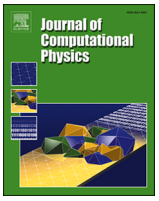
Journal of Computational Physics 556 (2026) 114808



Contents lists available at ScienceDirect

Journal of Computational Physics

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





Efficient entropy-conserving/stable discontinuous Galerkin solution of the multicomponent compressible Euler equations

D. Regener Roig [a,*], A. Crivellini [b], A. Colombo [a]

[a] University of Bergamo Department of Engineering and Applied Sciences, viale Marconi 5, Dalmine (BG), 24044, Italy
[b] Polytechnic University of Marche Department of Industrial Engineering and Mathematical Sciences, via Brecce Bianche 12, Ancona, 60131, Italy

ARTICLE INFO

Keywords:
Discontinuous galerkin
Entropy conservation/stability
Entropy variables
Direct enforcement of entropy balance
Entropy projection
Compressible multicomponent flows
Diffuse interface model

ABSTRACT

This paper presents the development of an efficient discontinuous Galerkin (dG) solver for the multicomponent compressible Euler equations. The method provides global entropy conservation/stability at the discrete level, contributing to the robustness of the computations, cf. [4,19]. The unsteady term of the governing equations is formulated for the conservative variables, while the spatial discretization is assembled from the L_2-projection of the entropy variables [4,19] onto the dG function space, as suggested by Chan et al. [44] and Alberti et al. [35]. This approach requires numerical *over-integration* to ensure entropy conservation/stability, significantly degrading the computational performance. The Direct Enforcement of Entropy Balance (DEEB) proposed by Abgrall in [11] is implemented to avoid this. The DEEB consists of an explicit correction to the discretization to avoid unphysical entropy evolution. As high-order discretizations give rise to spurious oscillations at flow discontinuities, a directional shock-capturing term is added to the discretized equations. The performance of the solver is compared to alternative approaches, *i.e.*, solving directly for the conservative or the entropy variables, by computing several one-dimensional cases. The convergence of the numerical solution is also tested using the method of manufactured solutions (MMS). The interactions of a shock wave with a circular and a square inhomogeneity are finally considered, assessing the accuracy of the solver for reproducing complex two-dimensional phenomena.

1. Introduction

The development of accurate and robust numerical methods for the solution of compressible multicomponent gas mixtures is crucial for understanding complex phenomena arising in such systems. These complexities mainly appear from the interactions of single-component-like non-linear waves with the inhomogeneous media, presenting material interfaces and steep gradients in the thermodynamic properties. Although often subtle, pure multicomponent effects can be critical in fields like combustion, detonation and aerospace applications. Of particular interest are the mixing mechanisms induced by large-scale phenomena and their interactions [1,2]. High-order discretizations are a popular choice for accurately resolving the small-scale structures triggered by these interactions due to their superior accuracy. Numerical approaches for multicomponent flows can be generally classified into two main categories: interface-capturing methods, which solve the profiles and positions of diffused interfaces (typically requiring a higher spatial resolution), and front-tracking methods, which use algorithms to establish the position of sharp interfaces, but they often lose

* Corresponding author.
E-mail addresses: daniel.regener@unibg.it (D. Regener Roig), a.crivellini@staff.univpm.it (A. Crivellini), alessandro.colombo@unibg.it (A. Colombo).

https://doi.org/10.1016/j.jcp.2026.114808
Received 12 April 2025; Received in revised form 9 February 2026; Accepted 22 February 2026
Available online 24 February 2026
0021-9991/© 2026 The Authors. Published by Elsevier Inc. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

the conservation properties [3]. In this work, we adopt the interface-capturing approach to preserve conservation constraints while accurately resolving geometrically complex interfaces.

The multicomponent Euler (MCE) model in symmetric form [4,5] is considered, which extends the classical Euler equations by separating the mass conservation law into individual equations for each component of the mixture. Additionally, the model assumes mechanical and thermal equilibrium, so in principle, the difference between mass fractions in each cell cannot be "too large" [2]. This assumption limits the accuracy of the model when dealing with highly heterogeneous mixtures. Additionally, the MCE model is susceptible to numerical artefacts that do not occur in the single-component Euler model. These include possible negative partial densities and pressure oscillations at the material interfaces [6]. To properly control these phenomena, evolving the solution through physically meaningful states, is mandatory.

Karni [6] demonstrated that pressure disequilibrium at material interfaces arises from how thermodynamic properties of the mixture are computed, which fails to maintain a constant pressure distribution in conservative schemes. To address this issue, a non-conservative method was introduced that accurately simulates contact discontinuities and preserves the positivity of the mass fractions [6], with conservation errors under control in the absence of strong shocks. To complement the non-conservative schemes, Abgrall [5] proposed a formulation that evolves $\gamma/(\gamma-1)$ instead of the mass fractions or the partial densities, becoming the reference solution to mitigate pressure oscillations. Alternatives include the simple single-fluid methods with a "double flux" approach [7], though these violate energy conservation. According to Hou and Le Floch [8], only conservative methods can correctly capture the shock position and speed, highlighting a fundamental limitation of non-conservative approaches. To maintain the conservation properties while mitigating the instability in dG methods, Franchina et al. [9] proposed to use the L_2-projection of the ideal gas equation of state, aligning pressure with the same functional space of the set of primitive variables. Recently, Ching et al. [10] proposed to maintain the pressure equilibrium by solving an equation for the pressure instead of total energy, and the conservation properties were recovered by means of an explicit correction term based on the L_2-projection of the derivative of the total energy [11,12]. Other recent fully conservative approaches have also been developed to simultaneously address pressure equilibrium and conservation, employing strategies based on discrete compatibility conditions [13], entropy stability [14], and approximate equilibrium-preserving fluxes for real-fluid equations of state [15]. In the present paper, we adopt a fully conservative approach, although we recognize its susceptibility to instabilities at contact discontinuities.

To handle the negative values of partial densities that may occur during computations, a common approach is to clip these values to zero. However, this method compromises the conservation properties of the scheme and introduces low-order errors [16]. A more refined approach involves limiting the mass fractions and rescaling them to ensure that they sum to one [9]. In modal dG methods, more sophisticated strategies can include damping higher modes from the solution until positivity is achieved [17,18]. For dG spectral element methods (dG-SEM), other limiters are available that preserve positivity at the nodal values while also applying a conservative correction [16,19] or filtering [20]. Another strategy to avoid obtaining physically non-realistic solutions is to use an alternative set of variables, since the limitations in stability when solving for the conservative set of variables are well-known [21]. With this objective, Bassi et al. [22] propose to evolve in time the primitive set but using the logarithm of the positive-defined variables of the vector. The comparison of this set with the conservative one shows that the former is substantially more robust for highly compressible flows [23]. Compared to using an entropy-conserving (EC)/-stable (ES) framework, this primitive-logarithmic variables have been shown to be more robust in extremely demanding scenarios [21], but it can be more computationally expensive in the context of modal dG methods. Entropy-conserving/-stable methods guarantee a physically consistent evolution of entropy over time, thereby ensuring compliance with the second law of thermodynamics. In this context, the work of Tadmor on EC/ES finite volume discretizations of conservation laws introduces properly designed numerical convective fluxes, forming the basis for many EC/ES numerical methods [24]. Ismail and Roe proposed a closed form for the EC convective flux of the Euler equations, along with an upwind-like ES dissipation operator that is now widely used, [25]. The construction of such schemes for more general single-component models is an active field of research [26–28]. Gouasmi et al., inspired by the work of Ismail and Roe [25] and Chandrashekar [29], addressed the entropy-conserving/stable discretization of the multicomponent Euler model [4]. Another form to construct entropy-stable schemes is to add an artificial viscosity which is function to the violation of the cell entropy inequality, as recently proposed by Chan [30]. Beyond the entropy stability, properties such as kinetic-energy- and pressure-equilibrium-preservation are increasingly recognized as essential for further enhancing the robustness of high-order methods, particularly for turbulent flows [31,32].

For finite element methods, global entropy-stability/conservation can be enforced by solving for the set of entropy variables as shown by Hughes et al. [33] and Chalot et al. [34], provided that an exact numerical quadrature is performed. In this context, Alberti et al. [21,35] constructed an explicit-in-time modal dG scheme where spatial entropy-conservation/stability was obtained by directly discretizing the non-stationary term in the conservative form while evaluating the spatial discrete operator using the L_2-projection of the conservative variables on the entropy ones, *i.e.*, *entropy projection*. The main advantage of this strategy is to reduce the elemental mass matrix to the identity when using an orthonormal modal basis defined in physical space, significantly enhancing the computational efficiency compared to directly solving for the entropy variables [36]. To further improve the performance of the scheme and eliminate the need for exact integration of the variational form, the Direct Enforcement of Entropy Balance, a correction term proposed by Abgrall et al. [11,12], was also employed by Alberti et al. [21,35]. The use of correction terms that balance the entropy production in space can also be found, for the ADER-dG framework, in [37].

This paper applies these numerical strategies to the multicomponent Euler model, enabling high-fidelity yet cost-effective solutions for complex gas mixture flows involving shock waves. A particular emphasis will be placed on evaluating the effects of the numerical framework on the specific numerical challenges presented by the multicomponent nature, such as sharply capturing the material interface. Lastly, this work is also a preliminary verification step for assessing the suitability of the current framework in the future simulation of compressible multiphase flows.

2. Governing equations and the entropy framework

The MCE model in d dimensions can be expressed for a generic set of variables $\mathbf{w} \in \mathbb{R}^m$ as

$$\frac{\partial \mathbf{q}}{\partial \mathbf{w}} \frac{\partial \mathbf{w}}{\partial t} + \frac{\partial \mathbf{F}_i(\mathbf{w})}{\partial x_i} = 0, \tag{1}$$

for $i = 1, \dots, d$ and where repeated indices imply summation. The conservative set of variables $\mathbf{q} \in \mathbb{R}^m$ is defined as

$$\mathbf{q} = \left[\rho_1, \ \dots, \ \rho_N, \ \rho\mathbf{u}, \ \rho e_t\right]^\top. \tag{2}$$

The partial densities of the N fluids composing the mixture are indicated as $\{\rho_k\}_{k=1}^N$, which implies that $m = N + d + 1$. The velocity is expressed as $\mathbf{u} \in \mathbb{R}^d$. The global density ρ and total specific energy e_t are defined as follows

$$\rho = \sum_{k=1}^{N} \rho_k, \qquad \rho e_t = \rho e + \frac{1}{2}\rho ||\mathbf{u}||^2, \tag{3}$$

where

$$\rho e = \sum_{k=1}^{N} \rho_k e_k = T \sum_{k=1}^{N} \rho_k c_{v,k}. \tag{4}$$

The symbols e and e_k represent the global and partial specific internal energy, respectively; T denotes the mixture temperature and $||\circ||$ stands for the Euclidean norm. The definition of ρe is based on the assumptions of ideal gases and thermal equilibrium between components. The thermodynamic coefficients of the mixture are expressed as follows

$$\rho c_v = \sum_{k=1}^{N} \rho_k \, c_{v,k}, \qquad \rho c_p = \sum_{k=1}^{N} \rho_k \, c_{p,k}, \qquad \gamma = \frac{c_p}{c_v}, \tag{5}$$

where $c_{v,k}$ and $c_{p,k}$ are the constant volume and constant pressure adiabatic expansion coefficients for each component k of the mixture. The physical flux $\mathbf{F} \in \mathbb{R}^{m \times d}$ is defined as

$$\mathbf{F} = \left[\rho_1 \mathbf{u}^\top, \ \dots, \ \rho_N \mathbf{u}^\top, \ \rho(\mathbf{u} \otimes \mathbf{u}) + p\mathbf{I}^d, \ \left(\rho e_t + p\right)\mathbf{u}^\top\right]^\top, \tag{6}$$

where $\mathbf{I}^d \in \mathbb{R}^{d \times d}$ is the identity matrix, the outer product is defined as $(\mathbf{u} \otimes \mathbf{v})_{ij} = u_i v_j$, and p is accumulated from the partial pressures p_k using Dalton's law. These partial pressures are defined by the equation of state for ideal gases, resulting in

$$p = \sum_{k=1}^{N} (\gamma_k - 1)\rho_k e_k. = \sum_{k=1}^{N} \rho_k r_k T; \tag{7}$$

The speed of sound c of the mixture can be obtained from the mixture specific internal energy e and the global heat capacity ratio γ as

$$c = \sqrt{\gamma(\gamma - 1)e}. \tag{8}$$

Furthermore, it is worth mentioning that for the present flow model the composition of the mixture is often alternatively expressed as the set of the mass fractions, each defined as $\{Y_k = \rho_k/\rho\}_{k=1}^N$.

Lax and Friedrichs [38] stated that a conservation equation like Eq. (1) can be equipped with a suitable entropy function-entropy flux pair such that

$$\left(\frac{\partial \mathcal{S}(\mathbf{q})}{\partial \mathbf{q}}\right)^\top \left(\frac{\partial \mathbf{F}_i(\mathbf{q})}{\partial \mathbf{q}}\right) = \frac{\partial \mathcal{F}_i(\mathbf{q})}{\partial \mathbf{q}}, \tag{9}$$

with the mathematical entropy $\mathcal{S}$ being a convex function of $\mathbf{q}$, and $\mathcal{F}_i$ being the entropy flux. For the MCE model, there exists a set of entropy variables $\mathbf{v}$ defined by

$$\mathbf{v}(\mathbf{q}) = \frac{\partial \mathcal{S}(\mathbf{q})}{\partial \mathbf{q}}, \tag{10}$$

with the mapping $\mathbf{q} \mapsto \mathbf{v}$ being one-to-one [39]. Furthermore, Eq. (1) becomes symmetric in the form

$$\frac{\partial \mathbf{q}(\mathbf{v})}{\partial \mathbf{v}} \frac{\partial \mathbf{v}}{\partial t} + \frac{\partial \mathbf{F}_i(\mathbf{q}(\mathbf{v}))}{\partial \mathbf{q}(\mathbf{v})} \frac{\partial \mathbf{q}(\mathbf{v})}{\partial \mathbf{v}} \frac{\partial \mathbf{v}}{\partial x_i} = 0. \tag{11}$$

Using this fact, Mock [39] proposed the entropy potential θ and the entropy potential flux ψ, defined as

$$\theta = \mathbf{v}^\top \mathbf{q} - \mathcal{S}, \qquad \psi_i = \mathbf{v}^\top \mathbf{F}_i - \mathcal{F}_i; \tag{12}$$

from which the relations ensue

$$\left(\frac{\partial \theta}{\partial \mathbf{v}}\right)^\top = \mathbf{q}, \qquad \left(\frac{\partial \psi_i}{\partial \mathbf{v}}\right)^\top = \mathbf{F}_i. \tag{13}$$

By computing the divergence of ψ_i and enforcing the compatibility condition of Eq. (9), the following statement is obtained

$$\frac{\partial \psi_i}{\partial x_i} = \frac{\partial \mathbf{v}^\top}{\partial x_i}\mathbf{F}_i. \tag{14}$$

This last relation will prove useful for the study of the discrete entropy stability, as to be shown in the following.

The system of governing equations of the MCE model, when multiplied by $(\partial S/\partial \mathbf{q})^\top$, results in an additional conservation law [39]

$$\frac{\partial \mathcal{S}}{\partial t} + \frac{\partial \mathcal{F}_i}{\partial x_i} = 0, \tag{15}$$

which states that, under the assumption of a smooth regime, entropy is conserved, which is a physically valid assumption in the smooth regime. For the case of non-smooth flows (*e.g.*, in the presence of shocks and sharp gradients) then the entropy relation should be extended to take into account irreversible effects and mandates an entropy inequality [40,41]

$$\frac{\partial \mathcal{S}}{\partial t} + \frac{\partial \mathcal{F}_i}{\partial x_i} \leq 0, \tag{16}$$

meaning that, under any regime, the mathematical entropy $\mathcal{S}$ of a hyperbolic conservation law must either be conserved or destroyed over time. Following the work of Renac [19] and Gouasmi et al. [4] a mathematical entropy-entropy flux pair is supplied to the MCE system

$$(\mathcal{S}, \mathcal{F}) = (-\rho s, -\rho \mathbf{u} s), \tag{17}$$

with the global specific entropy s defined as

$$\rho s = \sum_{k=1}^{N} \rho_k s_k \qquad \text{where} \qquad s_k = c_{v,k}\log(T) - r_k \log(\rho_k). \tag{18}$$

Therefore, it is established from Eq. (16) that the physical entropy ρs must be conserved or increased over time, following the second principle of thermodynamics. Provided the definition of $\mathcal{S}(\mathbf{q})$ and using Eq. (10), the set of entropy variables for the MCE model is defined

$$\mathbf{v} = \frac{1}{T}\left[h_1 - Ts_1 - \frac{||\mathbf{u}||^2}{2}, \ \ldots, \ h_N - Ts_N - \frac{||\mathbf{u}||^2}{2}, \ \mathbf{u}, \ -1\right]^\top \tag{19}$$

where the component-specific enthalpy $\{h_k\}_{k=1}^N$ is

$$h_k = e_k + r_k T, \tag{20}$$

and can be accumulated to compute the global specific enthalpy h and total specific enthalpy h^t, resulting in

$$\rho h = \sum_{k=1}^{N} \rho_k h_k \qquad \text{and} \qquad h^t = h + \frac{1}{2}||\mathbf{u}||^2. \tag{21}$$

Once the entropy variables are defined, Eq. (12) is used to obtain the entropy potential and the entropy potential flux

$$\theta = \sum_{k=1}^{N} r_k \rho_k, \qquad \psi_i = \sum_{k=1}^{N} r_k \rho_k u_i. \tag{22}$$

3. Numerical framework

The governing equations are spatially discretized using a modal dG method with an orthonormal and hierarchical basis, chosen for its well-demonstrated geometrical flexibility, while maintaining arbitrarily high-order accuracy and stencil compactness [42,43]. Three alternatives will be considered for the solution strategy: i) solve for the conservative variables; ii) solve for the entropy variables; iii) solve for the conservative variables but assemble the spatial residual according to the entropy variables, *i.e.*, *entropy projection*.

In the last case, the Direct Enforcement of Entropy Balance (DEEB) term is added to the discrete equations to avoid *over-integration*. An artificial diffusion term is furthermore employed as a shock-capturing strategy for the stabilization of the solution in the presence of shocks. Appendix A provides a compendium of useful relations for computing the needed terms. All three semi-discrete systems are advanced in time using an explicit Runge-Kutta method.

3.1. Spatial discretization

For the sake of presenting the dG spatial discretization of the MCE equations, we introduce the notation $\mathbf{P}(\mathbf{w})$ to represent the Jacobian that multiplies the temporal derivative in Eq. (1), resulting in

$$\mathbf{P}(\mathbf{w})\frac{\partial \mathbf{w}}{\partial t} + \frac{\partial \mathbf{F}_i(\mathbf{w})}{\partial x_i} = 0 \qquad \text{where} \qquad \mathbf{P}(\mathbf{w}) = \frac{\partial \mathbf{q}}{\partial \mathbf{w}}. \tag{23}$$

To proceed, we multiply Eq. (23) by a set of arbitrary smooth test functions $\Phi \in \mathbb{R}^m$ and integrate over the computational domain $\Omega \in \mathbb{R}^d$, which has a boundary $\partial\Omega$

$$\int_\Omega \Phi^\top \mathbf{P}(\mathbf{w}) \frac{\partial \mathbf{w}}{\partial t}\, d\Omega - \int_\Omega \left(\frac{\partial \Phi}{\partial x_i}\right)^\top \mathbf{F}_i(\mathbf{w})\, d\Omega + \int_{\partial\Omega} \Phi^\top \mathbf{F}_i(\mathbf{w}) n_i\, d\sigma = 0, \tag{24}$$

where n_i are the components of the normal vector pointing outwards of $\partial\Omega$. The physical domain is then discretized in a computational mesh $\mathcal{K}_h = \{K\}$ made of non-overlapping, arbitrarily-shaped elements K, such that

$$\overline{\Omega}_h = \bigcup_{K \in \mathcal{K}_h} \overline{K}. \tag{25}$$

A polynomial space for the discrete approximations of $\mathbf{w}$ and Φ, *i.e.* $\mathbf{w}_h$ and Φ_h, is chosen such as

$$\mathbb{P}^\kappa = \{v_h \in L^2(\Omega) \,|\, v_{h|K} \in \mathbb{P}^\kappa(K), \forall K \in \mathcal{K}_h\} \tag{26}$$

where $\kappa \in \mathbb{Z}_{\geq 0}$; and $\mathbb{P}^\kappa$ denote the restriction to K of the polynomial functions of total degree $\leq \kappa$. As a basis for this space the set of orthonormal and hierarchical functions, defined in the physical space, $\{\phi_j\}_{j=1}^{N_{dof}}\ \forall K \in \mathcal{K}_h$, where $N_{dof} = \prod_{i=1}^{d}(\kappa + i)/i$ is used, as proposed by Bassi et al. [42]. By replacing the continuous vectors $\mathbf{w}$ and Φ by their finite element approximations, both of them belonging to the discrete space $\mathcal{V}_h = \left[\mathbb{P}^\kappa(\mathcal{K}_h)\right]^m$, each component $\{w_{h,k}\}_{k=1}^m$ of $\mathbf{w}_h$ can be expressed in terms of the elements of the global vector of unknown degrees of freedom (DOFs) $\mathbf{W}$ as

$$w_{h,k} = \phi_j W_{k,j} \qquad \text{with} \qquad j = 1, \dots, N_{dof}, \quad k = 1, \dots, m. \tag{27}$$

To complete the dG discretization, we introduce the set of mesh faces $\mathcal{F}_h = \mathcal{F}_h^0 \bigcup \mathcal{F}_h^b$, where $\mathcal{F}_h^0$ are the internal faces of $\mathcal{K}_h$ and $\mathcal{F}_h^b$ are the faces on the boundary of the domain. Let also $\mathcal{F}_K = \{\mathcal{F}\}$ be the set of faces belonging to the element $K, \forall K \in \mathcal{K}_h$. Then, the dG solution of the vectorial conservation law of Eq. (23) consists in seeking the elements of $\mathbf{w}_h$ such that

$$\sum_{K \in \mathcal{K}_h} \int_K \Phi_h^\top \mathbf{P}(\mathbf{w}_h) \frac{\partial \mathbf{w}_h}{\partial t}\, d\Omega - \sum_{K \in \mathcal{K}_h} \int_K \left(\frac{\partial \Phi_h}{\partial x_i}\right)^\top \mathbf{F}_i(\mathbf{w}_h)\, d\Omega + \sum_{F \in \mathcal{F}_h} \int_F [\![\Phi_h]\!]^\top \hat{\mathbf{F}}(\mathbf{w}_h^\pm, \mathbf{n}_F)\, d\sigma = 0, \tag{28}$$

where $\mathbf{n}_F = \{n_{F,i}\}_{i=1}^d$ is the uniquely defined unit vector normal at any point of a possibly curved face F. The terms $\circ^-$ and $\circ^+$ denote the interior and exterior traces of $\circ$, where the normal $\mathbf{n}_F$ is defined to point from the interior (-) to the exterior (+) side of the face. The jump operator $[\![\circ]\!] = \circ^+ - \circ^-$ accounts for the discontinuous nature of the polynomial space at each mesh face $F \in \partial K^+ \bigcup \partial K^-$; $\hat{\mathbf{F}}$ is a suitable numerical flux function in the normal direction that will be discussed in detail in Section 3.5.

