Journal of Computational Physics 508 (2024) 113007

Contents lists available at ScienceDirect

Journal of Computational Physics

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

An entropy conserving/stable discontinuous Galerkin solver in entropy variables based on the direct enforcement of entropy balance

Luca Alberti [a], Emanuele Carnevali [a], Alessandro Colombo [b], Andrea Crivellini [a,*]

[a] Department of Industrial Engineering and Mathematical Sciences (DIISM), Marche Polytechnic University, Ancona, 60131, Italy
[b] Department of Engineering and Applied Sciences (DISA), University of Bergamo, Dalmine (BG), 24044, Italy

ARTICLE INFO

Keywords:
Discontinuous Galerkin
Entropy conservation/stability
Entropy variables
Direct enforcement of entropy balance
Entropy projection
Generalized Crank-Nicolson

ABSTRACT

Numerical schemes that guarantee the preservation of flow properties as prescribed by physical laws, ensure high-fidelity results and improved computational robustness. This work presents a highly accurate entropy conserving/stable solver for the solution of the compressible Euler equations. The method uses a modal Discontinuous Galerkin approximation and takes advantage of both entropy variables and the Direct Enforcement of Entropy Balance (DEEB), proposed by Abgrall [1] to fulfill the second law of thermodynamics at the discrete level. Thanks to the use of an orthonormal modal basis, the cost of adding the DEEB correction to the spatially discretized equations is negligible. Entropy variables are used directly, as unknowns, or to assemble the discrete operators while solving for the conservative variables. In the latter case, the conservative variables are "converted" into the entropy ones through an L_2 "entropy projection". If this strategy is coupled with DEEB, a very efficient alternative to the direct use of the entropy variables is obtained, with a dramatic improvement in robustness over the direct use of the conservative variables. For the time integration both an explicit Strong Stability Preserving Runge-Kutta scheme and the implicit Generalized Crank-Nicolson method are considered. The latter allows for the construction of a discrete entropy conservative/stable scheme both in space and time to be used as a reference. Convergence and conservation properties are demonstrated together with robustness and computational efficiency on a suite of two-dimensional test cases, showing that by coupling DEEB with entropy projection the solver robustness is improved while not affecting the stability limit.

1. Introduction

In this contribution we report on the development and assessment of a high-order, modal Discontinuous Galerkin (DG) method for the simulation of multi-scale compressible flow phenomena. The solver is designed to be efficient and guarantee entropy conservation/stability at the discrete level, which for fluid mechanics means following the second law of thermodynamics, i.e. physical entropy must not be destroyed over time.

* Corresponding author.
E-mail address: a.crivellini@staff.univpm.it (A. Crivellini).

https://doi.org/10.1016/j.jcp.2024.113007
Received 21 September 2023; Received in revised form 15 February 2024; Accepted 10 April 2024

Available online 15 April 2024

Since the seminal work of Hughes et al. [2] there has been a significant surge of interest, within the computational fluid dynamics community, in devising numerical solvers for the governing equations of fluid mechanics able to satisfy an additional scalar equation for the entropy. In the framework of Finite Element Methods, the need to adopt a scheme based on entropy variables [2,3], and exactly integrate the variational formulation, directed the research mostly toward the development of nodal and collocated methods.

The representation of the numerical solution in a suitably chosen set of quadrature points allows for the definition of Summation by Parts (SBP) operators [4], a finite difference construct that accurately discretizes spatial derivatives and enforces integration by parts to hold at the discrete level also. This approach, coupled with appropriate flux-differencing formulae, was exploited to design several entropy stable nodal DG schemes, see [5–10]. Examples of entropy conserving/stable DG formulations using modal approximations can be found in the work of Chan [11,12], where conservation is still enforced by means of SBP-like operators and adequate flux functions.

In Colombo et al. [13] the modal DG framework presented in this contribution was exploited to construct an entropy conserving/stable method without posing any constraint on the structure of the numerical scheme. This was achieved by making direct use of entropy variables and selecting suitable flux formulae. In this discretization the adoption of hierarchical and orthonormal modal basis defined in physical space [14] reduces the elemental mass matrix to the identity when the set of conservative variables is used. However, when a different set of variables is considered, e.g. the entropy ones, the local mass matrix becomes full and couples all degrees of freedom (DOFs) of the different equations. In such case, the cost of its assembly and inversion drastically reduces the overall performance of the scheme. Inspired by the work of Chan et al. [15], to eliminate this cost while preserving the good properties of entropy conserving/stable schemes, we propose an efficient alternative where the non-stationary term is expressed in conservative form, while the evaluation of the spatial discrete operator uses the L_2-projection of the conservative variables on the entropy ones. In any case, the computation of the spatial operators with the entropy variables, used either directly or through the "entropy projection", requires the variational formulation to be exactly integrated in order to guarantee entropy conservation/stability. From a practical viewpoint this results in a numerical *over-integration*, i.e. in the use of quadrature rules with a very high degree of exactness not known beforehand, causing a significant degradation of the computational performance. As a possible way to avoid over-integration, Chen and Shu [8] considered the Direct Enforcement of Entropy Balance (DEEB) approach, firstly developed by Abgrall [1] in the general framework of Residual Distribution. Later the approach was extended to DG in [16], while recent applications/developments in the ADER-DG framework can be found in [17]. In the DEEB algorithm an explicit correction term is added to the spatially discretized equations in such a way to remove spurious entropy variations and guarantee entropy conservation/stability.

When the entropy variables are the working set, the correction has the purpose to enforce the validity of the chain rule at the discrete level, but as pointed out by Abgrall et al. in [16], DEEB can be generalized to correct any source of spatial entropy error, like the projection error between conservative and entropy variables that appears when the conservative variables are the working set. The numerical results outlined throughout the paper demonstrate the improved robustness of "pure" and "projected" entropy variables over a standard implementation making use of conservative variables and highlight the beneficial role played by DEEB in making each set more robust.

For the time integration, both an explicit Strong Stability Preserving Runge-Kutta (SSPRK) scheme [18] and the implicit Generalized Crank-Nicolson (GCN) [19] method are implemented. Although classical explicit schemes, like the SSPRK, do not ensure a priori the conservation of entropy even for smooth solutions, see [20], the coupling with explicit time integrations is here particularly attractive as it completely avoids linear algebra. In this work, the GCN is mostly used as a reference to investigate the effects of spurious entropy production in time on the robustness, as it leads to a fully discrete entropy conserving/stable scheme.

The paper is organized as follows. In Sec. 2 the governing equations of inviscid fluid dynamics and their link with entropy conservation are presented. In Sec. 3 the numerical framework is introduced with details on the implementation of the "entropy projection" and DEEB when orthonormal basis functions are used. In Sec. 4 numerical results are presented discussing convergence and conservation properties, robustness and computational efficiency, with the underlying purpose of comparing: *i*) the direct use of the entropy variables with the "entropy projection"; *ii*) over-integration with the DEEB approach. Finally, concluding remarks and future research directions are given in Sec. 5.

2. The governing equations and the entropy framework

Treating only inviscid flows, our starting point are the compressible Euler equations in their conservative form

$$\begin{aligned}
&\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x_j}\left(\rho u_j\right) = 0,\\
&\frac{\partial}{\partial t}\left(\rho u_i\right) + \frac{\partial}{\partial x_j}\left(\rho u_j u_i\right) = -\frac{\partial p}{\partial x_i} + \rho g_i,\\
&\frac{\partial}{\partial t}\left(\rho E\right) + \frac{\partial}{\partial x_j}\left(\rho u_j H\right) = \rho g_i u_i,
\end{aligned} \tag{1}$$

where ρ is the fluid density, p is the pressure, $\mathbf{u} = \left[u_1, \dots, u_d\right]^\top$ is the velocity vector, E and H are the total energy and enthalpy, respectively, and $i, j = 1, \dots, d$, with d being the number of geometric dimensions. For a perfect gas $p = \rho RT$, where R is the specific gas constant and T is the thermodynamic temperature. Moreover, the pressure is given by $p = (\gamma - 1)\left(\rho E - |\rho \mathbf{u}|^2/2\rho\right)$ where $\gamma = c_p/c_v$ is the ratio of gas specific heats, here set to 1.4. Finally, g_i is the i-th component of the gravitational acceleration vector $\mathbf{g}$.

The system in Eq. (1) can be written in compact form as

$$\frac{\partial \mathbf{q}}{\partial t} + \frac{\partial \mathbf{F}_i}{\partial x_i} = \mathbf{b}, \tag{2}$$

with implied summation over the range $i = 1, \dots, d$. Here $\mathbf{q} = [\rho, \rho\mathbf{u}, \rho E]^\top$ is the vector of the conservative variables, $\mathbf{F} \in \mathbb{R}^{(2+d)\times d}$ is the convective flux function and $\mathbf{b} = \left[0, \rho\mathbf{g}, \rho g_i u_i\right]^\top$ is the possible source term due to the gravitational body force.

The Euler equations account for a well-known feature of nonlinear systems: initially smooth solutions may cease to be differentiable after a finite amount of time, developing discontinuities in the form of shocks. In such cases, solutions to the conservation equations exist only in an integral, weak sense with conservation at shock locations expressed by the Rankine-Hugoniot relation. These generalized, non-differentiable solutions to systems of conservation equations are known to be non-unique [21] for given initial conditions so that suitable constraints must be enforced to single out the physically meaningful ones.

Lax and Friedrichs [22] first outlined a general extension of the conservation system with an additional scalar inequality

$$\frac{\partial \mathcal{S}}{\partial t} + \frac{\partial \sigma_i}{\partial x_i} \leq 0, \tag{3}$$

upon the existence of a suitable compatibility condition

$$\frac{\partial \mathcal{S}}{\partial \mathbf{q}} \frac{\partial \mathbf{F}_i}{\partial \mathbf{q}} = \frac{\partial \sigma_i}{\partial \mathbf{q}}, \tag{4}$$

involving the entropy pair $(\mathcal{S}, \sigma_i)$.

Mock [3] later reformulated this framework as a change of variables $\mathbf{v}^\top \equiv \partial \mathcal{S}/\partial \mathbf{q} \mapsto \mathbf{q}(\mathbf{v})$

$$\frac{\partial \mathbf{q}}{\partial \mathbf{v}} \frac{\partial \mathbf{v}}{\partial t} + \frac{\partial \mathbf{F}_i}{\partial \mathbf{q}} \frac{\partial \mathbf{q}}{\partial \mathbf{v}} \frac{\partial \mathbf{v}}{\partial x_i} = \mathbf{b}, \tag{5}$$

where the convexity of $\mathcal{S}$ ensured the existence of a one-to-one mapping reducing Eq. (2) into its symmetric form (5). The analytical equivalence between Eqs. (2) and (5), coupled with the compatibility condition (4), ensures again the reduction of the mapped system to the scalar conservation statement of Eq. (3) upon left multiplication by $\mathbf{v}^\top$.

For the Euler equations, Harten et al. [23,24] demonstrated that the thermodynamic entropy density behaves exactly as a convex generalized entropy function. Hughes et al. [2] later showed that discretizing the Navier-Stokes equations in a Galerkin finite element framework, and using entropy variables, naturally induces an *implicit resolution* of Eq. (3) in integral form.

To maintain an entropy preserving property also at the discrete level, however, appropriate strategies involving specifically designed space/time algorithms must be used, see Sec. 3. According to Refs. [23,24,2] and as already reported in [13,19], here the following entropy pair is considered

$$\mathcal{S} = \frac{-\rho s}{\gamma - 1}, \tag{6}$$

$$\sigma_i = u_i \mathcal{S}, \tag{7}$$

where $s = \ln(p/\rho^\gamma)$ denotes the specific thermodynamic entropy. Due to the sign assumed by the mathematical entropy $\mathcal{S}$, the production of physical entropy s corresponds to the destruction of $\mathcal{S}$ and viceversa.

By taking the Jacobian of $\mathcal{S}$ with respect to the conservative state $\mathbf{q}$, the set of entropy variables is derived

$$\mathbf{v} = \left[\frac{\gamma - s}{\gamma - 1} - \frac{1}{2}\frac{\rho|\mathbf{u}|^2}{p}, \ \frac{\rho u_i}{p}, \ -\frac{\rho}{p}\right]^\top. \tag{8}$$

Note that, with this choice, the contribution to Eq. (4) of the gravitational source terms, which is $\mathbf{v}^\top\mathbf{b}$, cancels out.

