Available online at www.sciencedirect.com
SCIENCE DIRECT®
Journal of Computational Physics 213 (2006) 500–529
JOURNAL OF COMPUTATIONAL PHYSICS
www.elsevier.com/locate/jcp

Adaptive characteristics-based matching for compressible multifluid dynamics

R.R. Nourgaliev *, T.N. Dinh, T.G. Theofanous

Center for Risk Studies and Safety, University of California at Santa Barbara, 6740 Cortona Drive, Goleta, CA 93117, USA

Received 15 March 2005; received in revised form 14 July 2005; accepted 19 August 2005
Available online 19 October 2005

Abstract

This paper presents an evolutionary step in sharp capturing of shocked, high acoustic impedance mismatch (AIM) interfaces in an adaptive mesh refinement (AMR) environment. The central theme which guides the present development addresses the need to optimize between the algorithmic complexities in advanced front capturing and front tracking methods developed recently for high AIM interfaces with the simplicity requirements imposed by the AMR multi-level dynamic solutions implementation. The paper shows that we have achieved this objective by means of relaxing the strict conservative treatment of AMR prolongation/restriction operators in the interfacial region and by using a natural-neighbor-interpolation (NNI) algorithm to eliminate the need for ghost cell extrapolation into the other fluid in a characteristics-based matching (CBM) scheme. The later is based on a two-fluid Riemann solver, which brings the accuracy and robustness of front-tracking approach into the fast local level set front-capturing implementation of the CBM method. A broad set of test problems (including shocked multi-gaseous media, bubble collapse, underwater explosion and shock passing over a liquid drop suspended in a gaseous medium) was performed and the results demonstrate that the fundamental assumptions/approximations made in modifying the AMR prolongation/restriction operators and in using the NNI algorithm for interfacial treatment are acceptable from the accuracy point of view, while they enable an effective implementation and utility of the structured AMR technology for solving complex multiphase problems in a highly compressible setting.

Keywords: Compressible multiphase flows; Characteristics-based methods; Front capturing; Level set; Structured adaptive mesh refinement

1. Introduction

The purpose of this paper is to report on an evolutionary step in sharp capturing of shocked, high acoustic impedance mismatch (AIM) interfaces, in an adaptive mesh refinement (AMR) environment. In particular, we are interested in bridging the fundamental advances made by the level set (LS) approach and the ghost fluid

* Corresponding author. Tel.: +1 805 8934939; fax: +1 805 8934927.
E-mail addresses: robert@engr.ucsb.edu (R.R. Nourgaliev), nam@engr.ucsb.edu (T.N. Dinh), theo@engr.ucsb.edu (T.G. Theofanous).

0021-9991/$ - see front matter
doi:10.1016/j.jcp.2005.08.028

method (GFM) to utility in understanding and prediction of practical, multidimensional situations (hence the AMR need) involving such flows.

Starting with the choice[1] of structured adaptive mesh refinement (SAMR)[2] of Berger and co-workers [5–7], the major challenge is in the extrapolation step needed, at the end of each time step in the Runge–Kutta time advancement, for populating the ghost cells, several layers deep, depending on the stencil, on either side of the interface. This extrapolation can be based on either the partial-differential-equation (PDE) or the fast-marching (FM) algorithms. The PDE method requires pseudo-time marching to steady-state, and we found it difficult to incorporate into the already sophisticated physical time advancement of AMR. On the other hand, we found the FM-based extrapolation, a space-marching procedure founded on the Heap Sort algorithm, rather impractical to implement in the nested hierarchy of SAMR grids,[3] and its parallel structure.

Further difficulties derive from the present high AIM setting (as is the case for gas–liquid interfaces[4,5]), known to lead to loss of performance in the original GFM [13]. While several remedies are now at hand [14,24,29,35], they all rely on extrapolation techniques to populate ghost cells, leading to difficulties in AMR implementation as mentioned above. On the other hand, and very significantly, for high AIM systems one encounters also difficulties that derive from the AMR itself. In particular, we have found that the conservative coarse-to-fine and fine-to-coarse inter-level prolongation/restriction operators (of the SAMR algorithm [5]) lead to instability and disconvergence. This difficulty is somewhat akin to the pressure oscillation problems rooted at the "naive conservative flux treatment" near multi-material interfaces [1,13,26]; that is, coarsening/refinement of total energy leads to unacceptable pressure fields due to significant difference of component equations of state in gas–liquid flows.

The end-product of our efforts in addressing these issues is a "front capturing" method that is strongly influenced, in basic philosophy at least, by "front-tracking" fundamentals [9,17].

The key step is building a Riemann-solver-based boundary condition treatment on top of a fast local level set (FLLS) approach. Implementation involves dynamic generation of markers on the sub-cell level using the LS function, and the use of these markers for correction of the numerical fluxes near the interface. This correction is based on pseudo-multidimensional two-fluid Riemann solution in combination with a linear interpolation/extrapolation (depending on the subcell position of the interface) of the fluxes and the natural-neighbor-interpolation (NNI) algorithm [47,54]. The later is based on concepts of constructing Voronoi diagram on a sample of irregular datasets. Therefore, there is no need for ghost cells and thus we avoid the PDE and FM extrapolation issues discussed above. Implementation of this whole treatment to multi-dimensions is done dimension-by-dimension.

The other key ingredient of our approach is development of locally (within one cell from the interface) non-conservative prolongation/restriction operators used in coupling the multi-level solutions of the AMR algorithm. The particular implementation, based on the NNI approach mentioned above, seems to provide just about the right amount of flexibility in the numerics, without deleterious effects on the overall performance of the scheme.

Our plan for testing is as follows. We start with basic shock-tube type tests in 1D, with gas–gas, and gas–liquid systems, under extremely strong shocking (shock Mach numbers of 31 and 4, respectively). We check sensitivity to grid resolution, to spatial discretization scheme, and flux treatment; the standard being, of course, the analytical solutions. As next demonstration we select single-mode Richtmyer–Meshkov instability development in an air-SF_6 system, extending well into the non-linear regime. Then, we consider a combined Richtmyer–Meshkov and Kelvin–Helmholtz instability/mixing case for which well-qualified experimental data are available [21]. This involves an R_{22} (refrigerant) bubble in air, a shock Mach number of 1.22, and highly revealing comparisons are made in terms of Schlieren patterns. Next, we simulate bubble collapse under extremely strong shock waves (pressure ratio 10^4). Here we are interested in gains due to grid resolution (using

[1] The alternative, tree-based approach (i.e. [30]) is known to be problematic for compressible flows [31].

[2] A SAMR extension to incompressible flows is also available now [50].

[3] The FM-based extrapolation is implemented on the tree-based AMR for incompressible flow by Losasso et al. [30].

[4] Extension of SAMR algorithm to gas–gas compressible flows was presented by Quirk and Karni [41].

[5] SAMR implementation of the VOF method for problems involving multiple condensed phases is discussed by Miller and Puckett [33]. To our knowledge, this is the only paper that deals with high acoustic impedance mismatch interface in AMR environment.

up to six levels of adaptation), especially in revealing important fine structures (hypervelocity micro-jets, internally developed shock waves, and blast waves) at the final stages of collapse. A related test is provided at the opposite extreme of scales by considering an underwater explosion on a domain scale of 100 by 50 ms. This is made possible by the parallel capability of the code, and the use of six levels of adaptation (refinement ratio of two), which gave memory savings by a factor of 35. The peak pressure ratio here is $\sim 10^4$, and such simulations are of interest for characterizing the generated free-surface momentum wave (a measure of resulting damage potential). We leave for last the most challenging of all tests, that of a shock passing over a cylindrical liquid mass suspended in air. We show results of these first-of-a-kind simulations at several levels of grid resolution, and we discuss the root cause of the numerical challenge. While all test cases in this work were made in 1D and 2D, in inviscid formulation, we note that the method is readily extendable to 3D and viscous flows.

2. Governing equations

2.1. Conservation laws

Neglecting viscous, heat transfer and surface tension effects, two-dimensional multifluid compressible flows are described by the compressible Euler equations [27]:

$$\mathbf{U}_t + [\vec{\mathscr{F}}(\mathbf{U})]_x + [\vec{\mathscr{G}}(\mathbf{U})]_y = 0, \tag{1}$$

where t is time; $\boldsymbol{x} = (x, y)^{\mathrm{T}}$ is the position-vector; $\mathbf{U}$, $\vec{\mathscr{F}}$ and $\vec{\mathscr{G}}$ are the vectors of the conservation variables and physical fluxes in x and y directions:

$$\mathbf{U} = \begin{pmatrix} \rho \\ m_x \\ m_y \\ E \end{pmatrix}, \quad \vec{\mathscr{F}} = \begin{pmatrix} m_x \\ P + m_x u \\ m_x v \\ u(P+E) \end{pmatrix}, \quad \vec{\mathscr{G}} = \begin{pmatrix} m_y \\ m_y u \\ P + m_y v \\ v(P+E) \end{pmatrix}. \tag{2}$$

Here, ρ, P, $\boldsymbol{u} = (u,v)^{\mathrm{T}}$, $\boldsymbol{m} = \rho(u,v)^{\mathrm{T}}$, $e = i + \frac{u^2+v^2}{2}$, $E = \rho e$ and i are the fluid density, static pressure, velocity vector, momentum vector, specific total energy, total energy and specific internal energy, respectively. Hereafter, we will denote the vector of primitive variables as $\boldsymbol{V} = (P, \rho, u, v)^{\mathrm{T}}$.

2.2. Level set equation

Multimaterial interfaces are captured using the level set (LS) method [39,40,46], which is based on the solution of the following Hamilton–Jacobi equation for a signed distance function φ:

$$\partial_t \varphi + \mathbf{w} \cdot \nabla \varphi = 0, \tag{3}$$

where $\mathbf{w}$ is a speed function vector, for which several options can be considered as described below. Zero-level of the level set function $\varphi(\boldsymbol{x}) = 0$ represents a multimaterial interface which separates the computational domain into "negative" and "positive" fluids.

2.3. Equation of state

To complete the mathematical model, it is necessary to specify equations of state (EOS) for each of the fluids involved. In the present study, we use the *stiffened gas equation of state* [19], which can be written as

$$P = (\gamma - 1)\rho i - P_0, \quad P_0 = \gamma \Pi, \tag{4}$$

where the constants γ and Π are parameters specific to the material. This equation of state is a simple generalization of the ideal gas law and is known to provide a reasonable approximation of thermodynamic processes in gases, liquids and also in solids under high pressure conditions. The values of parameters γ and Π for some gases, liquids and solids are given in [45]. Other choices of EOS, such as the Tait for example, may present advantages specific to an application, and they can be readily incorporated.

3. Key elements of SAMR

We employ the structured adaptive mesh refinement (SAMR) algorithm, originally developed by Berger and co-authors [5–7]. The SAMR algorithm is based on a sequence of nested, logically rectangular meshes. Let $\underline{h_k} = \{h_0, h_1, \ldots, h_k\}|_{k=0,\ldots,L-1}$ denote a collection of mesh spacings, where h_k is a mesh spacing of a SAMR grid level k subject to $h_{k+1} \leqslant h_k$. A SAMR grid $\Omega^{\underline{h}_{L-1}}$ is a nested hierarchy of L grid levels $\Omega^{h_0} \supset \Omega^{h_1} \supset \cdots \supset \Omega^{h_{L-1}}$, where the coarsest grid Ω^{h_0} covers the entire computational domain. Each level Ω^{h_k} consists of a union of M_k logically rectangular regions, or patches, $\mathscr{G}_{k,m}|_{m=0,\ldots,M_k}$, at the same grid resolution h_k. Fig. 1(a) shows an example of a hierarchy of SAMR grids with three levels of adaptation. An example of a typical SAMR mesh with grid refinement around the interface and the shock is given in Fig. 1(b). Note that the levels are nested, but the patches on different levels are not, and the patches on the same level may overlap, i.e., $\mathscr{G}_{k,i} \cap \mathscr{G}_{k,j} \neq 0|_{i \neq j}$. However, the requirement is that the discrete solution must be independent of how Ω^{h_k} is decomposed into patches $\mathscr{G}_{k,m}$. Thus, a point $\mathbf{x}$ of the computational domain in general may exist on several grids. The solution vector $\mathbf{U}(\mathbf{x})$ is taken from the finest level. If there are several equally fine grids containing the point, one can use the value from any of them, since the solution on the intersection of overlapping same-level grids is identical.

Grids are refined in both time and space, using the same mesh refinement ratio $r = \frac{h_k}{h_{k+1}}$, i.e., $\frac{\Delta t_0}{h_0} = \frac{\Delta t_1}{h_1} = \cdots = \frac{\Delta t_{L-1}}{h_{L-1}}$. Thus, the same explicit difference scheme is stable on all levels. As a result, more time steps are taken on the finer grids than on the coarser grids. At the same time, the smallest time step of the finest level is not imposed globally.

Time update of a solution on a hierarchy is organized in such a way, as to proceed sequentially from the coarser to finer grids. More specifically, before updating at any level, the next-coarser level solution must be already available, so as to allow the complete inter-level communications needed for populating each patch's ghost cells by interpolation in both time and space. For parallelization, all patches of a hierarchy are distributed between different processors. In this work, the communication routines are provided by the "structured adaptive mesh refinement application infrastructure" (SAMRAI), developed in Lawrence Livermore National Laboratory [23,44,55].

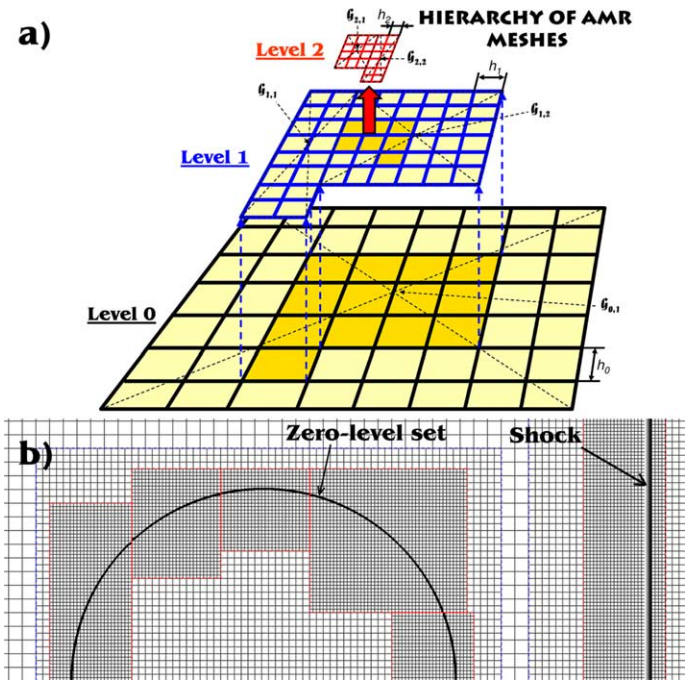

Fig. 1. (a) Example of a hierarchy of SAMR meshes. (b) Example of a SAMR mesh with three levels of adaptation and refinement ratio 2.

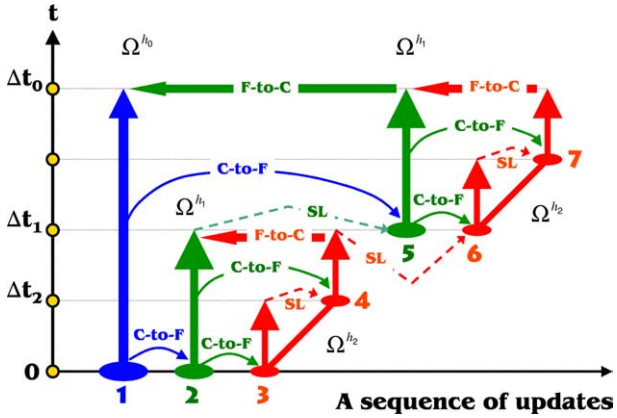

Fig. 2. A sequence of time updates inter-level communications on a hierarchy of three SAMR levels with refinement ratio 2. *Notation*: C-to-F is "coarse-to-fine", F-to-C is "fine-to-coarse" and SL is "same-level".

