INTERFACE TRACKING METHOD FOR COMPRESSIBLE MULTIFLUIDS

ALINA CHERTOCK[1], SMADAR KARNI[2] AND ALEXANDER KURGANOV[3]

Abstract. This paper is concerned with numerical methods for compressible multicomponent fluids. The fluid components are assumed immiscible, and are separated by material interfaces, each endowed with its own equation of state (EOS). Cell averages of computational cells that are occupied by several fluid components require a "mixed-cell" EOS, which may not always be physically meaningful, and often leads to spurious oscillations. We present a new interface tracking algorithm, which avoids using mixed-cell information by solving the Riemann problem between its single-fluid neighboring cells. The resulting algorithm is oscillation-free for isolated material interfaces, conservative, and tends to produce almost perfect jumps across material fronts. The computational framework is general and may be used in conjunction with one's favorite finite-volume method. The robustness of the method is illustrated on shock-interface interaction in one space dimension, oscillating bubbles with radial symmetry and shock-bubble interaction in two space dimensions.

1. Introduction

We are concerned with flow models describing the dynamics of fluids consisting of several (two) immiscible fluids. Assuming that all fluid components can be described by a single velocity and a single pressure, the governing equations are the compressible Euler equations, here written in two space dimensions (2-D):

$$\begin{pmatrix} \rho \\ \rho u \\ \rho v \\ E \end{pmatrix}_t + \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ u(E+p) \end{pmatrix}_x + \begin{pmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ v(E+p) \end{pmatrix}_y = 0, \tag{1.1}$$

where ρ is the density of the fluid mixture, u and v are the velocities, E is the total energy, and p is the pressure. The fluid components are separated by interfaces, and each fluid component is equipped with its own equation

[1] Department of Mathematics, North Carolina State University, Raleigh, NC 27695, USA. chertock@math.ncsu.edu

[2] Department of Mathematics, University of Michigan, Ann Arbor, MI 48109, USA. karni@umich.edu

[3] Mathematics Department, Tulane University, New Orleans, LA 70118, USA. kurganov@math.tulane.edu

of state (EOS). For example, the EOS for stiff gases is given by:

$$\rho e = \frac{p + \gamma p_\infty}{\gamma - 1}, \tag{1.2}$$

where e denotes the specific internal energy, γ the ratio of specific heats of the fluid component, and p_∞ its stiffness parameter, with $p_\infty = 0$ corresponding to the ideal gas case. The total energy E is then given by $E = \rho[e + \frac{1}{2}(u^2 + v^2)]$, and the speed of sound is:

$$c = \sqrt{\frac{\gamma(p + p_\infty)}{\rho}}. \tag{1.3}$$

The fluid components are identified by the variable ϕ, which may be taken to be a state variable, for example, the specific heat ratio γ [37] (see also [24]), or any function of γ (*e.g.*, $\phi = \frac{1}{\gamma-1}$ was advocated in [2], and extended to stiff gases in [38–40]). Related models take ϕ to be the mass fraction of the fluid component in the fluid mixture [1,2,6,31], or a level-set function, whose zero level-set defines the interface between the fluid components [10,11,34]. In all cases, ϕ propagates with the fluid velocity, hence satisfies the equation:

$$\phi_t + u\phi_x + v\phi_y = 0, \tag{1.4}$$

which may be combined with the first equation in (1.1) and recast in the conservation form:

$$(\rho\phi)_t + (\rho u\phi)_x + (\rho v\phi)_y = 0. \tag{1.5}$$

Within the popular finite-volume (FV) computational framework for conservation laws, numerical solutions are represented in terms of cell averages. When represented by its cell average, a solution discontinuity located at the interior of a cell inevitably generates intermediate solution values, and is generally represented as a diffused front (see, *e.g.*, [15,27,32,43]). In flows involving several fluid components, this diffusion of captured material fronts separating fluid components makes it necessary to consider the thermodynamics of mixed cells, even if the fluid components are assumed immiscible. It is well known that if mixed cells are treated naively, numerical methods produce unphysical pressure oscillations near material fronts (see, for instance, the review paper [4] and the references therein).

In recent years, numerous FV methods, capable to capture oscillation-free material interfaces, have been proposed, at the cost of not being strictly conservative, either in all conserved variables [11,16,17,24] or just in the total energy [3,4,23,25]. Loss of strict conservation compromises numerical convergence theory, and while the aforementioned methods were demonstrated to be essentially conservative, in that conservation error is typically proportional to the grid size and goes to zero with mesh refinement, the possibility remains that computed solutions may converge to incorrect weak limits, particularly in higher dimensions. Global conservation may be respected by using more complete flow models, which solve separately for the individual species energies [42,46,47] or the full multiphase flow model [5,9,26,38]. However, the individual species equations contain nonconservative products and do not completely circumvent the nonconservation doubt.

Front-tracking algorithms are able to represent discontinuities as sharp fronts, and thus avoid the need to consider the mixed cell EOS. They are easy to implement in the one-dimensional (1-D) case, but far less trivial to implement in several space dimensions [7,12–14,44]. Moving mesh techniques have been proposed as a way of keeping solution discontinuities sharp [20]. This is particularly appealing in the context of multifluid flows, as the need to consider mixed cells does not arise if the material front happens to coincide with a cell boundary. This observation is used in [8], where a locally moving mesh strategy ensures that the material interface always coincides with the moving cell boundary. The locally moving mesh algorithm works well for isolated material interfaces, but shock-interface interactions may produce spurious oscillations. Like front-tracking algorithms, moving mesh techniques are best suited for 1-D flows, and do not easily generalize for multi space dimensions.

In this paper, we propose an algorithm of a hybrid flavor. Of special relevance to the present work is [10], in which a level-set function is used to track the material front, and the Riemann problem solver is locally modified across the tracked interface in order to ensure oscillation-free propagation of the material front. The resulting algorithm yields a sharp resolution of the material front, but is non-conservative in all flow variables due to the interface procedure. The present algorithm carefully tracks material interfaces and respects the FV computational philosophy on a multi-cell block, for example, on a two-cell block in 1-D, as shown in Figure 3. The algorithm conserves total mass, momentum and energy, and yields sharp resolution of material interfaces, characteristic of front-tracking algorithms. The proposed computational framework is general and may be used in conjunction with one's favorite FV scheme. Away from the interface, characteristic information is less crucial, and a Riemann-problem-solver-free FV scheme may be used. In this paper, we have used the second-order central-upwind scheme developed in [28–30]. This Godunov-type scheme enjoys all major advantages of non-oscillatory central schemes and, at the same time, have a certain "built-in" upwind nature.

The paper is organized as follows. In Section 2, we derive the interface tracking method in 1-D, Section 2.1.1 presents the algorithm for the stiffened EOS (ideal fluids included), and Section 2.1.2 discusses its implementation to the Tait EOS. The algorithm is extended to "one-and-a-half" space dimensions radially symmetric flows in Section 2.2 and to 2-D in Section 2.3. Numerical results in 1-D are presented in Section 3.1, the dynamics of a compressible oscillating bubble is shown in Section 3.2. We conclude with two examples simulating the interaction of a shock wave with a bubble. The cases of Helium and R22 (freon) bubbles in air are shown.

2. THE INTERFACE TRACKING METHOD

2.1. One-dimensional method

We begin with a description of a general semi-discrete finite-volume framework. The essential part of the interface tracking method is presented in Sections 2.1.1 and 2.1.2.

We consider the 1-D version of the system (1.1):

$$\mathbf{w}_t + \mathbf{f}(\mathbf{w})_x = 0, \qquad \mathbf{w} := (\rho,\, \rho u,\, E)^T, \quad \mathbf{f}(\mathbf{w}) := (\rho u,\, \rho u^2 + p,\, u(E + p))^T, \tag{2.1}$$

closed with the EOS (1.2). For simplicity, we assume a uniform spatial grid, $x_\alpha = \alpha \Delta x$, and denote by $\overline{\mathbf{w}}_j(t)$ the cell averages of $\mathbf{w}(\cdot, t)$ over the corresponding intervals $I_j = (x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}})$:

$$\overline{\mathbf{w}}_j(t) := \frac{1}{\Delta x} \int_{I_j} \mathbf{w}(x, t) \, \mathrm{d}x. \tag{2.2}$$

Integrating (2.1) over I_j and dividing by Δx results in:

$$\frac{\mathrm{d}}{\mathrm{d}t} \overline{\mathbf{w}}_j(t) = -\frac{\mathbf{f}(\mathbf{w}(x_{j+\frac{1}{2}}, t)) - \mathbf{f}(\mathbf{w}(x_{j-\frac{1}{2}}, t))}{\Delta x}. \tag{2.3}$$

A semi-discrete FV scheme is obtained by approximating the fluxes at $x = x_{j\pm\frac{1}{2}}$ on the right-hand side (RHS) of (2.3), and can be written as:

$$\frac{\mathrm{d}}{\mathrm{d}t} \overline{\mathbf{w}}_j(t) = -\frac{\mathbf{H}_{j+\frac{1}{2}}(t) - \mathbf{H}_{j-\frac{1}{2}}(t)}{\Delta x}, \tag{2.4}$$

where $\mathbf{H}_{j+\frac{1}{2}}(t) = \mathcal{H}(\mathbf{w}_{j+\frac{1}{2}}^-(t), \mathbf{w}_{j+\frac{1}{2}}^+(t))$ is a numerical flux function. Here, $\mathbf{w}_{j+\frac{1}{2}}^+(t) := \mathcal{P}_{j+1}(x_{j+\frac{1}{2}}, t)$ and $\mathbf{w}_{j+\frac{1}{2}}^-(t) := \mathcal{P}_j(x_{j+\frac{1}{2}}, t)$ are the right and the left values at $x = x_{j\pm\frac{1}{2}}$ of a conservative, (essentially) non-oscillatory, piecewise polynomial interpolant,

$$\widetilde{\mathbf{w}}(x, t) = \mathcal{P}_j(x, t), \qquad x_{j-\frac{1}{2}} < x < x_{j+\frac{1}{2}}, \quad \forall j, \tag{2.5}$$

which is reconstructed at each time step from the previously computes cell averages, $\{\overline{\mathbf{w}}_j(t)\}$.

The formal spatial order of accuracy of the resulting scheme depends on the order of the piecewise polynomial reconstruction (2.5). We will restrict our consideration to second-order schemes that are typically based on a piecewise linear reconstruction which can be written as:

$$\mathcal{P}_j(x,t) = \overline{\mathbf{w}}_j + (\mathbf{w}_x)_j(x - x_j), \qquad x_{j-\frac{1}{2}} < x < x_{j+\frac{1}{2}}, \quad \forall j. \tag{2.6}$$

From now on, we will omit the time-dependence of all computed and reconstructed quantities. The second-order accuracy of the interpolant (2.6) is guaranteed provided $(\mathbf{w}_x)_j$ is (at least) first-order approximation of the derivative $\mathbf{w}_x(x_j, t)$. In order to ensure that $\mathcal{P}_j(x, t)$ is non-oscillatory, one has to use nonlinear limiters when computing the discrete derivatives $(\mathbf{w}_x)_j$. A library of such limiters is readily available in the literature (see, *e.g.*, [15,21,22,27,32,33,35,41,45]). In all our numerical experiments, we have used the multivariable minmod function (see, *e.g.*, [33,35,41,45]), which gives:

$$(\mathbf{w}_x)_j = \text{minmod}\left(\theta\frac{\overline{\mathbf{w}}_j - \overline{\mathbf{w}}_{j-1}}{\Delta x}, \frac{\overline{\mathbf{w}}_{j+1} - \overline{\mathbf{w}}_{j-1}}{2\Delta x}, \theta\frac{\overline{\mathbf{w}}_{j+1} - \overline{\mathbf{w}}_j}{\Delta x}\right), \qquad \theta \in [1, 2], \tag{2.7}$$

where the minmod function is defined as:

$$\text{minmod}(z_1, z_2, ...) := \begin{cases} \min_j\{z_j\}, & \text{if } z_j > 0 \ \ \forall j, \\ \max_j\{z_j\}, & \text{if } z_j < 0 \ \ \forall j, \\ 0, & \text{otherwise.} \end{cases} \tag{2.8}$$

Notice that larger θ's correspond to less dissipative and, in general, more oscillatory limiters.

To complete the semi-discrete formulation of the method one needs to select a formula for the numerical flux function $\mathcal{H}$. A variety of reliable functions $\mathcal{H}$ is available in the literature (see, *e.g.* [15,27,32,43], and references therein). In all our numerical experiments, we have used the second-order central-upwind scheme [28,30], whose brief description can be found in Appendix A.

The resulting semi-discretization (2.4) is a system of time-dependent ODEs. To obtain a fully-discrete scheme, the system (2.4) should be solved by a stable ODE solver. The formal temporal order of accuracy is then determined by the order of this solver. In all our numerical simulations, we have used the third-order strong stability preserving Runge-Kutta (SSP-RK) method [18].

2.1.1. *Interface treatment*

We focus on the two-fluid case with immiscible components, whose thermodynamics is given by an EOS of the general form (1.2) with different parameter values of γ and p_∞.

We assume that at some time $t \geq 0$ the solution, realized by the cell averages of the conserved quantities, $\{\overline{\mathbf{w}}_j = (\overline{\rho}_j, (\overline{\rho u})_j, \overline{E}_j)^T\}$, is available, and that the position of the material interface $x_{\text{int}} = x_{\text{int}}(t)$ is known and is contained in cell J; that is, $x_{J-\frac{1}{2}} \leq x_{\text{int}} \leq x_{J+\frac{1}{2}}$, and $\gamma = \gamma_L$, $p_\infty = p_{\infty,L}$ for $x < x_{\text{int}}$ and $\gamma = \gamma_R$, $p_\infty = p_{\infty,R}$ for $x > x_{\text{int}}$.

One step of the proposed time evolution algorithm consists of the following stages.

For each j, the computation of the slopes (using (2.7) or an alternative limiter) is based on the values of the *cell averages* $\overline{\mathbf{w}}_j$ and the cell averages $\overline{\mathbf{w}}_{j-1}$ and $\overline{\mathbf{w}}_{j+1}$ in the two neighboring cells. However, the J-th cell containing the material interface is occupied by both fluids and the corresponding cell averages, while correctly representing the total amounts of conserved quantities, may not be thermodynamically meaningful which may consequently cause the algorithm to fail. Our interface algorithm therefore refrains from using cell-averaged quantities in the mixed cell. This implies that the piecewise linear interpolant (2.6)–(2.7) needs to be modified in cells $j = J - 1, J, J + 1$.

To compute the point values $\mathbf{w}^+_{J-\frac{1}{2}}$ and $\mathbf{w}^-_{J+\frac{1}{2}}$ of the numerical solution at cell interfaces $x_{J\pm\frac{1}{2}}$, we note that cells $(J \pm 1)$ are occupied by single fluids, hence their cell averages are thermodynamically meaningful. We take lead from [10] and solve the Riemann problem between the cell average values $\overline{\mathbf{w}}_{J-1}$ and $\overline{\mathbf{w}}_{J+1}$ either exactly
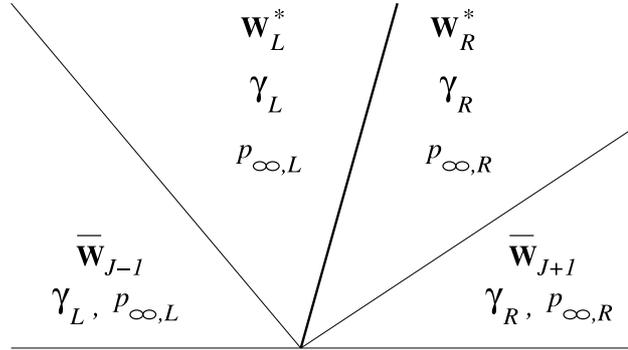
FIGURE 1. A typical (shock-contact-shock) self-similar solution of the Riemann problem between the cell average values $\overline{\mathbf{w}}_{J-1}$ and $\overline{\mathbf{w}}_{J+1}$, presented in the (x, t)-plane.

or approximately. The solution of this Riemann problem will now be used to determine the states $\mathbf{w}^+_{J-\frac{1}{2}}$ and $\mathbf{w}^-_{J+\frac{1}{2}}$ at the most left, $x = x_{J-\frac{1}{2}}$, and right, $x = x_{J+\frac{1}{2}}$, points of cell J.

