transport (treatment of the convective term); the other terms are treated exactly the same.

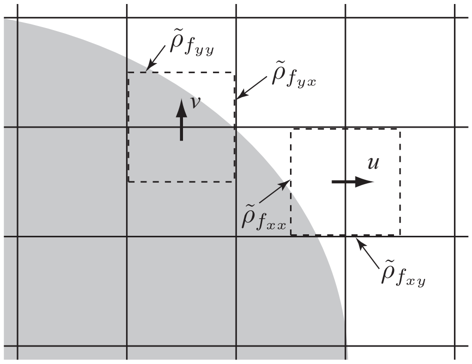

Figure 4. Flux densities across surfaces of the flow solver grid.

Translation of a 2D drop

To study the convergence rate of the new algorithm for calculating flux densities we first use a prescribed velocity field. Consider the translation of a circle of radius 0.2, positioned initially at the center of a 1×1 domain. Two uniform velocity fields are prescribed: $\boldsymbol{U} = (1,0)$ and $(0,1)$. The domain is discretized into N number of uniform grids in each direction, where $N \in \{10, 20, 40, 80, 160, 320\}$. For each grid size, we use two time steps, such that the CFL number is 0.25 and 0.75 and run the test for one time step. To calculate $\tilde{\rho}$, we choose the density to be 1000 inside the circle, and 1 outside (note that the flow equations are not solved here and these are just numerical values for the test). We compare the flux densities to those obtained geometrically using Youngs' method [15]. These geometrical results, which are considered benchmark solutions, are obtained by initializing volume fractions using recursive local mesh refinement, and refining to 16 levels in interfacial cells. This yields the volume fractions to machine precision (further subdivisions do not change the values). We also use the exact normal vectors for interface reconstruction, which is required for the geometrical calculation of fluxes. Since the test is only run for one time step and there are no advection errors in volume fractions, the geometrical results converge to exact values with mesh refinement.

We calculate L_1, L_2, and L_∞ norms of the nor-

---

[2] First-order time discretization is for demonstration only; in practice the temporal discretization is second-order accurate or higher.

malized error of $\tilde{\rho}$, defined as

$$Err. = \frac{|\tilde{\rho}^{\text{Geom.}} - \tilde{\rho}^{\text{LS}}|}{\tilde{\rho}^{\text{Geom.}}}, \tag{19}$$

where the superscripts Geom. and LS denote flux densities obtained from the geometrical method and the new LS-based algorithm, respectively. We evaluated flux densities on horizontal and vertical surfaces shown in Fig. 4. For the u velocity component, flux densities across vertical and horizontal faces are denoted by $\tilde{\rho}_{f_{xx}}$ and $\tilde{\rho}_{f_{xy}}$, respectively; $\tilde{\rho}_{f_{yy}}$ and $\tilde{\rho}_{f_{yx}}$ are flux densities across horizontal and vertical surfaces of the v velocity component, respectively.

Figures 5 (a) and (b) show the errors of $\tilde{\rho}_{f_{xx}}$ for $\boldsymbol{U} = (1, 0)$ at various grid sizes and CFL numbers 0.25 and 0.75, respectively. L_1 (dashed line), L_2 (dotted line) norms exhibit second-order convergence, while the convergence of L_∞ (dashed-dotted line) norm is almost first-order. The same test was performed for $\tilde{\rho}_{f_{yy}}$ with $\boldsymbol{U} = (0, 1)$, and expectedly, the results (not shown) are exactly the same as those presented in Fig. 5. Next, the errors of $\tilde{\rho}_{f_{xy}}$ are presented in Figures 5 (c) and (d) for $\boldsymbol{U} = (0, 1)$. They exhibit the same order of accuracy. The errors associated with $\tilde{\rho}_{f_{yx}}$ for $\boldsymbol{U} = (1, 0)$ are also exactly the same as $\tilde{\rho}_{f_{xy}}$ errors. As can be seen, the time step size has almost no effect on the convergence rate of errors.