Finally, the existence of the aforementioned symmetric operators further allows to introduce the entropy potentials, $\theta = \mathbf{v}^\top\mathbf{q} - \mathcal{S}$ and $\boldsymbol{\psi} = [\psi_1, \dots, \psi_d]^\top$, where $\psi_i = \mathbf{v}^\top\mathbf{F}_i - \sigma_i$ and $\mathbf{q}^\top = \partial\theta/\partial\mathbf{v}$, see Refs. [3,25]. The latter were proven to reduce to $\theta = \rho$ and $\psi_i = \rho u_i$ provided that the entropy variables are in the form presented in Eq. (8).

Some relevant relationships, peculiar to the entropy formulation, are given throughout this paper to clarify the nature of the correction introduced by the DEEB. These may be easily extracted from the founding contributions to the entropy framework of Refs. [2,23,3,25] but are outlined here in a different form and to provide a base for the considerations reported in Sec 3.2. We start by noting that computing the divergence of $\psi_i = \mathbf{v}^\top\mathbf{F}_i - \sigma_i$ and taking advantage of the compatibility condition of Eq. (4), from which we get $\partial\sigma_i/\partial x_i = \mathbf{v}^\top\partial\mathbf{F}_i/\partial x_i$, the following relation is obtained

$$\frac{\partial \psi_i}{\partial x_i} = \frac{\partial \mathbf{v}^\top}{\partial x_i}\mathbf{F}_i. \tag{9}$$

As it will be clear later, the fulfillment of Eq. (9) plays a crucial role in ensuring the entropy conservation/stability of the DG discretized Euler equations.

3. The discrete setting

The DG spatial discretization is built on top of a triangulation $\mathcal{K}_h = \{K\}$ of the computational domain Ω_h, approximating the physical space $\Omega \subset \mathbb{R}^d$ by means of non-overlapping mesh elements K. For the sake of a detailed presentation of the numerical method, the Euler equations are written as

$$\mathbf{P}(\mathbf{w})\frac{\partial \mathbf{w}}{\partial t} + \frac{\partial \mathbf{F}_i(\mathbf{w})}{\partial x_i} = \mathbf{b}(\mathbf{w}), \tag{10}$$

where $\mathbf{w} \in \mathbb{R}^{2+d}$ is the unknown solution vector and $\mathbf{P}(\mathbf{w}) = \partial \mathbf{q}(\mathbf{w})/\partial \mathbf{w} \in \mathbb{R}^{2+d} \otimes \mathbb{R}^{2+d}$ is the Jacobian matrix introduced by the possible change of variables from the conservative set $\mathbf{q}$ to the generic set $\mathbf{w}$.

The weak formulation of Eq. (10) is derived by scalar multiplication with an arbitrary smooth test function $\mathbf{\Phi} \in \mathbb{R}^{2+d}$, and integration over the entire domain Ω

$$\int_\Omega \mathbf{\Phi}^\top \mathbf{P}(\mathbf{w}) \frac{\partial \mathbf{w}}{\partial t}\, d\Omega - \int_\Omega \left(\frac{\partial \mathbf{\Phi}}{\partial x_i}\right)^\top \mathbf{F}_i(\mathbf{w})\, d\Omega + \int_{\partial\Omega} \mathbf{\Phi}^\top \mathbf{F}_i(\mathbf{w})\, n_i\, d\sigma - \int_\Omega \mathbf{\Phi}^\top \mathbf{b}(\mathbf{w})\, d\Omega = 0, \tag{11}$$

where integration by parts was exploited and $\{n_i\}_{i=1}^d$ identifies the outward-oriented, unit normal vector to the boundary $\partial\Omega$. As highlighted by Hughes et al. [2], by choosing $\mathbf{w} \equiv \mathbf{v}$, and with no a priori assumption on $\mathbf{\Phi}$, the weak formulation implies the integral version of Eq. (3) and induces the global constraint

$$\frac{d}{dt}\int_\Omega S(\mathbf{w}) d\Omega \leq 0. \tag{12}$$

To proceed with the discretization of Eq. (11) the continuous vector functions $\mathbf{w}$ and $\mathbf{\Phi}$ are replaced with finite element approximations $\mathbf{\Phi}_h$, $\mathbf{w}_h \in \left[\mathcal{V}_h\right]^{d+2}$, where

$$\mathcal{V}_h = \{\phi \in L_2(\Omega) \ : \ \phi|_K \in \mathbb{P}_d^k(K), \ \forall K \in \mathcal{K}_h\} \tag{13}$$

is the set of piecewise-continuous, k-th degree, d-dimensional polynomial functions with compact support on K. As a modal basis for this polynomial space, the set of orthonormal and hierarchical functions $\{\phi_j\}_{j=1}^{N_{dof}}$ defined in the physical space according to Bassi et al. [14] is used. Each component $w_{h,\kappa}$, $\kappa = 1, \dots, 2{+}d$ of the numerical solution $\mathbf{w}_h$ is then expressed, in terms of the elements of the global vector $\mathbf{W}$ of the unknown DOFs, as $w_{h,\kappa} = \phi_j W_{\kappa,j}$, $j = 1, \dots, N_{dof} = \prod_{m=1}^d (k+m)/m$, $\forall K \in \mathcal{K}_h$. For the sake of presentation, the set of mesh faces $\mathcal{F}_h = \mathcal{F}_h^0 \bigcup \mathcal{F}_h^b$, partitioned in the subsets of internal $\mathcal{F}_h^0$ and boundary $\mathcal{F}_h^b$ faces, is introduced. Let us further identify with $\mathcal{F}_K = \{F\}$ the set of faces F belonging to the element K, $\forall K \in \mathcal{K}_h$. Ultimately, the DG discretization of the governing equations consists in seeking, for $l = 1, \dots, 2{+}d$, the elements of $\mathbf{W}$ such that

$$\begin{aligned}\sum_{K\in\mathcal{K}_h}\int_K \phi_i\, P_{l,\kappa}\left(\mathbf{w}_h\right)\, \phi_j \frac{dW_{\kappa,j}}{dt} d\Omega - \sum_{K\in\mathcal{K}_h}\int_K \frac{\partial \phi_i}{\partial x_n} F_{l,n}\left(\mathbf{w}_h\right) d\Omega \\ + \sum_{F\in\mathcal{F}_h}\int_F [\![\phi_i]\!]\, \widehat{F}_l\left(\mathbf{w}_h^{\pm}, n_{F,n}\right) d\sigma - \sum_{K\in\mathcal{K}_h}\int_K \phi_i b_l\left(\mathbf{w}_h\right) d\Omega = 0,\end{aligned} \tag{14}$$

for $i, j = 1, \dots, N_{dof}$ and where repeated indices imply summation over the ranges $\kappa = 1, \dots, d+2$ and $n = 1, \dots, d$. In Eq. (14) $\widehat{\mathbf{F}}$ is the numerical flux function that replaced the convective physical flux $\mathbf{F}$ in normal direction and $\{n_{F,i}\}_{i=1}^d$ is the uniquely defined unit vector normal to F. The jump operator $[\![\circ]\!] = \circ^+ - \circ^-$ is also introduced to account for the discontinuities of functions at each mesh face $F \in \partial K^+ \cap \partial K^-$. Concerning the numerical flux function, Tadmor [25–27] provided the following analytical condition

$$[\![\mathbf{v}_h^\top]\!]\, \widehat{\mathbf{F}}(\mathbf{v}_h^{\pm}, n_{F,i}) - [\![\psi_i(\mathbf{v}_h)]\!] n_{F,i} = 0, \tag{15}$$

entailing a global entropy space conservation property. This is strictly true provided the adoption of entropy-conserving fluxes, such as the Ismail and Roe formulation [28]. On the other hand, choosing an entropy-stable flux, e.g. the Godunov flux [29], the above relation holds in its inequality form ($\geq$) with the interface discontinuities acting as spatial sources that contribute as a negative rate of variation of S.

Note that from here on, the source term $\mathbf{b}$ is excluded for brevity as it does not affect the presentation of the entropy projection and the Direct Enforcement of Entropy Balance.

3.1. Entropy projection

Before discussing the implementation of the DEEB approach, the projection of variables operator is presented here. The matrix $\mathbf{P}(\mathbf{w}_h)$ significantly affects the amount of operations required by the numerical solution procedure; this is due to the introduced modification to the identity structure of the elemental mass matrix in Eq. (14). To avoid its assembly and inversion, while preserving the entropy conservation/stability property obtained from the direct use of entropy variables, the "entropy projection" strategy from

Chan et al. [15] is investigated. This approach seeks the solution for the conservative variables while evaluating the spatial discrete operators with the element-wise L_2-projection of the conservative variables on the entropy ones. In this way Eq. (14) can be rewritten as

$$\sum_{K\in\mathcal{K}_h}\int_K \boldsymbol{\Phi}_h{}^\top \frac{\partial \mathbf{q}_h}{\partial t}d\Omega - \sum_{K\in\mathcal{K}_h}\int_K \left(\frac{\partial \boldsymbol{\Phi}_h}{\partial x_i}\right)^\top \mathbf{F}_i\left(\mathbf{v}_h^*\right)d\Omega + \sum_{F\in\mathcal{F}_h}\int_F [\![\boldsymbol{\Phi}_h^\top]\!]\,\widehat{\mathbf{F}}\left(\mathbf{v}_h^{*\pm}, n_{F,i}\right)d\sigma = 0, \tag{16}$$

where $i = 1,\dots,d$. As a result a standard conservative method (for mass, total energy and momentum) is retained, whereas the resolution for an arbitrary set $\mathbf{w}_h$ would introduce a relaxation of such property to the limit of $\Delta t \to 0$, see Sec. 4.1. The entropy projection can be considered as the best polynomial approximation of $\mathbf{v}_h$, which would naturally be rational for $\mathbf{q}_h \in \mathbb{P}_d^k(\mathcal{K}_h)$. Operatively, the entropy projection is given by

$$\int_K \boldsymbol{\Phi}_h^\top \mathbf{v}_h^* d\Omega = \int_K \boldsymbol{\Phi}_h^\top \mathbf{v}\left(\mathbf{q}_h\right) d\Omega, \tag{17}$$

that reduces, thanks to the orthonormality of the shape functions, in computing the degrees of freedom (DOFs) for the projected $\mathbf{v}_h^*$

$$V_{\kappa,j}^* = \int_K \phi_j v_\kappa\left(\mathbf{q}_h\right) d\Omega, \tag{18}$$

where $\kappa = 1,\dots,d+2$ and $j = 1,\dots,N_{dof}$.

3.2. Direct enforcement of entropy balance

In a general DG framework, the evolution equation for the entropy can be derived by substituting the shape functions $\boldsymbol{\Phi}_h$ with $\mathbf{v}_h^*$ in Eq. (14), i.e.

$$\sum_{K\in\mathcal{K}_h}\int_K \mathbf{v}_h^{*\top}\mathbf{P}\left(\mathbf{w}_h\right)\frac{\partial \mathbf{w}_h}{\partial t}d\Omega - \sum_{K\in\mathcal{K}_h}\int_K \frac{\partial \mathbf{v}_h^{*\top}}{\partial x_i}\mathbf{F}_i\left(\mathbf{w}_h\right)d\Omega + \sum_{F\in\mathcal{F}_h}\int_F [\![\mathbf{v}_h^{*\top}]\!]\,\widehat{\mathbf{F}}\left(\mathbf{w}_h{}^\pm, n_{F,i}\right)d\sigma = 0. \tag{19}$$

From this, exploiting the relations

$$\sum_{K\in\mathcal{K}_h}\int_K \mathbf{v}_h^{*\top}\mathbf{P}\left(\mathbf{w}_h\right)\frac{\partial \mathbf{w}_h}{\partial t}d\Omega = \sum_{K\in\mathcal{K}_h}\int_K \frac{\partial \mathcal{S}\left(\mathbf{w}_h\right)}{\partial \mathbf{w}_h}\frac{\partial \mathbf{w}_h}{\partial t}d\Omega, \tag{20}$$

$$\sum_{K\in\mathcal{K}_h}\int_K \frac{\partial \mathbf{v}_h^{*\top}}{\partial x_i}\mathbf{F}_i(\mathbf{w}_h)d\Omega = \sum_{K\in\mathcal{K}_h}\int_K \frac{\partial \psi_i(\mathbf{w}_h)}{\partial x_i}d\Omega = \sum_{F\in\mathcal{F}_h}\int_F [\![\psi_i(\mathbf{w}_h)]\!]n_{F,i}\,d\sigma, \tag{21}$$

$$\sum_{F\in\mathcal{F}_h}\int_F [\![\mathbf{v}_h^{*\top}]\!]\,\widehat{\mathbf{F}}\left(\mathbf{w}_h{}^\pm, n_{F,i}\right)d\sigma = \sum_{F\in\mathcal{F}_h}\int_F [\![\mathbf{v}_h(\mathbf{w}_h)^\top]\!]\,\widehat{\mathbf{F}}\left(\mathbf{w}_h{}^\pm, n_{F,i}\right)d\sigma, \tag{22}$$

the following *implicitly resolved* entropy equation may be obtained

$$\sum_{K\in\mathcal{K}_h}\int_K \frac{\partial \mathcal{S}\left(\mathbf{w}_h\right)}{\partial \mathbf{w}_h}\frac{\partial \mathbf{w}_h}{\partial t}d\Omega + \sum_{F\in\mathcal{F}_h}\int_F \left[[\![\mathbf{v}_h(\mathbf{w}_h)^\top]\!]\,\widehat{\mathbf{F}}\left(\mathbf{w}_h{}^\pm, n_{F,i}\right) - [\![\psi_i(\mathbf{w}_h)]\!]n_{F,i}\right]d\sigma = 0. \tag{23}$$

As long as the numerical flux satisfies Tadmor's condition of Eq. (15), entropy is conserved/dissipated over time by Eq. (23). However, the identities (20)− (22) generally hold true only in a continuous setting, and the passage from Eq. (19) to Eq. (23) is exact in the discrete framework only if the following conditions are met:

1. the entropy variables are used as the working set so that $\mathbf{v}(\mathbf{w}_h)$ and $\mathbf{v}_h^*$ are "interchangeable";
2. the time discretized LHS of Eq. (20) results in a "correct" discretization of the time derivative of $\mathcal{S}$;
3. all the integrals are evaluated exactly.