As an example of the time update sequence on a hierarchy, let us consider AMR with three levels, refined with ratio 2. In this case, the task consists of seven steps, as illustrated in Fig. 2. Each update of the first (Ω^{h_1}) and second (Ω^{h_2}) levels must be preceded by (a) communication with all neighbor patches of the same level and (b) coarse-to-fine interpolation in both time and space, all needed to fill-in ghost cells around all patches. At the end of each fine-grid update matching the coarser-level updates (e.g., step 4 and 7 for level 2; and step 5 for level 1), the fine-to-coarse interpolation is applied to overwrite/match solution on the coarser level. As one can see from the above-given example and Fig. 2, the synchronization of AMR patch updates on different levels is rather complex. Incorporation of additional pseudo-time marching (as in PDE-extrapolation) or space-marching (as in FM-extrapolation) may impose great difficulties.

4. Overview of the numerical approach

As illustrated in Fig. 3, time advancement on a SAMR hierarchy of grids requires the coordinated performance of three tasks; namely, inter-patch communication, generation/disposal of patches, and time update on a patch. As noted already, *inter-patch communication* is necessary for synchronization of time updates on

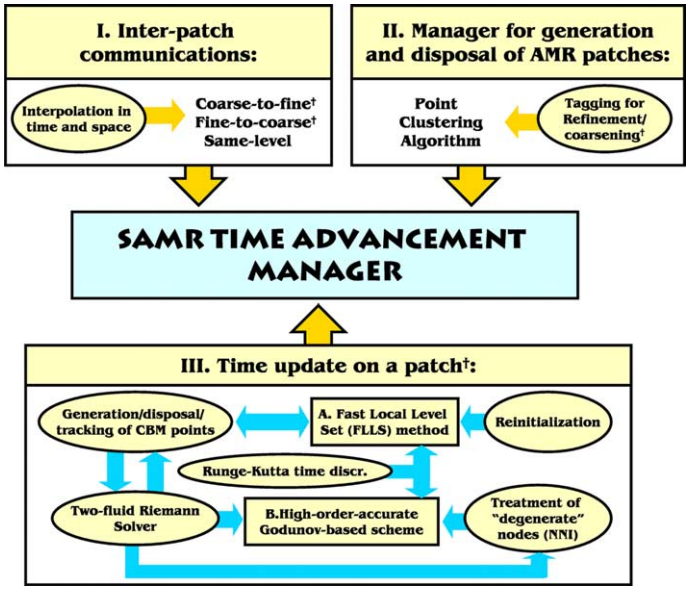

Fig. 3. Major components of the approach. Present study contributions are denoted by †.

different patches. More specifically, it is required to properly populate ghost cells around each patch, before one may proceed with its update. There are three types of inter-patch communications: same-level, coarse-to-fine and fine-to-coarse (see example in Fig. 2). The utilities for these operations are provided by SAMRAI. In the case of gas–liquid interfaces, the conservative coarse-to-fine and fine-to-coarse inter-level communications fail and must be modified. The source for these failures and the cure are described in Section 5.2.

The second important element of our approach is related to *adaptive generation/disposal of AMR patches* on different levels of a hierarchy, Fig. 3. The SAMR grid may be modified at discrete times. The finest level needs to be changed most often (patches are moved, added or deleted, if required). When the level Ω^{h_k} is changed, all finer levels $\Omega^{h_j}|_{j>k}$ are changed as well, but the coarser levels $\Omega^{h_i}|_{i<k}$ may remain fixed. The utilities for dynamic management of AMR patches, as provided by SAMRAI, require tagging criteria for refinement. In the case of compressible multiphase flows, the computational mesh needs to be refined around flow discontinuities, such as shocks, contacts and multimaterial interfaces. For interfaces, we use a distance-based criterium, and for shocks, we have a shock-detection criterium that leaves out rarefactions. Both are described in Section 5.1.

Lastly, the major components of our *patch time update strategy* are shown in Fig. 3. They include (a) a high-order-accurate Godunov-based conservative finite difference method for gas dynamics, to advance solution in the bulk fluids; (b) level-set-based capturing of interfaces, supplemented by localization (FLLS) and re-initialization algorithms, for computational efficiency and accuracy, respectively; and (c) the "characteristics-based matching" (CBM) for coupling solutions across the interface. The key features of the CBM are (i) a Riemann-solver-based coupling, which is essential for robustness in the case of high-AIM (i.e., gas–liquid) interfaces, and for accuracy in the case of very strong shock waves; and (ii) elimination of the need for ghost fields and corresponding ghost cells, which we found to be necessary for compatibility with AMR. The CBM is based on the generation/tracking/disposal of the subcell-interface-markers (denoted as CBM points), which exist on each patch only during one time step Δt_k. Applying the two-fluid pseudo-multidimensional Riemann solver at CBM points, the wave structure and gas dynamics solutions at the interface are computed and applied for direct modification of numerical fluxes in the Eulerian cells near the interface, using the subcell position of the interface and a flux inter-/extrapolation algorithm. The concept of subcell markers and two-fluid Riemann solutions are borrowed from front-tracking methods. On account of the semi-Lagrangian nature of our approach, the natural-neighbor interpolation (NNI) procedure is introduced, in order to correct numerical solutions at Eulerian computational cells, which change the fluid/(the sign of the level set), and denoted here as "degenerate" cells. The modification of numerical fluxes and treatment of "degenerate" cells are the substitutes for GFM's ghost fields/cells and related to them PDE- or FM-based extrapolation techniques. Algorithmic details of our patch time update approach are given in Section 6.

5. Grid refinement and inter-patch communications

5.1. Tagging criteria for refinement

To tag computational cells for refinement, we use the following two methods:

1. *Interface refinement*. All computational nodes of the kth AMR level are tagged for refinement near an interface, within a tube $\mathfrak{T}^R$ of the radius (ϱh_k), i.e., if $|\varphi_{(i,j)}| \leqslant (\varrho h_k)$. The radius ϱ is set to $\max(2s, \varsigma)$, where s is the stencil of the spatial discretization schemes for gas dynamics and level set equations, and ς is the radius of the level set localization (see Section 6.2). An example of a mesh refined near $\varphi \sim 0$ is shown in Fig. 1(b).

2. *Shock and contact discontinuity detection*. To determine whether the cell $_{(i,j)}$ should be refined due to the presence of a shock or contact discontinuity, we check the neighborhood of the cell in all gridline directions, defined by the following unit vectors:

$$\begin{aligned} &\textit{Orthogonal}: \quad \vec{\xi}_0 = (1,0), \qquad \vec{\xi}_1 = (0,1), \\ &\textit{Diagonal}: \quad \vec{\xi}_2 = \left(\tfrac{1}{\sqrt{2}}, \tfrac{1}{\sqrt{2}}\right), \quad \vec{\xi}_3 = \left(\tfrac{1}{\sqrt{2}}, -\tfrac{1}{\sqrt{2}}\right). \end{aligned} \tag{5}$$

As an example, we will describe the treatment in the $\vec{\xi}_0$-direction. First, compute the differences of the tested variable Φ (here, P and ρ) as

$$\Delta_0\Phi = \max(|\Phi_{(i,j)} - \Phi_{(i-1,j)}|, |\Phi_{(i,j)} - \Phi_{(i+1,j)}|),$$
$$\Delta_1\Phi = \Phi_{(i+1,j)} - \Phi_{(i-1,j)},$$
$$\Delta_2\Phi = \Phi_{(i+2,j)} - \Phi_{(i-2,j)}.$$

Then, the cell is tagged for refinement if the following criterion is satisfied:

$$\underline{\text{Shock onset detection}} \qquad \underline{\text{Gradient detection}}$$
$$\left\{ \begin{array}{ccc} \alpha|\Delta_2\Phi| & \leqslant & |\Delta_1\Phi| \\ & \text{or} & \\ \Delta_1\Phi\Delta_2\Phi & < & 0 \end{array} \right\} \quad \text{and} \qquad \Delta_0\Phi > \epsilon\Delta x. \tag{6}$$

There are two adjustable parameters in this approach: *gradient-tolerance* (ϵ) and *shock-onset* (α). The shock onset detection is designed to filter-out smooth rarefactions, from discontinuous shocks and contacts. In all computations presented here, we use the fixed value of the shock-onset $\alpha = 0.9$. The gradient-tolerance ϵ is adjusted before each calculation, in order to be able to "capture" jumps of densities and pressures, typical for the particular problem and grid resolution h_0. An example of the SAMR mesh near a shock is shown in Fig. 1(b).

After all cells on a particular level are tagged, they are collected into rectangular refinement patches using the point clustering algorithm developed by Berger and Rigoutsos [7].

5.2. Interlevel prolongation/restriction

Following [5], after every fine grid update matching coarse grid updates (e.g., update 5 for level 1 and updates 4, 7 for level 2 of Fig. 2), the coarse grid values are replaced using a *conservative average* of values from the finer grid cells, overlaying the coarse cell:

$$\mathbf{U}^{\text{coarse}}_{(i_c,j_c)} \leftarrow \frac{1}{N}\sum_{f=0}^{N} \mathbf{U}^{\text{fine}}_{(i_f,j_f)}, \tag{7}$$

where N is the number of finer cells overlaying the coarse cell.

In the presence of a gas–liquid interface, this restriction operator fails. An example of such a failure is illustrated by Fig. 4. Parameters of the EOSs are

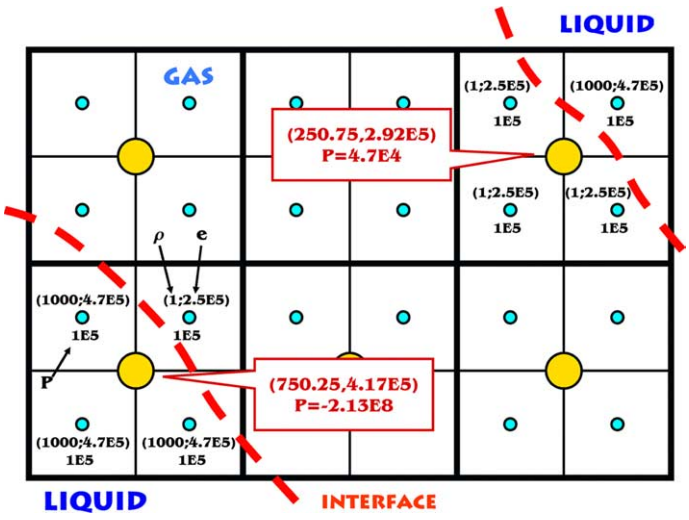

Fig. 4. On conservative fine-to-coarse grid interpolation for a gas–liquid interface problem.

$$\begin{bmatrix}\gamma\\ P_0\end{bmatrix}=\begin{bmatrix}2.8\\ 8.5\times 10^8\end{bmatrix}\quad\text{and}\quad\begin{bmatrix}1.4\\ 0\end{bmatrix}$$

for liquid and gas, respectively. The pressure field is uniform, $P=10^5$ Pa. The values of conservative variables on the fine grid for fluids at rest are shown in Fig. 4. Conservative averages on the coarse grid are computed by Eq. (7) and shown in the callouts. Using the corresponding equations of state, the pressures in coarse cells, computed from these conservative averages, can be seen to be significantly different from the uniform value of $P=10^5$ Pa. Moreover, for a liquid cell, the pressure is close to a critical value, $-P_0$, below which the sound speed $c_s=\sqrt{\frac{P\gamma+P_0}{\rho}}$ of the stiffened EOS of state (4) becomes complex. This causes stability problems during the next coarse cell update.

To eliminate this problem, we relieve the conservation constraint near the interface. More specifically, we first use bi-cubic spline interpolation [43] for a fine-to-coarse restriction of the level set function. Then, each coarse cell overlayed by fine cells is treated in the following way. If the signs of the level set functions in all overlaying fine cells coincide with the sign of the coarse cell level set, we use the conservative average, Eq. (7). Otherwise, we form an irregular data set from the fine cells which are "co-signed" with the coarse cell level set, and then use the natural neighbor interpolation (NNI) to compute the vector of conservative variables in the coarse cell.

Natural neighbor interpolation is based on the Thiessen polygon network (TPN) of the scatter point set. The TPN can be constructed from the Delauney triangulation which is a triangulated irregular network (TIN) module (used for surface modeling) that has been constructed so that the Delauney criterion has been satisfied. The Delauney criterion ensures that no vertex lies within the interior of any of the circumcircles of the triangles in the network. There is one Thiessen polygon in the network for each scatter point. The polygon encloses the area that is closer to the enclosed scatter point than any other scatter point. The polygons in the interior of the scatter point set are closed polygons and the polygons on the convex hull of the set are open polygons. Each Thiessen polygon is constructed using the circumcircles of the triangles resulting from a Delauney triangulation of the scatter points. The vertices of the Thiessen polygons correspond to the centroids of the circumcircles of the triangles. More about NNI and its implementation in two dimensions can be found in [47,54].

The above-discussed issues with restriction operator arise also for the coarse-to-fine prolongation operator, which are addressed similarly. First, we use bi-cubic spline interpolation [43] for a coarse-to-fine prolongation of the level set function. Then, we compose the dataset for bi-linear interpolation, using underlaying coarse cells. Next, we check for the proximity of the interface, using the sign of the level set function, and proceed with bi-linear interpolation of conservative variables $\mathbf{U}$ if the fine cell is far from the interface. Otherwise, we switch to NNI, discarding the coarse cells from the opposite fluid.

Finally, we would like to note that, since we use a three-level RK time discretization, the Berger and Colella's [(coarse) ← (fine)] flux synchronization procedure [5] cannot be applied. Thus, the communications of the fine and coarse levels rely entirely on the above-discussed space–time interlevel interpolation procedures.

6. Time update on a patch

6.1. Gas dynamics solver

To advance solution for gas dynamics in the bulk fluid at each patch of the AMR hierarchy, we employ a conservative finite difference approach, based on the *third-order-accurate Runge–Kutta* (RK_3) *total variation diminishing* (TVD) time discretization procedure [48].

The RK_3-TVD is a three-step procedure:

$$\begin{aligned}
&RK\ level\ 1: \mathbf{U}^{(1)}_{(i,j)}=\mathbf{U}^{(n)}_{(i,j)}+\Delta t_k\cdot\mathbb{S}^{(n)}_{(i,j)},\\
&RK\ level\ 2: \mathbf{U}^{(2)}_{(i,j)}=\frac{3}{4}\mathbf{U}^{(n)}_{(i,j)}+\frac{1}{4}\left(\mathbf{U}^{(1)}_{(i,j)}+\Delta t_k\cdot\mathbb{S}^{(1)}_{(i,j)}\right),\\
&RK\ level\ 3: \mathbf{U}^{(n+1)}_{(i,j)}=\frac{1}{3}\mathbf{U}^{(n)}_{(i,j)}+\frac{2}{3}\left(\mathbf{U}^{(2)}_{(i,j)}+\Delta t_k\cdot\mathbb{S}^{(2)}_{(i,j)}\right),
\end{aligned}\tag{8}$$

where Δt_k is a time step on the kth AMR level; and $^{(n)}$, $^{(1)}$, $^{(2)}$ and $^{(n+1)}$ denote the "old", "RK-intermediate" and "new" levels of time discretization, respectively.

Fig. 5. Level set function for numerical example III (see Section 7.3). Global level set w/o re-initialization (top) vs. fast local level set (bottom).

The spatial discretization operator at the mth RK level can be expressed as

$$\mathbb{S}^{(m)}_{(i,j)} = -\frac{\mathbb{F}^{(m)}_{(i+\frac{1}{2},j)} - \mathbb{F}^{(m)}_{(i-\frac{1}{2},j)}}{\Delta x_k} - \frac{\mathbb{G}^{(m)}_{(i,j+\frac{1}{2})} - \mathbb{G}^{(m)}_{(i,j-\frac{1}{2})}}{\Delta y_k}, \tag{9}$$

where $\mathbb{F}_{(i\pm\frac{1}{2},j)}$ and $\mathbb{G}_{(i,j\pm\frac{1}{2})}$ are the numerical fluxes at the edges of the computational cell $_{(i,j)}$. We employ *high-order-accurate polynomial reconstruction techniques* for discretization in space. The majority of simulations presented here were performed using the third-order-accurate MUSCL_3 scheme [53], limited by MinMod or van Albada's limiters [52]. In a few cases, we have employed the Nth-order-accurate WENO_N schemes ($N = 5$, 7 and 9) [4,25]. All schemes are supplemented by monotonicity-preserving (MP) procedures, described by Balsara and Shu [4].