Collapse of a water column

This test case is based on the experiments conducted by Martin and Moyce [17]. Consider a 2-D water (fluid 1) column in air (fluid 2), shown in Fig. 6, which corresponds to the experimental results of [17]. The initial height and width of the column are both 5.715 cm. $\rho_1 = 1000\,\text{kg/m}^3$, $\rho_2 = 1.226\,\text{kg/m}^3$, $\mu_1 = 1.137 \times 10^{-3}\,\text{kg/ms}$, $\mu_2 = 1.78 \times 10^{-5}\,\text{kg/ms}$, $\sigma = 0.0728\,\text{N/m}$, $g = 9.81\,\text{m/s}^2$. Taking the initial column width, a, as the characteristic length, we have $\text{Fr} = u/\sqrt{ag} = 1$, $\text{We} = \rho u^2 a/\sigma = 440$, $\text{Re} = \rho u a/\mu = 37635$. The domain size is 40×10 cm, and is discretized by 256×64 uniform grid points.

The results obtained from the new method and the non-conservative formulation are illustrated in Figs. 6 and 7, respectively, at various non-dimensional times $\text{T} = t\sqrt{g/h}$. The non-dimensional spread rates for these results are shown in Fig. 8 for each formulation, and are compared with the experimental results [17]. As seen, the results of the non-conservative form shows slow spread rate, which is also evident in Fig. 7. This is due to errors in momentum transfer across the interface, which does not vanish by increasing the grid resolution. These errors manifest themselves as slowdown

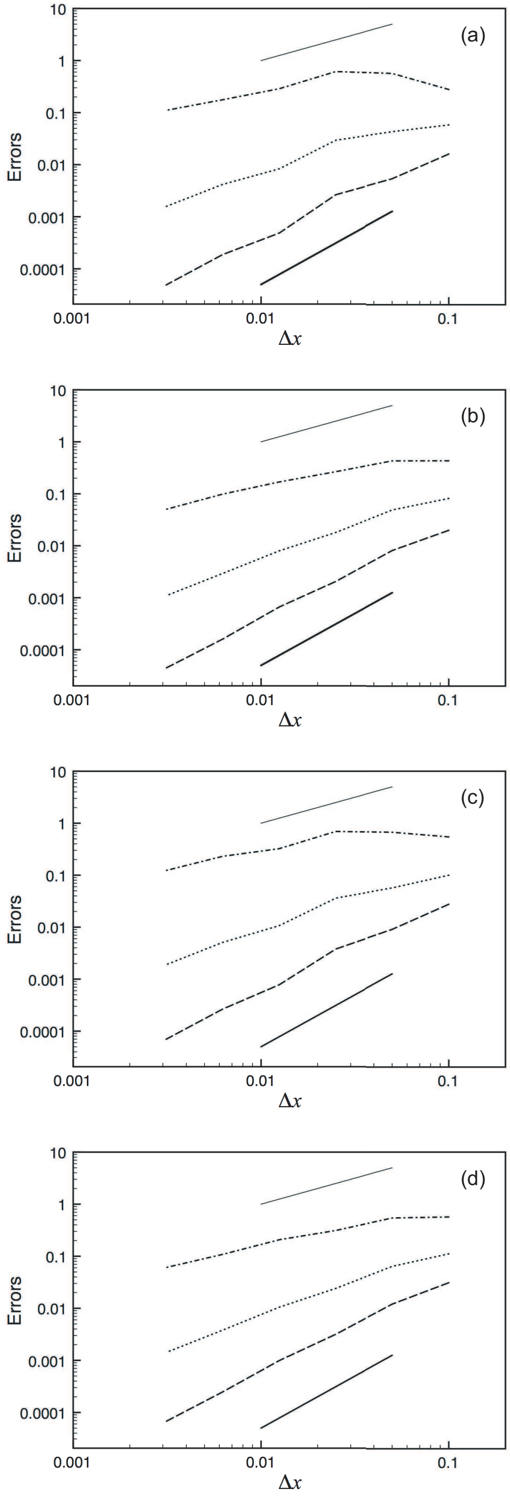

Figure 5. L_1 (dashed line), L_2 (dotted line) and L_∞ (dashed-dotted line) norms of normalized errors of $\tilde{\rho}_{f_{xx}}$ for CFL numbers (a) 0.25 and (b) 0.75, and for $\tilde{\rho}_{f_{xy}}$ for CFL numbers (c) 0.25 and (d) 0.75. Thin and thick lines represent first- and second-order accuracy, respectively.

