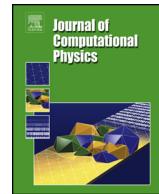




Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp



A seven-equation diffused interface method for resolved multiphase flows



Achyut Panchal^a, Spencer H. Bryngelson^b, Suresh Menon^{a,*}

^a School of Aerospace Engineering, Georgia Institute of Technology, Atlanta, GA, 30332, United States of America

^b School of Computational Science and Engineering, Georgia Institute of Technology, Atlanta, GA, 30332, United States of America

ARTICLE INFO

Article history:

Received 18 April 2022

Received in revised form 12 November 2022

Accepted 18 December 2022

Available online 28 December 2022

Keywords:

Multiphase flows

Diffused interface method

Seven-equation model

ABSTRACT

The seven-equation model is a compressible multiphase formulation that allows for phasic velocity and pressure disequilibrium. These equations are solved using a diffused interface method that models resolved multiphase flows. Novel extensions are proposed for including the effects of surface tension, viscosity, multi-species, and reactions. The allowed non-equilibrium of pressure in the seven-equation model provides numerical stability in strong shocks and allows for arbitrary and independent equations of states. Whereas, the allowed non-equilibrium of velocity allows the method to be useful even for unresolved particles where a finite-rate velocity relaxation is observed. A discrete equations method (DEM) models the fluxes. We show that even though stiff pressure- and velocity-relaxation solvers have been used, they are not necessarily needed for all the tests with DEM because the non-conservative fluxes are accurately modeled. An interface compression scheme controls the numerical diffusion of the interface, and its effects on the solution are discussed. Test cases are used to validate the computational method and demonstrate its applicability. They include multiphase shock tubes, shock propagation through a material interface and a dispersed particle curtain, a surface-tension-driven oscillating droplet, an accelerating droplet in a viscous medium, and shock-detonation interacting with a deforming droplet. Simulation results are compared against exact solutions and experiments when possible.

© 2022 Elsevier Inc. All rights reserved.

1. Introduction

Multiphase flows are characterized by more than one phase, each with its own material properties. Such flows are encountered in many situations, including liquid fuel-powered devices [1,2], solid explosives [3], pharmaceutical sprays [4], and more. The design and engineering of these applications require computational modeling. From this perspective, there are two sets of methods: resolved approaches where the multiphase entities (MPE) are fully-resolved on the computational grid [5,6], and dispersed approaches where the MPEs are unresolved on the grid and instead treated as point particles [7–10].

This work focuses on resolved multiphase flow modeling, which can be further segmented into sharp- or diffused-interface methods (SIM or DIM). The SIM naturally represents materials as infinitely sharp [11,12]. Specialized techniques are used to advect the interface, including the level-set method (LS) [11], the volume of fluid approach (VOF) [13], and

* Corresponding author.

E-mail addresses: apanchal7@gatech.edu (A. Panchal), suresh.menon@ae.gatech.edu (S. Menon).

the coupled LS-VOF method (CLSVOF) [14]. Example methods for handling jumps at the interface are continuum surface forces (CSF) [15] and ghost fluid methods (GFM) [16]. SIM are particularly attractive for their ease in specifying jump conditions and efficient interface tracking. However, they have historically been limited by low-speed flows, low density-viscosity ratios [17], and mass-conservation errors [11]. We note that some of these drawbacks are changing via bespoke methods [18–20].

Diffused interface methods (DIM) are hallmark by the Baer–Nunziato (BN) “seven-equation” formulation of a compressible and mixture conservative flow [6,21]. The equations follow from an averaging procedure to the phasic governing flow equations. The derived partial differential equation (PDE) solves for phasic volume fractions and phase-averaged quantities. This formulation evolves the conserved quantities on the grid, usually via numerical methods that allow for sharp gradients [22,23]. As a result, the interface naturally diffuses across multiple grid cells due to numerical diffusion [6]. Besides their conservation properties, DIMs also allow for the creation or destruction of interfaces, common during supercavitation and flashing [24–27].

The first numerical method for a single-component inviscid seven-equation model was developed by Saurel and Abgrall [28]. They used a two-step approach, where the two-phase equations were first solved using a hyperbolic solver, followed by a relaxation step to handle phase disequilibrium. For DIM, and particularly for the seven-equation model, modeling of the non-conservative terms have presented several challenges. Saurel and Abgrall [28] used a HLL Riemann solver and Andrianov and Warnecke [29] used an exact solution of a linearized Riemann solver for the conservative terms, while using the ‘free-stream’ condition to guide modeling of the non-conservative terms. Schwendeman et al. [30] developed an iterative Riemann solver and modeled the multiphase interface as of vanishingly small thickness with a smooth solution to model both the conservative and the non-conservative terms. Lochon et al. [31] used a HLLC Riemann solver but with an approximation for the two-phase contact. Taking a different route, Abgrall and Saurel [32] developed the discrete equations method (DEM), where instead of solving the two-phase problem Riemann problem with the phase volume-fraction as a variable, multiple Riemann sub-problems are solved with only pure phase phases as its left- and right-states. Modeling of both conservative and non-conservative terms naturally followed.

Even though the original work by Saurel and Abgrall [28] used a relaxation step to enforce pressure and velocity equilibrium between the phases, it was shown later that with the use DEM [32], since the non-conservative terms are computed accurately, the interphase exchange processes are accurately captured and the pressure- and the velocity-equilibrium can be maintained at the interface even without the relaxation step. Some other works, that computed the non-conservative terms accurately [30,31], also did not need a relaxation step, although these were limited to 1D shock-tube configurations.

After the initial efforts, however, reduced forms of the Baer–Nunziato equations were developed to relieve the otherwise needed stiff pressure- and velocity-relaxation procedures. These methods are broadly used today. Since these methods carry only a single-pressure or a single-velocity, the phase equilibrium (for pressure and velocity) is analytically enforced. For example, Kapila et al. [33] developed a reduced five-equation model with single velocity and pressure, and its use has been widespread [34–39]. This is partly due to the simplicity of including additional physical phenomena like surface tension [40], mass-transfer [27], viscous effects [41], flashing [42], multiple components [43], and cavitation [44]. The drawback to the five-equation approach is that a single-pressure requires establishing a mixture EOS with a non-monotonic sound speed [45]. This results in numerical instability near strong shocks, difficulty maintaining volume fraction-positivity, and inaccurate wave transmission through the interface [46]. The five-equation model by Allaire et al. [34] is relatively robust, but it cannot model the creation and destruction of the interface due to the absence of a volume fraction source term. A six-equation model was developed to resolve this, which used a single velocity but phase-independent pressures [46]. Extensions of this work include phase-change effects [47–50], viscous effects [51], and arbitrary numbers of components [43,52]. Further down the hierarchy of the reduced BN models [53], there are also four- [26,49,54,55] and three-equation [56,57] models, that, in addition to the pressure and the velocity equilibrium between the phases, also assume thermal, and both thermal and chemical equilibrium, respectively. However, their problems with handling large jumps in material properties and mixture EOS are well-known [28,57], and therefore they are not discussed here further.

Herein, we use the more extensive seven-equation model. Unlike the reduced models (e.g., [39]), the seven-equation model carries phase-independent pressures and velocities. Thus, both pressure and velocity non-equilibrium are allowed. For modeling resolved multiphase entities (MPE), compared to the well-established five-equation model, the benefits of carrying phase-independent pressures are the same as those of the six-equation model [46]. It can robustly represent strong shocks and large density ratios: $\rho^{(2)}/\rho^{(1)}$, $p^{(2)}/p^{(1)} \gg 1$. And furthermore, there is no need to formulate a mixture EOS, so one can use arbitrarily independent forms of the EOS for the phases. The seven-equation model also retains phase-independent velocities, which is not expected provide a clear benefit for modeling resolved multiphase flows, where the velocity equilibrium is supposed to be maintained at the interface, but it will allow representation of dispersed unresolved MPEs, where the velocities relax towards each other in a finite time via specified drag laws (e.g., [8]). Such a capability can be of interest for dense-to-dilute multiphase modeling, e.g., liquid fuel injection [58], multiphase explosions [59], etc., where it is not computationally feasible to resolve all the small-sized MPEs (size \sim nm– μ m), but in other dense regions of the flow, certain multiphase features (size \sim mm–m) have to be resolved (e.g., for modeling primary breakup). Various algorithms that consider a rule-based transition between the vastly different resolved and dispersed phase approaches exist [60–62], however, unlike those, the seven-equation model will allow to handle both resolved and dispersed consistently within the same numerical framework with only minor changes related to the modeling of the relaxation terms. Considering these points, the various features of SIM and DIM variants are summarized in Table 1.

Table 1

Various numerical methods for modeling resolved MPEs and their features. P: Possible; NP: Not possible; PNA: Possible, not attempted (prior to this work). **Bold text** refers to capabilities considered in this work.

	SIM	4-eq DIM	5-eq DIM	6-eq DIM	7-eq DIM
$\rho^{(2)}/\rho^{(1)} \gg 1, p^{(2)}/p^{(1)} \gg 1$	Limited [18–20,72]	Very limited [28]	Limited robustness [33]	P [46]	P [28,32]
Arbitrarily independent forms of EOS	P [16]	NP	NP	PNA	PNA
Viscous	P [72]	P [49,56]	P [41]	P [43]	PNA
Surface tension	P [73]	P [49]	P [40]	P [74]	P [65]
Flashing/transcritical	NP	P [26,55]	P [42,44]	P [47,48,75]	P [47]
Evaporation	P [18,76]	P [26,55]	P [27]	P [47,48,75]	P [47]
Dispersed	NP	NP	NP	NP	P [28]
Multi-species/reaction	P [18]	P [26,55]	PNA	PNA	PNA
Interface Compression	Not needed	PNA	Used [39,70,71]	Used [77]	PNA

After the initial development of the seven-equation model by Saurel and Abgrall [28], there have been relatively limited extensions and application use-cases [32,47,63–65]. This appears to be due to the complexity of formulating an efficient numerical method and the lack of a clear establishment of the benefit of using a seven-equation model over the other well-established reduced approaches. This work is a step towards addressing these. To demonstrate the unique potential of using the seven-equation model over the reduced models, a new test is considered. This contains unresolved particles in one part of the domain and the velocity non-equilibrium is required for accurate modeling of particle dispersion, whereas, a resolved multiphase interface is present in the other part of the domain that is modeled via DIM. Such modeling would not have been possible with the reduced five/six-equation models that enforce the velocity equilibrium analytically. To establish the use of the seven-equation model for modeling resolved MPEs, where the reduced models are already well-established, we start with the baseline DEM of Abgrall and Saurel [32] and show novel extensions to handle strong shocks and arbitrarily independent forms of EOS for the phases, viscous effects, surface tension, multiple species, and reactions within the same numerical framework. Here, the seven-equation model provides benefits over the five-equation model [33] by relieving the need to build a mixture EOS and due to the increased robustness. Finally, as noted earlier, the seven-equation DIM may require additional algorithms for maintaining velocity and pressure equilibrium at the resolved interface, i.e., stiff relaxation (SR) solver, and interface compression (IC) techniques may also be required to limit the numerical diffusion of the interface. Their need for the seven-equation model and for relatively complex configurations are not yet clear from the current literature, and it is evaluated in this work.

Particularly, some novel contributions of this work are as follows:

- **Velocity disequilibrium:** As noted earlier, to demonstrate the unique potential of the seven-equation model due to the allowed velocity disequilibrium, a new test-case is considered that contains both resolved and unresolved MPEs in the same domain. Additional improvements maybe required in the future for modeling dense-to-dilute flows, e.g., as noted by Saurel et al. [66], however, the objective here is limited to demonstrating the potential of the seven-equation model for such problems.
- **Pressure disequilibrium:** The developed computational framework can robustly deal with arbitrarily independent forms of the EOS for the phases and strong density and pressure jumps across the interface due to the allowed pressure non-equilibrium. This is demonstrated by using calorically perfect gas (CPG), thermally perfect gas (TPG), stiffened gas (SG), and Mie–Grüneisen (MG) EOS. These benefits are also expected of a six-equation model, and they are retained here.
- **Non-conservative and relaxation terms:** Various numerical methods have been developed for solving the BN PDE [28, 30–32,63–65]. Previous works that computed the non-conservative terms accurately [30–32] did not need a stiff relaxation solver as an additional step for maintaining pressure and velocity equilibrium at the interface for 1D shock-tube problems. This effect is further studied in this work for more complex problems. A spherical bubble collapse is also simulated, where maintaining the pressure equilibrium via the relaxation terms has been shown to be crucial for accurate modeling [67].
- **Surface tension** is required for modeling droplet breakup and deformation. Past works included it in the five-equation model [40] and a path-conservative framework for the seven-equation model [65] via source terms. We include it within the DEM approach via appropriate modifications to the HLLC Riemann solver. Parallel to this work, Furfarro et al. [68] also developed a similar approach of modifying the HLLC Riemann solver to include the surface tension effects. Unlike the two-phase Riemann problem that is utilized in their work with the volume fraction a variable across the interface, the modifications in this work are applied to the pure-phase Riemann sub-problems within the DEM.
- **Viscous effects** are necessary for applications such as drag computation of a deforming droplet or droplets interacting with turbulence. These have been previously considered in the DEM framework [41] for the five-equation model. This work extends it to the seven-equation model.
- **Multi-species mixtures and reactions** are included in the formulation with an example application such as droplets in a detonation wave [69]. Each phase in this model includes an arbitrary number of species and reactions.

- **Interface compression:** The DEM interface can numerically diffuse across multiple cells. Considering this, an interface compression (or regularization) scheme, initially developed for the five-equation model [39,70] is modified here to be used with the seven-equation model. An improved version of this approach was recently developed that maintains the conservative properties by including the regularization terms within the volume fraction evolution equation and using central-differencing for its evolution [71]. However, in this work, we stay with the original approach of Shukla et al. [70] to keep the interface compression algorithm separate from the DEM.

Herein, Section 2 describes the seven-equation mathematical model, Section 3 formulates the numerical method, and Section 4 shows verification and validation cases.

2. Mathematical formulation

2.1. Fine-grained description

We start from a fine-grained description of a multiphase flow before BN averaging. An indicator function for phase $\kappa = 1, 2$ is defined as $I^\kappa(x_i, t)$, which is 1 if the location x_i is in phase κ at time t , and 0 otherwise. An evolution equation for I^κ is

$$\frac{\partial I^\kappa}{\partial t} + u_i^I \frac{\partial I^\kappa}{\partial x_i} = 0 \quad (i = 1, 2, 3), \quad (2.1)$$

where u_i^I is the interface velocity. The single-phase Navier–Stokes equations accompany these for each phase as

$$\frac{\partial U}{\partial t} + \frac{\partial F_{i,inv}}{\partial x_i} + \frac{\partial F_{i,visc}}{\partial x_i} = \dot{S} \quad (i = 1, 2, 3), \quad (2.2)$$

where,

$$U = \begin{pmatrix} 1, \\ \rho, \\ \rho u_j, \\ \rho E, \\ \rho Y_m \end{pmatrix}, \quad F_{i,inv} = \begin{pmatrix} 0, \\ \rho u_i, \\ \rho u_i u_j, \\ \rho E u_i, \\ \rho Y_m u_i \end{pmatrix} + \begin{pmatrix} 0, \\ 0, \\ p \delta_{ij}, \\ p u_i, \\ 0 \end{pmatrix}, \quad F_{i,visc} = \begin{pmatrix} 0, \\ 0, \\ \tau_{ij}, \\ \tau_{ij} u_j + q_i, \\ \rho Y_m v_{im} \end{pmatrix}, \quad (2.3)$$

where U , $F_{i,inv}$, $F_{i,visc}$, and \dot{S} are solution variables, inviscid fluxes, viscous fluxes, and source terms respectively. The total flux is $F_i = F_{i,inv} + F_{i,visc}$, and ρ , u_j , ($j = 1, 2, 3$), p , Y_m , ($m = 1, \dots, N_{s,\kappa}$), and v_{im} represent density, velocity, pressure, mass-fractions for m^{th} species, and species diffusive velocities respectively. Note that $U(0) = 1$ is redundant but corresponds to a volume fraction evolution equation after the averaging operation. We will show this later. The number of species that are tracked, N_s^κ , can be different for different phases.

The total energy E is related to the internal energy $e = h_{0,mix} + C_{v,mix}T$ as $E = e + 1/2 u_i u_i$, where $C_{v,mix}$ is the mixture constant volume heat capacity and $h_{0,mix}$ is the mixture enthalpy of formation per mass. Pressure p is computed as $P^\kappa(\rho, e, Y_m)$ via an EOS. The mixture formation enthalpy per mass is computed as $h_{0,mix} = \sum_{m=1}^{N_s^\kappa} h_{0,m}^\kappa Y_m$ with $h_{0,m}^\kappa$ being the formation enthalpy per mass of m^{th} species. The mixture heat capacities are computed as

$$C_{v,mix} = \sum_{m=1}^{N_s^\kappa} C_{v,m}^\kappa Y_m \quad \text{and} \quad C_{p,mix} = \sum_{m=1}^{N_s^\kappa} C_{p,m}^\kappa Y_m. \quad (2.4)$$

The mixture specific enthalpy is $h_{mix} = h_{0,mix} + C_{p,mix}T$. The species-specific heat capacities are $C_{p,m}^\kappa$ and $C_{v,m}^\kappa$, and they can be curve-fits of the local temperature if needed. Both phases have independent EOS that determine P^κ and thermodynamic properties $h_{0,m}^\kappa$, $C_{p,m}^\kappa$, $C_{v,m}^\kappa$.

The viscous stress term τ_{ij} is computed from the shear stress tensor

$$\mathcal{T}^\kappa(u_i) = \mu^\kappa \frac{\partial u_i}{\partial x_j} - 2/3 \mu^\kappa \delta_{ij} \frac{\partial u_k}{\partial x_k}. \quad (2.5)$$

The heat flux is modeled as

$$q_i = \lambda^\kappa \frac{\partial T}{\partial x_i} + \sum_{m=1}^{N_s^\kappa} h_m^\kappa Y_m v_{im} \quad (2.6)$$

where μ^κ and λ^κ are viscosity and thermal conductivity. Similar to the thermodynamic properties, the transport properties \mathcal{T}^κ , μ^κ , λ^κ , and species diffusivities are independent of the phases. The same form of the shear stress tensor \mathcal{T}^κ is used

for both the phases (corresponding to a Newtonian fluid). However, a different form can also be used, for example, to model a solid material.

The source terms \dot{S} are $(0, 0, g_i, g_i u_i, \dot{\omega}_m)^T$, where g_i represents gravity and $\dot{\omega}_m$ are the chemical reaction rates. We do not consider gravity hereon but include it for completeness. In the presence of chemical reactions, we also have the relations

$$\sum_m \dot{\omega}_m^k = 0 \quad \text{and} \quad \sum_m Y_m v_{mi} = 0. \quad (2.7)$$

The kinetics that models the reaction rates $\dot{\omega}_m^k$ are also different and independent between the phases.

These equations are complemented by the jump conditions across the phases as

$$[(F_i - u_i^I U)] = \mathcal{J}_i^U \quad (i = 1, 2, 3), \quad (2.8)$$

where

$$\mathcal{J}_i^U = \begin{pmatrix} 0, \\ 0, \\ \sigma \beta n_i n_j, \\ \sigma \beta u_i^I, \\ 0 \end{pmatrix}, \quad (2.9)$$

and $[f]$ is the jump across the interface for any arbitrary field variable f , e.g., $f^{(2)} - f^{(1)}$. Here, since surface tension is considered, n_i is the interface surface normal pointing from the liquid ($\kappa = 2$) to the gas ($\kappa = 1$) phase, and σ and $\beta = \partial n_i / \partial x_i$ are the surface tension and the interface curvature, respectively. These jump conditions provide boundary conditions at the phase interfaces.

In the absence of any mass transfer across the phase interface and the presence of a thermal equilibrium due to heat transfer, these jump conditions transition into

$$[u_i] = 0, \quad [p] = \sigma \beta - [\tau_{ij} n_i n_j], \quad [T] = 0. \quad (2.10)$$

One would also maintain a Gibbs free energy equilibrium across the interface if mass transfer across the phase interface had been considered [47,49,75]. In the absence of mass-transfer the interface velocity u_i^I is u_i because $[u_i] = 0$.

2.2. Averaged equations

To derive the coarse-grained Baer–Nunziato formulation from the fine-grained description, an averaging operator $\langle \cdot \rangle$ is applied. The coarse-grained quantities are defined as $\bar{\alpha}^\kappa = \langle I^\kappa \rangle$ (volume fraction) and $\bar{f}^\kappa \bar{\alpha}^\kappa = \langle I^\kappa f \rangle$ (phase-averaged quantities). Here the operator $\bar{\cdot}$ is spatial filtering as a result of the phase-averaging. However, since both the flow features and the interface are considered resolved in this work, \bar{f}^κ is represented as f^κ from here onward. The generalized seven-equation formulation is derived from Eq. (2.1) and Eq. (2.2), by applying $\langle \cdot \rangle$ as

$$\underbrace{\frac{\partial \alpha^\kappa U^\kappa}{\partial t}}_I + \underbrace{\frac{\partial \alpha^\kappa F_{i,\text{inv}}^\kappa}{\partial x_i}}_{II} + \underbrace{\frac{\partial \alpha^\kappa F_{i,\text{visc}}^\kappa}{\partial x_i}}_{III} = \underbrace{\left(\left(F_{i,\text{inv}}^\kappa - u_i^\kappa U^\kappa \right) \frac{\partial I^\kappa}{\partial x_i} \right)}_{IV} + \underbrace{\left(F_{i,\text{visc}}^\kappa \frac{\partial I^\kappa}{\partial x_i} \right)}_V + \underbrace{\left(U^\kappa (u_i^\kappa - u_i^I) \frac{\partial I^\kappa}{\partial x_i} \right)}_{VI} + \underbrace{\alpha^\kappa \dot{S}^\kappa}_{VII}. \quad (2.11)$$

Here, term I is the time-derivative of flow-field variables, terms II and III are conservative fluxes, terms IV and V are non-conservative terms that are active only at the multiphase interfaces, and they are responsible for interphase exchanges. Term VI is mass transfer across the phase boundary, and they are neglected here but will be a part of future work. Term VII are source terms.

Regardless of whether the size of the MPE is larger or smaller than the filter-size of the averaging operator $\langle \cdot \rangle$ (implicitly same as the finite-volume grid), the same Eq. (2.11) can be used for modeling either dispersed or resolved multiphase flows. Numerical modeling of terms I, II, III, and VII is straightforward and they remain the same for either resolved or dispersed flows. IV and V are unclosed, and their modeling requires a specific focus. In the absence of mass transfer, these are typically modeled for the seven-equation model as [28,41,47,65]

$$(IV + V)^{(1)} = I^{(1)} + R_p^{(1)} + R_u^{(1)} + R_T^{(1)},$$

$$I^{(1)} = \begin{pmatrix} -u_i^I \frac{\partial \alpha^{(1)}}{\partial x_j}, \\ 0, \\ ((p^I - \sigma \beta \delta^{(1)}) \delta_{ij} + \tau_{ij}^I) \frac{\partial \alpha^{(1)}}{\partial x_j}, \\ \left((p^I - \sigma \beta \delta^{(1)}) u_i^I + \tau_{ij}^I u_j^I + q_i^I \right) \frac{\partial \alpha^{(1)}}{\partial x_i}, \\ 0 \end{pmatrix}, \quad (2.12)$$

$$R_p^{(1)} = \begin{pmatrix} \theta^p \Delta p, \\ 0, \\ 0, \\ (p^I - \sigma \beta \delta^{(1)}) \theta^p \Delta p, \\ 0 \end{pmatrix}, R_u^{(1)} = \begin{pmatrix} 0, \\ 0, \\ \theta^u \Delta u_i, \\ u_i^I \theta^u \Delta u_i \\ 0 \end{pmatrix}, R_T^{(1)} = \begin{pmatrix} \frac{1}{\kappa^I} \theta^T \Delta T, \\ 0, \\ 0, \\ \theta^T \Delta T, \\ 0 \end{pmatrix}$$

where $\Delta p = p^{(1)} - p^{(2)} + \sigma \beta$, $\Delta u_i = u_i^{(1)} - u_i^{(2)}$, and $\Delta T = T^{(1)} - T^{(2)}$ are differences of pressure, velocity, and temperature between the phases. Furthermore, p^I , u_i^I (modeling discussed in [28]), τ_{ij}^I , q_i^I (modeling discussed in [41]), and κ^I (modeling discussed in [47]) are interface-averaged terms within the filter $\langle \cdot \rangle$, and since they are unknown, they typically have to be modeled as listed in the references listed. For instance, p^I , u^I have been modeled in the past as $p^{(1)}$ and $u^{(2)}$ (with $\kappa = 1$ being the gas phase and $\kappa = 2$ as the liquid phase) for gas-liquid flows. In this work, however, the discrete equations method (DEM) is used [32], and its use circumvents the need for this modeling by solving Riemann problems at each discrete interface instead. A similar form as in Eq. (2.12) is also used for the phase (2), and it is not repeated here for brevity. As noted before, assuming that the phase (1) is a gas-phase the phase (2) is a liquid-phase, the surface tension force $\sigma \beta$ acts from the gas-phase towards the liquid-phase, and as a result, in Eq. (2.12), $\delta^{(1)} = 1$ for the gas-phase, whereas $\delta^{(2)} = 0$ for the liquid-phase [65].