As our base *flux discretization method*, we use the local Lax Friedrichs (LLF). However, our method is compatible with other flux treatments, as demonstrated in Section 7.1 for Godunov's exact Riemann-solver-based scheme (G), and Shu and Osher's (SO) [48] approximate Riemann solver. Schemes will be denoted by $\mathfrak{F}/\wp$, where $\mathfrak{F}$ = LLF, G or SO and $\wp = \text{MP-MUSCL}_3^{\text{Limiter}}$ or MP-WENO_N.

6.2. Fast local level set

The level set equation (3) is discretized using the third-order RK_3-TVD scheme in time and the seventh-order-accurate linear WENO scheme (LWENO_7) [37,38] in space. Following the fast local level set (FLLS) algorithm, as introduced by Peng et al. in [42], the level set equation is advanced within a tube $\mathfrak{T} = (\mathbf{x} : |\varphi(\mathbf{x})| < \varsigma\Delta h)$ of radius $\varsigma\Delta h$. The speed function vector $\mathbf{w}$ is based on material velocity,[6] $\mathbf{w} = \mathbf{u}$, but modified as $\mathbf{w}^{(\text{mod})} = C(\varphi)\,\mathbf{w}$, using the following cut-off function:

$$C(\varphi) = \begin{cases} 1 & \text{if } |\varphi| \leqslant \beta\Delta h, \\ \frac{2|\varphi| + \Delta h(\varsigma - 3\beta)}{(\varsigma-\beta)^3\Delta h^3}\left(|\varphi| - \varsigma\Delta h\right)^2 & \text{if } \beta\Delta h < |\varphi| \leqslant \varsigma\Delta h, \\ 0 & \text{if } |\varphi| > \varsigma\Delta h, \end{cases} \tag{10}$$

where $\beta\Delta h$ is the radius of an "inner tube" ($\beta < \varsigma$).

Reinitialization. As discussed in [42], the level set function requires periodical re-initialization, in order to maintain it as a signed distance function. The errors in the level set accumulate because of several reasons. *First*, due to the errors in the formulation of the speed function vector. In the case of compressible flows, the material velocity $\mathbf{u}$ is discontinuous near shocks, which causes steepening/flattening of the level set and considerable mass conservation errors. *Second*, due to discretization errors. *Third*, (in the case of the FLLS), due to cutoff function Eq. (10). Flat and steep regions, shown in Fig. 5 (top), make the computations highly inaccurate.

[6] As another option, one may employ the "extension velocity" approach [46], using the Riemann-solver-based velocity at the interface. This approach was originally used by Fedkiw et al. for GFM modeling of detonation waves [11,15], and, more recently, by the present authors [35] and by Hu and Khoo [24] for gas–liquid flows. The drawback of this treatment is that it requires extrapolation.

The procedure for level set reinitialization was introduced by Sussman et al. [49,50]. The basic idea is the following. Taking $\mathrm{d}(\boldsymbol{x}) = \varphi(\boldsymbol{x})$ as initial guess, solve the following partial differential equation to steady-state in pseudo-time τ:

$$\partial_\tau d + \mathfrak{S}(d)(|\nabla d| - 1) = 0. \tag{11}$$

At steady-state, the Eikonal equation $|\nabla d| = 1$ is recovered, and the level set is reinitialized as $\varphi(\boldsymbol{x}) = d(\boldsymbol{x})$.

The speed function $\mathfrak{S}(d)$ is defined as

$$\mathfrak{S}(d) = \left\{ \begin{array}{ll} -1 & \text{if } \varphi < 0 \\ 0 & \text{if } \varphi = 0 \\ 1 & \text{if } \varphi > 0 \end{array} \right\}.$$

Peng et al. [42] use

$$\mathfrak{S}(d) = \frac{d}{\sqrt{d^2 + |Dd|^2 \Delta h^2}} \tag{12}$$

which provides a better convergence when d becomes too flat. In the case of steep d, zero-level tends to relocate during re-initialization. To prevent that, we multiply Peng's et al. speed function by the smoothed Heaviside function $\mathfrak{H}_{\Delta h}$ as:

$$\mathfrak{S}(d) = \frac{2d|\mathfrak{H}_{\Delta h}(\varphi) - \frac{1}{2}|}{\sqrt{d^2 + |Dd|^2 \Delta h^2}}, \tag{13}$$

where Dd is a discrete operator used for ∇d, and $\mathfrak{H}_{\Delta h}$ is defined as

$$\mathfrak{H}_\epsilon(\varphi) = \left\{ \begin{array}{ll} 0 & \text{if } \varphi < -\epsilon, \\ \frac{1}{2}\left[1 + \frac{\varphi}{\epsilon} + \frac{\sin\frac{\pi\varphi}{\epsilon}}{\pi}\right] & \text{if } |\varphi| \leqslant \epsilon, \\ 1 & \text{if } \varphi > \epsilon. \end{array} \right. \tag{14}$$

For discretization of the Hamiltonian $\mathfrak{S}(d)(|\nabla d| - 1)$ in Eq. (11), we use a variant of the Engquist–Osher scheme [42,46].

In practice, there is no need to solve Eq. (11) to a complete steady-state. Instead, similar to Sussman et al. [50], we apply only a few (from 2 to 5) re-initialization pseudo-time steps at the end of each physical time step of the finest AMR level, which is enough to maintain the level set as a signed distance function within a tube (as shown in Fig. 5(bottom)).

It is important to note that the level set localization tube is fully confined within the finest level, which is enforced by setting the radius of the distance-based refinement tagging as $\varrho > \varsigma$. Therefore, the level set equation (3) is solved only on the finest level of a hierarchy, while the other (coarser) levels are populated using the bi-cubic spline-based restriction (see discussion in Section 5.2). As a consequence, there are no difficulties to organize a pseudo-time loop needed for the PDE-based reinitialization, as it does not interfere with the physical time advancement algorithm of SAMR (see Fig. 2 and discussion in Section 3), and only same-level (SL) communications are required at the end of each pseudo-time step, to communicate patches on the finest AMR level.

6.3. Characteristics-based matching

6.3.1. Subcell position of an interface

At the beginning of each time step, based on the currently available level set function, we identify all intersections of the zero-level-set with grid lines. These markers (called CBM_d) are utilized as reference points for the "characteristics-based matching" of the numerical solution across the fluid–fluid interface. In two-dimensions, we employ four types of CBM_d's, corresponding to the four possible gridline directions $_{d=0,1,2,3}$, see Eq. (5). The subcell position of the interface for each CBM_d point is computed from the currently available level set function. In the present paper, we use the following algorithm for reconstruction of the subcell position of the interface in the gridline direction $_{d=0}$:

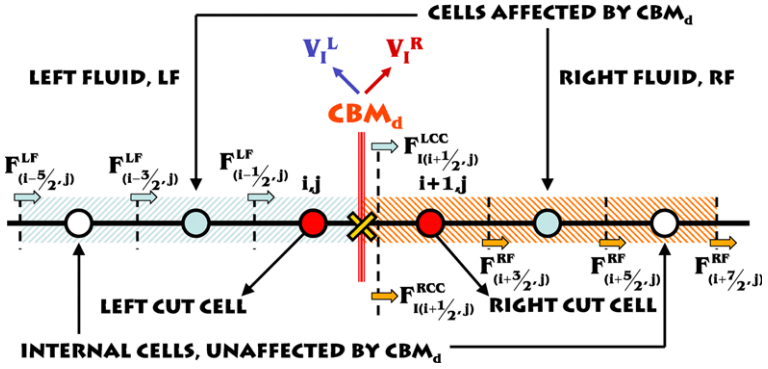

Fig. 6. On flux correction using CBM points. $s = 2$.

i. Compute an initial guess, using the following one-dimensional piecewise-linear equation:

$$\theta_{d=0} = \frac{|\varphi_{(i,j)}|}{|\varphi_{(i+1,j)}| + |\varphi_{(i,j)}|}. \tag{15}$$

ii. Construct a multidimensional bi-cubic spline [43] of the level set function

$f^{\mathrm{BiC,spline}}(\varphi_{(i,j)}, \varphi_{(i\pm1,j)}, \varphi_{(i,j\pm1)}, \mathbf{x})$.

iii. Solve iteratively the non-linear equation $f^{\mathrm{BiC,spline}}(x) = 0$ at the interval between $x_{(i,j)}$ and $x_{(i+1,j)}$, using the van Wijngaarden–Dekker–Brent method [43].

Similar procedures are applied for other gridline directions $_{d=1,2,3}$. The above-described algorithm is similar to the one described in Section 2.2 of the paper by Gibou and Fedkiw [16].

6.3.2. Two-fluid Riemann solver

At the beginning of each RK-level of time advancement, the pseudo-multidimensional two-fluid Riemann solver is applied at each CBM point in the following way. *First*, the left and right states ($\mathbf{U}_L$ and $\mathbf{U}_R$) of all CBM points are determined, using the MUSCL$_3$-based interpolation of flow variables along the corresponding gridlines and taking into account the subcell position of the interface θ_d. *Second*, the exact two-fluid Riemann solver is applied to determine the new state of the interface,

$$\mathbf{V}_I^{L,R} = \begin{bmatrix} P \\ \rho^L, \rho^R \\ V_\xi \end{bmatrix}^{(RS)}_{CBM_p},$$

where V_ξ is the normal component of the velocity, corresponding to the direction $_{(d)}$ of the CBM point.[7] The tangential components of the velocity vector are computed using the V_ξ-based upwinding [10].

6.3.3. Numerical fluxes

The coupling is performed by direct modification of the numerical fluxes at all computational cells near the interface, using the CBM points. For example, for fluxes in x-direction, this is done as follows.

(A) *Initial guess.* As a first step, we run over all $_{(i,j)}$ cells and compute numerical fluxes $\mathbb{F}_{(i+\frac{1}{2},j)}$ assuming that the cells are fully occupied by the "positive" fluid, if $\varphi_{(i,j)} \geqslant 0$, or by the "negative" fluid, if $\varphi_{(i,j)} < 0$. Simultaneously, we tag all cells into three groups:

[7] Preferably, one would like to solve Riemann problem in the normal-to-interface direction. Our experiments have shown though that this causes stability problems, most probably due to inconsistency with 'along-gridline inter-/extrapolation' used here, and requiring fully multidimensional treatment. This issue will be addressed in the future. As a consequence, unlike the original GFM method [13], our method is expected to be diffusive for a 'pure shear wave' test, a deficiency that is moderated by the AMR capability.

1. Each $\mathrm{CBM}_{d=0}$ point is bracketed by two cells, located on opposite sides of the interface. These cells are denoted as the *"Left and Right cut cells"*, LCC and RCC, see Fig. 6.
2. The cells, which are within the stencil (stencil s is 2, 3, 4 and 5 for MUSCL_3, WENO_5, WENO_7 and WENO_9 schemes, correspondingly) near CBM_0 points and which are not "cut cells", are tagged as "*affected by CBM$_d$*", Fig. 6.
3. All the rest cells are marked as "*internal*, *unaffected by CBM$_d$*".

(B) *Cells affected by CBM$_d$.* Next, we run over all "affected by CBM_d" cells $_{(i,j)}$ and re-compute numerical fluxes $\mathbb{F}_{(i+\frac{1}{2},j)}$. This involves $2s$ neighbor cells: $[_{(i-s+1,j)}\cdots_{(i+s,j)}]$. First, we create the matrix $\mathbb{V}$ of the size $4\times(2s-1)$, which contains the vectors of primitive variables as its rows, $\mathbf{V}_{(1-s),\ldots,(s)}$, where $\mathbf{V}=(P,\rho,u,v)^{\mathrm{T}}$. Then, we populate the matrix $\mathbb{V}$ in the following way. Similarly to the GFM method [13], the pressures and velocities are taken from the corresponding fluid cells, $V_{1_{(r+s)}}=P_{(i+r,j)}, V_{3_{(r+s)}}=u_{(i+r,j)}$ and $V_{4_{(r+s)}}=v_{(i+r,j)}$, $r=1-s,\ldots,s$. As initial guess, the density is taken from the corresponding fluid cells, $V_{2_{(r+s)}}=\rho_{(i+r,j)}$. Next, if there is a CBM_0 point to the left of the cell $_{(i,j)}$ and within the stencil range, the elements of the matrix $\mathbb{V}$ corresponding to the cells located to the left of this CBM point are populated using the left entropy state of the CBM_0 and the pressure taken from the corresponding fluid cells, i.e., $V_{2_{(r+s)}}=\rho(S_{\mathrm{I}}^{\mathrm{L}},P_{(i+r,j)})$. Similarly, if there is a CBM point to the right of the cell $_{(i,j)}$ and within a stencil range, the density elements of the matrix $\mathbb{V}$ are computed as $V_{2_{(r+s)}}=\rho(S_{\mathrm{I}}^{\mathrm{R}},P_{(i+r,j)})$. Once the matrix $\mathbb{V}$ is fully filled in, it is used to compute numerical fluxes $\mathbb{F}_{(i+\frac{1}{2},j)}$. Importantly, with this treatment, "thin" interface structures, developing in under-resolved flows, are not "transparent".

(C) *Cut cells.* All cut cells are treated non-conservatively, i.e., we use two sets of numerical fluxes at the interface – one for the update of the LCC, and another for the update of the RCC, denoted as $\mathbb{F}^{\mathrm{LCC}}_{\mathrm{I}(i+\frac{1}{2},j)}$ and $\mathbb{F}^{\mathrm{RCC}}_{\mathrm{I}(i+\frac{1}{2},j)}$ in Fig. 6. "Cut cell" fluxes are computed using Riemann-based solutions from the CBM point, $\tilde{\mathbb{F}}^{\mathrm{LCC}}_{\mathrm{I}(i+\frac{1}{2},j)}=\vec{\mathscr{F}}(\mathbf{V}_{\mathrm{I}}^{\mathrm{L}})$ and $\tilde{\mathbb{F}}^{\mathrm{RCC}}_{\mathrm{I}(i+\frac{1}{2},j)}=\vec{\mathscr{F}}(\mathbf{V}_{\mathrm{I}}^{\mathrm{R}})$, and then corrected to take into account the subcell position of the interface, using the following linear inter-/extrapolations of fluxes:

$$\mathbb{F}^{\mathrm{LCC}}_{\mathrm{I}(i+\frac{1}{2},j)}=\mathbb{F}^{\mathrm{LF}}_{(i-\frac{1}{2},j)}+\frac{\tilde{\mathbb{F}}^{\mathrm{LCC}}_{\mathrm{I}(i+\frac{1}{2},j)}-\mathbb{F}^{\mathrm{LF}}_{(i-\frac{1}{2},j)}}{\frac{1}{2}+\theta_{\mathrm{CBM}_d}} \tag{16}$$

and

$$\mathbb{F}^{\mathrm{RCC}}_{\mathrm{I}(i+\frac{1}{2},j)}=\mathbb{F}^{\mathrm{RF}}_{(i+\frac{3}{2},j)}+\frac{\tilde{\mathbb{F}}^{\mathrm{RCC}}_{\mathrm{I}(i+\frac{1}{2},j)}-\mathbb{F}^{\mathrm{RF}}_{(i+\frac{3}{2},j)}}{\frac{3}{2}-\theta_{\mathrm{CBM}_d}}. \tag{17}$$

Since the cut cell fluxes are Riemann-solver based, the loss of conservation does not affect the ability to describe strong shock waves, which is demonstrated by numerical examples in Section 7.

