SIAM J. SCI. COMPUT.
Vol. 31, No. 6, pp. 4795–4813

A SECOND-ORDER ACCURATE CONSERVATIVE FRONT-TRACKING METHOD IN ONE DIMENSION*

CAROLINE GATTI-BONO†, PHILLIP COLELLA‡, AND DAVID TREBOTICH†

Abstract. This paper presents a conservative front-tracking method for shocks and contact discontinuities that is second-order accurate. It is based on a volume-of-fluid method that treats the moving front with concepts similar to those of an embedded-boundary method. Special care is taken in the centering of the data to ensure the right order of accuracy at the front, and a redistribution step guarantees conservation. A suite of test problems, for tracking both shocks and contact discontinuities, is presented that confirms that the method is second-order accurate.

Key words. front-tracking, finite-volume, Godunov method, multifluids

AMS subject classifications. 35-XX, 65-XX, 76-XX

DOI. 10.1137/070704083

1. Introduction.

There are two approaches to computing time-dependent discontinuities as they arise in hyperbolic systems of conservation laws in multiple space dimensions. In shock capturing methods, the discontinuity is represented on the grid as a continuous transition across several grid cells, with the jump relations being satisfied because the discretization is in discrete conservation form [13]. The second method is front tracking, in which the discontinuity is represented as a moving free boundary, at which the jump relations are applied as boundary conditions. Both approaches have their advantages and disadvantages. Capturing relies on the cancellation of $O(1)$ errors in the fluxes, which can fail if, for example, the grid is not smoothly varying [20]. Also, there are fronts such as the interface between two materials with radically different equations of state for which defining the intermediate states obtained in the continuous transition is problematic. Front tracking has been primarily used if the discontinuity can be identified in the initial data and the surface of discontinuity separates the spatial domain into distinct disjoint regions. Because of these limitations, hybrid approaches have been developed, in which one or more discontinuities are identified in the initial data as tracked fronts, with the remaining ones represented by capturing, with conservative differencing used to discretize the conservation laws away from the front (for a review of the early work in this area, see [12]). Lagrangian and arbitrary Lagrangian Eulerian (ALE) methods are a version of this approach in that the tracked front (usually a material interface) is aligned with the faces of a moving grid. The disadvantage to the Lagrangian/ALE approach is that local distortions in the front lead to distortions in the grid, leading to a loss of smoothness in the grid and thereby a loss of accuracy in the capturing method.

The starting point for our work is the conservative hybrid tracking and capturing

*Received by the editors September 28, 2007; accepted for publication (in revised form) November 2, 2009; published electronically February 3, 2010. This manuscript was authored by a contractor of the U.S. Government under contract DE-AC02-05CH11231. This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under contract DE-AC52-07NA27344. The U.S. Government retains a nonexclusive, royalty-free license to publish or reproduce the published form of this contribution, or allow others to do so, for U.S. Government purposes.
http://www.siam.org/journals/sisc/31-6/70408.html
†Lawrence Livermore National Laboratory, Livermore, CA 94551 (CSBono@lbl.gov, DPTrebotich@lbl.gov).
‡Lawrence Berkeley National Laboratory, Berkeley, CA 94720 (pcolella@lbl.gov).

method in [4, 3, 14]. In this approach, the tracked front cuts through a Cartesian grid, and the intersection of the Cartesian cells with the regions on either side of the front forms a collection of time-dependent control volumes, over which the conservation laws can be integrated in space-time to derive a finite-volume discretization. Such an approach leads to a CFL stability issue due to the presence of arbitrarily small control volumes, which was addressed in this work by computing the difference between a conservative update and a nonconservative, but CFL stable, update, and redistributing that difference to maintain overall stability, accuracy, and conservation. Another approach is cell-merging. It was used by Glimm, Li, and Zhao [10], where a second-order accurate algorithm on the one-dimensional Burgers's equation is presented, and in Glimm et al. [9], where the method is extended to multiple dimensions. In cell-merging, the small control volumes are merged with adjacent cells to avoid singularities, but, while finding a full cell for each small cell is a well-behaved problem in one dimension, it is not always straightforward to do in multiple dimensions. A major advantage of redistribution is that it is a simple method for simultaneously maintaining numerical stability and conservation in multiple dimensions [3, 14].

The primary limitation of the methods as originally formulated [4, 3, 14] is the loss of accuracy at the irregular control volumes. Classical truncation error estimates for finite-volume methods rely on a cancellation of errors in the flux differences between successive cell faces in each coordinate direction, which in turn depends on smooth variation of the geometric properties of the grid such as face areas and face centroids. At the irregular control volumes formed at the intersection of the front with the grid, these cancellations no longer occur, and there is a loss of at least one order of accuracy relative to that obtained away from the front. In addition, the methods in [4, 3, 14] use quadrature methods for approximating the face averages in space-time that are only first-order accurate at the irregular control volumes, leading to $O(1)$ truncation errors. Nonetheless, these methods have proved surprisingly effective, particularly in the case of shock waves. The latter are noncharacteristic boundaries for gas dynamics, so that an $O(1)$ truncation error on a set of codimension one leads to first-order convergence in max norm for the solution error [6].

Our goal is to extend the ideas developed for embedded-boundary representations of conservation laws in [8] to obtain a method that is second-order accurate away from the irregular control volumes and formally consistent at the tracked front, with a truncation error that vanishes linearly with the mesh spacing. This is sufficient to obtain a second-order convergence in the solution error in L^1; for the case of shocks, the solution error is second-order in L^∞. As in the embedded-boundary case, our method is a generalization of a second-order accurate Godunov method and uses redistribution to deal with the small-cell CFL stability problem. However, the programming machinery required to generalize the approach in [8] to the case of moving tracked fronts is substantial. In the present work, our goal is to identify the fundamental ideas for performing that extension for the case of one-dimensional problems, prior to undertaking the more general case. One of the new issues that arises in the tracked front setting is the necessity to distinguish correctly between the averages of the conserved quantities over control volumes and the value at the center of the Cartesian cell, which differ by terms of first order in the mesh spacing at irregular control volumes. For nonmoving boundaries this is not necessary due to a cancellation of errors in the time discretization.

2. Algorithm description.

2.1. Equations of gas dynamics.

We consider a problem where the domain is filled by two fluids separated by a front. On either side of the front, the fluid is

assumed to be compressible and inviscid, and the equations in conservation form are

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}}{\partial x} = \mathbf{0}, \tag{1}$$

where

$$\mathbf{U} = (\rho,\, \rho u,\, \rho E)\,, \tag{2}$$

$$\mathbf{F} = \left(\rho u,\, \rho u^2 + p,\, \rho u E + up\right). \tag{3}$$

Here, ρ is the fluid density, u is the velocity, p is the pressure, and

$$\rho E = \frac{p}{\gamma - 1} + \rho \frac{u^2}{2}. \tag{4}$$

The speed of the front is given by

$$\frac{dx^f}{dt} = s^f, \tag{5}$$

where x^f is the position of the front. At the front, the variables satisfy the Rankine–Hugoniot conditions

$$\left[\left(\mathbf{F} - \mathbf{s^f U}\right)\cdot \mathbf{n}\right] = \left(\left(\mathbf{F} - \mathbf{s^f U}\right)\cdot \mathbf{n}\right)_L - \left(\left(\mathbf{F} - \mathbf{s^f U}\right)\cdot \mathbf{n}\right)_R = 0, \tag{6}$$

where L (respectively, R) represents the state of the left (respectively, right) fluid.

