A ROBUST FRONT TRACKING METHOD: VERIFICATION AND APPLICATION TO SIMULATION OF THE PRIMARY BREAKUP OF A LIQUID JET∗

WURIGEN BO†, XINGTAO LIU†, JAMES GLIMM†, AND XIAOLIN LI†

Abstract. This paper presents a method for simulating compressible two-phase flow by combining the best features of a front tracking method (FT) and a ghost fluid method (GFM). In contrast to GFM, a Riemann problem is solved to find the ghost states. And in contrast to FT, the front states used in the Riemann problem are not dynamic variables but are obtained by extrapolation from the interior (grid) states. Pressure jumps associated with surface tension forces are modeled in the Riemann problem. This method handles surface tension forces in a sharp way and avoids artificially spreading surface tension forces over the computational grid as used in continuous surface models. To handle the topological bifurcations of a three-dimensional (3D) surface mesh in the FT, an improved locally grid-based method (LGB) is proposed. The method is robust and minimizes the numerical mass diffusion due to interface reconstruction. The performance of the method is assessed from a broad set of test problems including compressible Kelvin–Helmholtz instabilities, parasitic currents, drop oscillation, bubble-shock interaction, and Rayleigh instabilities. The proposed new method is shown to be comparable to either of its constituent methods by themselves. The advantage of the new method lies in the removal of late time instabilities associated with both of the constituent methods when applied to the 3D simulation of a high speed jet.

Key words. front tracking method, ghost fluid method, primary jet breakup

AMS subject classifications. 76N15, 76T10, 76M25

DOI. 10.1137/10079135X

1. Introduction. For the simulation of two-phase flow, especially in the case of a strong density contrast between the phases, a special treatment of the interface is required. Front tracking is one of the popular methods. We improve on this method in three critical ways. For the coupling of the interface to the interior states, we combine ideas from our own method with those of others to propose a new coupling framework. We show that it is comparable to our previous method and to methods of others on test problems; we prefer it because of improved late time stability properties. We improve the method for the bifurcation of the interface, and we introduce an improved algorithm for the surface normal and curvature to improve the accuracy of the computation. We illustrate the new algorithm with a simulation of the primary breakup of a high speed jet. For this problem details will be published separately.

To evolve the liquid-gas interface, many methods have been proposed; of these, volume of fluid (VOF), level set, and front tracking methods (FTs) are the most popular. A complete review is beyond the scope of this paper. Readers are referred to the papers of Sethian [38], Scardovelli and Zaleski [37], Glimm et al. [9, 11], and Tryggvason et al. [44]. A comparison of these three methods [6] shows the strengths of front tracking.

∗Submitted to the journal's Methods and Algorithms for Scientific Computing section April 7, 2010; accepted for publication (in revised form) April 11, 2011; published electronically July 14, 2011. This work was supported by the Department of Energy awards DEFC5208NA28614 and DEFC0206ER25770 and the Army Research Office award W911NF0910306.

†Department of Applied Mathematics and Statistics, SUNY at Stony Brook, Stony Brook, NY 11794-3600 (bowrg@ams.sunysb.edu, liuxt12@ams.sunysb.edu, glimm@ams.sunysb.edu, linli@ams.sunysb.edu).

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.
We use an FT to solve the free surface flow. The FT is implemented in our code *FronTier*. Front tracking has many advantages for problems dominated by a geometrically complex and dynamically moving interface. It was used to simulate the Rayleigh–Taylor instability. Impressive results [26, 27, 23], including agreement with experiments in the overall growth rate as defined by the mixing growth parameter $\alpha$, were obtained. Since moving surfaces are tracked by marker particles, FTs are able to accurately model an absence of mass diffusion across the interface for immiscible fluids and limited physically correct mass diffusion across a concentration isosurface for miscible fluids [6], in contrast to capturing methods. They also differ from marker particle methods in that particles are located only on the interface rather than in a volume region near the interface. Geometrical information, such as the surface normal and curvature, is easily computed in FTs, as are surface related physical processes such as surface diffusion, surface tension, phase transition, mass transfer across a phase boundary, and surface mediated chemical reactions, due to the explicit representation of the interface by its own mesh.

In the FT, each point on the interface has a left state and a right state which describe the flow states on it. Riemann solutions are constructed at each interface point to propagate the solution in the normal and tangential directions. A finite difference method is applied to solve the flow states on each side of the interface. The stencil states on the other side of the interface are extrapolated from the interface states. Terashima and Tryggvason recently proposed a front tracking/ghost fluid method (FT/GFM) for compressible flow [43]. No interface states are needed in this method. Interface points are propagated by an interpolated flow velocity. The ghost fluid method (GFM) is used to couple the compressible flow solver with the FT. For large density differences across an interface, such as gas-water flow, a modified GFM is adopted [8]. This approach, however, was found to be sensitive to pressure oscillations for an air-water interaction [18, 24].