6.3.4. Treatment of "degenerate" points

Since the whole algorithm is Eulerian-grid based, there are always a number of cells which change their "signs" (fluid) due to motion of the level set at a particular time step. We denote these cells as "*degenerate*". The solution at this time step for "degenerate" cells is not appropriate, since it was obtained assuming that these cells are occupied by another fluid. In the GFM [13], this situation is taken care of by (i) keeping two sets of flow variables for each modeled fluid; and (ii) using extrapolation techniques to populate ghost nodes. As a ghost node is uncovered – the extrapolated solution is always available. Since we do not use ghost fields here, the "degenerate" cells are treated in the following way.

i. Scan the neighborhood of each "degenerate" cell and identify all neighbor cells which belong to the same fluid and "non-degenerate".
ii. All neighbor CBM points (both orthogonal and diagonal, see Eq. (5)) are identified and their positions are advanced in time using the available velocity from the CBM Riemann solver step.

iii. All these nodes compose an irregular data set, which is used for interpolation of flow variables into the "degenerate" cell. For this purpose, we use the *natural neighbor interpolation* (NNI) technique (see discussion in Section 5.2).

7. Numerical examples

7.1. Example I: 1D shock tube problems

Gas–gas. As our first numerical test we consider a strong-shock gas–gas interface problem introduced by Abgrall and Karni in [2]. The shock tube is 1-unit-long and the interface is located at $x = 0.5$. The initial left and right states, expressed in dimensionless units [2], are

$$\begin{aligned}(\rho, u, p, \gamma)_{\mathrm{L}} &= (1, 0, 500, 1.6)\\ (\rho, u, p, \gamma)_{\mathrm{R}} &= (1, 0, 0.2, 1.4)\end{aligned} \tag{18}$$

Calculations are performed on a sequence of uniform computational grids with resolution of $\Delta x = \frac{1}{800}, \frac{1}{1600}$ and $\frac{1}{3200}$. The CFL number was set to 0.8.

The analytical solution for this problem consists of the strong $M_{\mathrm{sh}} = 31$ transmitted shock, moving to the right, and the rarefaction wave, moving to the left, as shown in Fig. 7. The interface is sharply resolved

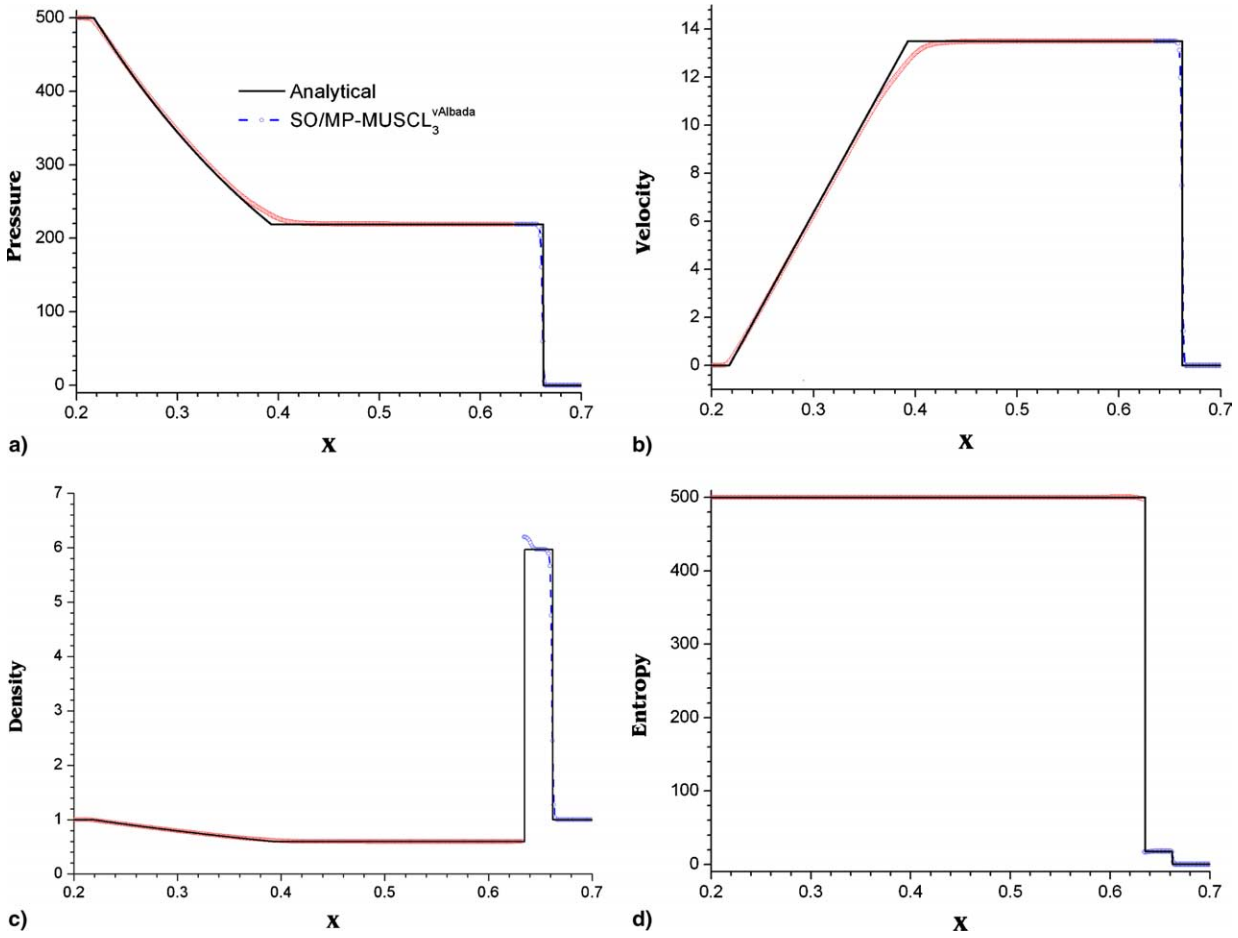

Fig. 7. Numerical solutions for the test of gas–gas interface under strong shock wave. $\Delta x = \frac{1}{800}$. $t = 0.01$.

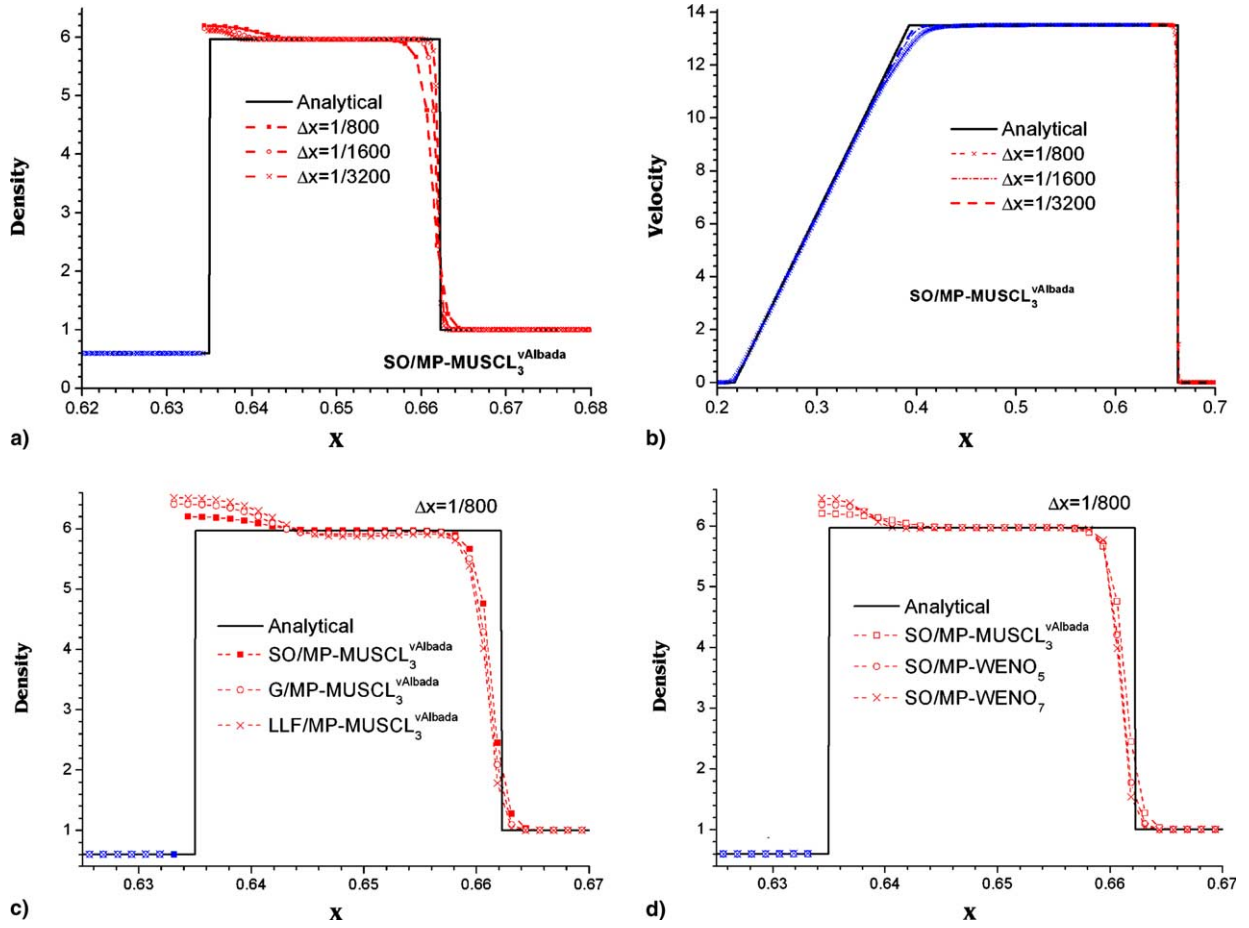

Fig. 8. Effect of (a,b) grid resolution; (c) flux treatment and (d) spatial discretization for the test of gas–gas interface under strong shock wave. $t = 0.01$.

without pressure oscillations. The underheating errors[8] are small, and tend to reduce with a better grid resolution, Fig. 8(a). As shown in Figs.7(a) and (b), there are small errors in the right tail of the rarefaction wave, which become smaller with grid refinement, Fig. 8(b). These errors exist for all methods [2,24] and, as noted by Abgrall and Karni in [2], they are related to the stiffness of the problem.

Figs. 8(c) and (d) demonstrate the performance of the CBM with different flux treatment and spatial discretization schemes. Among three tested flux treatments (Shu and Osher, SO [48]; Godunov, G; and local Lax Friedrichs, LLF), the SO scheme performs best, in terms of underheating errors and description of the interface position, Fig. 8(c). Finally, the effect of the spatial discretization order is presented in Fig. 8(d). The MUSCL_3 scheme features the smallest overheating errors. Probably, this is related to the compatibility between the subcell interpolation to the CBM points and the interpolation to the edges of the computational cells, since both use $\mathrm{MUSCL}_3^{\mathrm{vAlbada}}$.

Gas–liquid. As our next numerical example, we consider a gas–liquid (stiff) shock-tube problem introduced by Saurel and Abgrall [45]. The conditions of this test involve an interface located at $x = 0.7$ m, and initial left and right states specified as

[8] The origins of over/underheating errors are discussed by Noh [34] and more recently by Menikoff [32] and Fedkiw et al. [12].

$$(\rho, u, p, \gamma, \Pi)_{\mathrm{L}} = (1000, 0, 10^9, 4.4, 6 \times 10^8)$$
$$(\rho, u, p, \gamma, \Pi)_{\mathrm{R}} = (50, 0, 10^5, 1.4, 0) \tag{19}$$

Simulations are performed using a uniform grid of $\Delta x = \frac{1}{800}$ m and CFL = 0.8.

Computational results for time 0.24 ms are compared to the analytical solution in Fig. 9. The gas–liquid interface is resolved within one grid node, while the position of the transmitted shock is captured within two grid nodes. No overheating errors are observed at the gas–liquid interface. There are insignificant over-/undershoots of velocity and pressure at the trailing edge of the rarefaction wave. Our method is first-order accurate due to the effect of limiters in the presence of flow discontinuities. Grid convergence of $\mathscr{L}_1$-norm of error for pressure, density, velocity and entropy (S) is presented in Table 1, for two spatial discretization schemes.

7.2. Example II: single-mode Richtmyer–Meshkov instability in 2D

Formulation. As our first two-dimensional test, we consider the single-mode Richtmyer–Meshkov instability (RMI). We use a computational domain of size 4×1, with periodical boundary conditions (BCs) in y-direction and non-reflection BCs in x-direction. The initial position of the interface is generated by

$$\varphi(\mathbf{x}) = x_{\mathrm{I}} - x - \epsilon \sin\left(2\pi\left(\frac{y}{\lambda} + \frac{1}{4}\right)\right),$$

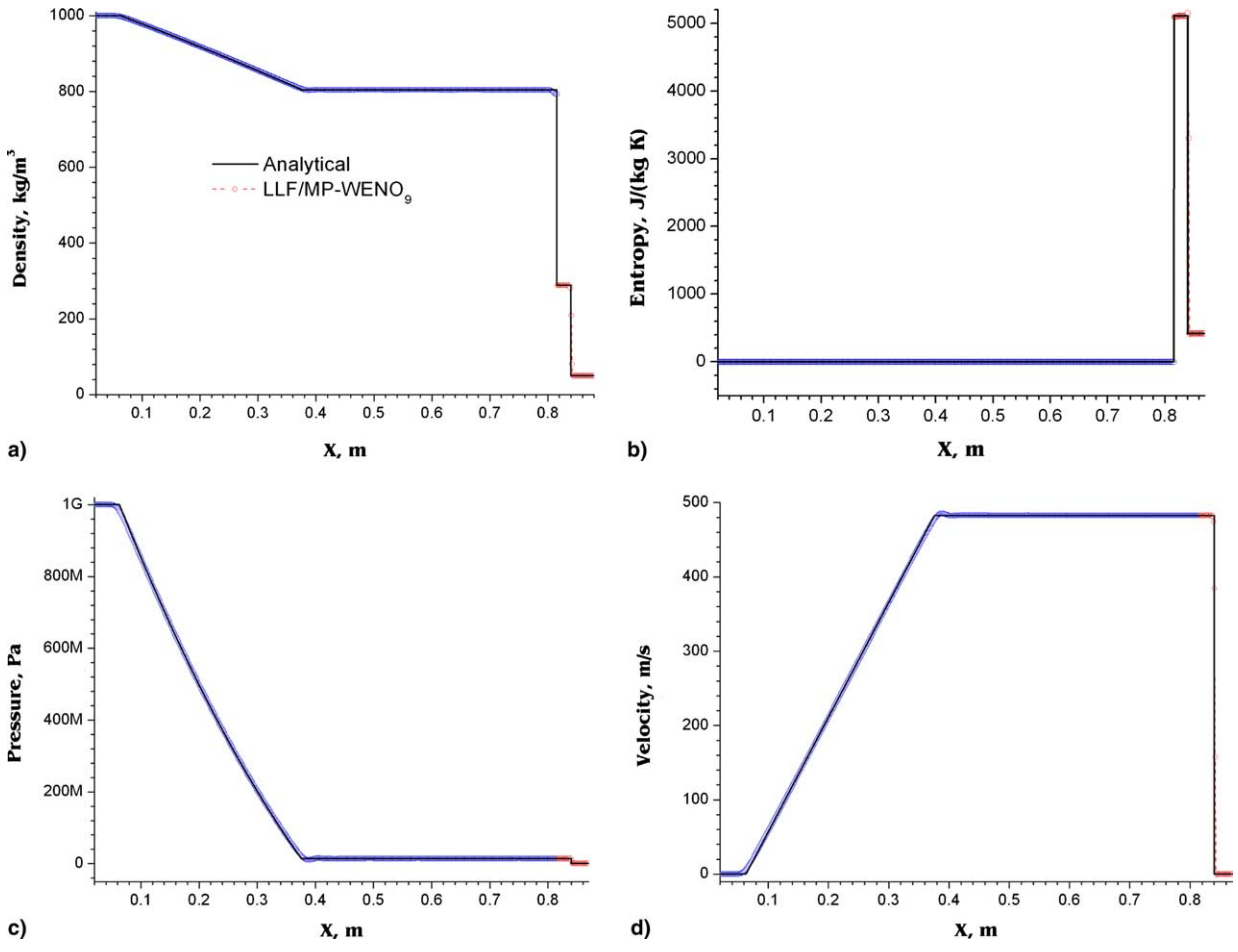

Fig. 9. Numerical solutions for the air–water shock tube problem. $t = 0.24$ ms.