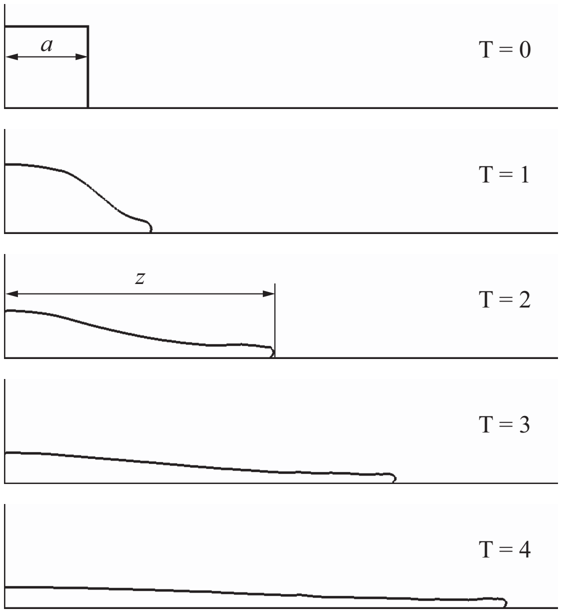

Figure 6. Collapse and spread of a water column in air simulated by the new method (density ratio: 815).

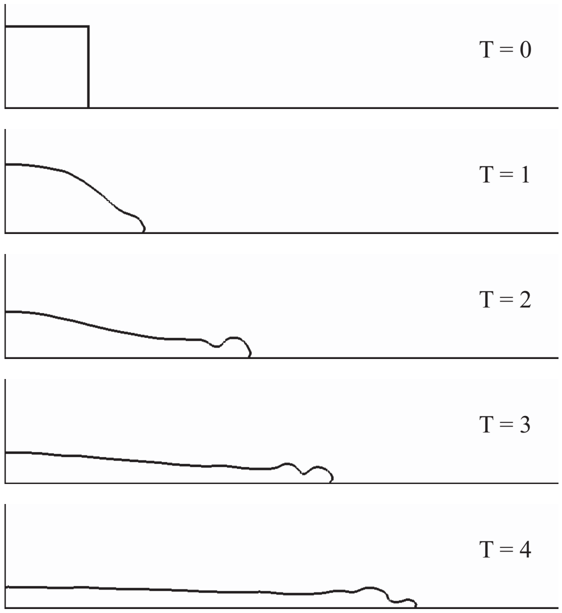

Figure 7. Collapse and spread of a water column in air simulated using the non-conservative formulation of the flow equation (density ratio: 815).

of the denser fluid (water) by the lighter fluid (air), which is obviously non-physical. The results of the new method does not show this non-physical behavior. The spread rate predicted by the new method matches the experimental spread rate very well.

As a numerical test case, we increased the density ratio to 10000 and ran the simulation with the new method. The new method is stable and able to handle this extreme case. The results (not shown) are almost identical to those presented in Fig. 6.

Oscillations of a viscous drop

As another validation test case, we study the oscillation of a viscous drop due to surface tension effect. The initial geometry is an ellipse of fluid 1, with semimajor axes of 0.3 and 0.2 in the x and y-directions, respectively, positioned at the center of a 1×1 domain, otherwise filled with fluid 2. $\rho_1 = 1000\,\mathrm{kg/m^3}$, $\rho_2 = 1\,\mathrm{kg/m^3}$, $\mu_1 = \mu_2 = 7.5 \times 10^{-3}\,\mathrm{kg/ms}$, $\sigma = 0.1\,\mathrm{N/m}$.

From the theoretical work of [18], the analytical decay of the total energy of the drop, denoted by E, is given by

$$E(t^*) = E_0 e^{-2n(n-1)\nu^* t^*} \tag{20}$$

where t^* is the dimensionless time, n is the mode number, $\nu^* = \mu_1/\sqrt{\rho_1 \sigma L}$ is a dimensionless parameter (Ohnesorge number), and L is the drop dimension.