The first condition requirement is a direct consequence of ensuring for Eq. (21) the validity of Eq. (9) at the discrete level, which is true when both $\mathbf{v}_h$ and $\mathbf{F}_i$ are computed as a function of the same set of (polynomial) variables. As the term $\partial \mathbf{v}_h^*/\partial x_i$ appears on the LHS of Eq. (21), the sole viable option is to set $\mathbf{w}_h = \mathbf{v}_h = \mathbf{v}_h^*$. Similarly, for Eq. (22) to be satisfied the identity $\mathbf{v}_h^* = \mathbf{v}_h(\mathbf{w}_h)$ should hold at each Gauss point.

An analogous reasoning applies to the second condition. The LHS of Eq. (20) is a "correct" discretization of the time derivative of $\mathcal{S}(\mathbf{w}_h)$ if and only if

$$\mathbf{v}_h^{*\top}\frac{\partial \mathbf{q}(\mathbf{w}_h)}{\partial \mathbf{w}_h}\frac{\partial \mathbf{w}_h}{\partial t} = \frac{\partial \mathcal{S}(\mathbf{w}_h)}{\partial \mathbf{w}_h}\frac{\partial \mathbf{w}_h}{\partial t}, \tag{24}$$

i.e.

$$\mathbf{v}_h^{*\top} = \frac{\partial \mathcal{S}(\mathbf{w}_h)}{\partial \mathbf{q}(\mathbf{w}_h)} \tag{25}$$

so that the chain rule

$$\frac{\partial \mathcal{S}(\mathbf{w}_h)}{\partial \mathbf{w}_h} = \frac{\partial \mathcal{S}(\mathbf{w}_h)}{\partial \mathbf{q}(\mathbf{w}_h)} \frac{\partial \mathbf{q}(\mathbf{w}_h)}{\partial \mathbf{w}_h}, \tag{26}$$

holds at the discrete level. Together with the obvious choice $\mathbf{w}_h = \mathbf{v}_h$, this requirement is also fulfilled by the conservative set $\mathbf{w}_h = \mathbf{q}_h$, where $\mathbf{P}(\mathbf{w}_h) = \mathbf{I}$.

Indeed, Eq. (17) makes the temporal discretization of the unsteady term $\int_K \mathbf{v}_h^{*\top}(\Delta \mathbf{q}_h/\Delta t)d\Omega$ equivalent to $\int_K \mathbf{v}(\mathbf{q}_h)^\top(\Delta \mathbf{q}_h/\Delta t)d\Omega$, with $\mathbf{v}(\mathbf{q}_h)^\top = \partial \mathcal{S}(\mathbf{q}_h)/\partial \mathbf{q}_h$, as $\Delta \mathbf{q}_h$ lays in the polynomials space of the solutions. In other words, the conservative set of variables provides a reliable approximation of the entropy time derivative in the *implicitly solved* entropy equation. It is worth noting that exact evaluation of the L_2-projection is not mandatory, as this result remains valid when both sides of Eq. (17) are consistently evaluated using the same Gauss quadrature rule.

In general, a time-integration scheme guarantees the full conservation (using the entropy-conserving flux) of $\mathcal{S}(\mathbf{w}_h)$ only when

$$\frac{\partial \mathcal{S}(\mathbf{w}_h)}{\partial \mathbf{w}_h} \frac{\Delta \mathbf{w}_h}{\Delta t} = \frac{\Delta \mathcal{S}(\mathbf{w}_h)}{\Delta t}, \tag{27}$$

and hence only as $\Delta t \to 0$ a general scheme becomes entropy conserving and stable, provided an appropriate numerical flux is chosen. It is worth noting that a similar phenomenon impacts the conservation of "standard" quantities like mass, energy, and momentum when the discretized scheme incorporates the Jacobian matrix $\mathbf{P}(\mathbf{w}_h)$ of the change of variables, as illustrated in Eq. (14). In the literature, there are few exceptions of fully discrete entropy conservative schemes such as, for example, relaxation Runge-Kutta methods [30] and the Generalized Crank-Nicolson [19,31] later introduced in Sec. 3.3.

Finally, the last condition highlights that, in the discrete setting, the use of the Gauss theorem to pass from the volume to the surface integral in Eq. (21) is only valid if all the integrals are computed exactly. From a practical perspective, this translates into the need for *over-integration* using appropriate quadrature rules, cf. [13]. Unfortunately, over-integration implies using quadrature rules with a very high degree of exactness, not known a priori, dramatically impacting the computational cost.

A very appealing strategy to avert the use of entropy variables and *over-integration* while preserving entropy conservation/stability properties is represented by the Direct Enforcement of Entropy Balance (DEEB) [1,16].

This approach adds a correction term to the governing equations that formally results from an optimization problem. In particular, its optimal nature lies in its minimization of the norm of the correction associated with each degree of freedom.

With the inclusion of the DEEB term, Eq. (16) becomes

$$\begin{aligned}
&\sum_{K\in\mathcal{K}_h} \int_K \mathbf{\Phi}_h^\top \mathbf{P}\left(\mathbf{w}_h\right) \frac{\partial \mathbf{w}_h}{\partial t} d\Omega - \sum_{K\in\mathcal{K}_h} \int_K \left(\frac{\partial \mathbf{\Phi}_h}{\partial x_i}\right)^\top \mathbf{F}_i\left(\mathbf{w}_h\right) d\Omega \\
&+ \sum_{F\in\mathcal{F}_h} \int_F [\![\mathbf{\Phi}_h^\top]\!] \widehat{\mathbf{F}}\left(\mathbf{w}_h{}^{\pm}, n_{F,i}\right) d\sigma + \sum_{K\in\mathcal{K}_h} \alpha_K \frac{\int_K \mathbf{\Phi}_h^\top \overline{\mathbf{v}}_h^* \, d\Omega}{\int_K \overline{\mathbf{v}}_h^{*\top} \overline{\mathbf{v}}_h^* \, d\Omega} = 0,
\end{aligned} \tag{28}$$

where α_K is the elemental correction factor and $\overline{\mathbf{v}}_h^* = \mathbf{v}_h^* - \mathbf{v}_{h,0}^*$, with $\mathbf{v}_{h,0}^*$ being the mean value of $\mathbf{v}_h^*$. In Eq. (28) the role of the term $\left(\int_K \mathbf{\Phi}_h^\top \overline{\mathbf{v}}_h^* \, d\Omega\right) / \left(\int_K \overline{\mathbf{v}}_h^{*\top} \overline{\mathbf{v}}_h^* \, d\Omega\right)$ is to distribute the correction as uniformly as possible among all the DOFs. In the present implementation, taking advantage of the orthonormality of the shape functions, this contribution to the correction term can be greatly simplified as

$$\frac{\int_K \phi_i \phi_j \overline{V}_{\kappa,j}^* d\Omega}{\int_K \phi_j \overline{V}_{s,j}^* \phi_\iota \overline{V}_{s,\iota}^* d\Omega} = \frac{\overline{V}_{\kappa,i}^*}{\overline{V}_{s,\iota}^* \overline{V}_{s,\iota}^*}, \tag{29}$$

with $\kappa, s = 1, \dots, 2+d$ and $i, j, \iota = 1, \dots, N_{dof}$, where repeated indices imply summation. To put it in clearer terms: *i*) the correction is applied to all equations except for those associated to the first test function, for which $\overline{V}_{k,1}^* = 0$, guaranteeing the conservation of mean flow quantities; *ii*) the numerator of Eq. (29) includes the DOF associated with the test function of the equation to be solved, while the denominator is composed by the global sum of all the squared DOFs. When dealing with solutions where the out-of-mean degrees of freedom are null, the correction term in Eq. (29) becomes an indeterminate form. Since in this situation α_K should converge to zero, in the code implementation the correction is omitted when the denominator is lower than 10^{-14}.

Following the work of Abgrall [1] the elemental correction term can be expressed in explicit form as

$$\alpha_K = \sum_{F\in\mathcal{F}_K} \int_F \left[\mathbf{v}(\mathbf{w}_h)^\top - \mathbf{v}_h^{*\top}\right] \widehat{\mathbf{F}}\left(\mathbf{w}_h{}^{\pm}, n_{F,i}\right) d\sigma + \int_K \left(\frac{\partial \mathbf{v}_h^*}{\partial x_i}\right)^\top \mathbf{F}_i\left(\mathbf{w}_h\right) d\Omega - \sum_{F\in\mathcal{F}_K} \int_F \psi_i(\mathbf{w}_h) n_{F,i} \, d\sigma, \qquad \forall K \in \mathcal{K}_h. \tag{30}$$

By adding and subtracting $\int_K \left(\partial \mathbf{v}(\mathbf{w}_h)/\partial x_i\right)^\top \mathbf{F}_i\left(\mathbf{w}_h\right) d\Omega = \int_K \partial \psi_i(\mathbf{w}_h)/\partial x_i d\Omega$ to the RHS of Eq. (30) the different sources of error that α_K is delegated to correct can be distinguished

$$\begin{aligned}\alpha_K = & \underbrace{\sum_{F\in\mathcal{F}_K}\int_F \left[\mathbf{v}(\mathbf{w}_h)^\top - \mathbf{v}_h^{*\top}\right]\widehat{\mathbf{F}}\left(\mathbf{w}_h{}^{\pm}, n_{F,i}\right)d\sigma}_{\text{projection error of } \mathbf{v} \text{ in the surface integral}} \\ & + \underbrace{\int_K \left[\left(\frac{\partial \mathbf{v}_h^*}{\partial x_i}\right)^\top - \left(\frac{\partial \mathbf{v}(\mathbf{w}_h)}{\partial x_i}\right)^\top\right]\mathbf{F}_i\left(\mathbf{w}_h\right)d\Omega}_{\text{projection error of } \mathbf{v} \text{ in the volume integral}} \\ & + \underbrace{\int_K \frac{\partial \psi_i\left(\mathbf{w}_h\right)}{\partial x_i}d\Omega - \sum_{F\in\mathcal{F}_K}\int_F \psi_i(\mathbf{w}_h) n_{F,i}\, d\sigma}_{\text{integration error}}.\end{aligned} \tag{31}$$

Although the definition of α_K applies to any possible set of working variables, the contribution of the two distinct sources of error depends on the specific set under consideration. If the convective flux function is evaluated with $\mathbf{v}$ in polynomial form, i.e. if the entropy variables are the working set, then the first two terms of Eq. (31) cancel out as $\mathbf{v}(\mathbf{v}_h) \equiv \mathbf{v}_h^*$. In this case α_K is $\mathcal{O}(h^{2k+d})$ upon adoption of Gauss quadrature formulas able to integrate exactly polynomials of order $2k$, as it is usually done in the present modal DG framework. When integrals are computed with quadrature rules of very high accuracy, the last term in Eq. (31) also becomes null. For conservative variables used as working set, this is possible even without over-integration since the mass flux ψ_i is a polynomial function of degree k. In this case α_K is a $\mathcal{O}(h^{k+d})$ term, due to the projection error.