In this paper, we couple the FT [6] with a GFM. The interface states are reconstructed from the interior states, and the normal velocity is solved from a Riemann problem. The ghost states are constructed by the interface interaction method of Hu and Khoo [18]. The surface tension force is modeled as a pressure jump across the contact wave. Compared with our previous FT, the proposed method avoids solving Navier–Stokes equations on the interface, which is computationally expensive when the interface is complex. Since the surface tension force is modeled as a pressure jump in Riemann problems, the method avoids artificially spreading surface tension forces over the computational grid, as is done in Terashima and Tryggvason’s FT/GFM.

We also improve the method for topological bifurcations. This improvement is needed for our application to the primary breakup of a high speed jet, as the problem is dominated by many bifurcations of the front topology. Three methods are implemented in *FronTier* to handle topological bifurcations. A grid-free method (GF) [10] is accurate but is prone to logical errors. A grid-based method (GB) [11] is robust but suffers from excessive interpolation and smoothing errors. A locally grid-based method (LGB) [6] combines the advantages of both but is still not robust enough for the large scale simulation of a turbulent jet breakup. In this paper, we propose an improved LGB. The method robustly handles the topological bifurcations of an interface with millions of triangles during a turbulent jet breakup simulation in a Bluegene/L computer with thousands of cores.

The normal and curvature calculations play an important role in the FT: the accuracy of the surface point propagation, the ghost state construction, and the surface
tension forces depend on the accuracy of normal and curvature calculations. We implement a local polynomial approximation to obtain normal vectors and curvature tensors [19]. The method is proved to be more accurate than our previous one [6].

In section 3, the FT/GFM approach to the discretization of the compressible Navier–Stokes equations is described, and an improved LGB is proposed to handle topological bifurcations in three-dimensional (3D) surface meshes. Section 4 will concentrate on simple verification cases: the parasitic currents, droplet oscillation, shock-bubble interaction, and the compressible Kelvin–Helmholtz and Rayleigh instabilities. To illustrate the robustness of our interface algorithm, we present results from the simulation of the turbulent atomization of a high speed liquid jet in section 5.

2. Governing equations. The governing equations are the 3D compressible Navier–Stokes equations written in conservative form,

\[
\frac{\partial U}{\partial t} + \frac{\partial E}{\partial x} + \frac{\partial F}{\partial y} + \frac{\partial G}{\partial z} = \frac{\partial E_v}{\partial x} + \frac{\partial F_v}{\partial y} + \frac{\partial G_v}{\partial z},
\]

with

\[
U = [\rho, \rho u, \rho v, \rho w, E]^T,
\]

\[
E = [\rho u, \rho u^2 + p, \rho uv, \rho uw, (E + p)u]^T,
\]

\[
F = [\rho v, \rho uv, \rho v^2 + p, \rho vw, (E + p)v]^T,
\]

\[
G = [\rho w, \rho uw, \rho vw, \rho w^2 + p, (E + p)w]^T,
\]

\[
E_v = [0, \tau_{xx}, \tau_{xy}, \tau_{xz}u + \tau_{xy}v + \tau_{xz}w]^T,
\]

\[
F_v = [0, \tau_{xy}, \tau_{yy}, \tau_{yz}u + \tau_{yy}v + \tau_{yz}w]^T,
\]

\[
G_v = [0, \tau_{xz}, \tau_{yz}, \tau_{zz}u + \tau_{yz}v + \tau_{zz}w]^T,
\]

where \(\tau_{ij}\) are the shear stress and \(E\) is the total energy per unit volume defined by

\[
E = \rho e + \frac{1}{2} \rho (u^2 + v^2 + w^2);
\]

here \(e\) is the specific internal energy. In addition to (2.1), the stiffened gas equation of state is used:

\[
p = (\gamma - 1)\rho e - \gamma p_{\infty}.
\]

Here the constants \(\gamma\) and \(p_{\infty}\) are characteristic parameters of material. This equation reduces to the ideal gas equation of state when \(p_{\infty}\) is zero. In all of the following simulations except the bubble-shock interaction test, the liquid is \(n\)-heptane with \(\gamma = 3.19, p_{\infty} = 3000\), and the gas is air with \(\gamma = 1.4, p_{\infty} = 0\). Across the interface, the pressure is not continuous, and the jump condition can be written as

\[
[p]_r = \sigma \kappa + 2[\mu]_r n^T \cdot \nabla u \cdot u,
\]

where \(\sigma\) is the surface tension, \(\kappa\) is the interface curvature, \(n\) is the interface normal, and \([p]_r = p_l - p_r, [\mu]_r = \mu_l - \mu_r\) are the jumps of the pressure and the dynamic viscosity across the interface.

3.1. Ghost fluid method. Hu and Khoo’s interface interaction method [18] is used to account for the density jump and the pressure jump due to the surface tension force, while the continuum surface force approach [3] is used to model the jump in viscous stress.

For FTs, two fluids are separated by an interface. In a finite difference implementation, the states on the other side of the interface need to be reconstructed to avoid numerical oscillations. The ghost cell method was first proposed by Glimm, Marchesin, and McBryan [13] to handle the discontinuous variables across the interface in the FT. Fedkiw used a GFM [7] in the level set method with discontinuous variables extrapolated to the ghost node on the other side of the interface and the continuous variables copied from the adjacent fluid states. Hu and Khoo [18] found this approach to be sensitive to small perturbations for air-water interaction, and they proposed an interface interaction method to obtain the states at ghost nodes.

