# A Higher-Order Generalized Ghost Fluid Method for the Poor for the Three-Dimensional Two-Phase Flow Computation of Underwater Explosions and Implosions

Charbel Farhat *, Arthur Rallu, Sriram Shankaran

*Department of Mechanical Engineering and Institute for Computational and Mathematical Engineering, Stanford University, Mail Code 3035, Stanford, CA 94305, U.S.A.*

**Abstract**

The ghost fluid method for the poor (GFMP) is an elegant, computationally efficient, and nearly conservative method for the solution of two-phase flow problems. It was developed in one dimension for the stiffened gas equation of state (EOS) and one-step time-discretization algorithms. It naturally extends to three dimensions but its extension to higher-order, multi-step time-discretization schemes is not straightforward. Furthermore, the original GFMP and many other ghost fluid methods fail to handle the large density and pressure jumps that are encountered in underwater explosions and implosions. Therefore, the GFMP is generalized in this work to arbitrary EOS and multi-fluid problems with multiple EOSs. It is also extended to three dimensions and developed for higher-order, multi-step time-discretization algorithms. Furthermore, this method is equipped with an exact two-phase Riemann solver for computing the fluxes across the material interface in order to address the stiff nature of the two-phase air/water problem and handle the large discontinuity of the density at the air/water interface. As the original GFMP, the resulting method is computationally efficient and nearly conservative. Its superior performance in the presence of large density and pressure jumps is demonstrated for shock-tube problems. Its practicality and accuracy are also highlighted with the simulation of the growth, collapse, and rebound of an underwater bubble and the comparison of the obtained numerical results to experimental data.

*Key words:* explosion, GFM for the Poor, implosion, Riemann solver, two-phase compressible flow, underwater

## 1 Introduction

The numerical simulation of compressible multi-medium flows arises in many applications including underwater bubble dynamics, shock wave interactions with material discontinuities, and combustion, to name only a few. The motivation for the present work is the accurate and computationally efficient prediction of the bubble dynamics and pressure signatures generated by underwater explosions and implosions. The large size of the bubbles and their energy content result in strong shock and expansion waves. Typically, these propagate through air (vapor) and water and can be reflected or refracted off the air/water interface, which calls for modeling water as a compressible fluid. The ratio of water and air densities ($\approx$ 1,000) is such that the air/water interface is well approximated by a free surface where the gas can only apply a pressure on the liquid.

Underwater explosions and implosions result in bubbles whose characteristic size is considerably larger than that of bubbles obtained in liquid suspensions. Hence, such bubbles are less affected by surface tension and viscous forces and therefore their dynamics can be modeled by the Euler equations. The numerical solution of these equations for a single fluid has reached a state of considerable maturity. Godunov-type schemes [1] and extensions to higher-order semi-discretizations [2, 3] are often the methods of choice for achieving crisp shock resolution in space. A variety of explicit and implicit temporal discretizations have been developed for these schemes and for both steady and unsteady problems. However, initial attempts [4, 5] at the extension of these numerical algorithms to multi-fluid problems suffered from numerical instabilities and oscillations, primarily around the material interface.

The common multi-fluid solution methods published in literature use either a Lagrangian or an Eulerian method. In a Lagrangian method, the computational mesh moves and distorts with the material interface. The interface itself is convected with the local fluid velocity and can be resolved sharply by controlling the numerical diffusion around it. However, if the problem induces large displacements of the interface, the resulting mesh distortions can adversely affect the accuracy and stability of the numerical solution process and often make the Lagrangian approach unpractical.

Eulerian methods use a fixed mesh and usually carry an auxiliary equation for tracking or capturing the material interface. In the VOF approach [6], each computational cell is assumed to possibly contain a mixture of both

* Corresponding author.
*Email addresses:* cfarhat@stanford.edu (Charbel Farhat), arallu@stanford.edu (Arthur Rallu), ssriram@stanford.edu (Sriram Shankaran).

fluids and the volume occupied by each fluid is represented by the volume fraction. The evolution of this fraction is governed by a transport equation where the speed of propagation is determined by the local fluid velocity. The VOF method has been predominantly used for incompressible flows where the knowledge of the interface position is sufficient to recover the density field. For compressible flows, recovering the density field and the internal energies in a cell containing both fluids does not seem to be an obvious task. Another class of Eulerian methods that has found wide-spread usage is based on the level-set equation [7] for capturing the interface. This equation falls under the general class of Hamilton-Jacobi equations. It can also be viewed as a particular case of the transport equation. It governs the evolution of the zero of the level-set function which marks the interface. The level-set equation naturally allows for merger and break-up of the interface, is relatively straightforward to implement, does not incur a significant computational overhead and therefore is an attractive candidate approach for interface capturing. In addition to the volume (or mass) fraction model and the level-set approach, a $\gamma$ model — where $\gamma$ denotes the ratio of specific heats for a given gas — has also been suggested for capturing the evolution of the interface [8]. Here again, the evolution equation is a transport equation. In principle, any function of $\gamma$ can be used as the interface marker, but the ratio $1/(\gamma - 1)$ has been shown to favor a non-oscillatory numerical solution of the pressure at the material interface [5, 9, 8, 10].

Whether in the context of a Lagrangian or Eulerian approach, the numerical treatment of the Euler equations at the material interface still needs to be addressed. Early attempts at the numerical solution of multi-medium flows in an Eulerian setting resulted either in mass fractions outside the valid range of $[0, 1]$, or in pressure oscillations across the material interface. These oscillations are present even in first-order, monotonicity preserving schemes. To suppress them, particular forms of the discretization of the conserved variables and/or particular functions for capturing the evolution of the interface have been proposed [8].

Amid attempts to prevent pressure-oscillations in multi-fluid calculations, the ghost fluid method (GFM) was developed as a more economical alternative solution method [7]. The main feature of this method is its simplicity: it allows multi-fluid computations to be performed in the vicinity of the material interface as if they pertained to a single medium domain. Given an interface capturing technique — usually, the level-set method — the GFM exploits the concept of ghost and real fluid cells and manages them with an overlapping Schwarz-like numerical procedure. In the material interface region, it sets the values of the pressure and normal velocity in the ghost fluid cells to those in the real fluid cells. To eliminate an otherwise spurious "over-heating" phenomenon, it computes the density of the ghost fluid using an isobaric technique [11]. In its basic form, the GFM is non conservative [12]. However, it

can be equipped with an *a posteriori* correction procedure that first measures the *discrete* conservation errors generated in the neighborhood of the material interface during a given time-step, then offsets them using an error redistribution technique. This correction procedure was proposed in [13] where it was applied to stiff detonation problems. Unfortunately, the GFM fails to solve most air/water problems of interest. For such problems, it either delivers inaccurate results because of spurious oscillations, or simply fails to deliver any result because of the large discontinuity of the density at the air/water interface [14]. An improved version of this method incorporating in a one-step time-integration scheme an approximate two-phase Riemann solver at the material interface that assumes either a two-shock or two-rarefaction wave structure was proposed in [14] for the solution of gas/water problems and illustrated with simple 1D and 2D calculations. Like the original GFM, this enhanced version relies on the isobaric technique for eliminating the spurious over-heating phenomenon. For two- and three-dimensional applications, this isobaric fix requires the solution of yet another auxiliary partial differential equation (PDE) [7] and therefore increases further the computational complexity of the method. Most recently, the approximate Riemann solver of [14] was replaced in [16] by an exact version to eliminate the need for the isobaric fix.

The "overlapping" aspect of the GFM induces a combined storage and computational overhead that is application dependent. The ghost fluid method for the poor (GFMP) [15] is a variant method which avoids most of this overhead by computing two numerical fluxes at the material interface: one using the thermodynamic parameters of the fluid on the left side of the interface, and another one using the thermodynamic parameters of the other fluid medium. It is an elegant, computationally efficient, and nearly conservative method in the sense that it conserves all conservative variables except the energy across the material interface. The GFMP was developed in [15] for one-dimensional problems using a one-step explicit time-integration algorithm and assuming that each given fluid is a stiffened gas. It involves a subtle but crucial conversion from conservative to primitive variables before and after advancing in time the solution of the level-set equation. Its extension to multiple dimensions is straightforward. However, as it will be shown in this paper, its extension to higher-order multi-step time-integrators requires a careful sequencing of its computational steps. More importantly, the GFMP does not apply as formulated in [15] to multi-fluid problems involving either an equation of state (EOS) that is different from that of a stiffened gas, or different EOSs on the two sides of a material interface. Hence, applying the GFMP to air/water problems calls for either modeling both fluid media as stiffened gases, or generalizing this method to a larger number of EOSs and extending it to multi-fluid problems with multiple EOSs. However even when both water and air are modeled as stiffened gases, it is the authors' experience that the GFMP, like the GFM, fails to solve most air/water problems of interest

because of the large discontinuity of the density at the air/water interface.

When an underwater explosion occurs, the resulting energy release creates an expanding gas bubble which undergoes a multiple expansion (explosion)/collapse (implosion) process and continuously loses energy until it breaks down. The bubble oscillation process, the initiation process, the source of the instability leading to bubble collapse, and the energy loss mechanism are not completely understood. Extensive experimental [17] and computational [18] investigations have been conducted to develop a better understanding of these phenomena. Early computational studies were reported in [18] using a simplified, one-dimensional computational model in which water was modeled as a compressible fluid and the bubble was assumed to maintain a spherical shape. The gas inside the bubble was assumed to have a polytropic EOS and to undergo isentropic changes [18]. Excellent correlation with experimental data was obtained for the bubble's radius time-history. In [19], the simplified model developed in [18] was modified to account for energy losses in the gas medium. The resulting computational model produced better correlations with experimental data for the amplitude and phase of the bubble oscillations [19]. However, because they assume spherical symmetry, both models developed in [18] and [19] cannot properly account for bubble migration due to buoyancy. Furthermore, they cannot account for shape changes during the collapse phase when the bubble motion is unstable. Three-dimensional simulations are required for capturing these important details.

Given the context set above, the main objectives of this paper are three-fold: (a) to generalize the GFMP to multi-fluid problems with multiple EOSs and to extend it to higher-order time-discretizations, (b) to enable its application to air/water problems by enhancing its robustness for two-phase problems with large contact discontinuities and strong pressure jumps at the material interface, and (c) to demonstrate its potential for the three-dimensional simulation of underwater explosions and implosions. To this effect, the remainder of this paper is organized as follows.

In Section 2, the governing equations of the two-phase air/water problems of interest are presented and discussed. In Section 3, the GFMP is briefly overviewed. In Section 4, it is generalized to multi-fluid problems with multiple EOSs. Its robustness with respect to a large discontinuity of the density and a strong pressure jump at the material interface is enhanced via the incorporation in Section 5 of an exact, local, one-dimensional, two-phase Riemann solver for computing the interfacial fluxes. The resulting multi-fluid method is referred to as the GFMP-ERS. In Section 6, a computational framework for extending the GFMP-ERS to higher-order multi-step time-discretization algorithms is proposed. In Section 7, the higher-order GFMP-ERS is evaluated using simple benchmark problems, some of which include representative features of underwater explosions and implosions. Then, the potential of the

GFMP-ERS is illustrated with the simulation of the growth, collapse, and rebound of an underwater bubble and the comparison of the obtained numerical results to experimental data. Finally, Section 8 concludes this paper.