In Ref. [8] Chen and Shu question the reliability of the correction term due to the denominator of the spreading term being of order $\mathcal{O}(h^2)$, despite α_K being of higher order. They express a concern that the correction could potentially compromise the consistency of the method. However, Chen and Shu were able to determine the optimal order of convergence, which was found to be $k+1$ for the two-dimensional Burger equation through numerical analysis. In Sec. 4.1, we will also verify the optimal convergence rate for the Euler equations. To demonstrate the conservation/dissipation of entropy guaranteed by DEEB, the explicit correction is added to Eq. (19)

$$\begin{aligned}&\sum_{K\in\mathcal{K}_h}\int_K \mathbf{v}_h^{*\top}\mathbf{P}\left(\mathbf{w}_h\right)\frac{\partial \mathbf{w}_h}{\partial t}d\Omega - \sum_{K\in\mathcal{K}_h}\int_K \left(\frac{\partial \mathbf{v}_h^*}{\partial x_i}\right)^\top \mathbf{F}_i\left(\mathbf{w}_h\right)d\Omega \\ &+ \sum_{F\in\mathcal{F}_h}\int_F \left[\!\left[\mathbf{v}_h^{*\top}\right]\!\right]\widehat{\mathbf{F}}\left(\mathbf{w}_h{}^{\pm}, n_{F,i}\right)d\sigma + \sum_{K\in\mathcal{K}_h}\alpha_K \frac{\int_K \mathbf{v}_h^{*\top}\overline{\mathbf{v}}_h^*\, d\Omega}{\int_K \overline{\mathbf{v}}_h^{*\top}\overline{\mathbf{v}}_h^*\, d\Omega} = 0.\end{aligned} \tag{32}$$

Since $\left(\int_K \mathbf{v}_h^{*\top}\overline{\mathbf{v}}_h^*\, d\Omega\right) / \left(\int_K \overline{\mathbf{v}}_h^{*\top}\overline{\mathbf{v}}_h^*\, d\Omega\right) = 1$, Eq. (32) reduces to Eq. (23) and if the numerical flux satisfies Tadmor's condition the scheme is entropy conserving/stable, net of the possible error introduced in the time integration. In conclusion, the DEEB algorithm can ideally fix the behavior of the *implicitly resolved* entropy equation for any choice of $\mathbf{w}_h$. However, α_K only corrects the spurious entropy production/dissipation of the spatial operators and does not act on the unsteady term of Eq. (19), so that only for $\mathbf{w}_h = \mathbf{v}_h$ or $\mathbf{w}_h = \mathbf{q}_h$ the term $\sum_{K\in\mathcal{K}_h}\int_K \left(\partial \mathcal{S}(\mathbf{w}_h)/\partial t\right) d\Omega$ is "correctly" represented in Eq. (32). For this reason, we focus on these two sets together with a "mixed" formulation given by the "entropy projection" discussed in Sec. 3.1, while the use of the DEEB on other sets $\mathbf{w}_h$ is left for future investigations.

3.3. Time integration

By numerically computing the volume and surface integrals in Eq. (28) and assembling all the elemental contributions, the system of Ordinary Differential Equations (ODEs) governing the evolution in time of the discrete solution can be written as

$$\mathbf{M}_\mathbf{P}(\mathbf{W})\frac{d\mathbf{W}}{dt} + \mathbf{R}(\mathbf{W}) = \mathbf{0}, \tag{33}$$

where the global matrix $\mathbf{M}_\mathbf{P}$ is block diagonal and responsible for coupling the DOFs within each element upon the choice of non-identity transformation $\mathbf{q} \mapsto \mathbf{w}(\mathbf{q})$. The nonlinear system of ODEs (33) is here integrated in time using both an explicit Strong Stability Preserving Runge-Kutta (SSPRK) [18] scheme and the implicit "entropy conserving" Generalized Crank-Nicolson (GCN) [32] method. As far as the explicit scheme is concerned, the five-stages third order scheme of Spiteri and Ruuth [18] is used

$$\begin{aligned}\mathbf{Y}_i &= \mathbf{W}^n + \Delta t \sum_{j=1}^{i-1} a_{ij} \tilde{\mathbf{R}}\left(\mathbf{Y}_j\right), \\ \mathbf{W}^{n+1} &= \mathbf{W}^n + \Delta t \sum_{j=1}^{s} b_j \tilde{\mathbf{R}}\left(\mathbf{Y}_j\right),\end{aligned} \tag{34}$$

with

$$\tilde{\mathbf{R}}(\mathbf{W}) = \mathbf{M}_{\mathbf{P}}(\mathbf{W})^{-1}\mathbf{R}, \tag{35}$$

where s is the number of stages and a_{ij}, b_j are the real coefficients reported in [18]. For the sake of compactness, we refer to this scheme as SSPRK35. No control over the entropy production/dissipation, nor any positivity-preserving limiter for the thermodynamic quantities is enforced for the time advancement of the solution. Notice that the possible dependence of $\mathbf{M}_{\mathbf{P}}$ on $\mathbf{W}$ implies, even for the explicit scheme, a matrix assembly at each time step and a significant overhead due to its inversion at each stage, whose number of operations scales as k^{3d}. For this reason, the use of the conservative variables for the working set, possibly in an "entropy projection" context, is very enticing for the computational cost of the method as $\mathbf{M}_{\mathbf{P}}$ reduces to the identity and is not involved in the solution process.

Regarding the GCN, as this method has been the focus of a dedicated paper [19], only a brief summary of the algorithm is reported here. The scheme may be thought of as the entropy projection method of Eq. (16), in which $\mathbf{v}_h^*$ is computed from $\int_K \mathbf{\Phi}_h^\top \mathbf{v}_h^* d\Omega = \int_K \mathbf{\Phi}_h^\top \mathbf{v}^{n+1/2} d\Omega$, where the intermediate entropy state $\mathbf{v}^{n+1/2}(\mathbf{q}_h^n, \mathbf{q}_h^{n+1})$ is the nonlinear vector function derived by Gouasmi et al. [31] to satisfy the constraint

$$(\mathbf{v}^{n+1/2})^\top(\mathbf{q}_h^{n+1} - \mathbf{q}_h^n) = \mathcal{S}(\mathbf{q}_h^{n+1}) - \mathcal{S}(\mathbf{q}_h^n). \tag{36}$$

The $\mathbf{v}^{n+1/2}$ function bypasses the need for numerical quadrature as requested in the seminal work of LeFloch et al. [32]. Thanks to the last identity, the discretized unsteady term

$$\int_K \mathbf{\Phi}_h^\top \frac{\mathbf{q}_h^{n+1} - \mathbf{q}_h^n}{\Delta t} d\Omega, \tag{37}$$

reduces, in the *implicitly solved* entropy equation, to

$$\int_K \frac{\mathcal{S}(\mathbf{q}_h^{n+1}) - \mathcal{S}(\mathbf{q}_h^n)}{\Delta t} d\Omega, \tag{38}$$

which is a conservative discretization of the time derivative of the entropy function.

Using the DEEB correction together with the GCN, the resulting discretization is entropy conserving/stable at the discrete level both in space and time. The scheme is also conservative in a "standard" sense, see Sec. 3.1, being the time derivative expressed for $\mathbf{q}_h$.

To conclude, for both temporal integrators the time step size is evaluated as $\Delta t = \min_{K\in\mathcal{K}_h}(\Delta t_K)$, where $\Delta t_K = \mathrm{CFL}(h_K/\lambda_K)$ is calculated by assuming a fixed value (for the whole simulation) of the CFL number. The elemental convective eigenvalue is computed from the cell mean values as $\lambda_K = \sqrt{\gamma p_{K,0}/\rho_{K,0}} + |\mathbf{u}_{K,0}|$ and the characteristic element size is defined as $h_K = d|K|/|\partial K|$, where $|K|$ is the cell area/volume, $|\partial K|$ is the measure of its boundary and d is the number of geometric dimensions.

4. Numerical results

The numerical experiments presented in this section consider the solution of the Euler equations using as unknown: *i*) the set of conservative variables, *cv*; *ii*) the set of entropy variables, *ev*; *iii*) the set of conservative variables but using the projected entropy ones for the assembly of the discrete operators, *cv2ev*. The influence of performing *over-integration* (OI) or not (NOI) and of directly enforcing the entropy balance (DEEB) are evaluated for all the sets. In the following, for the sake of synthesis in presenting the results, the superscripts NOI, OI or DEEB are applied to the acronym identifying the strategy, e.g. $cv2ev^{\mathrm{DEEB}}$. For NOI simulations the degree of exactness of the Gaussian quadrature formulae used to evaluate integrals in Eqs. (17), (28) and (30) is $k^q = 2k\,m + j$, where k is the polynomial degree of the numerical approximation, m is the geometrical order of the element and j is the polynomial degree of the Jacobian of the mapping (from reference to physical space). For triangles one has $j = 2m - 2$, while $j = 2m - 1$ and $j = m - 1$ for quadrangles and for edges, respectively. This definition of k^q corresponds to $\lceil (k^q + 1)/2 \rceil^d$ quadrature points and guarantees the exact volume and surface integration of polynomials of order $2k$ independently from the element shape. For the OI simulations the degree of exactness is arbitrarily set equal to $k^q = 30$. Such over-integration level seems excessive, especially at the lower polynomial orders. However, the convective fluxes are rational functions of both the conservative and the entropy variable sets, thus it is impossible to know a priori the k^q value required to reduce the integration error to machine precision. For this reason, for example, the entropy conservation/stability of the OI *ev* computations is ensured only for very large values of k^q. This is particularly true when the flow field is characterized by large variations of the thermodynamics variables, as in several of our test cases. The fixed value $k^q = 30$ is chosen for the sake of simplicity and is justified by the fact that it doubles the degree of exactness associated with OI with respect to NOI at the highest polynomial $k = 7$. This means that cubic functions of the variables are exactly

Table 1
IVC – summary of the grids used for the spatial discretization of the computational domain, composed by rectangular (rect), triangular (tri) and curvilinear (curv) elements. rect: Cartesian meshes made of quadrangular elements; tri: unstructured meshes made of triangular cells; curv: meshes made of quadrilaterals with parabolic edges. Grid fineness is given in terms of the number of cells with edges laying on the domain boundaries in the vertical and horizontal directions.

	rect	tri	curv
coarse	25×25	10×10	25×25
medium	50×50	20×20	50×50
fine	100×100	40×40	100×100

integrated, which is more or less the choice reported in several papers for the compressible Euler equations [9,33–36]. In any case, the impact of this choice on the performance of OI strategies is basically irrelevant. *Over-integrated* computations are in fact provided for reference only, as their computational cost is always much higher than the other strategies and precludes them from being an option for practical applications.

The performance of the method is tested by simulating several two-dimensional inviscid flow problems and focusing on *i*) the accuracy of the solution, *ii*) the conservation of entropy, mass and total energy, *iii*) the robustness and computational efficiency.

4.1. Convergence and conservation properties

To monitor the convergence and the conservation properties of the schemes, the following error measures are defined:

$$\eta(\circ, \bullet_{ref}) = (\Omega_h)^{-1/2} ||\circ - \bullet_{ref}||_{L_2}, \tag{39}$$

$$\varepsilon(\circ, \bullet_{ref}) = \Omega_h^{-1} \left| \int_{\Omega_h} \circ \, d\Omega - \int_{\Omega_h} \bullet_{ref} \, d\Omega \right|, \tag{40}$$

where $\circ$ and $\bullet_{ref}$ are the numerical and the reference solutions, respectively. The L_2–projection of the initial solution on the DG polynomial space is used as the reference state.