We introduce the vector $\mathbf{W}$:

$$\mathbf{W} = (\rho,\, u,\, p)\,. \tag{7}$$

The governing equations can be rewritten as

$$\frac{\partial \mathbf{W}}{\partial t} + \mathbf{A}\frac{\partial \mathbf{W}}{\partial x} = \mathbf{0} \tag{8}$$

with

$$\mathbf{A} = \begin{pmatrix} u & \rho & 0 \\ 0 & u & \frac{1}{\rho} \\ 0 & \rho c^2 & u \end{pmatrix}. \tag{9}$$

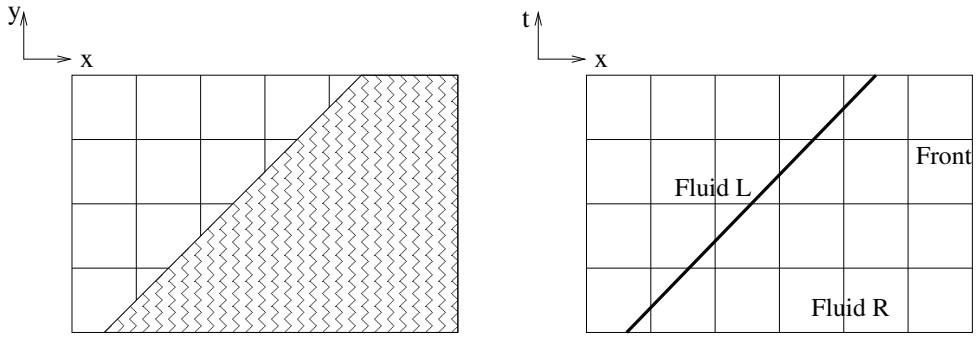
2.2. Finite-volume discretization for a moving boundary. The method is based on ideas akin to those of an embedded-boundary method, but we work on a space-time grid instead of a spatial grid. In an embedded-boundary approach, we superimpose a Cartesian grid onto the computational domain, and this can generate cells that are not rectangular because of the presence of geometry. The primary variables are the conservation variables $\mathbf{U}$ centered at the space centroids and, at time t^n, are denoted $\bar{\mathbf{U}}^n$. The fluxes are centered spatially at the faces and at the time centroids, and are denoted by $\bar{\mathbf{F}}_{i+\frac{1}{2}}$. The variables $\mathbf{W}$ are used at half time step and are centered spatially on the faces.

We use a volume-of-fluid method to discretize the Euler equations, using the divergence theorem to evaluate the second term in (1):

$$\operatorname{div}(\mathbf{F}) \approx \frac{1}{V}\int_{\Omega\cap\Upsilon_i} \operatorname{div}(\mathbf{F})\, d\mathbf{x} = \frac{1}{V}\oint_{\partial(\Omega\cap\Upsilon_i)} \mathbf{F}\cdot\mathbf{n} dS \tag{10}$$

$$= \frac{1}{\kappa_i h}\left(\sum_{\pm=+,-}\sum_{s=1}^{d} \pm\alpha_{i\pm\frac{1}{2}e_s}\bar{F}^s\left(x_{i\pm\frac{1}{2}e_s}\right) + \alpha_i^B \mathbf{F}(x_i^B)\cdot\mathbf{n}_i^B\right), \tag{11}$$

where Υ_i is the Cartesian cell at $\mathbf{i}$, Ω is the computational domain, $V = \int_{\Omega\cap\Upsilon_\mathbf{i}} d\mathbf{x}$ is the cell volume, $\mathbf{n}$ is the outward normal, $\bar{\mathbf{F}}$ represents the quantity $\mathbf{F}$ taken at the face centroid, κ is the volume fraction of the cell (i.e., the ratio of the current cell and the volume of a regular (uncut) Cartesian cell), α is the area fraction of the face (i.e., the ratio of the length of the current face and the length of a regular face), and the superscript ^{B} denotes quantities taken on the embedded boundary.



(a) Embedded–Boundary (b) Multifluid

FIG. 1. *Comparison between embedded-boundary and multifluid approaches.*

In an embedded-boundary tracking approach, each fluid cuts out a domain on a Cartesian grid in space and time, and a front represents the interface between them, as shown in Figure 1. The divergence theorem is applied in space and time:

$$\int_{[t^n;t^{n+1}]\times(\Omega^a(t)\cap\Upsilon_i)} \left(\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}}{\partial x}\right) dxdt = 0 \tag{12}$$

with $a \in \{L, R\}$.

Here, we use the same ideas as in (11), and we use the formula

$$\frac{\partial}{\partial t}\int_{\Omega(t)} U(\mathbf{x},t)d\mathbf{x} = \int_{\Omega(t)} \frac{\partial U}{\partial t} d\mathbf{x} + \oint_{\partial\Omega(t)} U s^f dS \tag{13}$$

for integration and differentiation with a variable volume.

Therefore, (12) becomes

$$\left[\int_{(\Omega^a(t)\cap\Upsilon_i)} \mathbf{U}dx\right]_{t^n}^{t^{n+1}} + \int_{\partial([t^n;t^{n+1}]\times(\Omega^a(t)\cap\Upsilon_i))\setminus\mathcal{F}(t)} \left(\frac{\partial \mathbf{F}}{\partial x}\right) dxdt + \int_{\mathcal{F}(t)} \left(\mathbf{F} - s^f\mathbf{U}\right)\cdot\mathbf{n}dS = 0, \tag{14}$$

where $\mathcal{F}(t)$ represents the front, and, discretizing in time and in space, we obtain

$$\begin{aligned}\kappa_{a,i}^{n+1}\left(\bar{\mathbf{U}}_{a,i}^{n+1} - \bar{\mathbf{U}}_{a,i}^{n,\,n+1}\right) &= \kappa_{a,i}^{n}\bar{\mathbf{U}}_{a,i}^{n} - \kappa_{a,i}^{n+1}\bar{\mathbf{U}}_{a,i}^{n,\,n+1} \\ &\quad - \Delta t\left(\frac{\alpha_{a,\,i+\frac{1}{2}}\bar{\mathbf{F}}_{a,\,i+\frac{1}{2}} - \alpha_{a,\,i-\frac{1}{2}}\bar{\mathbf{F}}_{a,\,i-\frac{1}{2}}}{\Delta x} \pm \frac{1}{\Delta x}\alpha^f\bar{\mathbf{F}}^f\right) \\ &\equiv -\kappa_{a,i}^{n+1}\Delta t\left(\overline{DF}\right)_i^C = -\kappa_{a,i}^{n+1}\Delta t\left.\frac{\partial F}{\partial x}\right|_{x_c^{n+1}}\Delta x + O(h\Delta t),\end{aligned} \tag{15}$$

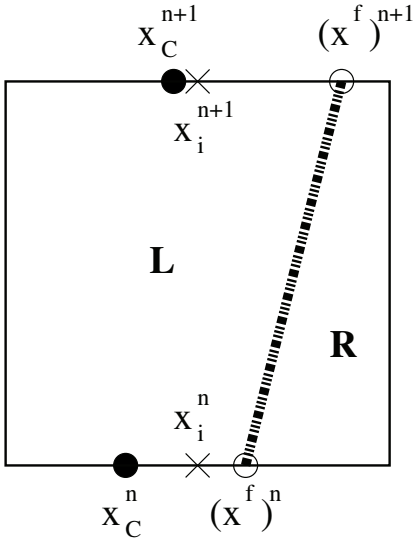


FIG. 2. *Location of the centroids, face centers, and front at times t^n and t^{n+1}.*

where we define $\bar{\mathbf{U}}^{n,\,n+1}$ as the quantity $\mathbf{U}$ at time t^n taken at the centroid for time t^{n+1}:

$$\bar{\mathbf{U}}^{n,\,n+1} = \mathbf{U}(x_c^{n+1}, t^n), \tag{16}$$

where x_c^{n+1} is the location of the centroid at time $n+1$ as shown in Figure 2, α^f is the area fraction of the front, and $\mathbf{F}^f$ is

$$\bar{\mathbf{F}}^f = \bar{\mathbf{F}} - s^f \bar{\mathbf{U}}, \tag{17}$$

which is single-valued at the front by (6). By construction, (15) satisfies the discrete conservation property

$$\sum \kappa_a^{n+1} \bar{\mathbf{U}}^{n+1} = \sum \kappa_a^n \bar{\mathbf{U}}^n + \text{boundary terms}. \tag{18}$$

The update equation (15) can be rewritten as

$$\bar{\mathbf{U}}_a^{n+1} = \bar{\mathbf{U}}_a^{n,\,n+1} - \Delta t \left(\overline{DF}\right)^C. \tag{19}$$

Because κ^{n+1} has the potential of being arbitrarily small, we use a weighted average for the update:

$$\bar{\mathbf{U}}_{a,i}^{n+1} = \bar{\mathbf{U}}_{a,i}^{n,\,n+1} - \Delta t \kappa_{a,i}^{n+1} \overline{DF}_{a,i}^{C} - \Delta t (1 - \kappa_{a,i}^{n+1}) \overline{DF}_{a,i}^{NC}, \tag{20}$$

where $\overline{DF}^{NC}$ is a stable nonconservative estimate of $\overline{DF}^{C}$. To ensure conservation, the mass difference between the conservative and the nonconservative update,

$$\delta M_{a,i} = \kappa_{a,i}^{n+1} \left[\overline{DF}^C - \left(\kappa_{a,i}^{n+1} \overline{DF}^C - (1 - \kappa_{a,i}^{n+1}) \overline{DF}^{NC} \right) \right] = O(h \Delta t), \tag{21}$$

is redistributed to (large) adjacent cells. The details of the redistribution algorithm are given in section 2.4.3. In one dimension, redistribution could have easily been avoided by using cell-merging (see, e.g., [10]). However, cell-merging in three dimensions leads to difficult geometric constructions that have not been fully worked out, and it is not clear how to combine them with discretization methods that maintain $O(h)$ truncation error near the embedded boundary.