## 2 Governing equations

### *2.1 Eulerian flow*

As already mentioned, underwater explosions and implosions generate bubbles that are usually considerably larger than those encountered in liquid suspensions. Hence, such bubbles are less affected by surface tension and viscous effects. For this reason, their dynamics is modeled in this paper by the Euler equations written in the familiar conservation form

$$\frac{\partial w}{\partial t} + \nabla \cdot \mathcal{F}(w) = 0 \tag{1}$$

where $t$, $w(X,t)$, $X = (x,y,z)$, and $\mathcal{F}$ denote time, the conservative fluid state vector, space, and the convective flux vector, respectively. The initial condition for the above PDE is written as

$$w(X,0) = g(X) \tag{2}$$

and its boundary conditions are not specified here as they are problem dependent.

### *2.2 Equations of state*

Two different EOSs are considered in this paper for modeling compressible water: (1) the stiffened gas equation, and (2) Tait's equation.

#### *2.2.1 The stiffened gas equation*

The stiffened gas equation is a generalization of the perfect gas EOS. It can be written as

$$(\gamma - 1)\rho e = p + \gamma\pi \tag{3}$$

where $\rho$, $e$, and $p$ denote the density, internal energy per unit mass and pressure, respectively, and $\gamma$ and $\pi$ are constants that need to be specified. This EOS is versatile: it has been used for modeling gas, liquid, and solid media. The constants $\gamma$ and $\pi$ are set so that the speed of sound in the medium of interest, $c$, is correctly predicted using this EOS and the definition

$$c = \sqrt{\left.\frac{\partial p}{\partial \rho}\right|_s} \tag{4}$$

where $s$ denotes the entropy.

To evaluate the sound speed $c$, the following thermodynamic equations are first recalled

$$e = c_v T \qquad Tds = de + pd\left(\frac{1}{\rho}\right) \tag{5}$$

where $c_v$ denotes the specific heat at constant volume. From Eq. (3) and Eqs. (5) it follows that

$$Tds = de + pd\left(\frac{1}{\rho}\right) = \left(\frac{p+\gamma\pi}{(\gamma-1)\rho c_v}\right) ds \tag{6}$$

$$= \left(\frac{1}{(\gamma-1)\rho}\right) dp - \left(\frac{p+\gamma\pi}{(\gamma-1)\rho^2}\right) d\rho - \left(\frac{p}{\rho^2}\right) d\rho \tag{7}$$

$$= \left(\frac{1}{(\gamma-1)\rho}\right) dp - \left(\frac{\gamma(p+\pi)}{(\gamma-1)\rho^2}\right) d\rho \tag{8}$$

$$\tag{9}$$

Hence,

$$ds = \left(\frac{c_v}{p+\gamma\pi}\right) dp - \left(\frac{\gamma c_v(p+\pi)}{(p+\gamma\pi)\rho}\right) d\rho \tag{10}$$

and the speed of sound in a stiffened gas is given by

$$c = \sqrt{\left.\frac{\partial p}{\partial \rho}\right|_s} = \sqrt{\frac{\gamma(\pi+p)}{\rho}} \tag{11}$$

For water, the following numerical values of $\pi$ and $\gamma$ are often found in the literature

$$\pi = 6.0 \times 10^8 \ Pa \qquad \gamma \in \{4.4, 5.5, 7.0\} \tag{12}$$

*REMARK 1.* For $\pi = 0$, Eq. (3) simplifies to the perfect gas equation.

### *2.2.2 Tait's equation*

The Tait equation of state models a liquid such as water as a compressible, barotropic liquid whose bulk modulus is an affine function of pressure. Hence, this EOS involves only the density and pressure variables. However, it is a highly non-linear equation of the form

$$p = \eta + \alpha\rho^\beta \tag{13}$$

where $\eta$, $\alpha$, and $\beta$ are three constants that can be determined from the assumption that the bulk modulus $K$ of the liquid is an affine function of pressure determined by two constants $k_1$ and $k_2$ and from the knowledge of a reference state $(\rho_0, p_0)$. Hence,

$$k_1 + k_2 p = K = \rho\frac{dp}{d\rho} = \beta\alpha\rho^\beta = \beta(p - \eta) \tag{14}$$

which gives

$$\eta = -\frac{k_1}{k_2} \qquad \beta = k_2 \tag{15}$$

Furthermore, writing $p_0 = p(\rho_0)$ gives

$$\alpha = \frac{p_0 + \frac{k_1}{k_2}}{\rho_0^{k_2}} \tag{16}$$

In the litterature, the following numerical values are often found for water

$$k_1 = 2.07 \times 10^9 \ kg.m^{-3}.s^{-2} \qquad k_2 = 7.15 \tag{17}$$

When water is modeled by Tait's EOS, the speed of sound in this fluid is given by

$$c = \sqrt{\frac{dp}{d\rho}} = \sqrt{\alpha\beta\rho^{\beta-1}} = \sqrt{\left(\frac{k_2 p_0 + k_1}{\rho_0}\right)\left(\frac{\rho}{\rho_0}\right)^{k_2 - 1}} \tag{18}$$

For $\rho_0 = 1,000\ kg.m^{-3}$ and $p_0 = 10^6\ Pa$, the predicted speed of sound is

$$c_0 = c(\rho_0) = \sqrt{\frac{k_2 p_0 + k_1}{\rho_0}} = 1,441.23\ m.s^{-1} \tag{19}$$

as expected.

*REMARK 2.* When a fluid is modeled by Tait's EOS, the energy equation becomes decoupled from the continuity and momentum equations.

*REMARK 3.* Consider a stiffened gas that is undergoing an isentropic transformation. Let $T$, $s$, and $h$ denote temperature, entropy and enthalpy, respectively. From the second principle of thermodynamics it follows that

$$0 = Tds = dh - \frac{dp}{\rho} = de + pd\left(\frac{1}{\rho}\right) \tag{20}$$

From the stiffened gas equation (3) and its differentiation it follows that

$$de = \left(\frac{1}{(\gamma-1)\rho}\right) dp - \left(\frac{p+\gamma\pi}{(\gamma-1)\rho^2}\right) d\rho \tag{21}$$

Substituting Eq. (21) into Eq. (20) yields after expansion

$$\begin{aligned} 0 &= \left(\frac{1}{(\gamma-1)\rho}\right) dp - \left(\frac{p+\gamma\pi}{(\gamma-1)\rho^2}\right) d\rho - \left(\frac{p}{\rho^2}\right) d\rho \\ &= \left(\frac{1}{p+\pi}\right) dp - \left(\frac{\gamma}{\rho}\right) d\rho \end{aligned} \tag{22}$$

From the integration of the above result, it follows that

$$\exists\ k_3 \in \mathbb{R}\ /\ p = k_3 \rho^\gamma - \pi \tag{23}$$

which shows that Tait's EOS (13) corresponds to the particular case of an isentropic stiffened gas EOS with $\gamma = \beta = k_2$, $\pi = -\eta = \dfrac{k_1}{k_2}$ and $k_3 = \alpha = (p_0 + k_1/k_2)/\rho_0^{k_2}$.

### *2.3 Conservative level-set approach*

In this work, the level-set method is adopted for capturing the material interface. More specifically, the level-set equation is written in conservation form

as follows:

$$\frac{\partial(\rho\phi)}{\partial t} + \nabla \cdot (\rho u \phi) = 0 \tag{24}$$

where $\rho$ and $u$ are the density and velocity vector of the fluid, respectively, and $\phi$ is a function initialized to the distance between each grid point and the material interface. Hence, $\phi = 0$ captures the interface. To ensure that the signed distance function property of $\phi$ is preserved during the computations, $\phi$ can be periodically re-initialized using the algorithm proposed in [20]. This algorithm consists of solving the auxiliary PDE

$$\frac{\partial \psi}{\partial t^*} + S(\phi)(\nabla(\psi) - 1) = 0 \qquad \text{where} \qquad S(\phi) = \frac{\phi}{\sqrt{\phi^2 + \epsilon^2}} \tag{25}$$

$t^*$ denotes a pseudo-time and $\epsilon$ is a small real number, and setting

$$\phi(X, t) = \psi^*(X) \tag{26}$$

where $\psi^*(X)$ is the steady-state solution of Eq. (25).

### *2.4 Semi-discretization*

The finite volume (FV) method is chosen here to semi-discretize all PDEs introduced above. Given a Computational Fluid Dynamics (CFD) grid, this method transforms the Euler flow equations into

$$\frac{\partial w}{\partial t} + \int_{C_i} \nabla \cdot \mathcal{F}(w) dV = 0 \tag{27}$$

where $C_i$ is the volume of the cell or control volume surrounding the $i$-th grid point. Throughout this paper, the control volumes are assumed to be constructed by connecting the centroids of the triangular faces of the tetrahedra and the midpoints of the edges (Fig. 1). The resulting grid is referred to as the "dual" CFD grid.

Using integration by parts, the volume integral in Eq. (27) is converted to a surface integral across the boundary of the control volume and approximated by

$$F_i(W) = \sum_{j \in \kappa(i)} \text{mes}(\partial C_{ij}) \Phi_{ij}(W_i, W_j, n_{ij}) \tag{28}$$

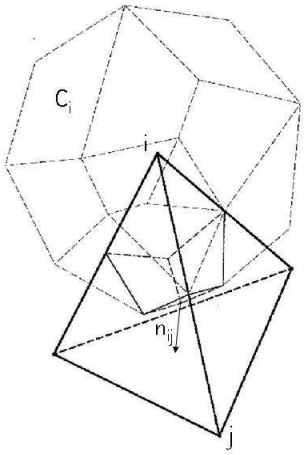


Fig. 1. Control volume (lighter lines) in an unstructured tetrahedral (heavier lines) mesh (only one of the tetrahedra needed to construct the graphically depicted control volume is shown)

where $\kappa(i)$ is the set of vertices connected by an edge to vertex $i$, $\partial C_{ij}$ is the segment of the boundary of $C_i$ that intersects edge $i$-$j$, mes$(\partial C_{ij})$ is its measure, $\Phi_{ij}$ denotes the numerical flux function across $\partial C_{ij}$, $W_i$ denotes the discrete fluid state vector at vertex $i$ and $n_{ij}$ is the unitary outer normal to $\partial C_{ij}$.

To achieve second-order spatial accuracy and address in this case potential numerical oscillations, the FV scheme is equipped with the MUSCL (Monotonic Upwinding Scheme for Conservation Laws) interpolation procedure [2] and a slope limiter. In this case, the approximation (28) is replaced by

$$F_i(W) = \sum_{j \in \kappa(i)} \text{mes}(\partial C_{ij}) \Phi(W_{ij}, W_{ji}, n_{ij}) \tag{29}$$

where $W_{ij}$ and $W_{ji}$ are two extrapolated and limited fluid state vectors. Note that for the MUSCL interpolation procedure, the computation of the gradients at a node is based on the state values of its neighbours that lie on the same side of the material interface, but not on the state values of its neighbours that lie in another fluid.