The test case selected as a benchmark is the 2D Isentropic Vortex Convection (IVC) problem, Refs. [37–39]. In this test case, an inviscid vortex, initially located at the center of a square domain, is convected diagonally. The doubly-periodic domain justifies the choice of this flow case, as it ensures that the integral quantity ρs is conserved/dissipated if the entropy conservation/stability property holds. The initial flow conditions are given, in dimensionless form, by

$$\begin{aligned} u_1 &= \sqrt{\gamma} - \frac{\alpha}{2\pi} \left(x_2 - x_{2_0} \right) e^{\beta(1-r^2)}, \\ u_2 &= \sqrt{\gamma} + \frac{\alpha}{2\pi} \left(x_1 - x_{1_0} \right) e^{\beta(1-r^2)}, \\ p &= \rho^\gamma, \\ \rho &= \left(1 - \frac{\alpha^2(\gamma-1)}{16\beta\gamma\pi^2} e^{2\beta(1-r^2)} \right)^{\frac{1}{\gamma-1}}, \end{aligned} \tag{41}$$

corresponding to a free stream Mach number equal to $\sqrt{2}$ and where $(x_{1_0}, x_{2_0}) = (5,5)$, $r = ||x_i - x_{i_0}||_{L_2}$, while the parameters determining the strength of the vortex are $\alpha = 5$ and $\beta = 1/2$. The $[0,10]^2$ square computational domain is discretized by means of meshes of different type and increasing fineness, see Fig. 1, using $k = 3,4$ DG approximations. To demonstrate the flexibility of the solver, the set of grids includes: *i*) Cartesian meshes made of quadrangular elements, *ii*) unstructured meshes made of triangular cells; *iii*) meshes made of quadrilaterals with parabolic edges. The curved edges are obtained by randomly shifting in the orthogonal direction the edge midpoints of a Cartesian mesh. The complete list of grids is collected in Table 1. Simulations are performed up to a nondimensional final time $t_f = 10/\sqrt{\gamma}$, corresponding to one period of vortex revolution. Both the temporal schemes described in Sec. 3.3, i.e. the explicit SSPRK35 and the implicit GCN, are used to advance the solution in time with the small time step size $\Delta t = 10^{-4} t_f$ (CFL $\approx 9 \cdot 10^{-3}$). For the numerical fluxes, the entropy conserving (EC) function of Ismail and Roe [28] and the entropy stable (ES) Godunov flux [29] are employed.

The plots resulting from the spatial convergence study, which refer to the first variable of the unknown solution vector of each set of variables, are collected in Fig. 2. For all the solution strategies, the convergence rate globally follows the results of Ref. [13], being sub-optimal (k) for the EC case and optimal ($k+1$) for the ES case. The only exception is observed when using the EC flux with meshes made of triangular cells, which for the even polynomial degree $k = 4$ recovers the optimal order of convergence of DG schemes. Note that the values of the error appear different for $cv2ev^{\mathrm{DEEB}}$ and ev^{DEEB} because they are not evaluated with respect

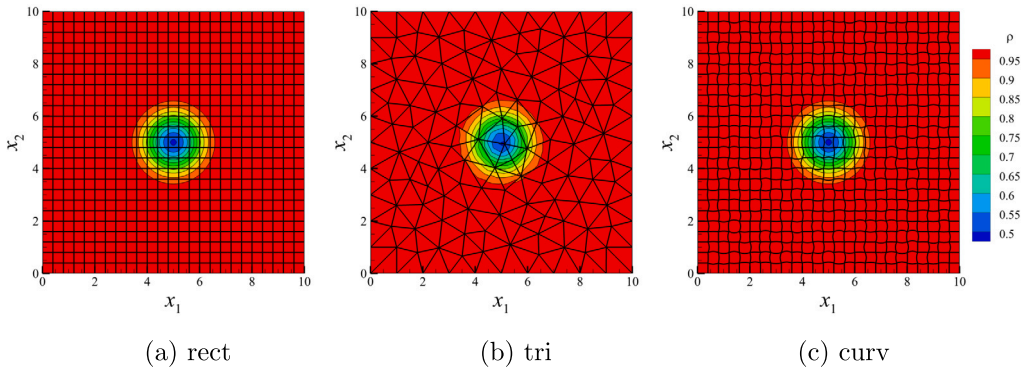


Fig. 1. IVC – $k = 4$ solution on the coarse grids at the final time. For interpretation of the colors in the figure(s) and table(s), the reader is referred to the web version of this article.

to the same variable, otherwise they would be practically identical. This convergence test demonstrates that both the use of entropy projection and DEEB do not alter the consistency of the method and the expected order of spatial convergence.

A temporal accuracy test is also performed on the coarse Cartesian grid at $k = 3$, using the SSPRK35 for the time discretization. As the scope is to highlight the DEEB ability to guarantee entropy conservation, we use the EC flux and monitor the error on the surface integral of $\rho s = -S(\gamma - 1)$. As showed in Fig. 3a, the $cv2ev$ approach coupled with over-integration or DEEB perfectly conserves mass and energy, since the non-stationary term maintains its conservative form, while the error on the entropy converges with the order of the temporal scheme. As expected, in the NOI case, the error fails to scale correctly due to the error committed in the discrete application of the Gauss theorem, as commented in Sec. 3.2. On the other hand, Fig. 3b shows that the ev strategy makes all the errors on conservation converge with the order of the temporal scheme, except for ρs in the ev^{NOI} case for the same reason as before. This behavior is expected as the governing equations in the ev case are no longer written in a conservative form due to the Jacobian matrix $\mathbf{P}(\mathbf{v}_h)$. Fig. 3c instead displays that the cv set shares the same trend as the projected entropy variables, with the only difference that the error on the entropy does not converge with the order of the temporal scheme even in the OI configuration. The reason is that here the error is related to the projection, namely to the fact $\mathbf{v}(\mathbf{q}_h) \neq \mathbf{v}_h^*$, as explained in Sec. 3.2. To summarize, when time integration introduces a negligible error, both entropy-based sets can conserve entropy, mass and energy if over-integration or DEEB is used. Otherwise, only mass and energy are conserved. Furthermore, the plots in Fig. 3 reveal that both entropy projection and DEEB do not modify the theoretical order of convergence of the temporal scheme.

Finally, interesting results are obtained from the spatial convergence analysis of the entropy conservation error. In this case time integration is performed through the GCN scheme to avoid the emergence of the entropy discretization error associated to the unsteady term. The results for the $cv2ev$ strategy presented in Fig. 4 show a perfect conservation of entropy when the EC flux function is used. When using the ES Godunov flux the conservation error decreases very rapidly as the mesh is refined, with an estimated order of $2k + 1$. This finding holds true across various time integration schemes and sets of variables. Moreover, a similar outcome for $\epsilon(\rho s, \rho s_{ref})$ is observed when the ES Rusanov flux is used. It is reasonable to expect that these results may extend to other numerical flux functions as well.

4.2. Robustness

In this section, the robustness of the method is investigated by considering: *i*) two test cases widely used in the Numerical Weather Prediction (NWP) community to assess the performance of the solvers for atmospheric dynamics; *ii*) two test cases involving the instability of an interface between flow regions with significantly different density. All the flow problems exhibit progressively smaller scales with spatial and temporal refinement but evolve through very different thermodynamic conditions. While the NWP flow problems, i.e. the colliding and rising thermal bubbles cases, involve minimal variations of thermodynamic variables, the Kelvin-Helmhotz and the Richtmyer-Meshkov instabilities push the method to conditions in which it is challenging to maintain the positiveness of thermodynamic variables. It is important to note that all simulations were performed without adding any artificial viscosity term, shock-capturing technique and/or positive-preserving limiter. The NWP test cases are simulated using both the EC and the ES flux, whereas in the flows characterized by shear instability we opt just for the ES Godunov flux due to hard numerical challenge posed by the latter. Unless otherwise specified, the SSPRK35 scheme is the one employed for the time advancement, with a constant CFL = 0.1. Under these conditions, it is clear that the higher k, the closer we are to the stability limit and the farther we get to evaluate accurately enough the integrals in the *over-integrated* case since k^q is fixed to 30.

Lastly, it is important to note that for NWP flow problems, the initial conditions and results are be presented using dimensional quantities, as is customary in the literature.

4.2.1. The 2D colliding thermal bubbles problem

This flow problem was proposed by Robert [40] and deals with the evolution of two air bubbles, one warmer and the other colder than the surrounding environment. The hot bubble is larger and lower than the cold one, and during the time evolution, they collide,

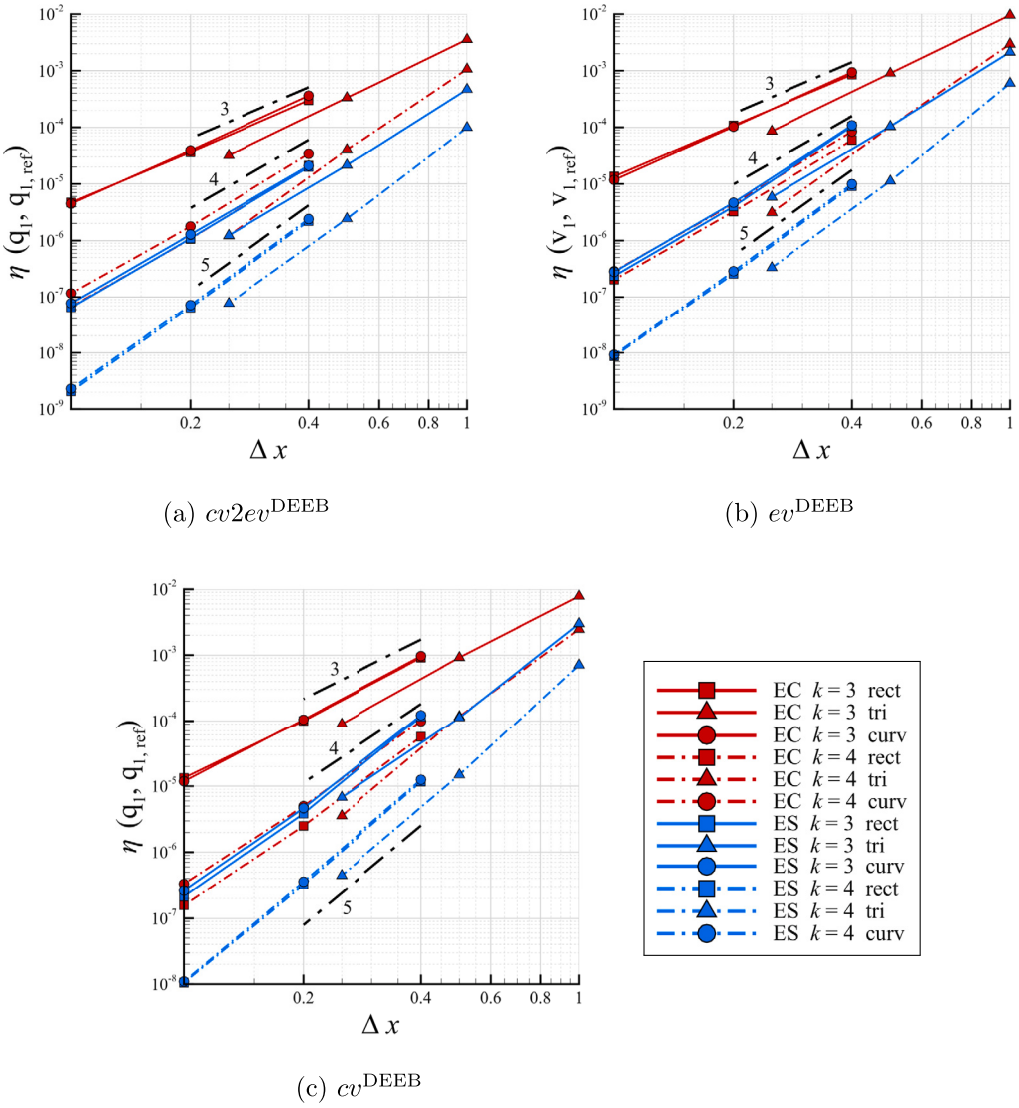


Fig. 2. IVC – spatial refinement test. Errors η on the first variable of vectors **q** (in $cv2ev$ and cv) and **v** (in ev). Simulations integrated in time via the explicit SSPRK35 with $\Delta t = 10^{-4}\, t_f$.

mix, and move toward the upper part of the computational domain. When dealing with atmospheric flows, two important variables to consider are the Exner pressure π and the potential temperature θ, defined as

$$\pi = \left(\frac{p}{p_0}\right)^{\frac{\gamma}{\gamma-1}}, \qquad \theta = \frac{T}{\pi}, \tag{42}$$

where p_0 is the reference pressure taken at $x_2 = 0$ m. As reported by Giraldo and Restelli [41], non-hydrostatic atmospheric flow solvers may implement different forms of the inviscid flow governing equations, often using π and/or θ as an unknown. In NWP, it is probably less common to directly solve for mass, momentum, and total energy conservation as it is usually done for aerodynamic problems. However, using this formulation has its advantages as it enables conservation up to machine accuracy of these quantities and facilitates the implementation of the viscous terms in conservation form [41,42].