2.3. Overview of the algorithm. This algorithm aims to compute $\bar{U}_a^{n+1}$ and $(x^f)^{n+1} = x^f((n+1)\Delta t)$ given $\bar{U}_a^n$ and $(x^f)^n = x^f(n\Delta t)$, where $(x^f)^n$ and $(x^f)^{n+1}$ are pictured in Figure 2. The quantity $\bar{U}_{a,i}^n$ is defined as

$$\bar{U}_{a,i}^n \approx \frac{1}{\kappa_{a,i}^n \Delta x} \int_{\Delta_{a,i}^n} U^n(x, n\Delta t)dx \qquad \forall \kappa_{a,i}^n > 0, \tag{22}$$

where

$$\kappa_{a,i}^n = \frac{1}{\Delta x} |\Upsilon_i \cap \Omega_a|. \tag{23}$$

Following [8], we will compute $O(h^2)$ approximations to the fluxes required to compute a consistent approximation to $\Delta t \frac{\partial F}{\partial x}$ given in (15), given $U_{a,i}^n \approx U_a(ih, n\Delta t)$, the values of the solution at the centers of Cartesian cells. For the cells that are not intersected by the tracked front, these are the same to $O(h^2)$ as the cell averages $\bar{U}_a^n$. However, the cell averages at irregular control volumes differ from the values at the Cartesian cell centers by $O(h)$.

- We compute $O(h^2)$ approximations of $U_{a,i}^n$ at such cells by linear extrapolation:

$$U_{a,i^f}^n = \bar{U}_{a,i^f}^n + \frac{(1-\kappa_{a,i^f}^n)}{(1+\kappa_{a,i^f}^n)}(\bar{U}_{a,i^f}^n - U_{a,i^f \pm 1}^n). \tag{24}$$

  In the case of the embedded-boundary representations of a nonmoving boundary, the primary dependent variables are the $U_{a,i}^n$, and the $O(h)$ errors associated with approximating the cell averages by $\bar{U}_{a,i}^n$ cancel to leading order in the expression for the truncation error, so this problem does not arise.
- We compute an initial estimate of the front speed by solving the Riemann problem

$$s^{f,n} = \mathcal{R}\left(\mathbf{W}_L^{f,n}, \mathbf{W}_R^{f,n}\right), \tag{25}$$

  where $\mathbf{W}_a^{f,n}$ is obtained by second-order extrapolation from $\mathbf{W}_a^n(x_i, t^n)$:

$$\mathbf{W}_a^{f,n} = \frac{1}{2}\left(\mathbf{W}_{a,i}^n + \mathbf{W}_{a,i\pm 1}^n\right) + \kappa_a^n \left(\mathbf{W}_{a,i}^n - \mathbf{W}_{a,i\pm 1}^n\right). \tag{26}$$

  From this, we obtain

$$x^{f,n+1} = x^{f,n} + \Delta t s^{f,n}. \tag{27}$$

  The geometry is advanced for the entire time step using this first estimate of the front speed.
- We compute the values of $\mathbf{U}$ for a newly uncovered cell if necessary, as well as the eigenvectors and eigenvalues:

$$\mathbf{U}_{i^{f^{n+1}}}^n = \mathbf{U}_{i^{f^n}}^n \pm \Delta \mathbf{U}, \tag{28}$$

  where

$$\Delta \mathbf{U} = \begin{cases} \min(2|\Delta U_{i^{f^n}-\frac{1}{2}}|, 2|\Delta U_{i^{f^n}-\frac{3}{2}}|, 2|\Delta U_{i^{f^n}-\frac{5}{2}}|, |\Delta U_{i^{f^n}-2}|) \\ \qquad \text{if } \Delta U_{i^{f^n}-\frac{1}{2}} \Delta U_{i^{f^n}-\frac{3}{2}} > 0,\ \Delta U_{i^{f^n}-\frac{3}{2}} \Delta U_{i^{f^n}-\frac{5}{2}} > 0, \\ 0 \qquad \text{otherwise} \end{cases} \tag{29}$$

with

$$\Delta U_{i-\frac{1}{2}} = U_i - U_{i-1}, \Delta U_{i-2} = \frac{U_{i-1} - U_{i-3}}{2}.$$

- We advance the variables $\mathbf{W}$ to half time step:

$$\mathbf{W}^{n+\frac{1}{2}}_{a,\, i+\frac{1}{2}} = \mathbf{W}_a \left(x_{i+\frac{1}{2}}, t^n + \frac{\Delta t}{2} \right). \tag{30}$$

Then we compute the fluxes $\mathbf{F}$ for the same centering:

$$\mathbf{F}^{n+\frac{1}{2}}_{a,\, i+\frac{1}{2}} = \mathbf{F} \left(\mathbf{W}^{n+\frac{1}{2}}_{a,i+\frac{1}{2}} \right). \tag{31}$$

- We advance the primary variables $\mathbf{U}_{a,i}$ using the fluxes at half time step:

$$\tilde{\mathbf{U}}^{n+1}_{a,i} = \mathbf{U}^n_{a,i} - \Delta t \, (DF)^{NC}_{a,i}, \tag{32}$$

$$(DF)^{NC}_{a,i} = \frac{\mathbf{F}^{n+\frac{1}{2}}_{a,i+\frac{1}{2}} - \mathbf{F}^{n+\frac{1}{2}}_{a,i-\frac{1}{2}}}{\Delta x}. \tag{33}$$

- We use this preliminary update to get a new estimate of the front velocity $s^{f,n}$ by solving the Riemann problem

$$s^{f,\, n+1} = \mathcal{R} \left(\tilde{\mathbf{W}}^{f,n+1}_L, \tilde{\mathbf{W}}^{f,n+1}_R \right), \tag{34}$$

where $\tilde{\mathbf{W}}^{f,n+1}_a$ is obtained using (26) and the quantities $\tilde{\mathbf{W}}^{f,n+1}_{a,i} = W(\tilde{\mathbf{U}^{n+1}_{a,i}})$. The second-order estimate of the front speed is given by

$$s^f = \frac{s^{f,\, n} + s^{f,\, n+1}}{2}. \tag{35}$$

The geometry is then reconstructed with the position of the front x^f given by

$$x^{f,\, n+1} = x^{f,\, n} + s^f \Delta t. \tag{36}$$

The geometry and eigenproperties are recomputed. If the front crosses into a new cell as a result of this recompute, we go through the steps above to have the right data in the newly uncovered cell.
- The solution $\bar{\mathbf{U}}$ is advanced using (20).
- We redistribute the excess mass according to (21).

2.4. Details of the algorithm.

2.4.1. Predictor step. In this section, we evaluate the variables that are needed to compute the final update of $\mathbf{U}$. This means that we will need to evaluate fluxes at time centers and time centroids. We take a standard Godunov approach where we advance the vector $\mathbf{W}$ from the left ($\mathbf{W}^+$) and the right side ($\mathbf{W}^-$) of the face we are looking at and obtain the final value on this face by picking the upwind state with a Riemann problem:

$$\mathbf{W}^{\pm}_a \left(x_i \pm \frac{\Delta x}{2}; t^n + \frac{\delta t}{2} \right) = \mathbf{W}^n_{a,\, i} + \frac{1}{2} \sum_k \alpha^k \left(\pm 1 - \frac{\delta t}{\Delta x} \max(\pm \lambda^k, 0) \right) \mathbf{r}^k, \tag{37}$$

where $a \in \{L, R\}$, $\delta t = \Delta t$ or $\delta t = 0$, $\mathbf{r}^k$ are right eigenvectors of the matrix $\mathbf{A}$, and λ^k are the eigenvalues. The coefficient α^k (not to be confused with the area fraction) is the projection, using a Van Leer limiter, of the slope onto the left eigenvector $\mathbf{l}^k$:

$$\Delta^{vl}\mathbf{W} = \sum_k \alpha_k \mathbf{r}^k, \tag{38}$$

where

$$\alpha^k = \begin{cases} sign\,(\alpha_C)\min\left(2|\alpha_L^k|,\, 2|\alpha_R^k|,\, |\alpha_C^k|\right) & \text{if } \alpha_R^k.\alpha_L^k > 0, \\ 0 & \text{otherwise} \end{cases} \tag{39}$$

and

$$\alpha_L^k = \mathbf{l}^k \cdot (\mathbf{W}_i - \mathbf{W}_{i-1})\,, \tag{40}$$

$$\alpha_R^k = \mathbf{l}^k \cdot (\mathbf{W}_{i+1} - \mathbf{W}_i)\,, \tag{41}$$

$$\alpha_C^k = \mathbf{l}^k \cdot \left(\frac{\mathbf{W}_{i+1} - \mathbf{W}_{i-1}}{2}\right). \tag{42}$$