Hao and Prosperetti [14] combined an FT and a GFM to solve 3D free-surface flow problems with an incompressible liquid and a compressible gas. Terashima and Tryggvason [43] coupled a GFM and an FT to solve fully compressible liquid-gas systems. We solve the compressible Euler or Navier–Stokes equations with an FT and a GFM. We combine these ideas as follows. First, we use Hu and Khoo’s interface interaction method [18] to obtain ghost states. Second, we reconstruct states on the interface from ghost states and regular grid cell states. We propagate interface points using interface states and a local Riemann problem solved there. These modifications are important to achieve late time stability of the interface.

For simplicity, we first describe the algorithm in one dimension. Suppose the interface lies between the node \( j \) and \( j + 1 \). Let the states at the node \( j \) and \( j + 1 \) be \( W_l = (\rho_l, u_l, p_l) \) and \( W_r = (\rho_r, u_r, p_r) \). A Riemann problem with \( W_l \) and \( W_r \) as its data is first solved, and then the left ghost velocity \( u_g^l \) and pressure \( p_g^l \) of node \( j \) can be obtained from the left states of the contact wave. The ghost node entropy is copied from \( W_l \), i.e., \( s_g^l = s_l \). The right ghost node state can be obtained in the same way. This method is very similar to Algorithm A in [18]. The difference is that we construct ghost states by solving an exact Riemann problem instead of using the method of characteristics. When there is surface tension on the interface, the Riemann problem for the ghost states is subject to the constraint

\[
p_l^0 - p_r^0 = 2\sigma \kappa,
\]

where \( \sigma \), \( \kappa \) are the surface tension coefficient and the mean curvature. The viscous stress terms in (2.2) are modeled by the continuum surface force approach [3] and do not appear in (3.1). In this way, the surface tension force is implemented in a sharp way, more consistent with the GFM than with the continuous surface model. In two and three dimensions, we adopt the method in [18] and construct ghost states in the normal direction of the interface.

We then show that our method is suitable to solve high acoustic impedance mismatch (AIM) interface problems such as gas-water interaction. When solving Euler equations numerically for high AIM interface, it is found that numerical oscillations may cause the breakdown of computation [25, 30]. Liu, Khoo, and Wang [24] proposed that an incorrect or inconsistent Riemann wave provided by the GFM Riemann problems in the respective real fluid may cause the numerical oscillations. They also proposed the following method to check the consistency of the two GFM Riemann problems.
Assume the real and ghost states at the node \(j\) and \(j+1\) to be \(W_l, W_r\) and \(W^g_l, W^g_r\). Solve two GFM Riemann problems: The first uses \(W_l, W^g_l\) as its data, and the second uses \(W^g_r, W_r\) as its data.

Check whether the following two conditions are satisfied:

1. The Riemann wave in the real fluid side for both GFM Riemann problems is initially consistent with that for the same side, as for the original Riemann problem.
2. The middle states according to both GFM Riemann problems for the pressure and velocity are sufficiently close.

We check the consistency of our GFM Riemann problems on a \(p-u\) plane using the above conditions. The left and right real states \((p_l, u_l)\) and \((p_r, u_r)\) are two points on the \(p-u\) plane. The solution of the original Riemann problem is the intersection point of the Riemann wave curves passing through \((p_l, u_l)\) and \((p_r, u_r)\); we denote the intersection point as \((p_g, u_g)\). Since the left and right ghost states are given by the same point \((p_g, u_g)\) on the \(p-u\) plane in our method, conditions (1) and (2) are satisfied. It is worth mentioning that Liu, Khoo, and Wang [24] show that the two conditions can be violated by both the original GFM [7] and the modified GFM [8]. Therefore, they may generate unphysical waves near the interface.

The accurate computation for differential quantities of an interface, such as normal vectors and curvature tensors, is necessary for the method to track the motion of the interface for a long time. We use the method proposed by Jiao and Zhao [19] to obtain the normal vector and the mean curvature. The method is proved to be second- and first-order accurate for the normal vector and the curvature tensor [19].

### 3.2. Interface propagation

The interface points are propagated in a Lagrangian fashion by integrating

\[
\frac{d\mathbf{x}_f}{dt} = \mathbf{V}_n(\mathbf{x}_f),
\]

where \(\mathbf{x}_f\) is the point on the interface. \(\mathbf{V}_n(\mathbf{x}_f)\) is the normal velocity at \(\mathbf{x}_f\), which in Terasshima and Tryggvason’s method is interpolated from the fixed grid points to the interface point by use of bilinear interpolation [44]. We adopt a modification of this approach, with a different method to obtain the normal velocity, which appears to have improved stability for a complicated interface with a large surface tension. For each point on the interface, we interpolate the fluid states on the left and right of the point using a bilinear (or trilinear in three dimensions) interpolation; see Figure 1.

With the left and the right states as data, we solve a Riemann problem along the normal direction to obtain the normal speed \(\mathbf{V}_n(\mathbf{x}_f)\).