The terms containing θ^p , θ^u , and θ^T are referred to as the relaxation terms for pressure, velocity, and temperature, respectively. The pressure relaxation results in volume exchange between the phases, the velocity relaxation results in interphase momentum exchanges, e.g., drag, and the temperature relaxation results in heat exchange between the phases. For resolved interfaces, these are typically modeled as $\theta^p, \theta^u, \theta^T, \theta^f \rightarrow \infty$, whereas, for dispersed phases, they are modeled using finite-time processes such as drag and heat transfer [28]. Enforcing this condition analytically would have resulted in a reduced five-/six-equation model [33,46]. Past works [28,47] had to use a stiff relaxation solver with the seven-equation model to enforce $\theta^p, \theta^u \rightarrow \infty$. However, the use of DEM circumvented this as it was shown for 1D canonical cases in [32]. This is further evaluated for more complex test cases in this work (Section 4 for more details).

2.3. Material properties

The governing equations are accompanied by an equation of state (P^K) and material properties for each phase. For demonstration, stiffened gas (SG, includes parameters p_0^K and γ^K) [28], calorically perfect gas (CPG, has parameter γ^K), thermally perfect gas (TPG, requires property curve-fits for individual species), and Mie–Grüneisen (MG, requires properties for individual species) [78] EOS are used in this work. These and the corresponding parameters are well-described in the available literature, and their forms are not repeated here for brevity. The corresponding parameters for various materials are described separately for each test as required in Section 4.

A Sutherland transport model is used for the gas-phase viscosity ($\mu^{(1)}$). The liquid-viscosity ($\mu^{(2)}$) and surface tension σ are assumed to be constants, although it is possible to use curve-fits for those as well. The thermal conductivity λ^K is computed from the viscosity μ^K using a constant Prandtl number ($Pr = 0.739$) as $\lambda^K = C_{p,mix}^K \mu^K / Pr$. The species diffusion is modeled using a unity Lewis number (Le) assumption, with $Le_m^K = \lambda^K / (\rho^K D_m^K C_{p,m}^K) = 1$. For the reactive simulations shown in this work, following previous work on hydrogen-air detonations [79], a 7-step, 6-species chemical mechanism [80] is used to model finite-rate kinetics.

3. Numerical method

A three-dimensional finite-volume computational framework is used to solve Eq. (2.11). The equations are implemented in LESLIE, a parallel multi-physics solver, which has been used extensively in the past for multiphase reacting flow simulations [8,81,82]. The numerical approach here primarily follows published literature [28,32,41,65,70], and every detail is not repeated for brevity. However, the following novel contributions are made, and details about them are provided in the following sections.

- Implementation of the DEM [32] for multi-block structured curvilinear grids.
- Modification in HLLC [83] Riemann solver within the DEM to include surface tension.
- Extension of viscous effects within the DEM [41] for multiple dimensions.
- Improved robustness of the stiff relaxation solver [28] for dealing with arbitrary EOS.

- Application of an interface compression scheme [70] for the seven-equation model.

3.1. Baseline approach

A multi-block structured grid is used, wherein C_{ijk} is a computational cell with i , j , and k as the indices in the three curvilinear directions. The corresponding cell volume is denoted as V_{ijk} , and $S_{l,ijk}$ is used to indicate the cell faces. Note that these i , j , k are different from the i , j , k that we used in Section 2 for the Einstein notation. Let (ξ, η, ζ) be the computational directions of increasing i , j and k , and $\partial\xi/\partial x_i$, $\partial\eta/\partial x_i$, $\partial\zeta/\partial x_i$ are the corresponding grid metrics. Quantities computed at the face-centers are denoted with sub-scripts $i+1/2$, $j+1/2$ and $k+1/2$. For example, $S_{i+\frac{1}{2}jk}$ and $N_{i,i+\frac{1}{2}jk}$ are the face-area and the face-normal, respectively, between the cells C_{i+1jk} and C_{ijk} .

A volume-integrated form of Eq. (2.11) is numerically solved on this grid. The cell-centered quantity to be updated at each time-step is $(\alpha^\kappa U^\kappa)_{ijk}$, and it is also referred to as Q_{ijk} . The corresponding time-update is denoted as $d(\alpha^\kappa U^\kappa)_{ijk}$ or dQ_{ijk} . The increment dQ (or $d(\alpha_\kappa U_\kappa)$) has contributions due to various terms in Eq. (2.11) and they are denoted using subscripts as dQ_{II} , dQ_{III} , etc.

The computation is advanced in time using an explicit two-stage MacCormack approach. The time-step Δt is dynamically computed considering convective, acoustic, and diffusive propagation speeds of both the phases as

$$\Delta t_{ijk}^\kappa = CFL \cdot V_{ijk} \left/ \left(\frac{1}{6} \sum_{l=1}^6 |v_{l,ijk}^\kappa| S_{l,ijk} + c^\kappa \bar{S}_{ijk} + \frac{2\gamma^\kappa v^\kappa}{Pr} \bar{S}_{ijk}^2 \right) \right), \quad \Delta t = \min_{ijk} \min \left(\Delta t_{ijk}^{(1)}, \Delta t_{ijk}^{(2)} \right) \quad (3.1)$$

where, $S_{l,ijk}$ are face area and $v_{l,ijk}^\kappa = \sum_{i=1}^3 u_{i,ijk}^\kappa n_{i,ijk}$ is the normal velocity on the l^{th} face of the cell C_{ijk} . The averaged face area for cell C_{ijk} is given by $\bar{S}_{ijk} = \frac{1}{6} \sum_{l=1}^6 S_{l,ijk}$, and γ^κ , $v^\kappa = \mu^\kappa / \rho^\kappa$, and Pr^κ are the heat capacity, the kinematic viscosity and the Prandtl number, respectively. The CFL is chosen as 0.5 for stability.

3.2. Inviscid fluxes

In the finite-volume scheme, the inviscid increments due to the terms II and IV in Eq. (2.11) are provided as

$$d(\alpha^\kappa U^\kappa)_{ijk,II/IV} = d(\alpha^\kappa U^\kappa)_{ijk,II/IV}^i + d(\alpha^\kappa U^\kappa)_{ijk,II/IV}^j + d(\alpha^\kappa U^\kappa)_{ijk,II/IV}^k, \quad (3.2)$$

where the superscripts i , j , k denote contributions from each direction. The fluxes in each direction are then computed using the DEM. Their computation is illustrated here only for the i -direction, but the same forms are used in the other directions as well.

3.2.1. Discrete equations method (DEM)

Instead of directly solving for the conservative and the non-conservative terms, DEM divides the problem into three Riemann problems at each cell-face. A schematic for this is shown in Fig. 1(a), where the three Riemann problems at each face are shown corresponding to three possibilities of the two-phase boundaries, i.e., (1)-(1) (R1), (1)-(2) (R2) and (2)-(2) (R3). Note that piece-wise constant fields are shown in each cell here for illustration. However, we go up to second-order accuracy with piece-wise linear fields for the numerical method implemented. On each face, left- and right-interpolations are handled using a first/second-order MUSCL approach [23], and the interpolated values are denoted with subscripts $i+1/2$, l , $i+1/2$, r , etc.

3.2.2. Riemann solution

Solutions to the Riemann sub-problems (1)-(1), (1)-(2), and (2)-(2) are now needed. An HLLC approximate Riemann solver [83] has been used traditionally [28,32]. However, modifications to it are required, including the surface tension. Parallel to this work, similar modifications in the HLLC solver for including surface tension effects have also been considered by Furfar et al. [68]. Even though there are similarities in the derivation, the modifications in the current work are in the context of solving a (1)-(2) Riemann sub-problem within the DEM, and not for a two-phase Riemann problem [68] with phase volume fraction as a variable. Note that for a (1)-(2) Riemann problem, a discontinuity in pressure would be maintained across the contact surface such as $p^{(2)} - p^{(1)} = p^\sigma$, whereas, the solution should stay unaffected for (1)-(1) or (2)-(2) problems. A typical 1D Riemann problem is depicted in Fig. 2, which could be either of the three sub-problems R1, R2, or R3. The state vector $\mathcal{U} = (1, \rho, \rho u_i, \rho E, \rho Y_m)^T$ is set equal to either $U^{(1)}$ or $U^{(2)}$ (on both the left and the right side of the face) depending on whether it belongs to the phase (1) or (2). In the case where the left and the right state vectors \mathcal{U}^L or \mathcal{U}^R belong to different phases, i.e., (1)-(2) Riemann problem, the contact wave separates the phases as the problem evolves.

A sharp jump is assumed across both the left and the right shock/expansion waves in the HLLC approximate Riemann solver [83], and these waves travel with the speeds c^L and c^R , respectively. These are the acoustic propagation speeds corresponding to the state vectors \mathcal{U}^L and \mathcal{U}^R , respectively. Considering this, we have the following relations across these jumps.

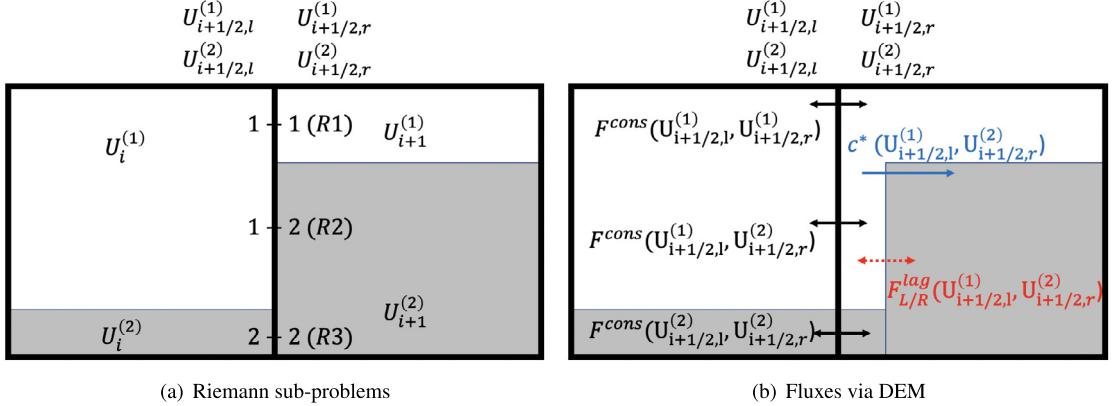


Fig. 1. Schematic showing the three Riemann problems (a) and the flux computation (b) with the DEM. Piecewise constant values are shown within the cell for illustration. However, the numerical scheme allows for second-order accuracy via piecewise linear interpolations. The Gray/white-colored region shows phases (2)/(1).

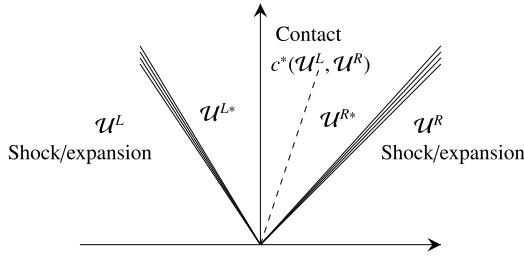


Fig. 2. A sample Riemann problem is shown here, where \mathcal{U}^L and \mathcal{U}^R are the initial left- and right-states, and \mathcal{U}^{L*} , \mathcal{U}^{R*} are the generated intermediate states. The waves on the extreme right/left could be a shock or an expansion wave (solid line), and the wave in the middle is a contact wave (dashed line), whose speed is denoted using c^* .

$$\mathcal{F}^{K*} - \mathcal{F}^K = c^K (\mathcal{U}^{K*} - \mathcal{U}^K), \quad K \in \{L, R\}, \quad \mathcal{F}^{R*} - \mathcal{F}^{L*} + \mathcal{F}^\sigma = c^* (\mathcal{U}^{R*} - \mathcal{U}^{L*}), \quad (3.3)$$

where $\mathcal{F} = (0, \rho v_n, \rho u_i v_n + p n_i, \rho E v_n + p v_n, \rho Y_m v_n)$ is the flux vector, and that is defined for each state in the Riemann problem. As noted before, $v_n = \sum_{i=1}^3 u_i n_i$ is the velocity normal to the corresponding cell-face. The term $\mathcal{F}^\sigma = (0, 0, p^\sigma n_i, p^\sigma c^*, 0)$ is the flux due to the surface tension at the interface boundary (at the contact wave in this case).

The pressures and the normal velocities across the interface (contact wave) are related as $p^{L*} = p^{R*} + p^\sigma$, $v_n^{L*} = v_n^{R*}$, where, $p^\sigma = n^{LR} \sigma \beta$ is the pressure jump across the interface due to the surface tension. The normal vector n^{LR} is 1 if the left phase is the liquid phase and the right phase is the gas phase, i.e., (2)-(1), -1 if the left phase is the gas phase and the right phase is the liquid-phase, i.e., (1)-(2), and 0 if the phases on both the side of the interface are the same, i.e., (1)-(1) or (2)-(2). Combining these equations and following the derivation in Toro [83] with these additional modifications due to the surface tension force, the speed of the contact wave and the intermediate states are given as

$$c^* (\mathcal{U}^L, \mathcal{U}^R) = \frac{p^R - p^L + p^\sigma + \rho^L v_n^L (c^L - v_n^L) - \rho^R v_n^R (c^R - v_n^R)}{\rho^L (c^L - v_n^L) - \rho^R (c^R - v_n^R)}, \quad \mathcal{F}^* = \rho^K \left(\frac{c^K - v_n^K}{c^K - c^*} \right) \quad (3.4)$$

$$\mathcal{U}^{K*} (\mathcal{U}^L, \mathcal{U}^R) = \begin{pmatrix} 1, \\ \mathcal{F}^*, \\ \mathcal{F}^* (u_i^K + (c^* - v_n^K) n_i), \\ \mathcal{F}^* E^K + \mathcal{F}^* (c^* - v_n^K) \left(c^* + \frac{p^K}{\rho^K (c^K - v_n^K)} \right), \\ \mathcal{F}^* Y_m^K \end{pmatrix}. \quad (3.5)$$

Once the intermediate states are known, the conservative Riemann flux at the cell face is computed as

$$F_{inv}^{cons} (\mathcal{U}^L, \mathcal{U}^R) = \begin{cases} \mathcal{F}(\mathcal{U}^L) & \text{if } 0 \leq c^L \\ \mathcal{F}(\mathcal{U}^{L*}) & \text{if } c^L \leq 0 \leq c^* \\ \mathcal{F}(\mathcal{U}^{R*}) & \text{if } c^* \leq 0 \leq c^R \\ \mathcal{F}(\mathcal{U}^R) & \text{if } c^R \leq 0 \end{cases} \quad (3.6)$$

Note that the conventional single-phase Riemann solution is recovered by letting $p^\sigma = 0$.

3.2.3. Inviscid conservative flux (Terms II)

Based on the Riemann solutions that are constructed for the sub-problems (1)-(1), (1)-(2), and (2)-(2), we can now compute the inviscid conservative fluxes. To illustrate this process with DEM, a schematic is shown in Fig. 1 (b). For this example, during the Riemann problem evolution, the interface that was initially at $i + 1/2$ has now moved within the cell on the right based on the speed of the contact wave c^* . F_{inv}^{cons} are the conservative fluxes for the Riemann problems (1)-(1), (1)-(2), and (2)-(2) that act at the $i + 1/2$ cell-face.

Considering the solutions from the three Riemann problems, the averaged i -directional conservative fluxes (term II in Eq. (2.11)) at face $i + 1/2$ for phase (1) are given using the DEM as

$$\begin{aligned} \langle I^{(1)}(F_{inv}^{(1)}) \rangle_{i+\frac{1}{2}} = & \min\left(\alpha_{i+\frac{1}{2},l}^{(1)}, \alpha_{i+\frac{1}{2},r}^{(1)}\right) F_{inv}^{cons}\left(U_{i+\frac{1}{2},l}^{(1)}, U_{i+\frac{1}{2},r}^{(1)}\right) \\ & + \max\left(\alpha_{i+\frac{1}{2},l}^{(1)} - \alpha_{i+\frac{1}{2},r}^{(1)}, 0\right) \left(\beta_{i+\frac{1}{2}}^{(1,2)}\right)^+ F_{inv}^{cons}\left(U_{i+\frac{1}{2},l}^{(1)}, U_{i+\frac{1}{2},r}^{(2)}\right) \\ & + \max\left(\alpha_{i+\frac{1}{2},l}^{(2)} - \alpha_{i+\frac{1}{2},r}^{(2)}, 0\right) \left(-\beta_{i+\frac{1}{2}}^{(2,1)}\right)^+ F_{inv}^{cons}\left(U_{i+\frac{1}{2},l}^{(2)}, U_{i+\frac{1}{2},r}^{(1)}\right), \end{aligned} \quad (3.7)$$

where F_{inv}^{cons} are the fluxes based on the Riemann solutions as discussed earlier, and their coefficients are the probabilities of encountering a gas-gas, a liquid-liquid, a gas-liquid, or a liquid-gas interface on a fine-grained discrete level. Further details and justifications are provided in the original DEM work [32], and they are not repeated here for brevity. Coefficients $\beta_{i+1/2}^{(p,q)}$ are computed as $\text{sign}(c^*(U_{i+1/2,l}^p, U_{i+1/2,r}^q))$, and $(\beta)^+ = \max(\beta, 0)$.

After computing the fluxes, the increment $d(\alpha^\kappa U^\kappa)$ due to the i -directional conservative inviscid flux (term II in Eq. (2.11)) is computed as

$$Vd(\alpha^\kappa U^\kappa)_{ijk,II}^i = S_{i+\frac{1}{2}jk} \langle I^\kappa(F_{inv}^\kappa) \rangle_{i+\frac{1}{2}jk} - S_{i-\frac{1}{2}jk} \langle I^\kappa(F_{inv}^\kappa) \rangle_{i-\frac{1}{2}jk}. \quad (3.8)$$

3.2.4. Inviscid non-conservative flux (Terms IV)

The non-conservative fluxes are referred to as the ones that act at a (1)-(2) interface (contact) but within a finite-volume cell once the multiphase interface has moved into it, e.g., the flux $F_{L/R}^{lag}$ acting on the interface that has moved into the cell $i + 1$ in Fig. 1 (b). The subscripts L/R correspond to whether the flux is acting on the material on the left or right side of the interface. Following [32], second-order i -directional update for phase (1) due to the inviscid non-conservative flux (term IV in Eq. (2.11)) is given as

$$\begin{aligned} Vd(\alpha^{(1)} U^{(1)})_{ijk,IV}^i = & \max\left(\alpha_{i+\frac{1}{2},l}^{(1)} - \alpha_{i+\frac{1}{2},r}^{(1)}, 0\right) \left(-\beta_{i+\frac{1}{2}}^{(2,1)}\right)^+ F_{L,inv}^{lag}\left(U_{i+\frac{1}{2},l}^{(1)}, U_{i+\frac{1}{2},r}^{(2)}\right) S_{i+\frac{1}{2}} \\ & - \max\left(\alpha_{i+\frac{1}{2},l}^{(2)} - \alpha_{i+\frac{1}{2},r}^{(2)}, 0\right) \left(-\beta_{i+\frac{1}{2}}^{(2,1)}\right)^+ F_{R,inv}^{lag}\left(U_{i+\frac{1}{2},l}^{(2)}, U_{i+\frac{1}{2},r}^{(1)}\right) S_{i+\frac{1}{2}} \\ & + \max\left(\alpha_{i-\frac{1}{2},l}^{(1)} - \alpha_{i-\frac{1}{2},r}^{(1)}, 0\right) \left(\beta_{i+\frac{1}{2}}^{(1,2)}\right)^+ F_{L,inv}^{lag}\left(U_{i+\frac{1}{2},l}^{(1)}, U_{i+\frac{1}{2},r}^{(2)}\right) S_{i-\frac{1}{2}} \\ & - \max\left(\alpha_{i-\frac{1}{2},l}^{(2)} - \alpha_{i-\frac{1}{2},r}^{(2)}, 0\right) \left(\beta_{i-\frac{1}{2}}^{(2,1)}\right)^+ F_{R,inv}^{lag}\left(U_{i-\frac{1}{2},l}^{(2)}, U_{i-\frac{1}{2},r}^{(1)}\right) S_{i-\frac{1}{2}} \\ & + \max\left(\alpha_{i-\frac{1}{2},r}^{(1)} - \alpha_{i+\frac{1}{2},l}^{(1)}, 0\right) F_L^{lag}\left(U_i^{(1)}, U_i^{(2)}\right) \frac{1}{2} (S_{i+\frac{1}{2}} + S_{i-\frac{1}{2}}) \\ & - \max\left(\alpha_{i-\frac{1}{2},r}^{(2)} - \alpha_{i+\frac{1}{2},l}^{(2)}, 0\right) F_R^{lag}\left(U_i^{(2)}, U_i^{(1)}\right) \frac{1}{2} (S_{i+\frac{1}{2}} + S_{i-\frac{1}{2}}) \\ & + \langle N_{int} \rangle \left(F_R^{lag}\left(U_i^{(2)}, U_i^{(1)}\right) - F_L^{lag}\left(U_i^{(1)}, U_i^{(2)}\right) \right) \frac{1}{2} (S_{i+\frac{1}{2}} + S_{i-\frac{1}{2}}), \end{aligned} \quad (3.9)$$

where $F_{L/R,inv}^{lag}(U^L, U^R)$ are fluxes between the phases at the interface, and they are computed as part of the previously defined (1)-(2) Riemann solution as

$$\begin{aligned} F_{L,inv}^{lag}(U^L, U^R) &= \mathcal{F}(\mathcal{U}^{L*}) - c^*(\mathcal{U}^L, \mathcal{U}^L) \mathcal{U}^{L*}, \\ F_{R,inv}^{lag}(U^L, U^R) &= \mathcal{F}(\mathcal{U}^{R*}) - c^*(\mathcal{U}^L, \mathcal{U}^R) \mathcal{U}^{R*} \end{aligned} \quad (3.10)$$

Note that based on the approximate Riemann solution derived earlier, we have $F_L^{lag} - F_R^{lag} = F^\sigma$ for flows with surface tension.

The first two terms in Eq. (3.9) correspond to the non-conservative fluxes that act within the cell i as a result of a multiphase interface that would have entered through the cell-face $i + 1/2$. The following two terms correspond to an interface that would have entered through the cell-face $i - 1/2$. The subsequent two terms are for second-order accuracy when piecewise linear interpolations are used within a cell. In the last term, $\langle N_{int} \rangle$ is the average number of internal interfaces within the cell, and this term was related to the relaxation processes within a cell, which would lead to local equilibrium. For resolved interfaces, this could be modeled via the stiff relaxation solvers as described later in Section 3.5. The reader can find more details about the DEM elsewhere [32], and they are not repeated here for brevity.

3.3. Viscous fluxes

Terms III and V in Eq. (2.11) are the viscous conservative and non-conservative updates. Abgrall and Rodio [41] extended the original DEM framework [32] for viscous fluxes. However, they only did this for 1D flows, and here it is also used for 2D/3D. The viscous updates $d(\alpha^\kappa U^\kappa)_{ijk,III}$ and $d(\alpha_\kappa U_\kappa)_{ijk,V}$ follow the same formula as Eq. (3.2) for their directional contributions from $i/j/k$. The flux computations are described here only for the i -direction, but the other directions use a similar form.