3.2. Entropy projection method

To avoid the computational burden of inverting $\mathbf{P}(\mathbf{v}_h)$, in this work the *entropy projection* strategy is adopted, as presented by Alberti et al. [35] according to Chan et al. [44]. In this approach, the numerical solution is advanced in time in terms of the degrees of freedom of the conservative variables, while the residual of the spatial discretization is computed from the L_2-projection of the entropy variables onto the dG function space. The projected entropy variables, denoted by $\mathbf{v}_h^*$, are then defined as

$$\int_K \Phi_h^\top \mathbf{v}_h^*\, d\Omega = \int_K \Phi_h^\top \mathbf{v}(\mathbf{q}_h)\, d\Omega, \tag{29}$$

where this projection can be seen as the best possible polynomial approximation of $\mathbf{v}_h$. Thanks to the orthonormality of the shape functions [45], the DOFs related to the projected entropy variables can be computed directly from $v_k(\mathbf{q}_h)$ as

$$V_{k,j}^* = \int_K \phi_j v_k(\mathbf{q}_h)\, d\Omega \quad \text{where} \quad k = 1, \dots, m \quad \text{and} \quad j = 1, \dots, N_{dof}. \tag{30}$$

3.3. Entropy stability at the discrete level

To obtain further insight into the numerical entropy stability, let the dG discretized equation for the evolution of S be presented as

$$\sum_{K \in \mathcal{K}_h} \mathbf{v}_h^{*\top} \mathbf{P}(\mathbf{w}_h) \frac{\partial \mathbf{w}_h}{\partial t}\, d\Omega - \sum_{K \in \mathcal{K}_h} \int_K \left(\frac{\partial \mathbf{v}_h^*}{\partial x_i}\right)^\top \mathbf{F}_i(\mathbf{w}_h)\, d\Omega + \sum_{F \in \mathcal{F}_h} \int_F [\![\mathbf{v}_h^*]\!]^\top \hat{\mathbf{F}}(\mathbf{w}_h^\pm, \mathbf{n}_F)\, d\sigma = 0, \tag{31}$$

which has been obtained by replacing in Eq. (28) the test functions with the projected entropy variables. It is desired to obtain an equation in conservative form starting from Eq. (31). Therefore, accounting with the following three relations, which always hold in a continuous setting, cf. [35],

$$\sum_{K \in \mathcal{K}_h} \int_K \mathbf{v}_h^{*\top} \mathbf{P}(\mathbf{w}_h) \frac{\partial \mathbf{w}_h}{\partial t}\, d\Omega = \sum_{K \in \mathcal{K}_h} \int_K \frac{\partial S(\mathbf{w}_h)}{\partial \mathbf{w}_h} \frac{\partial \mathbf{w}_h}{\partial t}\, d\Omega, \tag{32}$$

$$\sum_{K \in \mathcal{K}_h} \int_K \left(\frac{\partial \mathbf{v}_h^*}{\partial x_i}\right)^\top \mathbf{F}_i(\mathbf{w}_h)\, d\Omega = \sum_{K \in \mathcal{K}_h} \int_K \frac{\psi_i(\mathbf{w}_h)}{\partial x_i}\, d\Omega = \sum_{F \in \mathcal{F}_h} \int_F [\![\psi_i(\mathbf{w}_h)]\!] n_{F,i}\, d\sigma, \tag{33}$$

$$\sum_{F\in\mathcal{F}_h}\int_F [\![\mathbf{v}_h^*]\!]^\top \hat{\mathbf{F}}(\mathbf{w}_h^\pm,\mathbf{n}_F)\,d\sigma = \sum_{F\in\mathcal{F}_h}\int_F [\![\mathbf{v}_h]\!]^\top \hat{\mathbf{F}}(\mathbf{w}_h^\pm,\mathbf{n}_F)\,d\sigma; \tag{34}$$

the *implicitly resolved* equation for the evolution of S is obtained

$$\sum_{K\in\mathcal{K}_h}\int_K \frac{\partial S(\mathbf{w}_h)}{\partial \mathbf{w}_h}\frac{\partial \mathbf{w}_h}{\partial t}\,d\Omega + \sum_{F\in\mathcal{F}_h}\int_F \left[[\![\mathbf{v}_h]\!]^\top \hat{\mathbf{F}}(\mathbf{w}_h^\pm,\mathbf{n}_F) - [\![\psi_i(\mathbf{w}_h)]\!]n_{F,i}\right] d\sigma = 0. \tag{35}$$

Besides, according to Alberti et al. [35], relations Eqs. (32)–(34) hold in a discrete setting if and only if

- the entropy variables are used as the working set so that $\mathbf{v}(\mathbf{w}_h)$ and $\mathbf{v}_h^*$ are interchangeable in Eq. (34);
- the time discretized LHS of Eq. (32) results in a "correct" discretization of the time derivative of S and
- all the integrals are computed exactly, ensuring the Gauss theorem holds at a discrete level in Eq. (33).

From a practical perspective, the last requirement in general leads to numerical *over-integration*, requiring quadrature rules with a very high degree of exactness, typically unknown in advance and far exceeding the value commonly used in the present modal dG framework, *i.e.*, selecting Gauss formulas able to exactly integrate polynomials of order 2κ over the mesh, such as the entries of the mass matrix. It is worth noticing that this requirement is particularly important in modal dG formulations. In contrast, certain nodal dG frameworks employing Summation-By-Parts (SBP) compatible operators can circumvent this computational burden by design [19,46,47]. This need for exact integration is particularly problematic for non-polynomial functions, which cannot be integrated exactly by Gauss quadrature rules in the present framework. Consequently, *over-integration* inevitably results in a severe degradation of computational performance [35,36].

If Tadmor's condition [24] for a numerical EC/ES flux function is satisfied, *i.e.*,

$$[\![\mathbf{v}_h]\!]^\top \hat{\mathbf{F}}\left(\mathbf{w}_h^\pm,\mathbf{n}_F\right) - [\![\psi_i\left(\mathbf{w}_h\right)]\!]n_{F,i} \geq 0; \tag{36}$$

and incorporated into Eq. (32), it can be shown that the mathematical entropy is either preserved or decreases over time, as expressed by the relation

$$\sum_{K\in\mathcal{K}_h}\int_K \frac{\partial S(\mathbf{w}_h)}{\partial \mathbf{w}_h}\frac{\partial \mathbf{w}_h}{\partial t}\,d\Omega \leq 0. \tag{37}$$

3.4. Direct enforcement of the entropy balance

In this work, to circumvent the use of *over-integration* while preserving numerical entropy conservation/stability properties, the Direct Enforcement of Entropy Balance (DEEB) is considered [11,12]. The DEEB approach consists of adding a correction term to the discretized equations in Eq. (28), resulting in

$$\sum_{K\in\mathcal{K}_h}\int_K \Phi_h^\top \mathbf{P}(\mathbf{w}_h)\frac{\partial \mathbf{w}_h}{\partial t}\,d\Omega - \sum_{K\in\mathcal{K}_h}\int_K \left(\frac{\partial \Phi_h}{\partial x_i}\right)^\top \mathbf{F}_i(\mathbf{w}_h)\,d\Omega + \sum_{F\in\mathcal{F}_h}\int_F [\![\Phi_h]\!]^\top \hat{\mathbf{F}}(\mathbf{w}_h^\pm,\mathbf{n}_F)\,d\sigma + \sum_{K\in\mathcal{K}_h}\alpha_K\frac{\int_K \Phi_h^\top \overline{\mathbf{v}}_h^*\,d\Omega}{\int_K \overline{\mathbf{v}}_h^{*\top}\overline{\mathbf{v}}_h^*\,d\Omega} = 0, \tag{38}$$

where α_K is the elemental correction factor, $\overline{\mathbf{v}}_h^* = \mathbf{v}_h^* - \mathring{\mathbf{v}}_h^*$ and $\mathring{\mathbf{v}}_h^*$ is the mean value of $\mathbf{v}_h^*$. The role of the term $\int_K \Phi_h^\top \overline{\mathbf{v}}_h^* d\Omega / \int_K \overline{\mathbf{v}}_h^{*\top}\overline{\mathbf{v}}_h^* d\Omega$ is to distribute the correction as uniformly as possible among all the DOFs except the first one, with the objective of not affecting the conservation of the mean flow quantities. The distribution term becomes ill-defined when out-of-mean degrees of freedom are null; therefore, in the implementation, the correction is omitted when the denominator is lower than 10^{-13}. We remark that the optimal choice for this distribution term is a topic of recent research, with different possible alternatives, *e.g.*, reformulating α_K to not affect the average [48], formulating the spreading term from a filter [49,50], or making it mimic the natural dissipation [51]. As explained in [35], the orthonormality of the basis functions greatly simplifies the computation of the present distribution term

$$\frac{\int_K \phi_i\phi_j \overline{V}_{\varrho,j}^*\,d\Omega}{\int_K \phi_j \overline{V}_{s,j}^*\phi_l\overline{V}_{s,l}^*\,d\Omega} = \frac{\overline{V}_{\varrho,i}^*}{\overline{V}_{s,l}^*\overline{V}_{s,l}^*} \tag{39}$$

with $\varrho, s = 1,\dots,m$ and $i,j,l = 1,\dots,N_{dof}$. Regarding the elemental correction, when the *entropy projection* is applied, following the work of Abgrall [11] and adopting the formulation of Alberti et al. [35], it can be explicitly expressed as

$$\alpha_K = \int_K \left(\frac{\partial \mathbf{v}_h^*}{\partial x_i}\right)^\top \mathbf{F}_i\left(\mathbf{w}_h\right)\,d\Omega - \sum_{F\in\mathcal{F}_K}\int_F \psi_i(\mathbf{w}_h)n_{F,i}\,d\Omega, \tag{40}$$

where $i = 1,\dots,d$, the entropy potential flux ψ_i is evaluated using the internal state of $\mathbf{w}_h$, and the integrals have to be interpreted as numerically evaluated.

3.5. Numerical convective fluxes

As discussed in Section 3.3, the operator $\hat{\mathbf{F}}$ must satisfy Eq. (36) to ensure an entropy-conserving/stable semi-discrete scheme. For the sake of completeness, this section reports the various numerical fluxes considered in this work, expressed as $\hat{\mathbf{F}} = \left[\hat{F}_{1,\{1,\dots,N\}}, \hat{F}_{2,\{1,\dots,d\}}, \hat{F}_3\right]^\top$.

3.5.1. Entropy-conserving numerical flux

The flux function proposed by Gouasmi et al. [4], inspired by the work of Ismail and Roe [25] and Chandrashekar [29], is used in this paper as an entropy-conserving flux

$$\begin{aligned}
\hat{F}^{EC}_{1,k} &= \rho_k^{ln}\underline{u_n} \qquad \text{for} \qquad k = 1,\dots,N,\\
\hat{F}^{EC}_{2,i} &= \frac{n_{F,i}}{\underline{1/T}}\left(\sum_{k=1}^{N} r_k \underline{\rho_k}\right) + \underline{u_i}\sum_{k=1}^{N}\hat{F}^{EC}_{1,k} \qquad \text{for} \qquad i = 1,\dots,d,\\
\hat{F}^{EC}_{3} &= \sum_{k=1}^{N}\left(c_{v,k}\frac{1}{(1/T)^{ln}} - \frac{1}{2}\underline{u_i u_i}\right)\hat{F}^{EC}_{1,k} + \sum_{i=1}^{d}\underline{u_i}\,\hat{F}^{EC}_{2,i},
\end{aligned} \tag{41}$$

where $u_n = u_i n_{F,i}$. For any scalar quantity $\circ$ the arithmetic and logarithmic averages at a mesh face are defined as a function of the left $\circ^-$ and right $\circ^+$ states as

$$\underline{\circ} = \frac{1}{2}\left(\circ^+ + \circ^-\right), \quad \circ^{ln} = \begin{cases} 1 + \sum_{i=1}^{4}\frac{\mathtt{b}^i}{2i+1}, & \text{if} \quad \mathtt{b} < \epsilon \\ \dfrac{\log(\chi)}{2\mathtt{a}} & \text{otherwise} \end{cases}, \tag{42}$$

where

$$\chi = \frac{\circ^+}{\circ^-}, \qquad \mathtt{a} = \frac{\chi - 1}{\chi + 1}, \qquad \mathtt{b} = \mathtt{a}^2, \tag{43}$$

and $\epsilon \approx 0.02$ is a parameter to prevent singularities in the definition of the logarithmic mean. We remark that the definition of the logarithmic average used here follows Ismail and Roe [25], slightly differing from the definition used by Gouasmi et al. [4].

3.5.2. Rusanov-type numerical flux

A simple way to obtain a dissipative flux is to add a Rusanov-type dissipation to an entropy-conserving flux. Therefore, following Gassner et al. [52], it results that

$$\hat{\mathbf{F}}^{RU}\left(\mathbf{w}^{\pm}, \mathbf{n}_F\right) = \hat{\mathbf{F}}^{EC}\left(\mathbf{w}^{\pm}, \mathbf{n}_F\right) - \hat{\mathbf{D}}\left(\mathbf{w}^{\pm}\right), \tag{44}$$

where the stabilization term is defined as

$$\hat{\mathbf{D}}\left(\mathbf{w}^{\pm}\right) = \frac{1}{2}\lambda_{max}\frac{\partial \mathbf{q}}{\partial \mathbf{v}}(\tilde{\mathbf{w}})[\![\mathbf{w}]\!], \tag{45}$$

where the maximum wave velocity $\lambda_{\max}$ is given by

$$\lambda_{max} = \max\left(|u_n^+| + c^+, |u_n^-| + c^-\right), \tag{46}$$

being $c^{\pm}$ the left/right speed of sound of the mixture at the mesh interface. The Jacobian of the conservative set of variables with respect to the entropy ones is evaluated in the average primitive state defined by Gouasmi et al. [4] as

$$\tilde{\mathbf{w}} = \left[\rho_1^{ln},\ \dots,\ \rho_N^{ln},\ \underline{\mathbf{u}},\ 1/(1/T)^{ln}\right]^{\top}, \tag{47}$$

and taking c_v in Eq. (A.1) as the arithmetic mean at the mesh interface.

3.5.3. Godunov numerical flux

The Godunov flux [53] based on the exact Riemann solver of Borisov and Rykov [54] is also used for the definition of the numerical convective flux function. For the sake of completeness, the solution procedure is detailed in Appendix B. We remark that the Godunov flux has the salient feature of having the lowest numerical viscosity among the schemes that are entropy stable with respect to all convex entropies [53,55].

3.6. Shock capturing approach

In this work, to control the numerical spurious oscillations that may occur at shocks due to the high-order approximation, the strategy proposed by Bassi et al. [56,57] has been extended to stabilize the solution of the MCE equations. In practice, an artificial diffusion term is added to the discretized governing equations

$$\sum_{K\in\mathcal{K}}\int_K \varepsilon_p\left(\frac{\partial \Phi_h}{\partial x_i}b_i\right)^{\top}\left(\frac{\partial \mathbf{q}(\mathbf{w}_h)}{\partial x_i}b_i\right)d\Omega, \tag{48}$$

where the spatial gradient of the conservative variables is computed from the chain rule as

$$\frac{\partial \mathbf{q}(\mathbf{w}_h)}{\partial x_i}b_i = \frac{\partial \mathbf{q}(\mathbf{w}_h)}{\partial \mathbf{w}_h}\frac{\partial \mathbf{w}_h}{\partial x_i}b_i \tag{49}$$

and the components of the unit vector in the direction of the pressure spatial gradient, b_i, are defined as

$$b_i = \frac{\partial p(\mathbf{w}_h)/\partial x_i}{\sqrt{\left(\partial p(\mathbf{w}_h)/\partial x_j\right)\left(\partial p(\mathbf{w}_h)/\partial x_j\right)} + \epsilon} \tag{50}$$

with ϵ being a small value of the order of machine precision; the index $j = 1, \dots, d$; and repeated indices imply summation. The shock capturing sensor ε_p is determined as

$$\varepsilon_p = C_{SC}\left(h_K^{SC}\right)^2 \frac{|c_p| + |d_p|}{p(\mathbf{w}_h)} f_p, \tag{51}$$

being C_{SC} a user-defined constant. As long as there are discontinuities in the flow problem, this value is set to 0.2 in the present work. The terms composing the shock sensor can be evaluated as

$$c_p = \frac{\partial p(\mathbf{w}_h)}{\partial \mathbf{q}(\mathbf{w}_h)}\mathbf{c}, \tag{52}$$

$$d_p = \frac{\partial p(\mathbf{w}_h)}{\partial \mathbf{q}(\mathbf{w}_h)} \frac{\partial \mathbf{F}_i(\mathbf{w}_h)}{\partial x_i}, \tag{53}$$

$$f_p = \frac{1}{p(\mathbf{w}_h)} \sqrt{\frac{p(\mathbf{w}_h)}{\partial x_i} \frac{p(\mathbf{w}_h)}{\partial x_i}} \left(\frac{h_K^{SC}}{\max(1, \kappa)} \right). \tag{54}$$

In Eq. (52) the term $\mathbf{c} \in \left[\mathcal{V}_h\right]^m$ comes from the solution of the auxiliary problem

$$\int_K \Phi^\top \mathbf{c}\, d\Omega = \sum_{F \in \mathcal{F}_K} \int_F \Phi^\top \left[\hat{\mathbf{F}}(\mathbf{w}_h^\pm, \mathbf{n}_F) - \mathbf{F}_i(\mathbf{w}_h)\, n_{F,i} \right] d\sigma. \tag{55}$$

where the physical flux $\mathbf{F}_i$ is evaluated using the internal state of $\mathbf{w}_h$. The divergence of the physical flux in Eq. (53) is in practice computed by means of the chain rule

$$\frac{\partial \mathbf{F}_i(\mathbf{w}_h)}{\partial x_i} = \frac{\partial \mathbf{F}_i(\mathbf{w}_h)}{\partial \mathbf{w}_h} \frac{\partial \mathbf{w}_h}{\partial x_i}. \tag{56}$$

Note that, due to the use of an orthonormal basis, the degrees of freedom of the function $\mathbf{c}$ can be computed directly by evaluating a volume integral, avoiding the need to solve a linear system [42]. Finally, the characteristic size of the cell, h_K^{SC}, is defined as

$$h_K^{SC} = \frac{1}{\sqrt{\left(\frac{1}{\Delta x_i}\right)\left(\frac{1}{\Delta x_i}\right)}}, \tag{57}$$

where, for $d = 2$, $\{\Delta x_i\}_{i=1}^d$ represent the dimensions of the quadrilateral enclosing K, scaled such that their product equals the surface area of the cell. For $d = 1$, h_K^{SC} is set to the length of the segment.

For the sake of completeness, the closed forms for the pressure gradient in space, the pressure gradient with respect to entropy variables, the Jacobian of the flux function with respect to the working set of variables and the Jacobian of the change of variables are reported in Appendix A.