Table 1
On convergence for a gas–liquid (stiff) shock-tube problem

Resolution, m	$\mathscr{L}_1(P)$	$\mathscr{L}_1(\rho)$	$\mathscr{L}_1(u)$	$\mathscr{L}_1(S)$
LLF/MP-MUSCL$_3^{vAlbada}$				
$\frac{1}{200}$	5.38×10^{-3}	3.904×10^{-3}	6.333×10^{-3}	4.907×10^{-3}
Rate	1.055	1.006	1.008	1.023
$\frac{1}{400}$	2.589×10^{-3}	1.944×10^{-3}	3.148×10^{-3}	2.415×10^{-3}
Rate	1.013	1.058	0.994	0.914
$\frac{1}{800}$	1.283×10^{-3}	9.338×10^{-4}	1.58×10^{-3}	1.282×10^{-3}
Rate	1.034	0.93	0.948	1.017
$\frac{1}{1600}$	6.269×10^{-4}	4.9×10^{-4}	8.189×10^{-4}	6.338×10^{-4}
Rate	0.969	0.945	0.888	0.839
$\frac{1}{3200}$	3.202×10^{-4}	2.545×10^{-4}	4.424×10^{-4}	3.544×10^{-4}
LLF/MP-WENO$_9$				
$\frac{1}{200}$	5.265×10^{-3}	2.943×10^{-3}	5.282×10^{-3}	3.33×10^{-3}
Rate	0.982	0.993	0.956	0.9
$\frac{1}{400}$	2.666×10^{-3}	1.479×10^{-3}	2.722×10^{-3}	1.784×10^{-3}
Rate	0.972	0.991	0.97	0.788
$\frac{1}{800}$	1.359×10^{-3}	7.442×10^{-4}	1.39×10^{-3}	1.033×10^{-3}
Rate	0.962	0.917	0.898	0.937
$\frac{1}{1600}$	6.975×10^{-4}	3.941×10^{-4}	7.46×10^{-4}	5.392×10^{-4}
Rate	0.958	0.874	0.851	0.808
$\frac{1}{3200}$	3.59×10^{-4}	2.151×10^{-4}	4.137×10^{-4}	3.08×10^{-4}

where $x_I = 2.9$ is the location of the unperturbed interface, $\epsilon = 0.1$ is the amplitude of the imposed perturbation, $\lambda = 1$ is the wavelength, respectively. The interface separates SF_6 and air, both modeled as γ-gases with $\gamma_{SF_6} = 1.093$ and $\gamma_{air} = 1.4$. The density ratio across the interface is $\frac{\rho_{SF_6}}{\rho_{air}} = 5.04$. A planar $M_{sh} = 1.24$ shock wave is placed in the air at $x_{sh} = 3.2$, with the following pre- and post-shock conditions:

$$\mathbf{V}_{\text{pre-shock}} = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix} \quad \text{and} \quad \mathbf{V}_{\text{post-shock}} = \begin{bmatrix} 1.628 \\ 1.411 \\ -0.39 \\ 0 \end{bmatrix}.$$

The Atwood number based on the pre-shock densities is $At_{\text{pre-shock}} = 0.67$. Simulations are performed on a sequence of meshes with one, two and three levels of adaptation, refined with ratio 4. The LLF/MP-WENO$_5$ scheme and CFL = 0.4 are employed.

Dynamics. The incident shock hits the interface, resulting in a transmitted shock in SF_6 and a reflected shock in air. The accelerated interface moves from the light into the heavy gas, forming a bubble of air and a spike of SF_6, see Fig. 10. Due to the initial perturbation of the interface, during the residence of the transmitted and reflected shocks in the perturbed zone, there is a misalignment of pressure and density gradients, which results in a non-zero baroclinic term $(\frac{1}{\rho^2}\nabla P \times \nabla\rho)$ and causes a generation of the vorticity and a formation of rollups, as seen in Fig. 10 for time $t > 5$. Fig. 11 presents the position history of the leading edges for the bubble $x_b(t)$ and spike $x_s(t)$, together with the thickness of the mixing layer, defined as $h_{mix} = x_s - x_b$, for three grid resolutions. The grid convergence for these most important characteristics of RMI is apparent.

7.3. Example III: weak shock in 2D multimaterial gaseous media

As our next numerical test, we consider a weak shock in air interacting with cylindrical "bubble" of Refrigerant-22 (R_{22}) in air. This problem was studied experimentally by Haas and Sturtevant [21]. They obtained Schlieren images of the flow fields, which provide a very sensitive standard for comparison.

Problem formulation. A cylindrical bubble of R_{22} is suspended in initially motionless air, under $P = 10^5$ Pa, in a computational domain of size 445×89 mm. The initial position of the 50-mm-diameter bubble is

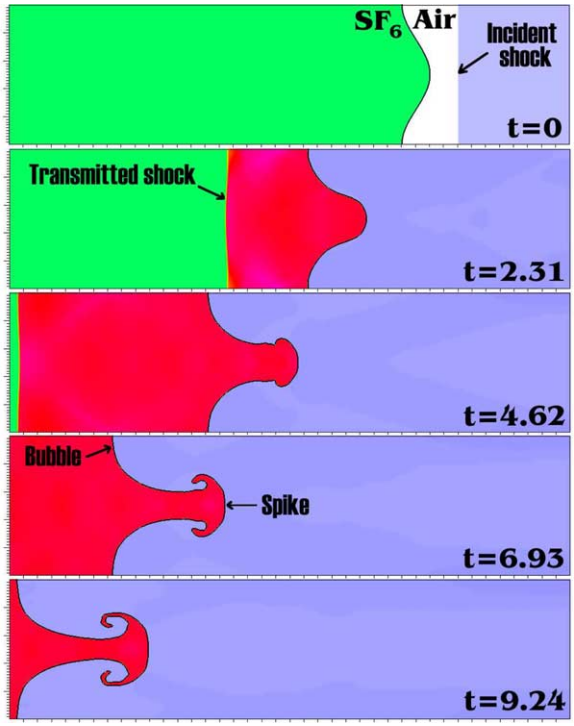

Fig. 10. Dynamics of the interface and density field for numerical example II. Two levels of adaptation with refinement ratio 4. Effective grid resolution (on the finest level) is 200 nodes per λ.

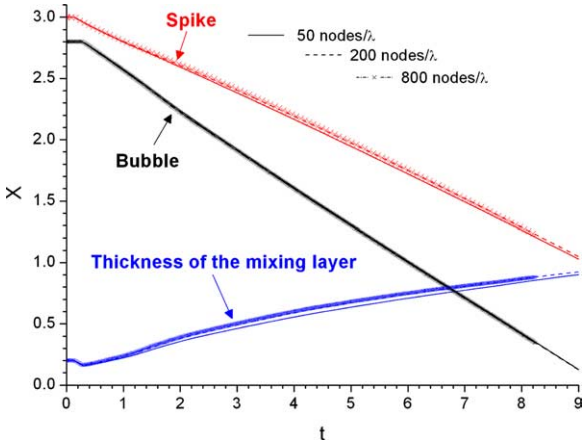

Fig. 11. Positions of the bubble and spike and thickness of the mixing layer as functions of time for numerical example II.

$\boldsymbol{x}_b = (245, 44.5)$ mm. Both fluids are modeled as γ-gases, with $\gamma = 1.4$ and 1.249 for air and R_{22}, respectively. Initial densities of air and R_{22} are 1 and 3.15385 kg/m^3. A planar $M_{sh} = 1.22$ shock wave with pre- and post-shock conditions

$$\mathbf{V}_{\text{pre-shock}} = \begin{bmatrix} 10^5 \\ 1 \\ 0 \\ 0 \end{bmatrix} \quad \text{and} \quad \mathbf{V}_{\text{post-shock}} = \begin{bmatrix} 1.57 \cdot 10^5 \\ 1.38 \\ -124.82 \\ 0 \end{bmatrix}$$

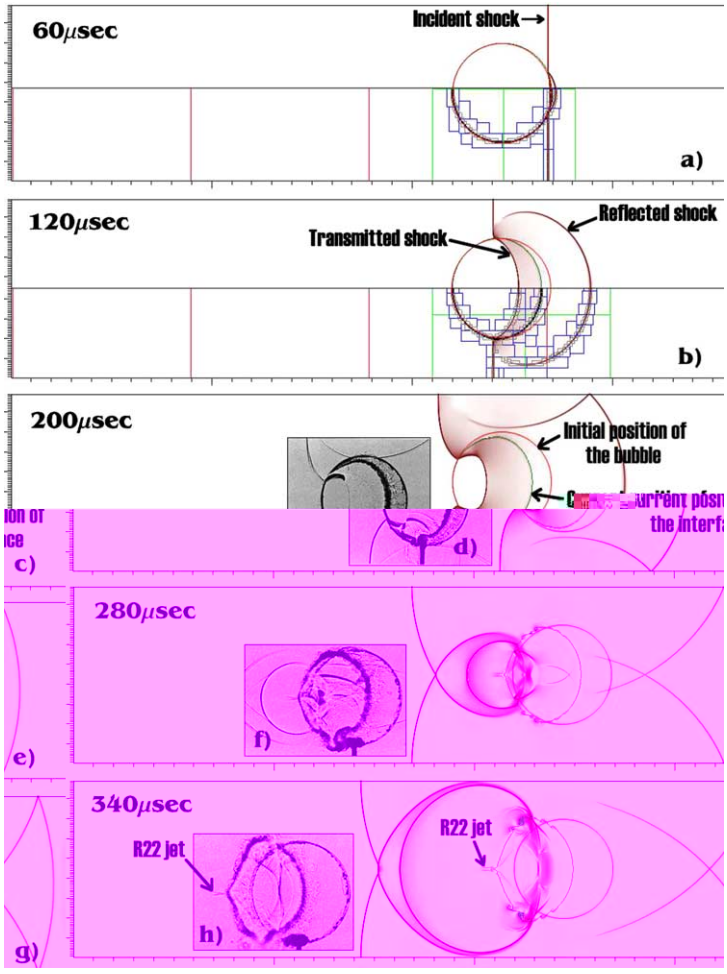

Fig. 12. Interaction of a weak air shock wave with refrigerant R22 cylinder. Numerical Schlieren images for the SAMR of three levels of adaptation with refinement ratio 4. Experimental Schlieren images from [21] are shown in (d), (f) and (h).

is placed initially at $x_{sh} = 295$ mm. BCs are periodical in the vertical direction and non-reflection for both left and right boundaries of the computational domain. Two meshes are employed – one with three and another with six levels of adaptation, using refinement ratios 4 and 2, respectively. The effective grid resolution on the finest level of both meshes was $\Delta h \approx 56$ μm, which corresponds to approximately 900 nodes per initial bubble's diameter. If this simulation was performed on a uniform mesh with the same grid resolution, it would require roughly 12.8 million nodes. With the SAMR, the effective number of nodes was around 1.2 million. Simulations are performed using the $\mathrm{LLF/MP-MUSCL}_3^{\mathrm{vAl}}$ scheme and $\mathrm{CFL} = 0.4$.

Computational results for selected moments of the simulation are shown in Figs. 12 and 13. The numerical Schlieren images are generated in the same way as described by Quirk and Karni in [41], using the following nonlinear shading function ϑ to accentuate weak flow features:

$$\vartheta = \exp\left(-\kappa \frac{|\nabla\rho|}{|\nabla\rho|_{\max}}\right). \tag{20}$$

The accentuation coefficients κ were set to 2000 and 400 for air and R_{22}, respectively. For comparison, we also included the experimental Schlieren images from [21] (Figs. 12(d), (f) and (h)), for approximately the same time moments and scales as the numerical Schlieren images.[9] As it can be seen from Fig. 12(c), (e) and (d),

[9] More detail comparisons for both R_{22} and Helium bubbles are presented in [51].

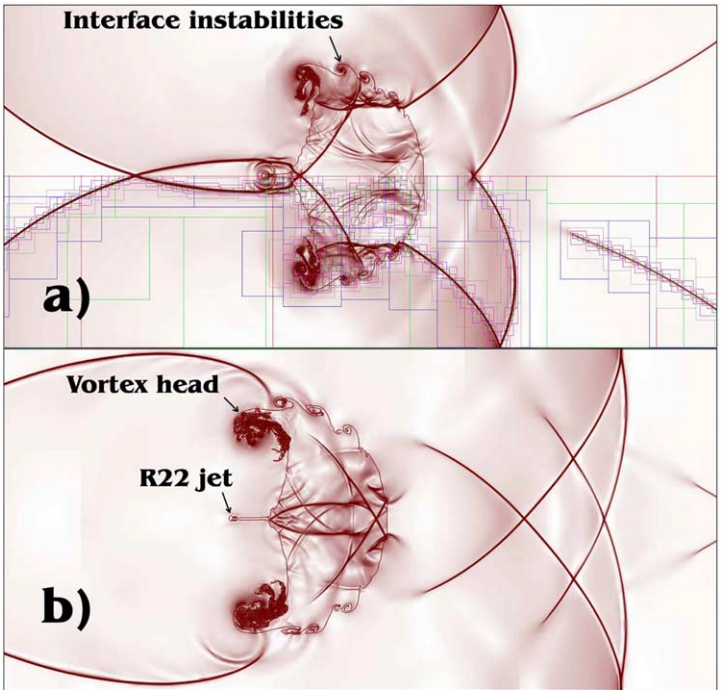

Fig. 13. Numerical Schlieren for (a) $t = 460$ μs and (b) 560 μs. Solution using six levels of adaptation with refinement ratio 2. SAMR patches are outlined in the lower half of the image for $t = 460$ μs.

the results of our simulations are in good agreement with the experimental data, resolving general shape of the bubble, complex shock refraction patterns, formation of vorticities due to Richtmyer–Meshkov (RM) and Kelvin–Helmholtz (KH) instabilities and the R_{22} jet.

Discussion. The first simulations of Haas and Sturtevant's experiments are due to Quirk and Karni [41]. More recently, these experiments were also computed by Fedkiw et al. [13] and Hu and Khoo [24] with the GFM method. To resolve vortex structures of the RM and KH instabilities at the interface, very high grid resolution is required, feasible only by AMR used here and in the paper by Quirk and Karni [41]. On the other hand, our "sharp-interface" approach is different from Quirk and Karni [41], who employed a diffusive mass-fraction-based method, limited to γ-gases and weak shocks. Our approach is capable to work for both gas–gas and gas–liquid flows, under both weak and strong shock wave conditions, which will be demonstrated in our next numerical example.

Since the level set is non-conservative approach, it is important to monitor mass conservation errors. We have found that these errors are directly related to grid resolution of interface structures. Due to RM and KH instabilities, the interface significantly deforms, sequentially breaking up into smaller and smaller pieces. Eventually, the fragments of R_{22} become under-resolved and disappear due to discretization errors and re-initialization procedure. For grid resolution of 900/Diameter, we have lost only 1.8% of mass by time 560 μs, when a significant vortex head has already been developed, Fig. 13.