3.3.1. Single-phase viscous fluxes

A Riemann problem is inherently defined for 1D inviscid flows, and therefore, it is not possible to completely replicate the inviscid computations shown earlier for the viscous terms. The computation of single-phase viscous fluxes $F_{i,visc}^\kappa$ is discussed first in this section, and their compilation into the two-phase conservative and non-conservative terms is described in the following sections.

Centered differencing and centered averaging are used for computing the terms $\partial_{x_i} f|_{i+1/2jk}$, $f_{i+1/2jk}$, etc. The physical derivatives $\partial_{x_i} f$, e.g., $\partial_{x_j} u_i$, $\partial_{x_i} T$, etc., are obtained from the computational derivatives as

$$\frac{\partial f}{\partial x_i} = \frac{\partial f}{\partial \xi} \frac{\partial \xi}{\partial x_i} + \frac{\partial f}{\partial \eta} \frac{\partial \eta}{\partial x_i} + \frac{\partial f}{\partial \xi} \frac{\partial \xi}{\partial x_i}, \quad (3.11)$$

where $\partial_\xi f$, $\partial_\eta f$, and $\partial_\xi f$ are computed from the flow-variables as

$$\begin{aligned} \left. \frac{\partial f}{\partial \xi} \right|_{i+\frac{1}{2}jk} &= f_{i+1jk} - f_{ijk}, \\ \left. \frac{\partial f}{\partial \eta} \right|_{i+\frac{1}{2}jk} &= \frac{1}{2} (f_{i+\frac{1}{2}j+1k} - f_{i+\frac{1}{2}j-1k}), \\ \left. \frac{\partial f}{\partial \xi} \right|_{i+\frac{1}{2}jk} &= \frac{1}{2} (f_{i+\frac{1}{2}jk+1} - f_{i+\frac{1}{2}jk-1}). \end{aligned} \quad (3.12)$$

In addition, any face-averaged quantities, e.g., α^κ , μ^κ , etc., are computed as

$$f_{i+\frac{1}{2}jk} = \frac{1}{2} (f_{i+1jk} + f_{ijk}). \quad (3.13)$$

With these, the i -directional single-phase viscous fluxes are computed as

$$F_{visc,i+\frac{1}{2}jk}^\kappa = F_{i,visc}^\kappa \left(\mu_{i+\frac{1}{2}jk}^\kappa, \dots, \left. \frac{\partial u_i^\kappa}{\partial x_i} \right|_{i+\frac{1}{2}jk}, \dots \right) \cdot n_{i,i+\frac{1}{2}jk}. \quad (3.14)$$

3.3.2. Viscous conservative flux (Terms III)

For the DEM computation of the viscous conservative fluxes, similar forms as Eq. (3.7) and Eq. (3.8) are used. However, instead of using the F_{inv}^{cons} from the Riemann solutions as before, the terms F_{visc}^κ at the cell-faces (e.g., $i + 1/2, j, k$) are now computed using Eq. (3.14). The DEM viscous fluxes are then given as

$$\begin{aligned} \left(I^{(1)} \left(F_{visc}^{(1)} \right) \right)_{i+\frac{1}{2}} &= \min \left(\alpha_{i+\frac{1}{2},l}^{(1)}, \alpha_{i+\frac{1}{2},r}^{(1)} \right) F_{visc}^{(1)} \\ &\quad + \max \left(\alpha_{i+\frac{1}{2},l}^{(1)} - \alpha_{i+\frac{1}{2},r}^{(1)}, 0 \right) \left(\beta_{i+\frac{1}{2}}^{(1,2)} \right)^+ F_{visc}^{(1)} \\ &\quad + \max \left(\alpha_{i+\frac{1}{2},l}^{(2)} - \alpha_{i+\frac{1}{2},r}^{(2)}, 0 \right) \left(-\beta_{i+\frac{1}{2}}^{(2,1)} \right)^+ F_{visc}^{(1)}, \end{aligned} \quad (3.15)$$

and a similar form is used for $\kappa = 2$ as well.

3.3.3. Viscous non-conservative (Terms V)

The non-conservative viscous terms follow the same form as Eq. (3.9), however, instead of $F_{L/R,\text{inv}}^{\text{lag}}$ coming from the 1D inviscid Riemann problems, $F_{L/R,\text{visc}}^{\text{lag}}$ are modeled as a function of the single-phase viscous terms F_{visc}^{κ} computed at cell-faces (e.g., $i + 1/2, j, k$) as

$$\begin{aligned} F_{L,\text{visc}}^{\text{lag}}(U_{i+\frac{1}{2},l}^{(1)}, U_{i+\frac{1}{2},l}^{(2)}) &= F_{R,\text{visc}}^{\text{lag}}(U_{i+\frac{1}{2},l}^{(2)}, U_{i+\frac{1}{2},l}^{(1)}) = F_{\text{visc},i+\frac{1}{2}}^{(1)}, \\ F_{L,\text{visc}}^{\text{lag}}(U_i^{(1)}, U_i^{(2)}) &= F_{R,\text{visc}}^{\text{lag}}(U_i^{(2)}, U_i^{(1)}) = \frac{S_{i+\frac{1}{2}} F_{\text{visc},i+\frac{1}{2}}^{(1)} + S_{i-\frac{1}{2}} F_{\text{visc},i-\frac{1}{2}}^{(1)}}{S_{i+\frac{1}{2}} + S_{i-\frac{1}{2}}}, \end{aligned} \quad (3.16)$$

where $\kappa = 1$ is assumed as the gas-phase, and therefore, it is assumed that $F_{\text{visc}}^I = F_{\text{visc}}^{(1)}$. This is similar to the previous viscous DEM approach of Abgrall and Rodio [41].

3.4. Curvature computation

The interface curvature β is required to compute $p^\sigma = \sigma\beta$ at the cell-centers (i, j, k) . Once the curvature is computed at the cell centers, it is then interpolated at the cell faces using the MUSCL approach in the same way as the other field quantities for flux computation. The curvature for a field f is defined as

$$\beta = \frac{\partial f}{\partial x_i} \left(\frac{\partial f}{\partial x_i} \right). \quad (3.17)$$

A previous approach for curvature computation is followed that used a vertex-based staggering [40,65], which is illustrated in 2D. The gradient of f at a grid vertex is first computed as

$$\begin{aligned} \frac{\partial f}{\partial x_1}_{i+\frac{1}{2},j+\frac{1}{2}} &= \frac{1}{2} \left(\frac{f_{i+1,j+1} - f_{i,j+1}}{\Delta x} + \frac{f_{i+1,j} - f_{i,j}}{\Delta x} \right), \\ \frac{\partial f}{\partial x_2}_{i+\frac{1}{2},j+\frac{1}{2}} &= \frac{1}{2} \left(\frac{f_{i+1,j+1} - f_{i+1,j}}{\Delta x} + \frac{f_{i,j+1} - f_{i,j}}{\Delta x} \right), \end{aligned} \quad (3.18)$$

where Δx and Δy are the grid sizes in the i and the j directions. Based on this gradient, a unit vector normal to the interface is computed as

$$n_{i,i+\frac{1}{2},j+\frac{1}{2}} = \begin{cases} \frac{\frac{\partial f}{\partial x_1}_{i+\frac{1}{2},j+\frac{1}{2}}}{\left\| \frac{\partial f}{\partial x_1}_{i+\frac{1}{2},j+\frac{1}{2}} \right\|}, & \text{if } \left\| \frac{\partial f}{\partial x_1}_{i+\frac{1}{2},j+\frac{1}{2}} \right\| > 0, \\ 0, & \text{otherwise,} \end{cases} \quad (3.19)$$

where, $\|\cdot\|$ is the magnitude of the vector. The normal vector is then interpolated at the cell-faces as

$$n_{i,i+\frac{1}{2},j} = \frac{1}{2} \left(n_{i,i+\frac{1}{2},j+\frac{1}{2}} + n_{i,i+\frac{1}{2},j-\frac{1}{2}} \right), \quad (3.20)$$

and finally, the curvature is computed at the cell-centers as

$$\kappa_{i,j} = \frac{n_{1,i+\frac{1}{2},j} - n_{1,i-\frac{1}{2},j}}{\Delta x} + \frac{n_{2,i,j+\frac{1}{2}} - n_{2,i,j-\frac{1}{2}}}{\Delta y}. \quad (3.21)$$

The previous approach [65] used $f = \alpha^{(1)}$ in Eq. (3.17) as that would result in the interface curvature at the droplet surface. However, $\alpha^{(1)}$ varies sharply from 0 to 1 across an interface, and this can result in numerical errors. To address this, in this work, $f = \phi$ is used, which is defined as

$$\phi = \frac{(\alpha^{(2)})^m}{(\alpha^{(2)})^m + (1 - \alpha^{(2)})^m} \quad \text{for } m < 1. \quad (3.22)$$

Shukla et al. [70] used this form of ϕ for their interface compression scheme, and it is used for the curvature computation in this work. The field ϕ is smoother as compared to $\alpha^{(2)}$, and m is a parameter that decides its smoothness. Note that $m = 1$ would result in $\phi = \alpha^{(2)}$. Finally, one would compute the curvature everywhere in the flow-field using Eq. (3.21). However, to reduce any numerical noise, it is only used in the regions where $0.99 > \alpha^{(1)} > 0.01$, and it is set to zero otherwise.

To verify the curvature computation, a circular drop with diameter $d_0 = 9.6$ mm is placed at the center ($x_0 = 0, y_0 = 0$) of a square domain of size $[6d_0, 6d_0]$. The computational domain is uniformly divided into 240×240 cells. The volume fraction for the droplet is initialized via

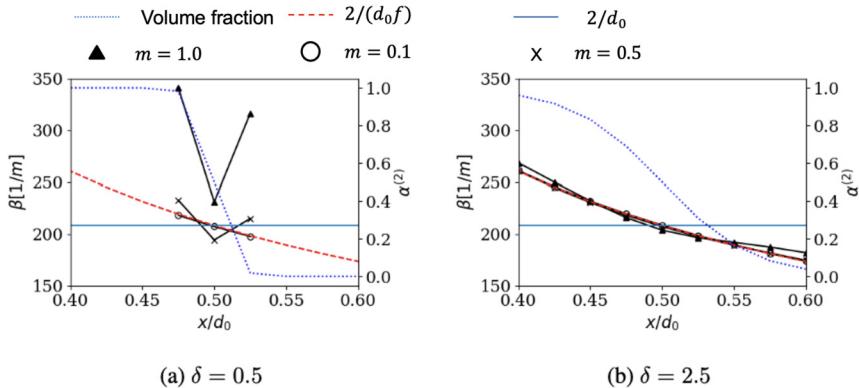


Fig. 3. The curvature and the volume fraction are shown for a circular drop along the centerline initialized as Eq. (3.23). The computed curvature using Eq. (3.21) and Eq. (3.22) is shown for $m = 0.1$, $m = 0.5$ and $m = 1.0$. The exact curvature along the droplet radius is shown as $2/d_0 f$, and the exact droplet curvature at the interface is shown as $2/d_0$.

$$f = \sqrt{\left(\frac{x - x_0}{d_0/2}\right)^2 + \left(\frac{y - y_0}{d_0/2}\right)^2}, \quad \text{and} \quad \alpha^{(2)} = \frac{1}{2} \tanh\left(\frac{(f - 1)d_0}{\delta 2 \Delta x}\right) + \frac{1}{2}, \quad (3.23)$$

where δ is the thickness over which the numerical interface is diffused initially. The corresponding volume fraction fields for $\delta = 0.5$ and $\delta = 2.5$, as well as the numerically computed curvature are shown in Fig. 3 along the center-line. The exact curvature for a droplet with diameter d_0 would be $2/d_0$ (shown by the solid horizontal line in Fig. 3). However, since the interface is diffused, if we were to compute the curvature using Eq. (3.17) and Eq. (3.23), then it would be $2/(d_0 f)$ along the radius of the droplet (shown via dashed line in Fig. 3).

A comparison of the curvature computed using Eq. (3.21) and Eq. (3.22) at m in Fig. 3 shows that for $\delta = 2.5$, when the interface is diffused across 8-10 cells, $m = 0.1, 0.5, 1$, provide a good match against $2/d_0 f$. The surface tension only acts at the sharp interface where the curvature is $2/d_0$ (horizontal line). Indeed, a numerically diffused volume fraction could result in further errors. Therefore, the curvature computation must work for sharp interfaces and should compute the curvature as $2/d_0$. For $\delta = 0.5$, the interface is diffused over 2-3 cells. For this case, choosing $m = 1$ (i.e., $f = \alpha^{(2)}$, which was used in earlier works [65]) results in about 50% errors, but with $m = 0.5$, the errors are < 5%.

Choosing $m = 0.1$ leads to slightly lower errors as compared to $m = 0.5$. However, the test considered here is an ideal scenario where the volume fraction is defined using Eq. (3.23), and as a result, f is monotonic. In reality, there could be numerical oscillations in the $\alpha^{(2)}$ field, and those, when coupled with the high gradients of $\partial f/\partial\alpha$ near $\alpha = 0, 1$, could result in a non-monotonic behavior of f for a lower m such as 0.1. This could lead to further errors in the computation, and considering this, $m = 0.5$ is chosen as a balance between the two effects in this work.

3.5. Relaxation process

The quantities R_u^K , R_p^K and R_T^K in Eq. (2.12) are relaxation terms. Physically, they represent the interface jump conditions of Eq. (2.9) and Eq. (2.10) for a resolved interface. Since $\theta^P, \theta^U, \theta^T \rightarrow \infty$ for the seven-equation model in the resolved MPE limit, the relaxation processes are stiff, and they are applied separately from the inviscid and the viscous updates through a Strang splitting [28]. The velocity relaxation term, θ^U is modeled as a finite-rate process in the dispersed phase limit [8], and further details of that are provided later in Section 4.4. Correspondingly, the flow solution is updated as $Q^{n+1} = \mathcal{R}^C \mathcal{R} \mathcal{L}^P(Q^n)$, where \mathcal{R} represents the stiff relaxation operator, \mathcal{L}^P and \mathcal{L}^C correspond to the predictor and the corrector steps of the time-update, and the superscript n is used to represent the temporal iteration number. The pressure, velocity and the temperature relaxation operators are represented as \mathcal{R}_p , \mathcal{R}_u , \mathcal{R}_T , respectively, and $\mathcal{R} = \mathcal{R}_T \mathcal{R}_u \mathcal{R}_p$.

In terms of the numerical modeling of the interphase exchanges through the DEM, the terms with $\langle N_{int} \rangle$ in Eq. (3.9) correspond to the relaxation terms as shown by Abgrall and Saurel [32]. Here, $\langle N_{int} \rangle$ represents the number of unresolved interfaces within a finite-volume cell, and this would typically be finite for dispersed flows where the droplets are unresolved. Still, it is supposed to be negligible for the resolved interfaces such as those considered here. As a result of this, while using the DEM, there should be no need to use a separate stiff relaxation solver for resolved interfaces [32], and this is further verified in this work (see Section 4).

Interestingly, a recent work [67] showed that it is not necessary to model the non-conservative terms I^K in Eq. (2.12) if the relaxation terms R^K are modeled consistently. This may suggest that both the non-conservative terms I^K and the relaxation terms R^K model the same interphase exchange effects but differently. Their work also showed that within the six-equation framework, neglecting the pressure relaxation terms was fine for shock-tube problems, but they were required for accurately modeling spherical collapse of a bubble, and this test is evaluated here as well in Section 4.7 within the context of the seven-equation model.

However, to provide a consistent comparison against the previous works that did use a relaxation solver [28], and because the seven-equation model would asymptote to the widely used five-equation model in the stiff pressure and velocity relaxation limit, the operators \mathcal{R}_p and \mathcal{R}_u are implemented in the current computational framework. The stiff temperature relaxation operator \mathcal{R}_T is not included in this work, so the thermal exchanges through the interface are only via the resolved interphase exchange terms I^K in Eq. (3.9) via the DEM.

The stiff velocity relaxation operator \mathcal{R}_u modifies the phase-averaged momentum and energy ($\rho^K \alpha^K u_i^K$, $\rho^K \alpha^K E^K$) while keeping the mixture momentum and energy conserved. The algorithm for this is described elsewhere [28] and it is not repeated here. The stiff pressure relaxation \mathcal{R}_p results in local expansion/compression of the phases via a change in the volume fraction α^K , and corresponding pressure work terms are included in the energy $\rho^K \alpha^K E^K$ (see Eq. (3.9)). The algorithm matches that of [28], but with the following improvements:

- The viscous and the surface tension forces are included in the pressure jump condition as shown in Eq. (2.10). As identified earlier [41], $\tau_{ij}^{(1)} n_i n_j = \tau_{ij}^{(2)} n_i n_j$ is maintained by the viscous DEM, and therefore, the pressure jump that would have to be applied at the interface via the relaxation process is $p^{(2)} - p^{(1)} = \sigma \beta$. We can combine the volume fraction and the energy equations, resulting in

$$\frac{\partial \alpha^{(1)} \rho^{(1)} E^{(1)}}{\partial t} = - \left(p^{(1)} - \sigma \beta \right) \frac{\partial \alpha^{(1)}}{\partial t} \quad \text{and} \quad \frac{\partial \alpha^{(2)} \rho^{(2)} E^{(2)}}{\partial t} = p^{(1)} \frac{\partial \alpha^{(1)}}{\partial t}. \quad (3.24)$$

An iterative method similar to [28] is used to solve these equations and compute an updated volume fraction $\alpha^{(1)}$ and energies E^K to obtain $p^{(2)} - p^{(1)} = \sigma \beta$. Unlike previous works without the surface tension, the energy exchange terms do not sum to zero, and the difference between the two, $\sigma \beta \theta^p \Delta p$, is a source/sink due to the interfacial surface tension energy.

- Gradient descent is used as an iterative scheme. However, an adaptive stepping makes the algorithm robust for the various EOSs considered in this work and for a wide range of pressures. Specifically, it is used for the gradient descent that would reduce the step size if either the pressures computed via the EOS (either $\kappa = 1, 2$) become negative during the iterations or the updated volume fraction goes beyond $[\epsilon, 1 - \epsilon]$ with $\epsilon = 10^{-10}$.

3.6. Interface compression

The interface numerically diffuses across multiple cells as the simulation evolves. Several techniques have been proposed to alleviate this. Modifications to the MUSCL face-reconstruction [84] or the limiter [77] and applications of higher-order schemes [52,85] have been useful for DIM. However, the DEM used in this work is a specialized technique, and we cannot directly apply it. Considering this, an interface compression scheme is borrowed here [70] which works independently of the other parts of the solver. Jain et al. [86] recently conducted a one-to-one comparison of various interface compression techniques and concluded that the choice would often depend on the modeling problem of interest. Their coupling with the seven-equation model can be considered in the future, but it is not addressed here.

The employed iterative technique for interface compression requires solving the following PDE in a pseudo-time τ :

$$\frac{\partial \alpha^{(1)}}{\partial \tau} = n_i \frac{\partial}{\partial x_i} \left(\epsilon_h \left| \frac{\partial \alpha^{(1)}}{\partial x_i} \right| - \alpha^{(1)} (1 - \alpha^{(1)}) \right) = n_i \frac{\partial f_{comp}}{\partial x_i}. \quad (3.25)$$

The right-hand side of this equation contains compression and diffusion terms and the interface normal, which balance each other when a steady-state is reached in the pseudo-time. This would result in a consistent thickness across the diffused interface, which would be a function of ϵ_h . Shukla et al. [70] used this approach for a five-equation DIM, where they had to use an additional correction for the mixture density. This approach is not required for the seven-equation model as the densities for the phases are computed separately.

In curvilinear coordinates Eq. (3.25) is written as

$$\frac{\partial \alpha^{(1)}}{\partial \tau} = \left(n_i \frac{\partial \xi}{\partial x_i} \frac{\partial f_{comp}}{\partial \xi} + n_i \frac{\partial \eta}{\partial x_i} \frac{\partial f_{comp}}{\partial \eta} + n_i \frac{\partial \zeta}{\partial x_i} \frac{\partial f_{comp}}{\partial \zeta} \right), \quad (3.26)$$

where, $\alpha^{(1)}$ is defined at the cell-centers and updates to it in the pseudo-time are denoted as $d\alpha_{ijk}^\tau$. First, f_{comp} in Eq. (3.25) is computed at the cell-faces. To compute $f_{comp,i+\frac{1}{2}jk}$ at the cell-faces, a centered averaging is used for $\alpha^{(1)}$ following Eq. (3.13), and the derivatives $\partial \alpha^{(1)} / \partial x_i$ are computed using Eq. (3.11) and Eq. (3.12). Instead of computing the derivatives $\partial \alpha^{(1)} / \partial x_i$ directly with $\alpha^{(1)}$, for numerical stability, they are computed as $\partial \phi / \partial x_i$ with the smooth field ϕ defined as previously in Eq. (3.22) [70].

As the next step, $\partial f_{comp} / \partial \xi$, etc., are obtained at the cell-centers via a centered differencing such as

$$\frac{\partial f_{comp}}{\partial \xi} \Big|_{ijk} = f_{comp,i+\frac{1}{2}jk} - f_{comp,i-\frac{1}{2}jk}. \quad (3.27)$$

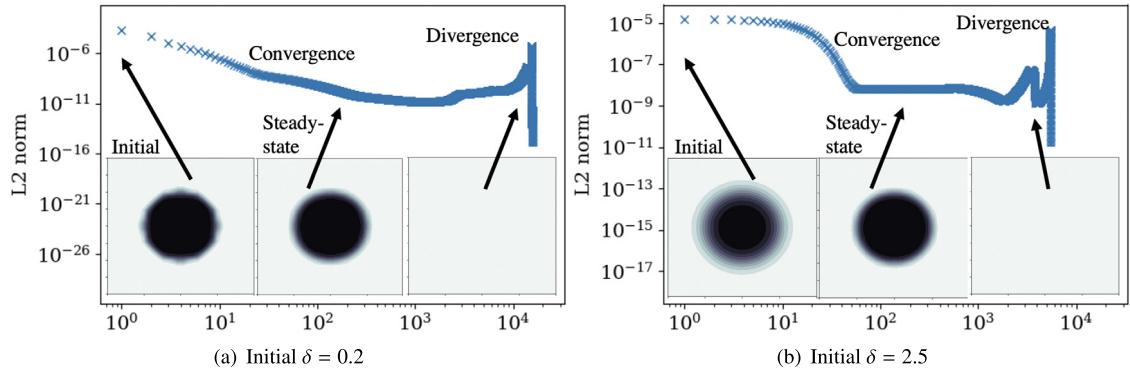


Fig. 4. L2 norms and volume fraction fields are shown with the interface compression algorithm iterations.

Similar forms are used for the j and the k -directions. Last, for the updates $d\alpha_{ijk}^\tau$ in Eq. (3.26), the interface normal

$$n_i = \frac{\partial \phi / \partial x_i}{|\partial \phi / \partial x_i|} \quad (3.28)$$

is computed at the cell-centers via Eq. (3.11), and the corresponding cell-centered derivatives are computed as $\partial \phi / \partial \zeta |_{i+\frac{1}{2},jk} = 1/2 (\phi_{i+1,jk} - \phi_{i-1,jk})$ (shown here representatively in the i -direction).

While applying the interface compression, the solution is evolved in the pseudo-time using explicit Euler until a steady-state solution is reached. The pseudo-time-step $d\tau$ is taken to be 0.2 for numerical stability [70]. The compression parameter ϵ_h is taken as $V_{ijk}^{1/3}/2$, which would result in the interface diffusing across 2-3 cells.

Eq. (3.25) is not in a strictly conservative form. A conservative form of this equation is available but it leads to spurious oscillations as also shown by Shukla et al. [70]. Recently, Jain et al. [71] showed success with the conservative form but by including the diffusion and the compression terms within the volume-fraction evolution equation and using a central differencing for the entire system of equations. Such an extension can be considered in the future, but for this work the interface compression is handled independently from the DEM for flux computation, and therefore, the non-conservative form of Shukla et al. [70] is retained. As a result, it was observed that when these pseudo-iterations are conducted for a very long time, the solution initially converges to a steady-state, but later it diverges. This is shown in Fig. 4 for example. For this test, a circular drop with diameter $d_0 = 9.6$ mm is placed at the center ($x_0 = 0, y_0 = 0$) of a square domain of size $[2d_0, 2d_0]$. The computational domain is uniformly divided into 30×30 cells. The volume fraction for the droplet is initialized using Eq. (3.23) with $\delta = 0.2$ and $\delta = 2.5$. The convergence is measured using an L_2 norm of $\alpha^{(1)}$ between the current and the previous iteration. For both the cases, initially, the iterations converge (in $\approx 10\text{-}30$ steps), and a steady-state solution is reached. However, after running this for a longer time, i.e. $> 10^3$ iterations, the solution diverges. The intermediate steady-state solution shows the interface diffused across 2-3 cells.