We ran the simulation to $t^* = 200$ using both the new method and the non-conservative formulation. Figure 9 shows the decay of the total energy in time. The new method predicts the theoretical decay rate very well. However, in the results obtained with the non-conservative formulation, the decay rate is incorrectly faster due to numerical errors in momentum transport. Similar to the previous test case, in the results obtained from the non-conservative formulation, the heavier fluid seems to be incorrectly slowed down by the lighter fluid due to the above errors, which, again, do not vanish with grid refinement.

Liquid sheet breakup

We conclude with a test case that involves more complicated interface topologies. This test is based on the experimental work of [19]. Consider a 2D liquid ethanol (fluid 1) sheet of thickness $d = 508\,\mu\mathrm{m}$, injected at $U_1 = 5\,\mathrm{m/s}$ with shear air (fluid 2) flowing on its top and bottom surfaces at U_2. We consider two cases: $U_2 = 30$ and $60\,\mathrm{m/s}$. $\rho_1 = 789\,\mathrm{kg/m^3}$, $\rho_2 = 1.226\,\mathrm{kg/m^3}$, $\mu_1 = 1.39 \times 10^{-3}\,\mathrm{kg/ms}$, $\mu_2 = 1.78 \times 10^{-5}\,\mathrm{kg/ms}$, $\sigma = 0.022\,\mathrm{N/m}$ (ethanol properties are taken from [19]). Taking the relative velocity $U_r = U_2 - U_1$ as the characteristic velocity and d as the characteristic length, the dimensionless numbers based on ethanol properties are $\mathrm{Oh} = \mu/\sqrt{\rho \sigma d} = 0.015$, $\mathrm{We} = 11400$,

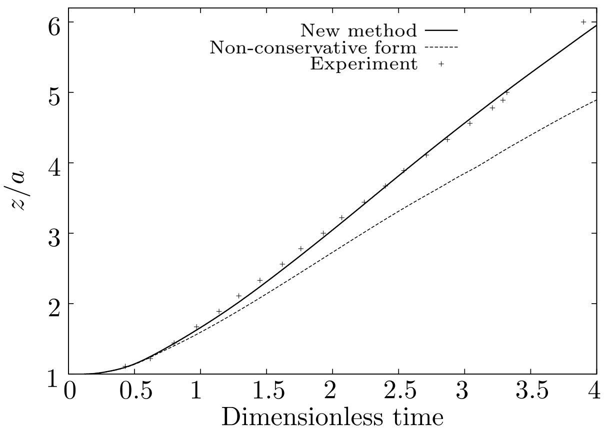

Figure 8. Non-dimensional front position of a collapsing water column in air (see Fig. 6) versus dimensionless time (density ratio: 815).

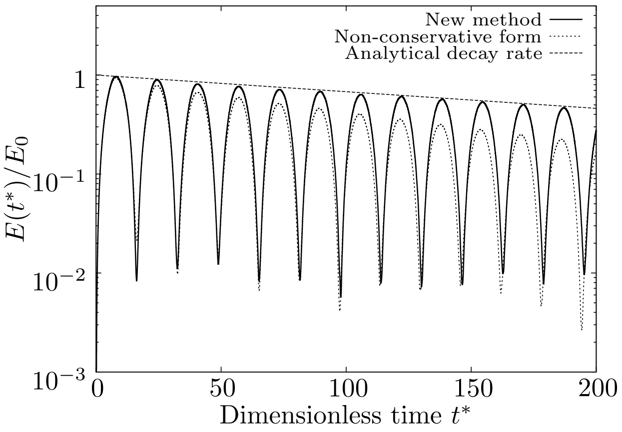

Figure 9. Decay of the total energy of an oscillating viscous drop (density ratio: 1000).

Re = 7200 for $U_r = 25\,\mathrm{m/s}$.

We used a 5 × 1 cm computational domain discretized by 2560 × 512 uniformly distributed grid points. The lip of the nozzle is included in the computational domain too, and is 500 (μm) long and 150 μm thick on each side.