The cell that contains the front has one face that is covered, and this requires the following extrapolation to get the left or right state for the Riemann solver (see Figure 3):

$$\mathbf{W}_a^{\pm}\left(x_i \pm \frac{\Delta x}{2}; t^n + \frac{\delta t}{2}\right) = \mathbf{W}^n_{a,\,i\pm1} + \frac{1}{2}\sum_k \alpha^k \left(\pm 1 - \frac{\delta t}{\Delta x}\max(\pm\lambda^k, 0)\right)\mathbf{r}^k. \tag{43}$$

The extrapolation happens across the front and should really take into account the jump conditions. Thus, the left and right states on the covered face are extrapolated back to the front, where a Riemann solve is performed and the final upwind state is extrapolated back to the covered face. For the fluid on the left side, this extrapolation step is as follows. First, we extrapolate the left and right states from the covered face $i + \frac{1}{2}$ back to the front:

$$W_{LL}(x^f, t^{n+\frac{1}{2}}) = W_{L,\,i+\frac{1}{2}} + \frac{x^f - x_{i+\frac{1}{2}}}{\Delta x}\Delta W_L, \tag{44}$$

$$W_{RL}(x^f, t^{n+\frac{1}{2}}) = W_{R,\,i+\frac{1}{2}} + \frac{x^f - x_{i+\frac{1}{2}}}{\Delta x}\Delta W_R. \tag{45}$$

Then, we solve a Riemann problem with the left and right sides being $W_{LL}(x^f, t^{n+\frac{1}{2}})$ and $W_{RL}(x^f, t^{n+\frac{1}{2}})$. This yields $W_L(x^f, t^{n+\frac{1}{2}})$ and takes into account the jump condition. Finally, this quantity is extrapolated back to the covered face:

$$W_{L,\,i+\frac{1}{2}} = W_L(x^f, t^{n+\frac{1}{2}}) - \frac{x^f - x_{i+\frac{1}{2}}}{\Delta x}\Delta W_L. \tag{46}$$

The data at the time centroid is computed from the quantities obtained in (37):

$$\mathbf{W}^{n_c}_{a,\,i+\frac{1}{2}} = \mathbf{W}^{n+\frac{1}{2}}_{a,\,i+\frac{1}{2}} \pm (1 - \alpha_a)\left(\mathbf{W}^{n+\frac{1}{2}}_{a,\,i+\frac{1}{2}} - \mathbf{W}^{n}_{a,\,i+\frac{1}{2}}\right), \tag{47}$$

where n_c is the time centroid.

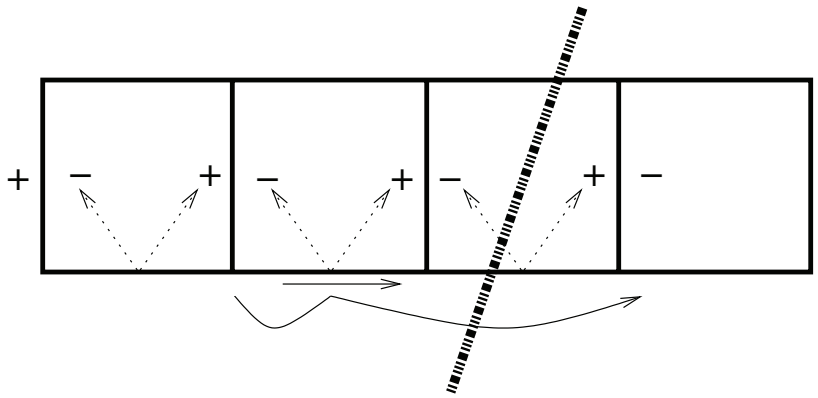


FIG. 3. *Extrapolation near the front.*

2.4.2. Corrector step. We use (20) to advance the solution

$$\bar{\mathbf{U}}_a^{n+1} = \bar{\mathbf{U}}_a^{n,\,n+1} - \Delta t \overline{DF}_a^{C} - \Delta t (1 - \kappa_a^{n+1}) \overline{DF}_a^{NC}. \tag{48}$$

According to (15) and (19), the conservative increment $\overline{DF}^{C}$ taken at the cell centroid is given by

$$\overline{DF}^{C} = \frac{1}{\Delta t} \left(\kappa_a^{n+1} \bar{\mathbf{U}}_a^{n,\,n+1} - \kappa_a^{n} \bar{\mathbf{U}}_a^{n} \right) + \frac{\alpha_{a,\,i+\frac{1}{2}} \bar{\mathbf{F}}_{a,\,i+\frac{1}{2}} - \alpha_{a,\,i-\frac{1}{2}} \bar{\mathbf{F}}_{a,\,i-\frac{1}{2}}}{\Delta x} \pm \frac{\alpha^f}{\Delta x} \bar{\mathbf{F}}^f. \tag{49}$$

To ensure accuracy and stability, the flux through the front $\bar{F}^f$ for a shock is evaluated from the expression of (49) on the low-pressure side Lo:

$$\begin{aligned} \mp \alpha^f \bar{F}^f = {} & \frac{\Delta x}{\Delta t} \left(\kappa_{Lo}^{n+1} \bar{\mathbf{U}}_{Lo}^{n,\,n+1} - \kappa_{Lo}^{n} \bar{\mathbf{U}}_{Lo}^{n} \right) + \frac{\alpha_{Lo,\,i+\frac{1}{2}} \bar{\mathbf{F}}_{Lo,\,i+\frac{1}{2}} - \alpha_{Lo,\,i-\frac{1}{2}} \bar{\mathbf{F}}_{Lo,\,i-\frac{1}{2}}}{\Delta x} \\ & - \kappa_{Lo}^{n+1} \Delta x \overline{DF}_{Lo}^{NC}. \end{aligned} \tag{50}$$

For a contact discontinuity, $\bar{F}^f$ is computed exactly as

$$\bar{F}^f = (0,\, p,\, up)^T, \tag{51}$$

where we have obtained u and p through a Riemann solve using an extrapolation of $\mathbf{W}$ on each side of the front, as pictured in Figure 4.

The nonconservative increment $\overline{DF}^{NC}$ is first computed at the cell center (CC) and then averaged to the cell centroid (Ce):

$$(DF)^{NC} = \frac{\mathbf{F}_{i+\frac{1}{2}}^{n+\frac{1}{2}} - \mathbf{F}_{i-\frac{1}{2}}^{n+\frac{1}{2}}}{\Delta x}, \tag{52}$$

$$\overline{DF}^{NC} = Av \left((DF)^{NC} \right)^{CC \to Ce}. \tag{53}$$

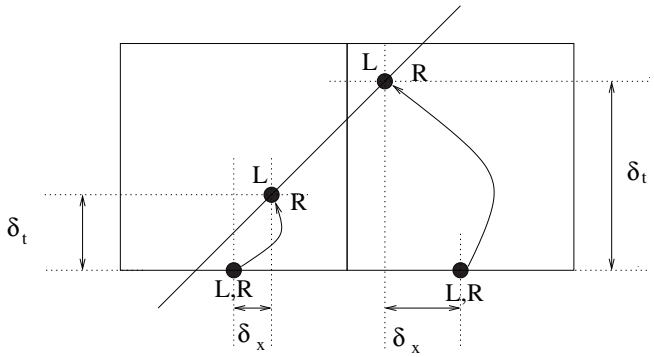


FIG. 4. *Extrapolation on the front.*

2.4.3. Redistribution. To ensure that the method is conservative, we need to redistribute the excess mass to larger adjacent cells:

$$\delta \mathbf{M}_a = -\Delta t \left(1 - \kappa_a^{n+1}\right) \kappa_a^{n+1} \left[\overline{DF}_a^{C} - \overline{DF}_a^{NC}\right] = \sum_k \beta_a^k \mathbf{r}_a^k. \tag{54}$$