The solution to this problem in a standalone computation of the interface compression technique is straightforward. We could use an absolute or a relative tolerance limit to stop the iterations. However, while the compression scheme is being used as part of the DIM simulations, this is not possible. In addition to the interface compression, the flow fields are also being updated every time step, and even if the interface compression update was to be applied only once after every certain number (called n^{comp}) of simulation time-steps, the multiphase simulations can go on for an arbitrarily long number of iterations that could eventually cause this divergence. As a solution to this, after every n^{comp} simulation time-steps, the interface compression scheme is iterated for at least 3 pseudo-time-steps, and the volume fraction field is updated only if the L_2 norms during this monotonically decrease. If not, the solution is reverted to its original state without any interface compression updates. The numerical results show that the interface compression did not cause any divergence by using this approach even when we ran the simulations for an arbitrarily long time (up to 10^7 flow-field iterations). The interface compression was still active but acted only when necessary, and it maintained the interface thickness diffused only across 2-3 cells. $n^{comp} = 100$ is used in this work. The results for only a circular drop are shown here, but the interface compression scheme has also been validated for more complex shapes, e.g., ellipse, two lobes, etc. Those results are not shown here for brevity. With the non-conservative form of the interface compression used here, the errors observed in the mass conservation are $\sim 0.2\text{-}2\%$ (for initial $\delta = 0.2\text{-}2.5$) for the standalone tests shown in this section and $\sim 0.03\%$ for the DIM simulations with interface compression in Section 4.5.

3.7. Boundary conditions

Seven different treatments are used in this work at the computational domain boundaries: slip walls, no-slip walls, subsonic inflow, subsonic outflow, supersonic inflow, supersonic outflow, periodic. For the no-slip walls, the wall-normal and the tangential velocities of both the phases are set to zero at the boundary face. Only the wall-normal component is set

Table 2

List of verification and validation test cases for the seven-equation DIM.

Feature and ability	Test-case and approach
Baseline solver	Periodic pulse advection (1D) - Order of scheme - Non-disturbing condition [28,32] - Linear behavior
$p^{(2)}/p^{(1)}, \rho^{(2)}/\rho^{(1)} \gg 1$	Multiphase shock tube (1D) - Comparison against exact solution [32]
Arbitrary EOS & shocks	Shock transmission through (1D) - Air-Aluminum (CPG/SG) [88] - Air-HMX (CPG/MIEG) interface - Comparison against an exact solution
Dispersed & resolved MPE	Shock transmission through a particle curtain [89] and a resolved material interface (1D)
Surface tension	Δp in a circular drop [40,65] (2D) - Comparison against exact jump - Analysis of parasitic currents
Wave generated at interface	Oscillating elliptical droplet (2D) - Compared against exact frequency [40,65]
Viscous effects	Spherical bubble in high-pressure (3D) - Effect of relaxation terms - Comparison against analytical solution [90]
Droplet deformation due to a shock	Two-phase lid-driven cavity (2D) - Comparison against exact solution [91]
Reactive flows	Cylindrical and deforming droplet drag (2D) - Comparison against C_d correlations [92]
	Shock-droplet interaction (2D) - Comparison against experiments [35,93,94]
	Detonation-droplet interaction demonstration (2D)

to zero for the slip walls, whereas a zero gradient Neumann condition is applied for the tangential component. All the other quantities, i.e., the temperature, the species mass-fractions, and the volume fraction, are set using a zero gradient Neumann condition at the walls.

The eigenvalues of the seven-equation system being solved here are $\{u^{(1)} - c^{(1)}, u^{(1)}, u^{(1)} + c^{(1)}, u^{(2)} - c^{(2)}, u^{(2)}, u^{(2)} + c^{(2)}, u^I\}$ [28]. Correspondingly, seven waves are propagating in this hyperbolic system of equations. The inflow/outflow treatment is dependent on the wave-propagation directions at the boundaries. For supersonic flows, i.e., $u^{(1)}, u^{(2)} > c^{(1)}$ and $u^{(1)}, u^{(2)} > c^{(2)}$, all the waves propagate in the same direction. Therefore, a supersonic inflow, where all the waves enter the domain, is set using a Dirichlet condition. A supersonic outflow, where all the waves leave the domain, is set using a zero-gradient Neumann condition.

For subsonic inflow/outflows, where certain waves are entering the domain and certain that are leaving the domain, characteristic-based boundary conditions [87] are used. These were initially developed for multi-species single-phase flows, but the same form is used here for the two-phase flows. Since the wave-propagation speeds considered within this single-phase framework only correspond to the gas-phase wave speeds $\{u^{(1)} - c^{(1)}, u^{(1)}, u^{(1)} + c^{(1)}\}$, it is implicitly assumed that the other phase properties, as well as the volume fraction, are propagating at speed $u^{(1)}$ as if they are passive scalars. This is not a valid approach in general. Still, it is retained here, considering that only a single phase (typically the gas) is present at the inflow/outflow boundaries for the configurations studied. The development of a characteristic-based boundary condition for fully compressible two-phase flows is the subject of future work.

4. Results

Various features of the seven-equation DIM framework are evaluated in this section. Table 2 lists the tests that are considered in this work and the features and the abilities that they validate. For the seven-equation model, the phase-averaged flow fields for both the phases (i.e., ρ^k , p^k , etc.) are defined everywhere in the computational domain. For instance, even in a region where only the phase (1) is present, $\rho^{(2)}$, $p^{(2)}$ etc. are still mathematically defined. The corresponding volume fraction of the phase (2) would be very small, i.e., $\alpha^{(2)} = \epsilon$, and this would result $f^{mix} = \sum_{k=1}^2 \alpha^k f^k \approx f^{(1)}$, so the phase (2) values, i.e., $\rho^{(2)}$, $p^{(2)}$, etc., aren't expected to affect the mixture flow-fields with $\epsilon \rightarrow 0$, but they still have to be defined and initialized. To avoid a division by zero and to make sure that the numerical solution does not get corrupted, the corresponding ϵ has to be larger than the machine precision (double precision used here, 10^{-15}). Furthermore, even small errors in $\alpha^{(k)} f^{(k)}$ (dependent on the numerical scheme) can amplify, as large errors in $f^{(k)}$ with small $\alpha^{(1)}$, and it can lead to robustness problems when f^k are used for equation of state, sound-speed computations, etc. Considering this, $\epsilon = 10^{-6}$

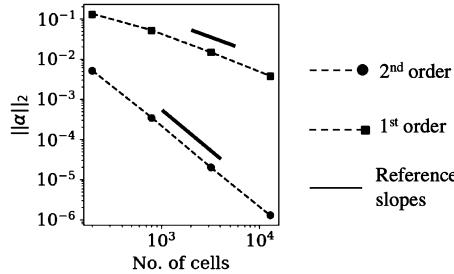


Fig. 5. Normalized L_2 error norms of the volume fraction (see Eq. (4.1)) for the 1D pulse advection test are shown with different grid sizes and numerical scheme orders. The solid lines have 1/1 and 1/2 slopes for reference.

is retained here for most simulations in this work to be consistent with Abgrall and Saurel [32]. The effect of ϵ is evaluated further in Section 4.3 for the shock impacting a material interface.

4.1. Advection

A 1D periodic computational domain of length $L = 1$ m is used for this test. The volume fraction $\alpha^{(1)}$ is initialized (at $t = t_0$) as a sine function varying between 0.25 and 0.75 as $\alpha^{(1)}(x) = (1/4) \sin(x/(2\pi L)) + 1/2$. Volume fraction for the other phase is set as $\alpha^{(2)} = 1 - \alpha^{(1)}$. Uniform pressure and velocity are specified as $p^{(1)} = p^{(2)} = 10^5$ Pa and $u^{(1)} = u^{(2)} = 100$ m/s. The densities are set as $\rho^{(1)} = 1.0$ kg/m³ and $\rho^{(2)} = 10^3$ kg/m³. The EOS for phase (1) is specified as CPG, with $\gamma^{(1)} = 1.4$, and for phase (2) it is specified as SG with $\gamma^{(2)} = 4.4$ and $p_0^{(2)} = 6 \times 10^8$ Pa. The surface tension, the viscous terms, and the interface compression are absent for this test, and a single specie is used for both the phases. Four different grids are tested with the number of cells varying as $N_x = 200, 800, 3200, 12800$ as well as the first- and the second-order accurate schemes. Simulations are conducted both with and without stiff relaxation.

The volume fraction pulse is allowed to advect in the periodic domain for 14 flow-through times ($t = 14t_f$). The solution at the end is compared against the initial solution ($t = t_0$), which serves as a ‘truth’ solution for this case. The ‘truth’ solutions are denoted with a superscript ‘truth’. The L_2 error norms for any field-variable f are computed as

$$\|f\|_2 = \sqrt{\frac{1}{N_x} \sum_{i_x=1}^{N_x} (f(i_x, t = 14t_f) - f^{truth}(i_x))^2}. \quad (4.1)$$

As noted in earlier studies [32], a “non-disturbing” condition for this case means that regardless of the volume fraction in the domain, since the simulation starts with a uniform pressure/velocity, one should maintain a uniform pressure/velocity at all times with machine accuracy. Both the pressure and the velocity maintain uniform values with $\|p^\kappa\|_2/p^{(1)} < 10^{-15}$ and $\|u^\kappa\|_2/u^{(1)} < 10^{-15}$. The volume fraction L_2 error norms are plotted in Fig. 5. The volume fraction suffers numerical diffusion. However, the scheme order of accuracy is maintained as expected. The differences in the reference and the numerical order of accuracy slopes are less than 5% suggesting a reasonable match. The results shown here are without using the stiff relaxation solver. Since the uniform pressure/velocity condition is maintained between both the phases anyway, using the stiff relaxation solver did not make any difference. Those results are not shown here for brevity.

4.2. Multiphase shock tube

The objective of this test is to evaluate the method’s ability to deal with sharp jumps across a material interface [32]. A 1D computational domain of length $L = 1$ m is used. A phase interface is initialized at $x/L = 0.8$. The flow variables at the initial time (t_0) are set as $\alpha^{(1)} = \epsilon$, $p^{(2)} = p^{(1)} = 2 \times 10^8$ Pa for $x/L < 0.8$, and $\alpha^{(2)} = \epsilon$, $p^{(2)} = p^{(1)} = 10^5$ Pa for $x/L \geq 0.8$. The initial velocities are $u^{(1)} = u^{(2)} = 0$ m/s everywhere. The initial densities are set as $\rho^{(1)} = 50$ kg/m³ and $\rho^{(2)} = 1000$ kg/m³ everywhere. Based on this, the pressure and the velocity equilibrium are already maintained at t_0 . The phase (1) is modeled as CPG with $\gamma^{(1)} = 1.4$ to mimic compressed air, whereas the phase (2) is modeled as SG with $\gamma^{(2)} = 4.4$, $p_0^{(2)} = 6 \times 10^8$ Pa to mimic liquid water. Supersonic outflows are used at both ends of the tube to let the waves pass without reflection. Surface tension, viscous terms, and interface compression are absent for this test. Four different grids with the number of cells $N_x = 200, 500, 1000, 2000$ are tested with both the first- and the second-order-accurate schemes.

The simulation is conducted for 0.2 ms, and the final solution is compared against an analytical ‘truth’ solution computed via an exact Riemann solver as shown in Fig. 6. A shock propagates on the right side of the interface (in air), and a rarefaction travels towards the left (in water). The contact wave, and therefore, the interface, also moves slightly towards the right. Note that the flow variables for both phases ($\kappa = 1, 2$) are evolved for the seven-equation model. However, for the resolved interface simulations that are considered here, only the mixture properties computed as f^{mix} are of relevance, and only those are compared. The simulation results match the exact solution with increasing accuracy with an increase in the grid resolution and the order of the scheme (see Fig. 6). For instance, the normalized L_2 error norm of the pressure

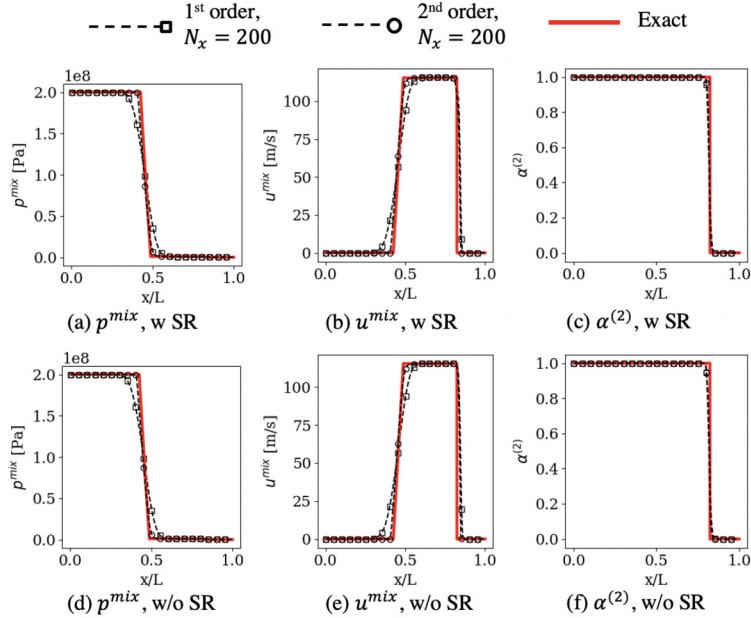


Fig. 6. Results for a multiphase shock tube using the seven-equation DIM are compared against an exact solution.

($\|p^{mix}\|_2/p_{x/L<0.8}^{(2)}$) reduces from 0.07 to 0.03 going from $N_x = 200$ to $N_x = 1000$ for the first order scheme, and almost a similar improvement in the accuracy is obtained for $N_x = 200$ by going from the first to the second order scheme.

Results with and without the stiff relaxation (SR) solver match each other closely at all grid resolutions and scheme orders. For example, the normalized L_2 norm of the difference in p^{mix} with and without the SR for $N_x = 200$ and the second-order-accurate scheme is 0.002. This shows that the use of the DEM relieves the need to use a separate stiff relaxation solver. The DEM accurately models the interphase exchanges. The pressure and the velocity equality between the phases are obtained within the numerically diffused interface ([32] for further details) even without enforcing it. When a stiff pressure relaxation is applied the pressure and the velocity equality are enforced everywhere, and therefore, $p^{mix} = p^1 = p^2$, $u^{mix} = u^1 = u^2$. However, without it, p^1 , p^2 and u^1 , u^2 can be different away from the numerically diffused interface, but $f^{mix} \approx f$ is still be maintained in the pure fluid regions as $\alpha^\kappa \approx 1$.

4.3. Shock interaction with material interface

The passage of a shock through a material interface is simulated in this test. A 1D computational domain of length $L = 1$ m is used. The region $x/L < 0.5$ is filled with air ($\alpha^{(2)} = \epsilon$, CPG, $\gamma^{(1)} = 1.4$), whereas, $x/L > 0.5$ is Aluminum ($\alpha^{(1)} = \epsilon$, SG, $p_0^{(2)} = 21.13 \times 10^9$ Pa, $\gamma^{(2)} = 3.8$). A shock of strength Mach 2 is initialized in the air at $x/L = 0.3$ to propagate towards the material interface. The pre-shock conditions are set to $p^{(1)} = p^{(2)} = 10^5$ Pa, $u^{(1)} = u^{(2)} = 0$ m/s, and $\rho^{(1)} = 1.2$ kg/m³, $\rho^{(2)} = 2784$ kg/m³. The post-shock conditions for $x/L < 0.3$ are computed using the Rankine-Hugoniot relations for air as $p^{(1)} = 4.56 \times 10^5$ Pa, $u^{(1)} = 429.0$ m/s, and $\rho^{(1)} = 3.211$ kg/m³.

Only pure air is present for $x/L < 0.3$ in reality. However, we also need to specify the flow fields for phase (2) for the seven-equation numerical modeling. The different initial conditions (IC) attempt to understand their effects on the final solution.

- IC-1: $\epsilon = 10^{-3}$, $p^{(2)} = p^{(1)}$ and $u^{(2)} = u^{(1)}$ for $x/L < 0.3$
- IC-2: $\epsilon = 10^{-6}$, $p^{(2)} = p^{(1)}$ and $u^{(2)} = u^{(1)}$ for $x/L < 0.3$
- IC-3: $\epsilon = 10^{-9}$, $p^{(2)} = p^{(1)}$ and $u^{(2)} = u^{(1)}$ for $x/L < 0.3$
- IC-4: $\epsilon = 10^{-12}$, $p^{(2)} = p^{(1)}$ and $u^{(2)} = u^{(1)}$ for $x/L < 0.3$
- IC-5: $\epsilon = 10^{-15}$, $p^{(2)} = p^{(1)}$ and $u^{(2)} = u^{(1)}$ for $x/L < 0.3$
- IC-6: $\epsilon = 10^{-3}$, $p^{(2)} = 10^5$ Pa and $u^{(2)} = 0$ for $x/L < 0.3$.

The first five IC compare the effect of ϵ . The pressure and the velocity equilibria are already attained between the phases at t_0 for these. These IC can cause a potential problem since the pre/post-shock conditions that are specified correspond to a Mach 2 shock in the air, and they are not for the liquid, which still has a volume fraction of ϵ in this region. This would create a Riemann problem for the liquid resulting in artificial waves even before the shock hits the material interface. IC-6 aims at solving this problem by initializing phase (2) with uniform pressure/velocity even though phase (1) is initialized as a shock. This means that the SR cannot be enforced for this, but there would not be any artificial waves at the start.

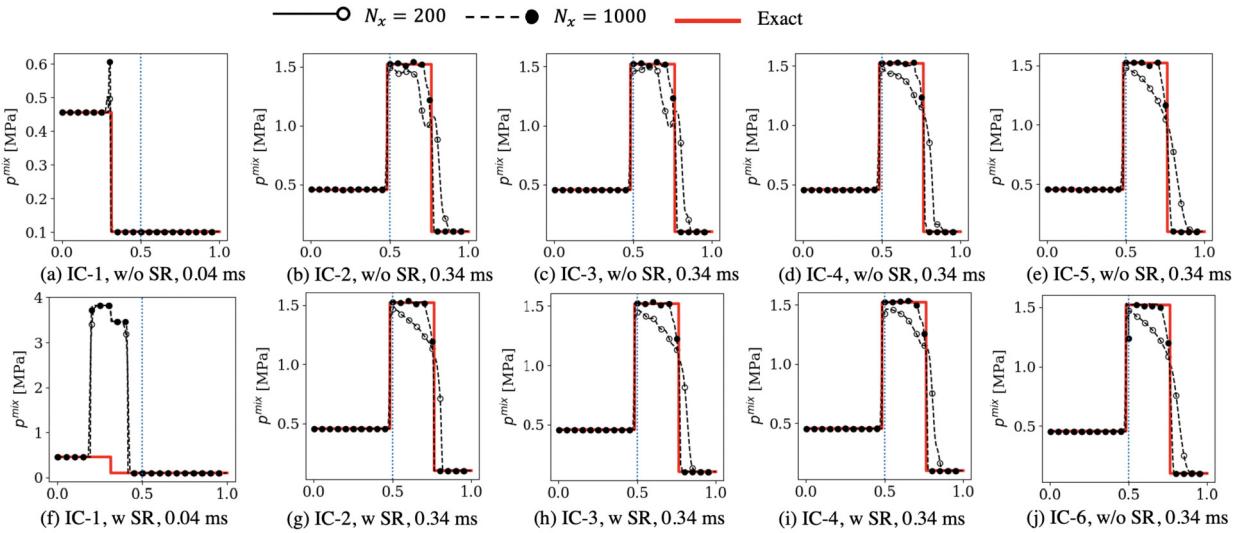


Fig. 7. The p^{mix} is shown for the 1D shock interacting with Air/Al material interface. The vertical dashed line shows the location of the material interface. The results shown here are with the second-order scheme. The results for IC-1 are shown at an earlier time. Simulations with IC-5 and IC-6 with the SR were not possible and they are not shown.

The results at $t = 0.3$ ms, after the shock has imparted the material interface, are analyzed and compared against an analytical solution [88] in Fig. 7. Depending on the impedance of the materials and the shock strength, the imparted shock would reflect/transmit as either a shock or a rarefaction. In this case, both the reflected and the transmitted waves are shocks. Due to the shock impact, the material interface also moves, although for this case, its velocity is only 0.095 m/s, and it is not noticeable from these plots.

The coarse-grid prediction ($N_x = 200$) shows noticeable errors in the transmitted shock, but the results improve with grid refinement. This is a challenging test due to the high density ratio ($\rho^{(2)}/\rho^{(1)} = 2320$) at the material interface on which the shock is impacted. As per the authors' knowledge, other multiphase modeling predictions for this configuration are not available in the literature for a one-to-one comparison. However, such comparison is provided later in Section 4.8 for a 2D shock-droplet interaction test. Either a higher-order extension of the discrete equations method (DEM), or an adaptive mesh refinement (AMR) can be considered in the future to improve the solution accuracy.

In terms of the effect of the IC, due to the inconsistent pre/post-shock conditions as discussed before and a higher ϵ , as expected, IC-1 shows significant errors resulting from the artificial waves created in the phase (2) near the initial shock location ($x/L = 0.3$). The errors are already present at $t=0.04$ ms, well before the shock hits the interface, and therefore, the results at later times are also corrupted (not shown here). All, IC-2, IC-3, IC-4, IC-5, show a significant improvement over IC-1. Since ϵ is smaller for them, even though the artificial waves are created in phase (2) initially, their effects on the mixture quantities, i.e., p^{mix}, u^{mix} seem to be minimal. IC-6 also seems to be a valid initialization approach for the seven-equation model and captures the shock transmission/reflection through the material interface, even with $\epsilon = 10^{-3}$ as there are no artificial waves due to the initialization anymore. However, as noted before, this IC cannot be used with a SR solver since it is not enforced at the t_0 .

The predictions for $\epsilon = 10^{-6}, 10^{-9}, 10^{-12}$ (IC-2, IC-3, IC-4) are similar to each other both with and without a relaxation solver. At the lower limit of ϵ , the simulations with $\epsilon = 10^{-15}$ are able to predict the mixture quantities (p^{mix}, u^{mix}) without the relaxation solver, but the phase-specific pressure of the gas-phase, i.e., $p^{(1)}$, shows a spike (~ 0.2 MPa) in the liquid-phase region (not shown here) that can lead to a crash in the EOS solver for more complex problems. On the other hand, the simulation with $\epsilon = 10^{-15}$ with the SR fail at the first step due to the failure of the iterative solver. Considering this, the choice of $\epsilon = 10^{-6}$ is justified for the other cases, although the results show that using $\epsilon = 10^{-9}, 10^{-12}$ is also possible.

The results shown here are only for the second-order scheme. Like the multiphase shock tube test earlier, the results with and without a stiff relaxation (SR) show a similar trend and reasonably match. For the coarse grid ($N_x = 200$), with $\epsilon = 10^{-6}, 10^{-9}$ (IC-2, IC-3), without the SR, the predictions do show numerical oscillations in the transmitted shock, whereas, the lower ϵ values (IC-4, IC-5) and the simulations with the SR show primarily diffusive errors. The predictions improve for the fine grid. This can be attributed to errors in modeling the non-conservative interface terms I^K , since in absence of the relaxation terms, they are entirely responsible for handling the interphase exchange and maintaining the pressure and the velocity equilibrium.