Figure 10 depicts the results for $U_2 = 60\,\mathrm{m/s}$. The liquid interface experiences complex deformations: upon injection, the tip of the liquid sheet expands normal to the streamwise direction, because of a pair of vortices formed at the tip of the nozzle. We also observe that shortly after injection perturbations start to appear along fluid interfaces. Some of these perturbations grow and form ligaments that are extending from the surface of the liquid sheet (see the blowup images). The ligaments either collapse on the liquid surface and trap air in some cases, or detach from the liquid sheet and break up into smaller structures and droplets. We also observe that at some locations the liquid sheet breaks due to necking and forms isolated liquid structures. These structures further break up into smaller liquid structures and droplets.

As we decrease the air velocity to $U_2 = 30\,\mathrm{m/s}$ (Fig. 11), the liquid sheet extends further downstream before it starts to break up. We still observe perturbations and formation of ligaments, although at smaller numbers compared to Fig. 10. Interacting with shear air, the liquid sheet develops an oscillaiting sinusoidal shape, which has been observed in experiments [19, 20] too. Similar to the previous case, the liquid sheet breaks at some locations due to necking.

Note that the main focus in these simulations was on assessing the performance of the new method in complex interface deformations during the primary breakup of the liquid sheet, and not on resolving small liquid structures detaching from the liquid sheet, which eventually disappear as their sizes fall below the grid size.

To further illustrate the interaction of shear air with the liquid sheet, we present in Fig. 12 the dimensionless vorticity as well as the fluid interfaces shown in red, for $U_r = 25\,\mathrm{m/s}$. The relative velocity U_r and the liquid sheet thickness were used to obtain dimensionless vorticity. Shortly after injection, a pair of vortices at the nozzle exit, makes the tip of the liquid sheet to expand normal to the streamwise direction. Because the density ratio of fluids is large, the liquid sheet acts as a "solid" object for the air flow, and similar to flows around solid objects, here, we see vortices being shed off the liquid sheet. Furthermore, as the liquid sheet deforms and its surface develops curves, vortices start to form through interaction of air flow with the cavities on the liquid sheet surface. Finally, we also see the wake of air flow around small liquid structures and droplets that were broken off the liquid sheet.

Conclusions

We presented a level set-based method to model interfacial flows with high density ratios. Following the ideas presented in [3, 4], in this method, mass and momentum are transported consistently by using the same flux densities. Furthermore, the interface location (i.e. density discontinuity) is taken into account in the calculation of flux densities. The "input" of this method is a signed distance function to the interface, and therefore, this method can be easily employed by any level set-based flow solver, regardless of the method used to solve for the evolution of the level set function. We assessed the performance of the algorithm in a flow solver and used a variety of test cases in which the density ratio ranges between 650 to 10000. The method is accurate and stable even at very high density ratios. We also showed that a flow solver that transports momentum in a non-conservative form (without any proper treatment for flows with large density ratio) yields inaccurate and non-physical results due to numerical errors that are more pronounced at high density ratios.

References

[1] C. W. Hirt and B. D. Nichols. *J. Comput. Phys.*, 39:201–225, 1981.

[2] R. Scardovelli and S. Zaleski. *Annu. Rev. Fluid Mech.*, 31:567–603, 1999.

[3] M. Rudman. *Int. J. Numer. Meth. Fluids*, 28:357–378, 1998.

[4] M. Bussmann, D. B. Kothe, and J. M. Sicilian. *Proceedings of ASME 2002 Fluids Engineering Division Summer Meeting*, Montreal, Canada, 2002.

[5] I.K. Park, H.K. Cho, H.Y. Yoon, and J.J. Jeong. *Nucl. Eng. Des.*, 239:2365–2371, 2009.

[6] M. Sussman, P. Smereka, and S. Osher. *J. Comput. Phys.*, 114(1):146–159, 1994.

[7] S. Osher and R. Fedkiw. *Level Set Methods and Dynamic Implicit Surfaces.* Springer-Verlag, New York, 2003.

[8] X. Y. Hu, B. C. Khoo, N. A. Adams, and F. L. Huang. *J. Comput. Phys.*, 219(2):553–578, 2006.

[9] O. Desjardins, G. Blanquart, G. Balarac, and H. Pitsch. *J. Comput. Phys.*, 227(15):7125–7159, 2008.