## 3 The ghost fluid method for the poor

The GFMP proposed in [15] is a computationally lighter alternative to the original GFM. It trades most of the redundant storage and computational

requirements of the GFM in the vicinity of the material interface with the evaluation of two numerical fluxes: one using the thermodynamic parameters of the fluid at the left side of the interface, and another one using the thermodynamic parameters of that on its right side.

When the level-set method is chosen for capturing the material interface between two fluids modeled by the stiffened gas EOS, Roe's solver [21] is chosen for computing the numerical flux functions in Eq. (28) and time-variation is discretized by the forward Euler scheme using a constant time-step $\Delta t$, the key computational steps of the GFMP between time $t^n = n\Delta t$ and time $t^{n+1} = (n+1)\Delta t$ can be summarized as follows using the notation adopted in this paper:

(1) Capture the material interface by checking the product of the values of the level-set function $\phi$ at vertices $i$ and $j$ of edge $i$-$j$. A positive value indicates that edge $i$-$j$ does not cross the material interface, in which case the numerical flux function is computed as usual. On the other hand, a negative value indicates that this edge crosses the material interface. In this case, compute two different fluxes: one using the coefficients $\gamma$ and $\pi$ of the stiffened gas where node $i$ lies and one using those of the stiffened gas where node $j$ lies. In algorithmic words, this can be written as follows:

$$\begin{aligned} &\text{If } \phi_i^n \times \phi_j^n > 0,\ \Phi_{ij} \ = \text{Roe}\left(W_i^n, W_j^n, (\gamma_i = \gamma_j, \pi_i = \pi_j), n_{ij}\right) \\ &\text{If } \phi_i^n \times \phi_j^n \leq 0, \text{ then} \\ &\qquad\qquad \Phi_{ij} \ = \text{Roe}(W_i^n, W_j^n, \gamma_i, \pi_i, n_{ij}) \\ &\qquad\qquad \Phi_{ji} \ = \text{Roe}(W_i^n, W_j^n, \gamma_j, \pi_j, n_{ij}) \end{aligned} \tag{30}$$

where $\gamma_i$ and $\pi_i$ denote the $\gamma$ and $\pi$ coefficients of the stiffened gas where node $i$ lies, respectively, and $\gamma_j$ and $\pi_j$ denote the $\gamma$ and $\pi$ coefficients of the stiffened gas where node $j$ lies, respectively.

(2) Time-advance the solution of the multi-fluid problem to compute a temporary value $\widetilde{W}_i^{n+1}$ of $W_i^{n+1}$

$$\widetilde{W}_i^{n+1} = W_i^n - \Delta t \sum_{j \in \kappa(i)} \text{mes}(\partial C_{ij}) \Phi_{ij}(W_i^n, W_j^n, n_{ij}) \tag{31}$$

(3) Using the value of the level-set function at time $t^n$, $\phi^n$, unpack the conservative fluid state vector $\widetilde{W}^{n+1}$ — that is, convert it to a vector $V^{n+1}$ of primitive variables to obtain $\tilde{\rho}^{n+1}$ and $\tilde{u}^{n+1}$

$$\widetilde{W}^{n+1} \xrightarrow{\phi^n} V^{n+1} \longrightarrow (\tilde{\rho}^{n+1}, \tilde{u}^{n+1}) \tag{32}$$

(4) Compute $\phi^{n+1}$ by time-advancing the solution of the level-set equation (24) using the forward Euler scheme and the values of $\tilde{\rho}^{n+1}$ and $\tilde{u}^{n+1}$ stored

in $V^{n+1}$

$$(\phi^n, \tilde{\rho}^{n+1}, \tilde{u}^{n+1}) \longrightarrow \phi^{n+1} \tag{33}$$

(5) Using the updated value of the level-set function $\phi^{n+1}$, pack $V^{n+1}$ — that is, transform it into the conservative fluid-state vector $W^{n+1}$

$$V^{n+1} \stackrel{\phi^{n+1}}{\longrightarrow} W^{n+1} \tag{34}$$

The GFMP summarized above was proposed in [15] in one-dimensional form using a one-step explicit time-integration scheme. While it has been used mostly with Roe's flux, it is equally applicable with any flux that preserves a contact discontinuity — that is, preserves a uniform pressure and uniform density input. Its generalization to multiple dimensions is straightforward (for example, Eq. (31) is already written in multiple dimensions). Its generalization to a non stiffened gas EOS and its formulation for multi-fluid problems with different EOSs on both sides of a material interface are also relatively simple. However, its extension to higher-order, multi-step, time-integration schemes is however more subtle as it requires a careful sequencing of the computational steps outlined above. These issues are discussed in Section 4 and Section 5, respectively.

## 4 Generalization to an arbitrary EOS and multi-fluid problems with multiple EOSs

### *4.1 Generalization to an arbitrary EOS*

The original GFMP is generalized here to an arbitrary EOS characterized by $n_q$ parameters $q_k$, $k = 1, \cdots, n_q$, by replacing Step (1) and Eq. (30) of the algorithm described in Section 3 by

$$\begin{aligned}
&\text{If } \phi_i^n \times \phi_j^n > 0,\ \Phi_{ij} = \text{Roe}\left(W_i^n, W_j^n, (q_{k_i} = q_{k_j},\ k = 1, \cdots, n_q), n_{ij}\right) \\
&\text{If } \phi_i^n \times \phi_j^n \leq 0, \text{ then} \\
&\qquad\qquad \Phi_{ij} = \text{Roe}\left(W_i^n, W_j^n, (q_{k_i},\ k = 1, \cdots, n_q), n_{ij}\right) \\
&\qquad\qquad \Phi_{ji} = \text{Roe}\left(W_i^n, W_j^n, (q_{k_j},\ k = 1, \cdots, n_q), n_{ij}\right)
\end{aligned} \tag{35}$$

where $q_{k_i}$ and $q_{k_j}$ denote the values of the parameters of the given EOS for the fluids where node $i$ and node $j$ lie, respectively. This generalization assumes that Roe's solver can be extended to the EOS of interest. This is true, for example, for Tait's equation described in Section 2.2.2.

For multi-medium flow problems with multiple EOSs — and more specifically, for the case where the two fluids on the left and right sides of a material interface are governed by different EOSs — the GFMP method is generalized here by replacing Step (1) and Eq. (30) of the original GFMP by

$$\begin{aligned} &\text{If } \phi_i^n \times \phi_j^n > 0,\ \Phi_{ij} &&= \text{Roe}\left(W_i^n, W_j^n, (\text{EOS}_i = \text{EOS}_j), n_{ij}\right) \\ &\text{If } \phi_i^n \times \phi_j^n \leq 0, \text{ then} \\ &\qquad\qquad \Phi_{ij} &&= \text{Roe}(W_i^n, W_j^n, \text{EOS}_i, n_{ij}) \\ &\qquad\qquad \Phi_{ji} &&= \text{Roe}(W_i^n, W_j^n, \text{EOS}_j, n_{ij}) \end{aligned} \tag{36}$$

where $\text{EOS}_i$ and $\text{EOS}_j$ denote the EOS governing the fluids where node $i$ and node $j$ lie, respectively.

The reader can observe that Eqs. (36) include Eqs. (35) as a particular case. Therefore, Eqs. (36) are adopted to describe the GFMP in the general case of multi-fluid problems with arbitrary and/or multiple EOSs.

## 5 The ghost fluid method of the poor with an exact two-phase Riemann solver

A large number of numerical experiments performed by the authors have revealed that the GFMP is not capable of solving air/water flow problems, particularly at practical mesh resolutions. More specifically, it was found that for this class of two-phase flow applications, the GFMP tends to predict negative pressures and densities on the water side of the material interface. This is because each interfacial flux computation performed in Step (1) of the GFMP uses two fluid state vectors $W_i^n$ and $W_j^n$ from the two different sides of the material interface (see Section 3) and therefore is not robust with respect to the strong interfacial contact discontinuity characterizing air/water flow applications ($\rho_{water}/\rho_{air} = 1,000$). This robustness issue is addressed here by modifying the interfacial flux computation performed in Step (1) of the generalized GFMP described in Section 4 to use two new fluid state vectors, $W_i^{\mathcal{R}^n}$

and $W_j^{\mathcal{R}^n}$, as follows

$$\begin{aligned}
&\text{If } \phi_i^n \times \phi_j^n > 0,\ \Phi_{ij} &&= \text{Roe}\left(W_i^n, W_j^n, (\text{EOS}_i = \text{EOS}_j), n_{ij}\right)\\
&\text{If } \phi_i^n \times \phi_j^n \leq 0, \text{ then}\\
&\qquad\qquad \Phi_{ij} &&= \text{Roe}(W_i^n, W_j^{\mathcal{R}^n}, \text{EOS}_i, n_{ij})\\
&\qquad\qquad \Phi_{ji} &&= \text{Roe}(W_i^{\mathcal{R}^n}, W_j^n, \text{EOS}_j, n_{ij})
\end{aligned} \tag{37}$$

In Eqs. (37) and throughout the remainder of this paper, $W_i^{\mathcal{R}^n}$ and $W_j^{\mathcal{R}^n}$ denote the conservative fluid state vectors associated with the *exact* solution at the air/water interface from the sides where node $i$ and node $j$ lie, respectively, of the following one-dimensional two-phase (air/water) Riemann problem

$$\begin{aligned}
&\frac{\partial w}{\partial t} + \frac{\partial \mathcal{F}}{\partial \xi}(w) = 0\\
&\qquad w(\xi, 0) = \begin{cases} W_i^n & \text{if } \xi \leq 0\\ W_j^n & \text{if } \xi > 0 \end{cases}
\end{aligned} \tag{38}$$

where $\xi$ is the abscissa along the edge $i$-$j$ that crosses the air/water interface and $\xi = 0$ at this interface (see Fig. 2).

Furthermore, the Riemann solver outlined above is also exploited to improve the accuracy of the GFMP applied to the solution of air/water flow problems as follows. At the end of each time-step $t^{n+1}$, Step (5) of this multi-fluid method (see Section 3) is modified to pack the primitive fluid state vector $V^{n+1}$ using $\phi^{n+1}$ for each node that remains on the same side of the material interface during the time-interval $[t^n,\ t^{n+1}]$, and set the conservative fluid state vector $W^{n+1}$ to $W^{\mathcal{R}^n}$ for each node that is traversed by the material interface during this time-interval.

To keep this paper as self-contained as possible, the important aspects of the Riemann problem (38) are described next.