7.4. Example IV: collapse of 2D gas bubble under strong shock in water

Problem formulation. A cylindrical air bubble, 6 mm in diameter, is immersed in a water pool, under the following initial conditions: $\mathbf{u} = (0,0)$ m/s, $P = 10^5$ Pa, $\rho_{\text{air}} = 1$ kg/m^3 and $\rho_{\text{water}} = 1000$ kg/m^3. The center of the bubble is located at $\mathbf{x}_{\text{b}} = (12,12)$ mm in the computational domain of size 24×24 mm. Parameters of the EOSs are

$$\begin{bmatrix} \gamma \\ \Pi \end{bmatrix} = \begin{bmatrix} 4.4 \\ 6 \times 10^8 \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} 1.4 \\ 0 \end{bmatrix}$$

for water and air, respectively. A planar incident $M_{\text{sh}} = 1.72$ shock wave with post-shock conditions

$$\mathbf{V}_{\text{post-shock}} = \begin{bmatrix} 1.9 \times 10^9 \\ 1323.65 \\ 681.58 \\ 0 \end{bmatrix}$$

is initially located 5.4 mm to the left of the air bubble center. BCs are periodical in the vertical direction and non-reflection at the left and right boundaries of the domain. Simulations were performed on a sequence of

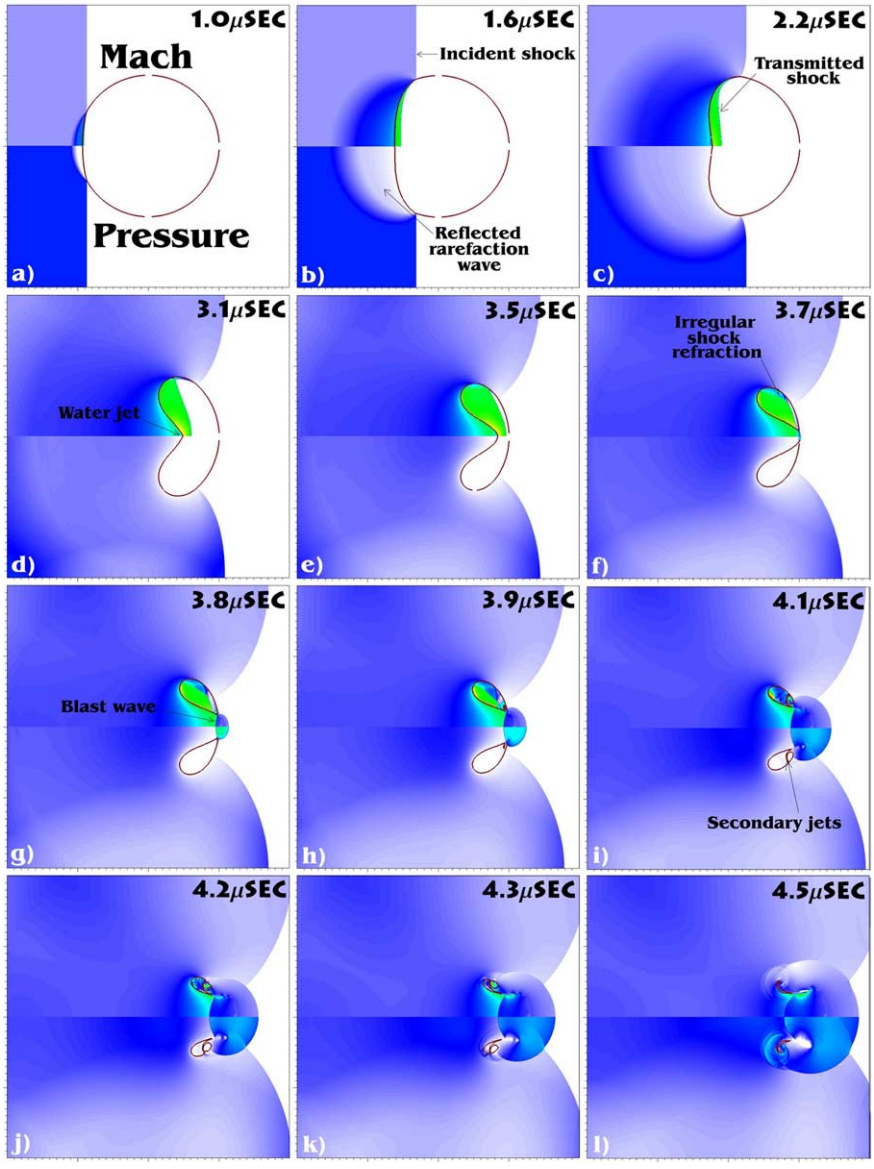

Fig. 14. Dynamics of the pressure and Mach fields for numerical example IV. Effective grid resolution is 800 nodes per initial bubble diameter.

three grids, with the effective resolution of 200, 400 and 800 computational nodes per initial bubble's diameter, corresponding to AMR with up to six levels of adaptation and refinement ratio two. The LLF/MP-WENO$_5$ scheme and CFL = 0.4 are applied.

Dynamics of the bubble collapse is shown in Fig. 14, as color-map-plots of Mach number and pressure for the highest grid resolution. Due to the large AIM at the water–air interface, the incident shock transmits a relatively weak shock into the air, producing a strong reflected rarefaction wave in the water, Fig. 14(b). The transmitted shock is only *relatively* weak. In fact, the post-shock pressure is quite high $P_{\text{post-shock}} = 2.3$ MPa, which corresponds to $M_{\text{sh}} = 4.7$. By approximately 2.3 μs, the air bubble becomes involuted, with a distinct water jet formed at the centerline, Fig. 14(d). At approximately 3.69 μs, the water jet hits the other side of the bubble with a velocity of 2.85 km/s, cutting the bubble in half. Upon impact, an intense blast wave, with maximum pressure of 10.1 GPa, is generated, as shown in Fig. 14(g). Caused by the blast wave, secondary jets penetrate into the smaller bubbles (see Fig. 14(i) and, more detailed, in Fig. 15), and finally cut the initial bubble into four pieces, Figs. 14(j) and 16. A close-look image of the Mach field, together with the mesh, is shown in Fig. 17, at the moment of the blast wave generation. As one can see, the irregular refraction inside the gas is well resolved. The time history of the dimensionless volume of gas (computed using dimensionless length units $x^\star = \frac{x}{D_b}$, where $D_b = 6$ mm) is presented in Fig. 18. The gas volume reaches its minimum at approximately 4.5 μs, starting to slowly grow after that (rebound). The maximum temperature of the gas at the moment of minimum volume is ≈25,000 K. Since we have not modeled heat transfer and real gas effects here, the temperature history can be regarded as qualitative. Nevertheless, the observed intense heating is consistent with the observation of luminescence in experiments by Bourne and Field [8], performed for similar conditions.

Grid convergence. The importance of the grid resolution and AMR capabilities is demonstrated in Fig. 15, showing zero-level-set, Mach and pressure fields at the late stage of the bubble collapse. As one can see, the details of the secondary jets can be well resolved only with the finest grid resolution of 800 nodes/initial bubble diameter.

Fig. 19 shows time histories of the mass conservation errors for three grid resolutions. For very poor grid of 50 nodes/Diameter (not shown in Fig. 19), the bubble disappears completely by time 4 μs. With the better grid resolutions of 200, 400 and 800 nodes/Diameter, 20%, 14% and 8% of the gas mass are lost, correspondingly.

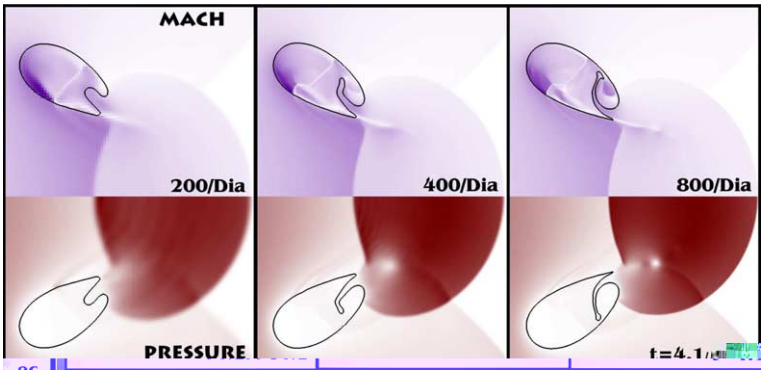

Fig. 15. Effect of the grid resolution on capturing of the secondary jets. Mach (top) and pressure (bottom) fields for $t = 4.1$ μs.

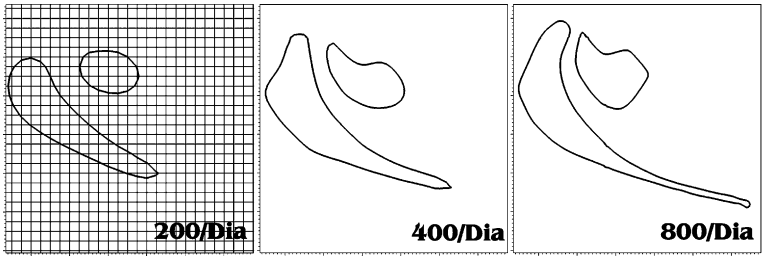

Fig. 16. Effect of the grid resolution on the final stage of bubble collapse. Zero level set for $t = 4.4$ μs.

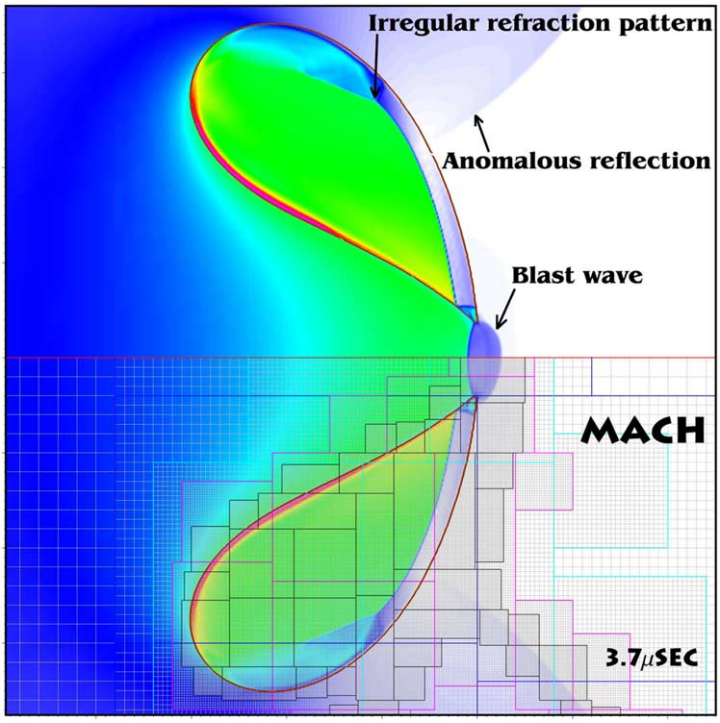

Fig. 17. Mach field and structure of the adaptive mesh for numerical example IV. Six levels of adaptation with refinement ratio 2. Effective grid resolution (on the finest level) is 800 nodes per initial bubble diameter.

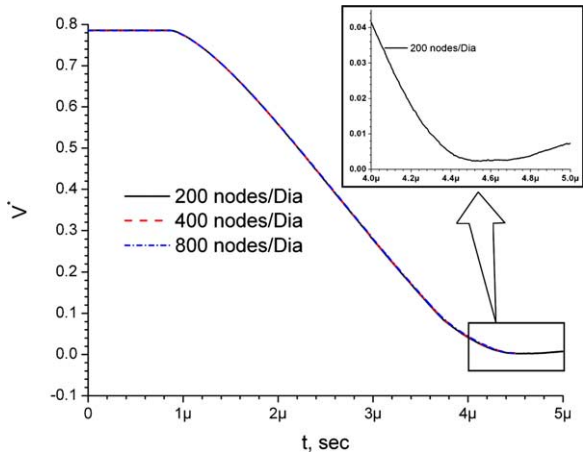

Fig. 18. Dynamics of gas volume.

Thus, there is a clear tendency to reduce mass conservation errors with grid refinement. The reason for mass loss can be seen from Fig. 16. For the grid of 200 nodes/Diameter, the effective grid resolution of the secondary bubbles ranges from 1 to 5 nodes, and the thin filaments of the gas are under-resolved, which causes significant discretization errors for the level set equation. These errors are reduced with finer grids of 400/Diameter and 800/Diameter, providing better resolutions of the thin gas structures.

Discussion. The above-discussed overall dynamics of the bubble collapse is similar to those of previous simulations of this problem by Ball et al. [3] (using so-called "free Lagrange" method), and by Hu and Khoo [24] (using the modified GFM). Compared to both previous simulations, we have obtained slightly higher velocity of the jet at the moment of its impact upon the opposite side of bubble, i.e., 2.85 vs. 2.4 km/s in [3], and

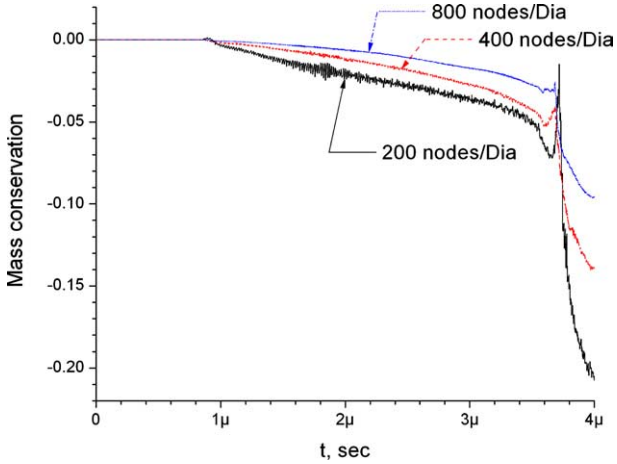

Fig. 19. On mass conservation for a bubble collapse problem.

2.8 km/s in [24]. The differences might be related to different equations of state (stiffened EOS here vs. Tait EOS in [3,24]) and to insufficient grid resolution of [3,24], who have employed the meshes with approximately 200 nodes/Diameter, which corresponds to the coarsest grid used here. Due to better grid resolution, our maximum gas temperature is higher than in [3,24], since $T_{\max}$ is sensitive to details of the gas dynamics at the latest stages of bubble collapse.

7.5. Example V: large scale, strong underwater explosion

In the present section, we consider underwater explosion. Similar, but smaller-scale simulations are previously performed by Liu et al. [28], and Grove and Menikoff [20]. Here, we focus on capability of simulating large-scale domain with very fine grid resolution near the interface.

Problem formulation. We consider an explosion in a 2D sea of water 14 m-deep and 100 m-long. The explosion zone is represented as a highly pressurized air cylinder, 2 m in diameter, initially located in the center of the computational domain with size (100×50 m), and positioned 4 m below the water surface. Initially, all fluids are motionless under 1 atm. Densities of water and atmospheric air were set to 1000 and 1 kg/m^3, respectively. The explosion zone is initially pressurized to $P = 9.12 \times 10^8$ Pa and $\rho = 1270$ kg/m^3, which corresponds to 7.163 GJ/m of the total stored energy. The air is considered as a γ-gas with $\gamma = 1.4$, while water was modeled using the stiffened gas EOS with $\gamma = 4.4$ and $\Pi = 6 \times 10^8$. The non-reflection BCs are applied at the top, left and right boundaries of the computational domain. Reflection (wall) BCs are used at the bottom. Six levels of AMR adaptation with refinement ratio 2 were used, which corresponds to the smallest grid size of 1.5625 cm (128 nodes per initial diameter of the explosion zone). If a uniform mesh were applied with this grid resolution, the size of the computational grid would be 20.5 million nodes. With AMR, only approximately 0.6 million nodes were used. We employed the LLF/MP-MUSCL$_3^{\text{MinMod}}$ scheme and CFL = 0.4.