The exact solution of the two-fluid Riemann problem is known (see, *e.g.*, [32] and the references therein). It consists of two nonlinear acoustic waves (either shocks or rarefactions) and a contact discontinuity which is also a material interface, as illustrated in Figure 1. We denote by $(\mathbf{w}^*_L; \gamma_L, p_{\infty,L})$ and $(\mathbf{w}^*_R; \gamma_R, p_{\infty,R})$ the two states to the left/right of the contact discontinuity $(\mathbf{w}^*_L := (\rho^*_L, \rho^*_L u^*_L, E^*_L)^T, \mathbf{w}^*_R := (\rho^*_R, \rho^*_R u^*_R, E^*_R)^T)$, and the corresponding velocities $u^*_L = u^*_R$. We note that u^*_L and u^*_R are constant across the contact discontinuity, and denote their common value by u^*, and the sound speeds $c^*_{L,R}$ given by (1.3).

Once the intermediate states of the Riemann problem solution are available, we proceed along the lines of [10] to compute the required values at the endpoints of the cell J:

$$
\begin{array}{ll}
\texttt{if } u^* - c^*_L < 0 \texttt{ then} & \texttt{if } u^* + c^*_R > 0 \texttt{ then} \\
\quad \mathbf{w}^+_{J-\frac{1}{2}} = \mathbf{w}^*_L, & \quad \mathbf{w}^-_{J+\frac{1}{2}} = \mathbf{w}^*_R, \\
\texttt{otherwise} & \texttt{otherwise} \\
\quad \mathbf{w}^+_{J-\frac{1}{2}} = \overline{\mathbf{w}}_{J-1}, & \quad \mathbf{w}^-_{J+\frac{1}{2}} = \overline{\mathbf{w}}_{J+1}.
\end{array}
\tag{2.9}
$$

The point values $\mathbf{w}^+_{J-\frac{1}{2}}$ and $\mathbf{w}^-_{J+\frac{1}{2}}$ are then used to reconstruct conservative linear pieces (2.6) at the neighboring cells I_{J-1} and I_{J+1}. The reconstruction is made non-oscillatory by computing the derivatives $(\mathbf{w}_x)_{J-1}$ and $(\mathbf{w}_x)_{J+1}$ with the help of the minmod limiter, applied to the cell averages at these cells and to the corresponding point values, already computed at the endpoints of the cells I_{J-2}, I_J, and I_{J+2}:

$$
(\mathbf{w}_x)_{J-1} = \text{minmod}\left(\frac{\overline{\mathbf{w}}_{J-1} - \mathbf{w}^-_{J-\frac{3}{2}}}{\Delta x / 2}, \frac{\mathbf{w}^+_{J-\frac{1}{2}} - \overline{\mathbf{w}}_{J-1}}{\Delta x / 2} \right),
\tag{2.10}
$$

$$
(\mathbf{w}_x)_{J+1} = \text{minmod}\left(\frac{\overline{\mathbf{w}}_{J+1} - \mathbf{w}^-_{J+\frac{1}{2}}}{\Delta x / 2}, \frac{\mathbf{w}^+_{J+\frac{3}{2}} - \overline{\mathbf{w}}_{J+1}}{\Delta x / 2} \right),
\tag{2.11}
$$

where the minmod function, given by (2.8), is applied componentwise, see Figure 2.

With the values $\{\mathbf{w}^\pm_{j+\frac{1}{2}}\}$ at hand, the cell averages $\{\overline{\mathbf{w}}_j\}$ are evolved from time t to time $t + \Delta t$ according to the FV scheme (2.4).
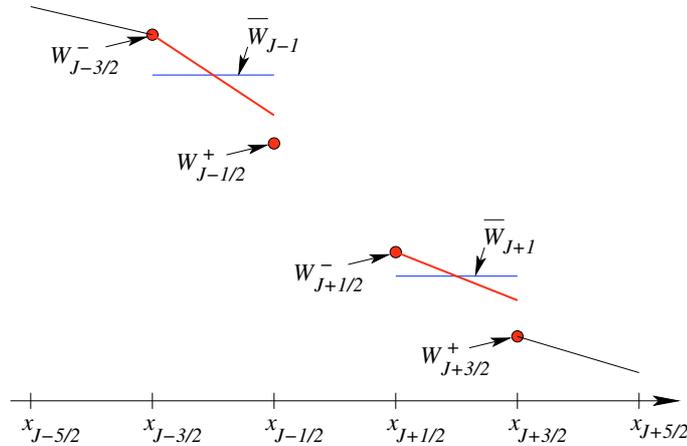
FIGURE 2. Special minmod reconstruction in cells $(J+1)$ and $(J-1)$.

The solution of the Riemann problem between the cell average values $\overline{\mathbf{w}}_{J-1}$ and $\overline{\mathbf{w}}_{J+1}$ is also used for propagating the material interface by numerically solving the ODE

$$\frac{\mathrm{d}x_{\mathrm{int}}}{\mathrm{d}t} = u^*.$$

Going from time t to $t+\Delta t$, two scenarios are possible: the interface, $x_{\mathrm{int}}(t+\Delta t)$, either remains in cell J or crosses over to one of its neighboring cells $(J \pm 1)$ (due to the CFL condition, the interface may not move by more than Δx per one time step).

If $x_{\mathrm{int}}(t+\Delta t)$ remains in cell J, the evolution step is complete and we proceed to the next time step.

Else, we follow [10] and use the solution of the Riemann problem between the cell average values $\overline{\mathbf{w}}_{J-1}$ and $\overline{\mathbf{w}}_{J+1}$ once again and overwrite $\overline{\mathbf{w}}_J^{n+1}$ by either $\mathbf{w}_L^*$ or $\mathbf{w}_R^*$ depending on the sign of u^*. This step in itself affects the overall conservation balance. To maintain global conservation, the new mixed cell (either the $(J+1)$-st or $(J-1)$-st one) is used as a conservation "storage" cell – its value is corrected so that the resulting method remains globally conservative. The correction procedure is summarized in the following algorithm:

$$
\begin{aligned}
&\text{if } x_{\mathrm{int}}(t+\Delta t) \in I_{J+1} \text{ then} & &\text{if } x_{\mathrm{int}}(t+\Delta t) \in I_{J-1} \text{ then} \\
&\quad \overline{\mathbf{w}}_J^{\mathrm{new}} = \mathbf{w}_L^*, & &\quad \overline{\mathbf{w}}_J^{\mathrm{new}} = \mathbf{w}_R^*, \\
&\quad \overline{\mathbf{w}}_{J+1}^{\mathrm{new}} = \overline{\mathbf{w}}_J + \overline{\mathbf{w}}_{J+1} - \overline{\mathbf{w}}_J^{\mathrm{new}}, & &\quad \overline{\mathbf{w}}_{J-1}^{\mathrm{new}} = \overline{\mathbf{w}}_{J-1} + \overline{\mathbf{w}}_J - \overline{\mathbf{w}}_J^{\mathrm{new}}.
\end{aligned}
\tag{2.12}
$$

Here, all the $\overline{\mathbf{w}}$ quantities are taken at time $t+\Delta t$.

Notice that this correction procedure (2.12) ensures a local conservation over a double-cell (either $(J, J+1)$ or $(J-1, J)$ depending on the direction of the interface propagation) since only the values in this double-cell are getting corrected and since either $\overline{\mathbf{w}}_{J+1}^{\mathrm{new}} + \overline{\mathbf{w}}_J^{\mathrm{new}} = \overline{\mathbf{w}}_{J+1} + \overline{\mathbf{w}}_J$ or $\overline{\mathbf{w}}_J^{\mathrm{new}} + \overline{\mathbf{w}}_{J-1}^{\mathrm{new}} = \overline{\mathbf{w}}_J + \overline{\mathbf{w}}_{J-1}$, respectively. This is illustrated in Figure 3.

This completes one evolution step of the proposed Interface Tracking Method.

Remark 2.1. The interface tracking algorithm may be implemented using an approximate Riemann problem solution instead of the exact one. Since the solution of the Riemann problem is only used to interpolate between the states $\overline{\mathbf{w}}_{J-1}$ and $\overline{\mathbf{w}}_{J+1}$ near the material interface, the exact solver does not have a particular advantage over an approximate one as long as the latter does not produce unphysical values of the interpolated quantities (our numerical experiments below clearly demonstrate this). Moreover, the approximate solver may be the only
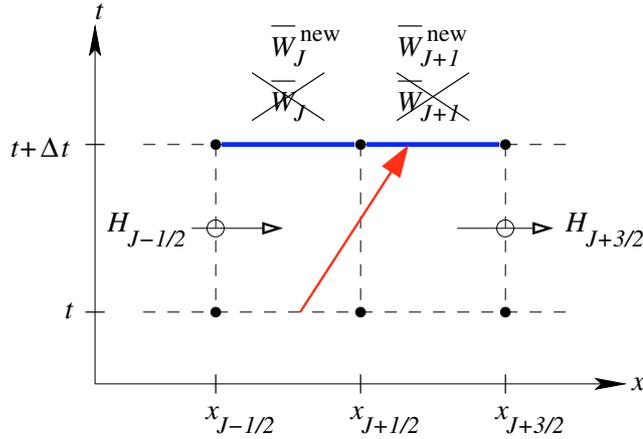
FIGURE 3. Typical interface propagation and double-cell conservation.

option if the exact one is unavailable (see, for example, the radially symmetric problem studied in Sect. 2.2). In our numerical experiments, we have used both the exact and the linearized Roe-type solver proposed in [24].

2.1.2. *Modeling liquids using the Tait equation of state*

We are interested in modeling liquid components of multifluids by compressible Euler equations. This can be done in several ways. For example, one may use the full gas dynamics system (2.1) subject to the stiff EOS: (1.3) with a large value of p_∞. Alternatively, the liquid component can be modeled by the isentropic Euler equations:

$$\begin{cases} \rho_t + (\rho u)_x = 0, \\ (\rho u)_t + (\rho u^2 + p)_x = 0, \end{cases} \tag{2.13}$$

closed with the Tait EOS:

$$p = B \left(\frac{\rho}{\rho_0} \right)^\gamma - B + A, \tag{2.14}$$

where γ, A, B, and ρ_0 are constants, see [48].

If the liquid is modeled by the full gas dynamics system with the stiff EOS, our interface tracking method can be applied straightforwardly to the liquid-gas multifluid. However, when the alternative approach is used, the interface treatment should be slightly modified.

First, notice that the number of equations in liquid and gas areas is different. As for the mixed cell J, which is served as a "conservation storage" cell, it would make sense to "store" there the density and momentum only since the overall model does not conserve the total energy.

To apply the semi-discrete FV scheme, we need, as before, to reconstruct the point values $\mathbf{w}^+_{J-\frac{1}{2}}$ and $\mathbf{w}^-_{J+\frac{1}{2}}$ by (approximately) solving the Riemann problem between the cell average values $\overline{\mathbf{w}}_{J-1}$ and $\overline{\mathbf{w}}_{J+1}$. Without loss of generality, let us assume that the gas component is on the left while the liquid one is on the right of cell J. Then the vectors $\overline{\mathbf{w}}_{J-1}$ and $\mathbf{w}^+_{J-\frac{1}{2}}$ have 3 components while the vectors $\overline{\mathbf{w}}_{J+1}$ and $\mathbf{w}^-_{J+\frac{1}{2}}$ have only 2 components. Since the gas and liquid systems are connected through the contact wave, one may still construct the (approximate) Riemann problem solver. For example, the shock-shock solver can be described as follows.

Given $(\overline{\mathbf{w}}_{J-1}, \gamma_L)$ and $(\overline{\mathbf{w}}_{J+1}, \gamma_R)$ in the phase space, we denote by $(\mathbf{w}^*_L, p^*, \gamma_L)$ and $(\mathbf{w}^*_R, p^*, \gamma_R)$ two intermediate constant states that appear in the solution of the Riemann problem, in which $\mathbf{w}^*_L = (\rho^*_L, \rho^*_L u^*, E^*_L)^T$ consists of 3 components while $\mathbf{w}^*_R = (\rho^*_R, \rho^*_R u^*)^T$ consists of 2 components only. Using the the Rankine-Hugoniot condition for each shock, we obtain ρ^*_L, u^* and p^* and then compute $\rho^*_R = \rho^*_R(p^*)$ from (2.14), see Figure 4. Having the intermediate states $(\mathbf{w}^*_L, p^*, \gamma_L)$ and $(\mathbf{w}^*_R, p^*, \gamma_R)$, we proceed as in Section 2.1.1
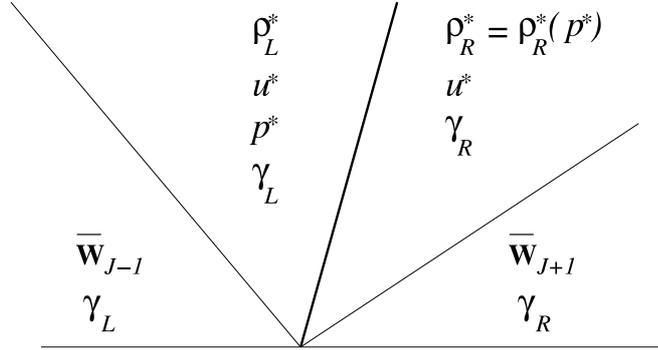
FIGURE 4. The same as in Figure 1, but now the liquid component on the right is modeled by the isentropic Euler equations (2.13) with the Tait EOS (2.14).

to determine $\mathbf{w}^{+}_{J-\frac{1}{2}}$ and $\mathbf{w}^{-}_{J+\frac{1}{2}}$ according to (2.9). Finally, when the interface moves to the neighboring cell, the cell averages at cell J are recovered following (2.12).

2.2. Oscillating bubble with radial symmetry

We have used the Euler equations to compute the dynamics of a compressible oscillating bubble with surface tension. The fluids inside and outside the bubble are separated by the bubble surface and are thermodynamically distinct. The bubble is in equilibrium if the difference between the inside and outside pressures is balanced by the force due to surface tension:

$$[p] = \kappa\tau,$$

where κ is the interface curvature and τ the surface tension coefficient. A larger pressure difference causes the bubble to expand, and by inertia to overexpand, leading to a pressure drop inside the bubble, causing the bubble to compress, and then overcompress. This repeated oscillatory motion is gradually damped by acoustic radiation of energy, and the bubble reaches a new equilibrium. In this model, the bubble is assumed to maintain its integrity at all times and to remain cylindrically symmetric. The force due to surface tension is a singular force that acts on the interface, and is modeled by a Dirac δ-function. The Euler equations with cylindrical symmetry take the form:

$$(r\rho)_t + (r\rho q)_r = 0, \tag{2.15}$$
$$(r\rho q)_t + (r\rho q^2 + rp)_r = p - r\tau\kappa\,\delta(r-R), \tag{2.16}$$
$$(rE)_t + (q(rE + rp))_r = -r\tau\kappa q\,\delta(r-R), \tag{2.17}$$

where ρ, q, p, and $E = \rho e + \frac{1}{2}\rho q^2$ denote the density, radial velocity, pressure, and total energy. The curvature $\kappa = 1/R$, where $R = R(t)$ is the bubble radius. The fluid components are assumed to obey the EOS for stiff gases (1.2).

We introduce a uniform spatial grid, in which the cells $I_j = (r_{j-\frac{1}{2}}, r_{j+\frac{1}{2}})$ of size $\Delta r := r_{j+\frac{1}{2}} - r_{j-\frac{1}{2}}$ are centered at the points $r = r_j$, and denote the vector of conservative variables by $\mathbf{w} := (r\rho, r\rho q, rE)^T$, the flux function by $\mathbf{f}(\mathbf{w}) := (r\rho q, r\rho q^2 + p, rq(E+p))^T$, and the source term by $\mathbf{S} := (0, p - r\tau\kappa\,\delta(r-R), -r\tau\kappa q\,\delta(r-R))^T$. As before, we assume that at some time level t, the solution is available and realized in terms of its cell averages $\{\overline{\mathbf{w}}_j(t)\}$. We also assume that the interface location $R(t)$ is known, and is contained in cell I_J, that is $r_{J-\frac{1}{2}} \le R(t) \le r_{J+\frac{1}{2}}$, and $\gamma = \gamma_{\mathrm{in}}$, $p_\infty = p_{\infty,\mathrm{in}}$ for $r < R(t)$ and $\gamma = \gamma_{\mathrm{out}}$, $p_\infty = p_{\infty,\mathrm{out}}$ for $r > R(t)$.