The mass increment is projected onto the eigenvectors to redistribute the mass according to characteristics. We denote by $p \in \{-1, 0, 1\}$ the wave family that we are tracking. For a shock ($p \in \{-1, 1\}$), the mass to be redistributed on each side of the shock is

$$\delta \mathbf{M}_L^{red} = \left(\frac{p+1}{2}\right) (\beta_L^p \mathbf{r}_L^p + \beta_R^p \mathbf{r}_R^p) + \sum_{k<p} \beta_L^k \mathbf{r}_L^k + \sum_{k<p} \beta_R^k \mathbf{r}_R^k, \tag{55}$$

$$\delta \mathbf{M}_R^{red} = \left(\frac{1-p}{2}\right) (\beta_L^p \mathbf{r}_L^p + \beta_R^p \mathbf{r}_R^p) + \sum_{k>p} \beta_L^k \mathbf{r}_L^k + \sum_{k>p} \beta_R^k \mathbf{r}_R^k. \tag{56}$$

For a contact, we take into account the reflection and the refraction of the waves to redistribute the mass

$$\delta \mathbf{M}_L^{red} = \beta_L^p \mathbf{r}_L^p + \sum_{k<p} \beta_L^k \mathbf{r}_L^k + \mu_R \sum_{k<p} \beta_R^k \mathbf{r}_R^k + (1 - \mu_L) \sum_{k>p} \beta_L^k \mathbf{r}_L^k, \tag{57}$$

$$\delta \mathbf{M}_R^{red} = \beta_R^p \mathbf{r}_R^p + \sum_{k>p} \beta_R^k \mathbf{r}_R^k + (1 - \mu_R) \sum_{k<p} \beta_R^k \mathbf{r}_R^k + \mu_L \sum_{k>p} \beta_L^k \mathbf{r}_L^k \tag{58}$$

according to acoustic impedances [11], where we have introduced

$$\mu_L = \frac{1/\rho_L c_L}{1/\rho_L c_L + 1/\rho_R c_R}, \tag{59}$$

$$\mu_R = \frac{1/\rho_R c_R}{1/\rho_L c_L + 1/\rho_R c_R}. \tag{60}$$

The new velocity is then obtained by

$$\bar{\mathbf{U}}_{a,i}^{n+1} = \bar{\mathbf{U}}_{a,i}^{n+1} + \frac{1}{1 + \kappa_a^{n+1}} \delta \mathbf{M}_a^{red}, \tag{61}$$

where $i \in \{i^f, i^f - 1\}$ for $a = L$, $i \in \{i^f, i^f + 1\}$ for $a = R$, and i^f is the index of the front at time t^{n+1}.

Now, we can write the conservation property in the form

$$\sum \kappa_a^{n+1} \bar{U}_a^{n+1} = \sum \kappa_a^n \bar{U}_a^n + \text{boundary terms}. \tag{62}$$

3. Algorithm for multimaterial flows. In this section, we revise the algorithm from section 2 to include the case of vastly different materials. We treat the case where one material is described as in section 2 and one material is described by Tait's equations

$$p = A\left(\frac{\rho}{\rho_o}\right)^{\gamma} - B, \tag{63}$$

where $A = 3.001\,10^8$, $B = 3\,10^8$, and $\gamma = 7$ for water.

Equations (1) and (8) remain the same. The matrix A in (9) has the same terms, but the expression of c will be different:

$$c^2 = \frac{dp}{d\rho} = \frac{A\gamma}{\rho_o^{\gamma}}\rho^{\gamma-1}. \tag{64}$$

We follow the same time-stepping procedure as that presented in the previous section. However, we add the following steps.

- When we compute the vector $\mathbf{W}$ given the vector $\mathbf{U}$ on the water side, we use the following relationship:

$$\rho = \mathbf{U}_1, \tag{65}$$

$$u = \frac{\mathbf{U}_2}{\rho}, \tag{66}$$

$$p = A\left(\frac{\rho}{\rho_o}\right)^{\gamma} - B. \tag{67}$$

- After the variables W have been advanced to half time step in (30), we enforce Tait's equation of state by modifying the first component ρ to be equal to

$$\rho = \rho_o\left(\frac{p+B}{A}\right)^{\frac{1}{\gamma}}. \tag{68}$$

  This step is necessary because p and ρ are not independent since they are linked by Tait's equation.
- Finally, the redistribution step (61) is changed so as to not redistribute mass across materials:

$$\bar{\mathbf{U}}_{a,\,i}^{n+1} = \bar{\mathbf{U}}_{a,\,i}^{n+1} + \frac{1}{1+\kappa_a^{n+1}}\delta\mathbf{M}_a. \tag{69}$$

4. Results. The classical approach to verifying the accuracy of this method is to apply it to problems whose solutions are piecewise-smooth and smooth on either side of the tracked front, and to apply Richardson error estimation. Away from the tracked front, the construction of such solutions is routine, since the only constraint is that the initial data be smooth. At the tracked front, the construction of general piecewise-smooth initial data is nontrivial. Obviously, the initial data must satisfy the jump relations. In addition, the solution derivatives must also satisfy appropriate jump relations; otherwise, discontinuities in the derivatives are generated at the front

and propagate into the interior. The one case for which it is easy to verify that the derivatives satisfy the appropriate jump relations is the case where the derivative on either side of the front vanishes at the front. Thus, for the purpose of verifying the accuracy of the method, we use such initial data. We also investigate the accuracy of the method in more classical problems in which the solution or its derivatives have discontinuities, for which we do not expect second-order convergence of the interior method.

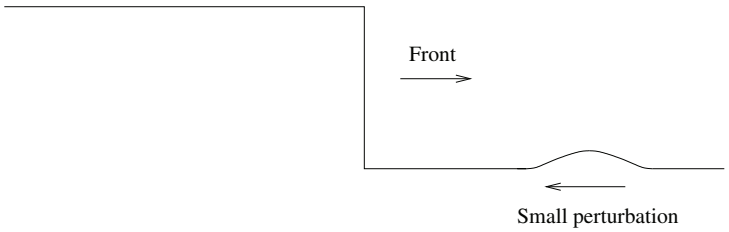


FIG. 5. *Piecewise-smooth solutions.*

4.1. Piecewise-smooth solutions.

4.1.1. Tracking a shock. For this test problem, we superimpose a perturbation in a single characteristic quantity onto a simple shock wave propagating to the right, as shown in Figure 5. The simple shock wave is described by

$$\begin{aligned} \rho_R &= 0.1, \\ u_R &= -11.6, \\ p_R &= 0.5, \\ p_L &= 14 \end{aligned}$$

with ρ_L and p_L determined by the Rankine–Hugoniot conditions and the fluid ideal gas law γ being $\gamma = 1.667$. The computational domain has a length of 1, and the position of the front x^f is at $x^f = 0.5001$. We use four resolutions in space to compute the convergence rates: $\Delta x = 1/100$, $\Delta x = 1/200$, $\Delta x = 1/400$, and $\Delta x = 1/800$. The CFL is chosen to be $\frac{1}{2}$.

The shape of the perturbation density is described as

$$\rho(x) = \rho_R \left(1 + a \left(1 - \left(\frac{x - x_i}{x_f - x_i} \right)^2 \right)^4 \right), \tag{70}$$

where $x_i = 0.65$ is the position of the beginning of the perturbation, $x_f = 0.85$ is the position of the end of the perturbation, and $a = (\rho_L - \rho_R)/10$ is the amplitude of the perturbation.

We take the perturbation to be characteristic in one of the waves, and the velocity and pressure for a plus or minus wave are given as

$$u^{\pm}(x) = u_R \pm \frac{2}{\gamma - 1} \sqrt{\frac{\gamma p_R}{\rho_R}} \left(\frac{\rho(x)}{\rho_R} \right)^{\frac{\gamma-1}{2}-1}, \tag{71}$$

$$p(x) = p_R \left(\frac{\rho(x)}{\rho_R} \right)^{\gamma}. \tag{72}$$

Tables 1–3 show the errors and the convergence rates after a time that corresponds to 200 time steps taken on the coarsest grid. As expected, the convergence is second-order in both the L^1 and the L^∞.

TABLE 1
Errors and convergence rates for a contact-wave perturbation.