3.7. Semi-discrete systems

The three different semi-discrete systems considered in this work are presented in the following, with a brief discussion on their main features. When directly solving for the conservative variables ($\mathbf{w}_h = \mathbf{q}_h$), the dG discretization of the MCE equations consists in seeking the elements of the vector of unknown degrees of freedom $\mathbf{W} = \mathbf{Q}$, such that

$$\sum_{K \in \mathcal{K}_h} \frac{dQ_{k,j}(t)}{dt} - \sum_{K \in \mathcal{K}_h} \int_K \frac{\partial \phi_j}{\partial x_i} F_{k,i}(\mathbf{q}_h)\, d\Omega + \sum_{F \in \mathcal{F}_h} \int_F [\![\phi_j]\!] \hat{F}_k(\mathbf{q}_h^\pm, \mathbf{n}_F)\, d\sigma = 0, \tag{58}$$

for $k = 1, \dots, m$ and $j = 1, \dots, N_{dof}$. When directly computing the residual of the spatial discretization from the conservative variables, the relations in Eqs. (33)–(34) no longer hold, and the scheme cannot achieve the entropy conservation/stability. Although this strategy is expected to be the least robust among those investigated, due to the lack of a constraint on the evolution of the numerical entropy, it is expected to be the most efficient, as it implies the lowest number of operations for the assembly of the residual and the advancement of the solution in time.

The second option is to solve for the entropy variables ($\mathbf{w}_h = \mathbf{v}_h$), thus advancing in time the modes $\mathbf{V}$, such that

$$\sum_{K \in \mathcal{K}_h} \int_K \phi_j P_{k,s}(\mathbf{v}_h) \phi_l \frac{dV_{s,l}(t)}{dt}\, d\Omega - \sum_{K \in \mathcal{K}_h} \int_K \frac{\partial \phi_j}{\partial x_i} F_{k,i}(\mathbf{v}_h)\, d\Omega + \sum_{F \in \mathcal{F}_h} \int_F [\![\phi_j]\!] \hat{F}_k(\mathbf{v}_h^\pm, \mathbf{n}_F)\, d\sigma = 0, \tag{59}$$

for $k, s = 1, \dots, m$ and $j, l = 1, \dots, N_{dof}$. As commented in Section 3.4, this strategy requires *over-integration* to ensure entropy conservation/stability, unless corrective terms such as DEEB are implemented. Additionally, the formulation involves inverting a local elemental matrix that arises from the discretization of the unsteady term integral with a significant impact on the computational cost.

Finally, the degrees of freedom of the conservative variables $\mathbf{Q}$ can be advanced in time, while the residual of the spatial discretization is assembled from the projected entropy variables $\mathbf{v}_h^* = \left[v_{h,1}^*, \dots, v_{h,m}^*\right]^\top$, such that

$$\sum_{K\in\mathcal{K}_h} \frac{dQ_{k,j}(t)}{dt} - \sum_{K\in\mathcal{K}_h} \int_K \frac{\partial \phi_j}{\partial x_i} F_{k,i}(\mathbf{v}_h^*)\, d\Omega + \sum_{F\in\mathcal{F}_h} \int_F [\![\phi_j]\!] \hat{F}_k(\mathbf{v}_h^{*\pm}, \mathbf{n}_F)\, d\sigma + \sum_{K\in\mathcal{K}_h} \alpha_K \frac{\int_K \phi_j \overline{v}_{h,k}^*\, d\Omega}{\int_K \overline{v}_{h,s}^* \overline{v}_{h,s}^*\, d\Omega} = 0, \tag{60}$$

for all $k, s = 1, \dots, m$ and $j = 1, \dots, N_{dof}$. The last term on the LHS is the DEEB correction, cf. Sections 3.4 and 3.2, included to enforce the discrete entropy conservation/stability while avoiding the need for *over-integration*. This approach leverages the advantages of the first two strategies and has already been shown to offer an excellent balance between robustness, accuracy and efficiency in the context of the single-component compressible Euler equations [21,35].

3.8. Time integration

By numerically computing the volume and surface integrals in Eqs. (58)–(60) and assembling all the elemental contributions, each formulation can be reduced to a system of Ordinary Differential Equations (ODEs) governing the evolution in time of the corresponding discrete solution

$$\frac{d\mathbf{W}}{dt} = \widetilde{\mathbf{R}}(\mathbf{W}), \tag{61}$$

with

$$\widetilde{\mathbf{R}}(\mathbf{W}) = \mathbf{M_P}(\mathbf{W})^{-1}\mathbf{R}(\mathbf{W}), \tag{62}$$

where $\mathbf{M_P}(\mathbf{W})$ comes from the discretization of the first integral in Eq. (28), while $\mathbf{R}(\mathbf{W})$ is the vector of the residuals of the spatial discretization. In Eq. (62), $\mathbf{M_P}(\mathbf{W})$ reduces to the identity when solving for the set of conservative variables. Otherwise, it incorporates the discretization of the $\partial\mathbf{q}/\partial\mathbf{w}$ matrix. In this latter case, solving the system requires an element-wise inversion of the block-diagonal $\mathbf{M_P}$ matrix.

Eq. (61) is here integrated by using explicit Runge-Kutta methods of different orders of accuracy. In particular, the optimal Strong-Stability-Preserving Runge-Kutta (SSPRK) schemes proposed by Spiteri and Ruuth in [58] are used. Their formulation can be written in compact form as

$$\begin{aligned} \mathbf{Y}_0 &= \mathbf{W}^n \\ \mathbf{Y}_i &= \mathbf{Y}_0 + \Delta t \sum_{j=1}^{i-1} a_{ij} \widetilde{\mathbf{R}}(\mathbf{Y}_j) \\ \mathbf{W}^{n+1} &= \mathbf{W}^n + \Delta t \sum_{j=1}^{s} b_j \widetilde{\mathbf{R}}(\mathbf{Y}_j), \end{aligned} \tag{63}$$

where s is the number of stages, a_{ij} and b_j are the coefficients of the scheme from the Butcher tableau. For the one-dimensional test cases presented in this work, the third-order, five-stage SSPRK scheme is used, whereas for the two-dimensional flow problems, the fourth-order, five-stage SSPRK scheme is employed. One could recover fully-discrete EC/ES properties using alternative time-integration strategies, *e.g.* [12,59–61], but we leave this improvement for future work, since the focus of the present article is the spatial discretization. A fixed time step size can be imposed based on physical arguments and considering the stability of the scheme, or, alternatively, Δt can be computed from the Courant-Friedrich-Lewy (CFL) condition $\Delta t = \min_{K\in\mathcal{K}}\{\mathrm{CFL}\, h_K/(c_{K,0} + |\mathbf{u}_{K,0}|)\}$ for a fixed CFL value. In this case, the elemental convective eigenvalue in the denominator is computed from the cell mean values, and h_K corresponds to the cell size.

4. Numerical results

In this section, we analyse the differences among the three distinct semi-discrete systems of Eqs. (58)–(60) in one dimension. This comparison is conducted through spatial refinement tests using the Method of Manufactured Solutions (MMS) and the advection of a density wave. For the latter, we also analyse the entropy conservation and compare the computational cost to gain further insight into the appealing properties of the *entropy projection* approach. The comparison is continued with a series of Riemann problems of increasing complexity, to investigate how the properties of the schemes affect the resolution of flows with discontinuities. Based on these results, we select the *entropy projection* strategy as a good compromise that improves robustness at a reasonable computational cost, and then study its performance in multidimensional cases.

Therefore, we first validate the approach in multiple dimensions by using the MMS. Additionally, we extend the implosion case from [62], originally designed for a single-component flow, to accommodate an additional component, and we examine the effect of κ refinement on the emerging travelling contact wave. More complex flow patterns are examined by analysing the interaction of a

Table 1
MMS1D – constants and trigonometric functions for the manufactured solution.

ψ	$\mathtt{a}_0^\psi$	$\mathtt{a}_1^\psi$	$\mathtt{f}_1^\psi$	$\mathtt{b}_1^\psi$
Y_1	0.5	0.1	sin	1.0
ρ	1	−0.1	sin	1.0
u	800	50	sin	1.5
p	1×10^5	0.2×10^5	cos	1.5

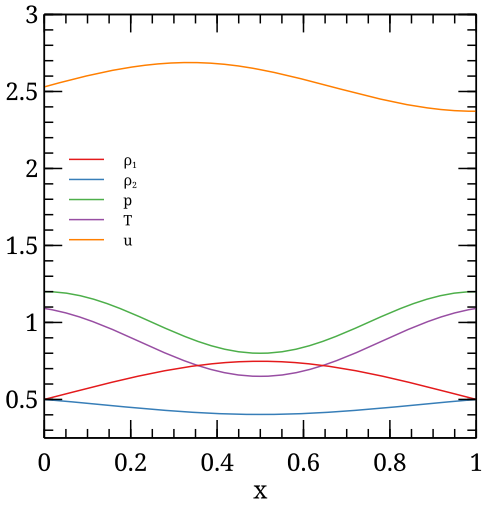


Fig. 1. MMS1D – Dimensionless solution for a $\mathbb{P}^6$ approximation on a mesh consisting of 32 elements in configuration P-G.

shock wave with a circular inhomogeneity [4,9,19]. Finally, we investigate the shock wave interaction with a square inhomogeneity [63], where the interface is sharply initialized by aligning it with the mesh faces. This setup allows us to assess the impact of κ refinement on the ability of the solver to capture and maintain a sharp bubble interface.

To ease the comparisons, we introduce the notation X-Y. Here, X denotes the solver type, *i.e.* based on *entropy projection* adding the DEEB term (P), based on entropy variables using *over-integration* (E) or based on conservative variables (C). Meanwhile, Y represents the numerical flux: Godunov flux (G), Rusanov-type flux (R) or entropy-conserving (C). For the E-Y solvers, the quadrature formulas require a higher degree of exactness, set to $n_{oi} = 4\kappa + 3$ unless stated otherwise.

4.1. Spatial refinement test in one dimension

To assess the spatial order of accuracy of the different formulations, we use the Method of Manufactured Solutions (MMS) from Steinberg and Roache [64]. In MMS a known analytical solution is imposed *a priori*, and a balancing source term is introduced to the governing equations. The solution is designed for a steady-state two-component flow case in one dimension (MMS1D), as an extension of the one proposed by Roy et al. [65]. The variables considered are $\psi \in [Y_1,\ \rho,\ u,\ p]$, expressed in SI units

$$\psi(x) = \mathtt{a}_0^\psi + \mathtt{a}_1^\psi\, \mathtt{f}_1\left(\mathtt{b}_1^\psi\, \frac{x_1}{L}\pi\right), \tag{64}$$

where constants and trigonometric functions are tabulated in Table 1.

The constants $\mathtt{a}^\psi$ share the same dimensions as the corresponding primitive variables, while the $\mathtt{b}^\psi$ coefficients are dimensionless. In our computations, the thermodynamic properties are set to $r_1 = 287.15\ J/(kg\,K)$, $r_2 = 344.58\ J/(kg\,K)$, $\gamma_1 = 1.4$, $\gamma_2 = 1.6$ and the solution is sought in the domain $\Omega = [0, L]$, with $L = 1\ m$, and Dirichlet conditions imposed at the boundaries. It is important to mention that the current solver is not designed for efficient steady-state time integration, and using purely explicit Runge–Kutta schemes is not the best option. However, these schemes are employed here only as a means to validate the implementation.

The solution is made dimensionless with the code reference state ($p_{ref} = \mathtt{a}_0^p$, $\rho_{ref} = \mathtt{a}_0^\rho$, $R_{ref} = r_1$ and $l_{ref} = L$, where $u_{ref} = \sqrt{p_{ref}/\rho_{ref}}$), and it is shown in Fig. 1 left for the seventh-order approximation on a mesh consisting of 32 elements using the P-G formulation. The monitored error is computed as

$$\eta(\circ, \bullet_{ref}) = \left(\Omega_h\right)^{1/2} ||\circ - \bullet_{ref}||_{L_2}, \tag{65}$$

where the obtained numerical solution is denoted by $\circ$, and the reference solution by $\bullet_{ref}$. The reference value is set to the L_2-projection of the exact solution on the dG polynomial space, and η is studied for different numbers of elements N_e.

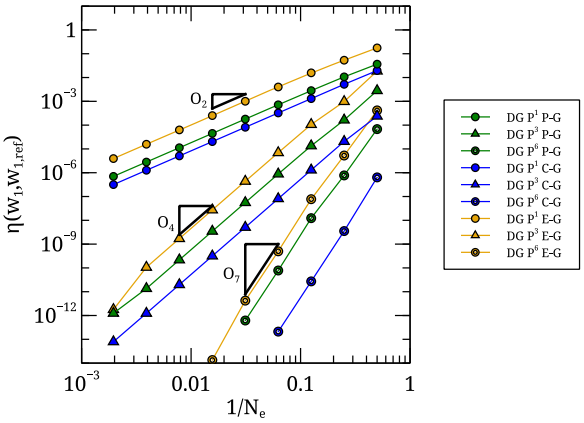


Fig. 2. MMS1D – Spatial refinement tests for the different schemes.

The results of spatial refinement tests converged for all the κ values studied. Particularly, Fig. 2 show the results of these tests for $\kappa = 1, 3, 6$, which are some representative lines of the results obtained selected for the sake of visualization. It is observed that the `E-G` scheme produces a slight over-convergence in the densest meshes for $\kappa = 3, 6$. This behaviour is attributable to the effect of increasing the mesh density in the fulfilment of the integration restrictions for achieving ES, as explained in Section 3.3, resulting in errors similar to the `P-G` scheme. Note that only the results for the entropy-stable flux are presented, as it was not possible to achieve a steady-state solution using the entropy-conserving flux. The reason behind this behaviour is that the entropy-conserving flux produces highly oscillatory solutions [36]. The entropy conservation/stability properties of the schemes will be addressed in Section 4.2.

4.2. Advection of density waves

Spatial accuracy is also assessed by simulating the advection of density waves in uniform velocity and pressure fields for a one-dimensional domain (ADW1D) [66,67]. As a pure advection problem, an exact solution is available for comparison. Let $\Omega = (0, 1]$ and the initial dimensionless primitive fields be defined as

$$\rho_1(x) = \left[\frac{\sin(2\pi x)}{2} + 1\right]\left[\frac{\sin(4\pi x)}{4} + \frac{1}{2}\right], \tag{66}$$

$$\rho_2(x) = -\left[\frac{\sin(2\pi x)}{2} + 1\right]\left[\frac{\sin(4\pi x)}{4} - \frac{1}{2}\right], \tag{67}$$

$$u(x) = 1, \tag{68}$$

$$T(x) = 1, \tag{69}$$

with periodic boundary conditions. The thermodynamic properties are set to $\gamma_1 = 1.6$, $\gamma_2 = 1.4$, $c_{v,1} = 5/3$ and $c_{v,2} = 5/6$. To ensure the time integration error remains below the spatial discretization error, the time step is fixed to $\Delta t = 5.0 \times 10^{-5}$, with an end time $t_{end} = 1$. As shown on the left side of Fig. 3, the theoretical order of convergence $\kappa + 1$ is recovered with the present methods. The convergence reaches a plateau at 10×10^{-9}, which is a phenomenon that was not observed in the previous section. As shown in Fig. 4, the solvers also fail to preserve the pressure equilibrium of the solution, leading to a small but resolution-independent error in the simulations. A similar behaviour was reported by Gassner et al. [68] in a comparable case using a different numerical framework. We conjecture that this phenomenon may be responsible for the stagnation in convergence. Addressing this issue will be the object of further studies in the future.

A comparison between the computational cost of the considered schemes is presented in Fig. 3 (right). The system `C-G` proves to be the most computationally efficient, whereas solving the system `P-G` requires approximately 25%–75% more computational time across the range of spatial meshes and polynomial approximations studied, with a very weak scaling (as expected) with respect to the density of the mesh. The `P-G` solution is far more efficient than the `E-G` solution, which needs approximately between 2 and 7 times more computational time. The cost strongly increases when refining the spatial resolution.

As long as no discontinuities are present, both the direct formulation in entropy variables given in Eq. (59) and the formulation using projected entropy variables in Eq. (60) satisfy Eq. (37) as an equality when an EC flux is used. Therefore, for the entropy conservation study, we replace the Godunov flux with the entropy-conserving flux. To ensure that the integral of physical entropy over the domain is conserved to machine precision, Eq. (32) must hold at the discrete level. For the SSPRK schemes, this condition is satisfied for a sufficiently small time step, which in the present simulations is set to $\Delta t = 1.0 \times 10^{-6}$. We monitor the error in the

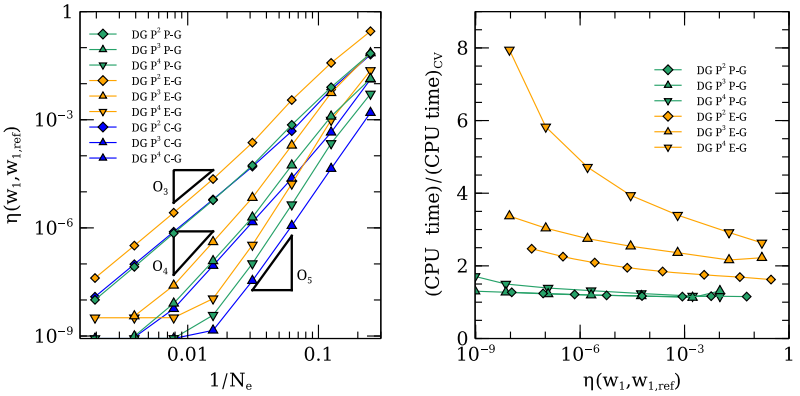


Fig. 3. ADW1D – Results of the spatial refinement test (left). Computational time made dimensionless with the CPU time of the solver in conservative variables for the same mesh and the same polynomial approximation (right).

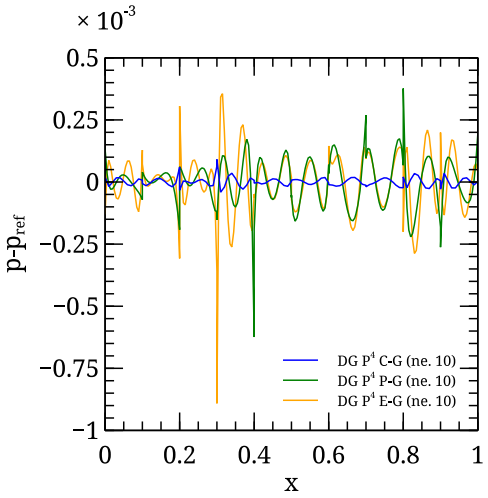


Fig. 4. ADW1D – Error in pressure at $t = 1$ using 10 elements.

conservation of a quantity as

$$\varepsilon(\circ, \bullet_{ref}) = \Omega_h^{-1}\left(\int_{\Omega_h} \circ \, d\Omega - \int_{\Omega_h} \bullet_{ref} \, d\Omega\right). \tag{70}$$

In Figs. 5 and 6, the evolution in time of the error in entropy conservation is displayed for different discrete systems. It is demonstrated that the `E-C` system cannot preserve entropy unless a very high degree of exactness for the quadrature rules is imposed. Entropy conservation for this setup may be possible with $n_{oi} < 50$, but the value is not known *a priori*, and we have included $n_{oi} = 50$ only for illustrative purposes. As expected, the `C-C` framework cannot maintain the global physical entropy. Finally, the `P-C` system preserves entropy without the need of *over-integration*, thanks to the inclusion of the DEEB term . To further study the entropy-conservation/stability properties of the different approaches, we consider the scaling of the entropy conservation error when refining the mesh, using a $\Delta t = 5 \times 10^{-5}$ and $t_{end} = 1$. In Fig. 7, it is shown that entropy-stable schemes (`P-G`) recover the scaling $2\kappa + 1$, as experienced previously by Alberti et al. [35] for the Euler equations. The entropy conservative flux (`P-C`) maintains a minimum error of $\approx 5 \times 10^{-12}$, which is attributable to explicit time integration, as we demonstrated previously, this error reduces when considering a smaller Δt. The simulation corresponding to $\mathbb{P}^2$ and 4 elements crashed at $t = 0.1$, due to the oscillations in the thermodynamic variables induced by the entropy-conserving flux [36]. In this study, it is also demonstrated that $n_{oi} = 4\kappa + 3$ is enough to keep the `E-G` system entropy-stable with the scaling $2\kappa + 1$. When using `E-C` with this degree of exactness, the entropy-conservation property of the scheme is lost for the coarser spatial resolutions, as the scheme fails to preserve entropy and may even lead to the destruction of physical entropy, as shown in the tests marked by some of the red markers (Fig. 8). However, by increasing the level of *over-integration* to $n_{oi} = 50$, the framework recovers its conservation property, as demonstrated by the purple line. The results obtained with `C-G` in Fig. 9 show that, for this simple one-dimensional case, the error in entropy conservation also scales as $2\kappa + 1$. In contrast, the `C-C` strategy lacked sufficient robustness for this analysis, resulting in multiple crashes of runs on coarser meshes.

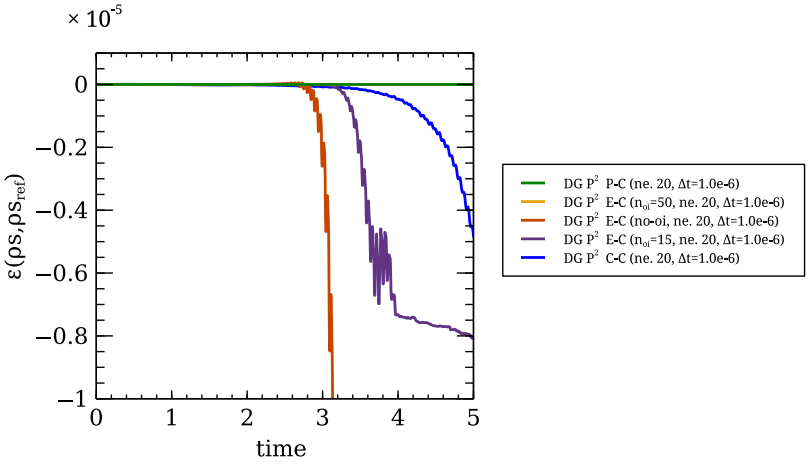


Fig. 5. ADW1D – Entropy conservation study using 20 elements. As *over-integration* for system `E-C`, a degree of exactness n_{oi} is considered. Conversely, "no-oi" indicates no *over-integration* has been taken for the `E-C` system.