Away from the interface, the solution cell averages are evolved in time according to the semi-discrete FV scheme:

$$\frac{\mathrm{d}}{\mathrm{d}t}\overline{\mathbf{w}}_j(t) = -\frac{\mathbf{H}_{j+\frac{1}{2}}(t) - \mathbf{H}_{j-\frac{1}{2}}(t)}{\Delta r} + \overline{\mathbf{S}}_j(t), \tag{2.18}$$

where $\mathbf{H}_{j+\frac{1}{2}} = \mathcal{H}(t, \mathbf{w}_{j+\frac{1}{2}}^-(t), \mathbf{w}_{j+\frac{1}{2}}^+(t))$ is a numerical flux function, $\mathbf{w}_{j+\frac{1}{2}}^+(t) := \widetilde{\mathbf{w}}(r_{j+\frac{1}{2}} + 0, t)$ and $\mathbf{w}_{j+\frac{1}{2}}^-(t) := \widetilde{\mathbf{w}}(r_{j+\frac{1}{2}} - 0, t)$ are the right and the left values at $r = r_{j\pm\frac{1}{2}}$ of a global, (essentially) non-oscillatory, conservative piecewise polynomial interpolant $\widetilde{\mathbf{w}}(r, t)$, reconstructed from the cell averages $\{\overline{\mathbf{w}}_j(t)\}$. The numerical source term $\overline{\mathbf{S}} = (0, \overline{S}^{(2)}, \overline{S}^{(3)})^T$ is:

$$\overline{S}_j^{(2)}(t) = \frac{1}{\Delta r} \int\limits_{I_j} p(r, t)\, \mathrm{d}r - R\tau\kappa\delta_j^J, \qquad \overline{S}_j^{(3)}(t) = -R\tau\kappa q^*\delta_j^J. \tag{2.19}$$

Here, δ_j^J is the Kronecker symbol, J is the number of the cell in which the bubble surface is located at time t, q^* is the interface velocity to be determined below (see p.), and the pressure integral should be obtained by an appropriate quadrature, for example, by a mid-point rule. In our numerical experiments, we have used the 1-D second-order central-upwind scheme, described in Appendix A.

2.2.1. *Interface treatment*

As in 1-D, we refrain from using the cell averages of the mixed cell, I_J, as they lead to thermodynamical inconsistencies and typically result in unphysical pressure oscillation. Instead, we obtain the point values of the solution at the cell edges $x_{J\pm\frac{1}{2}}$ by solving the Riemann problem between the cell averages in the neighboring cells $(J-1)$ and $(J+1)$. Here, we have used an approximate Riemann problem solver, described below. The governing equations (2.15)–(2.17) are nonconservative due to the non-Cartesian geometry and surface tension force. Ignoring the effects of the source terms, the solution for the Riemann problem in the radially symmetric setup is essentially the same as that in 1-D, with the density, pressure and energy scaled by r, that is, ρ, p and E replaced by $r\rho$, rp and rE, respectively. Surface tension sustains a pressure jump of size $[p] = \tau\kappa$ across the bubble surface, and thus the pressure is no longer continuous in the surface normal direction. This jump needs to be incorporated into the approximate Riemann problem solver. The pressure source term needs to be integrated over the control volume and is treated by a source linearization.

Approximate Riemann problem solver

The way in which we make use of the approximate Riemann problem solution does not require that the solution is conservative, as conservation is insured by the flux form of the semi-discrete scheme and by the conservation property of the correction procedure (2.12). We therefore write the governing equations (2.15)–(2.17) in terms of the primitive variables $\mathbf{U} = (\rho, u, p)^T$ and consider the Riemann problem:

$$\rho_t + q\rho_r + \rho q_r = -\frac{\rho q}{r}, \tag{2.20}$$

$$q_t + qq_r + \frac{1}{\rho}p_r = -\frac{1}{\rho}\tau\kappa\delta(r - R), \tag{2.21}$$

$$p_t + qp_r + \rho c^2 q_r = \frac{\rho q c^2}{r} - \tau\kappa q\delta(r - R), \tag{2.22}$$

with sound speed c is given by (1.3) and subject to the initial data

$$(\mathbf{U}(r, 0); \gamma, p_\infty) = \begin{cases} (\mathbf{U}_{\text{in}}; \gamma_{\text{in}}, p_{\infty,\text{in}}) := ((\rho_{J-1}, q_{J-1}, p_{J-1})^T; \gamma_{\text{in}}, p_{\infty,\text{in}}), & r < r_0, \\ (\mathbf{U}_{\text{out}}; \gamma_{\text{out}}, p_{\infty,\text{out}}) := ((\rho_{J+1}, q_{J+1}, p_{J+1})^T; \gamma_{\text{out}}, p_{\infty,\text{out}}), & r > r_0. \end{cases} \tag{2.23}$$

Since the exact solution of this Riemann problem is unavailable, we will use the solution of the corresponding linearized system. It should be pointed out, however, that the system (2.20)–(2.22) in its current form cannot be linearized due to the presence of the singular terms on the RHS of (2.21) and (2.22). Therefore, we will first modify the initial data (2.23) to reflect the effect of surface tension that supports the pressure jump, $[p] = \tau\kappa$, across the material interface: we will subtract the vector $(0, 0, \beta\tau\kappa)^T$ from $\mathbf{U}_{\text{in}}$ and add the vector

$(0, 0, (1-\beta)\tau\kappa)^T$ to $\mathbf{U}_{\text{out}}$. Then, we will remove the δ-source from the RHS of (2.21) and (2.22) and linearize the obtained system about an average state. Once the solution of the linearized system has been found, we will restore the approximate solution of the original Riemann problem by adding/subtracting the aforementioned vectors back to the corresponding inside/outside states.

We now turn to a detailed description of the algorithm. We begin by modifying the initial data as follows:

$$\widehat{\mathbf{U}}_{\text{in}} = \mathbf{U}_{\text{in}} - \begin{pmatrix} 0 \\ 0 \\ \beta\tau\kappa \end{pmatrix}, \qquad \widehat{\mathbf{U}}_{\text{out}} = \mathbf{U}_{\text{out}} + \begin{pmatrix} 0 \\ 0 \\ (1-\beta)\tau\kappa \end{pmatrix}, \qquad 0 \le \beta \le 1, \tag{2.24}$$

where β is a free parameter. We would like to recall that the goal of such a modification is to reflect the effect of surface tension so that $\widehat{\mathbf{U}}_{\text{out}} - \widehat{\mathbf{U}}_{\text{in}} = \mathbf{U}_{\text{out}} - \mathbf{U}_{\text{in}} + \tau\kappa$. The latter is true for any value of β. Our numerical experiments indicate that the resulting method is not too sensitive to the choice of β, so in all numerical results reported below, we have used $\beta = 0.5$.

We then remove the δ-source terms from the RHS of the system (2.20)–(2.22) and linearize it about some average state, again with no special regard to conservation, for example, about $\widehat{\mathbf{U}}_{\text{av}} := \frac{1}{2}(\widehat{\mathbf{U}}_{\text{in}} + \widehat{\mathbf{U}}_{\text{out}})$, and consider the linearized Riemann problem:

$$\begin{cases} \mathbf{U}_t + A(\widehat{\mathbf{U}}_{\text{av}})\mathbf{U}_x = \boldsymbol{\sigma}_{\text{av}}, \\ \mathbf{U}(r, 0) = \begin{cases} \widehat{\mathbf{U}}_{\text{in}}, & r < r_0, \\ \widehat{\mathbf{U}}_{\text{out}}, & r > r_0, \end{cases} \end{cases} \tag{2.25}$$

where $\boldsymbol{\sigma}_{\text{av}}$ is a piecewise constant approximation of the source term $\boldsymbol{\sigma} \equiv (\sigma^{(1)}, \sigma^{(2)}, \sigma^{(3)})^T := (-\rho q/r,\, 0,\, \rho q c^2/r)^T$, which is a nonsingular part of the source term in the system (2.20)–(2.22), and A is the coefficient matrix of this system:

$$A(\mathbf{U}) = \begin{pmatrix} q & \rho & 0 \\ 0 & q & 1/\rho \\ 0 & \rho c^2 & q \end{pmatrix}, \tag{2.26}$$

whose eigenvalues and the corresponding right eigenvectors are:

$$\lambda_1 = q - c, \qquad \lambda_2 = q, \qquad \lambda_3 = q + c, \tag{2.27}$$

$$\mathbf{r}_1 = \begin{pmatrix} \rho \\ -c \\ \rho c^2 \end{pmatrix}, \quad \mathbf{r}_2 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \quad \mathbf{r}_3 = \begin{pmatrix} \rho \\ c \\ \rho c^2 \end{pmatrix}. \tag{2.28}$$

Next, the difference $\Delta\widehat{\mathbf{U}} := \widehat{\mathbf{U}}_{\text{out}} - \widehat{\mathbf{U}}_{\text{in}}$ is projected onto the characteristic fields of the linearized system:

$$\Delta\widehat{\mathbf{U}} = \sum_{k=1}^{3} \alpha_k \mathbf{r}_k, \qquad \mathbf{r}_k = \mathbf{r}_k(\widehat{\mathbf{U}}_{\text{av}}),\ k = 1, 2, 3,$$

where the wave strengths α_k are given by:

$$\alpha_1 = \frac{\Delta p - \rho_{\text{av}} c_{\text{av}} \Delta q}{2\rho_{\text{av}} c_{\text{av}}^2}, \qquad \alpha_2 = \Delta\rho - \frac{\Delta p}{c_{\text{av}}^2}, \qquad \alpha_3 = \frac{\Delta p + \rho_{\text{av}} c_{\text{av}} \Delta q}{2\rho_{\text{av}} c_{\text{av}}^2},$$

and

$$c_{\text{av}}^2 := \frac{c_{\text{in}}^2 + c_{\text{out}}^2}{2}.$$

We note that this linearization may not correspond to a thermodynamically consistent state, that is, in general, $c_{\mathrm{av}}^2 \neq c_{\mathrm{av}}^2(\rho_{\mathrm{av}}, p_{\mathrm{av}})$. This is necessary in order to circumvent the need to define a mixed-cell EOS, since there may not be a thermodynamically meaningful way to define an average γ and p_∞.

The source difference $\Delta\boldsymbol{\sigma} := \boldsymbol{\sigma}(\widehat{\mathbf{U}}_{\mathrm{out}}) - \boldsymbol{\sigma}(\widehat{\mathbf{U}}_{\mathrm{in}})$ is also projected onto the characteristic fields of the linearized system:

$$\Delta\boldsymbol{\sigma} = \sum_{k=1}^{3} \delta_k \mathbf{r}_k,$$

where the coefficients are:

$$\delta_1 = \delta_3 = \frac{\Delta\sigma^{(3)}}{2\rho_{\mathrm{av}} c_{\mathrm{av}}^2}, \qquad \delta_2 = \Delta\sigma^{(1)} - \frac{\Delta\sigma^{(3)}}{c_{\mathrm{av}}^2}.$$

An appropriate piecewise constant approximation of the source term is then given by:

$$\boldsymbol{\sigma}_{\mathrm{av}} = \begin{cases} \boldsymbol{\sigma}(\widehat{\mathbf{U}}_{\mathrm{in}}), & r < r_0 + \lambda_1 t, \\ \boldsymbol{\sigma}(\widehat{\mathbf{U}}_{\mathrm{in}}) + \delta_1 \mathbf{r}_1, & r_0 + \lambda_1 t < r < r_0 + \lambda_2 t, \\ \boldsymbol{\sigma}(\widehat{\mathbf{U}}_{\mathrm{out}}) - \delta_3 \mathbf{r}_3, & r_0 + \lambda_2 t < r < r_0 + \lambda_3 t, \\ \boldsymbol{\sigma}(\widehat{\mathbf{U}}_{\mathrm{out}}), & r > r_0 + \lambda_3 t, \end{cases} \qquad (2.29)$$

where $\lambda_k = \lambda_k(\widehat{\mathbf{U}}_{\mathrm{av}})$, $k = 1, 2, 3$.

The solution of the initial value problem (IVP) (2.25)–(2.29) is:

$$\widehat{\mathbf{U}}(r, t) = \begin{cases} \widehat{\mathbf{U}}_{\mathrm{in}}(t), & r < r_0 + \lambda_1 t, \\ \widehat{\mathbf{U}}_{\mathrm{in}}^*(t), & r_0 + \lambda_1 t < r < r_0 + \lambda_2 t, \\ \widehat{\mathbf{U}}_{\mathrm{out}}^*(t), & r_0 + \lambda_2 t < r < r_0 + \lambda_3 t, \\ \widehat{\mathbf{U}}_{\mathrm{out}}(t), & r > r_0 + \lambda_3 t, \end{cases}$$

where

$$\widehat{\mathbf{U}}_{\mathrm{in}}(t) := \widehat{\mathbf{U}}_{\mathrm{in}} + t\,\boldsymbol{\sigma}(\widehat{\mathbf{U}}_{\mathrm{in}}), \qquad \widehat{\mathbf{U}}_{\mathrm{in}}^*(t) := \widehat{\mathbf{U}}_{\mathrm{in}} + \alpha_1 \mathbf{r}_1 + t\left(\boldsymbol{\sigma}(\widehat{\mathbf{U}}_{\mathrm{in}}) + \delta_1 \mathbf{r}_1\right),$$
$$\widehat{\mathbf{U}}_{\mathrm{out}}(t) := \widehat{\mathbf{U}}_{\mathrm{out}} + t\,\boldsymbol{\sigma}(\widehat{\mathbf{U}}_{\mathrm{out}}), \qquad \widehat{\mathbf{U}}_{\mathrm{out}}^*(t) := \widehat{\mathbf{U}}_{\mathrm{out}} - \alpha_4 \mathbf{r}_4 + t\left(\boldsymbol{\sigma}(\widehat{\mathbf{U}}_{\mathrm{out}}) - \delta_3 \mathbf{r}_3\right).$$

Note that the second components of both $\widehat{\mathbf{U}}_{\mathrm{in}}^*(0)$ and $\widehat{\mathbf{U}}_{\mathrm{out}}^*(0)$ are equal to the interface velocity q^*, used to evaluate the source cell average in (2.19).

Finally, we restore the solution of the original Riemann problem (2.20)–(2.23) by adding/subtracting, according to (2.24), the vectors $(0, 0, \beta\tau\kappa)^T$ and $(0, 0, (1 - \beta)\tau\kappa)^T$ from the above $\widehat{(\cdot)}$-values:

$$\mathbf{U}(r, t) = \begin{cases} \widehat{\mathbf{U}}(r, t) + (0, 0, \beta\tau\kappa)^T, & r < r_0 + \lambda_2 t, \\ \widehat{\mathbf{U}}(r, t) - (0, 0, (1 - \beta\tau\kappa))^T, & r > r_0 + \lambda_2 t. \end{cases}$$

This completes the description of our approximate Riemann problem solver.

Remark 2.2. Since the interface tracking method is based on the semi-discrete FV scheme, we only need the solution of the Riemann problem at time $t = 0+$ to evaluate the point values $\mathbf{w}_{J-\frac{1}{2}}^+$ and $\mathbf{w}_{J+\frac{1}{2}}^-$, as described in Section 2.1.1. Therefore, the piecewise constant approximation of the source, $\boldsymbol{\sigma}_{\mathrm{av}}$, is not used by our method, and formula (2.29) is only provided for the completeness of a description of the approximate Riemann problem solver, presented in this section.

2.3. **Two-dimensional method**

We now turn to the system (1.1) and describe our 2-D interface tracking method.