Another test, where a shock initiated in the Aluminum enters the air, is also simulated. To initiate the shock in Aluminum, the driven section is set to $p^{(1)} = p^{(2)} = 10^5$ Pa, the driver section is set to $p^{(1)} = p^{(2)} = 10^7$ Pa; and $u^{(1)} = u^{(2)} = 0$ m/s, $\rho^{(1)} = 1.2$ kg/m³, and $\rho^{(2)} = 2784$ kg/m³ are set everywhere in the domain. The results are shown in Fig. 8. The transmitted wave is a shock (not noticeable here), and the reflected wave is a rarefaction as expected due to the lower impedance of

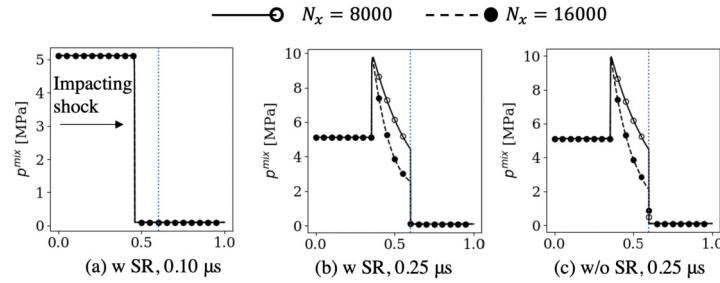


Fig. 8. The p^{mix} is shown for the 1D shock interacting with Al/Air material interface. The vertical dashed line shows the location of the material interface. The results shown here are with the second-order scheme.

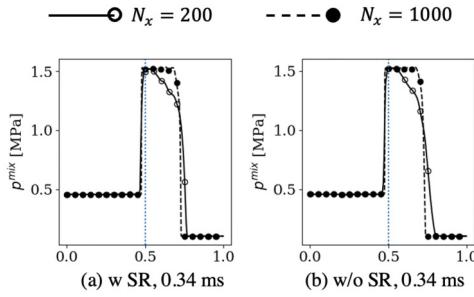


Fig. 9. The p^{mix} is shown for the 1D shock interacting with Air/HMX material interface. The vertical dashed line shows the location of the material interface. The results shown here are with the second-order scheme.

the air compared Aluminum [88]. Capturing the reflected rarefaction wave requires a fine grid, and the results are still similar with or without the SR. An exact solution was not available at these conditions for a comparison.

Next, to demonstrate the ability of this method to handle arbitrary EOS, the Al that was previously modeled using SG is now replaced with HMX, which is modeled using an MG EOS with the same parameters as used in an earlier work [78]. Unfortunately, an analytical solution to compare is not available here, but the obtained results are shown in Fig. 9. The results look qualitatively similar to the Al/air test, where the impacted shock resulted in a transmitted and a reflected shock at the interface. These results are obtained with the IC-2 approach, and they prove the robustness of this method to handle arbitrary EOS and large pressure and density ratios.

4.4. Shock interaction with unresolved MPEs and a resolved interface

This test is considered to demonstrate the unique potential of the seven-equation model due to the velocity non-equilibrium that is allowed between the phases. As noted earlier, carrying phase-independent velocities can allow the same method to be used for modeling both resolved and unresolved multiphase flows. The configuration of this numerical test is created by combining the experimental setup of Rogue et al. [89], where a shock interacts with a bed of small-sized particles that cannot be resolved on the computational grid, with another setup considered earlier in Section 4.3 of this work, where a shock impacts on a resolved material interface. Dispersion of the unresolved particles has been modeled successfully in the past using dispersed phase approaches such as Eulerian-Eulerian (EE) or Eulerian-Lagrangian (EL) [8,66], and the shock impact on a material interface also has been modeled using the reduced five/six-equation DIM [88], however, here, we model both with the same seven-equation model.

A shock tube of dimensions $6 \text{ m} \times 13 \text{ cm} \times 13 \text{ cm}$ is initialized with a Mach 1.3 shock at $x = 1 \text{ m}$, and a glass particle bed with particle radius $r_p = 750.0 \mu\text{m}$ and volume loading $\alpha^{(2)} = 0.65$ is placed at $x \in [2.00, 2.02] \text{ m}$. Initial pressure and temperature in the driven section ($x > 1 \text{ m}$) are 10^5 Pa and 298 K , respectively. Pressure in the driver section ($x < 1 \text{ m}$) is set corresponding to normal shock relations. Modifying the original experimental setup, a glass slab ($\alpha^{(2)} = 1 - \epsilon$) is placed from $x \in [4.00, 5.00] \text{ m}$, downstream of the unresolved particles. The computational domain is resolved with 3000 cells in 1D, resulting in a grid size of 2 mm. A symmetric boundary condition is used at $x = 0 \text{ m}$ and $x = 6 \text{ m}$ is set as a Neumann boundary condition. Both resolved and unresolved MPE are present in the same computational domain, and the same set of DIM Eq. (2.11) are solved everywhere. However, since the dispersed particles are not resolved on the computational grid, the velocity non-equilibrium between the phases in this region is modeled using a drag law. Correspondingly, in the region with dispersed particles, i.e., $x < 3 \text{ m}$, the velocity relaxation term in Eq. (2.12) is modeled as

$$\theta^u \Delta u_i = \frac{3\alpha^{(2)}}{8r_p} C_D \rho^{(1)} (u_i^{(2)} - u_i^{(1)}) |u_i^{(2)} - u_i^{(1)}|. \quad (4.2a)$$

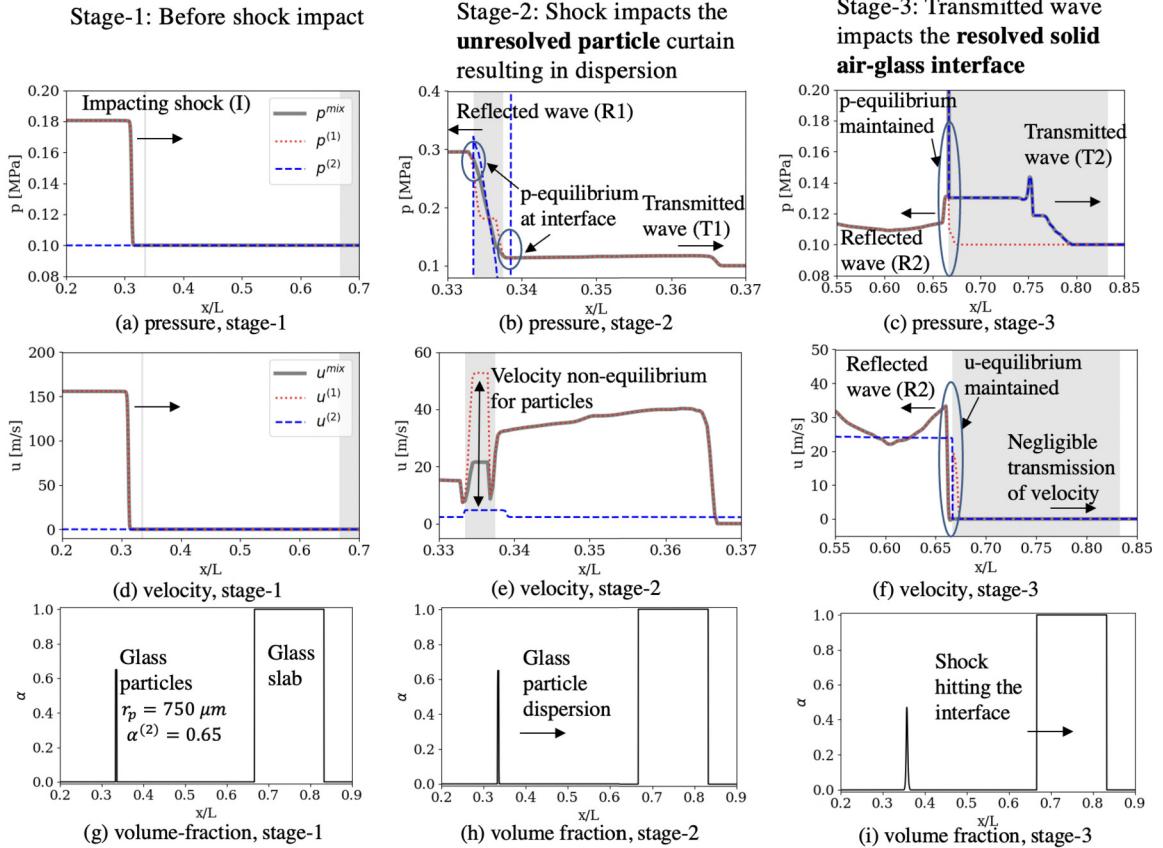


Fig. 10. Results are shown for a shock interacting with dispersed and resolved multiphase entities within the same domain. The gray regions in subfigures (a-f) denote the presence of the particles or the glass slab.

Here, C_D is the particle drag coefficient that is computed as a correlation function of the particle Reynolds number Re_p following Crowe et al. [95]. For this test, the SR solver is not used in the rest of the domain, although it is possible to use it if needed in the regions away from the dispersed particles. This test is a first step towards demonstrating the potential of using the seven-equation model for both resolved and unresolved MPE, there are several improvements that will be needed in the future for its full applicability to problems of practical relevance.

- The dispersed particles and the resolved interface are well-separated for this test and they do not directly interact with each other. As a result, for the current demonstration, it is possible to use a drag law only for $x < 3$ m. For its use in real applications, a hybridization of θ^u may have to be considered so that it is dynamically determined based on the local conditions.
- As identified by Saurel et al. [66], the BN seven-equation model results in physically inaccurate wave-speeds for dispersed particles, and a corresponding hybridization of the volume-fraction evolution equation [66] may have to be considered for dense-to-dilute modeling.
- Some additional features that are included in this work, e.g., surface tension, interface compression, etc., may not be relevant in regions with unresolved droplets and they would have to be dynamically switched off in these regions with dispersed particles.

While acknowledging these limitations of the current demonstration, the simulation results are shown in Fig. 10. Stage-1 corresponds to $t = 2$ ms, when the initial shock (I) is yet to impact the dispersed particle curtain. Stage-2 corresponds to $t = 4$ ms, which is after initial shock (I) has impacted the dispersed particle curtain, and after the impact, a transmitted wave (T1) continues to propagate downstream (see Fig. 10(b)). A reflected wave (R1) is also generated and it propagates back upstream that is not shown here. Even though the SR for pressure is not used here, a pressure equilibrium between the phases is maintained at the particle curtain boundary as shown in Fig. 10(b) due to the DEM modeling of the non-conservative terms I^K . A velocity non-equilibrium ($u^{(2)} - u^{(1)} \approx 50$ m/s) is observed in the dispersed particle region (see Fig. 10(e)), with the mixture velocity lying between the two phasic velocities. The specified drag law acts towards relaxing this difference in finite time. Capturing this velocity non-equilibrium and the corresponding particle dispersion would not have been possible with five/six-equation models that analytically enforce a velocity equilibrium.

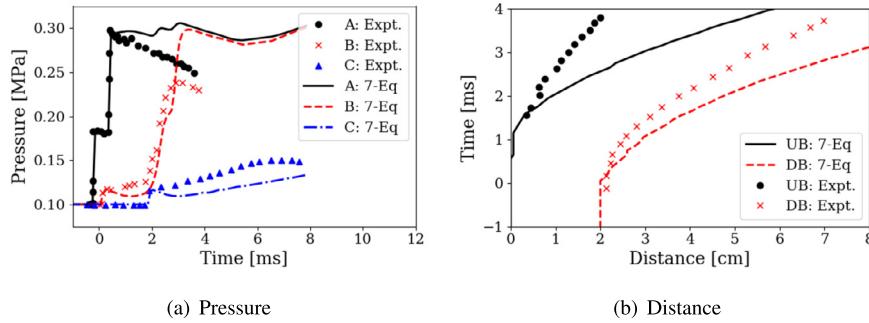


Fig. 11. Simulation results for dispersed particle shock interaction with the seven-equation model are compared against experiments of Rogue et al. [89]. Subfigures (a) compares pressure traces at locations 1.89 m (A), 2.04 m (B) and 2.72 m (C), respectively. Subfigure (b) shows time evolution of upper and lower boundaries of the particle curtain.

To validate the interaction of (I) with the dispersed particle curtain, a comparison against experimental data is provided in Fig. 11. This includes pressure traces at three locations, 1 cm upstream of the initial particle slab-start location (point A), and 2 cm (point B) and 70 cm (point C) downstream of the slab-end location. Pressure values at points A and C are a result of shock reflection and transmission through the particle curtain. The pressure at point B, that is located immediately downstream of the initial particle bed, is strongly affected by the dispersion of the particles. The first jump in pressure at point A is due to incident shock, and the second jump is from reflection of this wave through the curtain. Pressure jump at point C at $t = 2$ ms is due to arrival of the transmitted wave through the curtain. The pressure transmission and reflection are predicted reasonably well using the seven-equation model. The particle curtain displacement is over-predicted, but this behavior is consistent with the results of Saurel et al. [66], where they show that these errors are due to physically inaccurate wave-speeds of the seven-equation model in the region with dispersed particles as previously noted.

The transmitted wave (T1) continues to propagate and hits the resolved material interface during stage-3 ($t = 6$ ms). This process is qualitatively similar to the shock impact discussed in the previous section, but the wave (T1) is significantly weak here due to its earlier interaction with the particle curtain. After the impact of (T1) with the resolved material interface, a transmitted wave (T2) propagates into the glass, and a reflected wave (R2) travels upstream back into the air (see Fig. 10(c)). Due to the high density (and therefore, high impedance) of the glass as compared to air, there is a negligible transmission of velocity into the glass (see Fig. 10(f)). Even though SR is not used here, the pressure and the velocity equilibrium are maintained at the interface due to the DEM modeling of I^K .

4.5. Surface tension effects

Several tests are conducted for validating the surface tension modeling following Nguyen and Dumbser [65]. A 2D square computational domain of width $L = 1$ m is filled with air for the first test. A circular liquid droplet of radius $r_0 = 0.1549$ m is placed at its center. Slip walls are used as boundary conditions. As a result of the droplet curvature, higher pressure should be maintained within the droplet. The objective of the first test is to evaluate the ability of the DEM surface tension modifications to handle and maintain this pressure jump. Instead of computing the curvature from the volume fraction field, it is set to $\beta = 1/r_0$ everywhere. The volume fractions inside and outside the droplet are initialized as $\alpha^{(1)} = \epsilon$ and $\alpha^{(2)} = \epsilon$. The velocities are initialized as zero everywhere. The pressures are set as $p^{(1)} = 10^3$ Pa and $p^{(2)} = p^{(1)} + \sigma\beta$ everywhere, and this would result in a jump in p^{mix} at the droplet interface. In terms of the DEM flux computation, this is appropriate since this means that the pressure jump is only present at a phase (1)–(2) interface, where we included modifications for the surface tension. Whereas the pressure is uniform across the (1)–(1) and the (2)–(2) interfaces. The gas and the liquid phases are modeled using CPG and SG, respectively, with $\gamma^{(1)} = 1.4$, $\gamma^{(2)} = 4.4$, $p_0^{(2)} = 6 \times 10^8$ Pa. The surface tension is set to $\sigma = 342$ N/m. The gas-phase and the liquid-phase densities are $\rho^{(1)} = 1.0$ kg/m³ and $\rho^{(2)} = 100$ kg/m³, respectively. This test is conducted both with and without the stiff relaxation solver. However, the pressure difference $p^{(2)} - p^{(1)}$ is already maintained to $p_\sigma = \sigma\beta$ from the start, which would mean that the relaxation solver does not have to act.

Three different grids, 200×200 , 400×400 , and 800×800 and both the first and the second-order schemes are tested. The simulations evolve until a physical time of 0.085 s, and at the end of it, the L_2 error norms for p^{mix} , u^{mix} and $\alpha^{(2)}$ are computed considering the initial conditions as the truth. For all the grids, for all the orders of the scheme, with or without the stiff relaxation solver, and with or without the interface compression, the normalized errors are maintained to machine accuracy ($< 10^{-15}$), suggesting that the DEM modifications for the surface tension can handle the surface jump condition exactly.

For the second test, the same setup is used. However, the curvature is computed from the volume fraction field as described in Section 3.4 instead of setting it as a constant. The pressure within the domain is initially set as $p^{(1)} = p^{(2)} = p_{atm} = 10^3$ Pa, and as a result of the surface tension force, it is expected to rise within the droplet (ideally to $\sigma\beta = 2207$ Pa). The simulations are run for 0.034 s ($\sim 40,000$ steps for the 800×800 grid), and the pressure jump within the droplet at the end of it is measured by integrating over the entire computational domain Ω as

Table 3

Errors Δp^{err} [Pa], KE^{mix} [$\text{kg m}^2/\text{s}^2$], and L_2 for the circular droplet surface tension test are shown for various grids and various simulation options. Simulation results without the stiff relaxation solver (SR) and with the interface compression (comp.) are highlighted as they show the lowest errors.

Case	200×200	400×400	800×800
p^{err} [Pa]			
w/o SR, w/o comp.	1.06×10^2	2.64×10^2	2.09×10^2
w SR, w/o comp.	8.84×10^2	1.18×10^3	1.25×10^3
w/o SR, w comp.	3.52×10^1	1.49×10^1	5.86×10^1
w SR, w comp.	1.44×10^2	1.01×10^3	4.66×10^2
KE^{mix} [$\text{kg m}^2/\text{s}^2$]			
w/o SR, w/o comp.	7.99×10^{-6}	5.92×10^{-5}	4.35×10^{-5}
w SR, w/o comp.	1.58×10^{-4}	1.56×10^{-3}	2.18×10^{-3}
w/o SR, w comp.	2.26×10^{-6}	2.06×10^{-6}	2.10×10^{-6}
w SR, w comp.	1.68×10^{-4}	4.59×10^{-4}	1.49×10^{-3}
L_2 [Pa]			
w/o SR, w/o comp.	6.57×10^1	8.87×10^1	7.22×10^1
w SR, w/o comp.	2.58×10^2	3.45×10^2	3.84×10^2
w/o SR, w comp.	6.61×10^1	4.83×10^1	4.41×10^1
w SR, w comp.	1.09×10^2	2.96×10^2	2.46×10^2

$$\Delta p = \int_{\Omega} p^{(2)} \alpha^{(2)} / \int_{\Omega} \alpha^{(2)} - \int_{\Omega} p^{(1)} \alpha^{(1)} / \int_{\Omega} \alpha^{(1)}. \quad (4.3)$$

This is compared against the exact pressure difference based on Young-Laplace relation as $\Delta p^{err} = |\Delta p - \sigma \beta|$. Even though the background flow should have been at rest, parasitic currents are observed due to the numerical errors in the curvature computation. They are consistent with the previous works [65]. These are quantified using the kinetic energy as

$$KE^{mix} = \int_{\Omega} \left(\frac{1}{2} \alpha^{(1)} \rho^{(1)} u^{(1)2} + \frac{1}{2} \alpha^{(2)} \rho^{(2)} u^{(2)2} \right) / \int_{\Omega} (\alpha^{(1)} + \alpha^{(2)}). \quad (4.4)$$

A L_2 norm of the mixture pressure p^{mix} is computed as

$$L_2 = \sqrt{\frac{1}{\sigma \beta} \int_{\Omega} (p^{mix} - p^{exact})^2}, \quad (4.5)$$

where p^{exact} is p_{atm} for $r \geq r_0$ and $p_{atm} + \sigma \beta$ for $r < r_0$.

Values of Δp^{err} , KE^{mix} , and L_2 are reported in Table 3 for various grids and with/without the stiff relaxation (SR) and with/without the interface compression (comp.). Only the second-order scheme results are shown here. However, the results with the first-order scheme are also consistent with the conclusions here. The lowest errors are observed for the case without the SR but with the interface compression. The errors in the computed pressure jump are less than 2% for all the grids for this case, and the parasitic velocities are < 0.2 m/s. The parasitic currents and the p^{mix} within the droplet for this case are shown in Fig. 12 for reference.

Regardless of the interface compression, the errors with the SR are at least an order of magnitude higher than those without the SR. Further insights into this are provided below. The errors in p^{err} and KE^{mix} do not necessarily reduce with the grid, and this has also been observed in a past work [65]. This could be due to contrasting effects; as we would compute the curvature more accurately for the finer grids, however, the coarser grids would help the parasitic currents to diffuse numerically. The errors with the interface compression are also significantly lower than those without it since the droplet interface becomes irregular due to the parasitic currents. Still, the interface compression helps keep it regular, resulting in a more accurate curvature computation. The errors in L_2 , that have been used in the past to show grid convergence [65], do consistently reduce with grid refinement for the simulations without the SR and with the interface compression, but not for the other options. This suggests not using the SR and using the interface compression as the optimal choice, while modeling surface tension effects within our seven-equation numerical framework.

To understand the reasons for the large errors with the SR, p^k , ρ^k , and $\alpha^{(2)}$ are plotted along the centerline for the 800×800 test with and without the SR in Fig. 13. As noted before, the DEM modifications in the surface tension are designed

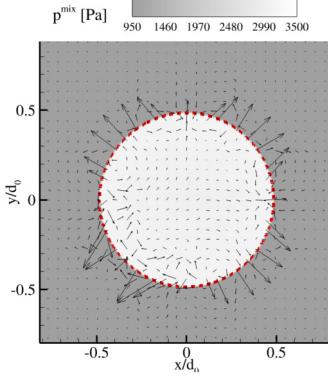


Fig. 12. Velocity vectors and p^{mix} contours are shown for the 800×800 cylindrical droplet case without the stiff relaxation and interface compression at $t=0.085$ s. The droplet interface $\alpha^{(2)} = 0.5$ is shown via a dashed line.

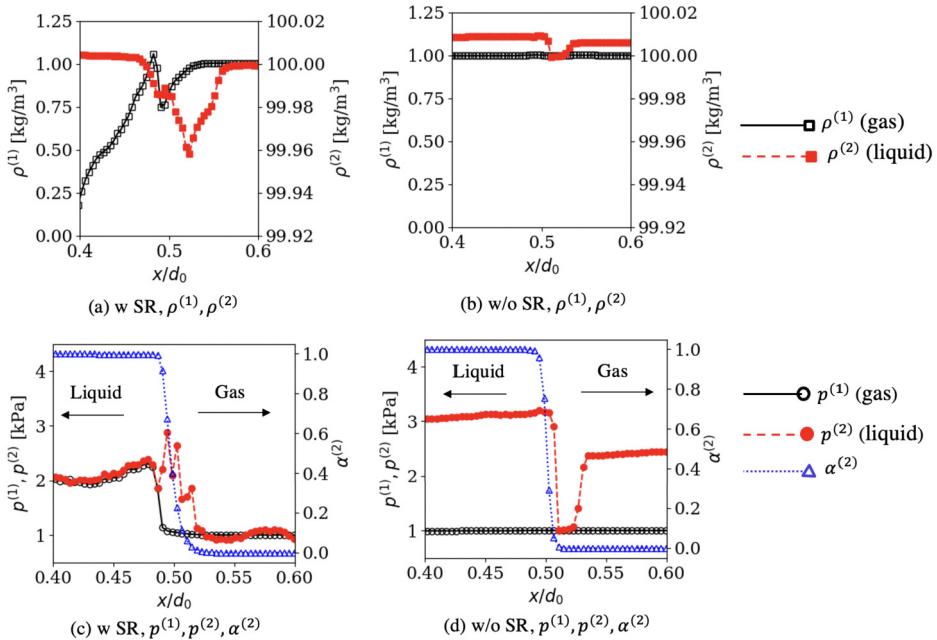


Fig. 13. $p^{(1)}$, $p^{(2)}$, $\rho^{(1)}$, $\rho^{(2)}$, and $\alpha^{(2)}$ are plotted along the centerline for the 800×800 test with and without the SR.

so that a pressure jump is handled across the (1)-(2) interface, whereas the (1)-(1) and the (2)-(2) interfaces should behave as before. As expected, the case with the SR maintains pressure equality away from the interface and enforces a pressure jump between $p^{(1)}$ and $p^{(2)}$ within the numerically diffused interface. However, because of this, pressure jumps are present even in the phasic fields $p^{(1)}$ and $p^{(2)}$ near the boundaries of the diffused interface. They result in pressure jumps at the (1)-(1) and the (2)-(2) interfaces within the DEM (in addition to the jump at (1)-(2) interface which is expected), possibly causing these larger errors.

On the other hand, for the case without the SR, $p^{(1)}$ stays constant, and only the $p^{(2)}$ changes within the numerically diffused interface. The liquid pressure $p^{(2)}$ reduces almost to the same value as $p^{(1)}$ right before the interface ($x/d_0 > 0.5$) and then increases back due to the surface tension terms at the interface. The initial reduction in it is still in the gas phase (i.e., $\alpha^{(2)}$ is small), and therefore, it does not adversely affect the overall mixture (*mix*) flow fields. The surface tension is accurately captured without the SR within the numerically diffused interface due to the DEM modifications in the (1)-(2) Riemann problem. The errors mentioned above due to the SR are also apparent in the density fields, as $\rho^{(1)}$ in the presence of the SR increases by almost 100% within the droplet. In contrast, the oscillations in both $\rho^{(1)}$ and $\rho^{(2)}$ are $< 0.5\%$ for the case without the SR.