[10] O. Desjardins, V. Moureau, and H. Pitsch. *J. Comput. Phys.*, 227(18):8395–8416, 2008.

[11] O. Desjardins and H. Pitsch. *J. Comput. Phys.*, 228(5):1658–1677, 2009.

[12] R. B. Bird, W. E. Stewart, and E. N. Lightfoot. *Transport Phenomena.* John Wiley and Sons, New York, 2nd edition, 2002.

[13] L. D. Landau and E. M. Lifshitz. *Fluid Mechanics.* Pergamon Press, Oxford, 2nd edition, 1987.

[14] M. Kang, R. Fedkiw, and X.-D. Liu. *J. Sci. Comput.*, 15(3):323–360, 2000.

[15] D. L. Youngs. In K. W. Morton and M. J. Baines, editors, *Numerical Methods for Fluid Dynamics*, pp. 273–285, Academic Press, New York, 1982.

[16] M. Raessi. *Annual Research Briefs-2008*, pp. 467–478. Center for Turbulence Research, Stanford, CA, 2008.

[17] J.C. Martin and W.J. Moyce. *Philos. Trans. R. Soc. London, Ser. A*, 244:312–324, March 1952.

[18] B. M. Rush and A. Nadim. *Eng. Anal. Boundary Elem.*, 24:43–51, 2000.

[19] B.E. Stapper, W.A. Sowa, and G.S. Samuelsen. *J. Eng. Gas Turbines Power*, 114:39–45, 1992.

[20] J. Park, K.Y. Huh, X. Li, and M. Renksizbulut. *Phys. Fluids*, 16(3):625–632, 2004.

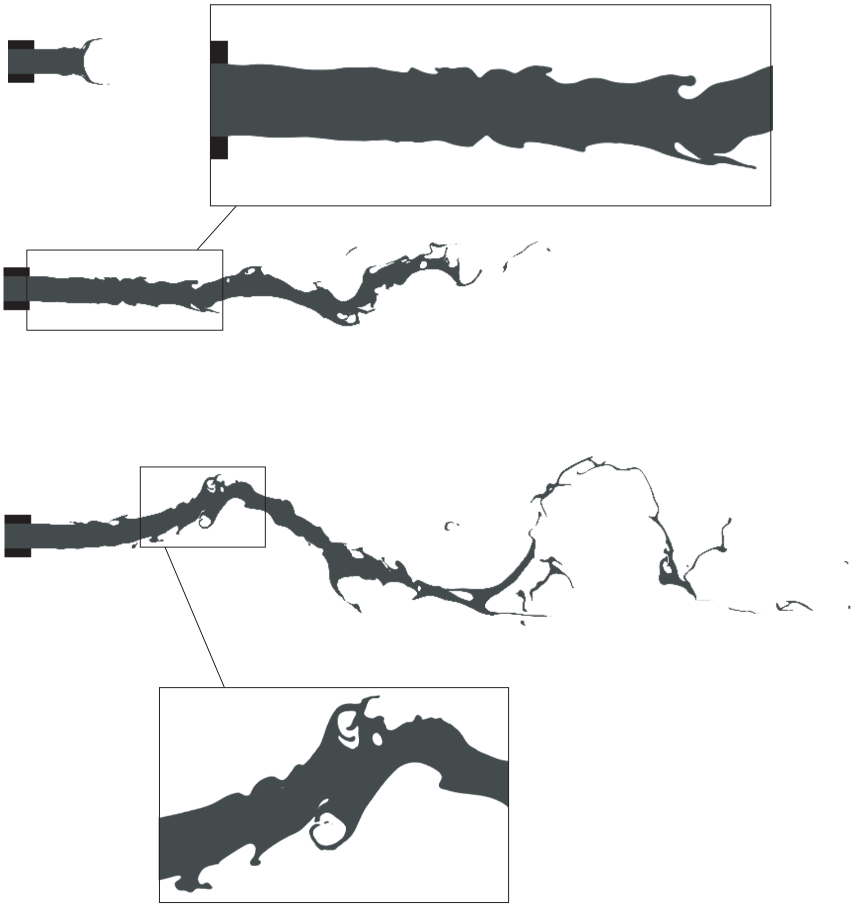

Figure 10. Breakup of ethanol in shear air flow ($U_r = 55\,\mathrm{m/s}$, density ratio: 650).