We consider a uniform spatial grid $x_\mu := \mu\Delta x$, $y_\nu := \nu\Delta y$, and denote the computed quantities, the cell averages, by:

$$\overline{\mathbf{w}}_{j,k}(t) := \frac{1}{\Delta x\Delta y} \iint\limits_{I_{j,k}} \mathbf{w}(\xi,\eta,t)\,\mathrm{d}\eta\,\mathrm{d}\xi, \quad I_{j,k} := [x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}}] \times [y_{k-\frac{1}{2}}, y_{k+\frac{1}{2}}]. \tag{2.30}$$

The cell averages are evolved in time according to the semi-discrete finite-volume scheme:

$$\frac{\mathrm{d}}{\mathrm{d}t}\overline{\mathbf{w}}_{j,k}(t) = -\frac{\mathbf{H}^x_{j+\frac{1}{2},k}(t) - \mathbf{H}^x_{j-\frac{1}{2},k}(t)}{\Delta x} - \frac{\mathbf{H}^y_{j,k+\frac{1}{2}}(t) - \mathbf{H}^y_{j,k-\frac{1}{2}}(t)}{\Delta y}, \tag{2.31}$$

where the numerical flux functions are $\mathbf{H}^x_{j+\frac{1}{2},k}(t) = \mathcal{H}^x(\mathbf{w}^{\mathrm{NE}}_{j,k}, \mathbf{w}^{\mathrm{E}}_{j,k}, \mathbf{w}^{\mathrm{SE}}_{j,k}; \mathbf{w}^{\mathrm{NW}}_{j+1,k}, \mathbf{w}^{\mathrm{W}}_{j+1,k}, \mathbf{w}^{\mathrm{SW}}_{j+1,k})$ and $\mathbf{H}^y_{j,k+\frac{1}{2}}(t) = \mathcal{H}^y(\mathbf{w}^{\mathrm{NW}}_{j,k}, \mathbf{w}^{\mathrm{N}}_{j,k}, \mathbf{w}^{\mathrm{NE}}_{j,k}; \mathbf{w}^{\mathrm{SW}}_{j,k+1}, \mathbf{w}^{\mathrm{S}}_{j,k+1}, \mathbf{w}^{\mathrm{SE}}_{j,k+1})$. Here, the point values in the middle of cell boundaries, $\mathbf{w}^{E(W,S,N)}$, and at the sell corners, $\mathbf{w}^{NE(NW,SE,SW)}$, are to be computed from a conservative, (essentially) non-oscillatory piecewise polynomial reconstruction. Second-order schemes typically utilize piecewise linear reconstructions:

$$\widetilde{\mathbf{w}}(x,y,t) = \overline{\mathbf{w}}_{j,k} + (\mathbf{w}_x)_{j,k}(x - x_j) + (\mathbf{w}_y)_{j,k}(y - y_k), \quad \text{for } (x,y) \in I_{j,k}, \tag{2.32}$$

and then the corresponding point values are:

$$\mathbf{w}^{\mathrm{E(W)}}_{j,k} := \overline{\mathbf{w}}_{j,k} \pm \tfrac{\Delta x}{2}(\mathbf{w}_x)_{j,k}, \quad \mathbf{w}^{\mathrm{N(S)}}_{j,k} := \overline{\mathbf{w}}_{j,k} \pm \tfrac{\Delta y}{2}(\mathbf{w}_y)_{j,k},$$

$$\mathbf{w}^{\mathrm{NE(NW)}}_{j,k} := \overline{\mathbf{w}}_{j,k} \pm \tfrac{\Delta x}{2}(\mathbf{w}_x)_{j,k} + \tfrac{\Delta y}{2}(\mathbf{w}_y)_{j,k}, \quad \mathbf{w}^{\mathrm{SE(SW)}}_{j,k} := \overline{\mathbf{w}}_{j,k} \pm \tfrac{\Delta x}{2}(\mathbf{w}_x)_{j,k} - \tfrac{\Delta y}{2}(\mathbf{w}_y)_{j,k}. \tag{2.33}$$

From now on, we will simplify the notation as in the 1-D case: time dependence of all computed and reconstructed quantities will be omitted.

To ensure a non-oscillatory property of the reconstruction (2.32), the slopes should be computed with the help of a nonlinear limiter. In our numerical experiments, we have used a 2-D extension of the generalized minmod limiter (2.7):

$$(\mathbf{w}_x)_{j,k} = \mathrm{minmod}\left(\theta\frac{\overline{\mathbf{w}}_{j+1,k} - \overline{\mathbf{w}}_{j,k}}{\Delta x}, \frac{\overline{\mathbf{w}}_{j+1,k} - \overline{\mathbf{w}}_{j-1,k}}{2\Delta x}, \theta\frac{\overline{\mathbf{w}}_{j,k} - \overline{\mathbf{w}}_{j-1,k}}{\Delta x}\right),$$

$$(\mathbf{w}_y)_{j,k} = \mathrm{minmod}\left(\theta\frac{\overline{\mathbf{w}}_{j,k+1} - \overline{\mathbf{w}}_{j,k}}{\Delta y}, \frac{\overline{\mathbf{w}}_{j,k+1} - \overline{\mathbf{w}}_{j,k-1}}{2\Delta y}, \theta\frac{\overline{\mathbf{w}}_{j,k} - \overline{\mathbf{w}}_{j,k-1}}{\Delta y}\right), \tag{2.34}$$

where $\theta \in [1,2]$.

2.3.1. *Interface treatment*

The extension of the 1-D interface treatment algorithm to 2-D is essentially in a dimension-by-dimension manner. However, since the geometry of the interface in 2-D is more complex, the extension is not straightforward as it needs to account for various possible 2-D scenarios.

We use a level-set formulation to determine the interface geometry. Within this framework, we solve equation (1.4) for the level set function ϕ, where ϕ is positive for one fluid, negative for the other and zero at the interface. Given the point values of $\phi_{j,k} := \phi_{j,k}(t) \approx \phi(x_j, y_k, t)$, if $\phi_{j,k}$ changes sign across any one of the cell interfaces (that is if either $\phi_{j,k}\phi_{j+1,k} < 0$, $\phi_{j,k}\phi_{j-1,k} < 0$, $\phi_{j,k}\phi_{j,k+1} < 0$, or $\phi_{j,k}\phi_{j,k-1} < 0$), then the cell (j,k) is flagged as mixed and is marked by 0. Otherwise, the cell (j,k) is marked by $\mathrm{sign}(\phi_{j,k}(t))$ and the fluid parameters γ

K	**1**	**0**	**−1**	K	**−1**	**0**	**1**	K	**1**	**0**	**1**	K	**−1**	**0**	**−1**
	J−1	*J*	*J+1*		*J−1*	*J*	*J+1*		*J−1*	*J*	*J+1*		*J−1*	*J*	*J+1*
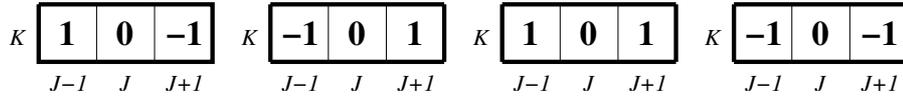
FIGURE 5. Mixed (J, K) cell and its single-fluid neighbors in the x-direction – configurations, for which the 1-D approach can be directly extended.

and p_∞ there are set accordingly. We further distinguish between mixed cells and denote by 0+ mixed cells with $\phi_{j,k} > 0$, and by 0− cells with $\phi_{j,k} < 0$.

In the following, we assume that cell (J, K) is mixed, and describe the algorithm for the computation of the numerical fluxes $\mathbf{H}^x_{J+\frac{1}{2},K}$ and $\mathbf{H}^x_{J-\frac{1}{2},K}$ ($\mathbf{H}^y_{J,K+\frac{1}{2}}$ and $\mathbf{H}^y_{J,K-\frac{1}{2}}$ are computed in a similar manner).

As in the 1-D case, we consider cell averages of mixed cells not to be physically meaningful and seek to interpolate the required point values in cell (J, K) by solving the Riemann problem between cells $(J − 1, K)$ and $(J + 1, K)$. However, the 2-D situation is more complicated since cells $(J − 1, K)$ and $(J + 1, K)$ are not necessarily *single-fluid* ones. If one or both of these cells are mixed, we will replace the cell averages there with the values interpolated from the nearest single-fluid cells and solve the corresponding Riemann problem. We now turn to a detailed description of the algorithm.

We first observe that the 1-D algorithm described in Section 2.1.1, can be straightforwardly extended to two space dimensions in the cases shown in Figure 5. We note that the two cases illustrated on the right are not likely to arise is 1-D computations. Nonetheless, (J, K) is a mixed cell, and as in the 1-D methodology we solve a Riemann problem between its left/right neighboring cells, which in these two case happen to be occupied by the same fluid.

As it has been already mentioned, in 2-D, it is possible for the left/right neighbor of a mixed cell to be also mixed, for example, if the interface is close to horizontal (respectively vertical). We consider in detail the case illustrated in Figure 6, where cell (J, K) is a 0+ cell, the right neighboring cell $(J + 1, K)$ is mixed. Other cases follow in complete analogy and are not described. We seek to replace the cell average in the mixed cell $(J+1, K)$ by extrapolating information from neighboring cells, and need to determine which of the neighboring cell averages to use in the extrapolation. If cell $(J+1, K)$ is a 0+ cell, we use its single-fluid neighbors which are marked by 1 (provided there is at least one such a neighbor). An example of a configuration that corresponds to this case is shown in Figure 6a. In such a situation, we replace $\overline{\mathbf{w}}_{J+1,K}$ with the weighted average

$$\frac{C_\mathrm{d}\overline{\mathbf{w}}_{J,K+1} + C_y\overline{\mathbf{w}}_{J+1,K+1} + C_\mathrm{d}\overline{\mathbf{w}}_{J+2,K+1} + C_x\overline{\mathbf{w}}_{J+2,K}}{2C_\mathrm{d} + C_x + C_y}, \tag{2.35}$$

where the weights, C_x, C_y and C_d, are inversely proportional to the distance between the single-fluid cells participating in (2.35) and cell $(J + 1, K)$, that is,

$$C_x = \frac{1}{\Delta x}, \quad C_y = \frac{1}{\Delta y}, \quad C_\mathrm{d} = \frac{1}{\sqrt{(\Delta x)^2 + (\Delta y)^2}}, \tag{2.36}$$

and we also set the fluid parameters γ and p_∞ in cell $(J + 1, K)$ to correspond to the fluid marked by 1. If none of the neighbors of this cell is marked by 1, then we interpolate using the values from the neighbors marked by −1. An example of such a situation is illustrated in Figure 6b. In this particular example, we replace $\overline{\mathbf{w}}_{J+1,K}$ with the following weighted average

$$\frac{C_\mathrm{d}\overline{\mathbf{w}}_{J+2,K+1} + C_x\overline{\mathbf{w}}_{J+2,K} + C_\mathrm{d}\overline{\mathbf{w}}_{J+2,K-1} + C_y\overline{\mathbf{w}}_{J+1,K-1} + C_\mathrm{d}\overline{\mathbf{w}}_{J,K-1}}{3C_\mathrm{d} + C_x + C_y},$$

where the weights, C_x, C_y and C_d, are still given by (2.36), and the parameters γ and p_∞ in cell $(J + 1, K)$ are set to correspond to the fluid marked by −1.
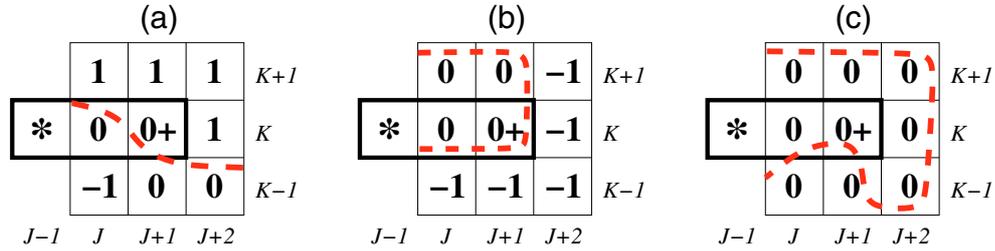
FIGURE 6. Typical cell configurations around mixed (J, K) and $(J+1, K)$ cells ($\phi_{J+1,K} > 0$).
The dashed line represents a possible interface path.

If all the surrounding cells are marked by 0 (see Fig. 6c), which may occur if the interface is highly curved on the scale of the grid size, it might be hard or even impossible to accurately interpolate the cell average values of $\overline{\mathbf{w}}_{J+1,K}$. In this case, we simply look for the nearest cell with single-fluid data.

Once $\overline{\mathbf{w}}_{J+1,K}$ and $\overline{\mathbf{w}}_{J-1,K}$ have been replaced by suitably extrapolated single-fluid data, we proceed according to the 1-D algorithm described in Section 2.1.1, and solve the x-direction Riemann problem between them. Then, as in Section 2.1.1, we use the obtained solution to evaluate the point values $\mathbf{w}_{J,K}^{\mathrm{W}}$ and $\mathbf{w}_{J,K}^{\mathrm{E}}$, needed for the computation of the x-slopes in the neighboring single-fluid cells and for the evaluation of x-numerical fluxes at $(x_{J-\frac{1}{2}}, y_K)$ and $(x_{J+\frac{1}{2}}, y_K)$.

The level set equation (1.4) is solved by an upwind finite-difference method with the corresponding velocities to be:

$$(u_{j,k}, v_{j,k}) = \begin{cases} ((\overline{\rho u})_{j,k}/\overline{\rho}_{j,k}, (\overline{\rho v})_{j,k}/\overline{\rho}_{j,k}), & \text{if } (j,k) \text{ is a single-fluid cell,} \\ (u_{j,k}^*, v_{j,k}^*), & \text{if } (j,k) \text{ is a mixed cell,} \end{cases}$$

where $u_{J,K}^*$ and $v_{J,K}^*$ are the intermediate x- and y-velocities obtained when solving Riemann problems in the x- and y-direction, respectively.

Finally, when the interface crosses a cell boundary, a cell status may change from mixed to single-fluid. We consider the following 3 typical scenarios illustrated in Figure 7. In the cases illustrated in Figures 7a and 7b, we conclude that the interface has moved from the cell (J, K) to the right, and thus we correct the values $\overline{\mathbf{w}}_{J,K}$ and $\overline{\mathbf{w}}_{J+1,K}$ according to the following conservation algorithm (compare with (2.12)):

$$\overline{\mathbf{w}}_{J,K}^{\mathrm{new}} = (\mathbf{w}_L^*)^x, \quad \overline{\mathbf{w}}_{J+1,K}^{\mathrm{new}} = \overline{\mathbf{w}}_{J,K} + \overline{\mathbf{w}}_{J+1,K} - \overline{\mathbf{w}}_{J,K}^{\mathrm{new}},$$

where $(\mathbf{w}_L^*)^x$ is the left intermediate state of the (approximate) solution of the 1-D Riemann problem in the x-direction between (the interpolated values of) $\overline{\mathbf{w}}_{J-1,K}$ and $\overline{\mathbf{w}}_{J+1,K}$, see Figure 1. Notice that the new value $\mathbf{w}_{J+1,K}^{\mathrm{new}}$ complies with the conservation requirements, that is, $\overline{\mathbf{w}}_{J+1,K}^{\mathrm{new}} + \overline{\mathbf{w}}_{J,K}^{\mathrm{new}} = \overline{\mathbf{w}}_{J,K} + \overline{\mathbf{w}}_{J+1,K}$. In the case shown in Figure 7c, we conclude that the interface has moved in the down-right direction and thus the above correction procedure should be modified, for instance, as follows:

$$\overline{\mathbf{w}}_{J,K}^{\mathrm{new}} = \frac{(\mathbf{w}_L^*)^x + (\mathbf{w}_R^*)^y}{2}, \quad \overline{\mathbf{w}}_{J+1,K}^{\mathrm{new}} = \frac{\overline{\mathbf{w}}_{J,K}}{2} + \overline{\mathbf{w}}_{J+1,K} - \frac{\overline{\mathbf{w}}_{J,K}^{\mathrm{new}}}{2}, \quad \overline{\mathbf{w}}_{J,K-1}^{\mathrm{new}} = \frac{\overline{\mathbf{w}}_{J,K}}{2} + \overline{\mathbf{w}}_{J,K-1} - \frac{\overline{\mathbf{w}}_{J,K}^{\mathrm{new}}}{2},$$

where $(\mathbf{w}_L^*)^x$ is the same as in the previous formula and $(\mathbf{w}_R^*)^y$ is the right intermediate state of the (approximate) solution of the 1-D Riemann problem in the y-direction between (the interpolated values of) $\overline{\mathbf{w}}_{J,K-1}$ and $\overline{\mathbf{w}}_{J,K+1}$. Once again, the new cell average values respect the multi-block conservation in the sense that $\overline{\mathbf{w}}_{J,K}^{\mathrm{new}} + \overline{\mathbf{w}}_{J+1,K}^{\mathrm{new}} + \overline{\mathbf{w}}_{J,K-1}^{\mathrm{new}} = \overline{\mathbf{w}}_{J,K} + \overline{\mathbf{w}}_{J+1,K} + \overline{\mathbf{w}}_{J,K-1}$.
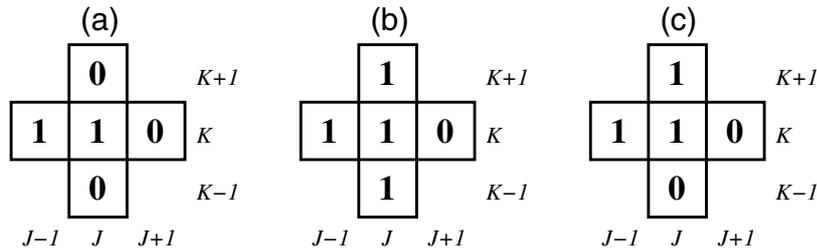
FIGURE 7. Typical cell configurations around cell (J, K), whose status has changed from the mixed to the single-fluid one after the latest time evolution step.