For the third test, instead of starting from a circular droplet, an elliptical drop is used with the following shape [40,65]

$$\frac{(x - x_0)^2}{0.2^2} + \frac{(y - y_0)^2}{0.12^2} = 1. \quad (4.6)$$

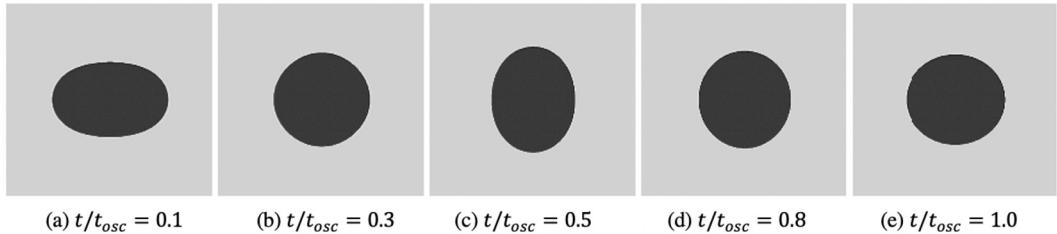


Fig. 14. Oscillating ellipsoidal droplet at various times for the 800×800 grid without the interface compression.

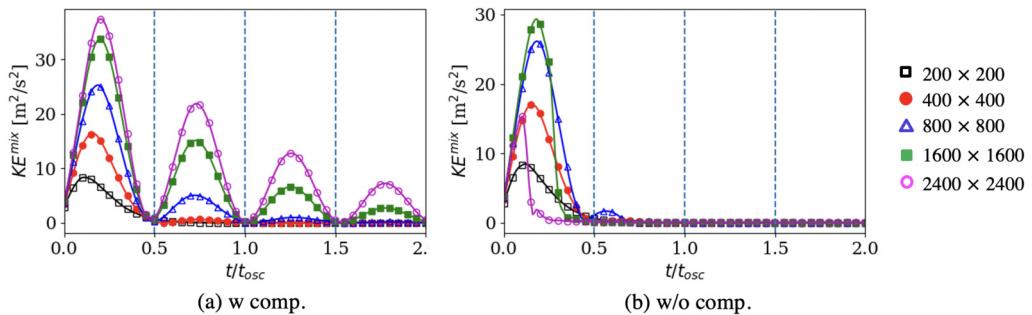


Fig. 15. KE^{mix} [$\text{kg m}^2/\text{s}^2$] for an oscillating ellipsoidal droplet. The dashed vertical lines correspond to the exact oscillation period. The simulation results have been shifted by $0.05t_{osc}$ on the bottom axis to exclude initial transients.

The ellipsoidal droplet starts converting into a circular shape because of the initial higher/lower curvature on the horizontal/vertical ends of the droplet. The potential energy stored in the droplet's surface tension converts into kinetic energy. The droplet oscillates from an initial ellipse into an intermediate circular shape and back to an ellipse. The kinetic energy oscillates during this process, and we can compute its frequency against an analytical value based on the Rayleigh formula [96]

$$\omega = (o^3 - o) \frac{\sigma}{(\rho^{(1)} + \rho^{(2)})r_0^3}, \quad (4.7)$$

where o is the oscillation mode. Corresponding to $o = 2$ and $r_0 = 0.1549$ m, an oscillation frequency is determined as $t_{osc} = 2\pi/\omega = 0.085$ s.

Based on the conclusions from the previous test, this test is simulated without the SR. The evolution of the droplet shape at various times is shown Fig. 14, and corresponding KE^{mix} is shown in Fig. 15. Similar to the previous test, interface compression is necessary for accurate curvature and surface tension computation. Since the pressure field is initialized uniformly, it takes approximately $0.05t_{osc}$ the pressure to develop within the domain and the oscillation sequence to set in. Accordingly, the simulation results are shifted on the time-axis such that the first crest in Fig. 15 matches with $0.5t_{osc}$ (first vertical line).

As expected, the initial elliptical droplet converts to a nearly circular droplet by $t/t_{osc} \approx 0.3$. This is accompanied by an increase in the KE^{mix} . The droplet regains an elliptical shape by $t/t_{osc} = 0.5$ and KE^{mix} shows reduces to zero. Ideally, this process should continue in the absence of any viscous effects. However, because of the numerical diffusion here, the peaks of KE^{mix} reduce with time. The 400×400 grid only captures two peaks of KE^{mix} , whereas the finer grids can capture more. The errors in the computed oscillation frequency with respect to its exact value are reported in Table 4 for different grids. The errors are larger for the coarser grids, but reduces to $< 8\%$ for grids larger than 800×800 .

4.6. Viscous effects

To verify the implementation of the viscous terms, a lid-driven cavity with a Reynolds number (Re) of 100 is simulated first. Initially, the fluid is at rest in a square domain of size L . The top wall is accelerated to a velocity u^∞ at $t = 0$, and all the other walls stay at rest. The flow within the domain starts moving and rotating due to the no-slip walls and the viscous effects. The simulations are run until a steady-state solution is achieved. Single-phase simulations [97] for this case have been established before, and the results match with an exact solution [91]. Two-phase results are obtained here with the DIM, where both the phases are set as CPG and with the same properties. The volume fraction $\alpha^{(2)}$ is initialized corresponding to a circular droplet within the domain using Eq. (3.23). Both the single-phase and the two-phase simulation results with and without the SR match the exact solution as shown in Fig. 16, verifying the implementation of the viscous terms. These results are with the second-order scheme.

Table 4

Errors in predicted oscillation frequency (ω) for droplets with surface tension. Results are only reported here for the case without SR and with interface compression.

Case	Error in ω prediction
200×200	66.0%
400×400	34.0%
800×800	7.89%
1600×1600	5.89%
2400×2400	3.89%

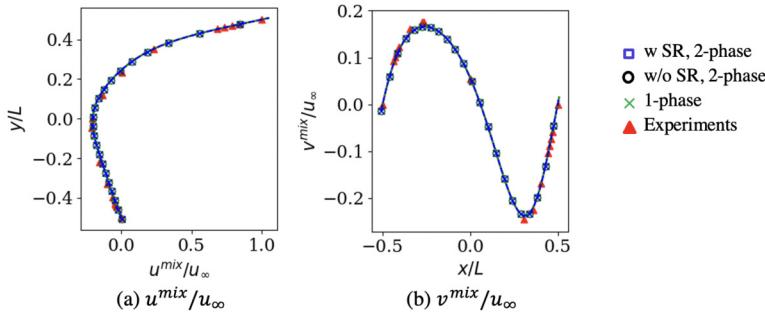


Fig. 16. Steady-state velocities are compared against an exact solution for the lid-driven cavity test.

Next, to validate the viscous effects in two-phase flows, the drag of a droplet in a viscous medium is computed. A 2D domain of length $L_x = 15r_0$ and width $L_y = 10r_0$ is used, where $r_0 = 0.5$ mm is the radius of an initially circular droplet initially placed at $(x, y) = (4r_0, 5r_0)$. The gas and the liquid phases are modeled using CPG and SG, respectively, with $\gamma^{(1)} = 1.4$, $\gamma^{(2)} = 4.4$, $p_0^{(2)} = 6 \times 10^8$ Pa. The velocity field surrounding the droplet is initiated using a potential flow solution, using a stream function

$$\psi = U_0 r \sin(\theta)[1 - R_0^2/r^2], \quad u_1^{(1)} = u_1^{(2)} = -\partial\psi/\partial y, \quad u_2^{(1)} = u_2^{(2)} = \partial\psi/\partial x, \quad (4.8)$$

where U_0 is the free-stream velocity, and $\theta = 1^\circ$ is used to trigger vortex shedding. The velocities inside the droplet are initialized as zero. The dynamic viscosity for the gas phase is set as $\mu^{(1)} = 1.8 \times 10^{-5}$ kg/(m-s), and for the liquid it is $\mu^{(1)} = 1.14 \times 10^{-3}$ kg/(m-s). The surface tension is specified as $\sigma = 0.0728$ N/m. The Reynolds number (Re) is defined as $Re = 2\rho^{(1)}r_0U_0/\mu_g$, and the Weber number (We) is defined as $We = 2\rho^{(1)}r_0U_0^2/\sigma$. Characteristic-based subsonic inflow and outflow boundary conditions (see Section 3.7) are used on the left and the right end of the computational domain, respectively. The inflow boundary condition is set to impose the velocity U_0 , and atmospheric pressure is imposed at the outflow. Symmetry boundary conditions are used on the top and the bottom.

Two conditions are simulated following a previous work using the five-equation model [98]. One with $U_0 = 5$ m/s, that corresponds to $Re = 81.3$ and $We = 0.1$, and another with $U_0 = 50$ m/s, that corresponds to $Re = 813$ and $We = 10$. A time-scale for normalization is defined as $t_0 = r_0/U_0$. Three grids are used with the number of cells across the droplet diameter $N_x = 20, 50, 100$. The simulations are conducted with and without the SR and with and without the interface compression. The interface diffuses irregularly without interface compression, similar to the previous surface tension test. The boundary and shear layers that are supposed to grow surrounding the sharp gas-liquid interface are wrongly predicted. Therefore, only the results with the interface compression are discussed further in this section.

The flow fields for the two simulated conditions are shown in Fig. 17. The droplet at $We = 0.1$ maintains a circular shape, but it deforms for $We = 10$. The low Re case develops a separation bubble behind the droplet, and the flow field remains relatively steady. On the other hand, the high Re case demonstrates a vortex shedding process. The positive and negative velocities surrounding the droplet are due to the viscous effects. During the quasi-steady behavior, i.e., $t/t_0 > 15$, the shedding frequency is predicted as 308 Hz. Due to the flattening of the droplet, the effective diameter that the incoming flow sees is $1.17d_0$. Correspondingly, the Strouhal number (St) is predicted as 0.18, which is close to the experimental value ≈ 0.2 reported for a solid cylinder at $Re \approx 813$ [99].

For further processing, the droplet velocity and a drag coefficient C_D are computed as

$$U_c = \int_{\Omega} \alpha^{(2)} u_1^{(2)} / \int_{\Omega} \alpha^{(2)}, \quad C_D = \frac{\rho^{(2)}}{\rho^{(1)}} \frac{dU_c}{dt} \frac{\pi r_0}{(U_0 - U_c)^2}. \quad (4.9)$$

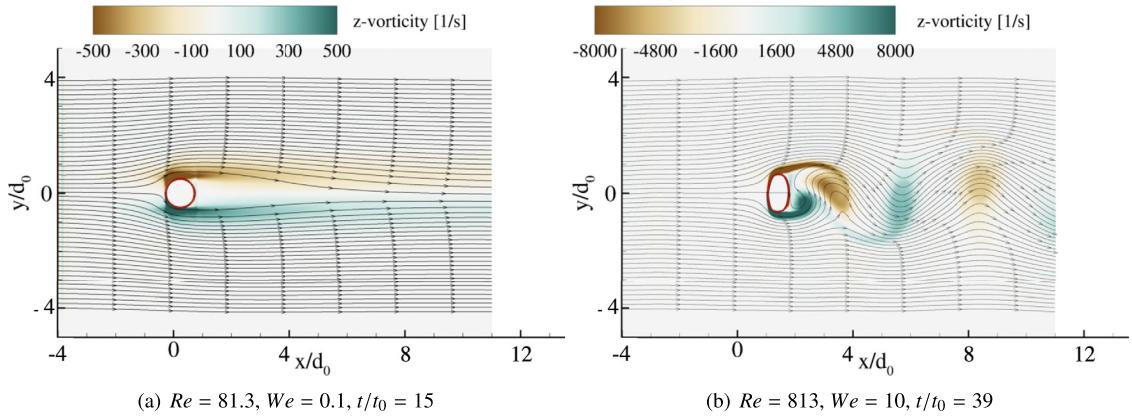


Fig. 17. Streamlines and vorticity contours are shown surrounding a droplet for the viscous computations. These results are without the SR with the interface compression and for the grid with $N_x = 100$. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

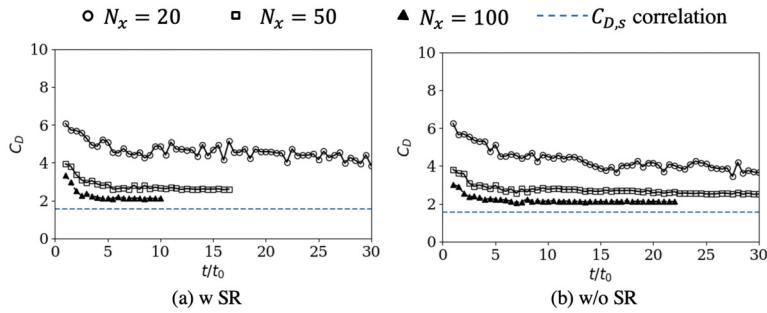


Fig. 18. Computed drag coefficient C_D and its evolution with time is shown for $Re = 81.3$, $We = 0.1$.

The computed C_D is compared against a steady-state drag correlation for a rigid cylinder $C_{D,S}(Re)$ [92] for $We = 0.1$ in Fig. 18. The droplet with $We = 10$ starts deforming, and therefore, a comparison against the correlation is not possible for that. These results show that establishing the viscous flow surrounding the droplet takes approximately $5t_0$, and C_D reaches a steady value after that. The C_D is overpredicted for the coarser grids. However, a better match against the correlation $C_{D,S}(Re)$ is achieved with grid-refinement. The C_D for the finest grid is also $\approx 10\%$ higher than the correlation $C_{D,S}(Re)$, but this could be attributed to the relative acceleration effect [100] that is not included in the correlation $C_{D,S}(Re)$, but are present in the current simulations since the relative velocity $U_r = U_c - U_0$ decreases with time. These results also show that even though the C_D with and without the SR do not match exactly, they are very similar, and this further indicates the efficacy of the viscous DEM in accurately modeling the non-conservative interface terms such that there is no need to use the SR with the seven-equation model.

4.7. Collapse of a spherical bubble

Computing collapse and rebound of a spherical bubble surrounded by a high-pressure liquid is a seemingly simple but challenging problem, where, instead of a wave impacting a multiphase interface, the waves are generated at the interface [44,67]. Modeling the pressure (or mechanical) equilibrium is crucial for obtaining accurate predictions for this test. As a result, the five-equation model without the volume fraction source term [34] showed inaccurate results, and the augmented model of Kapila et al. [33] was required [44]. The six-equation model was able to accurately model the collapse and the rebound of the bubble, however, only when the pressure relaxation terms (R_p^κ) are included [67]. Since the previous tests in this work have shown the ability of the DEM within the seven-equation framework to relieve the need to use the SR, this configuration is simulated to evaluate if the same holds true here.

Considering the high computational cost of simulating the entire spherical droplet, a 3D spherical sector with 5° angle is used as the computational domain. The phase (1), i.e., the bubble, is modeled as CPG with $\gamma^{(1)} = 1.4$, whereas the phase (2), i.e., the surrounding water, is modeled as SG with $\gamma^{(2)} = 2.35$, $p_0^{(2)} = 1 \times 10^9$ Pa [67]. Initially, $\alpha^{(2)} = \epsilon$ for $r < r_0$ and $\alpha^{(2)} = 1 - \epsilon$ for $r \geq r_0$, where r_0 is the initial bubble radius. The pressure within the bubble is set as $p^{(1)} = p^{(2)} = p_b$, and it relaxes in the radially outward direction to p_∞ as $p^{(1)}(r) = p^{(2)}(r) = p_\infty + \frac{r_0}{r}(p_b - p_\infty)$. Here, r is the distance from the center of the bubble. To avoid any effects of the outer boundary, the computational domain radially extends from

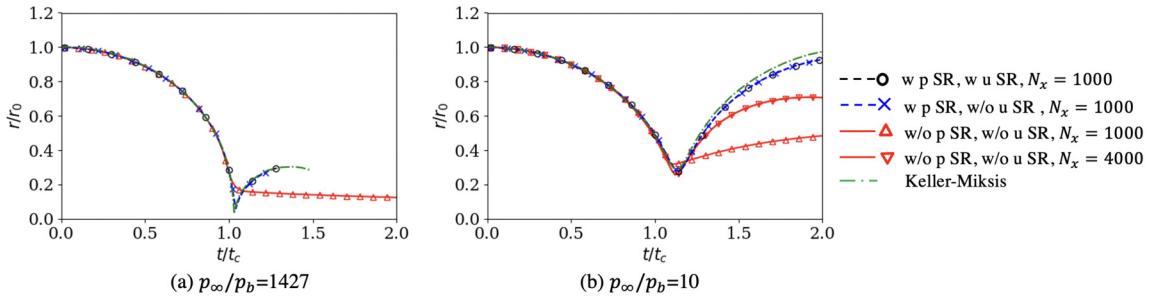


Fig. 19. Bubble radius evolution is plotted during the spherical bubble collapse for two p_∞/p_b and various combinations of using the stiff relaxation (SR) solver for the pressure (p) and the velocity (u). Higher grid resolutions are also shown for $p_\infty/p_b = 10$ without the SR. N_x refers to the number of cells in the radial direction. The results are compared against the semi-analytical solution of Keller and Miksis [90].

$r = 0.01r_0$ to $r = 32r_0$. Symmetry boundary conditions are used in the azimuthal direction and at $r = 0.01r_0$. A Neumann boundary condition is used for $r = 32r_0$.

The grid contains of $1000 \times 4 \times 4$ cells, with uniform cells of size $r_0/100$ distributed in the radial direction for $r < 2r_0$ and gradual coarsening in the outer region. This grid resolution is consistent with the previous works [44,67]. Two pressure ratios, i.e., $p_\infty/p_b = 10, 1427$, are considered [44]. The simulation results are compared against the semi-analytical solution obtained using the Keller–Miksis equation [90], which is a compressible form of the Rayleigh–Plesset equation. This equation uses an asymptotic expansion in Mach number for a spherical bubble, so its use here as the exact solution presumes that errors measured relative to it are larger than any errors associated with the asymptotic expansion and presumption of sphericity inherent to it.

The simulation results with the seven-equation model are shown in Fig. 19. Due to the higher pressure in the surrounding liquid, the bubble starts collapsing and the pressure inside it starts to increase. Once the pressure inside the bubble gets higher than the surrounding, it starts to rebound and the pressure inside starts relaxing. For both $p_\infty/p_b = 10, 1427$, the simulations that employ the SR are able to model the collapse and the rebound of the bubble. On the other hand, the simulation results without the SR are able to predict the bubble collapse accurately, but not its rebound. Additional simulations are considered where we employ the SR for the pressure but not for the velocity, and these simulations are also able to accurately predict the bubble collapse-rebound process for both p_∞/p_b , reconfirming the importance of maintaining the pressure equilibrium between the phases for this case.

Considering the previous discussion in Section 3.5 on the interchangeable roles of the non-conservative interphase exchange terms I^K and the relaxation terms R^K , one would expect the simulations even without the SR terms to capture the collapse-rebound process if I^K are modeled accurately. Previous tests with shock interactions showed this. To further understand this behavior, additional simulations with finer grids (2000, 4000 cells in the radial direction) are conducted for $p_\infty/p_b = 10$ without the SR, and these do approach towards the semi-analytical solution. This suggests that similar to the other tests considered in this work, even for this challenging problem, where modeling the effects of pressure equilibrium is crucial, the DEM can handle the pressure equilibrium through the interface terms I^K , relieving the need to model R^K , but it requires a much higher resolution. A similar grid refinement study can also be considered for $p_\infty/p_b = 1427$, but since the collapse-rebound process for that is much sharper, and the minimum radius is even smaller, it is expected that simulating this without the SR would require a tremendous resolution. In this context, including the pressure SR for such a problem is suggested with the seven-equation model.

4.8. Shock-droplet interaction

This test simulates the early-stage deformation of a cylindrical water droplet when hit by a shock wave. The setup follows the experiments by Igra et al. [93] and other numerical works that used a five-equation DIM [35,94]. A circular droplet of diameter (d_0) 4.8 mm is placed in a 2D computational domain of size $23d_0 \times 12d_0$. The background gas-phase is initialized with a shock of strength $M = 1.47$ that is located at a distance d_0 ahead of the droplet center. The pre-shock (background) pressure and velocity are $p^{(1)} = p^{(2)} = 101325$ Pa and $u^{(1)} = u^{(2)} = 0$ m/s, respectively. The corresponding post-shock conditions are subsonic, and they are set as $p^{(1)} = p^{(2)} = 238558$ Pa and $u^{(1)} = u^{(2)} = u_0 = 226$ m/s. The water density is set as $\rho^{(2)} = 1000$ m/s. The pre- and the post-shock gas-phase densities are $\rho^{(1)} = 1.204$ kg/m³ and $\rho^{(1)} = 2.18$ kg/m³, respectively. Note that these initial conditions are equivalent to the IC-2 as described earlier in Section 4.3. The liquid water is modeled with a SG EOS with $\gamma^{(2)} = 6.12$ and $p_0^{(2)} = 343.44 \times 10^6$ Pa. The gas phase is modeled with a CPG EOS with $\gamma^{(1)} = 1.4$. The volume fraction is initialized using Eq. (3.23) with $\delta = 0.5$.

The computational domain is discretized using a uniform grid $N_x = 17, 42, 83, 170, 255, 340$ cells across the droplet diameter. The top and the bottom boundaries are set as slip walls. The left boundary behind the shock is set as a non-reflective subsonic inflow at the post-shock velocity u_0 . The right boundary, ahead of the shock, is set as a supersonic outflow to let the incoming shock wave pass through without any reflections. The Reynolds number (Re) and Weber number (We) corresponding to this condition are $Re = 4860$ and $We = 72320$, respectively. Under these conditions, the droplet

deformation is primarily driven by inertia during the early stages. Therefore, the surface tension and viscous effects can be neglected, as confirmed in the earlier numerical studies [35,94]. Note that these effects would still be relevant near the final stages of the breakup and at much smaller length scales that are not resolved here. The simulations are conducted and compared with and without the surface tension and with and without the interface compression.

A non-dimensional time is defined as $\tau = t / (u_0/d_0 \sqrt{\rho^{(1)} / \rho^{(2)}})$, where the denominator characterizes time for breakup via Rayleigh–Taylor or Kelvin–Helmholtz instabilities [101]. The simulation results at various times are shown in Fig. 20 for the case with $N_x = 83$. These results can be qualitatively compared against those of Meng and Colonius [35] that were obtained with the five-equation model and $N_x = 100$. These are with the interface compression and with/without the stiff relaxation. Pressure contours and Numerical Schlieren ($|\partial \rho^{(1)} / \partial x_i|$) are used to identify the shock structure. The droplet is using a $\alpha_T^{(2)} = 0.5$ contour. As the initial shock hits the droplet, there are transmitted and reflected waves. Unlike the previous 1D case, the shock structures are no longer planar because of the circular droplet. The waves that propagate inside the cylinder transmit and reflect through the material interface. The droplet remains circular during the earlier stages (up to $\tau \approx 0.17$). However, it starts to deform due to the vortical structures created around it later. These features are qualitatively similar to the experimental observations [93] and previous numerical results [35,94]. The results with and without the SR match closely, reconfirming the ability of the DEM to alleviate the need to use an SR. Disturbances are noticeable behind the propagating shock in the Numerical Schlieren ($|\partial \rho^{(1)} / \partial x_i|$), particularly without the SR at $\tau = 0.03$ (Fig. 20(b)). The corresponding disturbances in terms of the primitive variables are very small, e.g., for $\rho^{(1)}$, they are $\sim 0.02\%$ without the SR and $\sim 0.09\%$ with the SR, and they can be considered errors related to the numerical scheme.

The experiments measured the droplet motion and the droplet deformation in four geometrical parameters. 1) Leading edge drift (x_L), that is, the x -location of the droplet leading edge measured with respect to its initial location, 2) droplet centerline width (w), which measures the width of the cylinder along the x -axis, 3) droplet spanwise diameter (d), that is the width of the droplet in the y -direction, and 4) droplet area (A). All these are computed from the simulation flow-fields using a threshold volume fraction $\alpha_T^{(2)}$, considering that $\alpha^{(2)} > \alpha_T^{(2)}$ is the droplet shape whose geometrical parameters have to be determined.