	ρ		ρu		ρE	
	L^1	L^∞	L^1	L^∞	L^1	L^∞
100/200	7.84×10^{-5}	1.42×10^{-3}	2.25×10^{-4}	2.50×10^{-3}	1.34×10^{-3}	1.94×10^{-2}
200/400	3.64×10^{-5}	9.00×10^{-4}	8.27×10^{-5}	1.58×10^{-3}	3.14×10^{-4}	3.29×10^{-3}
400/800	1.01×10^{-5}	2.92×10^{-4}	2.14×10^{-5}	5.15×10^{-4}	6.61×10^{-5}	7.70×10^{-4}
800/1600	2.45×10^{-6}	7.14×10^{-5}	5.15×10^{-6}	1.25×10^{-4}	1.52×10^{-5}	1.98×10^{-4}
100/200/400	1.10	0.65	1.44	0.66	2.09	2.56
200/400/800	1.85	1.62	1.95	1.61	2.25	2.09
400/800/1600	2.04	2.03	2.05	2.04	2.12	1.96

TABLE 2
Errors and convergence rates for a minus-wave perturbation.

	ρ		ρu		ρE	
	L^1	L^∞	L^1	L^∞	L^1	L^∞
100/200	5.06×10^{-5}	8.11×10^{-4}	1.99×10^{-4}	3.11×10^{-3}	1.59×10^{-3}	3.63×10^{-2}
200/400	2.32×10^{-5}	6.02×10^{-4}	6.17×10^{-5}	1.06×10^{-3}	3.31×10^{-4}	7.96×10^{-3}
400/800	7.96×10^{-6}	2.65×10^{-4}	1.82×10^{-5}	4.68×10^{-4}	7.79×10^{-5}	1.70×10^{-3}
800/1600	2.02×10^{-6}	7.41×10^{-5}	4.45×10^{-6}	1.30×10^{-4}	1.87×10^{-5}	4.56×10^{-4}
100/200/400	1.12	0.43	1.69	1.55	2.26	2.19
200/400/800	1.54	1.18	1.76	1.18	2.09	2.22
400/800/1600	1.98	1.84	2.03	1.85	2.06	1.90

TABLE 3
Errors and convergence rates for a plus-wave perturbation.

	ρ		ρu		ρE	
	L^1	L^∞	L^1	L^∞	L^1	L^∞
100/200	6.52×10^{-5}	1.10×10^{-3}	1.76×10^{-4}	1.94×10^{-3}	9.58×10^{-4}	6.87×10^{-3}
200/400	2.15×10^{-5}	4.62×10^{-4}	5.00×10^{-5}	8.15×10^{-4}	2.01×10^{-4}	1.67×10^{-3}
400/800	5.26×10^{-6}	1.25×10^{-4}	1.17×10^{-5}	2.21×10^{-4}	4.31×10^{-5}	3.25×10^{-4}
800/1600	1.25×10^{-6}	3.11×10^{-5}	2.78×10^{-6}	5.48×10^{-5}	9.95×10^{-6}	7.63×10^{-5}
100/200/400	1.60	1.25	1.82	1.25	2.25	2.04
200/400/800	2.03	1.88	2.09	1.88	5.54	2.36
400/800/1600	2.07	2.00	2.07	2.01	2.11	2.09

4.1.2. Tracking a contact discontinuity. In this example, the underlying wave is a contact wave described by

$$\begin{aligned} \rho_L &= 0.353, \quad \rho_R = 0.1, \\ u_L &= 0.5, \quad\quad u_R = 0.5, \\ p_L &= 1, \quad\quad\ p_R = 1, \end{aligned}$$

and we superimpose the same plus-wave perturbation as the one described in the previous section. Table 4 shows the errors and the convergence rates for this example. As expected, we obtain second order in L^1 and first order in L^∞ since the boundary is characteristic. The CFL is chosen to be $\frac{1}{2}$.

TABLE 4
Errors and convergence rates for a plus-wave perturbation for a tracked discontinuity.

	ρ		ρu		ρE	
	L^1	L^∞	L^1	L^∞	L^1	L^∞
100/200	2.34×10^{-4}	2.61×10^{-3}	3.43×10^{-4}	4.01×10^{-3}	2.03×10^{-3}	2.39×10^{-2}
200/400	1.31×10^{-4}	2.26×10^{-3}	1.99×10^{-4}	3.51×10^{-3}	1.19×10^{-3}	2.09×10^{-2}
400/800	4.19×10^{-5}	1.45×10^{-3}	6.37×10^{-5}	2.18×10^{-3}	3.78×10^{-4}	1.32×10^{-2}
800/1600	1.35×10^{-5}	7.68×10^{-4}	2.06×10^{-5}	1.17×10^{-3}	1.22×10^{-4}	7.01×10^{-3}
1600/3200	5.05×10^{-6}	3.72×10^{-4}	7.64×10^{-6}	5.64×10^{-4}	4.55×10^{-5}	3.39×10^{-3}
3200/6400	1.27×10^{-6}	1.21×10^{-4}	1.91×10^{-6}	1.84×10^{-4}	1.13×10^{-5}	1.11×10^{-3}
100/200/400	0.84	0.20	0.78	0.19	0.77	0.19
200/400/800	1.64	0.64	1.64	0.69	1.66	0.66
400/800/1600	1.63	0.92	1.63	0.90	1.63	0.91
800/1600/3200	1.42	1.04	1.43	1.05	1.42	1.05
1600/3200/6400	1.99	1.62	2.00	1.61	2.01	1.61

4.2. Shock tube. The first test is the classical shock tube problem discussed in Sod [18]. Initially, the left and right states are

$$\begin{aligned} \rho_L &= 1, \quad \rho_R = 0.125, \\ u_L &= 0, \quad u_R = 0, \\ p_L &= 1, \quad p_R = 0.1 \end{aligned}$$

with $\gamma = 1.4$.

When the membrane separating the two states is broken, three different waves appear: a rarefaction wave, a contact wave, and a shock. For test purposes, we consider two cases. We track the shock and capture the contact as shown with $\times$'s in Figure 6, and we track the contact and capture the shock as shown with $\circ$'s in Figure 6. A CFL of $\frac{1}{2}$ is used for both simulations, and the curves are exactly superimposed, except on the actual fronts, as expected. When the contact is tracked, a few points immediately to the right mark a small dip in the curve for the density. This is a result of errors generated at the very beginning when the three waves separate [15].

4.3. Interactions of two simple waves.

4.3.1. Tracked shock overtaking a rarefaction wave. In this example, we consider a shock wave and a rarefaction wave of the same family, propagating in opposite directions, as shown by the dotted line in Figure 7. The shock wave is tracked. We choose a CFL of $\frac{1}{2}$.

As expected [5], the shock slowly overtakes the rarefaction wave, weakening both the shock and the rarefaction wave. In the back of the shock, a contact wave and a compression wave are created. Table 5 shows that the position of the front converges to second order, and Table 6 shows the convergence rates in L^1-norm. In this example, the convergence is not second-order because the initial data is not smooth, and the loss of smoothness translates into a loss of accuracy.

4.3.2. Rarefaction wave overtaking a tracked contact wave. In this test problem, we consider a rarefaction wave and a contact wave, which are initially described by the dotted lines in Figure 8. The contact wave is tracked. As the simulation progresses, we expect the rarefaction wave to move toward the right and slowly overtake the contact wave. This is what is obtained and shown by the solid lines in Figure 8. The convergence rate for this problem is expected to be second-order in L^1; the convergence rates are given in Table 7. We obtain second-order convergence once the asymptotic regime is reached. A CFL of $\frac{1}{2}$ is used.

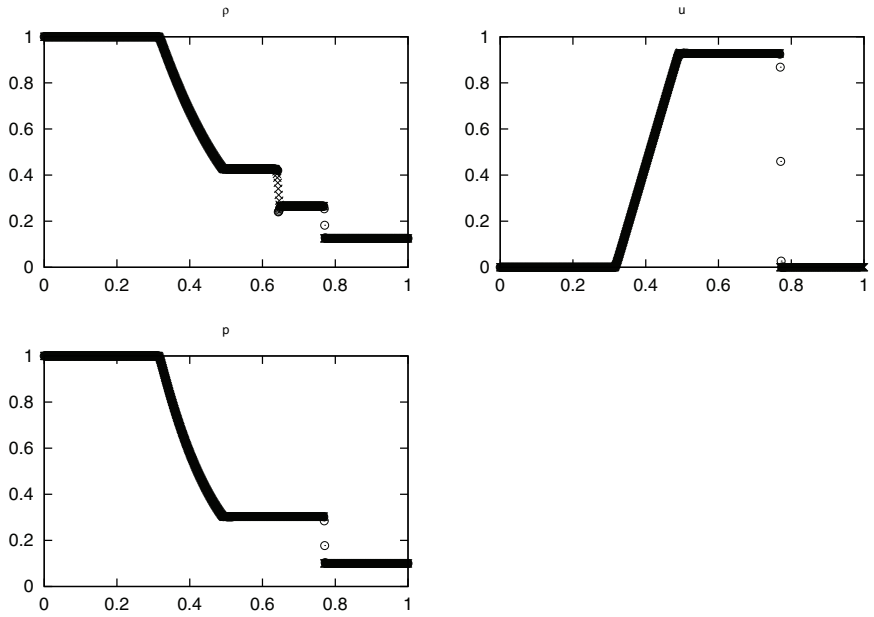


FIG. 6. *Density, velocity, and pressure for Sod's shock tube problem with tracked shock (crosses) and tracked contact (circles).* $h = 1/800$.