3. NUMERICAL EXAMPLES

In this section, we illustrate the performance of the interface tracking method by a number of 1-D and 2-D numerical experiments. In all examples, we use the semi-discrete second-order central-upwind schemes described in Appendix A. Away from material interfaces, the method employs either the generalized minmod reconstruction (2.6)–(2.7) with $\theta = 1.3$ (in the 1-D and radial symmetric cases) or its 2-D extension (2.32)–(2.34). To obtain the intermediate states needed for reconstructing required point values in the vicinity of the material interface, we use either approximate or exact Riemann problem solvers. The approximate solver is based on a local linearization of the governing equations about an average state, which has been taken as the arithmetic average of the single-fluid cell averages from both sides of the interface. The time integration is performed using the third-order SSP-RK method [18].

The interface tracking methods based on the exact and linearized Riemann problem solvers will be referred to as IT-EXACT and IT-LIN methods, respectively.

3.1. **One-dimensional examples**

Example 3.1.1 – Shock-tube problem

We first consider the system (2.1) subject to the initial conditions corresponding to a shock-tube problem:

$$(\rho, u, p; \gamma, p_\infty)^T = \begin{cases} (1.000, 0, 1.0; 1.4, 0)^T, & x < 0.5, \\ (0.125, 0, 0.1; 1.6, 0)^T, & x > 0.5. \end{cases}$$

We compute the solution of this IVP at time $t = 0.2$ on a uniform grid with $\Delta x = 5 \times 10^{-3}$. The densities obtained by the IT-LIN and IT-EXACT methods are shown in Figure 8. The results are very similar though the IT-EXACT method provides a slightly better resolution of the contact wave. There is almost no difference in the resolution of the velocity and pressure achieved by both methods. Therefore, in Figure 9 we plot u and p obtained by the IT-LIN method only. As one can see from Figures 8–9, the numerical solutions (dotted lines) are in a very good agreement with the corresponding exact solutions (solid lines). We note that in this example, our interface tracking method tends to capture the material front as a perfect jump. As our next examples demonstrate, a perfect resolution of material fronts is not guaranteed by the method, but in our experience, almost all contact discontinuities are captured in a sharp manner (similar to [10,11]). This property is particularly nice when the EOS of the fluid may not be valid for all flow regimes (as is the case with the Tait EOS) and may not be valid for intermediate values produced by a smeared material front.

ρ by IT–LIN

ρ by IT–EXACT
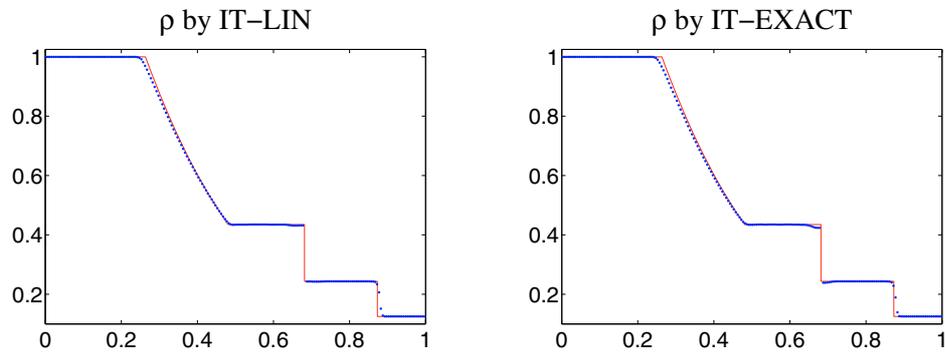
FIGURE 8. Solution (density) of the shock-tube problem by the IT-LIN and IT-EXACT methods.

u

p
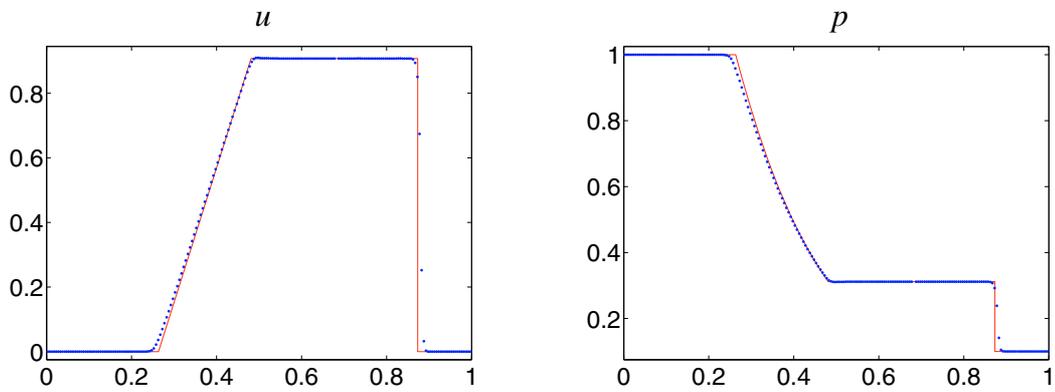
FIGURE 9. Solution (velocity and pressure) of the shock-tube problem by the IT-LIN method.

ρ by IT–LIN

ρ by IT–EXACT
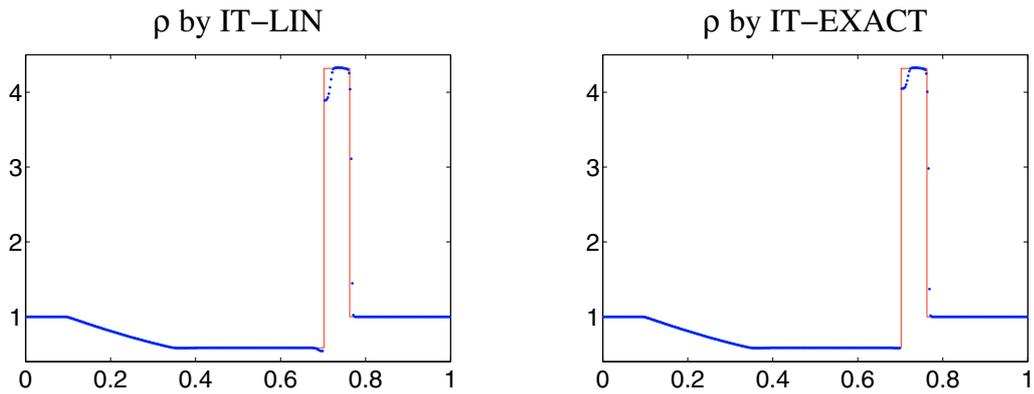
FIGURE 10. Solution (density) of the stiff shock-tube problem by the IT-LIN and IT-EXACT methods.
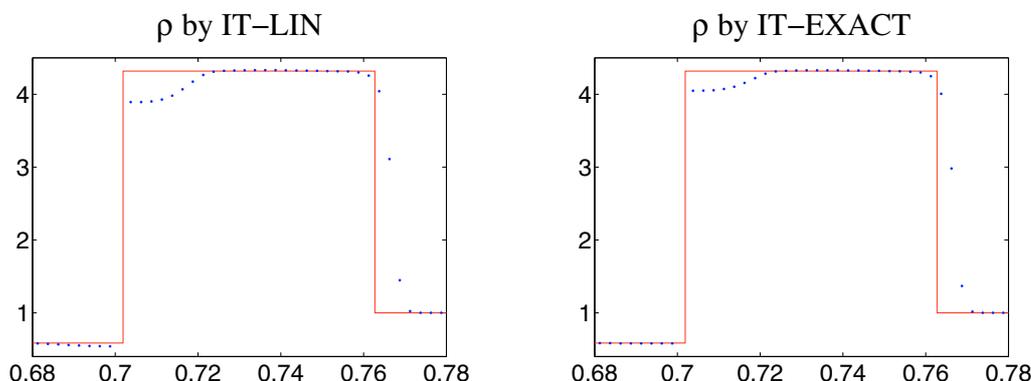
FIGURE 11. Solution (density) of the stiff shock-tube problem by the IT-LIN and IT-EXACT methods – zoom at $x \in [0.68, 0.78]$.
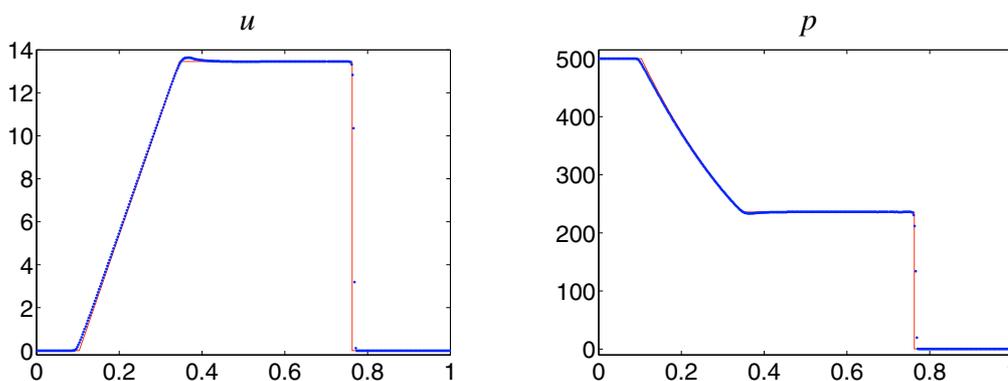
FIGURE 12. Solution (velocity and pressure) of the stiff shock-tube problem by the IT-LIN method.

Example 3.1.2 – Stiff shock-tube problem

We now turn to another shock-tube problem with stiffer initial data [4]:

$$(\rho, u, p; \gamma, p_\infty)^T = \begin{cases} (1.0, 0.0, 500; 1.4, 0)^T, & x < 0.5, \\ (1.0, 0.0, 0.2; 1.6, 0)^T, & x > 0.5. \end{cases} \tag{3.1}$$

The solutions of the IVP (2.1)-(3.1) at time $t = 0.015$ are plotted in Figures 10–12. The densities computed by the IT-LIN and IT-EXACT methods are shown in Figure 10 (see also zoom at the area of contact and shock discontinuities shown in Fig. 11) and the velocity and the pressure computed by the IT-LIN method are plotted in Figure 12 (the resolution of these fields achieved by the IT-EXACT method is similar). In all these figures, the dots represent the solutions computed on a uniform grid with $\Delta x = 2.5 \times 10^{-3}$, and the solid lines represent the corresponding exact solutions. As one can see from Figures 10 and 11, the IT-EXACT method achieves a somewhat better resolution of the contact discontinuity then the IT-LIN method, but both versions of our interface tracking method clearly demonstrates their ability to accurately capture strong contact waves. In particular, we note that all computed solutions are oscillation-free and the results obtained on the coarser grid are in excellent agreement with the reference solutions.
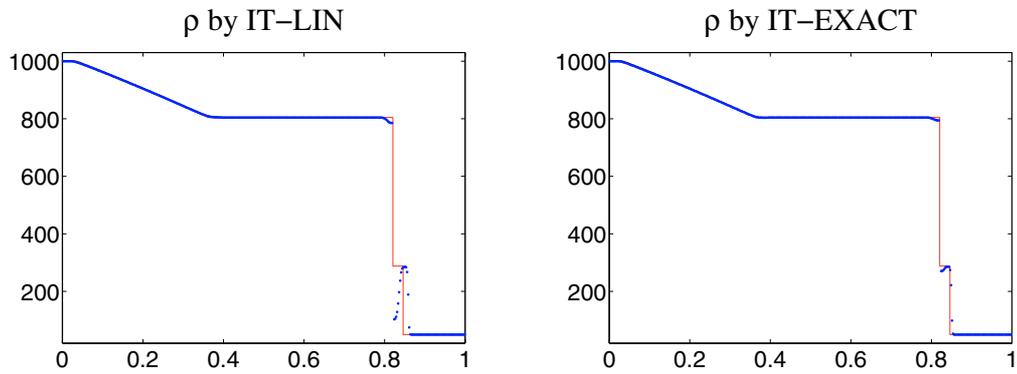
FIGURE 13. Solution (density) of the gas-liquid shock-tube problem by the IT-LIN and IT-EXACT methods.
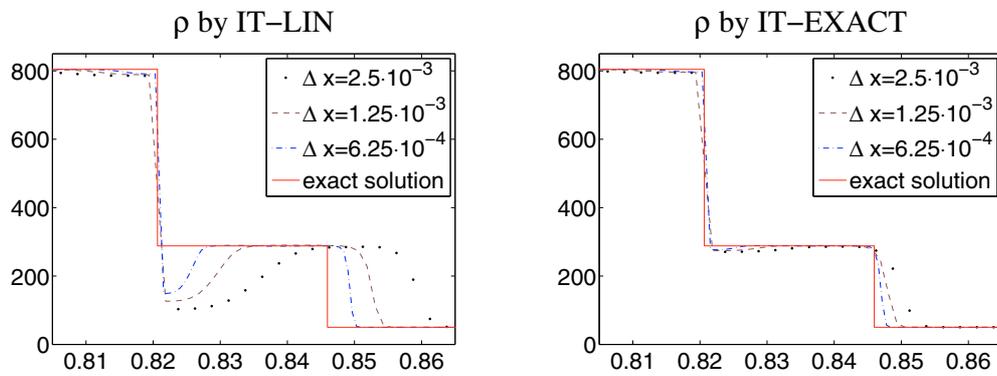
FIGURE 14. Solution (density) of the gas-liquid shock-tube problem by the IT-LIN and IT-EXACT methods – zoom at $x \in [0.805, 0.865]$.
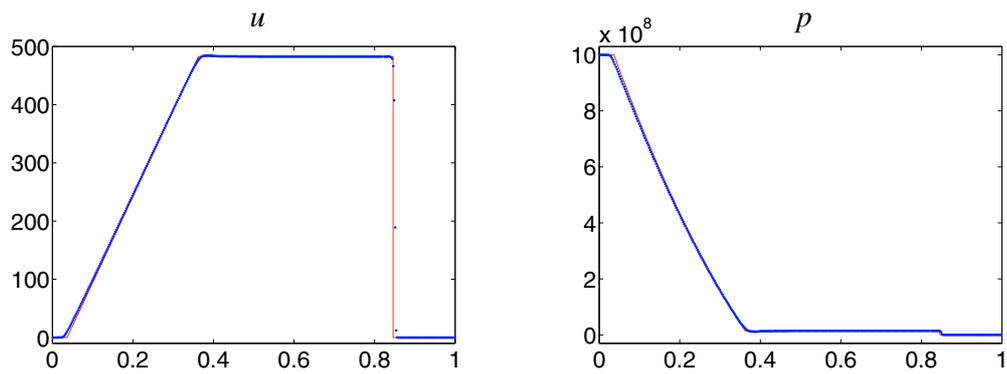
FIGURE 15. Solution (velocity and pressure) of the gas-liquid shock-tube problem by the IT-EXACT method.
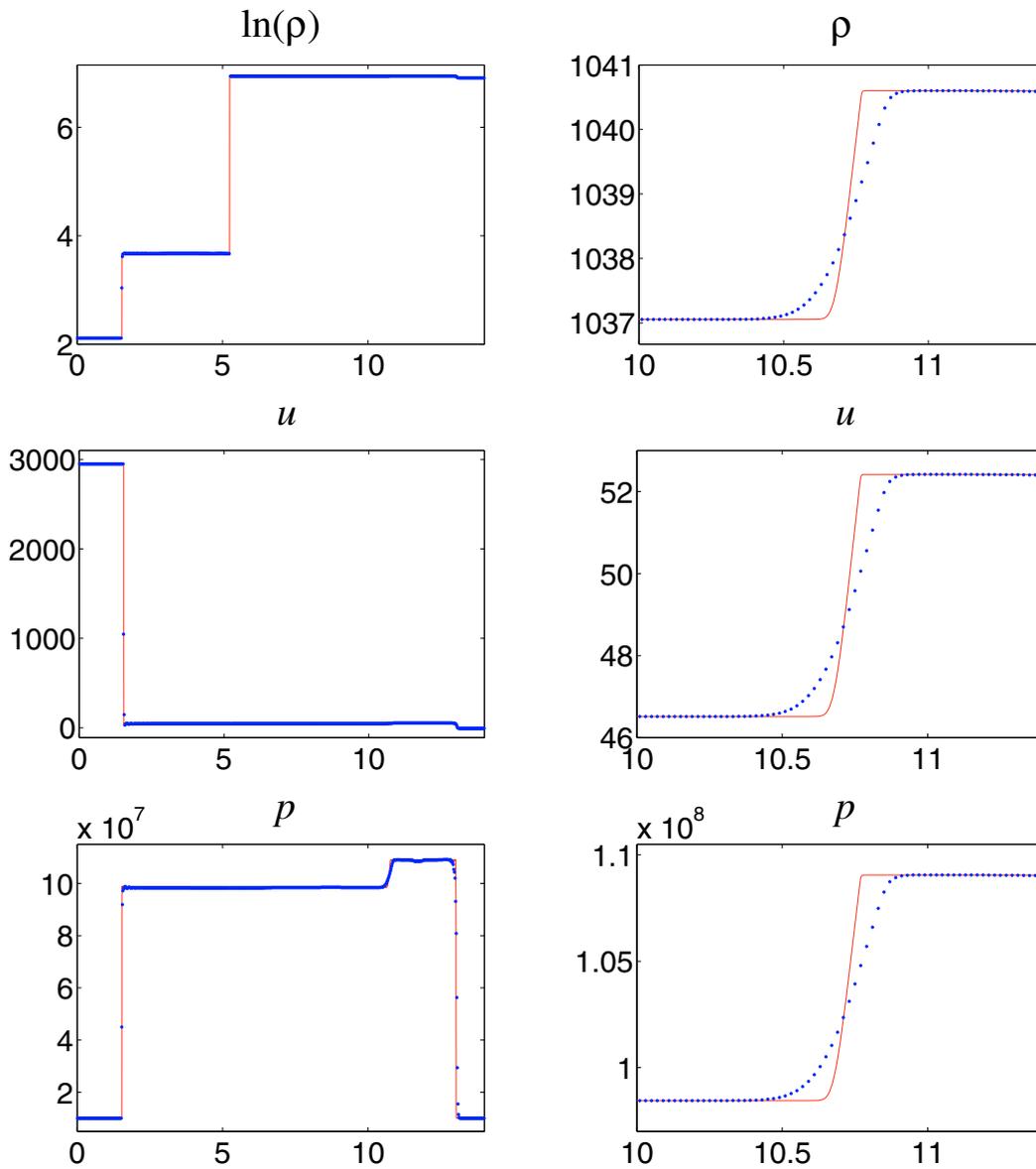
FIGURE 16. Solution of the gas-liquid problem by the Tait EOS for the water at $t = 0.006$. The graphs on the right zoom at the region $x \in [10, 11.4]$. The dots represent the solution computed on the coarser grid and the solid line corresponds to the reference solution.