### *5.1 One-dimensional two-phase Riemann problems*

At each time-step, the one-dimensional two-phase Riemann problem (38) is constructed along each edge $i$-$j$ that crosses the material interface which is designated here by the subscript $I$. This problem can be reduced to an explicit expression of the normal velocity at the material interface, $u_I$, as a function of the pressure at this material interface, $p_I$, and a non-linear equation in $p_I$. For

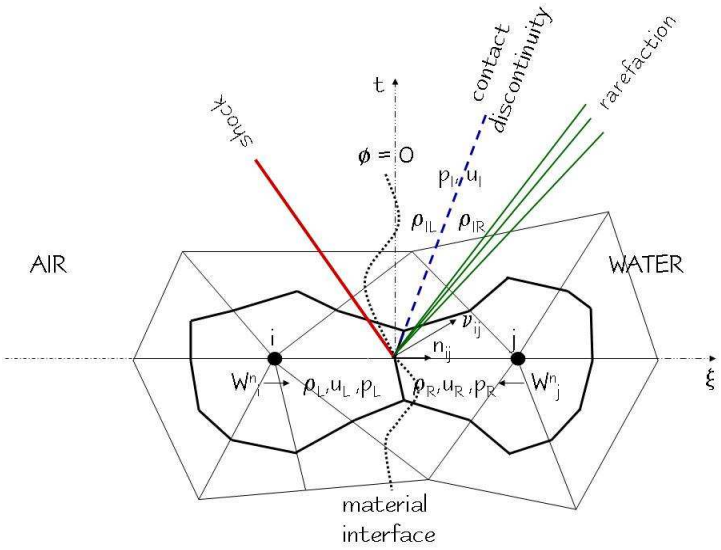


Fig. 2. Illustration of the local, one-dimensional, two-phase Riemann problem on a two-dimensional grid. (Subscripts $I$, $L$, and $R$ denote the material interface and the media at its left and right sides, respectively. $n_{ij}$ denotes the normal to the control volume at the face between nodes $i$ and $j$. $\nu_{ij}$ denotes the normal to the material interface at the same point).

example, when both media on the left and right sides of the material interface are modeled as stiffened gases, the local Riemann problem can be written as

$$\begin{aligned} u_I &= \tfrac{1}{2}(u_L + u_R) \\ &\quad + \tfrac{1}{2}\left(\mathcal{R}_R(p_I; p_R, \rho_R) - \mathcal{R}_L(p_I; p_L, \rho_L)\right) \\ \mathcal{R}(p_I; u_L, p_L, \rho_L, u_R, p_R, \rho_R) &= \mathcal{R}_L(p_I; p_L, \rho_L) + \mathcal{R}_R(p_I; p_R, \rho_R) \\ &\quad + u_R - u_L = 0 \end{aligned} \tag{39}$$

where the subscripts $L$ and $R$ designate the left and right sides of the material interface, respectively, $\mathcal{R}_L$ and $\mathcal{R}_R$ are two vector functions that depend on the structure of the wave solution at the left and right sides of the contact discontinuity (see Fig. 2), and a ";" is used to separate the unknown variables from known quantities.

When both media on the left and right sides of the material interface are modeled by Tait's EOS, the Riemann problem can be written as

$$\begin{aligned} u_I &= \tfrac{1}{2}(u_L + u_R) \\ &\quad + \tfrac{1}{2}\left(\mathcal{R}_R^*(p_I;\rho_R) - \mathcal{R}_L^*(p_I;\rho_L)\right) \\ \mathcal{R}^*(p_I;u_L,\rho_L,u_R,\rho_R) &= \mathcal{R}_L^*(p_I;\rho_L) + \mathcal{R}_R^*(p_I;\rho_R) \\ &\quad + u_R - u_L = 0 \end{aligned} \tag{40}$$

where $\mathcal{R}_L^*$ and $\mathcal{R}_R^*$ are two vector functions that depend on the structure of the wave solution at the left and right sides of the contact discontinuity (see Fig. 2). Analytical expressions for $\mathcal{R}_L$, $\mathcal{R}_R$, $\mathcal{R}_L^*$, and $\mathcal{R}_R^*$ can be obtained from the Rankine-Hugoniot jump conditions for shocks and the isentropic relations for rarefactions [22]. For the sake of completeness, these are given in Section 5.1.1 — Section 5.1.4. Once Eq. (39) is solved for $p_I$ — for example, using Newton's method — the computation of other interface quantities such as $u_I$, $\rho_{I_L}$ and $\rho_{I_R}$ becomes straigthforward.

### *5.1.1 Shock wave relations for a stiffened gas*

The shock wave relations for a stiffened gas can be written in terms of the unknown value of the pressure at the material interface, $p_I$, as follows

$$\mathcal{R}_K(p_I;p_K,\rho_K) = \left(\sqrt{\frac{a_K}{\bar{p}_I + b_K}}\right)(p_I - p_K) \tag{41}$$

where the subscript $K$ designates either the medium at the left ($L$) or that at the right ($R$) of the material interface,

$$a_K = \frac{2}{(\gamma_K+1)\rho_K} \qquad b_K = \left(\frac{\gamma_K - 1}{\gamma_K + 1}\right)\bar{p}_K \qquad \bar{p}_K = p_K + \pi_K \tag{42}$$

and $\gamma_K$ and $\pi_K$ have been defined in Section 2.2.1 and correspond to the EOS on the $K$ side of the interface as indicated by the subscript.

The pressure derivative of $\mathcal{R}_K$ at the material interface is given by

$$\mathcal{R}_K'(p_I;p_K,\rho_K) = \frac{d\mathcal{R}_K}{dp_I} = -\frac{a_K}{2(\bar{p}_I + b_K)^2} \tag{43}$$

### 5.1.2 Expansion wave relations for a stiffened gas

The expansion wave relations for a stiffened gas are given by

$$\mathcal{R}_K(p_I; p_K, \rho_K) = \left(\frac{2c_K}{\gamma_K - 1}\right)\left(\left(\frac{\bar{p}_I}{\bar{p}_K}\right)^{\frac{\gamma_K - 1}{2\gamma_K}} - 1\right) \tag{44}$$

and

$$\mathcal{R}'_K(p_I; p_K, \rho_K) = \left(\frac{c_K}{\gamma_K \bar{p}_K^{\frac{\gamma_K - 1}{2\gamma_K}}}\right)\left(\frac{\bar{p}_I}{\bar{p}_K}\right)^{-\left(\frac{\gamma_K + 1}{2\gamma_K}\right)} \tag{45}$$

where $c_K$ denotes as before the speed of sound (see Eq. (11)) on the side $K$ of the interface.

### 5.1.3 Shock wave relations for Tait's EOS

For Tait's EOS, the shock wave relations can be written as

$$\mathcal{R}^*_K(p_I; \rho_K) = \sqrt{\frac{(p_I - p_K)(\rho_I - \rho_K)}{\rho_K \rho_I}} \tag{46}$$

and

$$\begin{aligned}\mathcal{R}^{*'}_K(p_I; \rho_K) &= \frac{d\mathcal{R}^*_K}{dp_I} \\ &= \left(\frac{1}{2\mathcal{R}^*_K}\right)\left(\frac{\rho_I(\rho_I - \rho_K) + (p_I - p_K)\rho_K \rho'_I}{\rho_K \rho_I^2}\right)\end{aligned} \tag{47}$$

where

$$\rho_I = \left(\frac{p_I - \eta_K}{\alpha_K}\right)^{\beta_K{}^{-1}} \qquad \rho'_I = \frac{\rho_I^{1-\beta_K}}{\alpha_K \beta_K} \qquad p_K = \eta_K + \alpha_K {\rho_K}^{\beta_K} \tag{48}$$

$\alpha_K$ and $\beta_K$ and $\eta_K$ have been defined in Section 2.2.2 and correspond to the EOS on the side $K$ of the interface.

### 5.1.4 Expansion wave relations for Tait's EOS

On the other hand, the expansion wave relations governing a medium modeled by Tait's EOS are given by

$$\mathcal{R}^*_K(p_I;\rho_K) = \left(\frac{2c_K}{\beta_K - 1}\right)\left(\left(\frac{\bar{p}_I}{\bar{p}_K}\right)^{\frac{\beta_K-1}{2\beta_K}} - 1\right) \tag{49}$$

and

$$\mathcal{R}^{*'}_K(p_I;\rho_K) = \frac{c_K}{\beta_K \bar{p}_K}\left(\frac{\bar{p}_I}{\bar{p}_K}\right)^{-\frac{\beta_K+1}{2\beta_K}} \tag{50}$$

### 5.1.5 Local solution by Newton's method

The application of Newton's method to the solution of the local non-linear equation (39) for the interface pressure, $p_I$, by Newton's method generates the following sequence of iterate values of $p_I$

$$p_I^{(m+1)} = p_I^{(m)} - \frac{\mathcal{R}(p_I^{(m)})}{\mathcal{R}'(p_I^{(m)})} \tag{51}$$

where $m$ designates the Newton iteration. In this work, convergence of the sequence (51) is declared when

$$\frac{2|p_I^{(m+1)} - p_I^{(m)}|}{p_I^{(m+1)} + p_I^{(m)}} < \epsilon \tag{52}$$

where $\epsilon$ is a specified tolerance.

## 5.2 Implementational details

For the purpose of solving the Riemann problem (38), the intersection of the instantaneous position of the material interface and the dual CFD grid is assumed to coincide with the intersection of the boundaries of the control volumes and the edges of the original CFD grid (see Fig. 2). This faciliates the computation of the interfacial fluxes but raises the issue of which normal to use in this computation: that to the material interface or that to the corresponding face of the control volume, since both are available but are different except for one-dimensional problems. In order to remain consistent with the principles of

the finite volume method, the normal to the face of the control volume, $n_{ij}$, is always chosen here for computing a flux $\Phi_{ij}$. However, in order to capture the physics of the given multi-fluid problem, the normal to the material interface is chosen for computing the input and output entities of the Riemann solver (see Fig. 2). This normal between two connected grid points $i$ and $j$ on both sides of the material interface is evaluated using the gradient of the level-set function

$$\nu_{ij} = \nabla\phi_{ij} \approx \frac{1}{2}(\nabla\phi_i + \nabla\phi_j) \tag{53}$$

where a least square technique [23] is used to compute the nodal gradient $\nabla\phi_k$, $k = i, j$.

### *5.3 Summary: the GFMP-ERS*

The generalized GFMP described in Section 4 and equipped as described in the previous subsections with an exact, local, one-dimensional, two-phase Riemann solver is referred to in the remainder of this paper as the GFMP-ERS (for GFMP with Exact Riemann Solver). When equipped with the forward Euler time-integrator, this multi-fluid method can be summarized as follows.