Because of the uncertainty in the experimental measurements, it is unclear what value $\alpha_T^{(2)}$ should take to best match the data. Even though the gas–liquid interface is supposed to remain sharp in reality, this region may appear diffused depending on the measurement technique because of the small droplets strip surrounding the parent droplet surface. In addition to this, the interface also numerically diffuses for the simulations. The results are analyzed for $\alpha_T^{(2)} = 0.1, 0.5, 0.9$, but only those with $\alpha_T^{(2)} = 0.9$ are shown here. Sensitivity of five-equation DIM results in the choice of $\alpha_T^{(2)}$ can be found elsewhere [35] and our observations are similar for the cases without the interface compression. The simulation results with the interface compression do not show a sensitivity to $\alpha_T^{(2)}$ since a regular interface thickness is maintained. However, those results are considered erroneous for this test, and further details are discussed later.

The simulations with the SR and the interface compression take longer to run, e.g., the simulations are $1.6 \times$ slower with the interface compression and $5.6 \times$ slower with the SR. Because of the cost of these simulations, the finer grids are not simulated until completion with the SR and the interface compression. As discussed next, the results are still considered sufficient to draw valuable conclusions. To provide a comparison of our approach against the available numerical studies, our predictions are also compared against those of Chen [94] and Meng and Colonius [35]. There are certain differences in the simulation approach. Chen [94] used free-stream boundary conditions with $N_x = 48$, and Meng and Colonius [35] used characteristic-based non-reflective boundary conditions [102] and $N_x = 100$. Both these studies used the five-equation model. This study uses the seven-equation model and simplified subsonic characteristic-based boundary conditions [87] that treat the liquid-phase quantities as passive scalars (see Section 3.7). The grid resolution $N_x = 83$ from the current simulations is the closest to [35].

The droplet leading edge (x_L/d_0) evolution is shown in Fig. 21. All grid resolutions show a similar trend, but a grid convergence is achieved for $N_x \geq 170$. Stiff relaxation or interface compression does not result in a noticeable difference in leading-edge predictions. Our results initially ($\tau < 0.7$) match with those of Meng and Colonius [35] and the experiments [93], but later on diverge away from it, while coming closer to those of Chen [94]. The difference between the predictions and the experiments eventually grows to almost 100% by $\tau = 1.3$. This could be due to the subsonic inflow boundary conditions used on the left boundary for the simulations to maintain the post-shock velocity. The experiments were conducted within a longer shock tube where we did not need this.

The centerline droplet width (w) decreases with time due to the droplet compression, and the spanwise droplet diameter (d) increases as the droplet gets elongated. These are plotted in Fig. 22 and Fig. 23, respectively. Without the SR and without the interface compression, the predicted droplet width w matches well against the data (<5% error) for all grid resolutions. These results also comparable with Meng and Colonius [35].

On the other hand, the droplet height d is grossly underpredicted at coarser resolutions, and a grid convergence is obtained only for $N_x \geq 255$. The fine grid results capture the same trend as the experimental measurements [93]. However, they show about 30–40% overprediction. Still, the current results with $N_x = 83$, $\alpha_T^{(2)} = 0.9$ are closer to $N_x = 100$, $\alpha_T^{(2)} = 0.99$ of Meng and Colonius [35], whereas, the grid converged results for $N_x = 170, 255, 340$ lie between the two limits of $\alpha_T^{(2)} = 0.25 – 0.99$ in [35].

The droplet area is expected to reduce with time for the experiments due to the stripping of small-sized droplets. The simulations capture this without the SR and without the interface compression as shown in Fig. 24. The coarsest grid shows

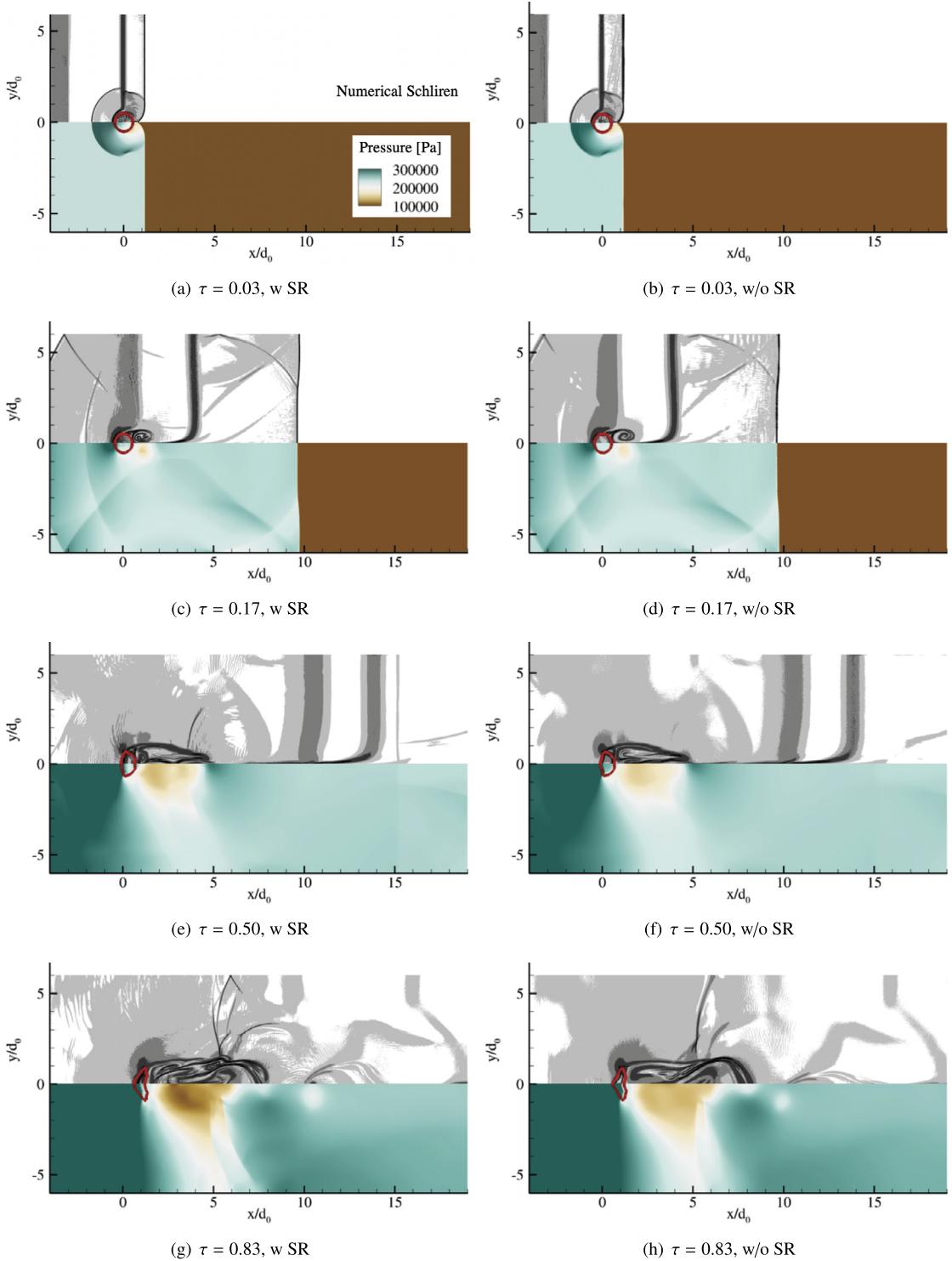


Fig. 20. Pressure contours (bottom) and numerical Schlieren (top) image are shown for the 2D shock-droplet interaction with $N_x = 83$, with interface compression, and with/without the stiff relaxations. The droplet is identified using a $\alpha^{(2)} = 0.5$ contour (thick red line). (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

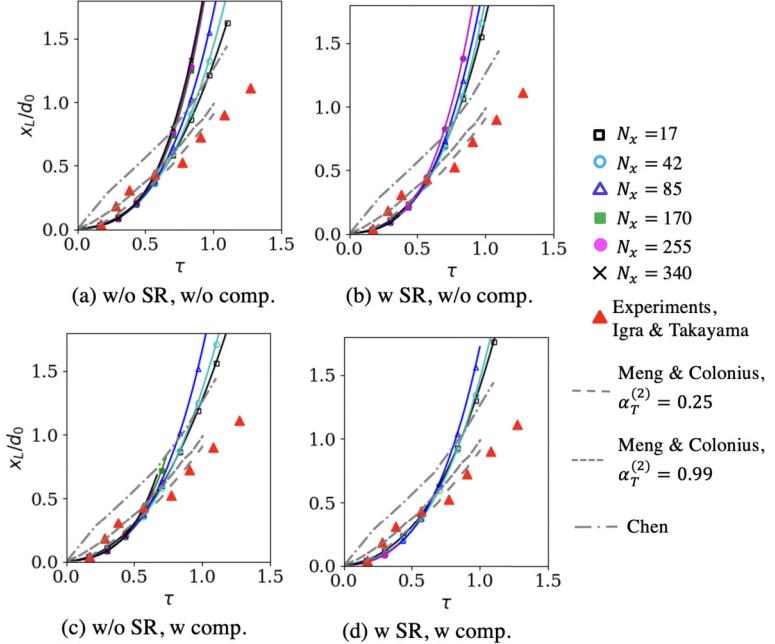


Fig. 21. Time-evolution of the droplet leading edge (x_L) compared against the experimental measurements of Igra et al. [93] and numerical results of Meng and Colonius [35] and Chen [94].

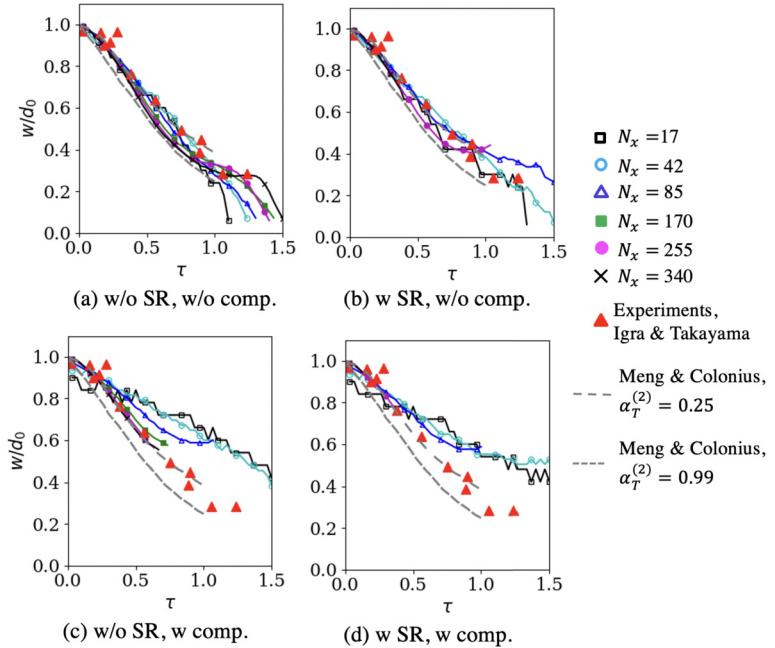


Fig. 22. Time-evolution of the droplet width (d) compared against the experimental measurements [93] and a previous numerical study [35].

up to 25% underprediction. However, a grid convergence is obtained for $N_x \geq 170$, and the finer grids' results match the measurements [93] and the previous numerical results [35] with $<5\%$ error.

Regarding the stiff relaxation solver usage, similar to the leading-edge droplet predictions, the predictions for all three geometrical features (d , w , A) are similar with/without the SR. The most significant differences are observed later in d and A predictions, which are still $<10\text{--}15\%$. This further confirms the efficacy of the DEM in alleviating the need to use a separate SR solver, which, as demonstrated for this case, could be as much as 5.6-times as costly.

The effect of the interface compression on the prediction is widely apparent on w , d , and A . In reality, the droplet area would reduce with time due to the stripping of small-sized droplets. One could estimate the size of these stripped

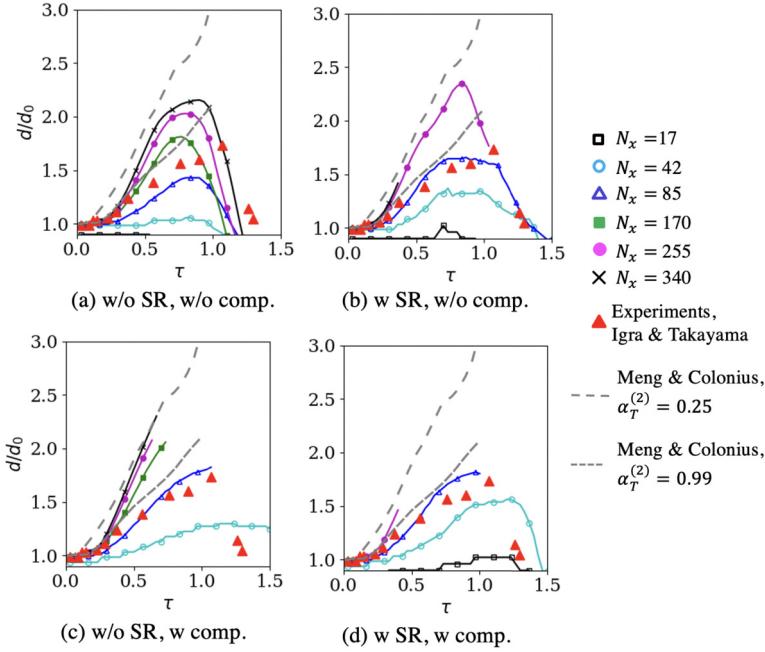


Fig. 23. Time-evolution of the droplet height (d) compared against the experimental measurements [93] and a previous numerical study [35].

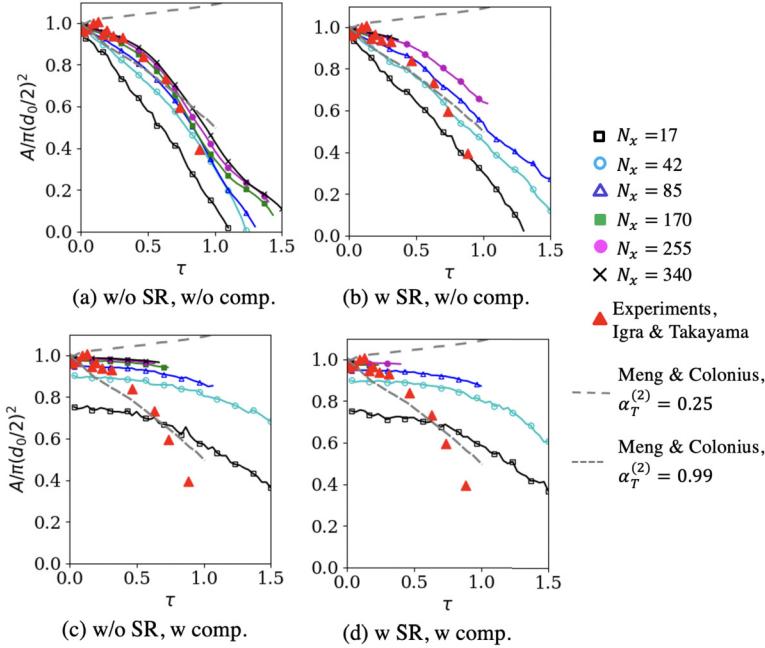


Fig. 24. Time-evolution of the droplet area (A) compared against the experimental measurements [93] and a previous numerical study [35].

droplets as $\sim 6 \mu\text{m}$ based on $We \sim 1$. It is not possible to resolve these small droplets with our computational grid, and in the DIM simulations considered here, they would be represented as intermediate volume fractions ($\epsilon < \alpha^{(2)} < 1 - \epsilon$). The interface compression approach that is used here can only work with the resolved fluids, and it wrongly interprets the regions with $\epsilon < \alpha^{(2)} < 1 - \epsilon$ as numerically diffused regions, applying the compression there. As a result of this, the results with the interface compression are insensitive to the threshold volume fraction $\alpha_T^{(2)}$, but they do not match the data. Transition to a dispersed phase Eulerian–Eulerian and Eulerian–Lagrangian approaches such as [8] to model these regions with $\epsilon < \alpha^{(2)} < 1 - \epsilon$ can be devised, however, that is not the focus of this work, and it would be considered in the future.

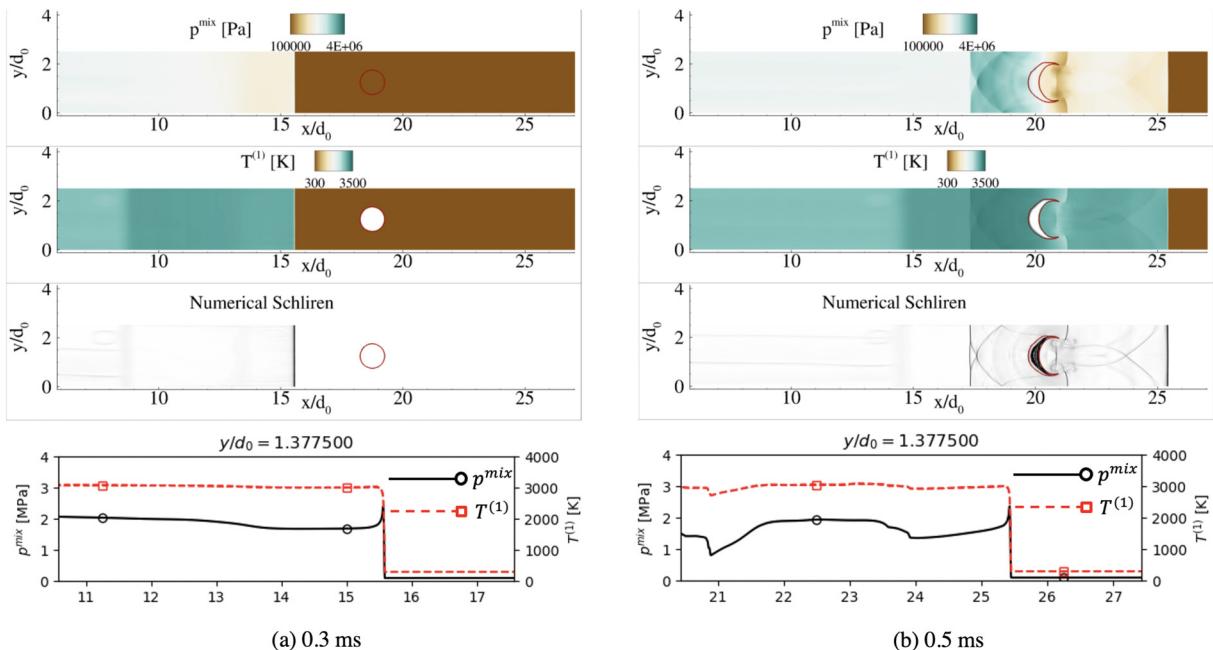


Fig. 25. Pressure contours (first row), temperature contours (second row), numerical Schlieren (third row), and flow-field quantities along the centerline are shown for the 2D shock droplet interaction test. The droplet is identified using a $\alpha^{(2)} = 0.5$ contour (thick red line). (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

4.9. Detonation-droplet interaction

One could encounter a detonation-droplet interaction in practical applications such as a liquid-fueled or liquid-assisted rotating detonation engine (RDE) [103,104] or an energetic material packed with metal particles [105]. Some recent efforts have focused on studying this phenomenon experimentally [69]. However, the simulations here serve as a qualitative demonstration without detailed measurements for quantitative validation.

A 2D computational domain of size (1500, 100) mm is used. A liquid droplet ($\kappa = 2$) of diameter d_0 mm is initialized at $(x, y) = (750, 50)$ mm. A uniform grid with $N_x = 100$ cells across the droplet diameter is used. The left and the right boundaries are set as supersonic outflows to let the internal waves pass through without reflection. The top and the bottom boundaries are set as slip walls. The entire domain is at rest initially ($u^{(1)} = u^{(2)} = 0$) and the pressure are set as $p^{(1)} = p^{(2)} = 10^5$ Pa. The densities are $\rho^{(1)} = 1.204 \text{ kg/m}^3$, and $\rho^{(2)} = 1000 \text{ kg/m}^3$. The gas-phase ($\kappa = 1$) contains gaseous H₂ and air at stoichiometry to mimic conditions in a typical rotating detonation engine [79].

In order to initiate a 2D detonation wave, $x < 10$ mm is initially filled with detonation products at $T^{(1)} = 3500$ K and $p^{(1)} = 5 \times 10^6$ Pa. A circular disturbance of radius 1 mm is asymmetrically added in the domain to trigger the transverse detonation waves [106]. The droplet volume fraction $\alpha^{(2)}$ is initialized using Eq. (3.23) with $\delta = 0.5$. The droplet diameter $d_0 = 40$ mm is relatively large compared to real applications. However, it is chosen here considering the cost and the grid requirements for resolving a droplet.

The liquid phase is modeled using an SG EOS with $\gamma^{(2)} = 4.1$ and $p^0 = 6 \times 10^8$, and the gas phase is modeled using a TPG EOS that uses NASA polynomial curve-fits for the species. The droplet would eventually vaporize because of the high temperature behind the detonation. However, for the current conditions, the ratio of the droplet vaporization time (t_{drop}) to the detonation passage time (t_{deton}) is estimated to be substantial, $t_{drop}/t_{deton} \sim 10^6$, suggesting that one can simulate the early stages of this detonation-droplet interaction without using any heat/mass-transfer models. Similar to the previous test in Section 4.8, the inertia primarily drives the droplet deformation at these high-speed conditions. Therefore, the viscous and the surface tension effects are neglected for this test.

The simulation results at various time instances are shown in Fig. 25. The results shown here are with the interface compression scheme and the SR. However, simulations without them have also been conducted. As a result of the initiation, a 2D detonation wave propagates in the gas phase before reaching the droplet. The structure of this detonation wave is confirmed by a pressure spike (of 2.55 MPa), an increase in the temperature (to 3078 K) due to coupled shock, and burning at the front. The detonation speed before reaching the droplet is measured as 1968 m/s, which is extremely close to the Chapman–Jouguet velocity of 1964 m/s computed using a Chemical Equilibrium Applications (CEA) code at the same conditions.

Similar to the shock-droplet interaction, as the detonation wave hits the droplet, it results in the transmission and reflection of non-planar waves. A circular reflected shock is observed propagating back into the detonation products. As

the wave propagates after crossing the droplet, it eventually regains its reactive detonation structure. The droplet remains circular at initial times. However, it deforms and elongates, as was also the case with the shock-droplet interaction.

5. Conclusions

A seven-equation diffused interface method (DIM) framework is developed in this work. Unlike the commonly used reduced DIM models, the seven-equation model does not enforce a velocity/pressure equilibrium. Alleviating a strict pressure relaxation allows this method for arbitrarily high pressure/density ratio and arbitrarily independent forms of the EOS of the phases, and alleviating the strict velocity equilibrium would enable it to be used in a dispersed phase regime where the MPE are unresolved. The seven-equation model has been used for simplistic 1D tests [28,32]. However, it is extended in this work to be used in 2D/3D with arbitrary EOS, surface tension, viscous effects, and multi-species reacting flows with the discrete equations method (DEM) [32] as a baseline numerical method. Various tests, i.e., multiphase shock tube, shock-material interaction, oscillating droplet, drag in a viscous medium, shock-droplet interaction, and detonation-droplet interaction, are conducted to validate different parts of the method. Following are some novel contributions and conclusions of this work.

Potential of the seven-equation model

- To demonstrate the benefits of allowing a velocity non-equilibrium between the phases, a new test is considered where a shock first passes through a curtain of unresolved MPEs and then through a resolved multiphase interface. A velocity non-equilibrium between the phases is required for accurate modeling of particle dispersion via drag laws, whereas, a velocity equilibrium is maintained at the resolved multiphase interface due to the DEM.
- The benefits of not enforcing a pressure equilibrium analytically are similar to those of the six-equation model, and these are demonstrated in this work simulating tests with high density/pressure ratios ($\rho^{(2)}/\rho^{(1)} \sim 1 - 10^3$, $\rho^{(2)}/\rho^{(1)} \sim 1 - 10^5$), and arbitrary EOS (SG, CPG, TPG, MG).