Example 3.1.3 – Water-air model using the stiff equation of state

We now consider a gas-liquid multifluid system and model the liquid component using the stiff EOS (1.2). The initial conditions correspond to a severe water-air shock tube problem and are given by:

$$(\rho, u, p; \gamma, p_\infty)^T = \begin{cases} (1000, 0, 10^9; 4.4, 6 \cdot 10^8)^T, & x < 0.7, \\ (50, 0, 10^5; 1.4, 0)^T, & x > 0.7. \end{cases} \tag{3.2}$$
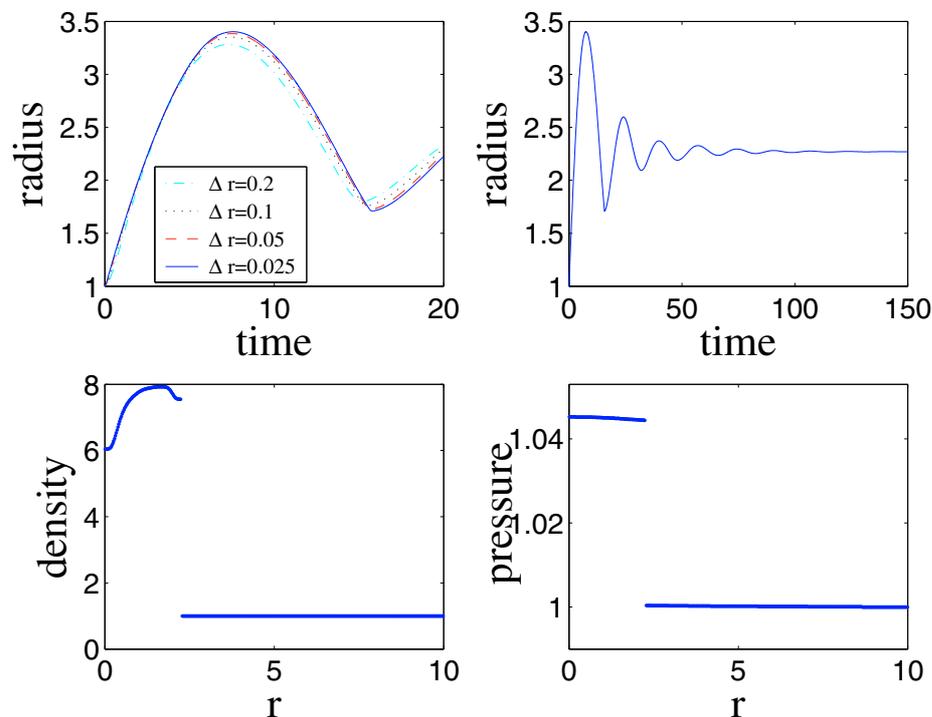
FIGURE 17. Bubble radius *vs.* time: grid convergence (upper left) and time convergence (upper right). Density (lower left) and pressure (lower right) at the new equilibrium, computed by the IT-LIN method.

We apply both the IT-LIN and IT-EXACT methods to the IVP (2.1)-(3.2). In Figures 13 and 15, the numerical solutions at time $t = 0.00025$, computed on an uniform (with $\Delta x = 2.5 \times 10^{-3}$, dots) grid are plotted along with the corresponding exact solutions (solid lines). As in the previous examples, we show the densities obtained by both the IT-LIN and IT-EXACT methods, while the velocity and the pressure are shown only once. One can also observe, the IT-EXACT method outperforms the IT-LIN one though both methods seem to converge to the correct solution as we show in Figure 14. In this figure, we zoom at the shock and contact waves area of the densities computed on four different uniform grids with $\Delta x = 2.5 \times 10^{-3}$, $\Delta x = 1.25 \times 10^{-3}$, and $\Delta x = 6.25 \times 10^{-4}$.

We note that the advantage of the IT-EXACT method over the IT-LIN one is more pronounced in this example than in the previous one, which is likely related to the stiffness of the problem.

Example 3.1.4 – Water-air model using the Tait equation of state

As in the previous example, we consider a gas-liquid system. The gas component is modeled using the full gas dynamics system (2.1)-(1.2) with $p_\infty = 0$ and $\gamma = 1.25$, while the water component is now modeled using (2.13)–(2.14) with $\gamma = 7.15$, $A = 10^5$, $B = 3.31 \times 10^8$, and $\rho_0 = 1000$. We apply the IT-EXACT method to this

system subject to the initial data containing a shock in each fluid:

$$(\rho, u, p) = \begin{cases} (8.26605505, 2949.97131, 10^7), & x < 1, \\ (1, 0, 10^5), & 1 < x < 5, \end{cases}$$

$$(\rho, u) = \begin{cases} (1000, 0), & 5 < x < 9.6, \\ (1004.1303, -6.3813588), & x > 9.6. \end{cases}$$

The interface is initially located at $x = 5$ with a gas on the left and water on the right. This problem is similar to the one from [11,48], where an initial-boundary value problem with the same initial data was considered (here, we solve the IVP only).

We take two uniform grids with $\Delta x = 0.02$ and $\Delta x = 0.002$ (the solution computed on the finer grid serves as a reference solution). The computed solutions at time $t = 0.006$ are shown in Figure 16, where the graphs on the right zoom at the region $x \in [10, 11.4]$. One can clearly see that the proposed interface tracking method, described in Section 2.1.2, sharply captures the contact wave and provides a high overall quality of the computed solution despite some small oscillations.

3.2. Radially symmetric oscillating bubble

We apply the radial version of the IT-LIN method, presented in Section 2.2, to the example taken from [25]. We consider an oscillating compressible cylindrical bubble, whose motion is governed by the system (2.15)–(2.17). Initially, the radius of the bubble is $R = 1$ and the data inside/outside the bubble are:

$$(\rho, u, p; \gamma, p_\infty)(r, 0) = \begin{cases} (40, 0, 10; 1.4, 0), & r < 1, \\ (1, 0, 1; 1.67, 2), & r > 1. \end{cases}$$

The surface tension coefficient is $\tau = 0.1$.

Due to a large initial pressure difference, the bubble starts oscillating. Each oscillation dissipates energy by radiating acoustic waves away from the bubble and this causes the convergence towards a new equilibrium, as illustrated in Figure 17 (upper right), where we plot the computed bubble radius $vs.$ time. A grid convergence is demonstrated in Figure 17 (upper left). Finally, the lower two graphs in Figure 17 show the density and pressure, computed at the new equilibrium (at $t = 200$) on a uniform grid with $\Delta r = 0.025$. As in the 1-D case, the contact surface is sharply resolved. At the same time, our results are in good agreement with the results reported in [25].

3.3. Two-dimensional examples

In this section, we compute the interaction between a shock wave and a bubble by a 2-D IT-LIN method described in Section 2.3. A weak shock wave in air (shock Mach number ≈ 1.19) hits a bubble at rest, containing either Helium (Ex. 3.3.1) or R22 (Ex. 3.3.2). With Helium being lighter than air, and R22 being heavier, wave refraction at the bubble interface and the resulting wave structures are very different. The examples follow experiments reported in [19] and computationally studied in [36]. The fluids are modeled as ideal gases using (1.2) with $p_\infty = 0$, and have different values of specific heat ratio γ. A schematic of the initial setup is shown in Figure 18. The top and bottom boundaries are solid walls, the left and right boundaries are open, with the domain set to be large enough so that waves do not reach the boundaries by the final time of the computation. In both examples, we have used a uniform spatial grid with $\Delta x = \Delta y = 0.02$, yielding an effective grid of 25×25 over the initial bubble. Figures 19–22, show Schlieren images of the numerical results displaying the magnitude of the density gradient field, $|\nabla \rho|$, for different times during the interaction. In these figures, we have used the following shading function:

$$\exp\left(-\frac{C|\nabla \rho|}{\max(|\nabla \rho|)}\right), \quad C = 300,$$

where the numerical density derivatives are computed using central differencing.
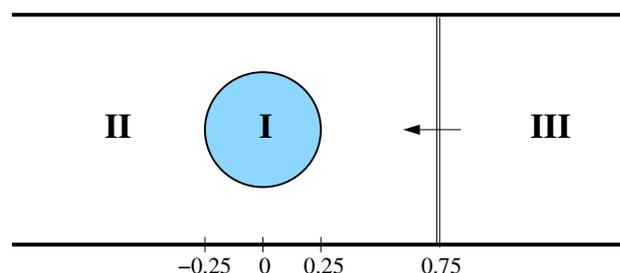
FIGURE 18. Initial setting for the 2-D numerical examples.

Example 3.3.1 – Helium bubble

We numerically solve the system (1.1)–(1.2) with the following initial conditions:

$$(\rho, u, v, p; \gamma, p_\infty) = \begin{cases} (4/29, 0, 0, 1; 5/3, 0), & \text{in region I,} \\ (1, 0, 0, 1; 1.4, 0), & \text{in region II,} \\ (4/3, -0.3535, 0, 1.5; 1.4, 0), & \text{in region III,} \end{cases}$$

where regions I, II, and III are shown in Figure 18.

Different stages if the shock-bubble interaction are shown in Figures 19 and 20. Upon hitting the bubble, the shock wave partly refracts, and partly reflects. The refracted shock is curved and propagates faster inside the helium bubble than in the air, as sound speed in helium is higher than sound speed in air. The bubble compresses under the impact of the shock and is set into motion. Being lighter than air, the helium accelerates more under the impact of the shock, and shear is produced across the bubble interface. The refracted shock sweeps over the bubble, and continues to interacts with the bubble interface, giving rise to highly complicated refraction patterns. After the shock completes its sweep over the bubble, the bubble continues to evolve slowly into a familiar kidney shape as a result of the vorticity generated at the bubble interface, before losing its integrity and breaking up. Interface roll-up due to shear instability may be observed, as well as wave reflection from the solid top/bottom boundaries. The results are in good agreement with the results reported in [36].

Example 3.3.2 – R22 bubble

We numerically solve the system (1.1)–(1.2) subject to the following initial conditions:

$$(\rho, u, v, p; \gamma, p_\infty) = \begin{cases} (3.1538, 0, 0, 1; 1.249, 0), & \text{in region I,} \\ (1, 0, 0, 1; 1.4, 0), & \text{in region II,} \\ (4/3, -0.3535, 0, 1.5; 1.4, 0), & \text{in region III,} \end{cases}$$

where, as in the previous example, regions I, II, and III are specified in Figure 18.