In the Colliding Thermal Bubbles (CTB) problem, the flow instability is caused by initial perturbations $\Delta\theta_c$ and $\Delta\theta_h$ applied to a constant environmental potential temperature θ_0, i.e. $\theta(x_1, x_2) = \theta_0 + \Delta\theta_c(x_1, x_2) + \Delta\theta_h(x_1, x_2)$. The perturbation for both the cold (subscript x = c) and hot (subscript x = h) bubbles is defined as

$$\Delta\theta_{\mathrm{x}}(x_1, x_2) = \begin{cases} A_{\mathrm{x}}, & \text{if } r_{\mathrm{x}} \le a_{\mathrm{x}} \\ A_{\mathrm{x}} e^{-(r_{\mathrm{x}} - a_{\mathrm{x}})^2 / s_{\mathrm{x}}^2}, & \text{if } r_{\mathrm{x}} > a_{\mathrm{x}}, \end{cases} \tag{43}$$

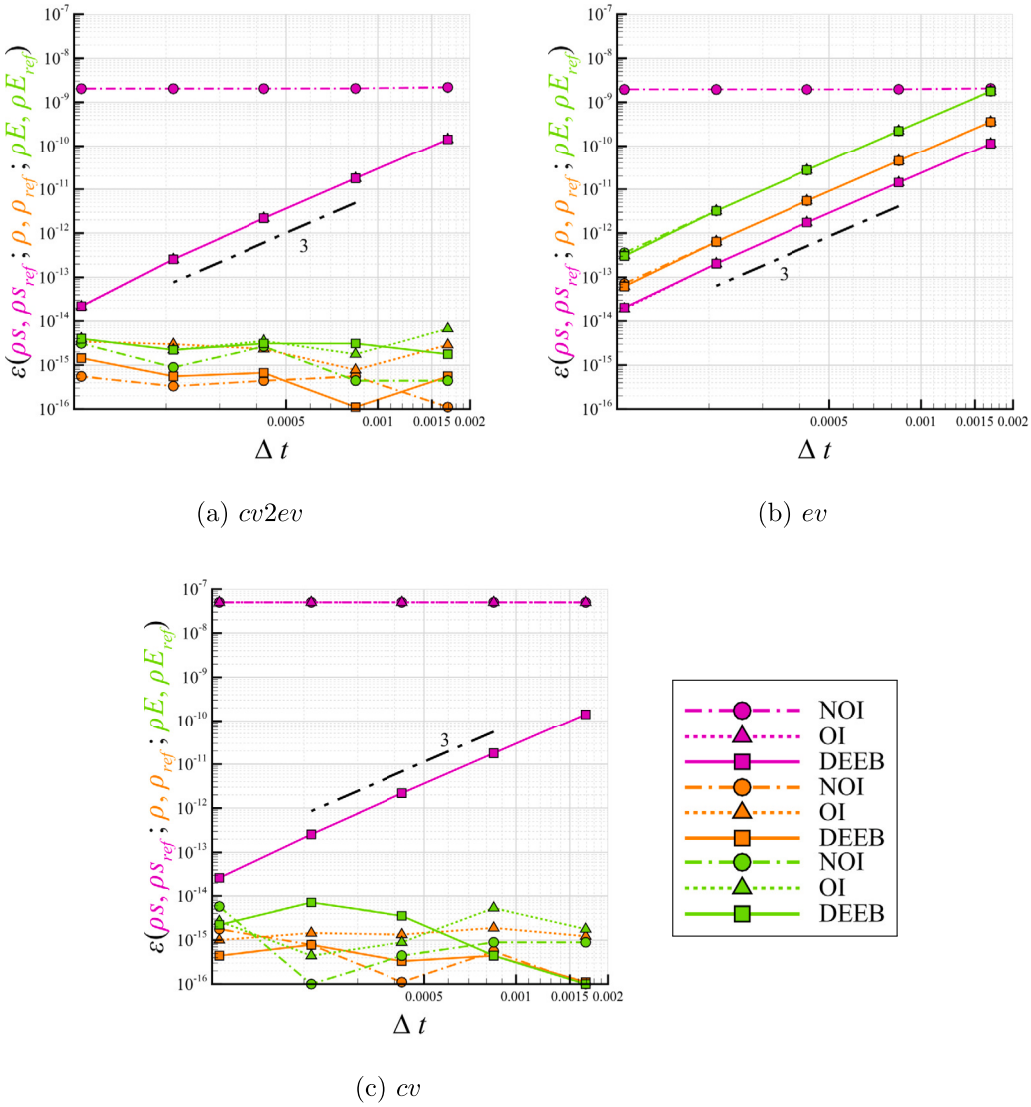


Fig. 3. IVC – temporal refinement test. Errors on the conservation of entropy, mass and energy. Simulations performed using $k = 3$ on the Cartesian 25×25 mesh, the explicit SSPRK35 scheme and the EC numerical flux.

where $r_x = \sqrt{\left(x_1 - x^0_{x,1}\right)^2 + \left(x_2 - x^0_{x,2}\right)^2}$ and $(x^0_{x,1}, x^0_{x,2})$ is the initial position of the center of the bubble. The flow problem parameters are $A_c = -0.15$ K, $a_c = 0$ m, $s_c = 50$ m, $(x^0_{c,1}, x^0_{c,2}) = (560, 640)$ m, $A_h = 0.5$ K, $a_h = 150$ m, $s_h = 50$ m and $(x^0_{h,1}, x^0_{h,2}) = (500, 300)$ m. The background potential temperature is set to 300 K while the initial pressure distribution is computed as $p(x_2) = p_0 \pi(x_2)^{\gamma/(\gamma-1)}$ where

$$\pi(x_2) = 1 - \frac{(\gamma - 1)g}{\gamma R \theta_0} x_2, \tag{44}$$

in which $p_0 = 100\,000$ Pa and $R = 287.058$ kJ/kg K. By defining a reference velocity $U_0 = \sqrt{R\theta_0}$ and Froude number $Fr_0 = U_0/\sqrt{gL_x}$, where L_x is the horizontal extension of the domain, the Exner pressure can be expressed as

$$\pi(x_2) = 1 - \frac{\gamma - 1}{\gamma} \left(\frac{1}{Fr_0^2} \right) \frac{x_2}{L_x}, \tag{45}$$

where $Fr_0 = 2.96286$. Finally, the initial temperature distribution is computed as $T(x_1, x_2) = \pi(x_2)\theta(x_1, x_2)$ and velocity components are set to zero. The domain extension is $[0, 1000] \times [0, 1000]$ m and is discretized with a Cartesian mesh made of 20×20 cells ($\Delta x_{1,2} = 50$ m).

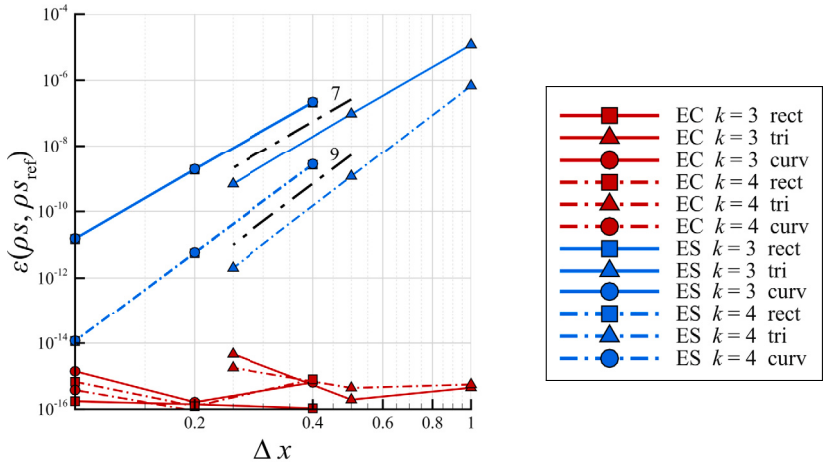


Fig. 4. IVC – spatial refinement test for the $cv2ev$ strategy. Errors on the conservation of entropy. Simulations integrated in time via the implicit GCN with $\Delta t = 10^{-4}\, t_f$.

Simulations for the $k = \{3,4,5,6,7\}$ approximations are integrated in time up to $t_f = 600$ s. Fig. 5 shows the evolution in time of the bubbles for the $k = 7$ solution computed using the $cv2ev^{\mathrm{DEEB}}$ approach and the ES Godunov flux. The overall pattern compares qualitatively well with the results of Robert [40] and Yelash et al. [43], while the development of the small-scale structures strongly depends on the spatial resolution, as commented by Abdi et al. [44], and as it can be observed by increasing the polynomial degree of the approximation for the given mesh, see Fig. 6. Moreover, some oscillating features resembling Gibbs-like phenomena may be observed in the reported fields. Clearly, rather than being related to abrupt variations of the evolving quantities, these oscillations are caused by the high-order and discontinuous character of the finite element representation. On top of this, it is relevant to point out that such oscillations are detectable only when plotting the deviation from the background quantity of interest, involving hence variations which are order of magnitudes smaller than the reference value. Indeed, minimal variations of the thermodynamic variables over the computational domain characterize the flow case. According to the $k = 7$ approximation at the final time, the temperature and density vary by approximately 3% and 8%, respectively, compared to their maximum values across the domain. No robustness issues are observed for the simulations performed using the Godunov flux, regardless of the strategy used for the solution, i.e. cv, ev or $cv2ev$, and the use of DEEB or over-integration. In all the cases, the error in entropy conservation at the end time is $\varepsilon(\rho s, \rho s_{ref}) \approx 10^{-8}$.

The solution strategies manifest a remarkably different behavior when using the EC Ismail and Roe numerical flux function. Results for the $k = \{3,4,5,6,7\}$ approximations are summarized in Table 2 in terms of possible simulation crashes and errors in entropy conservation. The ev and $cv2ev$ approaches reveal identical behaviors, with the need for over-integration or DEEB to fulfill the entropy conservation over the simulation. The $\varepsilon(\rho s, \rho s_{ref}) \approx 10^{-11}$ value for the $k = 7$ simulations is due to the use of a non-entropy-conserving time integrator and scales with the order of the SSPRK35 scheme as the time step is refined. When standard integration (NOI) is used, odd-degree polynomial simulations crash, while even degrees reach the end time but with a non-negligible error on entropy conservation ($\approx 10^{-9}$). Using conservative variables as unknowns works only when combined with DEEB. Additionally, when approximating the solution with odd degrees, the value of $\varepsilon(\rho s, \rho s_{ref})$ at the end time is observed to be significantly higher than machine precision. Again, this effect is attributed to a temporal integrator that does not conserve entropy. Indeed, it has been verified that the error decreases with the order of accuracy of the temporal scheme when the time step is reduced.

The CTB flow problem shows that even when thermodynamic conditions are not severe, using entropy variables or conservative variables with entropy projection is better than directly using the conservative set. Additionally, the $cv2ev$ strategy performs similarly to ev but saves many operations as it does not require the inversion of the change-of-variable matrix, as stated in Sec. 3. This makes the $cv2ev$ approach particularly appealing for production runs.

4.2.2. The 2D rising thermal bubble problem

The flow problem deals with the evolution of an air bubble warmer than the surrounding environment. As the bubble rises, it deforms due to the velocity gradients initially generating counter-rotating trailing vortices that break up into smaller structures during evolution. Different polynomial approximations, i.e. $k = \{3,5,7\}$, are considered to showcase how high-order accuracy can better capture small-scale structures.

The most commonly used setup is the one proposed by Robert in [40]. Several authors have simulated this configuration, e.g. [45, 42,46,47], often with slightly different initial conditions, and a comprehensive review of the literature is given by Andrés-Carcasona et al. [48]. As for the CTB, in the rising thermal bubble (RTB) flow problem the flow instability originates from an initial perturbation $\Delta\theta$ applied to a constant environmental potential temperature θ_0, i.e. $\theta(x_1, x_2) = \theta_0 + \Delta\theta(x_1, x_2)$. The variable $\Delta\theta$ follows Eq. (43) where $(x_{\mathrm{x},1}, x_{\mathrm{x},2}) = (500, 260)$ m, $A_{\mathrm{x}} = 0.5$ K, $a_{\mathrm{x}} = 50$ m and $s_{\mathrm{x}} = 100$ m. In this paper, the Gaussian-shaped perturbation originally proposed by Robert [40] is used, while some other authors prefer to utilize a cosine-shaped bubble, see [41]. According to the literature, no distinct differences in the solution are to be expected [48].

Table 2
CTB – end time in seconds [red and blue] and order of magnitude of $\epsilon(\rho s, \rho s_{ref})$ [black] of the simulations using the EC Ismail and Roe flux. ⋄ if $\epsilon(\rho s, \rho s_{ref})$ is of the order of the machine precision, × if simulation crashed.