### 3.3. Time step

The time step \(\Delta t\) at time \(t^n\) is determined by restrictions due to the CFL condition, viscosity, surface tension, and interface velocity [41],

\[
(\text{3.3}) \quad \Delta t = \text{CFL} \cdot \min \left\{ \min_{i,j,k} \left( \frac{\Delta x}{|u^n| + c^n}, \frac{\rho^n \Delta x^2}{\mu}, \sqrt{\frac{\rho_l + \rho_r}{8\pi\sigma}} \frac{\Delta x^{3/2}}{p} \right), \min_{p} \frac{\Delta x}{|u^n|} \right\},
\]

where the subscripts \(i, j, k\) and \(p\) represent the indices of cell centers in the flow field and the interface points, respectively, and \(c^n\) is the sound speed at time \(t^n\). \(\text{CFL}\) is a stability coefficient \(\text{CFL} = 0.75\) for all our simulations.
3.4. Interface redistribution and reconstruction. As the interface expands or contracts, the finite resolution of the interface leads to a poor distribution of interfacial element sizes, which can in turn lead to the overdevelopment of interfacial instabilities. Therefore, we periodically reinterpolate the interface points. In 3D space, all triangles are sorted according to their size and aspect ratios. Small triangles are merged with neighbors, while large triangles are subdivided. Bad aspect ratio triangles are flipped along their diagonals to improve the aspect ratios. Details of the algorithm were given in [12]. For all the simulations in this paper, we have found that redistributing the interface mesh every five steps is enough to maintain the mesh quality.

Topological bifurcation occurs frequently during the primary breakup of a liquid jet. Triangles on the surface mesh must be allowed to reconnect with each other when they are tangled. Three methods have been implemented to resolve the tangled interface in Frontier. A grid-free method (GF) propagates the interface without reference to the underlying rectangular grid, except for occasional redistribution to achieve uniformity of triangles. The tangled interface is resolved by methods of interface surgery and topological consistency. It is very accurate but the untangling step is prone to logical errors and loss of robustness. At the other extreme is a grid-based method (GB) [11], which reconstructs the interface at each time step based on the crossings of the propagated interface with the edges of the rectangular grid. Given these crossing points, a set of templates fills in the interior of each grid cell with a piecewise linear surface to represent the interface. Since the reconstructed interface respects the crossings, it must agree with neighbor cells at these crossings. A further step ensures that it agrees on faces, and so the global interface reconstructed is well defined. This GB interface is overly smoothed and of low quality. The resolution of this choice between the robustness of GB and the quality of GF is a locally grid-based method (LGB), which is GB only in space time regions of a tangled interface and is GF elsewhere.

The LGB thus identifies some bad triangles of the propagated interface. It isolates these and preserves the intersections of the interface with the grid cell edges to allow...
A ROBUST FRONT TRACKING METHOD

Fig. 2. Two examples of invalid cases.

a GB reconstruction locally near the bad region. Triangles neighboring the bad region are removed, and so there is a gap separating the good part of the interface from the reconstruction of the bad part of the interface. The major step is to reseal this gap.

In order to keep the triangular mesh topologically valid, we impose two constraints on the triangular mesh:
(1) One side of a triangle connects with at most one triangle.
(2) One nonboundary vertex has only one associated list of triangles formed by linking successively adjacent triangles.

Figure 2 shows two examples of invalid cases. We describe the main procedures of our improved LGB reconstruction method. The moving interface is represented by a surface $S$ which is comprised of triangles. $B$ is a 3D rectangular box. $S$ is tangled inside box $B$. Let $S_1$ be a subsurface of $S$ which is comprised of all the triangles inside or intersecting $B$.

(a) $S_1$ is removed from $S$. We rename $S$ to stand for $S\setminus S_1$. Polygonal holes form on $S$ after this step.
(b) If a vertex is common for two or more polygonal holes, the vertex is separated so that the polygonal holes are merged. This step is repeated until there are no common vertices for the polygonal holes. We use $P_1$ to denote the set of the sides of the polygonal holes after this step.
(c) $S_1$ is reconstructed in $B$ using the GB. We use $S_2$ to denote the reconstructed surface. The boundary of $S_2$ consists of polygons due to the GB reconstruction [11]. The set of the sides of the polygons is denoted by $P_2$.
(d) A new surface $S_3$ is constructed to connect $S$ and $S_2$ using the following method: For any two sides $v_1v_2 \in P_1$ and $v_1'v_2' \in P_2$, we make a pair of triangles $v_1v_2v_1'$ and $v_2'v_1'v_2$ and add them to $S_3$ if the following two constraints are satisfied:
   (i) There exist three vertices out of $v_1, v_2, v_1', v_2'$ which lie in the same triangle from $S_1$.
   (ii) If triangles $v_1v_2v_1'$ and $v_2'v_1'v_2$ are added to $S_3$, the connected surface $S \cup S_2 \cup S_3$ satisfies constraints (1) and (2).