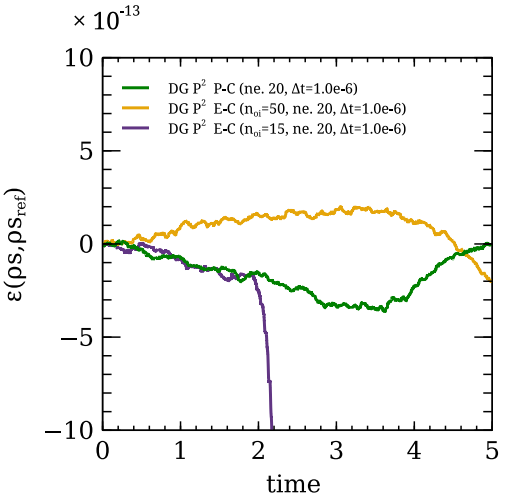


Fig. 6. ADW1D – Entropy conservation study using 20 elements, detail around the x axis.

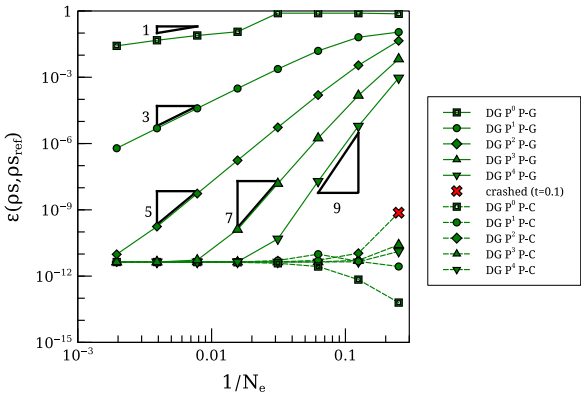


Fig. 7. ADW1D – Spatial refinement test for entropy using `P-Y`. The simulation crashed for $\mathbb{P}^2$ and 4 elements.

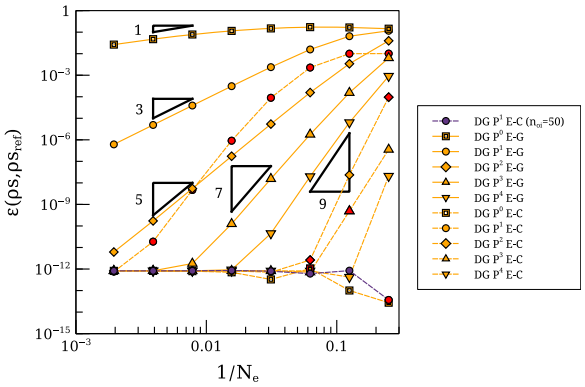


Fig. 8. ADW1D – Spatial refinement test for entropy using `E-Y`. The points • and ◆ indicate negative values, and the absolute value has been shown.

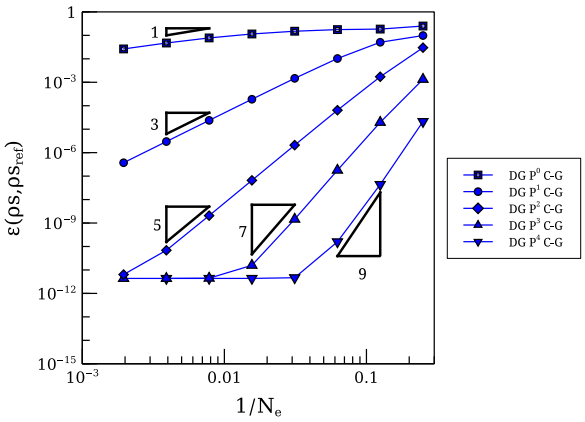


Fig. 9. ADW1D – Spatial refinement test for entropy using `C-G`.

4.3. One-dimensional Riemann problems

The present section is devoted to the performance of different solvers in presence of flow discontinuities. The Riemann problems (RPs), within the domain $\Omega = [0, 1]$ associated with the initial condition

$$\mathbf{w}(x) = \begin{cases} \mathbf{w}_L & \text{if} \quad x < x_0 \\ \mathbf{w}_R & \text{if} \quad x \geq x_0 \end{cases}, \tag{71}$$

are studied, where x_0 is the starting position of the interface. Table 2 details the initial conditions of the problems, the coordinate x_0, the end time t_{end} and the thermodynamic properties. Unless stated otherwise, all solutions are using the shock capturing strategy with $C_{SC} = 0.2$. The problem RP0 is a multicomponent classical Sod problem [19] used here to study the performance of the different numerical fluxes when using the `P-Y` strategy. The problem RP1, in Fig. 12 left, with the initial data from [19], results in the development of a weak shock, a contact wave and a rarefaction wave. The case RP2 is a multicomponent modified Sod problem inspired by [69,70], and the results can be observed in the in Fig. 12 right. In the case RP3, in Fig. 13 left, the accuracy of maintaining a static interface between two regions is tested. The receding flow is denoted as RP4 and represents a multicomponent extension of the problem introduced by Gouasmi et al. [4], illustrated on the right side of Fig. 13. The results of the last two cases, RP5 and RP6, are presented in Fig. 14. These are multicomponent extensions of the setups proposed by Liska and Wendroff [62], designed to evaluate the capabilities of the methods in the presence of very strong shock waves.

All the simulations presented in this section use $\mathbb{P}^3$ polynomial approximations advanced in time using a prescribed CFL number. For consistency and ease of comparison, a common CFL value or CFL evolution strategy is employed across all formulations for a given test case. Most simulations use quite low CFL numbers, as the initial stages of a Riemann problem evolution are typically challenging, especially in complex cases. After this initial phase, the CFL value can often be increased manually; alternatively, an adaptive time-stepping strategy can be employed. The latter is a particularly appealing approach and will certainly be considered for future research, *e.g.*, see Ref. [71]. Only for cases RP5 and RP6 the CFL number is increased empirically and manually during the initial phase. In fact, these problems are extremely challenging because of the very large pressure jump and require very low CFL values for startup.

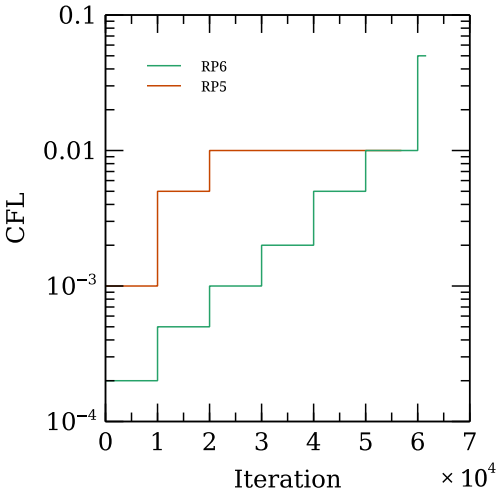


Fig. 10. RP5 & RP6 – Prescribed CFL values during the simulation.

For the RP0 case (CFL = 0.05), we considered the use of the entropy-conservative flux without shock capturing, although this configuration, denoted as `P-C`, is inherently unable to handle the entropy generated at shocks. As shown in Fig. 11, the resulting solution exhibits severe oscillations also interacting with the Dirichlet boundary conditions and reflecting back into the domain, cf. [36]. Interestingly, activating the shock-capturing term significantly "stabilizes" the solution, allowing it to approximate the correct overall behaviour, though distributed small-amplitude high-frequency oscillations remain. The entropy-stable `P-G` method, on the other hand, shows only very limited oscillations near discontinuities and agrees very well with the exact solution. In both RP1 (CFL = 10^{-2}) and RP2 (CFL = 10^{-3}), we appreciate that the positions of the various waves are accurately captured thanks to the conservation properties of all the methods considered, *i.e.*, `P-G`, `E-G`, and `C-G`. These results also demonstrate that the shock-capturing term works effectively for all the formulations examined in this work. In the case of the static contact discontinuity in RP3 (CFL = 0.1), all strategies are shown to exactly capture the discontinuity when the Godunov flux is used. In contrast, as expected, the use of a Rusanov flux, *i.e.*, `P-R`, produces oscillations at the interface. For RP4 (CFL = 10^{-2}), which involves a more complex wave pattern, all formulations using the Godunov flux are shown to be able of reproducing the rarefaction waves that develop, while maintaining sharp the contact wave in the central region. Finally, let us comment on RP5 and RP6, which are among the most challenging test cases computed. These problems involve significant pressure and density discontinuities. RP6 also features a narrow, high-density region; this is why RP6 is commonly referred to as the "peak" case, see Fig. 14. In both RP5 and RP6, all the schemes exhibit oscillatory behaviour in specific regions. Due to the complexity of these cases, simulations required low initial CFL values to ensure robustness. The evolution of the CFL values during the simulation of each of these two cases is reported in Fig. 10.

The hard constraint for RP6 was inherited from the `E-G` formulation and then applied to all the solvers for a fair comparison. It is worth noting that the `P-G` strategy can support larger CFL values. While we observe that the discontinuities of RP5 are sharply captured by all the formulations and the "peak" in density characteristic of RP6 is not smeared by `E-G` and `P-G`, we conjecture that the observed oscillations are a side effect of the conservative properties of the methods, which can induce oscillations at contact discontinuities, as observed by Karni for the multicomponent Euler model in [6]. This hypothesis is supported by the detail of the velocity distribution in RP6, where the region between the shock and contact wave remains free of oscillations. It is important to mention that for the RP6, no CFL number was found that allowed to solve the problem with the `C-G` approach. In contrast, the other formulations produced results in reasonable agreement with the exact solution, likely due to the enhanced robustness conferred by satisfying the implicitly resolved equation for the evolution of entropy. The results from the suite of Riemann problems considered here demonstrate that the entropy-stable `P-G` formulation offers an optimal balance between computational cost and robustness among the strategies studied. Therefore, we will focus exclusively on this solver in the rest of the paper. Finally, we emphasize once more that all simulations confirm the effectiveness of the shock-capturing approach originally introduced by Bassi et al. [57] for single-component flows, which has here been successfully employed for the high-order discretizations of the multicomponent Euler equations.

4.4. Spatial refinement test in two dimensions

To assess the spatial accuracy in multiple dimensions of the numerical framework based on the entropy projection, we again consider the use of the MMS. The solution is still based on the fields proposed by Roy et al. [65], but with the objective of simulating a steady-state two-component flow case (MMS2D). Therefore, now the variables considered are $\psi \in [Y_1,\ \rho,\ u_1,\ u_2,\ p]$, expressed in SI units

$$\psi(x_1, x_2) = \mathtt{a}_0^\psi + \mathtt{a}_1^\psi\, \mathtt{f}_1\left(\mathtt{b}_1^\psi\, \frac{x_1}{L}\pi\right) + \mathtt{a}_2^\psi\, \mathtt{f}_2\left(\mathtt{b}_2^\psi\, \frac{x_2}{L}\pi\right) + \mathtt{a}_{1,2}^\psi\, \mathtt{f}_{1,2}\left(\mathtt{b}_{1,2}^\psi\, \frac{x_1 x_2}{L^2}\pi\right), \tag{72}$$

Table 2
Initial conditions for the Riemann problems, interface position x_0, end time t_{end} and thermodynamic properties.

	$Y_{1,L}$	ρ_L	u_L	p_L	$Y_{1,R}$	ρ_R	u_R	p_R	x_0	t_{end}	γ_1	$c_{v,1}$	γ_2	$c_{v,2}$
RP0	0.5	1.0	0	1.0	0.5	0.125	0	0.1	0.5	0.2	1.5	2.0	1.3	2.0
RP1	0.4	2.0	0	1.0	0.6	1.5	0	2.0	0.5	0.2	1.5	2.0	1.3	2.0
RP2	0.2	1.0	0	1.0	0.8	0.125	0	0.1	0.2	0.2	1.4	2.5	1.6	2.0
RP3	2/3	0.45	0	1.0	3/20	1.15	0	1.0	0.5	1.0	1.4	2.5	1.6	2.0
RP4	0.2	1.0	−0.4	1.3598	0.5	1.0	0.4	2.0	0.5	0.18	1.5	1.6	1.3	2.0
RP5	0.2	5.9992	19.580	460.89	0.8	5.9924	−6.1963	46.095	0.4	0.035	1.4	2.5	1.6	2.0
RP6	0.2	0.1262	8.9905	782.93	0.8	6.5914	2.2654	3.1545	0.5	0.0039	1.4	2.5	1.6	2.0

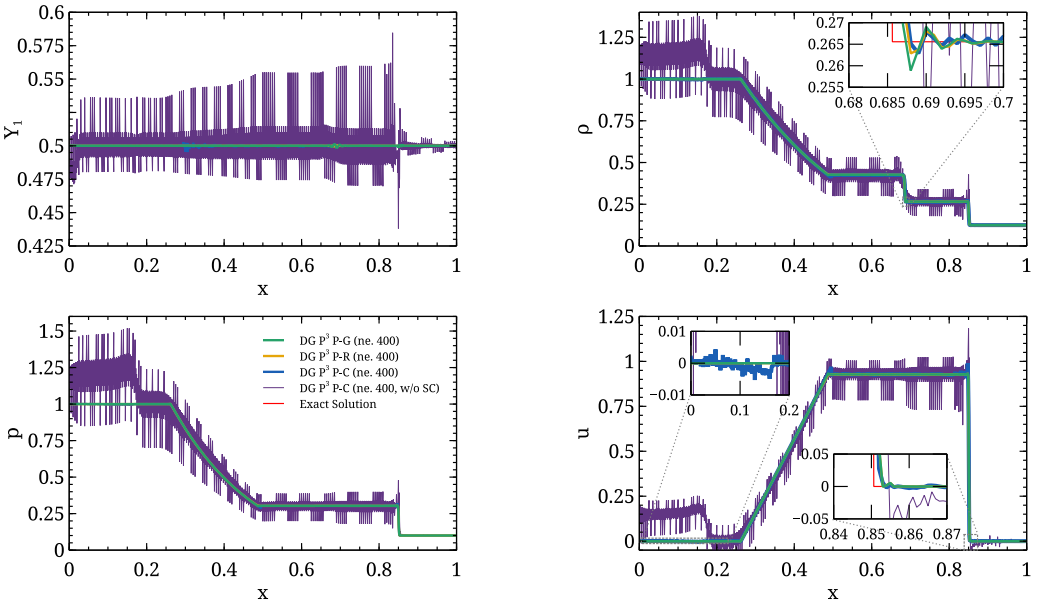


Fig. 11. RP0 – Results for different numerical fluxes using $\mathbb{P}^3$ and 400 elements. The label "w/o SC" indicates that no shock capturing was used for this simulation.

Table 3
MMS2D – spatial refinement test for $d = 2$. Constants and trigonometric functions for the manufactured solution.

ψ	a_0^ψ	a_1^ψ	a_2^ψ	$a_{1,2}^\psi$	f_1^ψ	f_2^ψ	$f_{1,2}^\psi$	b_1^ψ	b_2^ψ	$b_{1,2}^\psi$
Y_1	0.5	0.1	0.15	0	sin	cos	cos	1.0	2.0	0
ρ	1	−0.1	0.15	0	sin	cos	cos	1.0	0.5	0
u	800	50	−30	0	sin	cos	cos	1.5	0.6	0
v	800	−75	40	0	cos	sin	cos	0.5	2/3	0
p	1×10^5	0.2×10^5	0.5×10^5	0	cos	sin	sin	1.5	0.6	0

where constants and trigonometric functions are presented in Table 3.

Again, the thermodynamic properties are set to $r_1 = 287.15\ J/(kg\ K)$, $r_2 = 344.58\ J/(kg\ K)$, $\gamma_1 = 1.4$ and $\gamma_2 = 1.6$. The solution is sought in the domain $\Omega = [L \times L]$ with $L = 1\,m$ and Dirichlet boundary conditions. We insist that the use of explicit Runge–Kutta schemes for searching for a steady-state solution is not optimal; this approach is employed here solely for validation purposes. The domain is discretized by means of Cartesian meshes of $2^i \times 2^i$ elements, where $i = 2, \dots, 5$. Like in MMS1D, the solution is made dimensionless with the code reference state. Contours are shown in Fig. 15 for the fifth-order approximation for the $i = 4$ mesh.

The spatial refinement test for this case, using Godunov's flux (`P-G`), is shown in Fig. 16 for different polynomial approximations, demonstrating that the theoretical convergence order of $\kappa + 1$ is achieved. Note that, as for the MMS1D case, only the results for the entropy-stable flux are presented, as it was not possible to achieve a steady-state solution with the entropy-conserving flux.

4.5. The implosion case

To assess the capability of the solver to capture discontinuities and small-scale flow features when increasing the polynomial degree, a multicomponent extension of the implosion two-dimensional Riemann problem presented in [62,72] (MIM) was simulated.

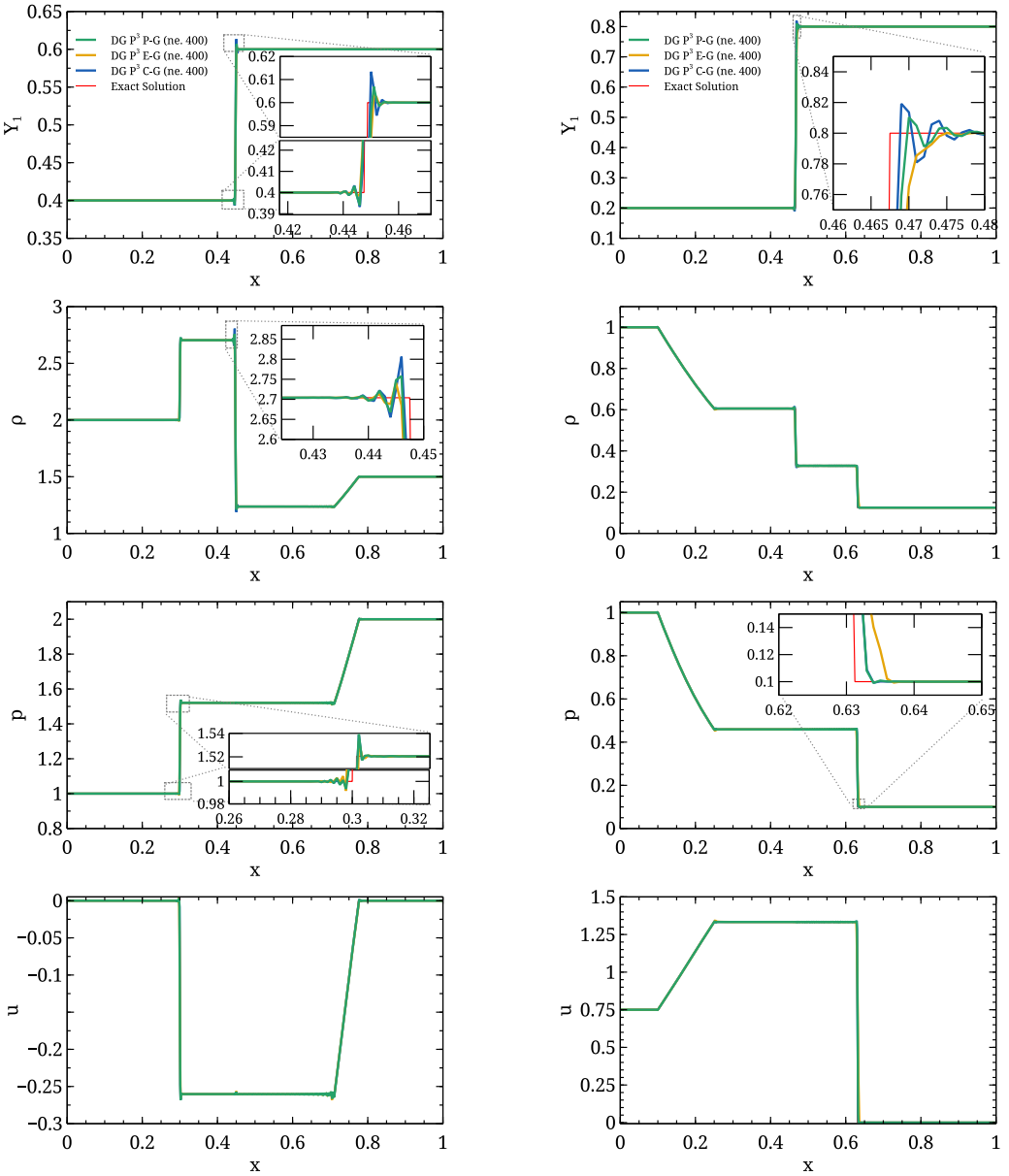


Fig. 12. RP1 – Results (left); RP2 – Results (right). $\mathbb{P}^3$ approximations on 400 elements.