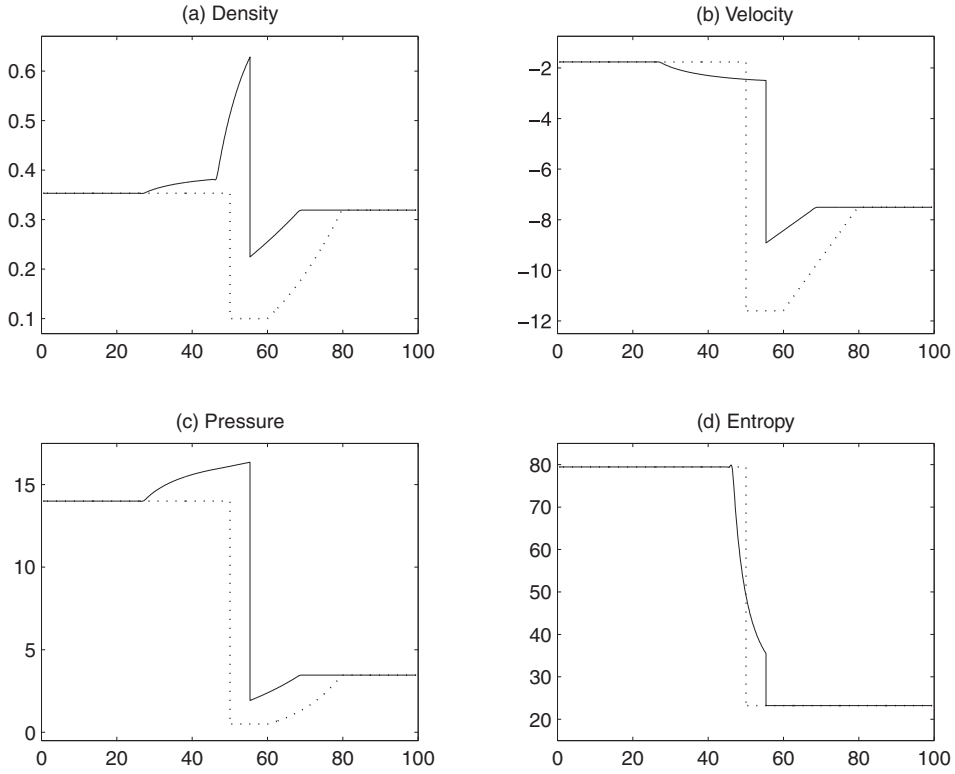


FIG. 7. *Shock overtaking a rarefaction wave. The dotted line is initial conditions.*

TABLE 5
Convergence rate for the position of the front x^f.

N_{pts}	100	200	400	800
$x^f_{coarse} - x^f_{fine}$	7.03×10^{-6}	1.29×10^{-6}	3.27×10^{-7}	9.09×10^{-8}
Convergence rate		1.70	1.99	1.84

TABLE 6
Errors and convergence rates in L^1 for a shock-rarefaction interaction.

	ρ	ρu	ρE
100/200	2.76×10^{-4}	1.16×10^{-3}	9.98×10^{-3}
200/400	9.21×10^{-5}	4.65×10^{-4}	3.94×10^{-3}
400/800	2.98×10^{-5}	1.59×10^{-4}	1.43×10^{-3}
800/1600	1.00×10^{-5}	5.31×10^{-5}	4.86×10^{-4}
1600/3200	3.56×10^{-6}	1.84×10^{-6}	1.56×10^{-4}
100/200/400	1.59	1.32	1.34
200/400/800	1.63	1.55	1.46
400/800/1600	1.58	1.58	1.56
800/1600/3200	1.49	1.53	1.64

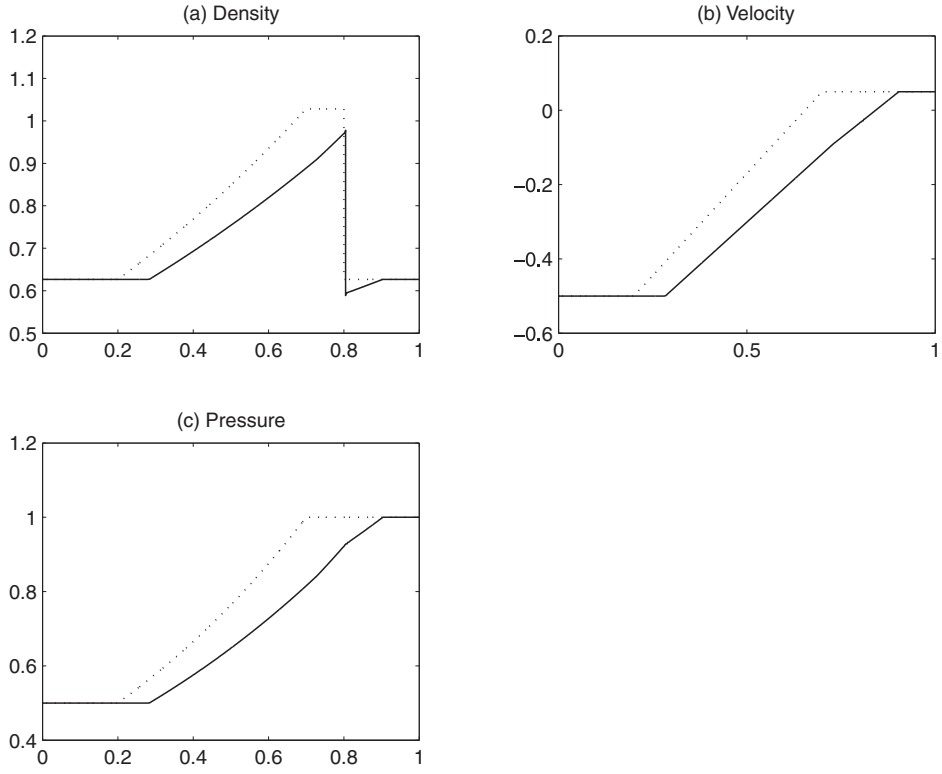


FIG. 8. *Rarefaction overtaking a contact wave. The dotted line is initial conditions.*

TABLE 7
Errors and convergence rates in L^1 for a rarefaction-contact interaction.

	ρ	ρu	ρE
100/200	2.60×10^{-4}	2.43×10^{-4}	9.06×10^{-4}
200/400	1.18×10^{-4}	2.18×10^{-4}	8.75×10^{-4}
400/800	9.12×10^{-5}	1.58×10^{-4}	8.01×10^{-4}
800/1600	4.63×10^{-5}	6.40×10^{-5}	1.75×10^{-4}
1600/3200	1.27×10^{-5}	1.22×10^{-5}	$4.54 \times \times 10^{-5}$
100/200/400	0.37	0.15	0.05
200/400/800	0.37	0.47	0.13
400/800/1600	0.98	1.30	2.19
800/1600/3200	1.87	2.39	1.98