(1) Capture the material interface by checking the product of the values of the level-set function $\phi$ at vertices $i$ and $j$ of edge $i$-$j$. A positive value indicates that edge $i$-$j$ does not cross the material interface, in which case the numerical flux function is computed as usual. On the other hand, a negative value indicates that this edge crosses the material interface, in which case two different fluxes are computed after a local, one-dimensional, two-phase Riemann problem along this edge is solved

exactly. In algorithmic words, this can be written as follows:

$$
\begin{aligned}
&\text{If } \phi_i^n \times \phi_j^n > 0,\\
&\qquad \Phi_{ij} = \text{Roe}\left(W_i^n, W_j^n, (\text{EOS}_i = \text{EOS}_j), n_{ij}\right)\\
&\text{If } \phi_i^n \times \phi_j^n \leq 0,\\
&\quad - \text{ extract the density } \rho_k^n, \text{ the velocity vector } u_k^n, \text{ and the pressure } p_k^n\\
&\qquad \text{from } W_k^n, k = i, j\\
&\quad - \text{ decompose each of the velocity vectors } u_k^n, k = i, j, \text{into a}\\
&\qquad \text{normal component } u_{\nu_{ij_k}}^n \text{ and a tangential component } u_k^n - u_{\nu_{ij_k}}^n \nu_{ij}\\
&\qquad \text{where } \nu_{ij} = \frac{\nabla\phi_{ij}}{|\nabla\phi_{ij}|} \text{ is computed using Eq. (53)}\\
&\quad - \text{ solve exactly the two-phase Riemann problem (38) along the edge } i-j \qquad (54)\\
&\qquad \text{using } \rho_k^n, u_{\nu_{ij_k}}^n, \text{ and } p_k^n, k = i, j \text{ as inputs and compute } \rho_{I_L}^n, \rho_{I_R}^n, u_I^n \text{ and } p_I^n\\
&\qquad \text{where } \rho_{I_L}, \rho_{I_R}, u_I, \text{ and } p_I \text{ have the same meaning as in Fig. 2}\\
&\quad - \text{ reconstruct the velocity vectors at both nodes } i \text{ and } j \text{ as}\\
&\qquad u_k^{\mathcal{R}^n} = u_k^n - u_{\nu_{ij_k}}^n \nu_{ij} + u_I^n \nu_{ij}, k = i, j\\
&\quad - \text{ construct } W_i^{\mathcal{R}^n} \text{and } W_j^{\mathcal{R}^n}\\
&\quad - \text{ compute the two fluxes}\\
&\qquad \Phi_{ij} = \text{Roe}\left(W_i^n, W_i^{\mathcal{R}^n}, \text{EOS}_i, n_{ij}\right)\\
&\qquad \Phi_{ji} = \textit{Roe}\left(W_j^{\mathcal{R}^n}, W_j^n, \text{EOS}_j, n_{ij}\right)
\end{aligned}
$$

(2) Time-advance the solution of the multi-fluid problem to compute a temporary value $\widetilde{W}_i^{n+1}$ of $W_i^{n+1}$

$$\widetilde{W}_i^{n+1} = W_i^n - \Delta t \sum_{j\in\kappa(i)} \text{mes}(\partial C_{ij})\Phi_{ij}(W_i^n, W_j^n, n_{ij}) \qquad (55)$$

(3) Using the value of the level-set function at time $t^n$, $\phi^n$, unpack the conservative fluid state vector $\widetilde{W}^{n+1}$ — that is, convert it to a vector $V^{n+1}$ of primitive variables to obtain $\tilde{\rho}^{n+1}$ and $\tilde{u}^{n+1}$

$$\widetilde{W}^{n+1} \xrightarrow{\phi^n} V^{n+1} \longrightarrow (\tilde{\rho}^{n+1}, \tilde{u}^{n+1}) \qquad (56)$$

(4) Compute $\phi^{n+1}$ by time-advancing the solution of the level-set equation (24) using the forward Euler scheme and the values of $\tilde{\rho}^{n+1}$ and $\tilde{u}^{n+1}$ stored in $V^{n+1}$

$$(\phi^n, \tilde{\rho}^{n+1}, \tilde{u}^{n+1}) \longrightarrow \phi^{n+1} \qquad (57)$$

(5) For each node $k$,

if $\phi_k^{n+1}\phi_k^n \geq 0$, pack $V_k^{n+1}$ using $\phi_k^{n+1}$: $V_k^{n+1} \xrightarrow{\phi_k^{n+1}} W_k^{n+1}$
if $\phi_k^{n+1}\phi_k^n < 0$, set $W_k^{n+1} = W_k^{\mathcal{R}^n}$

*REMARK 4.* The extension of the GFMP-ERS summarized above to a higher-order, explicit or implicit, one-step time-integrator is straightforward. Essentially, the desired one-step time-integrator is introduced in Step (2) outlined above and all other steps of this multi-fluid method are kept unchanged.

## 6 Extension to higher-order multi-step time-integrators

Extending the GFMP-ERS summarized in Section 5.3 to a higher-order, explicit or implicit, $k$-step time-integrator requires paying special attention to Step (3) of this method (see Section 5.3) — that is, the unpacking of the conservative fluid state vector $\widetilde{W}^{n+1}$. Straightforward extensions turned out to be numerically unstable. On the other hand, the following is a proposed extension which achieved excellent results for a large number of different multi-fluid problems benchmarked by the authors:

(1) Capture the material interface and compute the numerical fluxes

If $\phi_i^n \times \phi_j^n > 0$,

$$\Phi_{ij} = \text{Roe}\left(W_i^n, W_j^n, (\text{EOS}_i = \text{EOS}_j), n_{ij}\right)$$

If $\phi_i^n \times \phi_j^n \leq 0$,

- extract the density $\rho_k^n$, the velocity vector $u_k^n$, and the pressure $p_k^n$ from $W_k^n, k = i, j$
- decompose each of the velocity vectors $u_k^n, k = i, j,$ into a normal component $u_{\nu_{ij_k}}^n$ and a tangential component $u_k^n - u_{\nu_{ij_k}}^n \nu_{ij}$ where $\nu_{ij} = \dfrac{\nabla\phi_{ij}}{|\nabla\phi_{ij}|}$
- solve exactly the two-phase Riemann problem (38) along the edge $i - j$ using $\rho_k^n, u_{\nu_{ij_k}}^n,$ and $p_k^n, k = i, j$ as inputs and compute $\rho_{I_L}^n, \rho_{I_R}^n, u_I^n$ and $p_I^n$ where $\rho_{I_L}, \rho_{I_R}, u_I,$ and $p_I$ have the same meaning as in Fig. 2 (58)
- reconstruct the velocity vectors at both nodes $i$ and $j$ as $u_k^{\mathcal{R}^n} = u_k^n - u_{\nu_{ij_k}}^n \nu_{ij} + u_I^n \nu_{ij}, k = i, j$
- construct $W_i^{\mathcal{R}^n}$ and $W_j^{\mathcal{R}^n}$
- compute the two fluxes

$$\Phi_{ij} = \text{Roe}\left(W_i^n, W_i^{\mathcal{R}^n}, \text{EOS}_i, n_{ij}\right)$$
$$\Phi_{ji} = \textit{Roe}\left(W_j^{\mathcal{R}^n}, W_j^n, \text{EOS}_j, n_{ij}\right)$$

(2) Compute a temporary value $\widetilde{W}_i^{n+1}$ of $W_i^{n+1}$ by time-advancing the solution of the multi-fluid problem using the chosen higher-order, $k$-step time-integrator

$$(W^{n-k+1}, \ldots, W^{n-1}, W^n) \longrightarrow \widetilde{W}^{n+1} \qquad \text{(explicit case)} \quad (59)$$
$$(W^{n-k+1}, \ldots, W^{n-1}, W^n, W^{n+1}) \longrightarrow \widetilde{W}^{n+1} \qquad \text{(implicit case)} \quad (60)$$

(3) Using the values of the level-set function $\phi^{n-k+1}, \ldots, \phi^{n-1}$, and $\phi^n$, unpack the conservative fluid state vectors $W^{n-k+2}, \ldots, W^{n-1}$, and $\widetilde{W}^{n+1}$, respectively

$$\begin{aligned}
\widetilde{W}^{n+1} &\xrightarrow{\phi^n} V^{n+1} \xrightarrow{\phi^n} (\tilde{\rho}^{n+1}, \tilde{u}^{n+1}) \\
W^n &\xrightarrow{\phi^n} V^n \\
W^{n-1} &\xrightarrow{\phi^{n-1}} V^{n-1} \\
&\vdots
\end{aligned}$$

$$W^{n-k+2} \quad \stackrel{\phi^{n-k+2}}{\longrightarrow} \quad V^{n-k+2} \tag{61}$$

(4) Compute $\phi^{n+1}$ by time-advancing the solution of the level-set equation (24) using the chosen higher-order, $k$-step time-integrator and the values of $\tilde{\rho}^{n+1}$ and $\tilde{u}^{n+1}$ stored in $V^{n+1}$

$$(\phi^{n-k+1}, \ldots, \phi^{n-1}, \phi^n, \tilde{\rho}^{n+1}, \tilde{u}^{n+1}) \longrightarrow \phi^{n+1} \tag{62}$$

(5) Using the updated value of the level-set function $\phi^{n+1}$, pack appropriately all of $V^{n-k+2}, \ldots, V^{n-1}, V^n$, and $V^{n+1}$

$$\begin{array}{ccc}
V^{n+1} & \stackrel{\phi^{n+1}}{\longrightarrow} & W^{n+1} \\
\multicolumn{3}{c}{(\text{if } \phi_k^{n+1}\phi_k^n < 0, \text{ set } W_k^{n+1} = W_k^{\mathcal{R}^n})} \\
V^n & \stackrel{\phi^{n+1}}{\longrightarrow} & W^n \\
V^{n-1} & \stackrel{\phi^{n+1}}{\longrightarrow} & W^{n-1} \\
 & \vdots & \\
V^{n-k+2} & \stackrel{\phi^{n+1}}{\longrightarrow} & W^{n-k+2}
\end{array} \tag{63}$$

Note that at each time-station $t^{n+1}$, Step (5) of the above GFMP-ERS not only constructs the solution at $t^{n+1}$, $W^{n+1}$, but also re-evaluates the solutions at the previous $k$ time-stations.

## 7 Applications and performance assessments

First, a series of one-dimensional two-phase flow problems in a shock tube are considered to illustrate the behavior and performance of the various methods and extensions presented in this paper. More specifically, the shock tube is assumed to have a unit length in the $x$ direction. It contains two different fluids that are initially at rest and separated by a thin membrane. At $t = 0$, the two-phase flow is generated by the bursting of the membrane. This flow is one-dimensional, but all calculations are performed on a three-dimensional unstructured mesh with either 201 or 801 grid points along the $x$ direction.

Next, the potential of the GFMP-ERS for the solution of realistic, three-dimensional, multi-fluid problems is demonstrated with the simulation of the growth, collapse, and rebound of an underwater bubble and the comparison of the obtained numerical results to experimental data.

Various single- and multi-step time-integrators are considered but in all cases, the governing Euler and level-set equations are semi-discretized by the second-order FV scheme outlined in Section 2.4.