The original converging shock problem considers a gas confined within a square domain. A smaller, centrally located square region, rotated by $\pi/4$, contains gas with initially lower density and pressure than the surrounding. To account for multicomponent effects, we introduce a third region in the top-right of the domain with a different mass fraction to investigate the interaction between the shock and the contact discontinuity. The computation is performed for the only upper right quadrant, within a square domain of side length 0.3, *i.e.* $\Omega \in [0.3 \times 0.3]$. Symmetry conditions are imposed on all the boundaries of the computational domain and the initial condition is defined as

$$[\rho_1, \rho_2, u_1, u_2, p] = \begin{cases} [0.1, 0.025, 0, 0, 0.1] & \text{if} \quad 0 \leq x_1 + x_1 \leq 0.15 \\ [0.2, 0.8, 0, 0, 1] & \text{if} \quad 0.15 < x_1 \text{ and } 0.15 < x_2\,, \\ [0.8, 0.2, 0, 0, 1] & \text{otherwise} \end{cases} \tag{73}$$

with the components having the following thermodynamic properties $\gamma_1 = 1.5$, $\gamma_2 = 1.3$ and $c_{v,1} = c_{v,2} = 2$. A sketch of the configuration is shown in Fig. 17. The mesh used for this κ-refinement study consists of 10 050 elements arranged as in Fig. 18, where triangular elements are only placed to sharply capture the initial condition. Due to the complex interaction of strong waves, the simulation is initiated with a low CFL number of 0.02 up to $t = 0.1$. After this point, the CFL value is increased to 0.1 and kept fixed until the

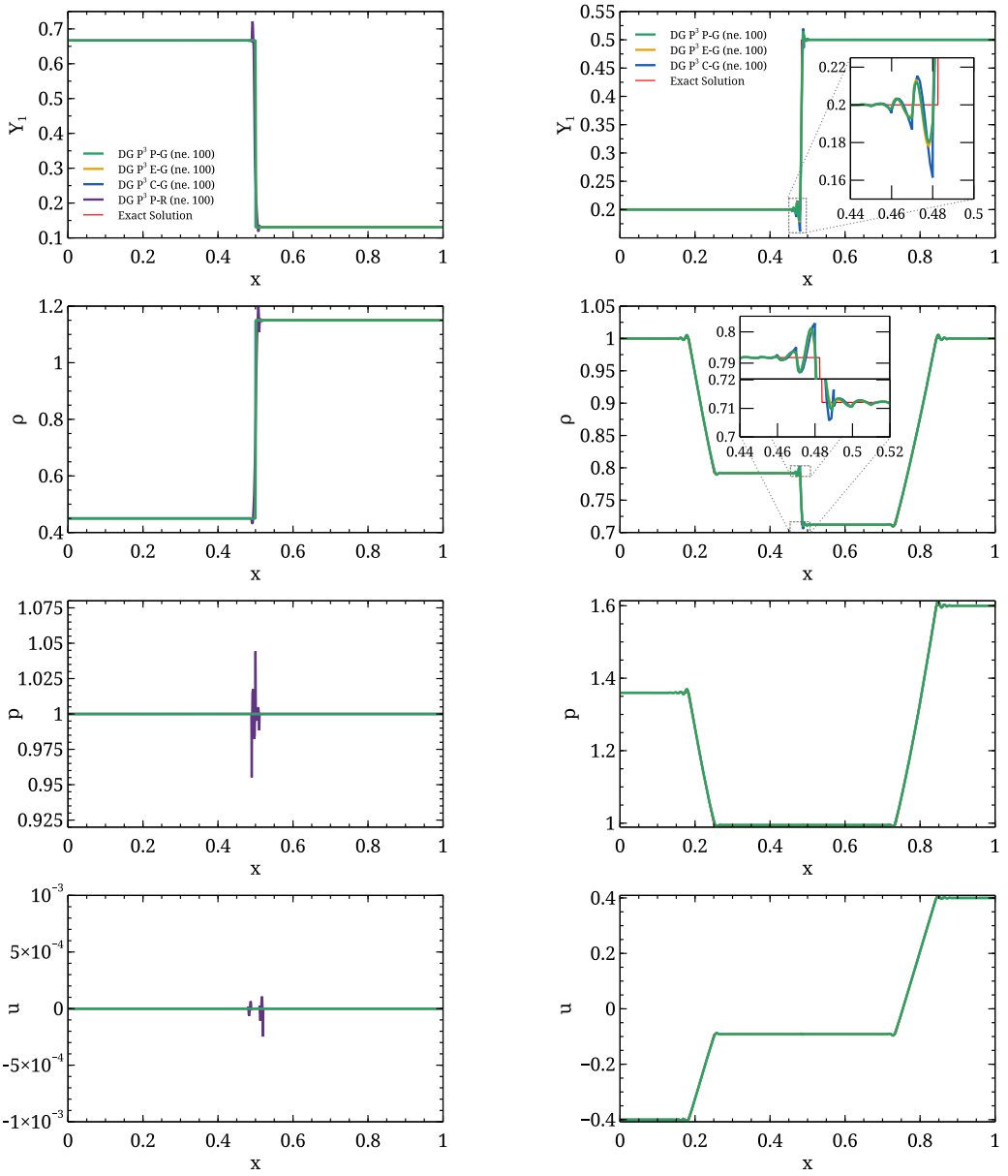


Fig. 13. RP3 – Results (left); RP4 – Results (right). $\mathbb{P}^3$ approximations on 400 elements.

final simulation time. On the left of Fig. 19a, it can be observed that the scheme is able to keep Y_1 constant at the bottom-left region besides the shock and the rarefaction waves. On the right of Fig. 19b, the contact wave separating the low and high-density regions is shown. In Fig. 19b (left), it can be observed that the rarefaction wave has started interacting with the top-right contact discontinuity. The presented discretization is capable of maintaining the interface sharp at this stage. In Fig. 19b (right), the interaction between the shock waves is shown, exhibiting a similar behaviour to the original single-component problem presented in [62]. The left side of Fig. 19c demonstrates that the successive impact of reflected shock waves has finally deformed the interface. Furthermore, on the right side of Fig. 19c, it is observed that the vortex at the bottom-left of the domain is deviated by the presence of the interface upstream. In general, the results for global density and pressure are similar to those presented in [62] for the single-component case. As in the original inspiration for this case, Fig. 20 shows that the most noticeable change due to the refinement of the spatial discretization is in the resolution of the vortex, which exhibits more scales as the polynomial degree increases. This is expected, as scale-resolving capabilities are a key feature of high-order methods, such as the dG method.

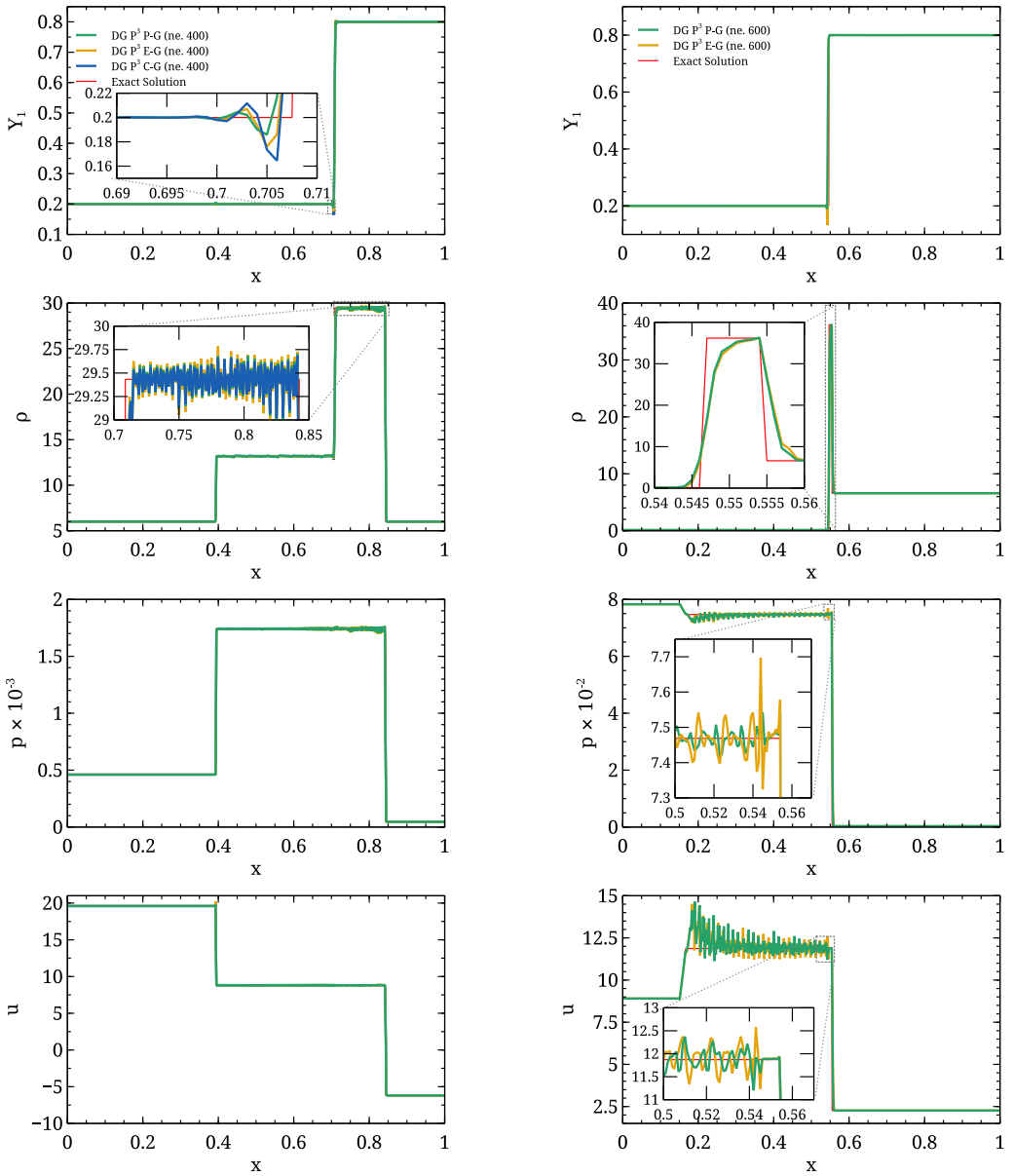


Fig. 14. RP5 – Results using $\mathbb{P}^3$ and 400 elements (left); RP6 – Results using $\mathbb{P}^3$ and 600 elements (right).

4.6. *The shock wave-helium bubble interaction problem*

Let us now consider the interaction of a shock wave with a quiescent cylindrical helium inhomogeneity (SBI) that is in mechanical equilibrium with the surrounding air. This case has been extensively studied experimentally [1], and numerically [4,9,19,73,74] to evaluate the resolution capabilities of the schemes when applied to multicomponent mixtures. The flow problem has been made dimensionless by the quantities indicated in Table 4.

To leverage the symmetry of the case, we define a rectangular domain $\Omega = [23.57 \times 0.89]$ with extrapolated boundary conditions in the x_1 limits and symmetry boundary conditions in the x_2 boundaries, as depicted in Fig. 21. The domain is filled with air ($r_1 = 1$, $\gamma_1 = 1.4$) contaminated with 2.45% in mass of helium ($r_2 = 5.498$, $\gamma_2 = 1.647$). A semicircular inhomogeneity with radius $R_b = 0.5$ is centred at $x_b = [11.8, 0]$, and contains helium contaminated with an 11.89% in mass of air. A Mach $M_S = 1.22$ shock wave is located at $x_s = 12.8$. This setup defines three distinct regions in terms of primitive variables, as outlined in Table 5.

To prevent the initial condition from introducing sharp contact discontinuities within the cells, which could generate spurious oscillations due to the high-order nature of the solution, the bubble-air interface is regularized by applying the method proposed by

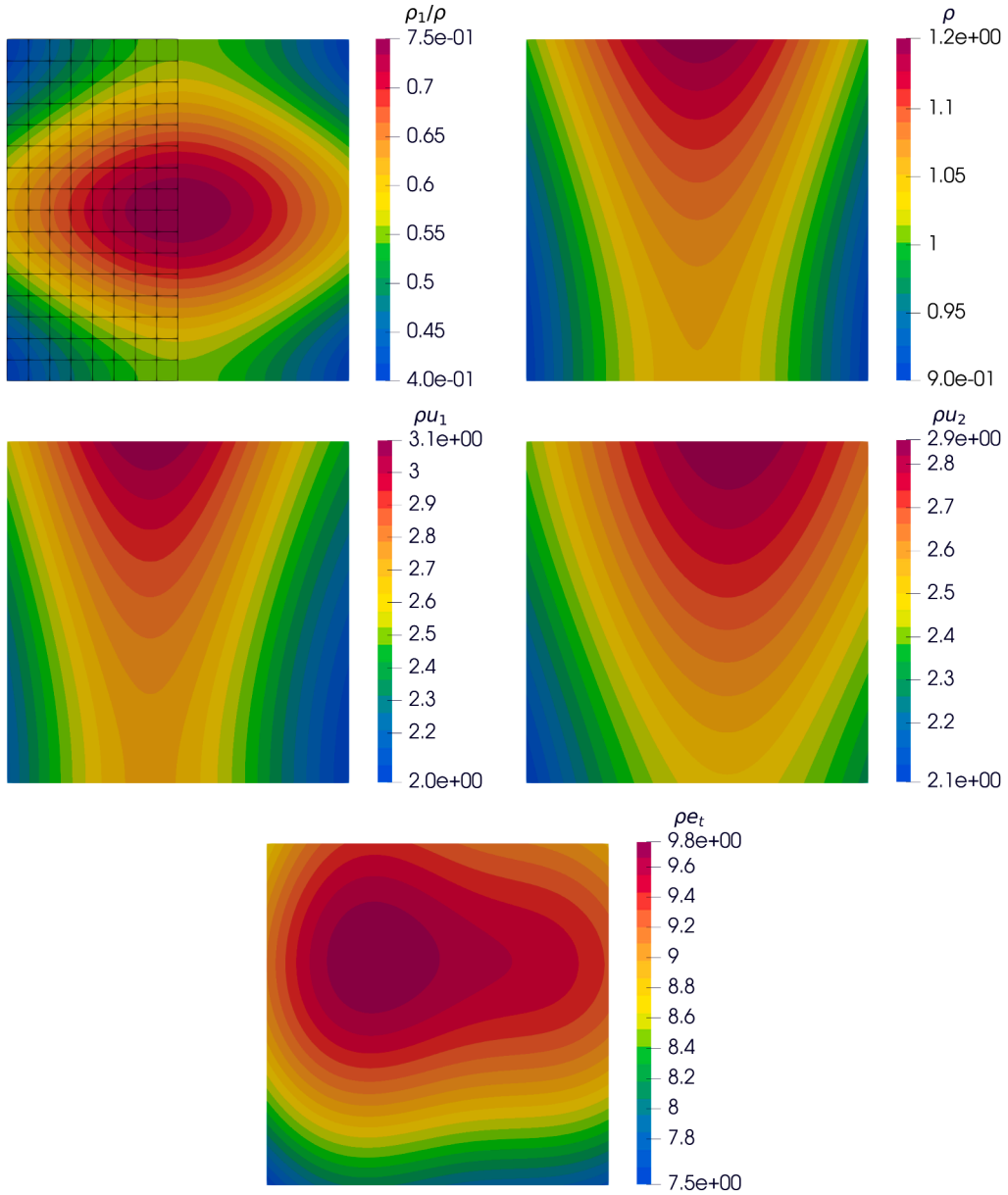


Fig. 15. MMS2D – Contours of the dimensionless flow variables for a $\mathbb{P}^4$ approximation on a mesh consisting of 16×16 elements, half superimposed to ρ_1/ρ.

Kawai and Terashima [74] to the primitive variables

$$\mathbf{w} = \mathbf{w}_b\left(1 - f_{sm}\right) + \mathbf{w}_s f_{sm}, \tag{74}$$

where $\mathbf{w}_b$ and $\mathbf{w}_s$ are the primitive states in the bubble and the surrounding medium, respectively, and

$$f_{sm} = \left[1 + \mathrm{erf}(\Delta R/\epsilon)\right]/2 \qquad \text{and} \qquad \epsilon = C_\epsilon \Delta x \times 10^{-4}. \tag{75}$$

Here, ΔR is the distance to the bubble interface, with the constants set as $\Delta x = 5$ and $C_\epsilon = 5$. The computational mesh consists of quadrilateral elements. To mitigate spurious reflections at the boundaries, following the approach in [75], the domain is discretized into three regions with varying mesh densities. These regions are separated by the vertical lines at $x_1 = 7.2$ and $x_1 = 16.2$. Specifically:

- the first region is defined by $x_1 < 7.2$ and is discretized by using 150×45 elements;
- the second region is defined by $7.2 < x_1 < 16.2$ and is discretized by using 450×45 elements;
- the third region is defined by $x_1 > 16.2$ and is discretized by using 150×45 elements.

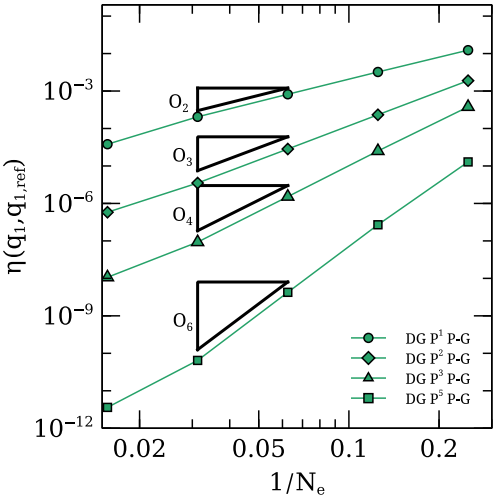


Fig. 16. MMS2D – Spatial refinement test for $d = 2$. L_2 norm of the error in ρ_1, and N_e is the number of elements per direction.

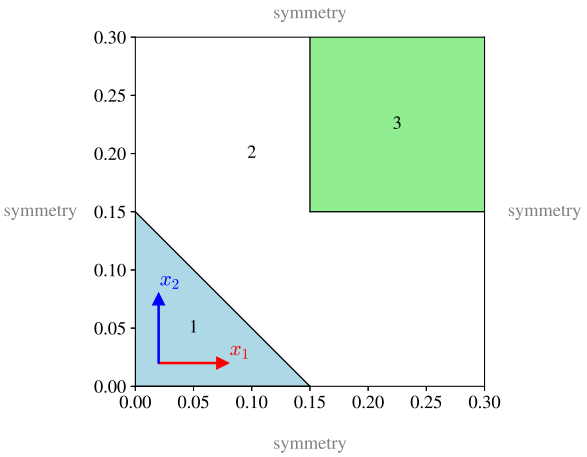


Fig. 17. MIM – Initial condition. The region 1 is a low-density and low-pressure region, while a contact discontinuity separates region 2 from region 3.

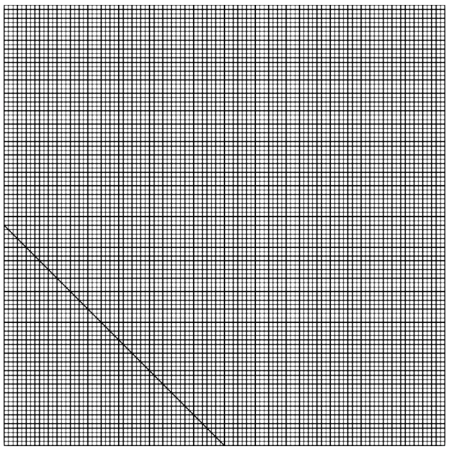
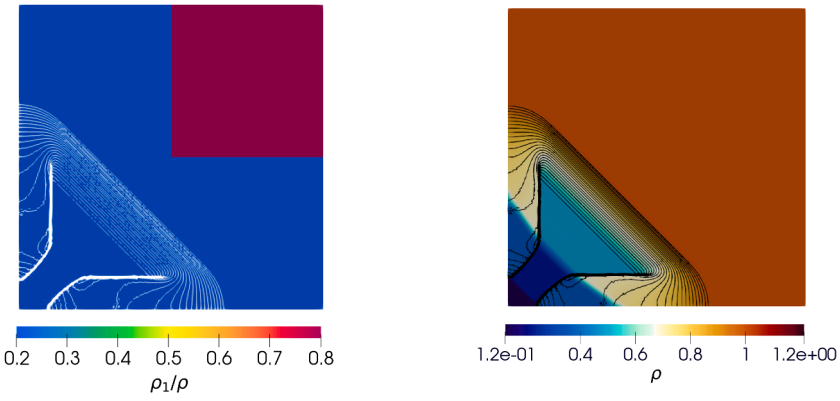
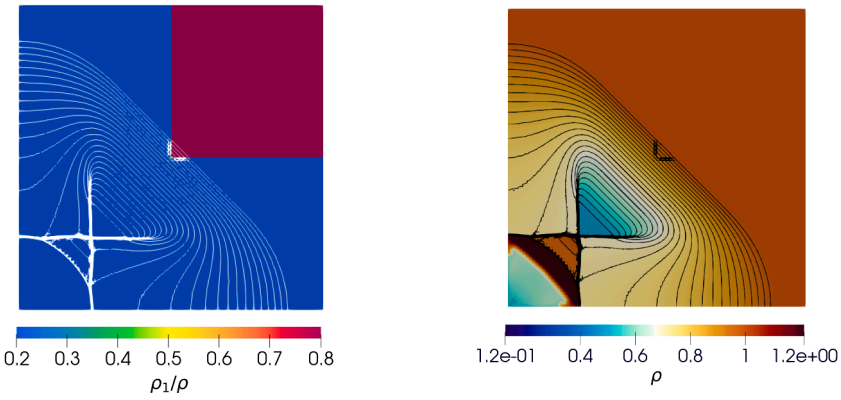


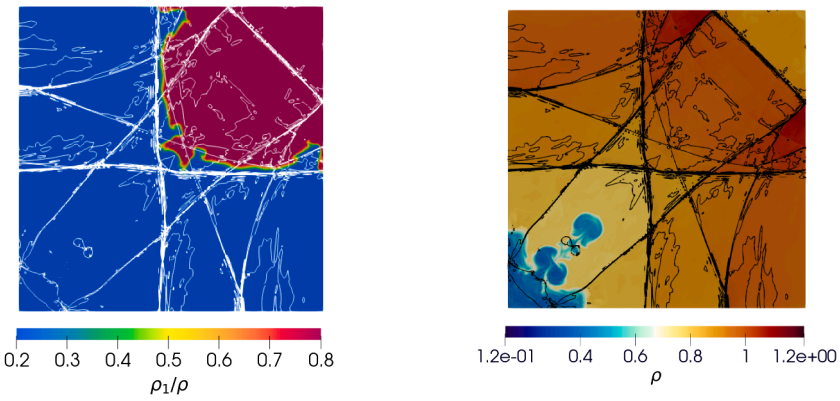
Fig. 18. MIM – Computational mesh made of 10 050 cells.