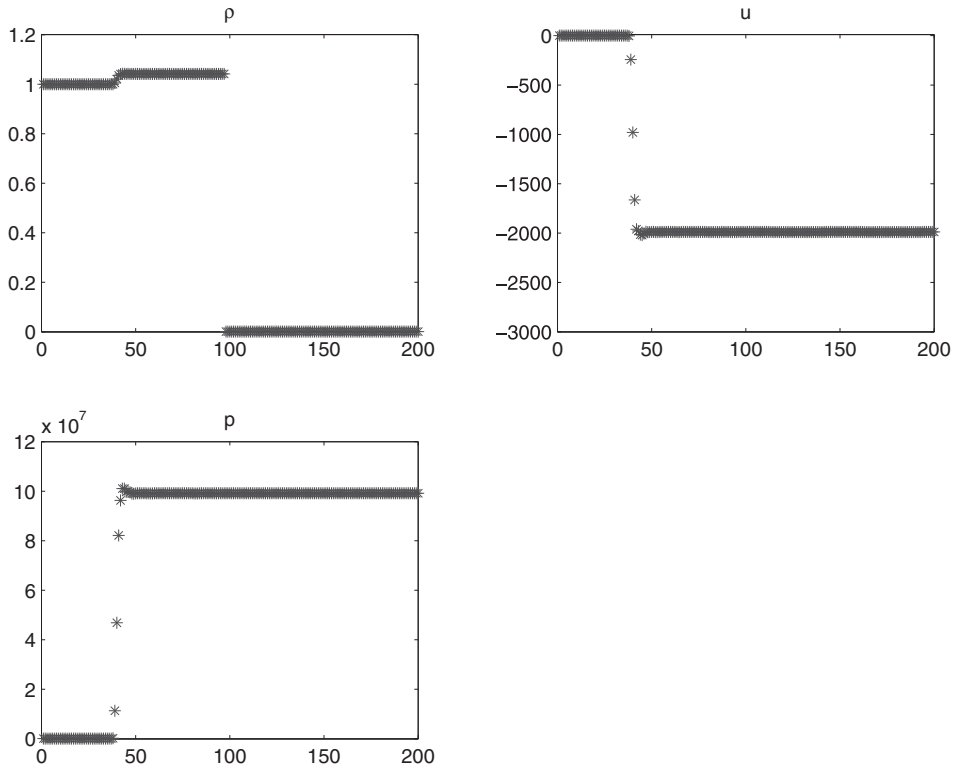


FIG. 9. *Multimaterial case with water on the left and air on the right. A compression wave is propagating in the water.*

4.4. Multimaterial case. In this test case, the interface is in the middle, and the right and left states are

$$\begin{aligned} \rho_L &= 1, & \rho_R &= 1.29e-3, \\ u_L &= 0, & u_R &= 0, \\ p_L &= 10^5, & p_R &= 10^8. \end{aligned}$$

Right after the initial state, a compression wave forms in the water and propagates to the left. The results after 800 time steps are shown in Figure 9. One should also

note that the front between the water and the air has moved significantly during that time, about $2.3\Delta x$.

5. Conclusion. This paper presents a new conservative algorithm for tracking a moving discontinuity, whether it be a shock or a contact, generalizing the ideas in [8] to the case of moving free boundaries. The method is based on a finite-volume discretization derived from applying the divergence theorem on the irregular regions in space-time obtained by intersecting Cartesian cells with the region on either side of the front. Small-cell stability problems are avoided by updating irregular control volumes using a linear combination of conservative and nonconservative methods, combined with redistribution to maintain conservation. The resulting method has a truncation error that is $O(h)$ at the irregular cells and $O(h^2)$ elsewhere, leading to a method whose solution error is $O(h^2)$. The previous methods for free boundaries using this approach had $O(1)$ truncation errors at the tracked front, leading to a method that was no more than first-order accurate in solution error. We also presented an extension to a multimaterial case where the interface separates very different materials, namely, water and air, that have different equations of state.

The extension of this method to multiple dimensions will require the application of the machinery in [8] to compute the required time-centered fluxes for the conservative and nonconservative discretizations. To carry this out, we will need to generalize two of the constructions in this paper. One is relatively straightforward: the calculation of $O(h^2)$ values at cell centers based on the values at cell centroids, corresponding to (24). We expect that a least-squares approach to computing the gradient, similar to that used in [17], combined with monotonicity constraints, will provide a sufficiently accurate and robust method. The second issue, which is the calculation of second-order accurate dynamics of the front location and the corresponding finite-volume discretization data, requires a more significant effort. In the one-dimensional case, the evolution is done using a second-order accurate integration of the ODE for a Lagrangian point representing the front, and the geometric constructions of the finite-volume discretization data are trivial. The use of Lagrangian points in more than one dimension requires regular remapping of the Lagrangian points, which is difficult to do while maintaining second-order accuracy, particularly in three dimensions. The approach we will take is based on the use of a level-set representation of the front, along the lines of that described in [1]. In this approach, the speed is computed at the front using jump relations and extended to the interior using transport equations, leading to a method that preserves distance functions. Given an implicit function representation in space-time, it is then straightforward to compute the finite-volume discretization information using repeated applications of the divergence theorem. However, this approach will require the development of some extensions to the level-set approach in [1] that will preserve second-order accuracy while preserving robustness in the presence of kinks in the front.

REFERENCES

[1] D. Adalsteinsson and J. A. Sethian, *Transport and diffusion of material quantities on propagating interfaces via level set methods*, J. Comput. Phys., 185 (2003), pp. 271–288.

[2] B. N. Azarenok and T. T. Tang, *Second-order Godunov-type scheme for reactive flow calculations on moving meshes*, J. Comput. Phys., 206 (2005), pp. 48–80.

[3] J. Bell, P. Colella, and M. Welcome, *Conservative front-tracking for inviscid compressible flows*, in Proceedings of the AIAA 10th Computational Fluid Dynamics Conference, Honolulu, HI, 1991, pp. 814–822.

[4] I.-L. Chern and P. Colella, *A Conservative Front Tracking Method for Hyperbolic Conservations Laws*, LLNL Report UCRL-97200, Lawrence Livermore National Laboratory, Livermore, CA, 1987.
[5] P. Colella, *Glimm's method for gas dynamics*, SIAM J. Sci. Statist. Comput., 3 (1982), pp. 76–110.
[6] P. Colella, *Volume-of-fluid methods for partial differential equations*, in Godunov Methods—Theory and Applications, Kluwer/Plenum, New York, 2001, pp. 161–177.
[7] P. Colella, D. Graves, T. Ligocki, and B. Van Straalen, *EBAMRGodunov Documentation*, Lawrence Berkeley National Laboratory, Berkeley, CA.
[8] P. Colella, D. T. Graves, B. J. Keen, and D. Modiano, *A Cartesian grid embedded boundary method for hyperbolic conservation laws*, J. Comput. Phys., 211 (2006), pp. 347–366.
[9] J. Glimm, X. Li, Y. Liu, Z. Xu, and N. Zhao, *Conservative front tracking with improved accuracy*, SIAM J. Numer. Anal., 41 (2003), pp. 1926–1947.
[10] J. Glimm, X. L. Li, and N. Zhao, *Conservative front tracking and level set algorithms*, Proc. Natl. Acad. Sci. USA, 98 (2001), pp. 14198–14201.
[11] J. Hilditch and P. Colella, *A projection method for low Mach number fast chemistry reacting flow*, in Proceedings of the AIAA Aerospace Sciences Meeting, Reno, NV, 1997, AIAA paper 97-0263.
[12] J. M. Hyman, *Numerical methods for tracking interfaces*, Phys. D, 12 (1984), pp. 396–407.
[13] P. Lax and L. Baumhoff, *On Discontinuous Initial Value Problems for Nonlinear Equations and Finite Difference Schemes*, Technical report LAMS-1332, Los Alamos Scientific Laboratory, Los Alamos, NM, 1952.
[14] G. Miller and P. Colella, *A conservative three-dimensional Eulerian method for coupled solid-fluid shock capturing*, J. Comput. Phys., 183 (2002), pp. 26–82.
[15] W. F. Noh, *Errors for calculations of strong shocks using an artificial viscosity and an artificial heat flux*, J. Comput. Phys., 72 (1987), pp. 78–120.
[16] P. Schwartz, D. Adalsteinsson, P. Colella, A. P. Arkin, and M. Onsum, *Numerical computation of diffusion on a surface*, Proc. Natl. Acad. Sci. USA, 102 (2005), pp. 11151–11156.
[17] P. Schwartz, M. Barad, P. Colella, and T. J. Ligocki, *A Cartesian grid embedded boundary method for the heat equation and Poisson's equation in three dimensions*, J. Comput. Phys., 211 (2006), pp. 531–550.
[18] G. Sod, *A survey of several finite difference methods for systems of nonlinear hyperbolic conservation laws*, J. Comput. Phys., 27 (1978), pp. 1–31.
[19] H. Tang and T. Tang, *Adaptive mesh methods for one- and two-dimensional hyperbolic conservation laws*, SIAM J. Numer. Anal., 41 (2003), pp. 487–515.
[20] P. Woodward, *Piecewise-parabolic methods for astrophysical fluid dynamics*, in Proceedings of the NATO Advanced Workshop on Astrophysical Radiation Hydrodynamics, Munich, West Germany, 1982.