This step is repeated until no new pair of triangles can be added to $S_3$.
(e) $S$, $S_2$, and $S_3$ are connected to form a new surface $S_4$. For each polygonal hole on $S_4$, if two consecutive sides are two sides of a triangle, then the triangle is removed from $S_4$.
(f) Each polygonal hole on $S_4$ is sealed by a constrained triangulation considering constraints (1) and (2).
Figure 3 depicts the substeps for the method; each frame corresponds to the identically labeled step of the method. Since the surface in each substep satisfies constraints (1) and (2), the reconstructed surface \( S_4 \) also satisfies constraints (1) and (2). In this way, the method handles topological bifurcations robustly. This is the major advantage of the improved LGB.

4. Results. To assess the capability of the proposed method in multiphase flow, several test cases involving surface tension and topological bifurcations are studied. We compare the results from the present method to those from our previous FT [6] and the method of [43] (FT/GFM).

4.1. Kelvin–Helmholtz instability. The Kelvin–Helmholtz instability arises when two fluids are separated by an interface across which the tangential velocity is discontinuous. Such a flow is unstable under a sinusoidal perturbation of the interface. The Kelvin–Helmholtz instability plays an important role in the primary breakup of a liquid jet especially for high pressure combustors [36, 35]. The evolution of the two-dimensional (2D) Kelvin–Helmholtz instability has been studied in theory [1, 32] and through numerical simulations [17, 42, 15].

Here we investigate numerically the Kelvin–Helmholtz instability using the compressible Navier–Stokes equations. The Reynolds numbers are very large in the simulations, so that the initial growth rates of sinusoidal waves are well predicted by the inviscid linear stability theory. We first consider the Kelvin–Helmholtz instability in the linear (small amplitude) regime. In a 2D domain \([-1, 1] \times [-2, 2] \), two fluids are initially separated by a perturbed interface. The fluid below the interface is moving with velocity \( U_1 \), and the fluid above is moving with velocity \( U_2 \). The density of the bottom and top fluids is \( \rho_1 \) and \( \rho_2 \), respectively. The computational domain has periodic horizontal boundaries and rigid, no-slip walls at the top and the bottom. The domain length is the wavelength \( \lambda \). The interface is perturbed by a sinusoidal wave

\[ A = \text{Im}[A_0 e^{ik(nt+x)}], \]

where \( k = 2\pi/\lambda \) is the wavenumber, \( n \) is the growth rate, and \( A_0 \) is the initial amplitude of the perturbation. For a small initial amplitude, the growth rate can be predicted by the linear stability theory. For compressible fluids, we have the dispersion
relations \[32\]

\[
(4.2) \quad \rho_1(n + U_1)^2 \sqrt{1 - \left(\frac{n + U_1}{c_1}\right)^2} \rho_2(n + U_2)^2 \sqrt{1 - \left(\frac{n + U_2}{c_2}\right)^2} = \sigma k^2,
\]

where \(\sigma\) is the surface tension coefficient and \(c_i\) is the sound speed for fluid \(i\). As \(c_1\) and \(c_2\) approach infinity, the denominators on the left-hand side of the equality go to unity and (4.2) converges to the dispersion relation for incompressible fluids \[4\],

\[
(4.3) \quad \rho_1(n + U_1)^2 + \rho_2(n + U_2)^2 = \sigma k^2.
\]

Here we introduce a dimensionless time

\[
(4.4) \quad \hat{t} = tk\Delta U, \quad \Delta U = |U_2 - U_1|.
\]

All times used below are given in these dimensionless units. The Weber number is defined as

\[
(4.5) \quad We = \frac{\rho_2\Delta U^2\lambda}{\sigma}.
\]

We investigate Kelvin–Helmholtz instability with density ratio \(\rho_1/\rho_2 = 1\) and \(\rho_1/\rho_2 = 100\). In our simulations, the initial amplitude is taken to be 2.5% of the wavelength. A normal mode analysis is applied to the Euler equations to obtain the initial velocity, pressure, and density. A detailed derivation was given in \[32\]. The initial interface is given by a grid-based construction; i.e., for a \(40 \times 80\) mesh, the initial curve contains 40 points.

We first compare the growth rates from our simulations with that from the dispersion relation. We then study the convergence of the growth rate. We finally study the late time behavior of Kelvin–Helmholtz instability up to a time \(\hat{t} = 60\).

In Figure 4, the computed initial growth rate is plotted versus the inverse Weber number for density ratio 100. The growth rate is normalized by the maximal growth.
rate. Here the Mach number is $M = 0.2$. The solid line is the growth rate predicted by (4.2). The dotted line is the growth rate predicted by (4.3). Open circles represent results from simulations. An $80 \times 160$ mesh is used in the simulation. The growth rate agrees with the linear stability analysis for compressible fluid. Due to the small Mach number, the growth rates predicted by the compressible and incompressible analysis show little difference.

In Table 1, the relative errors of the growth rate under different mesh resolutions are given for density ratio $\rho_1/\rho_2 = 1$. Since the interfaces roll up after $\tilde{t}_0 = 6$, the linear stability analysis fails after this time. The following averaged growth rate is computed over the linear growth regime:

\begin{equation}
\tilde{n} = \frac{1}{\tilde{t}_0} \int_0^{\tilde{t}_0} n(t) d\tilde{t}.
\end{equation}

First-order convergence is observed for all three FTs. The relative error of the present method is comparable to the method in [43] but is larger than our previous FT.