Set	k				
	3	4	5	6	7
$cv2ev^{\mathrm{OI}}$	600 ⋄	600 ⋄	600 ⋄	600 ⋄	600 ⋄
$cv2ev^{\mathrm{NOI}}$	417 ×	600 10^{-9}	426 ×	600 10^{-9}	471 ×
$cv2ev^{\mathrm{DEEB}}$	600 ⋄	600 ⋄	600 ⋄	600 ⋄	600 10^{-11}
ev^{OI}	600 ⋄	600 ⋄	600 ⋄	600 ⋄	600 ⋄
ev^{NOI}	417 ×	600 10^{-9}	426 ×	600 10^{-9}	471 ×
ev^{DEEB}	600 ⋄	600 ⋄	600 ⋄	600 ⋄	600 10^{-11}
cv^{OI}	292 ×	362 ×	291 ×	366 ×	308 ×
cv^{NOI}	274 ×	342 ×	275 ×	340 ×	297 ×
cv^{DEEB}	600 10^{-6}	600 ⋄	600 10^{-5}	600 ⋄	600 10^{-4}

Table 3
RTB – end time in seconds of the simulations using the EC Ismail and Roe flux.

Set	k		
	3	5	7
$cv2ev^{\mathrm{OI}}$	1080	1080	1080
$cv2ev^{\mathrm{NOI}}$	528	538	557
$cv2ev^{\mathrm{DEEB}}$	1080	1080	1080
ev^{OI}	1080	1080	1080
ev^{NOI}	528	538	558
ev^{DEEB}	1080	1080	1080
cv^{OI}	476	480	507
cv^{NOI}	483	484	508
cv^{DEEB}	1080	1080	1080

The domain extension is $[0, 1000]\times[0, 1500]$ m and is discretized through a Cartesian mesh made of 40×60 cells ($\Delta x_{1,2} = 25$ m). The potential background temperature θ_0 is set to 300 K, and the initial pressure and temperature distributions are calculated identically to those in Sec. 4.2.1. Simulations are integrated in time up to $t_f = 1080$ s. Fig. 7 depicts the evolution in time of the warm bubble using the ev^{DEEB} approach and the ES Godunov flux for the different polynomial approximations. As expected, by increasing k the evolution of smaller flow structures over time is observed. It is also important to mention that the flow symmetry is broken by increasing the polynomial degree. This behavior was also observed by Andrés-Carcasona et al. [48] using different numerical methods. We attribute this behavior to the high-accuracy proper to the method, which captures and propagates even the slightest numerical perturbations that may occur during the solution process. As for the CTB, minimal changes in the thermodynamic variables are measured over the domain. Indeed, according to the ES $k = 7$ final solution ($cv2ev^{\mathrm{DEEB}}$), the temperature and density vary by approximately 5% and 12% of their maximum values, respectively. All ES computations reach the final simulation time, regardless of the solution strategy and the use of DEEB or over-integration. In contrast, some crashes are registered for the EC computations. The results of EC simulations are summarized in Table 3 in terms of elapsed simulation time. As reported for the CTB case, the ev and $cv2ev$ approaches are characterized by an identical behavior, with the need for over-integration or DEEB to complete the run. On the other hand, only DEEB enables the run to be completed successfully when seeking the solution for the conservative variables and directly using them also for the operator assembly, i.e. in the cv strategy. These results confirm the overall potential of DEEB in slightly improving computational robustness at negligible cost.

4.2.3. *The Kelvin-Helmhotz instability problem*

The Kelvin-Helmhotz instability (KHI) problem is featured by two fluids with a velocity shear at their interface. The flow evolution is governed by the Atwood number $A = (\rho_2 - \rho_1)/(\rho_1 + \rho_2) \in [0, 1)$, quantifying the density difference between fluid 1 and 2. Notice

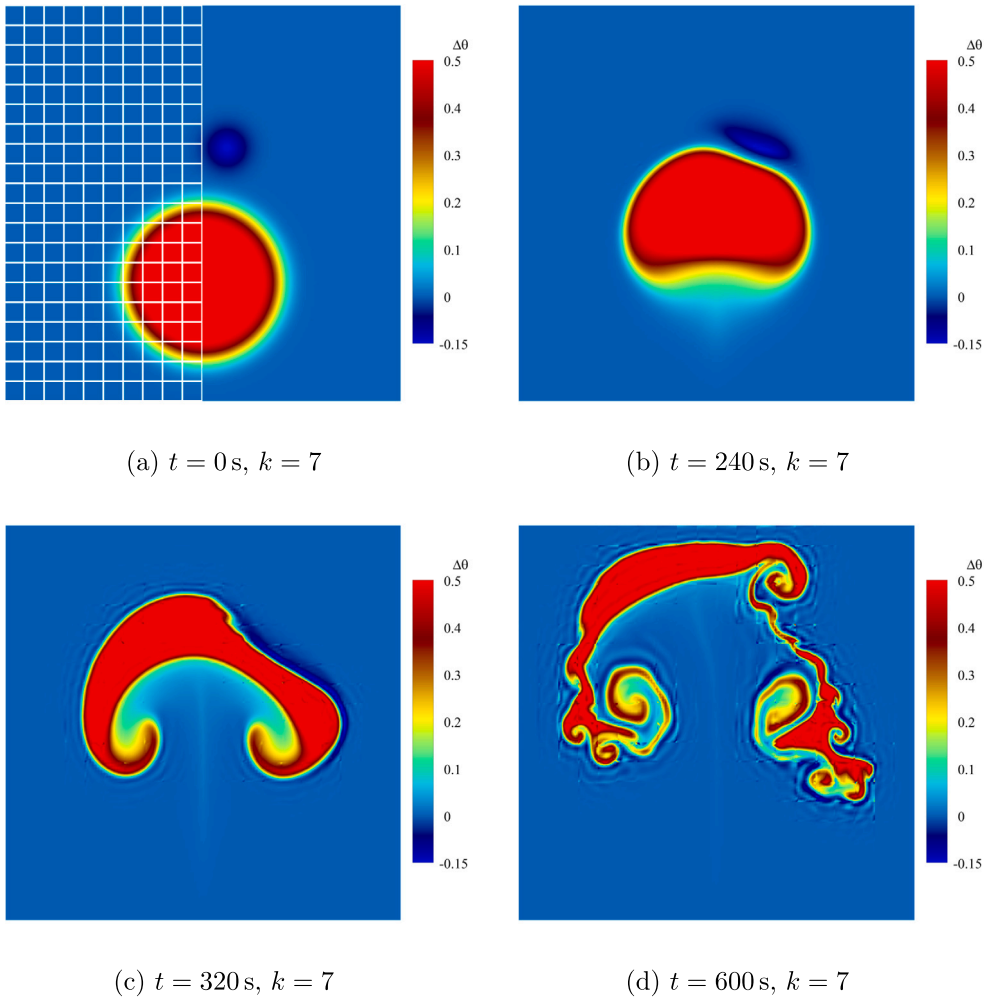


Fig. 5. CTB – time evolution of the $\Delta\theta$ contours using the ES Godunov flux, $cv2ev^{\mathrm{DEEB}}$, $k = 7$ solution. For the frame $t = 0$ s half of the computational mesh is superimposed.

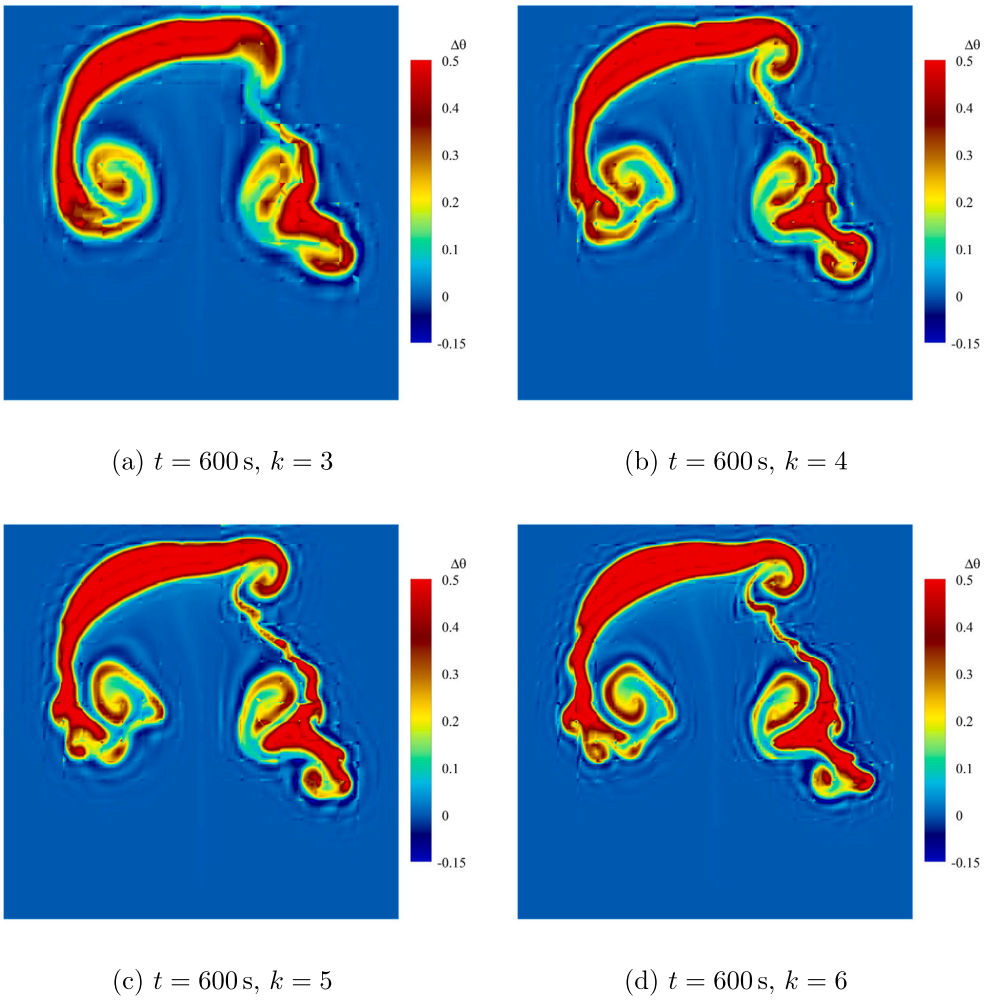
that when $A = 0$ the two fluids have the same density, while for $A \to 1$ their density ratio tends to infinity. The Atwood number can be considered as a measure of how critical the flow problem is. Indeed A is directly linked to the instability driving force and the Mach number, with the initial flow condition becoming supersonic for $A = 0.75$. Following the work of Chan et al. [15], the flow is initialized as

$$\begin{aligned} u_1 &= \left(f(x_2) - \frac{1}{2}\right), \\ u_2 &= \frac{1}{10}\sin(2\pi x_1), \\ p &= 1, \\ \rho &= \rho_1 + f(x_2)\left(\rho_2 - \rho_1\right), \end{aligned} \tag{46}$$

where $\rho_1 = 1$ and $f(x_2) = (1/2)[\tanh(15x_2 + 7.5) - \tanh(15x_2 - 7.5)]$ is a smoothing function to avoid field discontinuities.

The computational domain is a $[-1, 1]^2$ square with doubly-periodic boundary conditions, discretized with $k = \{1, \ldots, 7\}$ DG approximations on a set of 16×16 (coarse) and 32×32 (fine) grids of different type, i.e. Cartesian, with triangular cells and with curved edges. The triangular elements are obtained by cutting in half the quadrangles of the Cartesian mesh. All the simulations are integrated in time up to $t_f = 14$. Fig. 8 shows the density contours for the $k = 7$ approximation on the coarse meshes. It is not surprising that, for this type of intrinsically under-resolved flow problems, the solutions significantly vary depending on the type of mesh used.

The robustness of the different solution strategies is extensively tested for the cases $A = [0.33, 0.75]$. Moreover, similar to Chan et al. [15], the conditions $A = [0.33, 0.57, 0.75, 0.86, 0.89, 0.92, 0.95]$ are simulated with $k = 3$ on the fine mesh and $k = 7$ on the coarse one. To give a quantitative measure of robustness, a Completion Index (CI) is defined as



(a) $t = 600\,\mathrm{s},\ k = 3$ (b) $t = 600\,\mathrm{s},\ k = 4$

(c) $t = 600\,\mathrm{s},\ k = 5$ (d) $t = 600\,\mathrm{s},\ k = 6$

Fig. 6. CTB – $\Delta\theta$ contours using the ES Godunov flux, $cv2ev^{\mathrm{DEEB}}$, $k = 3 - 6$ solutions.

$$\mathrm{CI} = \frac{\sum_{j=1}^{N_{tot}} (t_{end})_j}{N_{tot}\, t_f}, \tag{47}$$

where N_{tot} is the total amount of polynomial approximations or Atwood numbers analyzed and $(t_{end})_j$ indicates the end time for the simulation at a specific k or A. Essentially, CI summarizes the time interval successfully integrated across all polynomial degrees. The bar charts in Fig. 9 show the completion index for the lowest Atwood number case and the three solution strategies, i.e. cv, $cv2ev$, ev. Details on the final execution time for the Cartesian mesh and the grid made of triangles are tabulated for each k value in Tables 4 and 5. The results for the mesh with curved edges are not displayed since they overlap with those of the Cartesian mesh.