Dynamics. The results of numerical simulations are presented in Figs. 20 and 21 as dynamics of the Mach number, pressure and numerical Schlieren fields. Schlieren images are created using Eq. (20) with $\kappa = 50$ and 100 for water and air, respectively. Outlines of SAMR patches are shown in Figs. 20(d), (g) and 21. After the explosion initiation, the main shock expands radially outwards into the water. Simultaneously, there is a rarefaction wave inside the explosion zone, traveling inwards. The main shock hits the free-surface at approximately 1.3 ms, causing formation of the reflected Prandtl–Meyer rarefaction wave, Fig. 20(b), and a weak transmitted air shock, Fig. 20(d). At approximately 2.7 ms, the reflected rarefaction wave interacts with the explosion zone's interface, leading to a formation of the reflected compression wave and a transmitted rarefaction wave, Fig. 20(c). Next, due to the interaction between the reflected rarefaction wave and upper boundary of the bubble, the top section of the bubble is accelerated in the vertical direction, causing elongated

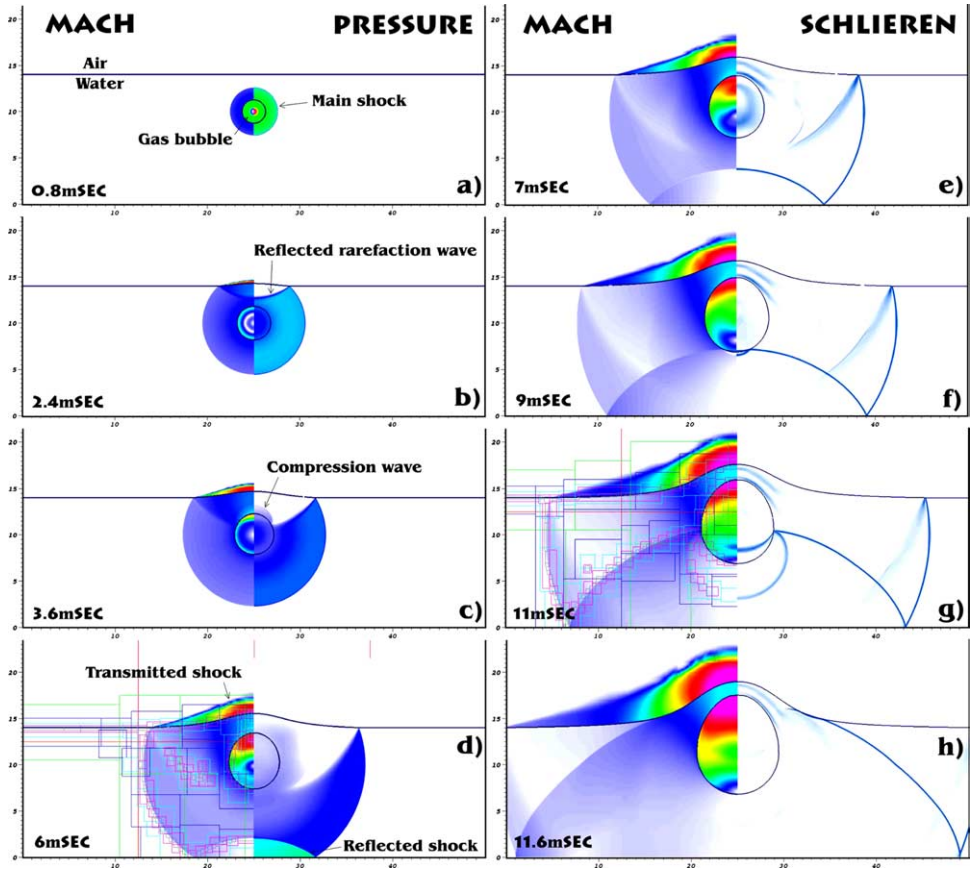

Fig. 20. Mach, pressure and Schlieren fields for numerical example V. Six levels of adaptation with refinement ratio 2. The effective grid resolution (on the finest level) is 128 nodes per initial diameter of explosion zone.

bubble's shape, Fig. 20(h). At the same time, the free surface is disturbed and moves upwards due to relief of the main shock. The dynamics of the free surface motion is given more specifically in Fig. 22.

Discussion. One of the purposes of this type of simulation is to estimate the source for generated surface momentum waves. As shown in Fig. 22, by the time of 24 ms, the explosion has produced a nearly 10 m-high wave, which is about to travel horizontally due to gravity (not modeled here). The acceleration of the free surface at $t = 24$ ms is approximately 433 g, which indicates that, under these conditions, the gravity effects on the initial generation of the surface wave are negligible. Finally, since the effects of cavitation are not modeled here, our simulations do not represent free-surface spallation phenomena. However, we believe that the spallation should not significantly affect the height of the generated surface wave.

7.6. Example VI: Interaction of a shock wave with a cylindrical liquid mass

This problem is particularly difficult, because it involves a curved gas/liquid slow–fast interface, and associated with this rather complex irregular shock refraction patterns,[10] including the transition from the free precursor (FPR) to the free von Neumann refraction (FNR), back-refraction with anomalous reflection, and strong rarefactions due to focusing geometry of the interface. It turns out that numerical schemes in general are sensitive to resolution of irregular refractions, especially in the case of gas–liquid configuration, when the errors are easily amplified due to the stiffness of EOS for liquid [36].

[10] Irregular refraction patterns were first studied by Henderson et al. [22] for a flat slow–fast gas–gas interface, and, more recently, by present authors, for both gas–gas and gas–liquid interfaces, in slow–fast and fast–slow configurations [36].

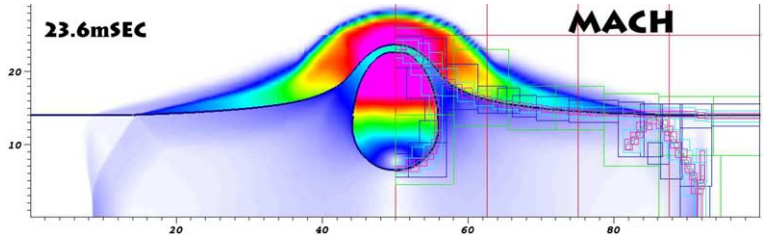

Fig. 21. Mach number field and outline of AMR patches for $t = 23.6$ ms, for numerical example V.

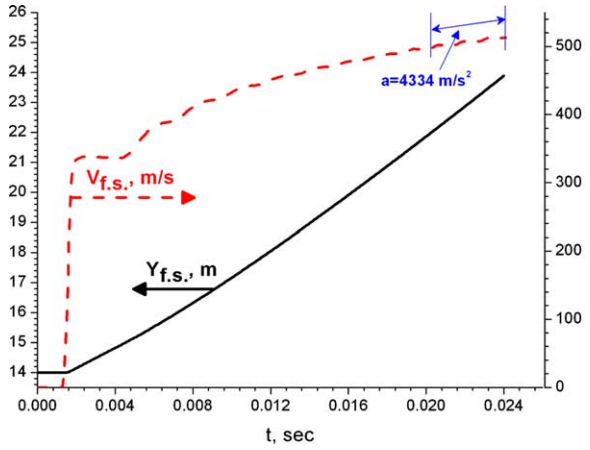

Fig. 22. Dynamics of the explosion-generated free surface wave for numerical example V.

Problem formulation. A cylindrical liquid mass is positioned at the center of the computational domain of size 32×32 mm. The diameter of the cylinder is 6.4 mm. Initially, the liquid is suspended in motionless gas, under the following conditions $\mathbf{u} = (0,0)$ m/s, $P = 0.1$ MPa, $\rho_{\text{gas}} = 1$ kg/m^3 and $\rho_{\text{liquid}} = 1000$ kg/m^3. The following parameters of the stiffened equation of state are used

$$\begin{bmatrix} \gamma \\ \Pi \end{bmatrix} = \begin{bmatrix} 2.8 & \\ 3.036 & 10^8 \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} 1.4 \\ 0 \end{bmatrix}$$

for liquid and gas, respectively. A planar incident $M_{\text{sh}} = 1.47$ shock wave with post-shock conditions

$$\mathbf{V}_{\text{post-shock}} = \begin{bmatrix} 2.35 & 10^5 \\ 1.811 & \\ 246.24 & \\ 0 & \end{bmatrix}$$

is initially placed 4 mm to the left of the cylinder's center. BCs are periodical in the vertical direction and non-reflection for both left and right boundaries of the domain. Two computational meshes are used, with three and six AMR levels of adaptation, both at a refinement ratio 2. The effective grid resolutions for these meshes were 80 and 640 computational nodes per cylinder's diameter. With the finest mesh, high-quality numerical Schlieren images are feasible, created using Eq. (20) with $\kappa = 10^5$ and 5×10^3 for the liquid and gas, respectively. We employed the LLF/MP $\text{MUSCL}_3^{\text{MinMod}}$ scheme and CFL = 0.2.

Dynamics. Figs. 23 and 24 show the dynamics by means of the numerical Schlieren during the first 36 μs of the interaction. When the incident shock hits the liquid, it is reflected as a shock, transmitting a pressure wave into the liquid. Due to the curved geometry of the interface, the incident shock is refracted, forming the so-called "regular refraction pattern" (RRR) [36], when the incident shock angle β is sufficiently small. The intersection of the incident, reflected shocks and transmitted pressure wave is a "refraction node", which exists only during the first 2–3 μs. When the incident angle becomes sufficiently large ($t = 4$ μs), the transmitted

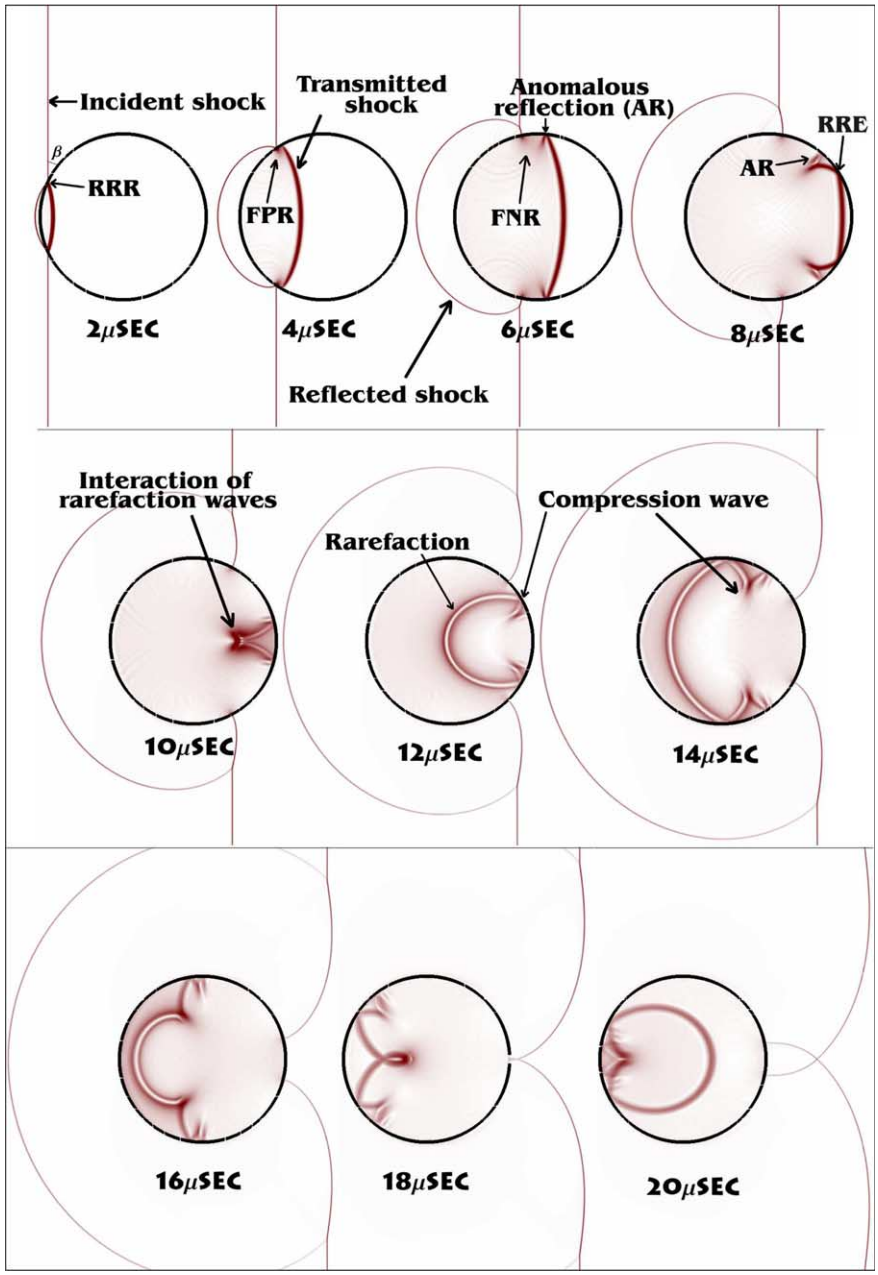

Fig. 23. Dynamics of the Schlieren for numerical example VI. Six levels of adaptation with refinement ratio 2. Effective grid resolution (on the finest level) is 640 nodes per drop diameter.

pressure wave ''peels-off'' the refraction node, transforming into a ''precursor'' pressure wave, and the refraction pattern corresponds to the so-called irregular ''free precursor refraction'' (FPR) [36]. With a further increase of the incident angle β, the combination of the incident and reflected shocks transforms into the Mach reflection, forming the ''free von Neumann refraction'' (FNR) irregular pattern [36]. When the transmitted pressure wave passes the equator of the cylinder at $t = 6$ μs, it is back-refracted in a ''fast–slow'' configuration. Since the transmitted pressure wave at this moment is nearly perpendicular to the interface, the back-refraction pattern is irregular, corresponding to the so-called anomalous reflection (AR) [20]. The back-transmitted pressure wave in the gas is extremely weak due to high acoustic impedance mismatch, and it is indistinguishable in the plotted numerical Schlieren field. By $t = 8$ μs, the incident angle of the back-(fast–slow)-refraction

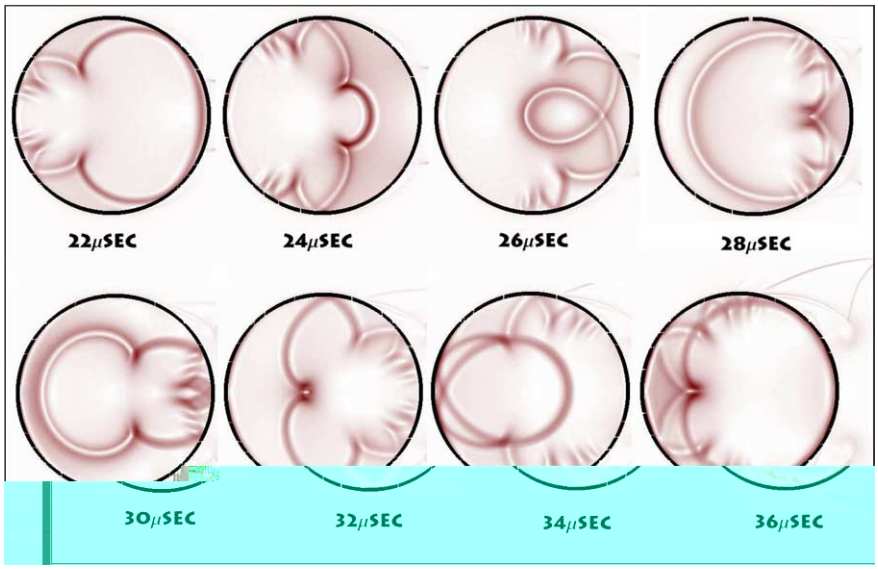

Fig. 24. Dynamics of the Schlieren for numerical example VI (continued).

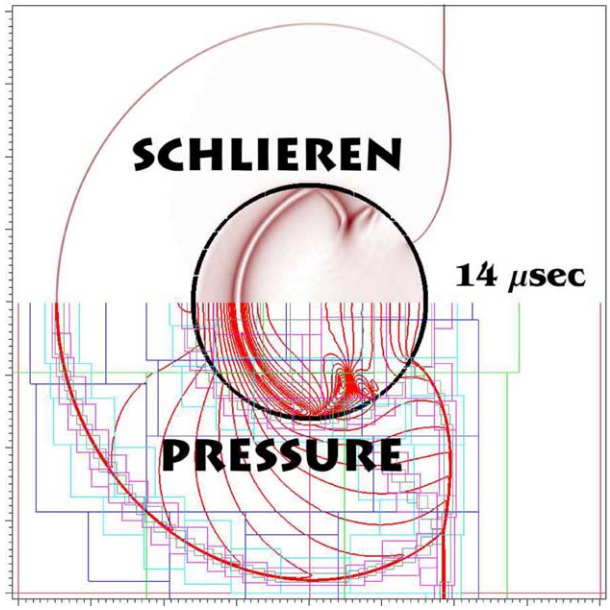

Fig. 25. Numerical Schlieren (top), pressure (bottom) and outline of the SAMR patches for numerical example VI. Six levels of adaptation with refinement ratio 2. The effective grid resolution (on the finest level) is 640 nodes per drop diameter.

becomes smaller, which results in formation of the regular pattern with reflected rarefaction, denoted as RRE [36]. Due to the focusing geometry of the interface, the two reflected rarefaction waves collide with each other at $t = 10$ μs, forming a strong rarefaction wave with negative pressure (cavitation). Even though the stiffened EOS (4) is not designed to properly represent the physics of the cavitation, it is capable of handling negative pressures up to $-P_0$, without a failure of the numerical scheme. At this point, we have found that the numerics is sensitive to discretization errors, which could cause pressure to be unacceptable $P < -P_0$, and a proper grid resolution is a must for stability.