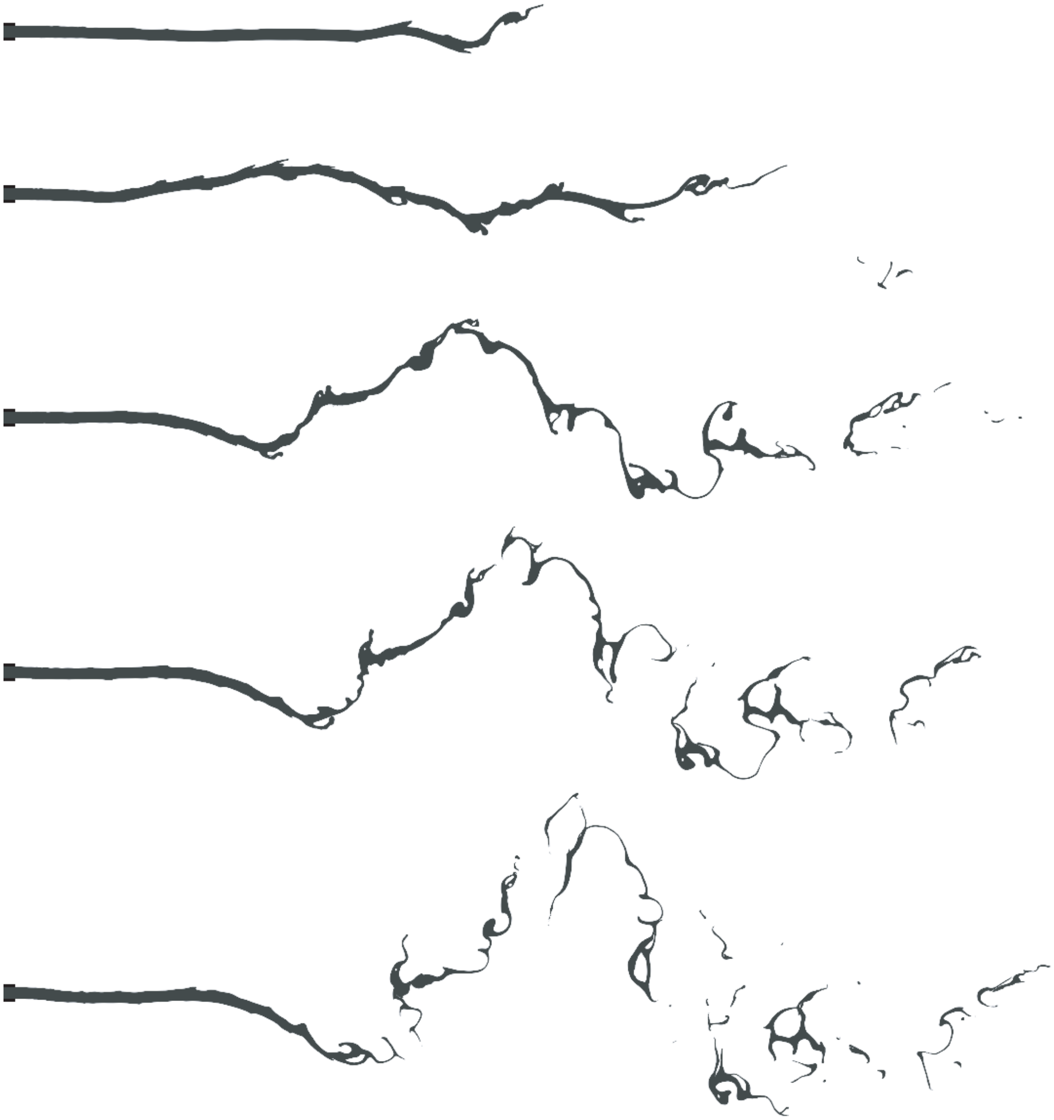

Figure 11. Breakup of ethanol in shear air flow ($U_r = 25\,\mathrm{m/s}$, density ratio: 650).