In Figures 21 and 22, we show different stages if the shock-bubble interaction. Upon being hit by the shock wave, the bubble compresses and undergoes a deformation, and the shock partly reflects and partly refracts. The speed of sound inside the R22 bubble is lower than outside, causing the refracted shock wave to move more slowly than the shock outside the bubble. The bubble is heavier than the surrounding air, and accelerates less than the air under the impact of the shock. As a result, vorticity is generated at the bubble interface. The refracted shock can be observed to focus inside the bubble, causing a rise in pressure and resulting in a visible forward jet along the center of the bubble. The shear at the boundary interface causes the interface to roll-up, as the bubble continues to evolve under the induced vorticity field. The results are in good agreement with the results reported in [36].
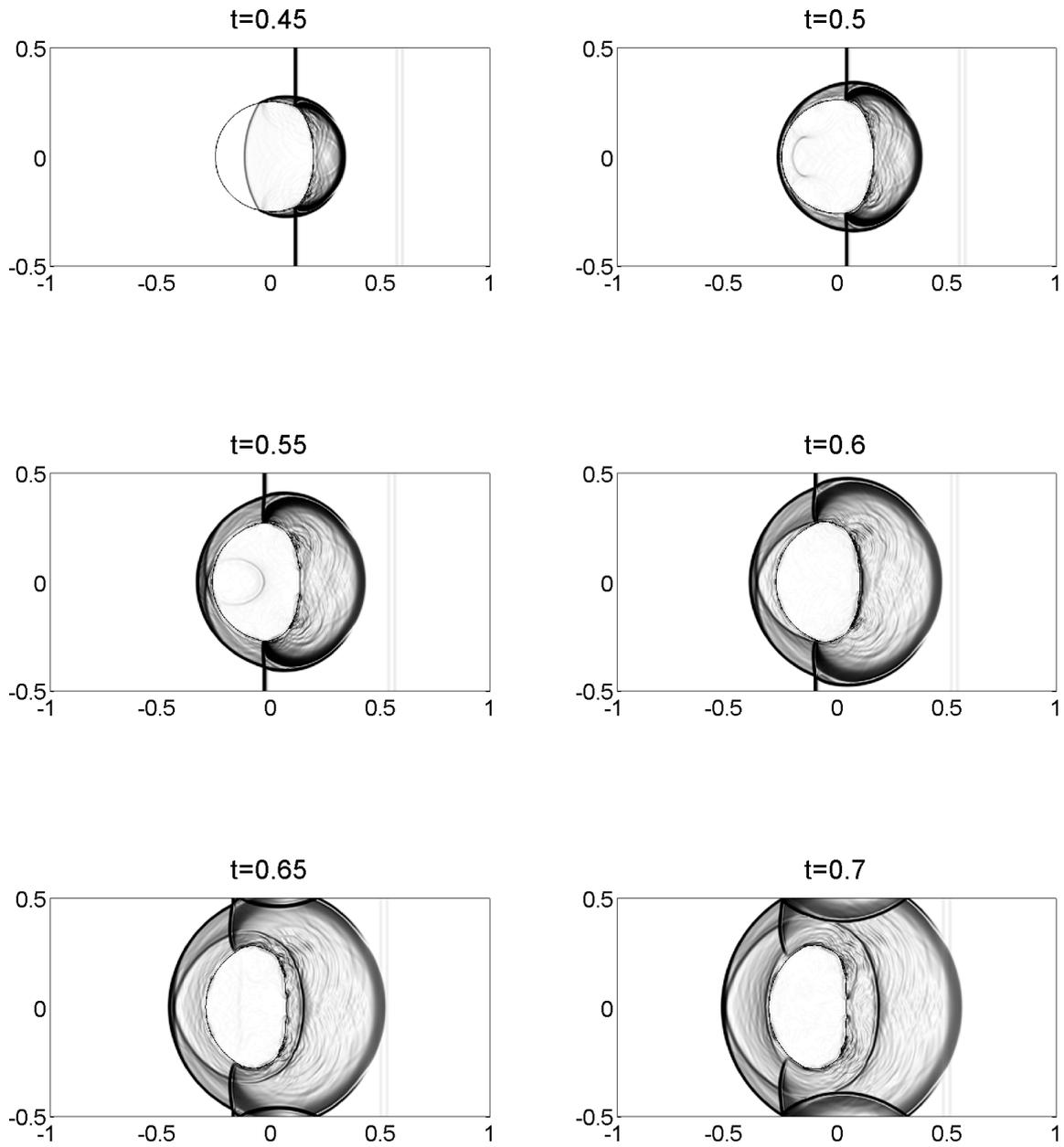
FIGURE 19. Shock-helium bubble interaction. Earlier times.
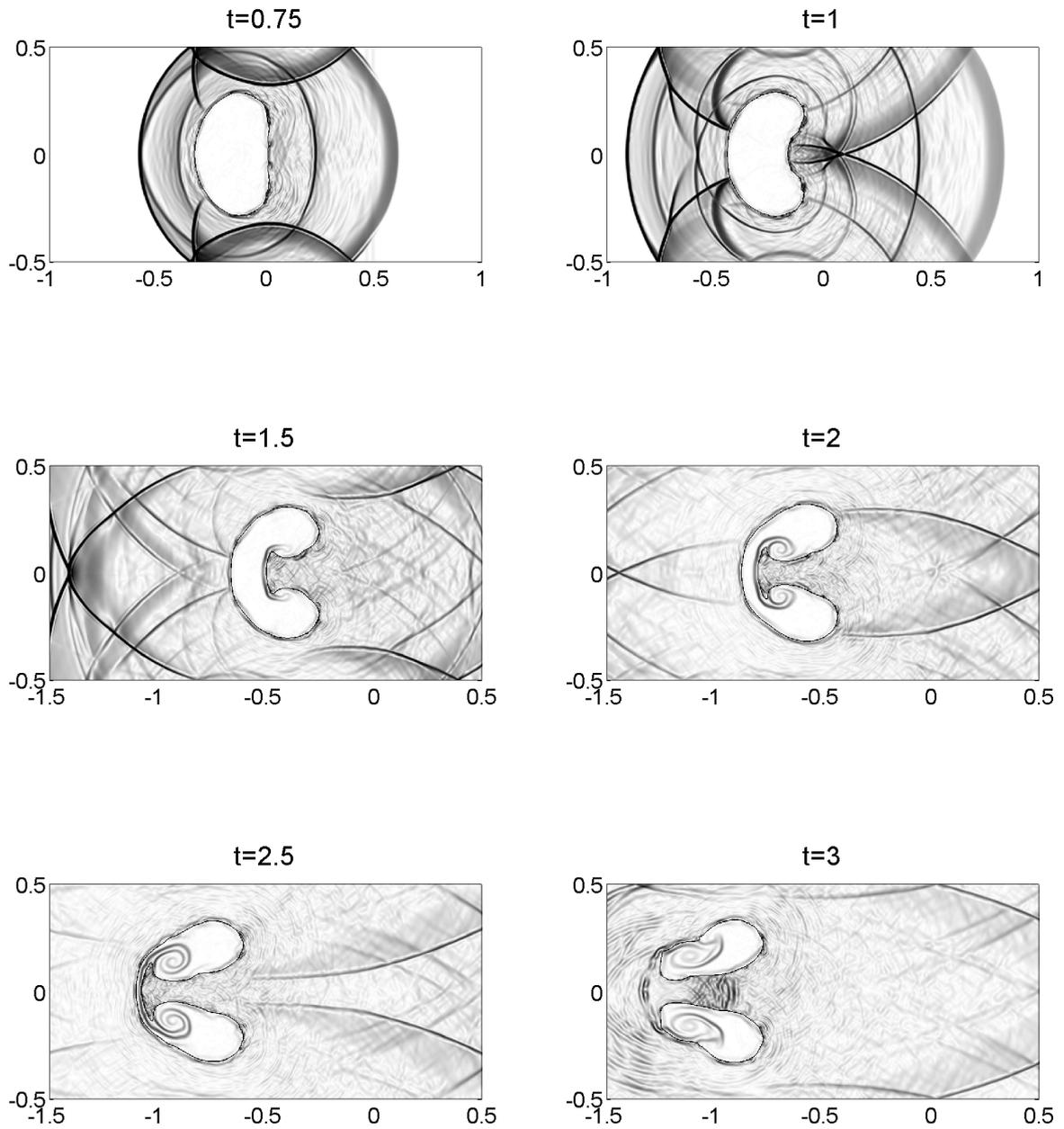
FIGURE 20. Shock-helium bubble interaction. Later times.
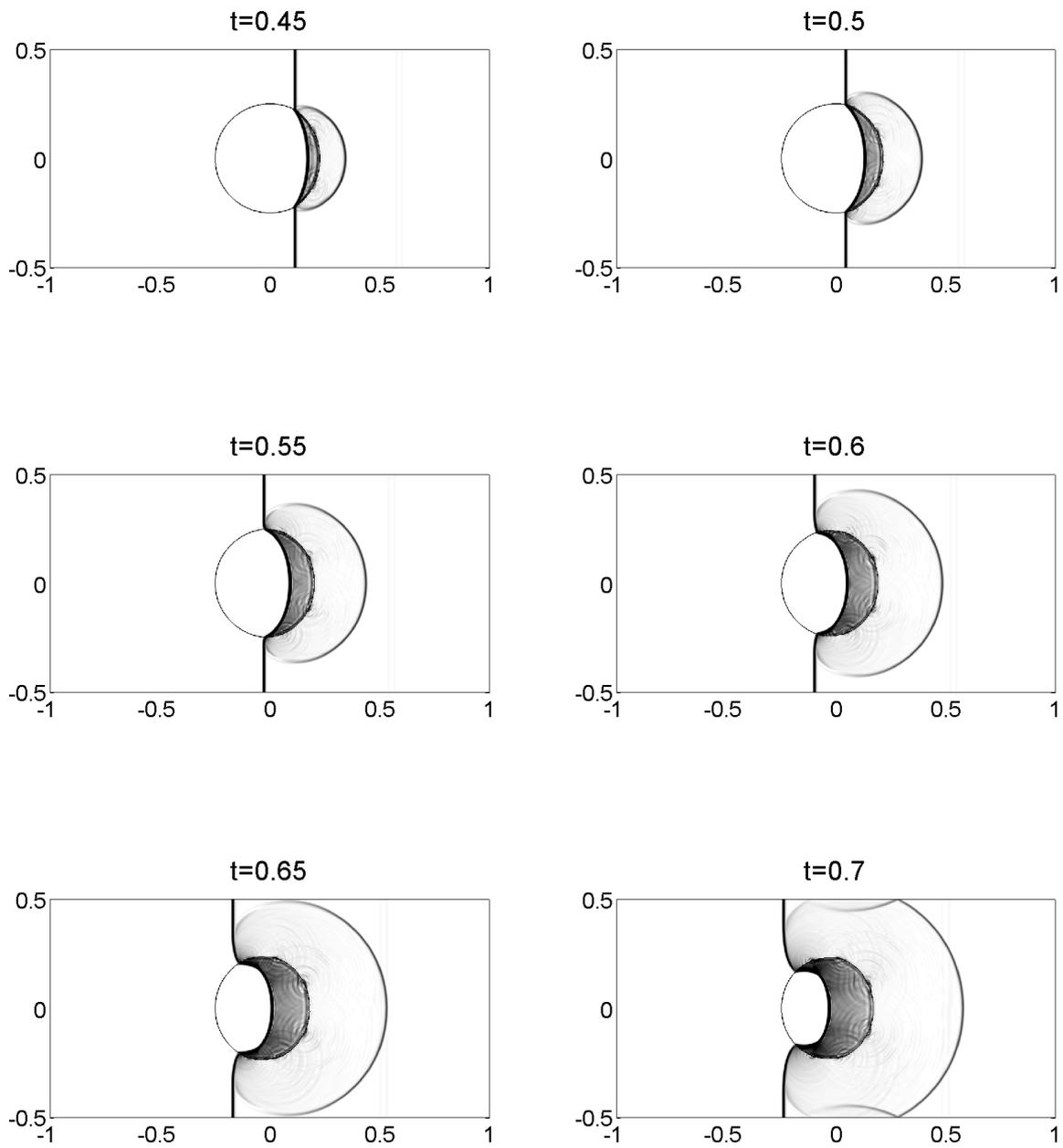
FIGURE 21. Shock-R22 bubble interaction. Earlier times.
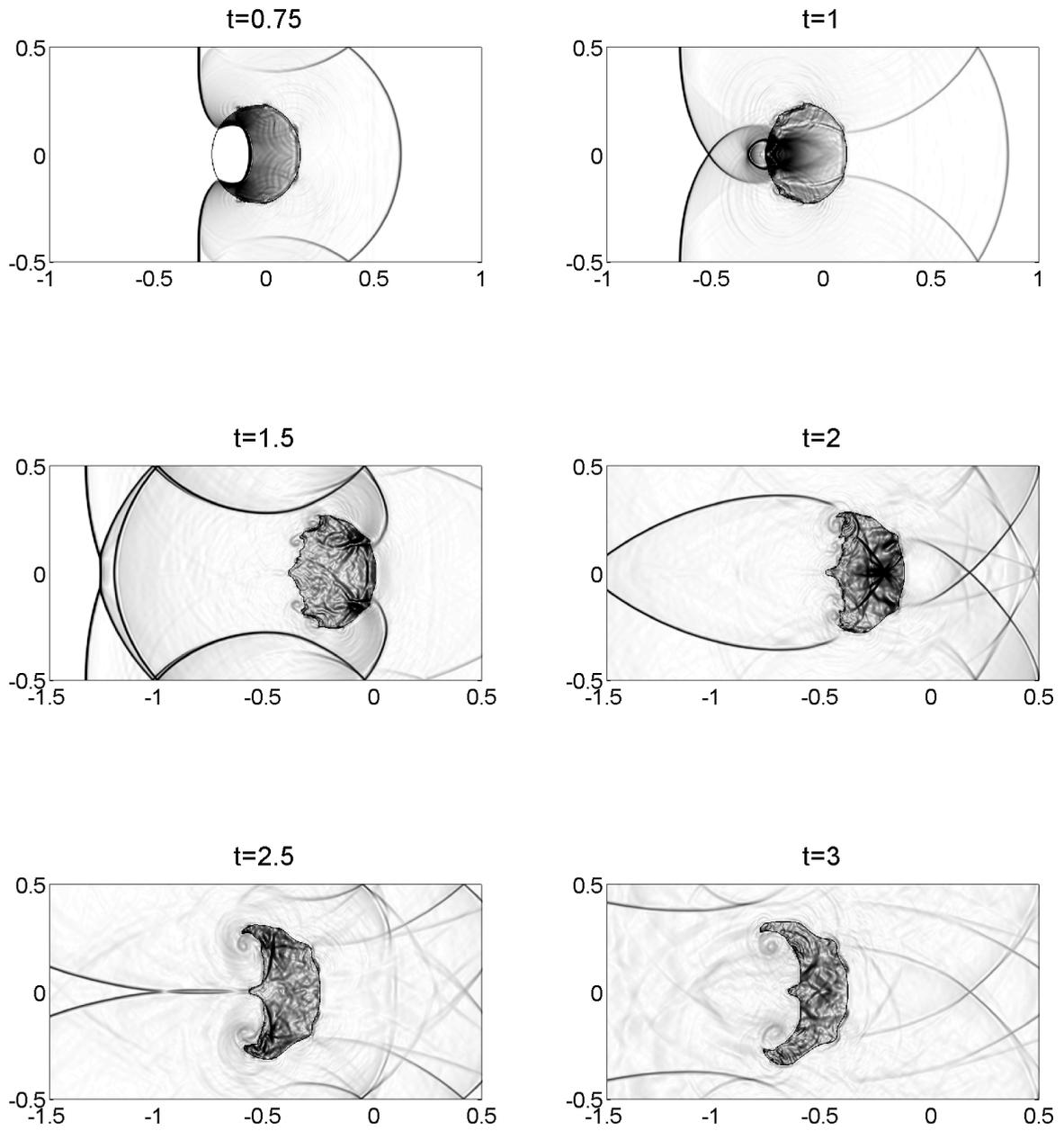
FIGURE 22. Shock-R22 bubble interaction. Later times.

APPENDIX A: SEMI-DISCRETE CENTRAL-UPWIND SCHEMES

Here, we briefly describe the second-order semi-discrete central-upwind schemes.

We begin with the 1-D case. The cell averages (2.2) are evolved in time according to (2.4) with the following central-upwind fluxes (see [28,30]):

$$\mathbf{H}_{j+\frac{1}{2}} := \frac{a_{j+\frac{1}{2}}^+ \mathbf{f}(\mathbf{w}_{j+\frac{1}{2}}^-) - a_{j+\frac{1}{2}}^- \mathbf{f}(\mathbf{w}_{j+\frac{1}{2}}^+)}{a_{j+\frac{1}{2}}^+ - a_{j+\frac{1}{2}}^-} + a_{j+\frac{1}{2}}^+ a_{j+\frac{1}{2}}^- \left[\frac{\mathbf{w}_{j+\frac{1}{2}}^+ - \mathbf{w}_{j+\frac{1}{2}}^-}{a_{j+\frac{1}{2}}^+ - a_{j+\frac{1}{2}}^-} - \mathbf{Q}_{j+\frac{1}{2}} \right], \qquad (A.1)$$

where $\mathbf{w}_{j+\frac{1}{2}}^\pm$ are the point values, reconstructed at $x = x_{j+\frac{1}{2}}$, $a_{j+\frac{1}{2}}^\pm$ are the right- and left-sided local speeds of propagation that, for the Euler equations of gas dynamics, can be estimated by:

$$a_{j+\frac{1}{2}}^+ = \max\left\{ u_{j+\frac{1}{2}}^+ + c_{j+\frac{1}{2}}^+, u_{j+\frac{1}{2}}^- + c_{j+\frac{1}{2}}^-, 0 \right\}, \quad a_{j+\frac{1}{2}}^- = \min\left\{ u_{j+\frac{1}{2}}^+ - c_{j+\frac{1}{2}}^+, u_{j+\frac{1}{2}}^- - c_{j+\frac{1}{2}}^-, 0 \right\},$$

and $\mathbf{Q}_{j+\frac{1}{2}}$ is the "anti-diffusion" terms that help to reduce numerical dissipation present at non-oscillatory central schemes [28]:

$$\mathbf{Q}_{j+\frac{1}{2}} = \text{minmod}\left(\frac{\mathbf{w}_{j+\frac{1}{2}}^+ - \mathbf{w}_{j+\frac{1}{2}}^{\text{int}}}{a_{j+\frac{1}{2}}^+ - a_{j+\frac{1}{2}}^-}, \frac{\mathbf{w}_{j+\frac{1}{2}}^{\text{int}} - \mathbf{w}_{j+\frac{1}{2}}^-}{a_{j+\frac{1}{2}}^+ - a_{j+\frac{1}{2}}^-} \right),$$

$$\mathbf{w}_{j+\frac{1}{2}}^{\text{int}} = \frac{a_{j+\frac{1}{2}}^+ \mathbf{w}_{j+\frac{1}{2}}^+ - a_{j+\frac{1}{2}}^- \mathbf{w}_{j+\frac{1}{2}}^- - \left\{ \mathbf{f}(\mathbf{w}_{j+\frac{1}{2}}^+) - \mathbf{f}(\mathbf{w}_{j+\frac{1}{2}}^-) \right\}}{a_{j+\frac{1}{2}}^+ - a_{j+\frac{1}{2}}^-}.$$