(a) $t = 0.045$



(b) $t = 0.1$



(c) $t = 2.5$

Fig. 19. MIM – Results for different dimensionless time levels. $\mathbb{P}^4$ solution on a mesh made of 10 050 cells. Thirty pressure isocontours are superimposed for reference.

Table 4
SBI – Reference values for dimensionless quantities.

Quantity	Symbol	Value
diameter of the bubble	D_b^*	$50\,mm$
ideal gas constant of air	r_1^*	$0.287\,J\,kg^{-1}\,K^{-1}$
pre-shock density	ρ^*	$1.225\,kg\,m^{-3}$
pre-shock pressure	p^*	$101\,325\,Pa$

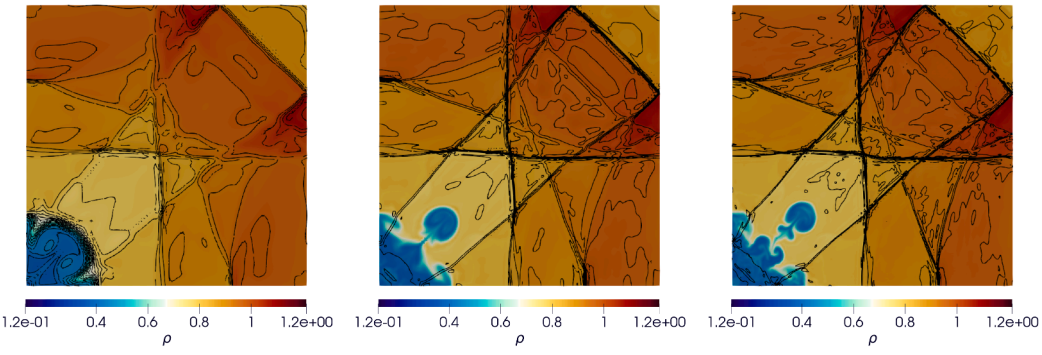


Fig. 20. MIM – Results at final time for different polynomial approximations. The colour map represents the mixture density, with 30 pressure isocontours superimposed.

Table 5
SBI – Dimensionless state vectors for the different regions in the initial condition.

Region	State $[\rho_1, \rho_2, u_1, u_2, p]$
pre-shock air	[0.9976, 0.0245, 0, 0, 1]
bubble	[0.0245, 0.1819, 0, 0, 1]
post-shock air	[0.9976, 0.0245, −0.3947, 0, 1.5699]

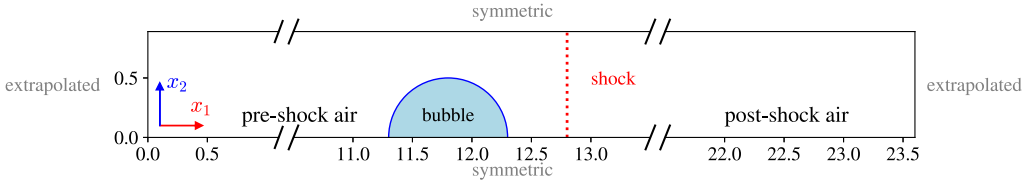


Fig. 21. SBI – Scaled schematic of the initial and boundary conditions.

All the simulations are performed using the fixed CFL value of 0.065. This relatively low value is dictated by the robustness of the highest-degree, *e.g.*, $\mathbb{P}^4$, especially considering that the initial phase of the shock-bubble interaction can induce significant oscillations in the flow variables. For visualization purposes, we obtain Schlieren-like images by plotting $\exp\left(||\nabla\rho||/||\nabla\rho||_{\max}\right)$ following [9]. Results are presented for $\mathbb{P}^2$ and $\mathbb{P}^4$ approximations in Figs. 22 and 23, respectively. The simulation using $\mathbb{P}^2$ contains 202 500 DOFs per equation, while the simulation using $\mathbb{P}^4$ contains 506 250 DOFs per equation. For $t = 32\,\mu s$ it can be observed that the transmitted (left) and reflected (right) waves are properly captured by the method. At $t = 62\,\mu s$, the transmitted wave abandons the bubble, and if compared with $t = 102\,\mu s$, it can be seen that, as in the experimental results from [1], the left front of the bubble has not been affected by this transmission. Besides, the secondary reflected wave, provoked by the passage of the primary transmitted wave, is not captured as in [1,4]. Since this wave is very weak, a finer discretization would likely allow for its detection. The quadruple shock intersection at the top of the picture is well resolved. It can also be observed that the interior Richmyer-Meshkov instability has already started to form at $t = 240\,\mu s$, evolving and causing the bubble to nearly split ($t = 427\mu s$ to $t = 674\,\mu s$). At the end time of the simulation, the differences between the lower order and higher order solutions is clearly appreciated, especially in the instabilities formed by the entering jet at the top and bottom of the bubble. These instabilities develop two vortices for the $\mathbb{P}^4$ solution, but only one in the $\mathbb{P}^2$ solution. The results are, in general, in reasonable qualitative agreement with the experimental results [1] and with numerical simulations [4,19].

4.7. The shock wave-square helium bubble interaction problem

The interaction of a $M_S = 1.22$ shock wave with a square helium inhomogeneity surrounded by nitrogen is studied in this section (SQBI). The present case is reminiscent of the setup proposed by Singh and Torrilhon [63], with significant differences that will be highlighted. The aim of this setup is to perform a proof-of-concept of solving a multicomponent case where is not needed to regularize the initial condition at the interface. Symmetry boundary conditions are applied at the top and bottom of the domain, while extrapolated conditions at the left and right. To minimize spurious reflections in the extrapolated boundaries, the domain in enlarged in the x_1 direction and coarser meshes are used at left and right. Consequently, taking as reference the side of the bubble, the simulations are conducted in a dimensionless domain $[29 \times 1.125]$, where three distinct regions are defined based on the

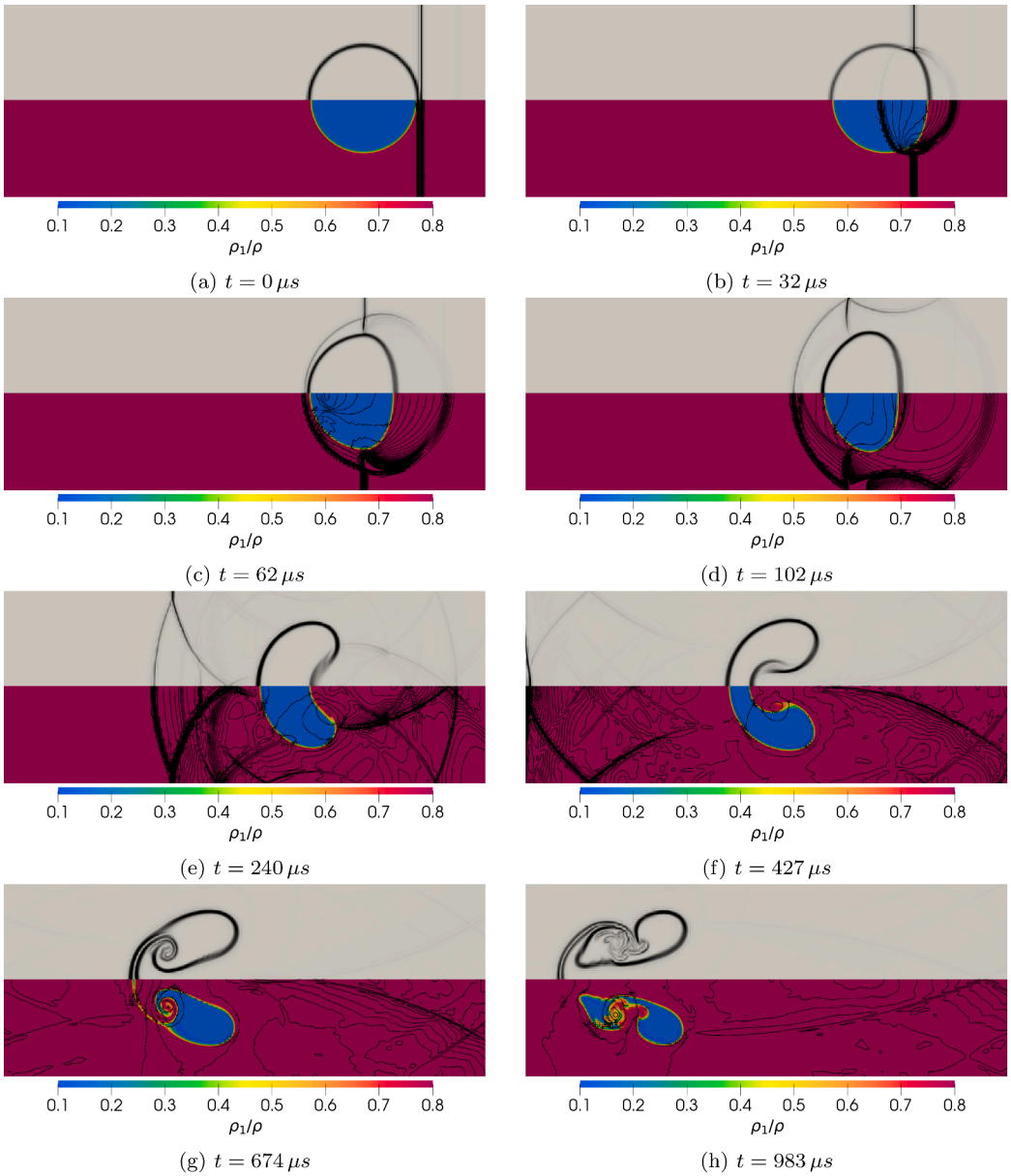


Fig. 22. SBI – Results for different time levels. The computational mesh consists of 579×45 elements, and a polynomial approximation $\mathbb{P}^2$ has been used. Pseudo-Shlieren contours (top-half) and mass fraction of air with 70 pressure isocontours (bottom-half).

initial conditions, as shown in Fig. 24. The surrounding gas and the bubble are initialized at rest and in mechanical equilibrium at 101 300 Pa. The densities are set to $\rho_{bubble} = 0.16\,kg/m^3$ and $\rho_{pre-shock} = 1.25\,kg/m^3$, and the post-shock conditions are initialized using Rankine-Hugoniot conditions as in [63]. Finally, the resulting condition is contaminated, imposing a small mass fraction $Y_{con} = 0.01$ of the non-dominant component in each region to avoid zero or negative partial densities during computations. The case is made dimensionless by the values reported in Table 7, and the resulting configuration is summarized in Table 6. For the equation of state, the dimensionless constants $\gamma_{He} = 1.667$, $\gamma_{N_2} = 1.4$, $c_{v,He} = 1.4993$ and $c_{v,N_2} = 0.32$ are considered.

As in the previous case, the computational mesh is composed of quadrilateral elements, and distributed as follows

- the first region is defined by $x_1 < 12$, and is discretized by using 40×45 elements;
- the second region is defined by $12 < x_1 < 17$, and is discretized by using 200×45 elements;
- the third region is defined by $x_1 > 17$, and is discretized by using 40×45 elements.

We define the characteristic time t_0 as the time required for the shock to reach the bubble interface. Consequently, the time frames of the results are expressed in terms of the dimensionless time $\tau = t/t_0$. As for the implosion case, cf. Section 4.5, for the present complex

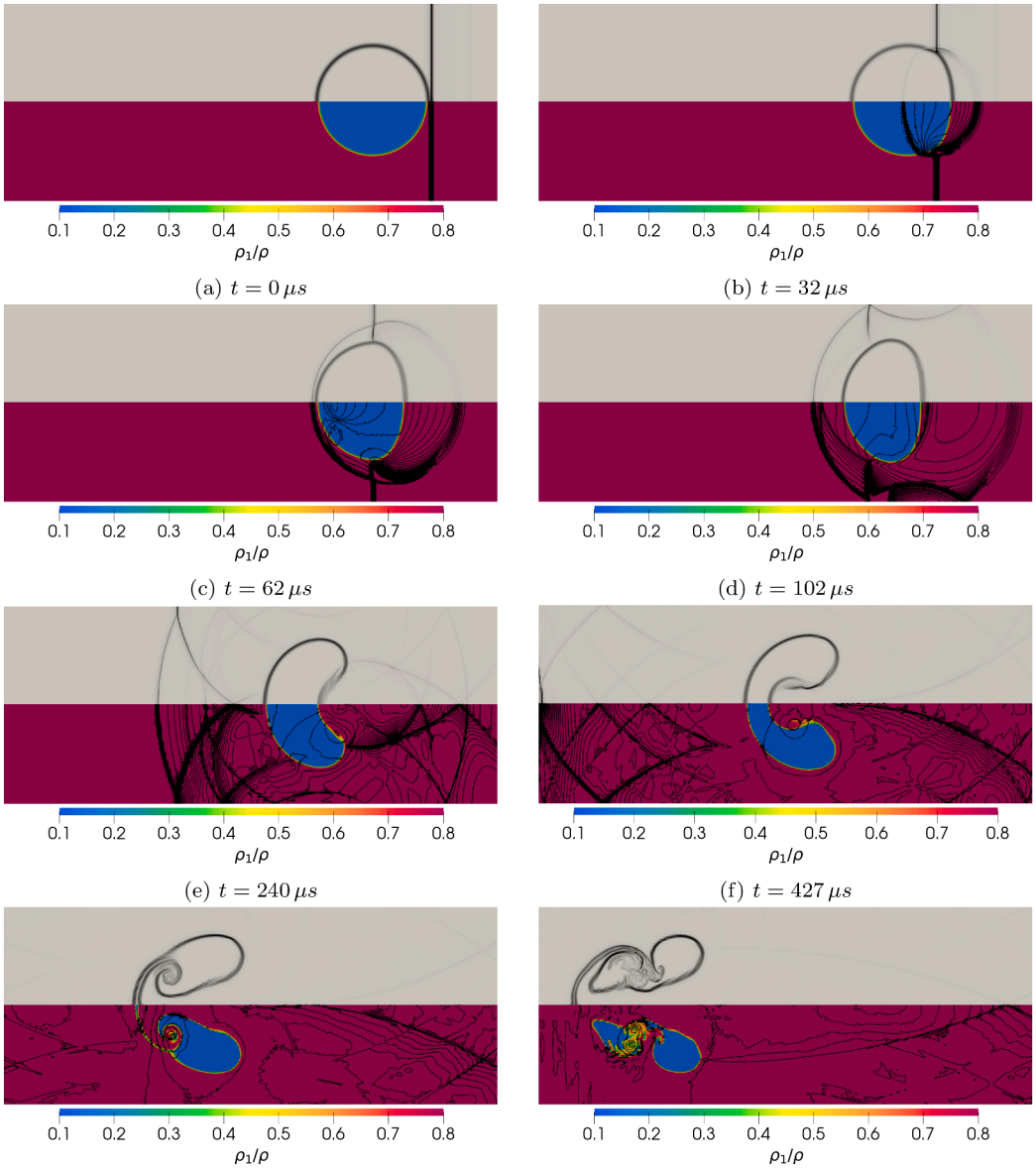


Fig. 23. SBI – Results for different time levels. The computational mesh consists of 579×45 rectangular elements, and a polynomial approximation $\mathbb{P}^4$ has been used. Pseudo-Shlieren contours (top-half) and mass fraction of air with 70 pressure isocontours (bottom-half).

Table 6
SQBI – Dimensionless state vectors for the different regions in the initial condition.

Region	State $[\rho_{He}, \rho_{N_2}, u_1, u_2, p]$
pre-shock nitrogen	$[0.01, 0.99, 0, 0, 1]$
helium bubble	$[0.1267, 1.3 \times 10^{-3}, 0, 0, 1]$
post-shock nitrogen	$[0.0136, 1.3454, 0, 0, 1.5415]$

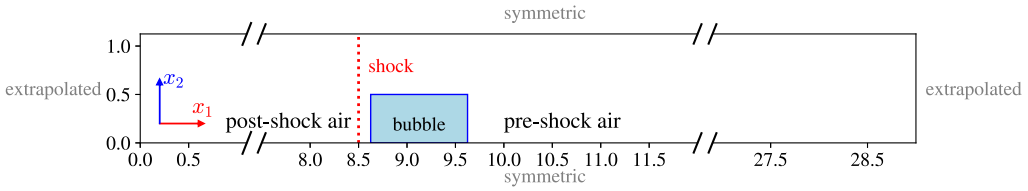


Fig. 24. SQBI – Scaled schematic of the initial and boundary conditions.

Table 7
SQBI – Reference values for dimensionless quantities.

Reference	Symbol	Value
side of the bubble	L_a^*	$40\,mm$
ideal gas constant of helium	r_1^*	$2.1614\,J\,kg^{-1}\,K^{-1}$
pre-shock density	ρ^*	$1.25\,kg\,m^{-3}$
pre-shock pressure	p^*	$101\,325\,Pa$

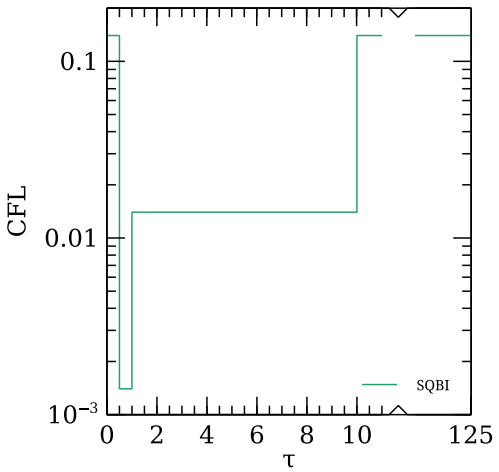
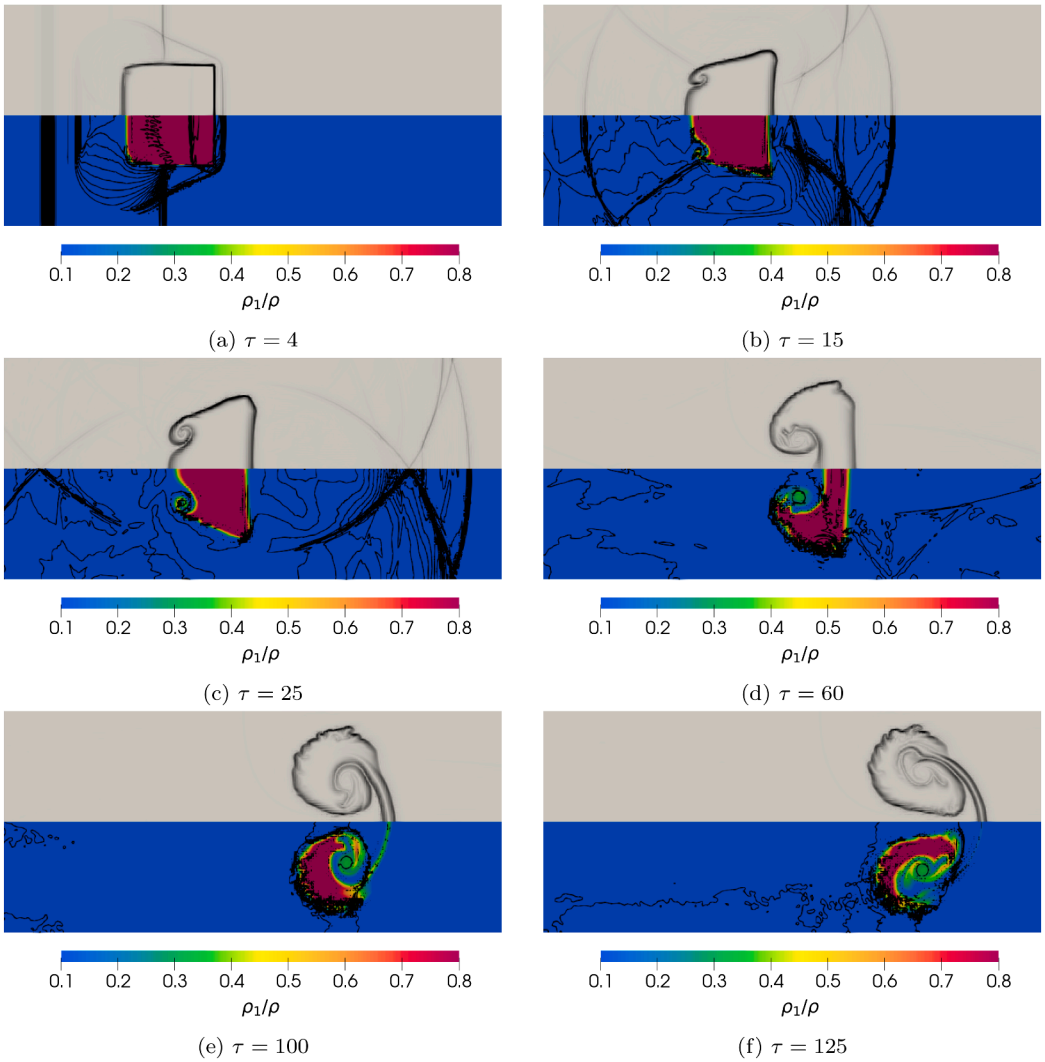


Fig. 25. SQBI – Prescribed CFL number during the simulation.

flow problem, the CFL number was manually adjusted during the simulation to enhance robustness, especially in the initial phase when the shock interacts with the bubble. A single strategy, derived from the highest-resolution simulation, *i.e.*, $\mathbb{P}^4$, was applied uniformly across all runs to favour a direct comparison. The evolution of the imposed CFL value during all simulations is shown in Fig. 25. Although this approach is not the most efficient, it is considered adequate for the scope of the present work. Alternative time-stepping strategies, including adaptive methods based on embedded explicit Runge–Kutta schemes, will be investigated in future work.