As the reflected rarefaction wave moves backwards, $t = 12$ μs, it interacts with non-uniform flow field, which weakens the rarefaction, and, when the equator is passed at $t = 14$ μs, converts it into a compression

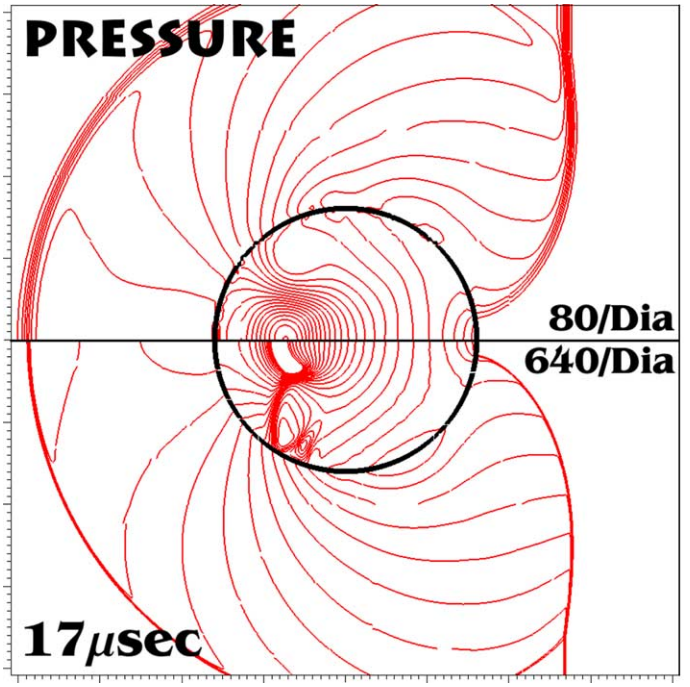

Fig. 26. Effect of the grid resolution on pressure field for numerical example VI. Three levels of adaptation (80 nodes/Diameter) vs. six levels of adaptation (640 nodes/Diameter), with refinement ratio 2.

wave, due to the converging geometry of the interface. Pressure wave dynamics for time from 16 to 36 μs is rather complex, involving multiple interactions of different refraction patterns, as shown in Figs. 23 and 24.

The detailed numerical Schlieren field, pressure contours and outlines of the SAMR patches are shown in Fig. 25. In Fig. 26, we show the comparison of the coarse- and fine-grid simulations, as pressure contours at $t = 17$ μs. As it can be seen, under the coarse grid, many details of pressure wave refractions are lost due to under-resolution. In addition, there is a small noise in the pressure field near the interface, which is essentially eliminated in the fine-grid results.

8. Conclusion

This paper reports on an evolutionary step in sharp capturing of shocked, high acoustic impedance mismatch (AIM) interfaces, in an adaptive mesh refinement (AMR) environment. A broad set of test problems (including shocked multi-gaseous media, bubble collapse, underwater explosion and shock passing over a liquid drop suspended in a gaseous medium) was considered and the results demonstrate that the fundamental assumptions/approximations made in modifying the AMR prolongation/restriction operators and in using the NNI algorithm for interfacial treatment are acceptable from the accuracy point of view, while they enable an effective implementation and utility of the structured AMR technology for solving complex multiphase problems in highly compressible settings. Through the development and testing exercises we have gained significant insights about interactions between different elements of the treatment and their roles in an AMR environment. Specifically, our main conclusions and recommendations are as follows.

1. The key was to overcome complexities in implementing AMR's multi-level dynamic solutions, in time and space, with a moving, high acoustic impedance mismatch interface. This was made possible by relaxing the strict conservative treatment of AMR prolongation/restriction operators in the interfacial region, and by using a natural-neighbor-interpolation (NNI) algorithm, which eliminates the need for extrapolation into the other fluid, in a characteristics-based matching (CBM) scheme.

2. The use of CBM and the NNI algorithm for interfacial treatment (including flux extrapolation from inner points to obtain interfacial values) does not introduce detectable errors even in highly compressible situations with shocks passing through high acoustic impedance mismatch interfaces.
3. We show that strict conservative treatment in prolongation/restriction operators in the interfacial region, as needed for synchronizing multi-level AMR solutions, leads to significant algorithmic difficulties and numerical errors, and that it can be relaxed within the high-resolution interfacial region made possible by AMR.
4. Similarly, with AMR, dimension-by-dimension treatment for computing fluxes at interfaces in multi-dimensions seems to be a viable approach. However, further improvements in this area are possible.
5. Shocks and rarefactions at gas–liquid interfaces can lead to fine structure behaviors that unless properly resolved will lead to disconvergence.

Acknowledgments

This work was supported by the Lawrence Livermore National Laboratory (“ALPHA”, “MIX” and “ASOS” projects), the National Ground Intelligence Center (NGIC) and the US Army. Support of Drs. Frank Handler and Glen Nakafuji at LLNL, Dr. Rick Babarsky at NGIC, Ed Conley, JBCCOM at RDEC, and Mr. John Pace and Mr. Charles Fromer of the JSTO at DTRA is gratefully acknowledged. We thank LLNL’s SAMRAI team and, in particular, Drs. Richard Hornung and Andy Wissink for help and kindly allowing us to use SAMRAI’s utilities.

References

[1] R. Abgrall, Generalization of Roe scheme for the computation of mixture of perfect gases, Rech. Aérosp. 6 (1988) 31.
[2] R. Abgrall, S. Karni, Computations of compressible multifluids, J. Comput. Phys. 169 (2001) 594–623.
[3] G.J. Ball, B.P. Howell, T.G. Leighton, M.J. Schofield, Shock-induced collapse of a cylindrical air cavity in water: a free-Lagrange simulation, Shock Waves 10 (2000) 265–276.
[4] D.S. Balsara, C.-W. Shu, Monotonicity preserving weighted essentially non-oscillatory schemes with increasingly high order of accuracy, J. Comput. Phys. 160 (2000) 405–452.
[5] M.J. Berger, P. Colella, Local adaptive mesh refinement for shock hydrodynamics, J. Comput. Phys. 82 (1989) 67–84.
[6] M.J. Berger, J. Oliger, Adaptive mesh refinement for hyperbolic partial differential equations, J. Comput. Phys. 53 (1984) 482–512.
[7] M.J. Berger, I. Rigoutsos, An algorithm for point clustering and grid generation, IEEE Trans. Syst. Man Cybernet. 21 (5) (1991) 1278–1286.
[8] N.K. Bourne, J.E. Field, Shock induced collapse of single cavities in liquids, J. Fluid Mech. 244 (1992) 225.
[9] I.L. Chern, J. Glimm, O. McBryan, B. Plohr, S. Yaniv, Front tracking for gas dynamics, J. Comput. Phys. 62 (1985) 83–110.
[10] J.-P. Cocchi, R. Saurel, A Riemann problem based method for the resolution of compressible multimaterial flows, J. Comput. Phys. 137 (1997) 265–298.
[11] R.P. Fedkiw, The ghost fluid method for discontinuities and interfaces, in: E.F. Toro (Ed.), Godunov Methods, Kluwer Academic Publishers, New York, 2001.
[12] R.P. Fedkiw, A. Marquina, B. Merriman, An isobaric fix for the overheating problem in multimaterial compressible flows, J. Comput. Phys. 148 (1999) 545–578.
[13] R.P. Fedkiw, T. Aslam, B. Merriman, S. Osher, A non-oscillatory Eulerian approach to interfaces in multimaterial flows (the ghost fluid method), J. Comput. Phys. 152 (1999) 457–492.
[14] R.P. Fedkiw, Coupling an Eulerian fluid calculation to a Lagrangian solid calculation with the ghost fluid method, J. Comput. Phys. 175 (2002) 200–224.
[15] R.P. Fedkiw, T. Aslam, S. Xu, The ghost fluid method for deflagration and detonation discontinuities, J. Comput. Phys. 154 (1999) 393–427.
[16] F. Gibou, R. Fedkiw, A fourth order accurate discretization for the Laplace and heat equations on arbitrary domains, with applications to the Stefan problem, J. Comput. Phys. 202 (2005) 577–601.
[17] J. Glimm, J.W. Grove, X.L. Li, K.-M. Shyue, Y. Zeng, Q. Zhang, Three-dimensional front tracking, SIAM J. Sci. Comput. 19 (1998) 703–727.
[19] S.K. Godunov, A.V. Zabrodin, M.Ya. Ivanov, A.N. Kraiko, G.P. Prokopov, Numerical Modeling of Multdimensional Gas Dynamics Problems, Nauka, Moscow, 1976 (in Russian).
[20] J.W. Grove, R. Menikoff, Anomalous reflection of a shock wave at a fluid interface, J. Fluid Mech. 219 (1990) 313–336.
[21] J.-F. Haas, B. Sturtevant, Interaction of weak shock waves with cylindrical and spherical gas inhomogeneties, J. Fluid Mech. 181 (1987) 41–76.
[22] L.-F. Henderson, P. Colella, E.G. Puckett, On the refraction of shock waves at a slow–fast gas interface, J. Fluid Mech. 224 (1991) 1–27.

[23] R.D. Hornung, S.R. Kohn, Managing application complexity in the SAMRAI object-oriented framework, Concurrency Computation: Practice Experience 14 (special issue) (2002) 347–368.
[24] X.Y. Hu, B.C. Khoo, An interface interaction method for compressible multifluids, J. Comput. Phys. 198 (1) (2004) 35–64.
[25] G.S. Jiang, C.-W. Shu, Efficient implementation of weighted ENO schemes, J. Comput. Phys. 126 (1996) 202–228.
[26] S. Karni, Multicomponent flow calculations by a consistent primitive algorithm, J. Comput. Phys. 112 (1994) 31–43.
[27] L.D. Landau, E.M. Lifschitz, Theoretical Physics, v.1: Mechanics, fourth ed., Nauka, Moscow, 1988.
[28] T.G. Liu, B.C. Khoo, K.S. Yeo, The simulation of compressible multi-medium flow. II. Applications to 2D underwater shock refraction, Comput. Fluids 30 (2001) 315–337.
[29] T.G. Liu, B.C. Khoo, K.S. Yeo, Ghost fluid method for strong shock impacting on material interface, J. Comput. Phys. 190 (2003) 651–681.
[30] F. Losasso, F. Gibou, R. Fedkiw, Simulating water and smoke with an octree data structure, in: SIGGRAPH 2004, ACM TOG 23, 2004, pp. 457–462.
[31] F. Losasso, R. Fedkiw, S. Osher, Spatially adaptive techniques for level set methods and incompressible flow, Comput. Fluids (in press).
[32] R. Menikoff, Errors when shock waves interact due to numerical shock width, SIAM J. Sci. Comput. 15 (5) (1994) 1227.
[33] G.H. Miller, E.G. Puckett, A high-order Godunov method for multiple condensed phases, J. Comput. Phys. 128 (1996) 134–164.
[34] W. Noh, Errors for calculations of strong shocks using an artificial viscosity and an artificial heat flux, J. Comput. Phys. 72 (1978) 78.
[35] R.R. Nourgaliev, T.N. Dinh, T.G. Theofanous, Direct numerical simulation of compressible multiphase flows: interaction of shock waves with dispersed multimaterial media, in: Proceedings of the 5th International Conference on Multiphase Flow, ICMF'04, Yokohama, Japan, May 30–June 4, Paper No.494, 2004 (CD-ROM).
[36] R.R. Nourgaliev, S.Y. Sushchikh, T.N. Dinh, T.G. Theofanous, Shock wave refraction patterns at interfaces, Int. J. Multiphase Flow (in press).
[37] R.R. Nourgaliev, S. Wiri, T.N. Dinh, T.G. Theofanous, On improving mass conservation of level set by reducing spatial discretization errors, Int. J. Multiphase Flow (in press).
[38] R.R. Nourgaliev, S. Wiri, T.N. Dinh, T.G. Theofanous, Adaptive strategies for mass conservation in level set treatment, in: Proceedings of the 17th AIAA Computational Fluid Dynamics Conference, AIAA Paper 2005-5348, June 6–9, Toronto, Canada, 2005.
[39] S. Osher, R. Fedkiw, Level set methods and dynamic implicit surfacesApplied Mathematical Sciences, vol. 153, Springer, New York, Berlin, Heidelberg, 2003.
[40] S. Osher, J.A. Sethian, fronts propagating with curvature-dependent speed: algorithms based on Hamilton–Jacobi formulations, J. Comput. Phys. 79 (1988) 12–49.
[41] J.J. Quirk, S. Karni, On the dynamics of a shock–bubble interaction, J. Fluid Mech. 318 (1996) 129–163.
[42] D.P. Peng, B. Merriman, S. Osher, H. Zhao, M. Kang, A PDE-based fast local level set method, J. Comput. Phys. 155 (1999) 410–438.
[43] W.H. Press, S.A. Teukolsky, W.T. Vetterling, B.P. Flannery, Numerical Recipies in C (The Art of Scientific Computing), second ed., Cambridge University Press, Cambridge, 1997.
[44] SAMRAI: Structured Adaptive Mesh Refinement Application Infrastructure. Available from: <www.llnl.gov/CASC/SAMRAI>.
[45] R. Saurel, R. Abgrall, A simple method for compressible multifluid flows, SIAM J. Sci. Comput. 21 (3) (1999) 1115–1145.
[46] J.A. Sethian, Level Set Methods and Fast Marching Methods, Cambridge University Press, Cambridge, 1999.
[47] R. Sibson, A brief description of natural neighbor interpolation, in: V. Barnett (Ed.), Interpreting Multivariate Data, Wiley, New York, 1981, pp. 21–36.
[48] C.-W. Shu, S. Osher, Efficient implementation of essentially non-oscillatory shock-capturing schemes II (two), J. Comput. Phys. 83 (1989) 32–78.
[49] M. Sussman, P. Smereka, S. Osher, A level set method for computing solutions to incompressible two-phase flow, J. Comput. Phys. 119 (1994) 146.
[50] M. Sussman, A.S. Almgren, J.B. Bell, P. Colella, L.H. Howell, M.L. Welcome, An adaptive level set approach for incompressible two-phase flows, J. Comput. Phys. 148 (1999) 81–124.
[51] T.G. Theofanous, R.R. Nourgaliev, G.J. Li, T.N. Dinh, Compressible multi-hydrodynamics (CMH): breakup, mixing, and dispersal, of liquids/solids in high speed flows, in: A. Prosperetti, S. Balachandar (Eds.), Computational Approaches to Disperse Multiphase Flow, Springer Verlag, Heidelberg, 2005, in press.
[52] G.D. van Albada, B. van Leer, W.W. Roberts, A comparative study of computational methods in cosmic gas dynamics, Astron. Astrophys. 8 (1982) 76–84.
[53] B. van Leer, Towards the ultimate conservative difference scheme, a second order sequel to Godunov's method, J. Comput. Phys. 32 (1979) 101–136.
[54] D. Watson, nngridr – An Implementation of Natural Neighbor Interpolation, Dave Watson Publisher, Claremont Australia, 1994.
[55] A. Wissink, D. Hysom, R. Hornung, Enhancing scalability of parallel structured AMR calculations, in: Proceedings of the 17th ACM International Conference on Supercomputing (ICS03), San Francisco, CA, June 23–26, 2003, pp. 336–347.