### *7.1.1 Perfect gas - perfect gas computations*

The purpose of this first example problem, which was also considered in [15], is to illustrate the extension of the basic GFMP to a multi-step time-integrator. The two fluids are in this case perfect gases and the membrane is positioned at $x = 0.5$. The initial states of the gases at the left and right sides of the membrane and the constants of their EOSs are

$$\begin{aligned} &\rho_L = 1.0 \qquad u_L = 0 \quad p_L = 1.0 \quad \gamma_L = 1.4 \\ &\qquad\qquad \text{and} \\ &\rho_R = 0.125 \quad u_R = 0 \quad p_R = 0.1 \quad \gamma_R = 1.2 \end{aligned} \tag{64}$$

and therefore

$$\frac{\rho_L}{\rho_R} = 8 \tag{65}$$

The spatial discretization is performed with 201 grid points along the length of the tube. Two computations are performed: one using the GFMP with a 4th-order Runge-Kutta (RK4) time-integrator operating at CFL = 0.8, and one using the GFMP with a three-point backward difference implicit (3PBDF) time-integrator operating at CFL = 8.0. Fig. 3 which reports the numerical results at $t = 0.2$ and compares them to the analytical solution shows that the GFMP equipped with the RK4 reproduces correctly the variations of the density, velocity, and pressure along the tube. The GFMP equipped with the 3PBDF implicit scheme is also reported to correctly reproduce the variations of these quantities, except for the small bumps it introduces in the pressure and velocity at the shock. These bumps are not due to the GFMP but to the second-order time-accurate 3PBDF operating at CFL = 8.0. At lower CFL values, the bumps become even smaller. In any case, the computed solutions at the material interface are shown to be in very good agreement with the analytical solution.

### *7.1.2 Perfect gas - perfact gas problem with a reflectionless shock*

The shock tube problem considered here was also discussed in [14, 16]. In this case, the membrane is positioned at $x = 0.2$ and separates two perfect gases

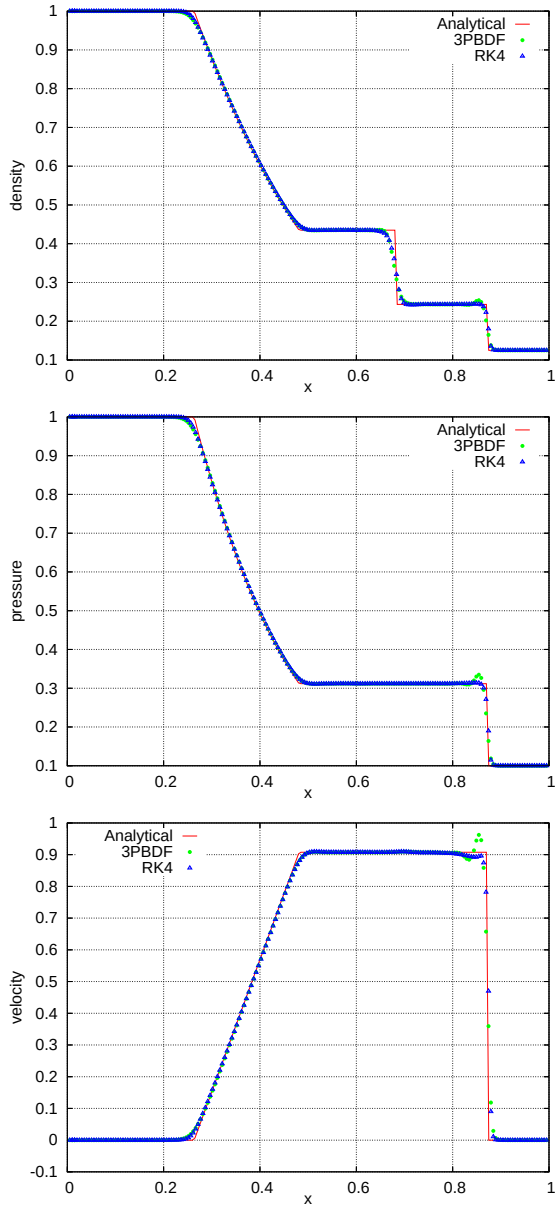


Fig. 3. Perfect gas - perfect gas: variations of the density, pressure, and velocity at $t = 0.2$ along the length of the shock-tube (GFMP, $\Delta x = 1/201$)

whose initial states and EOS constants are

$$\begin{aligned} &\rho_L = 3.2 \quad u_L = 9.43499279 \quad p_L = 100.0 \quad \gamma_L = 5/3 \\ &\qquad\qquad \text{and} \\ &\rho_R = 1.0 \quad u_R = 0 \qquad\qquad p_R = 1.0 \quad \gamma_R = 1.2 \end{aligned} \tag{66}$$

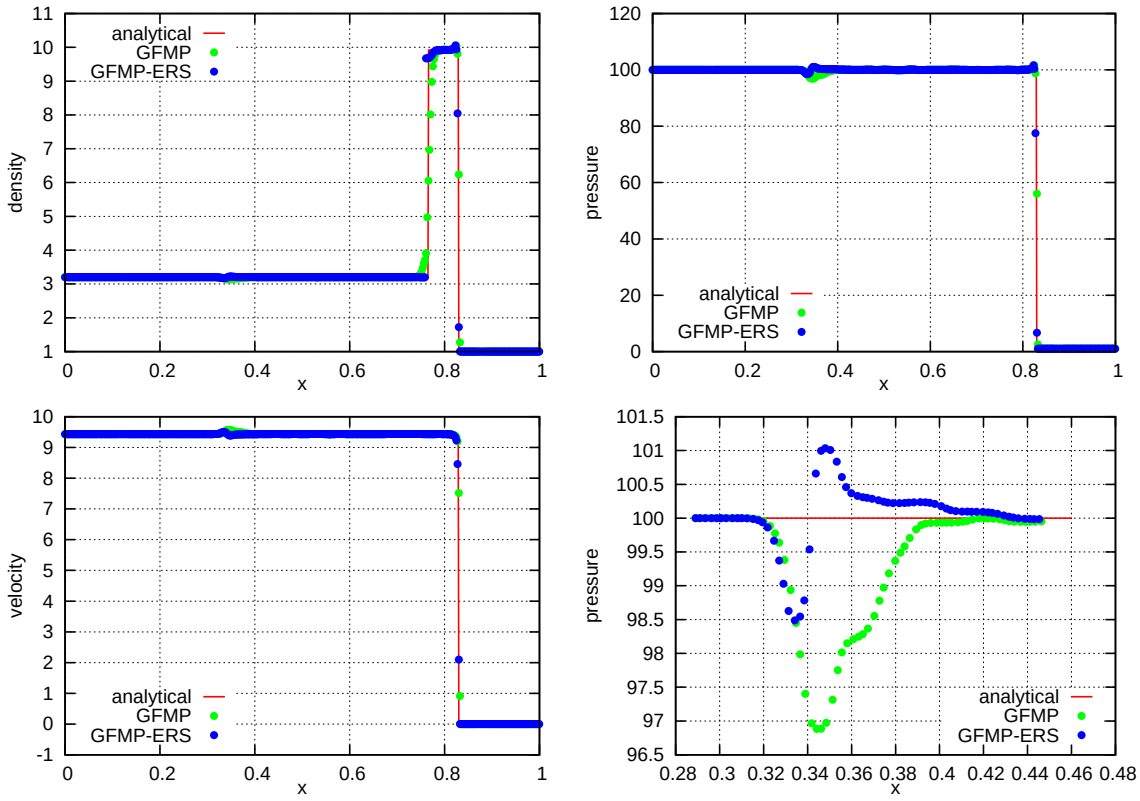


Fig. 4. Reflectionless gaseous shock problem: variations of the density, pressure, and velocity at $t = 0.06$ along the shock tube (GFMP-ERS, $\Delta x = 1/201$) — Zoom on the main oscillation is shown for the pressure field at the bottom right part of the figure

Hence,

$$\frac{\rho_L}{\rho_R} = 3.2 \tag{67}$$

This problem is easier than the previous one from the density ratio viewpoint. However, it is more challenging than the previous problem from the viewpoint that its analytical solution consists of a shock wave and a contact discontinuity only that propagate to the right side of the material interface. Therefore, the difficulty of this shock tube problem stems from the absence of any wave travelling to the left along the shock tube. Most if not all computational algorithms applied to the solution of this problem can be expected to generate a non-physical reflection at the material interface.

Two numerical computations are performed on the mesh with 201 grid points in the $x$ direction: one using the GFMP and one using the GFMP-ERS. In both cases, the RK4 time-integrator is chosen and the CFL number is set to 0.8. The results obtained at $t = 0.06$ are reported in Fig. 4. Some small amplitude oscillations can be observed in the computed solutions. They are due

to the reflection at the material interface. As mentioned in [14], this spurious reflection is difficult to remove. The amplitude of the main oscillation exhibited in the GFMP-ERS solution is shown to be twice as small as that exhibited by the GFMP solution. In any case, the numerical solution delivered here by the GFMP-ERS appears to be more accurate than that reported in [14] and comparable to that reported in [16].

### *7.1.3 Perfect gas - stiffened gas system with a density ratio of 20*

The purpose of this third example problem which, was also considered in [15], is to demonstrate the improvement delivered by the GFMP-ERS over the basic GFMP. The membrane is positioned in this case at $x = 0.3$. The fluid at the left side of this membrane is a perfect gas. The fluid at the right side of the membrane is water and is modeled as a stiffened gas. The initial states of both fluids and the constants of their EOSs are

$$\begin{aligned} &\rho_L = 50.0 \qquad u_L = 0 \quad p_L = 10^5 \quad \gamma_L = 1.4 \\ &\qquad\qquad\qquad \text{and} \\ &\rho_R = 1000.0 \quad u_R = 0 \quad p_R = 10^9 \quad \gamma_R = 4.4 \quad \pi_R = 6.0 \times 10^8 \end{aligned} \tag{68}$$

and therefore

$$\frac{\rho_L}{\rho_R} = 20 \tag{69}$$

Two meshes are generated: one with 201 grid points along the $x$ direction, and one with 801 grid points along this direction.

On each mesh, two computations are performed: the first one using the GFMP and the second one using the GFMP-ERS. In both cases, the time-discretization is performed using the RK4 time-integrator and the CFL number is set to 0.8. The results at $t = 2.4 \times 10^{-4}$ are reported in Fig. 5 ($\Delta x = 1/201$) and Fig. 6 ($\Delta x = 1/801$). On the mesh with 801 grid points in the $x$ direction, both of the GFMP and GFMP-ERS perform well. However, the GFMP-ERS predicts a sharper density jump close to the material interface. On the coarser mesh with 201 grid points in the $x$ direction, only the GFMP-ERS captures the density plateau between the shock and the contact surface. This underscores the superior performance of the GFMP-ERS and its potential for achieving accuracy at practical mesh resolution.