We also study the long-time evolution of the Kelvin–Helmholtz instability for Weber number $We = 6$ and density ratio $\rho_1/\rho_2 = 1$. This case is also studied by Tauber, Unverdi, and Tryggvason [42]. Figure 5 shows the temporal evolution of the interface from our present method. As two fingers continue to grow and extend to the opposite fluids, viscous effects increase the thickness of the shear layer, and the fingers are pulled back by surface tension forces. Similar behavior has been observed in the simulations of Tauber, Unverdi, and Tryggvason [42] and Herrmann [15] for incompressible flows.

4.2. Parasitic currents. In numerical simulations, the imbalance of the grid representation of pressure and surface tension force leads to unphysical flow near the interface. One example is a static spherical drop in zero gravity. For this example, the fluid velocities remain zero for all times in physics. Numerical calculation may...
generate small velocity fluctuation. To assess the level of these parasitic currents, a static 2D circle with a diameter of $D = 0.5cm$ placed in the center of a $1 \times 1$ unit box resolved by a $25 \times 25$ grid is computed. At the initial time, there are 78 points on the circle. We perform a simulation with Laplace number $La = \sigma \rho D / \mu^2 = 10,000$. The levels of parasitic currents are measured by the capillary number $Ca = U_{max} \mu / \sigma$, where $U_{max}$ is the magnitude of the maximum velocity in the simulation. Figure 6 shows the evolution of capillary numbers. In the initial time, the capillary number reaches a spike and then drops down to a constant. Earlier simulations by Tryggvason et al. [44] and Shin and Juric [39] also show that $Ca \sim O(10^{-4})$ up to $La \sim O(10^5)$ by their incompressible FTs. $La \sim O(10^4)$ is the limit of our method since our simulations become unstable after thousands of steps with $La \sim O(10^5)$.

4.3. Drop oscillation. The oscillation of drops in zero gravity is a classical test case of the accuracy of numerical schemes for two-phase flows [28]. A small amplitude oscillation is applied to a spherical interface between two fluids initially at rest. Due to the influence of surface tension, the interface oscillates around its equilibrium position. In three dimensions we compare our simulations to Lamb’s analytic solution [21]. According to Lamb the oscillation frequency of an inviscid drop is

$$\omega_n^2 = \frac{n(n - 1)(n + 2)\sigma}{\rho_l R_0^5}, \tag{4.7}$$

where $\rho_l$ is the density of the droplet, $\sigma$ is the surface tension coefficient, and $R_0$ is the droplet radius. He also found that with viscosity the amplitude would decay as

$$a_n(t) = a_0 e^{-t/\tau}, \tag{4.8}$$

where $\tau = \frac{R_0^5}{(n-1)(2n+1)\nu}$ and $\nu$ is the kinematic viscosity of the drop. The radial position of the drop interface is

$$R(\theta, t) = R_0 + \epsilon P_n(\cos \theta) \sin(\omega_n t), \tag{4.9}$$

where $P_n$ is the Legendre polynomial of order $n$ and $\theta$ is the polar angle between 0 and $\pi$. We simulate with mode $n = 2$ and initialize at time $t = \pi / (2 \omega_n)$ and $\epsilon = 0.05 R_0$. 
Fig. 7. Solid line: Radius versus time for 3D drop oscillations in zero gravity. Dashed line: The theoretical envelope of the radius.

The drop radius is 1. The size of the computed domain is $4^3$ with a $25^3$ mesh. The initial interface has 1170 points. The density ratio and dynamic viscosity of the inner fluid and outer fluid are 600 and 1000. Figure 7 shows the radius of the drop at $\theta = 0$ versus time. The time is nondimensionalized by the theoretical period. The theoretical prediction for the envelope of the radius versus time is also plotted. The oscillation period calculated in our simulation is 5% longer than the theoretical value, and the error in drop amplitude is about 12%. Shin obtained 7% and 10% for the two errors with level contour reconstruction method [39].

4.4. Bubble-shock interaction. To test the stability of our method for high acoustic impedance mismatch (AIM) interface, we perform a simulation for a 2D gas bubble under strong shock in water. The initial condition is the same as those used in earlier studies [30, 43]. The parameters in equations of states are

\begin{align}
\gamma &= 1.4, \quad p_\infty = 0 \quad \text{for air,} \\
\gamma &= 4.4, \quad p_\infty = 6000 \text{ bar} \quad \text{for water.}
\end{align}

The dimension of the computational domain is $2.4cm \times 1.2cm \times 1.2cm$. $u = (0, 0)$, $p = 1$bar, $\rho_{\text{air}} = 0.001g/cm^3$, and $\rho_{\text{water}} = 1g/cm^3$. The air drop with radius 0.3cm is located in the center of the computational domain. A planar incident shock wave with postshock conditions $p = 1.8 \times 10^4$bar, $\rho = 1.323g/cm^3$, $u = 681.58m/s$, and $v = 0m/s$ is initially located 0.54cm to the left of the air bubble center. Boundary conditions are periodical in the vertical direction and nonreflecting at the left and right boundaries of the domain. Simulations are performed on a $256 \times 128$ grid. The initial curve has 402 points.