The flow evolution is shown in Fig. 26. For $\tau = 4$, it is shown that the second transmitted shock is already created at the right of the bubble, and followed by the free precursor shock waves. The instabilities that will lead to the primary vortex rings already started to form at the left of the bubble. At $\tau = 15$ the diffracted shock is well captured by the method, forming an intersection with the precursor wave that has been reflected by the symmetry boundary condition. At $\tau = 25$ the waves are travelling away from the bubble. At $\tau = 60$, the primary vortex rings begin to form the entering jet, with the Kelvin-Helmholtz instability captured by the diffuse interface and developing along the upper and lower sides of the bubble. Between $\tau = 100$ and $\tau = 125$, the bubble undergoes its final stages of evolution, characterized by the emergence of the connecting bridge, while its upper and lower halves are nearly separated.

In Fig. 27, the final shape of the bubble is compared for $\kappa = 1, \dots, 3$, corresponding to 37 800, 75 600 and 126 000 DOFs per equation, respectively. Unlike the SBI case, the interface is significantly sharper as κ is refined, due to the fact that regularization is no longer required in the initial condition. This allows the method to effectively capture instabilities at the bubble interface, even with the coarse spatial resolution employed.

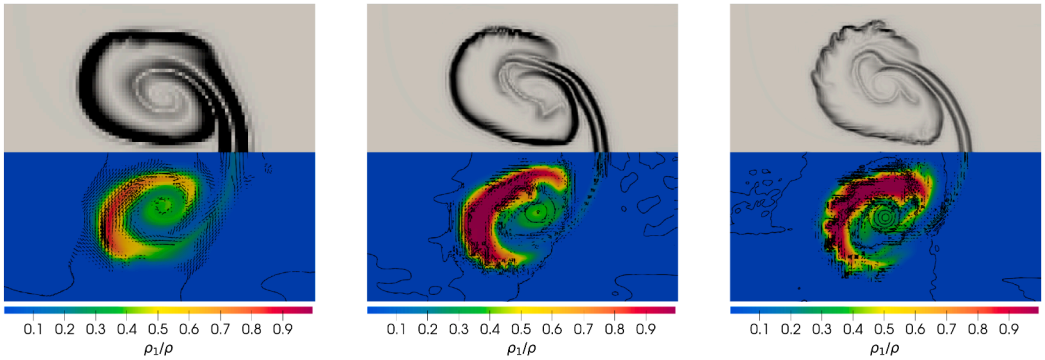


(a) $\tau = 4$ (b) $\tau = 15$

(c) $\tau = 25$ (d) $\tau = 60$

(e) $\tau = 100$ (f) $\tau = 125$

Fig. 26. SQBI – Results for different time levels. The computational mesh consists of 280×45 elements, and a polynomial approximation $\mathbb{P}^3$ is used. Pseudo-Shlieren contours (top-half) and mass fraction of helium with 70 pressure isocontours (bottom half).



(a) $\mathbb{P}^1$ (b) $\mathbb{P}^2$ (c) $\mathbb{P}^3$

Fig. 27. SQBI – Detail at $\tau = 125$ for different polynomial approximations on a mesh made of 280×45 elements. Pseudo-Shlieren contours (top-half) and mass fraction of helium with 70 pressure isocontours (bottom-half).

5. Conclusions

We have proposed an efficient and highly accurate modal dG method for the numerical solution of the multicomponent Euler equations. The strategy evolves the conservative variables while assembling the residual of the spatial discretization from the L_2-projection of the entropy variables onto the dG space [44]. This approach eliminates the need for costly matrix inversions for each element, which are typically required in modal dG methods when solving for the entropy variables. To guarantee entropy conservation and stability while avoiding the computational overhead of *over-integration*, we incorporate the DEEB correction term [11] into the discretization. As a result, the proposed method provides an affordable, semi-discrete entropy-conserving/stable strategy for solving inviscid multicomponent flows. To evaluate the accuracy of the method, we assessed the convergence to the analytical solution using the Method of Manufactured Solutions in both one and two dimensions. Additionally, we demonstrated that the framework is entropy-conserving by design in the absence of shocks, achieving a 30%-78% reduction in CPU time compared to directly solving for the entropy variables using *over-integration*. The robustness of the proposed strategy has been assessed through a series of one-dimensional Riemann problems, including multicomponent extensions of challenging Euler test cases. The results also demonstrate that the employed shock-capturing strategy effectively and selectively damps spurious oscillations in the presence of shock waves. The accuracy and scale-resolving capability of the solver in multiple dimensions are demonstrated through the successful simulation of problems featuring non-trivial flow phenomena: i) a multicomponent implosion, where geometrically complex interacting waves are sharply resolved; ii) the interaction of a shock wave with a light circular inhomogeneity, where the intricate deformation of the interface at later stages is well predicted; iii) the interaction of a shock wave with a square inhomogeneity, where the method captures the generation and evolution of small-scale instabilities at the interface.

Future work will focus on extending the multicomponent model to incorporate the stiffened gas equation of state, with the goal of simulating multiphase flows, which present additional challenges for the numerical scheme. To enhance the robustness of the approach, we will explore conservative, high-order strategies to keep the pressure oscillations under control, together with adaptive time-stepping methods that provide stability to the time integration. Finally, we will consider the implementation of the method for hybrid CPU-GPU hardware to leverage the new-generation High-Performance Computing platforms, taking full advantage of the reduced computational complexity of the proposed framework.

CRediT authorship contribution statement

D. Regener Roig: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Resources, Methodology, Investigation, Formal analysis, Data curation, Conceptualization; **A. Crivellini:** Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization; **A. Colombo:** Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization.

Data availability

Data will be made available on request.

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.

Acknowledgments

This research is in **partnership** with the International Research Training Group (IRTG): DROPIT (Droplet Interaction Technologies GRK 2160/2).

Appendix A. Useful matrices and relations

This section presents the various analytical gradients and Jacobians required in developing the semi-discrete models. To maintain clarity, the formulations are provided in three dimensions, with the understanding that they can be easily reduced to fewer dimensions as needed. Additionally, the operators are expressed in terms of the primitive variables to ensure more concise and tractable expressions.

A.1. Change of variables Jacobian

The change of variables Jacobian $\partial\mathbf{q}/\partial\mathbf{v}$ is needed in Eq. (49) for computing the spatial gradient of the conservative variables when entropy variables are used. It is symmetric and reads

$$\frac{\partial \mathbf{q}}{\partial \mathbf{v}} = \begin{bmatrix} \frac{\rho_1}{r_1} & \dots & 0 & \frac{\rho_1}{r_1}u_1 & \frac{\rho_1}{r_1}u_2 & \frac{\rho_1}{r_1}u_3 & \frac{\rho_1}{r_1}e_1^t \\ & \ddots & & \vdots & \vdots & \vdots & \vdots \\ & & \frac{\rho_N}{r_N} & \frac{\rho_N}{r_N}u_1 & \frac{\rho_N}{r_N}u_2 & \frac{\rho_N}{r_N}u_3 & \frac{\rho_N}{r_N}e_N^t \\ & & & \Psi(u_1) & u_1u_2\,\mathtt{d} & u_1u_3\,\mathtt{d} & u_1\,\kappa \\ & & \text{Sym.} & & \Psi(u_2) & u_2u_3\,\mathtt{d} & u_2\,\kappa \\ & & & & & \Psi(u_3) & u_3\,\kappa \\ & & & & & & \mathtt{c}+T\rho\left(\|\mathbf{u}\|^2+Tc_v\right) \end{bmatrix}, \tag{A.1}$$

where $e_k^t = e_k + \frac{1}{2}\|\mathbf{u}\|^2$ and the notation has been shortened using

$$\mathtt{c} = \sum_{k=1}^N \frac{\rho_k}{r_k}(e_k^t)^2, \qquad \mathtt{d} = \sum_{k=1}^N \frac{\rho_k}{r_k}e_k, \qquad \mathtt{e} = \sum_{k=1}^N \frac{\rho_k}{r_k}, \tag{A.2}$$

$$\kappa = \rho T + \mathtt{d}, \qquad \Psi(\varphi) = T\rho + \varphi^2\mathtt{e}. \tag{A.3}$$

A.2. Convective flux function Jacobian

The convective flux Jacobian with respect to the conservative set is needed in Eq. (56) to compute the spatial gradient of the physical flux when the residual is computed using the conservative set of variables. For each dimension, it reads

$$\frac{\partial \mathbf{F}_1(\mathbf{q})}{\partial \mathbf{q}} = \begin{bmatrix} -u_1\breve{Y}_1 & \dots & -u_1Y_1 & Y_1 & 0 & 0 & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \\ -u_1Y_N & \dots & -u_1\breve{Y}_N & Y_N & 0 & 0 & 0 \\ \chi_1 - u_1^2 & \dots & \chi_N - u_1^2 & u_1(3-\gamma) & -u_2\breve{\gamma} & -u_3\breve{\gamma} & \breve{\gamma} \\ -u_1u_2 & \dots & -u_1u_2 & u_2 & u_1 & 0 & 0 \\ -u_1u_3 & \dots & -u_1u_3 & u_3 & 0 & u_1 & 0 \\ \left(\chi_1 - h^t\right)u_1 & \dots & \left(\chi_N - h^t\right)u_1 & h^t - u_1^2\breve{\gamma} & -u_1u_2\breve{\gamma} & u_1u_3\breve{\gamma} & \gamma u_1 \end{bmatrix}, \tag{A.4}$$

$$\frac{\partial \mathbf{F}_2(\mathbf{q})}{\partial \mathbf{q}} = \begin{bmatrix} -u_2\breve{Y}_1 & \dots & -u_2Y_1 & 0 & Y_1 & 0 & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \\ -u_2Y_N & \dots & -u_2\breve{Y}_N & 0 & Y_N & 0 & 0 \\ -u_1u_2 & \dots & -u_1u_2 & u_2 & u_1 & 0 & 0 \\ \chi_1 - u_2^2 & \dots & \chi_N - u_2^2 & -u_1\breve{\gamma} & u_2(3-\gamma) & -u_3\breve{\gamma} & \breve{\gamma} \\ -u_2u_3 & \dots & -u_2u_3 & 0 & u_3 & u_2 & 0 \\ \left(\chi_1 - h^t\right)u_2 & \dots & \left(\chi_N - h^t\right)u_2 & -u_1u_2\breve{\gamma} & h^t - u_2^2\breve{\gamma} & u_2u_3\breve{\gamma} & \gamma u_2 \end{bmatrix}, \tag{A.5}$$

$$\frac{\partial \mathbf{F}_3(\mathbf{q})}{\partial \mathbf{q}} = \begin{bmatrix} -u_3\breve{Y}_1 & \dots & -u_3 Y_1 & 0 & 0 & Y_1 & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \\ -u_3 Y_N & \dots & -u_3\breve{Y}_N & 0 & 0 & Y_N & 0 \\ -u_1 u_3 & \dots & -u_1 u_3 & u_3 & 0 & u_1 & 0 \\ -u_2 u_3 & \dots & -u_2 u_3 & 0 & u_3 & u_2 & 0 \\ \chi_1 - u_3^2 & \dots & \chi_N - u_3^2 & -u_1\breve{\gamma} & -u_2\breve{\gamma} & u_3(3-\gamma) & \breve{\gamma} \\ \left(\chi_1 - h^t\right) u_3 & \dots & \left(\chi_N - h^t\right) u_3 & -u_1 u_3\breve{\gamma} & u_2 u_3\breve{\gamma} & h^t - u_3^2\breve{\gamma} & \gamma u_3 \end{bmatrix}, \tag{A.6}$$

where we have simplified the notation using

$$\breve{\gamma} = \gamma - 1, \qquad \breve{Y}_k = Y_k - 1, \qquad \chi_k = h_k - \gamma e_k + \breve{\gamma}\frac{\|\mathbf{u}\|^2}{2}. \tag{A.7}$$

When computing the same term for systems `P-Y` and `E-Y`, the Jacobian of the physical flux with respect to the entropy variables is needed, and it reads

$$\frac{\partial \mathbf{F}_1(\mathbf{v})}{\partial \mathbf{v}} = \begin{bmatrix} u_1\frac{\rho_1}{r_1} & \dots & 0 & \rho_1\Gamma_1(u_1) & u_1u_2\frac{\rho_1}{r_1} & u_1u_3\frac{\rho_1}{r_1} & \rho_1 u_1\,\beta_1 \\ & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots \\ & & u_1\frac{\rho_N}{r_N} & \rho_N\Gamma_N(u_1) & u_1u_2\frac{\rho_N}{r_N} & u_1u_3\frac{\rho_N}{r_N} & \rho_N u_1\,\beta_N \\ & & & u_1\Theta(u_1) & u_2\Psi(u_1) & u_3\Psi(u_1) & \Upsilon(u_1) \\ & & \text{Sym.} & & u_1\Psi(u_2) & u_1u_2u_3\mathtt{e} & u_1u_2\,\kappa \\ & & & & & u_1\Psi(u_3) & u_1u_3\,\kappa \\ & & & & & & u_1\vartheta \end{bmatrix}, \tag{A.8}$$

$$\frac{\partial \mathbf{F}_2(\mathbf{v})}{\partial \mathbf{v}} = \begin{bmatrix} u_2\frac{\rho_1}{r_1} & \dots & 0 & u_1u_2\frac{\rho_1}{r_1} & \rho_1\Gamma_1(u_2) & u_2u_3\frac{\rho_1}{r_1} & \rho_1 u_2\,\beta_1 \\ & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots \\ & & u_2\frac{\rho_N}{r_N} & u_1u_2\frac{\rho_N}{r_N} & \rho_N\Gamma_N(u_2) & u_2u_3\frac{\rho_N}{r_N} & \rho_N u_2\,\beta_N \\ & & & u_2\Psi(u_1) & u_1\Psi(u_2) & u_1u_2u_3\mathtt{e} & u_1u_2\,\kappa \\ & & \text{Sym.} & & u_2\Theta(u_2) & u_3\Psi(u_2) & \Upsilon(u_2) \\ & & & & & u_2\Psi(u_3) & u_2u_3\,\kappa \\ & & & & & & u_2\vartheta \end{bmatrix}, \tag{A.9}$$

$$\frac{\partial \mathbf{F}_3(\mathbf{v})}{\partial \mathbf{v}} = \begin{bmatrix} u_3\frac{\rho_1}{r_1} & \cdots & 0 & u_1u_3\frac{\rho_1}{r_1} & u_2u_3\frac{\rho_1}{r_1} & \rho_1\Gamma_1(u_3) & \rho_1u_3\beta_1 \\ & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots \\ & & u_3\frac{\rho_N}{r_N} & u_1u_3\frac{\rho_N}{r_N} & u_2u_3\frac{\rho_N}{r_N} & \rho_N\Gamma_N(u_3) & \rho_Nu_3\beta_N \\ & & & u_3\Psi(u_1) & u_1u_2u_3\mathtt{e} & u_1\Psi(u_3) & u_1u_3\kappa \\ & & \text{Sym.} & & u_3\Psi(u_2) & u_2\Psi(u_3) & u_2u_3\kappa \\ & & & & & u_3\Theta(u_3) & \Upsilon(u_3) \\ & & & & & & u_3\vartheta \end{bmatrix}, \tag{A.10}$$

where the notation is shortened using

$$\begin{aligned} \beta_k &= \frac{e_k^t}{r_k} + T, & \Gamma_k(\varphi) &= \frac{\varphi^2}{r_k} + T, \\ \vartheta &= 6\left[\mathtt{c} + T\rho\left(\|\mathbf{u}\|^2 + h^t + e^t + \gamma e\right)\right], & \Upsilon(\varphi) &= T\rho h^t + \varphi^2(2T\rho + \mathtt{d}), \\ & & \Theta(\varphi) &= 3T\rho + \varphi^2\mathtt{e}. \end{aligned} \tag{A.11}$$

A.3. Pressure gradients

The pressure gradient with respect to the conservative set of variables is needed in the Eqs. (52) and (53) for computing the shock capturing sensor. Taking the definition of the pressure provided the ideal gases equation of state as in Eq. (7), it is expressed in terms of the conservative variables

$$p = \sum_{k=1}^{N} \rho_k r_k T = \breve{\gamma}\left(q_m - \frac{\sum_{i=N+1}^{N+d} q_i^2}{2\sum_{k=1}^{N} q_k}\right), \tag{A.12}$$

and the partial derivatives composing the gradient result

$$\frac{\partial p}{\partial q_k} = \breve{\gamma}\frac{\sum_{i=N+1}^{m-1} q_i^2}{2\left(\sum_{j=1}^{N} q_j\right)^2} \quad \text{for} \quad k = 1, \dots, N, \tag{A.13}$$

$$\frac{\partial p}{\partial q_i} = -\breve{\gamma}\frac{q_i}{\sum_{k=1}^{N} q_k} \quad \text{for} \quad i = N+1, \dots, N+d \tag{A.14}$$

and

$$\frac{\partial p}{\partial q_m} = \breve{\gamma}. \tag{A.15}$$

We can use this last result for obtaining the pressure gradient in space when solving system `C-Y`, as the derivatives of the conservative variables are available in its dG discretization. Therefore, it is calculated using the chain rule, *i.e.*

$$\frac{\partial p}{\partial x_i} = \frac{\partial p}{\partial \mathbf{q}}\frac{\partial \mathbf{q}}{\partial x_i}. \tag{A.16}$$

Finally, we present the formulae for calculating the gradient of pressure in space as a function of the entropy variables set and its derivatives. Let the pressure gradient be expressed as

$$\frac{\partial p}{\partial x_i} = T\sum_{k=1}^{N} r_k\frac{\partial \rho_k}{\partial x_i} + \rho r\frac{\partial T}{\partial x_i}, \tag{A.17}$$

where T and ρ can be calculated using Eq. (19) and the closed forms for the gradients of density and temperature read

$$\frac{\partial \rho_k}{\partial x_i} = \frac{\rho_k}{r_k}\left[\frac{\partial v_k}{\partial x_i} + \frac{\partial v_m}{\partial x_i}\left(\frac{-c_{v,k}}{v_m} + \frac{1}{2v_m^2}\sum_{j=N+1}^{N+d} v_j^2\right) - \frac{1}{v_{N+1}}\sum_{j=N+1}^{N+d}\frac{\partial v_j}{\partial x_j}\right] \tag{A.18}$$

and

$$\frac{\partial T}{\partial x_i} = \frac{1}{v_m^2}\frac{\partial v_m}{\partial x_i}. \tag{A.19}$$

Appendix B. Exact Riemann solver for the MCE model

In this section, a brief description of the algorithm for the exact Riemann solver from Borisov and Rykov [54] is provided for completeness. We remark that is a simplified version of the solver, since here the model is provided with the ideal gas equation of state. The Riemann problem is defined by

$$\left[\gamma_\iota,\ \rho_\iota,\ u_\iota,\ p_\iota\right]^\top \qquad \text{for} \qquad \iota \in \{L, R\}, \tag{B.1}$$

where γ_ι is the global specific heat ratio of the mixture, ρ_ι is the global density and L and R denote the left and right states respectively. We seek for a solution $\left[\{\tilde{Y}_k\}_{k=1}^N, \tilde{\rho}, \tilde{u}, \tilde{p}\right]^\top$ in a position defined by the similarity parameter $\xi = \tilde{x}/t$, with $\tilde{x} = x - x_0$ being the distance to the interface of the Riemann problem. The equation for the pressure in the intermediate state reads

$$f\left(p, \gamma_L, \rho_L, p_L\right) + f\left(p, p_R, \rho_R, \gamma_R\right) = u_L - u_R, \tag{B.2}$$

where

$$f\left(p, \gamma_\iota, \rho_\iota, p_\iota\right) = \begin{cases} \dfrac{p - p_\iota}{\rho_\iota c_\iota \sqrt{\mathtt{a}_\iota \pi_\iota + \mathtt{b}_\iota}} & \text{if} \quad p \geq p_\iota \\ \dfrac{2}{\gamma_\iota - 1} c_\iota \left[\pi_\iota^{\mathtt{b}_\iota} - 1\right] & \text{if} \quad p < p_\iota \end{cases} \tag{B.3}$$

where c_ι is the speed of sound at side ι, $\mathtt{a}_\iota = \frac{\gamma_\iota + 1}{2\gamma_\iota}$, $\mathtt{b}_\iota = \frac{\gamma_\iota - 1}{2\gamma_\iota}$ and $\pi_\iota = \frac{p}{p_\iota}$. The next step is to solve Eq. (B.2) using a Newton method, therefore we need an initial guess, for example

$$p^{(0)} = \frac{p_L + p_R}{2} \tag{B.4}$$

and solve the system iteratively for the nth iteration using

$$p^{(n)} = p^{(n-1)} - \frac{f\left(p^{(n-1)}, \gamma_L, \rho_L, p_L\right) + f\left(p^{(n-1)}, \gamma_R, \rho_R, p_R\right) - \left(u_L - u_R\right)}{\dfrac{\partial f}{\partial p}\left(p^{(n-1)}, \gamma_L, \rho_L, p_L\right) + \dfrac{\partial f}{\partial p}\left(p^{(n-1)}, \gamma_R, \rho_R, p_R\right)}, \tag{B.5}$$

where

$$\frac{\partial f}{\partial p}\left(p, \gamma_\iota, \rho_\iota, p_\iota\right) = \begin{cases} \dfrac{\left(\gamma_\iota + 1\right)\pi_\iota + \left(3\gamma_\iota - 1\right)}{4\gamma_\iota \rho_\iota c_\iota \left(\mathtt{a}_\iota \pi_\iota + \mathtt{b}_\iota\right)^{3/2}} & \text{if} \quad p \geq p_\iota \\ \dfrac{1}{\gamma_\iota p} c_\iota \left(\pi_\iota\right)^{\mathtt{b}_\iota} & \text{if} \quad p < p_\iota \end{cases} \tag{B.6}$$