Features for modeling resolved interfaces

- Modifications for modeling 2D and 3D viscous effects within the DEM framework have been included. Droplet evolution in a viscous medium is simulated. The computed drag coefficient C_D matches a steady-state drag correlation for a low Weber number case where the droplet does not deform. Vortex shedding is observed for a higher Re case, and the droplet also deforms due to the higher We .
- To include surface tension, a modification is proposed in the inviscid fluxes of DEM and the corresponding HLLC Riemann solver. This proposed method can handle a surface-tension-related pressure jump across a material interface in an exact numerical manner. The pressure jump within a circular droplet is captured with $< 2\%$ error with numerically computed curvature. An oscillating droplet where the surface potential energy converts back and forth into the kinetic energy is simulated, and the oscillation period matches the exact value with a $< 8\%$ error for grids finer than 800×800 .
- A shock-droplet interaction case is simulated at experimental conditions, and the computed droplet deformation parameters are shown to match against available data and past numerical results. This would create many small droplets due to the atomization process at such high Re and We in reality. However, since the grid does not resolve these, they manifest as regions of intermediate volume fractions ($\epsilon > \alpha^{(2)} > 1 - \epsilon$) with DIM. We can consider modeling these regions using a dispersed phase approach in the future.
- The scheme has been demonstrated to work for multi-species, reacting flows via a detonation-droplet interaction problem. This test is a qualitative demonstration without any validation/verification data.

Need of stiff relaxation and interface compression

- A stiff relaxation (SR) solver is often needed to enforce the pressure and the velocity equilibrium between the phases at the interface [28]. The approach in current work is consistent with the previous work [28], but an adaptive-step gradient descent was used for pressure relaxation to make the scheme robust and useful for arbitrary EOS. Since both the interface terms I^K and the relaxation terms R^K model interphase exchanges at the interface, with accurate modeling of I^K using the DEM [32], the need of SR is relieved for most tests (i.e., shock-tube, shock-impact, viscous drag). The surface tension test shows inaccurate results with the SR, as enforcing a pressure jump between the phases creates an artificial pressure-jump in the DEM-based Riemann sub-problems. On the other hand, for predicting collapse-rebound of a spherical bubble, pressure SR (R_p^K) is still needed with coarse grids, and modeling this without the SR, i.e., only through I^K , requires a significantly higher grid resolution. A generalized approach may have to be devised in the future which either relieves the need to use the pressure relaxation solver even for challenging problems such as a spherical bubble collapse at reasonable grid resolutions, or allows its use in presence of surface tension.
- An interface compression scheme initially used for the five-equation model is modified for the seven-equation model. The interface compression was necessary for the surface tension tests and the viscous drag computation. Otherwise, the droplet interface gets irregular, leading to difficulties in curvature computation and boundary and shear layer predictions surrounding the droplet surface. However, the interface compression scheme was detrimental for the 2D shock-droplet

interaction, since it would treat the dispersed phase regions with unresolved MPEs as numerically diffused regions and apply compression. An interface compression scheme should still be used for resolved MPE modeling, however, a hybridization approach should be considered in the future so that it is dynamically switched off in regions with unresolved MPE.

This work does not consider heat and mass transfer at the droplet interface, which is an obvious extension of this work. Infinite [47,48] or finite-rate [107] thermochemical relaxation solvers can be used. A key benefit of the seven-equation model is its possible extension to dispersed-phase regions where an equilibrium between the velocities is not maintained. The potential for this is demonstrated in this work, and the next step could be to dynamically use drag and heat and mass transfer laws for interphase exchange in the regions where there are unresolved droplets, along with additional modifications for obtaining physically accurate wave-speeds in dense-to-dilute regions [8,66]. Development of a generalized approach to handle the SR, improvement of the interface compression scheme to maintain conservative properties [71], and corresponding modifications to use them for both resolved and unresolved MPEs, are other considerations for future work.

CRediT authorship contribution statement

Achyut Panchal: Conceptualization, Methodology, Validation, Formal analysis, Software, Investigation, Visualization.
Spencer H. Bryngelson: Supervision, Reviewing. **Suresh Menon:** Supervision, Reviewing, Resources.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgement

This work was supported by NASA Glenn Research Center [grant number NNX15AU91A].

References

- [1] A.H. Lefebvre, *Gas Turbine Combustion*, CRC Press, 1998.
- [2] G.P. Sutton, *History of Liquid Propellant Rocket Engines*, AIAA, 2006.
- [3] C.M. Tarver, S.K. Chidester, A.L. Nichols, Critical conditions for impact-and shock-induced hot spots in solid explosives, *J. Phys. Chem.* 100 (1996) 5794–5799.
- [4] R. Vehring, Pharmaceutical particle engineering via spray drying, *Pharm. Res.* 25 (2008) 999–1022.
- [5] M. Gorokhovski, M. Herrmann, Modeling primary atomization, *Annu. Rev. Fluid Mech.* 40 (2008) 343–366.
- [6] R. Saurel, C. Pantano, Diffuse-interface capturing methods for compressible two-phase flows, *Annu. Rev. Fluid Mech.* 50 (2018) 105–130.
- [7] S. Balachandar, J.K. Eaton, Turbulent dispersed multiphase flow, *Annu. Rev. Fluid Mech.* 42 (2010) 111–133.
- [8] A. Panchal, S. Menon, A hybrid Eulerian-Eulerian/Lagrangian method for dense-to-dilute dispersed phase flows, *J. Comput. Phys.* 439 (2021) 110339.
- [9] S.H. Bryngelson, K. Schmidmayer, T. Colonius, A quantitative comparison of phase-averaged models for bubbly, cavitating flows, *Int. J. Multiph. Flow* 115 (2019) 137–143.
- [10] S.H. Bryngelson, R.O. Fox, T. Colonius, Conditional moment methods for polydisperse cavitating flows, *arXiv preprint arXiv:2112.14172*, 2021.
- [11] J.A. Sethian, P. Smereka, Level set methods for fluid interfaces, *Annu. Rev. Fluid Mech.* 35 (2003) 341–372.
- [12] M. Sussman, K.M. Smith, M.Y. Hussaini, M. Ohta, R. Zhi-Wei, A sharp interface method for incompressible two-phase flows, *J. Comput. Phys.* 221 (2007) 469–505.
- [13] C.W. Hirt, B.D. Nichols, Volume of fluid (VOF) method for the dynamics of free boundaries, *J. Comput. Phys.* 39 (1981) 201–225.
- [14] M. Sussman, E.G. Puckett, A coupled level set and volume-of-fluid method for computing 3d and axisymmetric incompressible two-phase flows, *J. Comput. Phys.* 162 (2000) 301–337.
- [15] J. Brackbill, D.B. Kothe, C. Zemach, A continuum method for modeling surface tension, *J. Comput. Phys.* 100 (1992) 335–354.
- [16] 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.
- [17] S. Elghobashi, Direct numerical simulation of turbulent flows laden with droplets or bubbles, *Annu. Rev. Fluid Mech.* 51 (2019) 217–244.
- [18] P. Das, H. Udaykumar, A sharp-interface method for the simulation of shock-induced vaporization of droplets, *J. Comput. Phys.* 405 (2020) 109005.
- [19] H. Terashima, G. Tryggvason, A front-tracking/ghost-fluid method for fluid interfaces in compressible flows, *J. Comput. Phys.* 228 (2009) 4012–4037.
- [20] P.T. Barton, B. Obadia, D. Drikakis, A conservative level-set based method for compressible solid/fluid problems on fixed grids, *J. Comput. Phys.* 230 (2011) 7867–7890.
- [21] M. Baer, J. Nunziato, A two-phase mixture theory for the deflagration-to-detonation transition (DDT) in reactive granular materials, *Int. J. Multiph. Flow* 12 (1986) 861–889.
- [22] X.-D. Liu, S. Osher, T. Chan, Weighted essentially non-oscillatory schemes, *J. Comput. Phys.* 115 (1994) 200–212.
- [23] B. Van Leer, Towards the ultimate conservative difference scheme. V. A second-order sequel to Godunov's method, *J. Comput. Phys.* 32 (1979) 101–136.
- [24] F. Petitpas, J. Massoni, R. Saurel, E. Lapebie, L. Munier, Diffuse interface model for high speed cavitating underwater systems, *Int. J. Multiph. Flow* 35 (2009) 747–759.

- [25] K. Schmidmayer, F. Petitpas, S. Le Martelot, E. Daniel, ECOGEN: an open-source tool for multiphase, compressible, multiphysics flows, *Comput. Phys. Commun.* 251 (2020) 107093.
- [26] R. Saurel, P. Boivin, O. Le Métayer, A general formulation for cavitating, boiling and evaporating flows, *Comput. Fluids* 128 (2016) 53–64.
- [27] M.G. Rodio, R. Abgrall, An innovative phase transition modeling for reproducing cavitation through a five-equation model and theoretical generalization to six and seven-equation models, *Int. J. Heat Mass Transf.* 89 (2015) 1386–1401.
- [28] R. Saurel, R. Abgrall, A multiphase Godunov method for compressible multifluid and multiphase flows, *J. Comput. Phys.* 150 (1999) 425–467.
- [29] N. Andrianov, G. Warnecke, The Riemann problem for the Baer–Nunziato two-phase flow model, *J. Comput. Phys.* 195 (2004) 434–464.
- [30] D.W. Schwendeman, C.W. Wahle, A.K. Kapila, The Riemann problem and a high-resolution Godunov method for a model of compressible two-phase flow, *J. Comput. Phys.* 212 (2006) 490–526.
- [31] H. Lochon, F. Daude, P. Galon, J.-M. Hérard, HLLC-type Riemann solver with approximated two-phase contact for the computation of the Baer–Nunziato two-fluid model, *J. Comput. Phys.* 326 (2016) 733–762.
- [32] R. Abgrall, R. Saurel, Discrete equations for physical and numerical compressible multiphase mixtures, *J. Comput. Phys.* 186 (2003) 361–396.
- [33] A. Kapila, R. Menikoff, J. Bdzil, S. Son, D.S. Stewart, Two-phase modeling of deflagration-to-detonation transition in granular materials: reduced equations, *Phys. Fluids* 13 (2001) 3002–3024.
- [34] G. Allaire, S. Clerc, S. Kokh, A five-equation model for the simulation of interfaces between compressible fluids, *J. Comput. Phys.* 181 (2002) 577–616.
- [35] J. Meng, T. Colonius, Numerical simulations of the early stages of high-speed droplet breakup, *Shock Waves* 25 (2015) 399–414.
- [36] N. Liu, Z. Wang, M. Sun, R. Deiterding, H. Wang, Simulation of liquid jet primary breakup in a supersonic crossflow under adaptive mesh refinement framework, *Aerosp. Sci. Technol.* 91 (2019) 456–473.
- [37] A. Murrone, H. Guillard, A five equation reduced model for compressible two phase flow problems, *J. Comput. Phys.* 202 (2005) 664–698.
- [38] F. Petitpas, E. Franquet, R. Saurel, O. Le Metayer, A relaxation-projection method for compressible flows. Part II: artificial heat exchanges for multiphase shocks, *J. Comput. Phys.* 225 (2007) 2214–2248.
- [39] A. Tiwari, J.B. Freund, C. Pantano, A diffuse interface model with immiscibility preservation, *J. Comput. Phys.* 252 (2013) 290–309.
- [40] G. Perigaud, R. Saurel, A compressible flow model with capillary effects, *J. Comput. Phys.* 209 (2005) 139–178.
- [41] R. Abgrall, M.G. Rodio, Discrete equation method (DEM) for the simulation of viscous, compressible, two-phase flows, *Comput. Fluids* 91 (2014) 164–181.
- [42] R. Saurel, F. Petitpas, R. Abgrall, Modelling phase transition in metastable liquids: application to cavitating and flashing flows, *J. Fluid Mech.* 607 (2008) 313–350.
- [43] S.H. Bryngelson, K. Schmidmayer, V. Coralic, J.C. Meng, K. Maeda, T. Colonius, MFC: an open-source high-order multi-component, multi-phase, and multi-scale compressible flow solver, *Comput. Phys. Commun.* 266 (2021) 107396.
- [44] K. Schmidmayer, S.H. Bryngelson, T. Colonius, An assessment of multicomponent flow models and interface capturing schemes for spherical bubble dynamics, *J. Comput. Phys.* 402 (2020) 109080.
- [45] A.B. Wood, *A Textbook of Sound*, G. Bell and Sons Ltd., 1930.
- [46] R. Saurel, F. Petitpas, R.A. Berry, Simple and efficient relaxation methods for interfaces separating compressible fluids, cavitating flows and shocks in multiphase mixtures, *J. Comput. Phys.* 228 (2009) 1678–1712.
- [47] A. Zein, M. Hantke, G. Warnecke, Modeling phase transition for compressible two-phase flows applied to metastable liquids, *J. Comput. Phys.* 229 (2010) 2964–2998.
- [48] M. Pelanti, K.-M. Shyue, A numerical model for multiphase liquid-vapor-gas flows with interfaces and cavitation, *Int. J. Multiph. Flow* 113 (2019) 208–230.
- [49] A.D. Demou, N. Scapin, M. Pelanti, L. Brandt, A pressure-based diffuse interface method for low-Mach multiphase flows with mass transfer, *J. Comput. Phys.* 448 (2022) 110730.
- [50] M. Rodriguez, S.H. Bryngelson, S. Cao, T. Colonius, Acoustically-induced bubble growth and phase change dynamics near compliant surfaces, in: 11th International Symposium on Cavitation, 2021.
- [51] B. Dorschner, L. Biasiori-Poulanges, K. Schmidmayer, H. El-Rabii, T. Colonius, On the formation and recurrent shedding of ligaments in droplet aero-breakup, *J. Fluid Mech.* 904 (2020).
- [52] V. Coralic, T. Colonius, Finite-volume weno scheme for viscous compressible multicomponent flows, *J. Comput. Phys.* 274 (2014) 95–121.
- [53] H. Guillard, M. Labois, *Numerical Modelling of Compressible Two-Phase Flows*, 2006.
- [54] P. Gaillard, V. Giovangigli, L. Matuszewski, A diffuse interface lox/hydrogen transcritical flame model, *Combust. Theory Model.* 20 (2016) 486–520.
- [55] A. Chiapolino, P. Boivin, R. Saurel, A simple and fast phase transition relaxation solver for compressible multicomponent two-phase flows, *Comput. Fluids* 150 (2017) 31–45.
- [56] E. Goncalves, R.F. Patella, Numerical simulation of cavitating flows with homogeneous models, *Comput. Fluids* 38 (2009) 1682–1696.
- [57] S. Clerc, Numerical simulation of the homogeneous equilibrium model for two-phase flows, *J. Comput. Phys.* 161 (2000) 354–375.
- [58] S.V. Apte, P. Moin, Spray modeling and predictive simulations in realistic gas-turbine engines, in: *Handbook of Atomization and Sprays*, Springer, 2011, pp. 811–835.
- [59] F. Zhang, D. Frost, P. Thibault, S. Murray, Explosive dispersal of solid particles, *Shock Waves* 10 (2001) 431–443.
- [60] M. Herrmann, A dual-scale LES subgrid model for turbulent liquid/gas phase interface dynamics, in: ASME/JSME/KSME 2015 Joint Fluids Engineering Conference, American Society of Mechanical Engineers, 2015, V001T21A001.
- [61] L. Han, X. Hu, N.A. Adams, Scale separation for multi-scale modeling of free-surface and two-phase flows with the conservative sharp interface method, *J. Comput. Phys.* 280 (2015) 387–403.
- [62] G. Tomar, D. Fuster, S. Zaleski, S. Popinet, Multiscale simulations of primary atomization, *Comput. Fluids* 39 (2010) 1864–1874.
- [63] C.-H. Chang, M.-S. Liou, A robust and accurate approach to computing compressible multiphase flow: stratified flow model and AUSM+-up scheme, *J. Comput. Phys.* 225 (2007) 840–873.
- [64] S. Tokareva, E.F. Toro, HLLC-type Riemann solver for the Baer–Nunziato equations of compressible two-phase flow, *J. Comput. Phys.* 229 (2010) 3573–3604.
- [65] N.T. Nguyen, M. Dumbser, A path-conservative finite volume scheme for compressible multi-phase flows with surface tension, *Appl. Math. Comput.* 271 (2015) 959–978.
- [66] R. Saurel, A. Chinnayya, Q. Carmouze, Modelling compressible dense and dilute two-phase flows, *Phys. Fluids* 29 (2017) 063301.
- [67] K. Schmidmayer, J. Caze, F. Petitpas, E. Daniel, N. Favrie, Modelling interactions between waves and diffused interfaces, *Int. J. Numer. Methods Fluids* (2021) 1–27.
- [68] D. Furfarò, R. Saurel, L. David, F. Beauchamp, Towards sodium combustion modeling with liquid water, *J. Comput. Phys.* 403 (2020) 109060.
- [69] D. Dyson, S. Vasu, A. Arakelyan, N. Berube, S. Briggs, J. Ramirez, E.M. Ninnemann, K. Thurmond, G. Kim, W.H. Green, et al., Detonation wave-induced breakup and combustion of RP-2 fuel droplets, in: AIAA SciTech 2022 Forum, 2022, AIAA–2022–1453.
- [70] R.K. Shukla, C. Pantano, J.B. Freund, An interface capturing method for the simulation of multi-phase compressible flows, *J. Comput. Phys.* 229 (2010) 7411–7439.
- [71] S.S. Jain, A. Mani, P. Moin, A conservative diffuse-interface method for compressible two-phase flows, *J. Comput. Phys.* 418 (2020) 109606.

- [72] M.S. Dodd, A. Ferrante, On the interaction of Taylor length scale size droplets and isotropic turbulence, *J. Fluid Mech.* 806 (2016) 356–412.
- [73] S. Fechter, C.-D. Munz, C. Rohde, C. Zeiler, A sharp interface method for compressible liquid-vapor flow with phase transition and surface tension, *J. Comput. Phys.* 336 (2017) 347–374.
- [74] K. Schmidmayer, F. Petitpas, E. Daniel, N. Favrie, S. Gavrilyuk, A model and numerical method for compressible flows with capillary effects, *J. Comput. Phys.* 334 (2017) 468–496.
- [75] M. Pelanti, K.-M. Shyue, A mixture-energy-consistent six-equation two-phase numerical model for fluids with interfaces, cavitation and evaporation waves, *J. Comput. Phys.* 259 (2014) 331–357.
- [76] S. Tanguy, T. Ménard, A. Berlemont, A level set method for vaporizing two-phase flows, *J. Comput. Phys.* 221 (2007) 837–853.
- [77] A. Chiapolino, R. Saurel, B. Nkonga, Sharpening diffuse interfaces with compressible fluids on unstructured meshes, *J. Comput. Phys.* 340 (2017) 389–417.
- [78] M. Akiki, T.P. Gallagher, S. Menon, Mechanistic approach for simulating hot-spot formations and detonation in polymer-bonded explosives, *AIAA J.* 55 (2017) 585–598.
- [79] M. Salvadori, P. Tudisco, D. Ranjan, S. Menon, Numerical investigation of mass flow rate effects on multiplicity of detonation waves within a H₂/Air rotating detonation combustor, *Int. J. Hydrog. Energy* 47 (2022) 4155–4170.
- [80] R. Baurle, G. Alexopoulos, H. Hassan, Assumed joint probability density function approach for supersonic turbulent combustion, *J. Propuls. Power* 10 (1994) 473–484.
- [81] N. Patel, S. Menon, Simulation of spray-turbulence-flame interactions in a lean direct injection combustor, *Combust. Flame* 153 (2008) 228–257.
- [82] K. Balakrishnan, D. Nance, S. Menon, Simulation of impulse effects from explosive charges containing metal particles, *Shock Waves* 20 (2010) 217–239.
- [83] E.F. Toro, The hll and hllc Riemann solvers, in: *Riemann Solvers and Numerical Methods for Fluid Dynamics*, Springer, 2009, pp. 315–344.
- [84] K.-M. Shyue, F. Xiao, An Eulerian interface sharpening algorithm for compressible two-phase flow: the algebraic THINC approach, *J. Comput. Phys.* 268 (2014) 326–354.
- [85] L.D. Gryngarten, S. Menon, A generalized approach for sub-and super-critical flows using the local discontinuous Galerkin method, *Comput. Methods Appl. Mech. Eng.* 253 (2013) 169–185.
- [86] S.S. Jain, M.C. Adler, J.R. West, A. Mani, P. Moin, S.K. Lele, Assessment of diffuse-interface methods for compressible multiphase fluid flows and elastic-plastic deformation in solids, *arXiv preprint arXiv:2109.09729*, 2021.
- [87] T.J. Poinsot, S.K. Lele, Boundary conditions for direct simulations of compressible viscous flows, *J. Comput. Phys.* 101 (1992) 104–129.
- [88] P. Sridharan, T.L. Jackson, J. Zhang, S. Balachandar, Shock interaction with one-dimensional array of particles in air, *J. Appl. Phys.* 117 (2015) 075902.
- [89] X. Rogue, G. Rodriguez, J. Haas, R. Saurel, Experimental and numerical investigation of the shock-induced fluidization of a particles bed, *Shock Waves* 8 (1998) 29–45.
- [90] J.B. Keller, M. Miksis, Bubble oscillations of large amplitude, *J. Acoust. Soc. Am.* 68 (1980) 628–633.
- [91] U. Ghia, K. Ghia, C. Shin, High-Re solutions for incompressible flow using the Navier-Stokes equations and a multigrid method, *J. Comput. Phys.* 48 (1982) 387–411.
- [92] F.M. White, I. Corfield, *Viscous Fluid Flow*, vol. 3, McGraw-Hill, New York, 2006.
- [93] D. Igra, T. Ogawa, K. Takayama, A parametric study of water column deformation resulting from shock wave loading, *At. Sprays* 12 (2002).
- [94] H. Chen, Two-dimensional simulation of stripping breakup of a water droplet, *AIAA J.* 46 (2008) 1135–1143.
- [95] C.T. Crowe, J.D. Schwarzkopf, M. Sommerfeld, Y. Tsuji, *Multiphase Flows with Droplets and Particles*, CRC Press, 2011.
- [96] L. Rayleigh, et al., On the capillary phenomena of jets, *Proc. R. Soc. Lond.* 29 (1879) 71–97.
- [97] T. Gallagher, M. Akiki, S. Menon, V. Sankaran, V. Sankaran, Development of the generalized MacCormack scheme and its extension to low Mach number flows, *Int. J. Numer. Methods Fluids* 85 (2017) 165–188.
- [98] N.V. Patel, Simulation of hydrodynamic fragmentation from a fundamental and an engineering perspective, Ph.D. thesis, Georgia Institute of Technology, 2007.
- [99] U. Fey, M. König, H. Eckelmann, A new Strouhal-Reynolds-number relationship for the circular cylinder in the range $47 < \text{re} < 2 \times 10^5$, *Phys. Fluids* 10 (1998) 1547–1549.
- [100] S. Temkin, S.S. Kim, Droplet motion induced by weak shock waves, *J. Fluid Mech.* 96 (1980) 133–157.
- [101] M. Pilch, C. Erdman, Use of breakup time data and velocity history data to predict the maximum size of stable fragments for acceleration-induced breakup of a liquid drop, *Int. J. Multipiph. Flow* 13 (1987) 741–757.
- [102] K.W. Thompson, Time dependent boundary conditions for hyperbolic systems, *J. Comput. Phys.* 68 (1987) 1–24.
- [103] M. Salvadori, A. Panchal, D. Ranjan, S. Menon, Numerical study of detonation propagation in H₂-Air with kerosene droplets, in: *AIAA SciTech 2022 Forum*, 2022, AIAA-2022-0394.
- [104] J. Kindracki, Experimental research on rotating detonation in liquid fuel-gaseous air mixtures, *Aerosp. Sci. Technol.* 43 (2015) 445–453.
- [105] M. Gogulya, M. Makhov, A.Y. Dolgorobodov, M. Brazhnikov, V. Arkhipov, V. Shchetinin, Mechanical sensitivity and detonation parameters of aluminized explosives, *Combust. Explos. Shock Waves* 40 (2004) 445–457.
- [106] K. Kailasanath, E. Oran, J. Boris, T. Young, Determination of detonation cell size and the role of transverse waves in two-dimensional detonations, *Combust. Flame* 61 (1985) 199–209.
- [107] M. Pelanti, Arbitrary-rate relaxation techniques for the numerical modeling of compressible two-phase flows with heat and mass transfer, *arXiv preprint arXiv:2108.00556*, 2021.