Figure 8 shows the response of the air bubble to the shock wave at six times. The Mach number and pressure are shown. After hitting the air bubble, a weak shock is transmitted in the air bubble. By approximate 3.6$\mu$s two sides of the air bubble hit each other and the air bubble breaks up into two parts. Upon impact, a blast wave is generated. The air bubble finally disappears completely. The results agree very well with earlier results in [30, 43].

4.5. Rayleigh instability. Four different regimes describe the phenomenology of liquid jet breakup. The first of these with the smallest values of $\Delta U$ was studied.
Fig. 8. Dynamics of the pressure and Mach fields for bubble-shock interaction test at time 1.6, 3.1, 3.5, 3.8, 3.9, 4.1 μs. Grid resolution is 64 per initial bubble diameter.

Theoretically by Rayleigh [34] using linear stability theory. For $\Delta U = 0$, a stationary jet whose length is longer than its perimeter is unstable under surface tension forces. Any disturbance can initiate the growth of a wave along the jet that will lead to breakup.

We perform simulations in a 3D domain $[-3r, 3r] \times [-3r, 3r] \times [0, \lambda]$, where $r = 2 \times 10^{-4}$ cm is the radius of the jet and $\lambda$ is the wave length. The computational domain has periodic boundary conditions in $x, y$, and $z$ directions. Following the linear stability theory, a sinusoidal perturbation is imposed in the initial time:

$$A = \text{Im}[A_0 e^{ikz}],$$

(4.12)
where $A_0$ is the initial perturbation amplitude set to $2.5\% \lambda$. We keep the grid spacing to 20 cells across the radial direction of the liquid column and change the disturbance wavelength $\lambda$. We compute the nondimensional growth rate $\beta/\beta_0$, where $\beta_0 = \sqrt{\sigma/\left(\rho r^3\right)}$. The growth rate is a function of the nondimensional wavenumber $\eta = 2\pi r/\lambda$. We compare our results to the linear stability theory of Weber [45] in Figure 9.

We then compare the convergence rate for $\lambda = 10r$ using $15 \times 50$, $30 \times 100$, and $60 \times 200$ meshes. The number of interface points are 3394, 6788, and 13576, respectively. The relative errors for the growth rates are summarized in Table 2. All methods have a first-order convergence rate, and our present method is more accurate than the other two methods. We compare the results after jet breakup obtained by two FTs. Figure 10 shows the temporal evolution of the interface using the present method and our previous FT under a $30 \times 30 \times 100$ mesh. Both methods give similar results.

We also study the convergence of the interface shape before and after pinch-off with the three different meshes. Figure 11 shows the interface location for the different meshes. It can be observed that the medium and fine mesh interfaces are close to each other, while the coarse mesh interface is very different. The relative $L_1$ errors of the radial distance before and after pinch-off are measured in a cylindrical coordinate system by

$$E_c = \frac{\int_0^\lambda \int_0^{2\pi} |r_c(\theta, z) - r_f(\theta, z)|d\theta dz}{\int_0^\lambda \int_0^{2\pi} |r_f(\theta, z)|d\theta dz},$$

where the subscripts $c$ and $f$ represent the coarse and fine grids. $\theta$ and $z$ are the angular coordinate and the height, respectively. As shown in Table 3, we obtain about first-order accuracy before and after pinch-off.

![Dispersion relation for Rayleigh instability.](image)

**Table 2**

The relative errors of the growth rate for Rayleigh instability.

<table>
<thead>
<tr>
<th>Mesh</th>
<th>15 × 50</th>
<th>30 × 100</th>
<th>60 × 200</th>
</tr>
</thead>
<tbody>
<tr>
<td>Present method</td>
<td>0.111</td>
<td>0.062</td>
<td>0.032</td>
</tr>
<tr>
<td>FT</td>
<td>0.130</td>
<td>0.065</td>
<td>0.033</td>
</tr>
<tr>
<td>FT/GFM</td>
<td>0.129</td>
<td>0.086</td>
<td>0.048</td>
</tr>
</tbody>
</table>

Weber analysis

Simulations

© by SIAM. Unauthorized reproduction of this article is prohibited.
Fig. 10. Evolution of the interface for Rayleigh instability. Top: Present method. Bottom: FT.

Table 3
The convergence of the numerical solution to the Rayleigh instability before and after pinch-off.

<table>
<thead>
<tr>
<th>Mesh</th>
<th>$15 \times 15 \times 50$</th>
<th>$30 \times 30 \times 100$</th>
<th>$60 \times 60 \times 200$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Before pinch-off</td>
<td>0.0736</td>
<td>0.0392</td>
<td>N/A</td>
</tr>
<tr>
<td>After pinch-off</td>
<td>0.220</td>
<td>0.0949</td>
<td>N/A</td>
</tr>
</tbody>
</table>
5. Primary breakup of a liquid jet. In order to assess the robustness of our algorithm, we perform a simulation of a liquid jet injected into quiescent high pressure gas. The simulation is a numerical test for our algorithm. The simulation is performed in the computational domain $3R \times 3R \times 40R$, where $R = 0.01 \text{cm}$ is the nozzle radius. Only a quarter of the jet is simulated considering the rotational symmetry of the jet. The parameters in our simulation are summarized in Table 4. The Weber number and density ratio have been decreased by factors of 5 and 4, respectively, from Parker and Rainaldi’s experiments [31] to make our simulation feasible. The jet is in the second wind-induced regime with these parameters. The whole region is discretized by using a uniform Cartesian mesh $120 \times 120 \times 1600$. At the beginning of the simulation, the domain is filled with high pressure gas. The liquid jet is injected from the left side of the domain. The inlet turbulent velocity is given by a filter-based generator with a prescribed length scale [20]. In our simulation, the integral length scale is taken to be the nozzle radius $\Lambda = R$. The turbulence intensity is 0.056 of the mean inlet velocity. Reflecting boundary conditions are used on the planes $y = 0$ and $z = 0$. All other boundary conditions are flow-through.