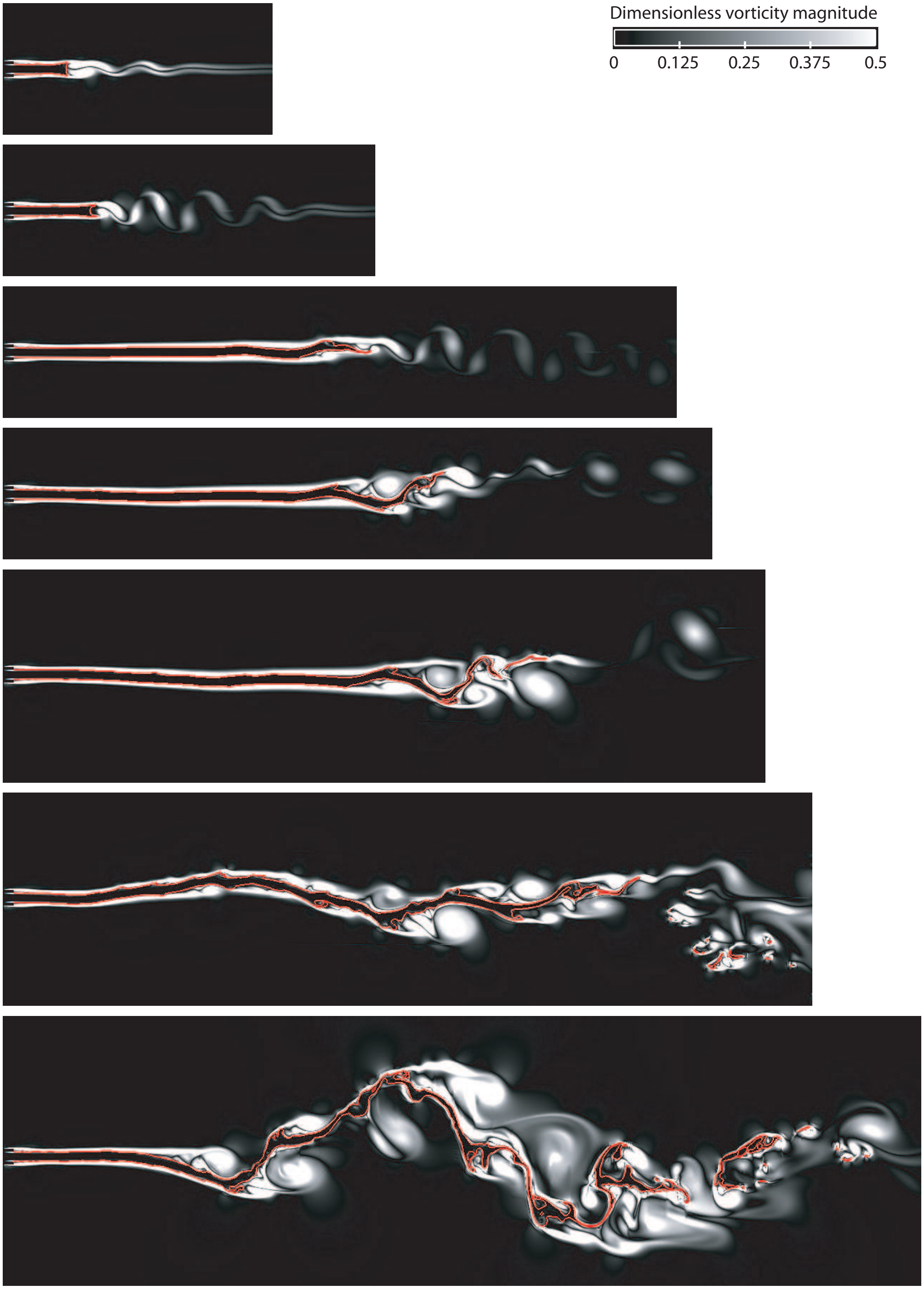

Figure 12. Dimensionless vorticity during breakup of ethanol in shear air flow. Fluid interfaces are shown in red. ($U_r = 25\,\mathrm{m/s}$, density ratio: 650).