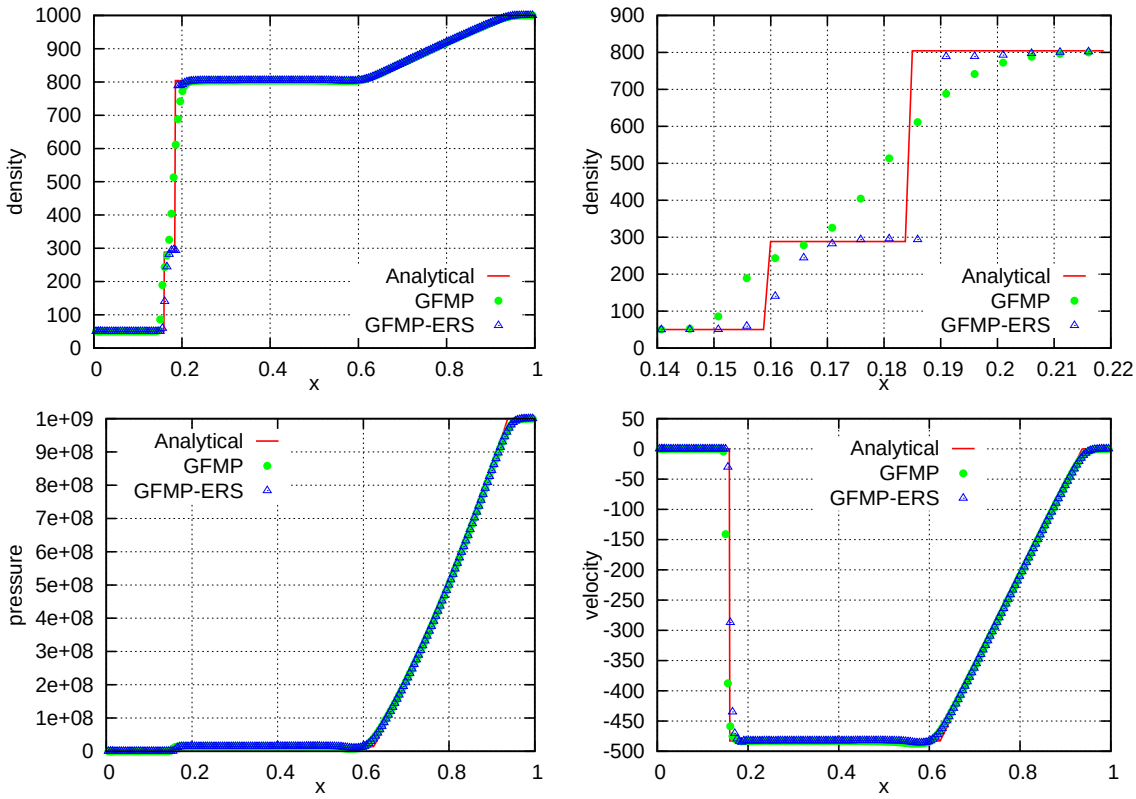


Fig. 5. Perfect gas - stiffened gas: variations of the density, pressure, and velocity at $t = 2.4 \times 10^{-4}$ along the length of the shock-tube ($\Delta x = 1/201$) — Zoom on the "plateau" region is shown for the density field at the top right part of the figure

### *7.1.4 Gas - water system with a density ratio of 1,000*

Here, a stiffer model problem with two differents EOSs for modeling gas and water is considered. The membrane is positioned at $x = 0.3$. The initial conditions for both fluid media are more relevant than previously to underwater explosions and implosions where the ratio of densities at the material interface is approximately 1,000 as they are set to

$$\begin{aligned} \rho_L &= 1.0 \qquad & u_L &= 0 \qquad & p_L &= 10^5 \\ \rho_R &= 1000.0 \qquad & u_R &= 0 \qquad & p_R &= 10^7 \end{aligned} \tag{70}$$

and therefore

$$\frac{\rho_L}{\rho_R} = 1000 \tag{71}$$

Two computations are performed on the mesh with 201 points in the $x$ direction. In the first one, the water is modeled by Tait's EOS with $k_1 = 2.07 \times 10^9$, $k_2 = 7.15$, $p_0 = p_R$ and $\rho_0 = \rho_R$. In the second one, the water is modeled by

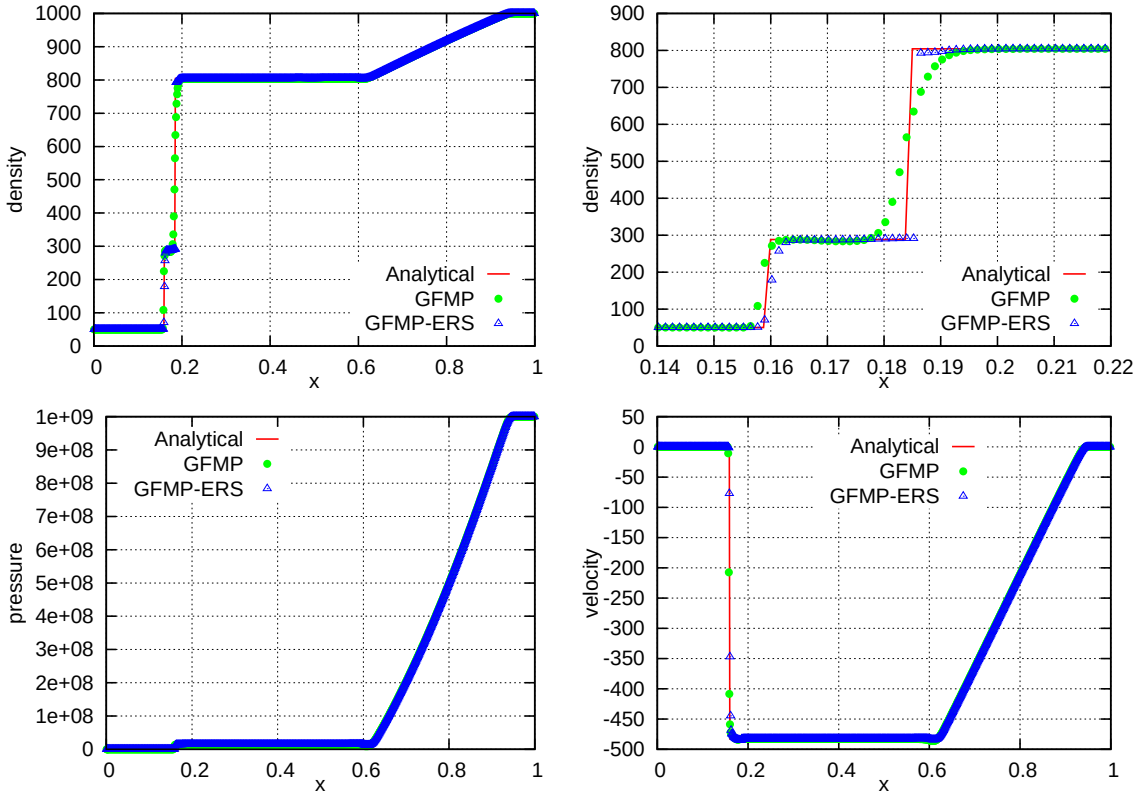


Fig. 6. Perfect gas - stiffened gas: variations of the density, pressure, and velocity at $t = 2.4 \times 10^{-4}$ along the length of the shock-tube ($\Delta x = 1/801$) — Zoom on the "plateau" region is shown for the density field at the top right part of the figure

the stiffened gas EOS with $\gamma = 7.15$ and $\pi = \dfrac{k_1}{k_2} = \dfrac{2.07 \times 10^9}{7.15}$ In both cases, the air is modeled by the perfect gas EOS with $\gamma = 1.4$.

For this problem, the GFMP fails early on in the simulation because of the presence of a strong contact discontinuity. On the other hand, the GFMP-ERS equippped with the RK4 time-integrator operating at CFL $= 0.8$ delivers in both cases excellent results, as shown in Fig. 7 for $t = 4 \times 10^{-4}$.

Note that the analytical solution of the above problem is the same whether the water is modeled as a barotropic fluid or as a stiffened gas. The structure of this solution consists of a shock wave travelling in the air, a contact discontinuity, and a rarefaction wave propagating in the water. This is consistent with both the physics of the problem and the chosen models. Indeed, as shown in REMARK 3 (see Section 2.2.2), a stiffened gas behaves during an isentropic transformation like a barotropic fluid modeled by Tait's EOS.

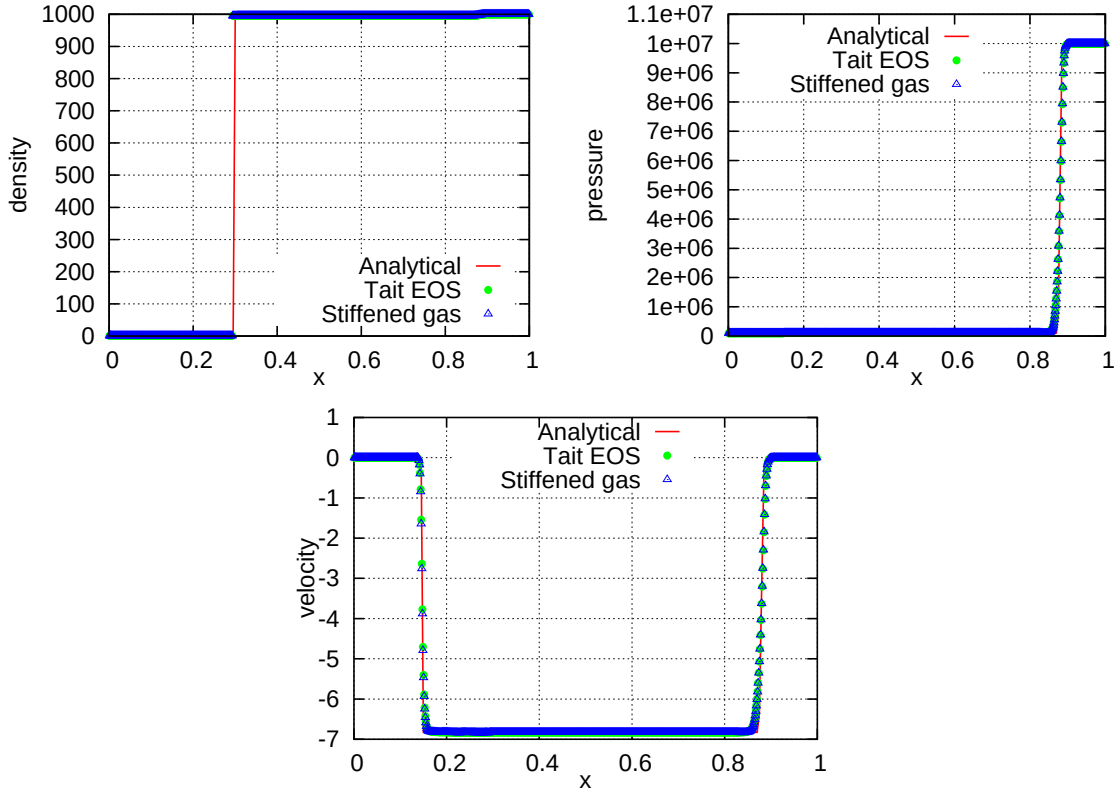


Fig. 7. Air (perfect gas) - water (barotropic fluid/stiffened gas): variations of the density, pressure, and velocity at $t = 4.0 \times 10^{-4}$ along the length of the shock-tube (GFMP-ERS, $\Delta x = 1/201$)

### *7.2 Pulsating underwater explosion bubble*

The GFMP-ERS was implemented in the AERO-F flow code [24, 25] and applied to the numerical simulation of the underwater explosion of a charge of $2.27 \times 10^{-1}$ kg of Tetryl placed at 91.4 m below water. The corresponding experiment is described in [26] which also contains the test data and results. The explosion bubble contains a highly pressurized gaseous mixture that is modeled here by the perfect gas EOS with a specific heat ratio $\gamma_g = 1.4$. It is assumed to sit initially in still water that is modeled by the stiffened gas EOS with $\gamma_w = 4.4$ and $\pi_w = 5.0 \times 10^8$ Pa. Here, the subscripts $g$ and $w$ designate gas and water, respectively. The initial conditions are reconstructed from the test data reported in [26] as follows. The gas is assumed to reside inside a spherical bubble of radius $r_0 = 3.0 \times 10^{-2}$ m and to have the following initial conditions: $\rho_g^0 = 1.2 \times 10^3$ kg/m$^3$, $p_g^0 = 5.0 \times 10^7$ Pa, and $u_g^0 = 0$ m/s. The initial and far-field conditions of the surrounding water are set to $\rho_w^0 = 10^3$ kg/m$^3$, $p_w^0 = 10^6$ Pa, and $u_w^0 = 0$ m/s.