Figure 12 shows snapshots of the simulation result. In the snapshots, the quarter jet surface is reflected to form a whole jet for better visualization. The jet has a clear intact core near the nozzle exit, which is noticed in X-ray images [33]. The surface instability is first initialized by the inflow turbulence. Then, these instabilities grow into films due to the Kelvin–Helmholtz instability. Finally, the films are unstable and break up into filaments, which further break up into droplets. Liquid films, filaments, and droplets can be observed from Figure 13, which shows the detail of the jet surface at the end of the simulation.

Many 3D simulations have been performed to study the details of primary breakup using incompressible code. Using VOF–large eddy simulations (VOF-LES), Bianchi et al. [2] performed simulations under both laminar and turbulent conditions. The results confirm the role of turbulence in determining the onset of jet surface breakup. Using a refined level set grid method, Herrmann [16] simulated primary jet breakup with several levels of grid. The resulting droplet size distributions are of log normal type. Desjardins, Moureau, and Pitsch [5] reported results for turbulent jet atomization using a conservative level-set method coupled with a GFM. Menard, Tanguy, and Berlemont [29] employed the coupled level set VOF (CLSVOF) method to study the breakup of a turbulent jet. However, the resulting Sauter mean diameters (SMDs) from the above studies are not compared with those from experiments. Moreover, the gas Mach number of a jet in a diesel engine can reach 0.9 [31]. It is necessary to model the compressibility of the gas, although it may cause smaller time step sizes and thus longer simulation time compared with those from the incompressible simulations.
We evaluate the distribution of droplet diameters in the simulation. In Figure 14, the probability density function (PDF) of droplet diameters and its log normal fit are plotted together. The PDF follows a log normal distribution, as expected by many experiments [22]. We compare the resulting SMD from our simulation with that from Wu and Faeth’s experimentally validated correlation [46],

\[
SMD / \Lambda [1 + 0.04(\rho_g / \rho)(\bar{u}_0 / \bar{v}_0)^2(\Lambda / \text{SMD})^{2/3}]^{3/5} = 76(We_f \Lambda)^{-0.69}.
\]

Here $\bar{u}_0$ and $\bar{v}_0$ are the averaged streamwise velocity and root mean square velocity in the nozzle exit, $\Lambda$ is the integral turbulence length scale, and $We_f \Lambda$ is the liquid
Weber number with $\Lambda$ as the length scale. The SMD from (5.1) is 25 $\mu m$, which is close to 20 $\mu m$ from our simulation. We also study the range of the droplet diameter distribution. The ratio of the mass median diameter (MMD) to the SMD is proposed to describe the width of the droplet distribution [22]. Many authors found the relation $\text{MMD}/\text{SMD} = 1.2$ to hold after primary breakup for direct injection nozzles [40, 46]. We obtain $\text{MMD}/\text{SMD}=1.23$ from the simulation, which means that the droplet diameter distribution is nearly equal to but slightly wider than that in experiments.

6. Conclusions. We couple the FT with a GFM introduced in [18]. The proposed method improves our previous FT [6] and the FT/GFM method introduced in [43]. The improvement accomplishes improved late time stability. The GFM is used to couple the FT with the interior flow solver. The interface states are reconstructed from the interior states, and the normal velocity is solved from a Riemann problem. Since the method avoids solving Navier–Stokes equations on interface, it is more robust and less computationally expensive than our previous FT. A second-order surface normal algorithm [19] is implemented in the method to improve the accuracy of surface propagation. The surface tension force is modeled in a sharp way instead of spreading over the computational grid. An improved LGB is presented to handle the topological bifurcation of the tracked 3D surface mesh. It minimizes the numerical mass diffusion due to interface reconstruction. Our numerical results show the robustness of the proposed algorithm for the simulation of an interface with millions of triangles.

We verify the method by investigating numerically a broad set of test problems including compressible Kelvin–Helmholtz instabilities, parasitic currents, drop oscillation, bubble-shock interaction, and Rayleigh instabilities. The proposed new method is shown to be comparable to either of its constituent methods by themselves. To assess the robustness of our algorithm, we perform a simulation of a liquid jet injected into quiescent gas. The detailed process of the jet breakup is identified in the simulation. The physical aspects of the simulation will appear in a separate paper.

REFERENCES


A ROBUST FRONT TRACKING METHOD

1523