The numerical procedure is stopped when

$$|p^{(n)} - p^{(n-1)}| < \epsilon \min\{p_L, p_R\}, \tag{B.7}$$

where $\epsilon = 5 \cdot 10^{-11}$ for the present work. Once the intermediate pressure is found, we set $p = p^{(n)}$ and find the intermediate velocity using

$$u = \frac{1}{2}\left[u_L - f\left(p, \gamma_L, \rho_L, p_L\right) + u_R + f\left(p, \gamma_R, \rho_R, p_R\right)\right]; \tag{B.8}$$

and we establish the wave pattern and wave velocities for the direction ι as

1. if $(p - p_\iota) \leq \epsilon \min\{p_L, p_R\}$, there is a rarefaction wave moving to direction ι, therefore

$$c_\iota^* = c_\iota + \frac{1}{2}(\gamma_\iota - 1)\left(u_\iota - u\right), \qquad \rho_\iota^* = \frac{\gamma_\iota p}{\left(c_\iota^*\right)^2}, \tag{B.9}$$

$$D_\iota = u_\iota - c_\iota, \qquad D_\iota^* = u - c_\iota^*; \tag{B.10}$$

2. if $(p - p_\iota) > \epsilon\ min\{p_L, p_R\}$, there is a shock wave moving to direction ι, so

$$\rho_\iota^* = \rho_\iota \frac{(\gamma_\iota + 1)p + (\gamma_\iota - 1)p_\iota}{(\gamma_\iota - 1)p + (\gamma_\iota + 1)p_\iota}, \tag{B.11}$$

$$D_\iota = D_\iota^* = \frac{\rho_\iota u_\iota - \rho_\iota^* u}{\rho_\iota \rho_\iota^*}. \tag{B.12}$$

After that, we use ξ to find which is the state at the evaluated position of the solution

1. if $\xi \leq D_L$, the state at the evaluated position is in the non-perturbed left state;

2. if $D_L < \xi \leq D_L^*$, the evaluated position is in a rarefaction wave, therefore

$$\tilde{c} = \frac{2c_L + (\gamma_L - 1)(u_L - \xi)}{\gamma_L + 1}, \qquad \tilde{\rho} = \rho_L \left(\frac{\tilde{c}}{c_L} \right)^{\frac{2}{\gamma_L - 1}}, \tag{B.13}$$

$$\tilde{u} = \xi + \tilde{c}, \qquad \tilde{p} = \frac{\tilde{\rho}\tilde{c}^2}{\gamma_L}, \tag{B.14}$$

$$\tilde{Y}_k = Y_{k,L} \qquad \text{for} \qquad k = 1, \dots, N; \tag{B.15}$$

3. if $D_L^* < \xi \leq u$, the evaluated position is between the left genuinely nonlinear wave and the contact wave, therefore

$$\tilde{\rho} = \rho_L^*, \qquad \tilde{u} = u, \qquad \tilde{p} = p, \tag{B.16}$$

$$\tilde{Y}_k = Y_{k,L} \qquad \text{for} \qquad k = 1, \dots, N; \tag{B.17}$$

4. if $u < \xi \leq D_R^*$, the evaluated position is between the contact wave and the right genuinely nonlinear wave, therefore

$$\tilde{\rho} = \rho_R^*, \qquad \tilde{u} = u, \qquad \tilde{p} = p, \tag{B.18}$$

$$\tilde{Y}_k = Y_{k,R} \qquad \text{for} \qquad k = 1, \dots, N; \tag{B.19}$$

5. if $D_R^* < \xi \leq D_R$ the evaluated position is in the right rarefaction wave

$$\tilde{c} = \frac{2c_R + (\gamma_R - 1)(u_R - \xi)}{\gamma_R + 1}, \qquad \tilde{\rho} = \rho_R \left(\frac{\tilde{c}}{c_R} \right)^{\frac{2}{\gamma_R - 1}}, \tag{B.20}$$

$$\tilde{u} = \xi + \tilde{c}, \qquad \tilde{p} = \frac{\tilde{\rho}\tilde{c}^2}{\gamma_R}, \tag{B.21}$$

$$\tilde{Y}_k = Y_{k,R} \qquad \text{for} \qquad k = 1, \dots, N; \tag{B.22}$$

6. if $D_R < \xi$ the state at the evaluated position is in the non-perturbed right state.

Finally, we recall that the Godunov flux will be calculated evaluating the physical flux at the solution obtained at $\xi = 0$ for the Riemann problems at the cell interfaces.

References

[1] J.-F. Haas, B. Sturtevant, Interaction of weak shock waves with cylindrical and spherical gas inhomogeneities, J. Fluid Mech. 181 (1987) 41–76.
[2] J.J. Quirk, S. Karni, On the dynamics of a shock-bubble interaction, J. Fluid Mech. 318 (1996) 129–163.
[3] J. Glimm, J.W. Grove, X.L. Li, K.-m. Shyue, Y. Zeng, Q. Zhang, Three-dimensional front tracking, SIAM J. Sci. Comput. 19 (3) (1998) 703–727.
[4] A. Gouasmi, K. Duraisamy, S.M. Murman, Formulation of entropy-stable schemes for the multicomponent compressible euler equations, Comput. Methods Appl. Mech. Eng. 363 (2020) 112912.
[5] R. Abgrall, How to prevent pressure oscillations in multicomponent flow calculations: a quasi conservative approach, J. Comput. Phys. 125 (1996) 150–160.
[6] S. Karni, Multicomponent flow calculations by a consistent primitive algorithm, J. Comput. Phys. 112 (1994) 31–43.
[7] R. Abgrall, S. Karni, Computations of compressible multifluids, J. Comput. Phys. 169 (2) (2001) 594–623.
[8] T.Y. Hou, P.G.L. Floch, Why nonconservative schemes converge to wrong solutions: error analysis, Math. Comput. 62 (206) (1994) 497–530.
[9] N. Franchina, M. Savini, F. Bassi, Multicomponent gas flow computations by a discontinuous Galerkin scheme using L2-projection of perfect gas EOS, J. Comput. Phys. 315 (2016) 302–322.
[10] E.J. Ching, R.F. Johnson, A.D. Kercher, Conservative, Pressure-Based Discontinuous Galerkin Method for Compressible, Multicomponent Flows, American Institute of Aeronautics and Astronautics, 2025.
[11] R. Abgrall, A general framework to construct schemes satisfying additional conservation relations. application to entropy conservative and entropy dissipative schemes, J. Comput. Phys. 372 (2018) 640–666.
[12] R. Abgrall, P. Öffner, H. Ranocha, Reinterpretation and extension of entropy correction terms for residual distribution and discontinuous Galerkin schemes: application to structure preserving discretization, J. Comput. Phys. 453 (2022) 110955.
[13] Y. Fujiwara, Y. Tamaki, S. Kawai, Fully conservative and pressure-equilibrium preserving scheme for compressible multi-component flows, J. Comput. Phys. 478 (2023) 111973.
[14] V. Badrkhani, T.J.P. Karpowsk, C. Hasse, An entropy-stable/double-flux scheme for the multi-component compressible Navier-Stokes equations, (2025) https://arxiv.org/abs/2506.13231.
[15] H. Terashima, N. Ly, M. Ihme, Approximately pressure-equilibrium-preserving scheme for fully conservative simulations of compressible multi-species and real-fluid interfacial flows, J. Comput. Phys. 524 (2025) 113701.
[16] E.J. Ching, R.F. Johnson, A.D. Kercher, Positivity-preserving and entropy-bounded discontinuous Galerkin method for the chemically reacting, compressible Euler equations. Part I: the one-dimensional case, J. Comput. Phys. 505 (2024) 112881.
[17] X. Zhang, C.-W. Shu, On positivity-preserving high order discontinuous Galerkin schemes for compressible Euler equations on rectangular meshes, J. Comput. Phys. 229 (23) (2010) 8918–8934.
[18] X. Zhang, C.-W. Shu, Positivity-preserving high order discontinuous Galerkin schemes for compressible Euler equations with source terms, J. Comput. Phys. 230 (4) (2011) 1238–1248.

[19] F. Renac, Entropy-stable, robust, and high-order DGSEM for the compressible multicomponent Euler equations, J. Comput. Phys. 445 (2021) 110584.
[20] T. Dzanic, F.D. Witherden, Positivity-preserving entropy-based adaptive filtering for discontinuous spectral element methods, J. Comput. Phys. 468 (2022) 111501.
[21] L. Alberti, F. Bassi, E. Carnevali, A. Colombo, A. Crivellini, A. Nigro, A comparative study of different sets of variables in a discontinuous Galerkin method with entropy balance enforcement, Int. J. Comput. Fluid Dyn. 37 (2024) 487–508.
[22] F. Bassi, L. Botti, A. Colombo, A. Crivellini, A. Ghidoni, F. Massa, On the development of an implicit high-order discontinuous Galerkin method for DNS and implicit LES of turbulent flows, Eur. J. Mech. B Fluids 55 (2016) 367–379.
[23] L. Alberti, E. Cammalleri, E. Carnevali, A. Nigro, On the use of different sets of variables for solving unsteady inviscid flows with an implicit discontinuous Galerkin method, Fluids 9 (11) (2024) 248.
[24] E. Tadmor, The numerical viscosity of entropy stable schemes for systems of conservation laws. I, Math. Comput. 49 (1987) 91–103.
[25] F. Ismail, P.L. Roe, Affordable, entropy-consistent Euler flux functions II: entropy production at shocks, J. Comput. Phys. 228 (2009) 5410–5436.
[26] A. Aiello, C. De Michele, G. Coppola, Entropy conservative discretization of compressible Euler equations with an arbitrary equation of state, J. Comput. Phys. 528 (2025) 113836.
[27] C. De Michele, A.K. Edoh, G. Coppola, Finite-difference compatible entropy-conserving schemes for the compressible Euler equations, J. Comput. Phys. 540 (2025) 114262.
[28] A. Aiello, C. De Michele, G. Coppola, Formulation of entropy-conservative discretizations for compressible flows of thermally perfect gases, 2025 https://arxiv.org/abs/2507.08115. 2507.08115
[29] P. Chandrashekar, Kinetic energy preserving and entropy stable finite volume schemes for compressible Euler and Navier-Stokes equations, Commun. Comput. Phys. 14 (2013) 1252–1286.
[30] J. Chan, An artificial viscosity approach to high order entropy stable discontinuous Galerkin methods, J. Comput. Phys. 543 (2025) 114380.
[31] C. DeGrendele, N. Ly, F. Cadieux, M. Barad, D. Lee, J. Duensing, On the construction of high-order and exact pressure equilibrium schemes for arbitrary equations of state, 2025 https://arxiv.org/abs/2512.04450. 2512.04450
[32] Y. Wang, A. Wehrfritz, E.R. Hawkes, Physically consistent formulations of split convective terms for turbulent compressible multi-component flows, J. Comput. Phys. 540 (2025) 114269.
[33] T.J.R. Hughes, L.P. Franca, M. Mallet, A new finite element formulation for computational fluid dynamics: I. Symmetric forms of the compressible Euler and Navier–Stokes equations and the second law of thermodynamics, Comput. Methods Appl. Mech. Eng. 54 (1986) 223–234.
[34] F. Chalot, T.J.R. Hughes, F. Shakib, Symmetrization of conservation laws with entropy for high-temperature hypersonic computations, Comput. Syst. Eng. 1 (2) (1990) 495–521. Computational Technology for Flight Vehicles.
[35] L. Alberti, E. Carnevali, A. Colombo, A. Crivellini, An entropy-conserving/stable discontinuous Galerkin solver in entropy variables based on the direct enforcement of entropy balance, J. Comput. Phys. 508 (2024) 113007.
[36] A. Colombo, A. Crivellini, A. Nigro, On the entropy conserving/stable implicit DG discretization of the Euler equations in entropy variables, Comput. Fluids. 232 (2022) 105198.
[37] E. Gaburro, P. Öffner, M. Ricchiuto, D. Torlo, High order entropy preserving ADER-DG schemes, Appl. Math. Comput. 440 (2023) 127644.
[38] K.O. Friedrichs, P.D. Lax, Systems of conservation equations with a convex extension, Proc. Natl Acad. SCI. 68 (1971) 1686–1688.
[39] M.S. Mock, Systems of conservation laws of mixed type, Differ. Eq. 37 (1980) 70–88.
[40] P.D. Lax, Hyperbolic Systems of Conservation Laws and the Mathematical Theory of Shock Waves, Society for Industrial and Applied Mathematics, 1973.
[41] S.N. Kružkov, First order quasilinear equations in several independent variables, Math. USSR-Sbornik 10 (1970) 217.
[42] F. Bassi, L. Botti, A. Colombo, D.A. Di Pietro, P. Tesini, "]On the flexibility of agglomeration-based physical space discontinuous Galerkin discretizations, J. Comput. Phys. 231 (1) (2012) 45–65.
[43] F. Bassi, L. Botti, A. Colombo, S. Rebay, Agglomeration based discontinuous Galerkin discretization of the Euler and Navier-Stokes equations, Comput. Fluids 61 (2012) 77–85.
[44] J. Chan, H. Ranocha, A.M. Rueda-Ramírez, G. Gassner, T. Warburton, On the entropy projection and the robustness of high order entropy stable discontinuous Galerkin schemes for under-resolved flows, Front. Phys. 10 (2022) 898028.
[45] F. Bassi, A. Colombo, A. Crivellini, K.J. Fidkowski, M. Franciolini, A. Ghidoni, G. Noventa, Entropy-adjoint p-adaptive discontinuous Galerkin method for the under-resolved simulation of turbulent flows, AIAA J. 58 (2020) 3963–3977.
[46] T.C. Fisher, M.H. Carpenter, High-order entropy stable finite difference schemes for nonlinear conservation laws: finite domains, J. Comput. Phys. 252 (2013) 518–557.
[47] G.J. Gassner, A skew-symmetric discontinuous Galerkin spectral element discretization and its relation to SBP-SAT finite difference methods, SIAM J. Sci. Comput. 35 (2013) A1233–A1253.
[48] Y. Mantri, P. Öffner, M. Ricchiuto, Fully well-balanced entropy controlled discontinuous Galerkin spectral element method for shallow water flows: global flux quadrature and cell entropy correction, SIAM J. Sci. Comput. 498 (2024) 112673.
[49] A.K. Edoh, Conservative correction procedures utilizing artificial dissipation operators, SIAM J. Sci. Comput. 504 (112880) (2024) 112880.
[50] M.J. Picklo, A.K. Edoh, Entropy correction with SIAC filters for high-order DG methods, SIAM J. Sci. Comput. 104 (2) (2025) 39.
[51] L. Alberti, E. Carnevali, A. Colombo, A. Crivellini, Continued development of an entropy-aware high-order modal discontinuous Galerkin solver for the Navier-Stokes equations, Comput. Fluids 299 (106730) (2025) 106730.
[52] G.J. Gassner, A.R. Winters, D.A. Kopriva, Split form nodal discontinuous Galerkin schemes with summation-by-parts property for the compressible Euler equations, J. Comput. Phys. 327 (2016) 39–66.
[53] S.K. Godunov, I. Bohachevsky, Finite difference method for numerical computation of discontinuous solutions of the equations of fluid dynamics, Matematičeskij Sbornik 47(89) (1959) 271–306.
[54] V. Borisov, Y. Rykov, An exact riemann solver in the algorithms for multicomponent gas dynamics, Keldysh Institute Preprints (2018) 1–28.
[55] E. Tadmor, Chapter 18 - entropy stable schemes, in: R. Abgrall, C.-W. Shu (Eds.), Handbook of Numerical Methods for Hyperbolic Problems, 17 of *Handbook of Numerical Analysis*, Elsevier, 2016, pp. 467–493.
[56] F. Bassi, A. Crivellini, A. Ghidoni, S. Rebay, High-order discontinuous Galerkin discretization of transonic turbulent flows, in: 47th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition, American Institute of Aeronautics and Astronautics, 2009.
[57] F. Bassi, L. Botti, A. Colombo, A. Crivellini, N. Franchina, A. Ghidoni, S. Rebay, Very high-order accurate discontinuous Galerkin computation of transonic turbulent flows on aeronautical configurations, in: N. Kroll, H. Bieler, H. Deconinck, V. Couaillier, H. van der Ven, K. Sørensen (Eds.), ADIGMA - A European Initiative on the Development of Adaptive Higher-Order Variational Methods for Aerospace Applications, Springer Berlin Heidelberg, Berlin, Heidelberg, 2010, pp. 25–38.
[58] R.J. Spiteri, S.J. Ruuth, A new class of optimal high-order strong-stability-preserving time discretization methods, SIAM J. Numer. Anal. 40 (2002) 469–491.
[59] A. Colombo, A. Crivellini, A. Nigro, Entropy conserving implicit time integration in a discontinuous Galerkin solver in entropy variables, J. Comput. Phys. 472 (2023) 111683.
[60] H. Ranocha, M. Sayyari, L. Dalcin, M. Parsani, D.I. Ketcheson, Relaxation Runge–Kutta methods: fully discrete explicit entropy-stable schemes for the compressible Euler and Navier–Stokes equations, SIAM J. Sci. Comput. 42 (2) (2020) A612–A638.
[61] M.R. Najafian, B.C. Vermeire, Quasi-orthogonal Runge-Kutta projection methods, J. Comput. Phys. 530 (2025) 113917.
[62] R. Liska, B. Wendroff, Comparison of several difference schemes on 1D and 2D test problems for the Euler equations, SIAM J. Sci. Comput. 25 (3) (2003) 995–1017.
[63] S. Singh, M. Torrilhon, On the shock-driven hydrodynamic instability in square and rectangular light gas bubbles: a comparative study from numerical simulations, Phys. Fluids 35 (1) (2023) 012117.
[64] S. Steinberg, P.J. Roache, Symbolic manipulation and computational fluid dynamics, J. Comput. Phys. 57 (2) (1985) 251–284.

[65] C.J. Roy, C.C. Nelson, T.M. Smith, C.C. Ober, Verification of Euler/Navier-Stokes codes using the method of manufactured solutions, Int. J. Numer. Methods Fluids 44 (6) (2004) 599–620.
[66] F. Renac, Entropy-stable DGSEM for nonlinear hyperbolic systems in nonconservative form with application to two-phase flows, J. Comput. Phys. 382 (2019) 1–26.
[67] F. Coquel, C. Marmignon, P. Rai, F. Renac, An entropy stable high-order discontinuous Galerkin spectral element method for the Baer-Nunziato two-phase flow model, J. Comput. Phys. 431 (2021) 110135.
[68] G.J. Gassner, M. Svärd, F.J. Hindenlang, Stability issues of entropy-stable and/or split-form high-order schemes, J. Sci. Comput. 90 (2022) 79.
[69] E. Toro, Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction, Springer Berlin Heidelberg, 2009.
[70] S. Dellacherie, J. Jung, P. Omnes, P.A. Raviart, Construction of modified Godunov-type schemes accurate at any Mach number for the compressible Euler system, Math. Models Methods Appl. Sci. 26 (13) (2016) 2525–2615.
[71] H. Ranocha, L. Dalcin, M. Parsani, D.I. Ketcheson, Optimized Runge-Kutta methods with automatic step size control for compressible computational fluid dynamics, Commun. Appl. Math. Comput. 4 (2022) 1191–1228.
[72] W.H. Hui, P.Y. Li, Z.W. Li, A unified coordinate system for solving the two-dimensional Euler equations, J. Comput. Phys. 153 (2) (1999) 596–637.
[73] E. Gaburro, W. Boscheri, S. Chiocchetti, M. Ricchiuto, Discontinuous Galerkin schemes for hyperbolic systems in non-conservative variables: quasi-conservative formulation with subcell finite volume corrections, Comput. Methods Appl. Mech. Eng. 431 (2024) 117311.
[74] S. Kawai, H. Terashima, A high-resolution scheme for compressible multicomponent flows with shock waves, Int. J. Numer. Methods Fluids 66 (2011) 1207-1225.
[75] F. Diegelmann, S. Hickel, N.A. Adams, Three-dimensional reacting shock-bubble interaction, Combust. Flame. 181 (2017) 300–314.