In the 2-D case, the structure of the second-order semi-discrete central-upwind scheme is similar. The cell averages (2.30) are evolved in time according to (2.31) with the numerical fluxes given by (see [28] for their derivation):

$$\mathbf{H}_{j+\frac{1}{2},k}^x := \frac{a_{j+\frac{1}{2},k}^+ \mathbf{f}(\mathbf{w}_{j,k}^{\text{E}}) - a_{j+\frac{1}{2},k}^- \mathbf{f}(\mathbf{w}_{j+1,k}^{\text{W}})}{a_{j+\frac{1}{2},k}^+ - a_{j+\frac{1}{2},k}^-} + a_{j+\frac{1}{2},k}^+ a_{j+\frac{1}{2},k}^- \left[\frac{\mathbf{w}_{j+1,k}^{\text{W}} - \mathbf{w}_{j,k}^{\text{E}}}{a_{j+\frac{1}{2},k}^+ - a_{j+\frac{1}{2},k}^-} - \mathbf{Q}_{j+\frac{1}{2},k}^x \right], \qquad (A.2)$$

and

$$\mathbf{H}_{j,k+\frac{1}{2}}^y := \frac{b_{j,k+\frac{1}{2}}^+ \mathbf{g}(\mathbf{w}_{j,k}^{\text{N}}) - b_{j,k+\frac{1}{2}}^- \mathbf{g}(\mathbf{w}_{j,k+1}^{\text{S}})}{b_{j,k+\frac{1}{2}}^+ - b_{j,k+\frac{1}{2}}^-} + b_{j,k+\frac{1}{2}}^+ b_{j,k+\frac{1}{2}}^- \left[\frac{\mathbf{w}_{j,k+1}^{\text{S}} - \mathbf{w}_{j,k}^{\text{N}}}{b_{j,k+\frac{1}{2}}^+ - b_{j,k+\frac{1}{2}}^-} - \mathbf{Q}_{j,k+\frac{1}{2}}^y \right]. \qquad (A.3)$$

The point values participating in this formula are defined in (2.33), the one-sided local speeds for the Euler equations are:

$$a_{j+\frac{1}{2},k}^+ = \max\left\{ u_{j+1,k}^{\text{W}} + c_{j+1,k}^{\text{W}}, u_{j,k}^{\text{E}} + c_{j,k}^{\text{E}}, 0 \right\}, \quad a_{j+\frac{1}{2},k}^- = \min\left\{ u_{j+1,k}^{\text{W}} - c_{j+1,k}^{\text{W}}, u_{j,k}^{\text{E}} - c_{j,k}^{\text{E}}, 0 \right\},$$

$$b_{j,k+\frac{1}{2}}^+ = \max\left\{ u_{j,k+1}^{\text{S}} + c_{j,k+1}^{\text{S}}, u_{j,k}^{\text{N}} + c_{j,k}^{\text{N}}, 0 \right\}, \quad b_{j,k+\frac{1}{2}}^- = \min\left\{ u_{j,k+1}^{\text{S}} - c_{j,k+1}^{\text{S}}, u_{j,k}^{\text{N}} - c_{j,k}^{\text{N}}, 0 \right\},$$

and the built-in "anti-diffusion" terms are given by [28]:

$$\mathbf{Q}^x_{j+\frac{1}{2},k} = \text{minmod}\left(\frac{\mathbf{w}^{\text{NW}}_{j+1,k} - \mathbf{w}^{\text{int}}_{j+\frac{1}{2},k}}{a^+_{j+\frac{1}{2},k} - a^-_{j+\frac{1}{2},k}}, \frac{\mathbf{w}^{\text{int}}_{j+\frac{1}{2},k} - \mathbf{w}^{\text{NE}}_{j,k}}{a^+_{j+\frac{1}{2},k} - a^-_{j+\frac{1}{2},k}}, \frac{\mathbf{w}^{\text{SW}}_{j+1,k} - \mathbf{w}^{\text{int}}_{j+\frac{1}{2},k}}{a^+_{j+\frac{1}{2},k} - a^-_{j+\frac{1}{2},k}}, \frac{\mathbf{w}^{\text{int}}_{j+\frac{1}{2},k} - \mathbf{w}^{\text{SE}}_{j,k}}{a^+_{j+\frac{1}{2},k} - a^-_{j+\frac{1}{2},k}}\right),$$

$$\mathbf{Q}^y_{j,k+\frac{1}{2}} = \text{minmod}\left(\frac{\mathbf{w}^{\text{SW}}_{j,k+1} - \mathbf{w}^{\text{int}}_{j,k+\frac{1}{2}}}{b^+_{j,k+\frac{1}{2}} - b^-_{j,k+\frac{1}{2}}}, \frac{\mathbf{w}^{\text{int}}_{j,k+\frac{1}{2}} - \mathbf{w}^{\text{NW}}_{j,k}}{b^+_{j,k+\frac{1}{2}} - b^-_{j,k+\frac{1}{2}}}, \frac{\mathbf{w}^{\text{SE}}_{j,k+1} - \mathbf{w}^{\text{int}}_{j,k+\frac{1}{2}}}{b^+_{j,k+\frac{1}{2}} - b^-_{j,k+\frac{1}{2}}}, \frac{\mathbf{w}^{\text{int}}_{j,k+\frac{1}{2}} - \mathbf{w}^{\text{NE}}_{j,k}}{b^+_{j,k+\frac{1}{2}} - b^-_{j,k+\frac{1}{2}}}\right),$$

where

$$\mathbf{w}^{\text{int}}_{j+\frac{1}{2},k} = \frac{a^+_{j+\frac{1}{2},k}\mathbf{w}^{\text{W}}_{j+1,k} - a^-_{j+\frac{1}{2},k}\mathbf{w}^{\text{E}}_{j,k} - \left\{\mathbf{f}(\mathbf{w}^{\text{W}}_{j+1,k}) - \mathbf{f}(\mathbf{w}^{\text{E}}_{j,k})\right\}}{a^+_{j+\frac{1}{2},k} - a^-_{j+\frac{1}{2},k}},$$

$$\mathbf{w}^{\text{int}}_{j,k+\frac{1}{2}} = \frac{b^+_{j,k+\frac{1}{2}}\mathbf{w}^{\text{S}}_{j,k+1} - b^-_{j,k+\frac{1}{2}}\mathbf{w}^{\text{N}}_{j,k} - \left\{\mathbf{g}(\mathbf{w}^{\text{S}}_{j,k+1}) - \mathbf{g}(\mathbf{w}^{\text{N}}_{j,k})\right\}}{b^+_{j,k+\frac{1}{2}} - b^-_{j,k+\frac{1}{2}}}.$$

Remark. In the original semi-discrete central-upwind scheme [30], both $\mathbf{Q}_{j+\frac{1}{2}}$ in (A.1) and $\mathbf{Q}^x_{j+\frac{1}{2},k}$ and $\mathbf{Q}^y_{j,k+\frac{1}{2}}$ in (A.2)–(A.3) were chosen to be zero.

REFERENCES

[1] R. Abgrall, Generalization of the Roe scheme for the computation of mixture of perfect gases. *Rech. Aérosp.* **6** (1988) 31–43.

[2] R. Abgrall, How to prevent pressure oscillations in multicomponent flows: A quasi conservative approach. *J. Comp. Phys.* **125** (1996) 150–160.

[3] R. Abgrall and S. Karni, Ghost-fluids for the poor: a single fluid algorithm for multifluids, in *Hyperbolic problems: theory, numerics, applications*, Vols. I, II (Magdeburg, 2000), Birkhäuser, Basel, *Internat. Ser. Numer. Math.* **140** (2001) 1–10.

[4] R. Abgrall and S. Karni, Computations of compressible multifluids. *J. Comp. Phys.* **169** (2001) 594–623.

[5] R. Abgrall and R. Saurel, Discrete equations for physical and numerical compressible multiphase flow mixtures. *J. Comp. Phys.* **186** (2003) 361–396.

[6] R. Abgrall, B. N'Konga and R. Saurel, Efficient numerical approximation of compressible multi-material flow for unstructured meshes. *Comput. Fluids* **4** (2003) 571–605.

[7] I.-L. Chern, J. Glimm, O. McBryan, B. Plohr and S. Yaniv, Front tracking for gas dynamics. *J. Comp. Phys.* **62** (1986) 83–110.

[8] A. Chertock and A. Kurganov, Conservative locally moving mesh method for multifluid flows. *Proceedings of the Fourth International Symposium on Finite Volumes for Complex Applications*, Marrakech (2005) 273–284.

[9] F. Coquel, K. El Amine, E. Godlewski, B. Perthame and P. Rascle, A numerical method using upwind schemes for the resolution of two-phase flows. *J. Comp. Phys.* **136** (1997) 272–288.

[10] S.F. Davis, An interface tracking method for hyperbolic systems of conservation laws. *Appl. Numer. Math.* **10** (1992) 447–472.

[11] R.P. Fedkiw, T. Aslam, B. Merriman and S. Osher, A non-oscillatory Eulerian approach to interfaces in multimaterial flows (the ghost fluid method). *J. Comp. Phys.* **152** (1999) 457–492.

[12] J. Glimm, J.W. Grove, X.L. Li, K.-M. Shyue, Y. Zeng and Q. Zhang, Three-dimensional front tracking. *SIAM J. Sci. Comput.* **19** (1998) 703–727.

[13] J. Glimm, X.L. Li, Y. Liu and N. Zhao, Conservative front tracking and level set algorithms. *Proc. Natl. Acad. Sci. USA* **98** (2001) 14198–14201.

[14] J. Glimm, Y. Liu, Z. Xu and N. Zhao, Conservative front tracking with improved accuracy. *SIAM J. Numer. Anal.* **41** (2003) 1926–1947.

[15] E. Godlewski and P.-A. Raviart, *Numerical approximation of hyperbolic systems of conservation laws.* Springer-Verlag, New York (1996).

[16] E. Godlewski and P.-A. Raviart, The numerical interface coupling of nonlinear hyperbolic systems of conservation laws. I. The scalar case. *Numer. Math.* **97** (2004) 81–130.

[17] E. Godlewski, K.-C. Le Thanh, P.-A. Raviart, The numerical interface coupling of nonlinear hyperbolic systems of conservation laws. II. The case of systems. *ESAIM: M2AN* **39** (2005) 649–692.

[18] S. Gottlieb, C.-W. Shu and E. Tadmor, High order time discretization methods with the strong stability property. *SIAM Rev.* **43** (2001) 89–112.

[19] J.-F. Haas and B. Sturtevant, Interaction of weak shock waves with cylindrical and spherical gas inhomogeneities. *J. Fluid Mech.* **181** (1987) 313–336.

[20] A. Harten and J.M. Hyman, Self-adjusting grid methods for one-dimensional hyperbolic conservation laws. *J. Comp. Phys.* **50** (1983) 235–269.

[21] A. Harten and S. Osher, Uniformly high-order accurate nonoscillatory schemes, I. *SIAM J. Numer. Anal.* **24** (1987) 279–309.

[22] A. Harten, S. Osher, B. Engquist and S.R. Chakravarthy, Some results on uniformly high order accurate essentially non-oscillatory schemes. *Appl. Numer. Math.* **2** (1986) 347–377.

[23] P. Jenny, B. Mueller and H. Thomann, Correction of conservative Euler solvers for gas mixtures. *J. Comp. Phys.* **132** (1997) 91–107.

[24] S. Karni, Multicomponent flow calculations by a consistent primitive algorithm. *J. Comp. Phys.* **112** (1994) 31–43.

[25] S. Karni, Compressible bubbles with surface tension, in *Sixteenth International Conference on Numerical Methods in Fluid Dynamics* (Arcachon, 1998), Springer, Berlin, *Lecture Notes in Physics* **515** (1998) 506–511.

[26] S. Karni, E. Kirr, A. Kurganov and G. Petrova, Compressible two-phase flows by central and upwind schemes. *ESAIM: M2AN* **38** (2004) 477–493.

[27] D. Kröner, *Numerical Schemes for Conservation Laws*. Wiley, Chichester (1997).

[28] A. Kurganov and C.-T. Lin, On the reduction of numerical dissipation in central-upwind schemes. *Commun. Comput. Phys.* **2** (2007) 141–163.

[29] A. Kurganov and E. Tadmor, New high-resolution central schemes for nonlinear conservation laws and convection-diffusion equations. *J. Comp. Phys.* **160** (2000) 241–282.

[30] A. Kurganov, S. Noelle and G. Petrova, Semi-discrete central-upwind schemes for hyperbolic conservation laws and Hamilton-Jacobi equations. *SIAM J. Sci. Comput.* **21** (2001) 707–740.

[31] B. Larrouturou, How to preserve the mass fractions positivity when computing compressible multi-component flows. *J. Comp. Phys.* **95** (1991) 59–84.

[32] R. LeVeque, *Finite volume methods for hyperbolic problems, Cambridge Texts in Applied Mathematics*. Cambridge University Press (2002).

[33] K.-A. Lie and S. Noelle, On the artificial compression method for second-order nonoscillatory central difference schemes for systems of conservation laws. *SIAM J. Sci. Comput.* **24** (2003) 1157–1174.

[34] W. Mulder, S. Osher and J.A. Sethian, Computing interface motion in compressible gas dynamics. *J. Comp. Phys.* **100** (1992) 209–228.

[35] H. Nessyahu and E. Tadmor, Non-oscillatory central differencing for hyperbolic conservation laws. *J. Comp. Phys.* **87** (1990) 408–463.

[36] J.J. Quirk and S. Karni, On the dynamics of a shock-bubble interaction. *J. Fluid Mech.* **318** (1996) 129–163.

[37] P.L. Roe, Fluctuations and signals – a framework for numerical evolution problems, in *Numerical Methods for Fluid Dynamics*, Academic Press, New York (1982) 219–257.

[38] R. Saurel and R. Abgrall, A multiphase Godunov method for compressible multifluid and multiphase flows. *J. Comp. Phys.* **150** (1999) 425–467.

[39] K.-M. Shyue, An efficient shock-capturing algorithm for compressible multicomponent problems. *J. Comp. Phys.* **142** (1998) 208–242.

[40] K.-M. Shyue, A fluid-mixture type algorithm for compressible multicomponent flow with van der Waals equation of state. *J. Comp. Phys.* **156** (1999) 43–88.

[41] P.K. Sweby, High resolution schemes using flux limiters for hyperbolic conservation laws. *SIAM J. Numer. Anal.* **21** (1984) 995–1011.

[42] V. Ton, Improved shock-capturing methods for multicomponent and reacting flows. *J. Comp. Phys.* **128** (1996) 237–253.

[43] E.F. Toro, *Riemann solvers and numerical methods for fluid dynamics. A practical introduction*. Second edition, Springer-Verlag, Berlin (1999).

[44] G. Tryggvason, B. Bunner, A. Esmaeeli, D. Juric, N. Al-Rawahi, W. Tauber, J. Han, S. Nas and Y.-J. Jan, A front-tracking method for the computations of multiphase flow. *J. Comp. Phys.* **169** (2001) 708–759.

[45] B. van Leer, Towards the ultimate conservative difference scheme, V. A second order sequel to Godunov's method. *J. Comp. Phys.* **32** (1979) 101–136.

[46] J. Wackers and B. Koren, *Five-equation model for compressible two-fluid flow*. Report MAS-E0414, CWI, Amsterdam (2004). Available at http://ftp.cwi.nl/CWIreports/MAS/MAS-E0414.pdf

[47] S.-P. Wang, M.H. Anderson, J.G. Oakley, M.L. Corradini and R. Bonazza, A thermodynamically consistent and fully conservative treatment of contact discontinuities for compressible multicomponent flows. *J. Comp. Phys.* **195** (2004) 528–559.

[48] A. Wardlaw, *Underwater explosion test cases*. IHTR 2069 (1998).