The computational domain is chosen to be the sphere of radius $r_{\text{inf}} = 100 \times r_0 = 3$ m. It is discretized in 7,203,307 tetrahedra and 1,340,213 grid points

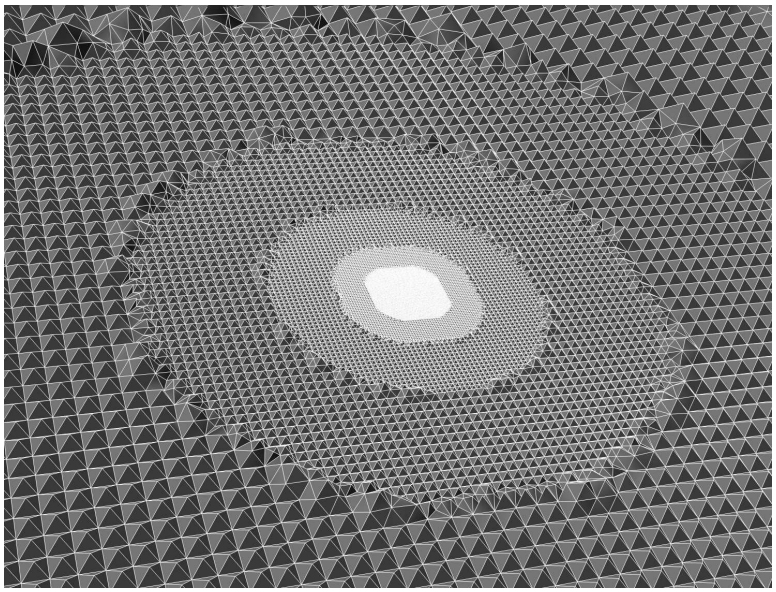

Fig. 8. Simulation of the dynamics of a pulsating explosion bubble: slice view of the three-dimensional unstructured mesh

organized in several layers. These are obtained by geometrical stretching of the edges from the center of the sphere to its external surface using a stretching ratio of 1.2. Hence, the mesh is finest near the initial position of the bubble. A slice of this unstructured mesh is shown in Fig 8.

The numerical simulation is performed with the GFMP-ERS and the RK4 time-integrator operating at CFL = 2.2. Fig. 9 reports the time-history of the simulated mean radius of the bubble and compares it to that measured experimentally. The GFMP-ERS computation is shown to track the mean radius of the bubble during its growth, collapse, and rebound remarkably accurately during the first period and a half of the oscillation. During the second and subsequent periods of oscillation, the deviation of the numerical results from their corresponding experimental data can be attributed to a variety of reasons, including the following ones:

(1) The conversion of the solid explosive into a gaseous mixture is a simplified approach for modeling the outcome of the explosion.
(2) Determining adequate initial conditions for this gaseous mixture is a real challenge.
(3) Harris' remark by email
(4) Since the bubble does not retain a spherical shape during its collapse, extracting a mean radius from each of the simulation and experimental data introduces discrepencies in the comparison between these two sets of results.
(5) In [17], the experimentalists acknowledge that around the end of the collapse time, tracking the radius of the bubble becomes a challenging task and generates data of questionable reliability.

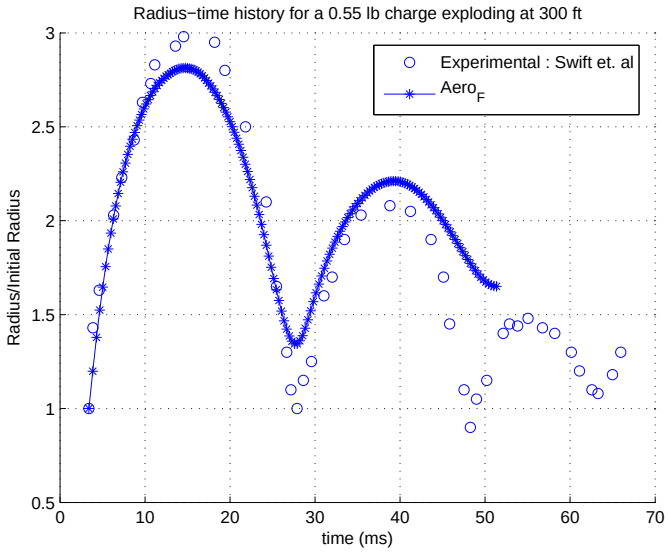


Fig. 9. Simulation of the dynamics of a pulsating explosion bubble: comparison of the simulated mean radius time-history using the GFMP-ERS and the corresponding experimental data

## 8 Conclusions

The ghost fluid method for the poor (GFMP) was developed for the solution of two-phase flow problems using the stiffened gas equation of state and one-step time-discretization algorithms. It is a nearly conservative and computationally efficient method. However, it cannot handle problems with strong contact discontinuities, particularly at practical mesh resolutions. As such, it is not applicable to the solution of two-phase air/water applications such as underwater explosions and implosions. In this paper, the GFMP was generalized to arbitrary equations of state and multi-fluid problems with multiple equations of state. It was also extended to higher-order multi-step time-integrators. In order to make it robust with respect to a large discontinuity of the density and a strong pressure jump at the material interface, it was equipped with an exact, local, one-dimensional, two-phase Riemann solver for computing the interfacial fluxes. Consequently, the resulting multi-fluid method was labeled the GFMP-ERS. As the original GFMP, the GFMP-ERS is computationally efficient and nearly conservative. Its application in this paper to the solution of various shock tube problems with stiffened gas and barotropic equations of state has revealed a superior performance in the presence of large density and pressure jumps. Also, its successful application to the simulation of the growth, collapse, and rebound of an underwater bubble for which experimental data is available has highlighted its potential for the analysis of underwater implosion and explosion problems.

## References


[1] S.K. Godunov. A finite difference method for the computation of discontinuous solutions of the equations of fluid dynamics. *Mat. Sb.*, 47:357, 1959.

[2] B. VanLeer. Towards the ultimate conservative difference scheme, v. a second order sequel to godunov's method. *J. Comput. Phys.*, 32:1011, 1979.

[3] B. VanLeer. Flux-vector splitting for the euler equations. Technical report, Lecture Notes in Physics, Springer, Berlin, 1982.

[4] R. Abgrall. Generalization of the Roe scheme for the computation of mixture of perfect gases. *La Recherche Aerospatiale*, 6:31, 1988.

[5] S. Karni. Multicomponent flow calculations by a consistent primitive algorithm. *J. Comput. Phys.*, 112:31, 1994.

[6] C.W. Hirt and B.D. Nichols. Volume of fluid method (vof) for dynamics of free boundaries. *J. Comput. Phys.*, 31:201, 1981.

[7] B. Merriman R. Fedkiw, T. Aslam and S. Osher. A non-oscillatory eulerian approach to interfaces in multimaterial flows (the ghost fluid method). *J. Comput. Phys.*, 152:457, 1999.

[8] R. Abgrall. How to prevent pressure oscillation in multicomponent flow calculation : A quasi-conservative approach. *J. Comput. Phys.*, 125:150, 1996.

[9] S. Karni. Hybrid multifluid algorithms. *SIAM J. Sci. Comp.*, 17:1019, 1996.

[10] R. Saurel and R. Abgrall. A simple method for compressible multifluid flows. *SIAM J. Sci. Comp.*, 21(3):1115, 1999.

[11] A. Marquina R. Fedkiw and B. Merriman. An isobaric fix for the overheating problem in multimaterial compressible flows. *J. Comput. Phys.*, 148:545, 1999.

[12] Y. Liu J. Glimm, X.L. Li and N. Zhao. Conservative front tracking and level set algorithms. *PNAS*, 98(25):14198, 2001.

[13] F. Gibou D. Nguyen and R. Fedkiw. A fully conservative ghost fluid method & stiff detonation waves. Technical report, 12th Int. Detonation Symposium, San Diego, CA, 2002.

[14] B.C. Khoo T.G. Liu and C.W. Wang. The ghost fluid method for compressible gas-water simulations. *J. Comput. Phys.*, 204:193, 2005.

[15] R. Abgrall and S. Karni. Compressible multifluid flows. *J. Comput. Phys.*, 169(2):594, 2001.

[16] C.W. Wang, T.G. Liu and B.C. Khoo. A real ghost fluid method for the simulation of mutimedium compressible flow. *SIAM J. Sci. Comput.*,28:278, 2006.

[17] R.H. Cole. *UnderWater Explosions*. Dover, New York, 1965.

[18] J.B. Keller and I.I. Kolodner. Damping of underwater explosion bubbles. *J. Appl. Phys.*, 27(10):1152, 1956.

[19] T.L. Geers R.S. Lagumbay and O.V. Vasilyev. Numerical modeling and

simulation of an underwater explosion bubble. Technical report, American Physical Society, Chicago, November 2005.

[20] P. Smerka M. Sussman and S. Osher. A level-set approach to computing solutions to two phase incompressible flows. *J. Comput. Phys.*, 114:146, 1994.

[21] P.L. Roe. Approximate riemann solver, parameters vectors and difference schemes. *J. Comput. Phys.*, 43:357, 1981.

[22] D.M. Causon M.J. Ivings and E.F. Toro. On riemann solvers for compressible liquids. *Int. J. for Num. Meth. in Fluids*, 28:395, 1998.

[23] A. Haselbacher and J. Blazek. Accurate and efficient discretization of Navier-Stokes equations on mixed grids. *AIAA J.*, 38:2094, 2000.

[24] P. Geuzaine, G. Brown, C. Harris and C. Farhat. Aeroelastic dynamic analysis of a full F-16 configuration for various flight conditions. *AIAA J.*, 41:363, 2003.

[25] C. Farhat, P. Geuzaine and G. Brown. Application of a three-field nonlinear fluid-structure formulation to the prediction of the aeroelastic parameters of an F-16 fighter. *Comput. & Fluids*, 32:3, 2003.

[26] E. Swift and J. Decius. Measurement of bubble pulse phenomena. *Navord Report 97-46*, 1946.