For the meshes made of quadrangles, the $cv2ev$ and ev strategies completed the whole set of simulations, whereas cv experienced few crashes. Computations on the triangular elements emphasize the beneficial effect of directly enforcing the entropy balance. DEEB allows nearly matching the robustness of the over-integrated cases for ev and $cv2ev$ and surpasses it when directly using the conservative set, cv. It should be noted that in Fig. 9b, the slightly lower confidence interval (CI) for triangular elements as shown by $cv2ev^{\mathrm{DEEB}}$ when compared to $cv2ev^{\mathrm{OI}}$ can be erratic and must be interpreted with caution. In fact, this could be considered an outlier caused by an unlucky single crash for the highest polynomial degree case. The charts portrayed in Fig. 10 refer to the KHI simulations with the highest Atwood number, i.e. $A = 0.75$. Additionally, Tables 6 and 7 contain, for the same case, details on the computations carried out with the Cartesian meshes and the grids made of triangular cells. The entropy set ev is the most robust when coupled with over-integration or DEEB, but a comparable performance is obtained when using the projected entropy variables $cv2ev$ with DEEB. The performance is dramatically better in these cases than directly using conservative variables, cv. However, DEEB positively affects cv by allowing us to compute the solution for values of k larger than those obtained without it.

The influence of the Atwood number on robustness is summarized in Figs. 11 and 12 through the CI for the polynomial degrees $k = 3$ and $k = 7$, respectively.

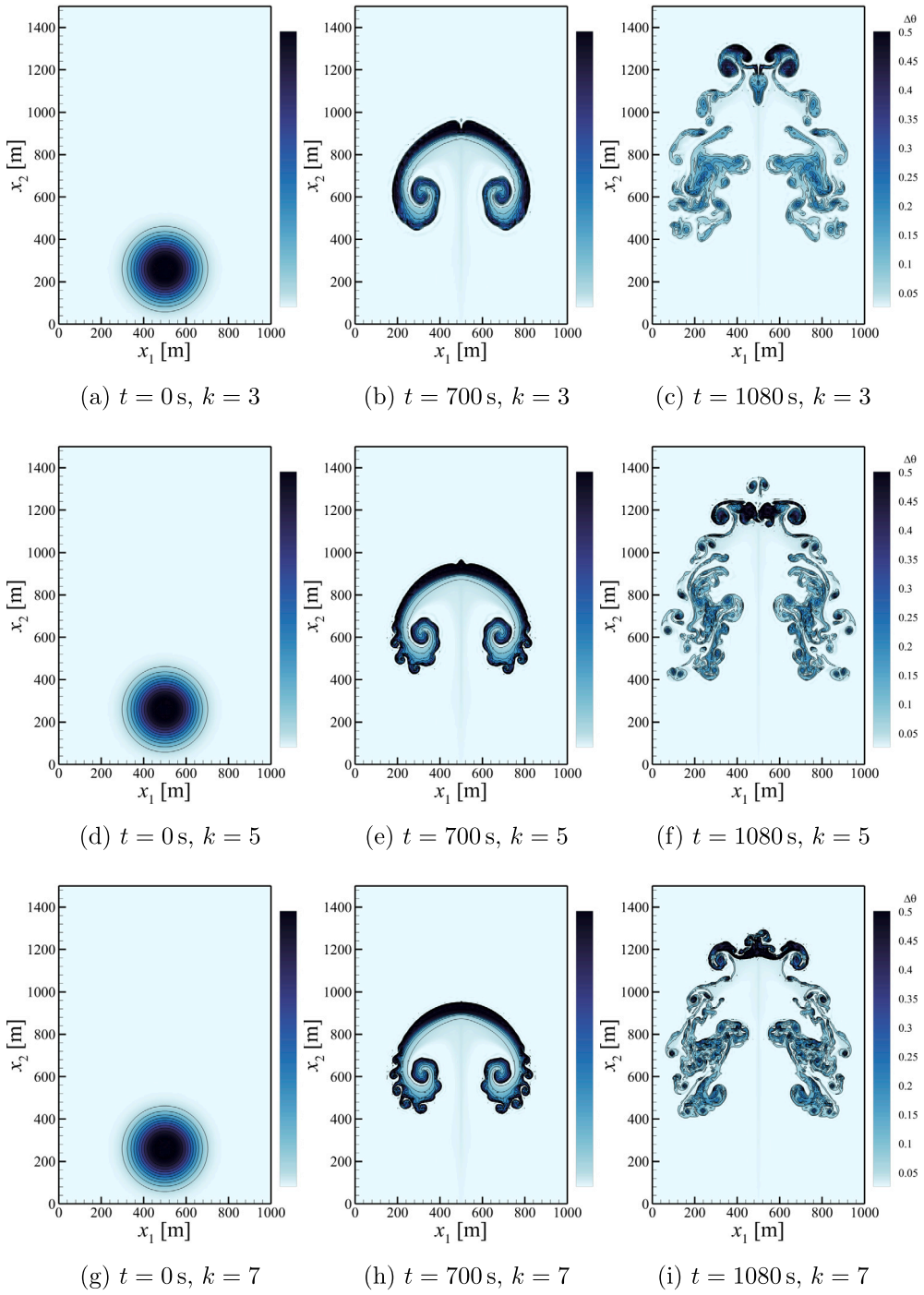


Fig. 7. RTB – $\Delta\theta$ contours using the ES Godunov flux, ev^{DEEB}, $k = \{3, 5, 7\}$ solutions.

For the fourth-order simulations ($k = 3$), DEEB allows to successfully run up to $A = 0.75$ with $cv2ev$ and $A = 0.89$ with ev. When ev is coupled with over-integration, the $A = 0.95$ simulation can even be performed. When considering the eighth-order ($k = 7$) computations in Fig. 12, it becomes evident that for a specific approach, the differences between DEEB, NOI, and OI lessen as the Atwood number rises. Hence, it becomes challenging to classify these strategies in terms of robustness. As expected though, we can observe that the performance of cv declines sooner, at $A = 0.57$, than $cv2ev$ and ev. Overall, the DEEB correction applied to ev and $cv2ev$ allows to achieve a similar robustness to that guaranteed by OI.

Since DEEB acts as a source term added to the discretized governing equations, it is also worth to investigate its possible effect on the stability limit. For this purpose, we test whether or not the use of DEEB affects the maximum value of CFL that allows for the

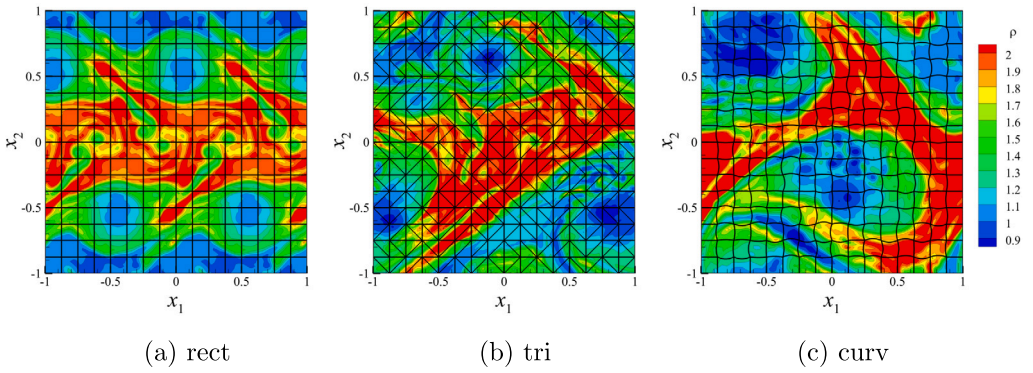


Fig. 8. KHI – density contours for the coarse grids at final time, $A = 0.33$, ev^{DEEB}, $k = 7$ solution.

Table 4
KHI, $A = 0.33$ – end time of the simulations.

Set	k						
	1	2	3	4	5	6	7
16×16 grid (rect)							
$cv2ev^{\mathrm{OI}}$	14	14	14	14	14	14	14
$cv2ev^{\mathrm{NOI}}$	14	14	14	14	14	14	14
$cv2ev^{\mathrm{DEEB}}$	14	14	14	14	14	14	14
ev^{OI}	14	14	14	14	14	14	14
ev^{NOI}	14	14	14	14	14	14	14
ev^{DEEB}	14	14	14	14	14	14	14
cv^{OI}	13.9	14	14	3.8	14	14	14
cv^{NOI}	14	14	14	14	14	14	14
cv^{DEEB}	14	14	14	14	14	14	14
32×32 grid (rect)							
$cv2ev^{\mathrm{OI}}$	14	14	14	14	14	14	14
$cv2ev^{\mathrm{NOI}}$	14	14	14	14	14	14	14
$cv2ev^{\mathrm{DEEB}}$	14	14	14	14	14	14	14
ev^{OI}	14	14	14	14	14	14	14
ev^{NOI}	14	14	14	14	14	14	14
ev^{DEEB}	14	14	14	14	14	14	14
cv^{OI}	14	14	5.9	8.2	6.0	14	5.3
cv^{NOI}	14	14	14	14	14	14	14
cv^{DEEB}	14	14	14	14	14	14	14

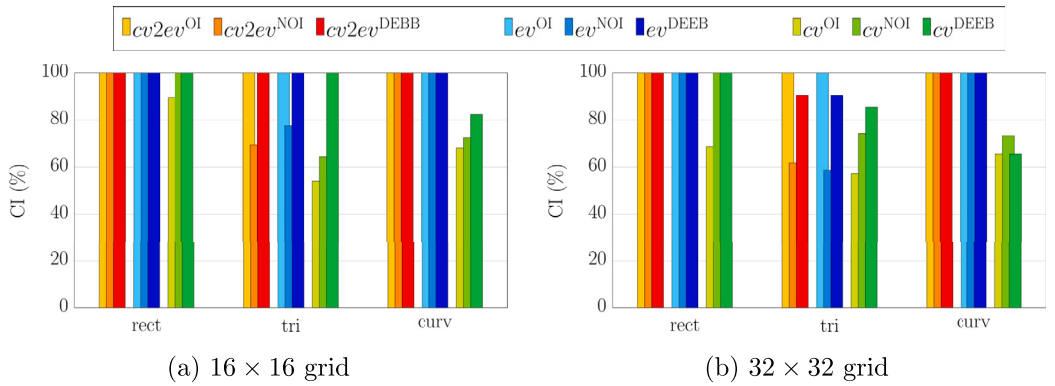


Fig. 9. KHI, $A = 0.33$ – robustness comparison for the different solution strategies. The completion index covers the range $k = 1, \ldots, 7$.

successful run of the KHI case. Progressively larger values of the CFL are then used for computations with all the strategies, i.e. ev, cv and $cv2ev$, until a simulation crash occurs. Two flow configurations, characterized by $A = 0.2$ and $A = 0.7$, are chosen to test the CFL stability. All the simulations are performed using a $k = 3$ approximation and the 16×16 Cartesian grid, providing a reasonable

Table 5
KHI, $A = 0.33$ – end time of the simulations.

Set	k						
	1	2	3	4	5	6	7
16×16 grid (tri)							
$cv2ev^{\mathrm{OI}}$	14	14	14	14	14	14	14
$cv2ev^{\mathrm{NOI}}$	14	14	14	2.1	14	6.3	3.8
$cv2ev^{\mathrm{DEEB}}$	14	14	14	14	14	14	14
ev^{OI}	14	14	14	14	14	14	14
ev^{NOI}	14	14	14	2.1	14	14	4.1
ev^{DEEB}	14	14	14	14	14	14	14
cv^{OI}	14	14	4.7	4.6	4.7	5.7	5.3
cv^{NOI}	14	14	4.9	5.2	5.5	14	5.5
cv^{DEEB}	14	14	14	14	14	14	14
32×32 grid (tri)							
$cv2ev^{\mathrm{OI}}$	14	14	14	14	14	14	14
$cv2ev^{\mathrm{NOI}}$	14	14	14	5.9	4.2	5.1	3.4
$cv2ev^{\mathrm{DEEB}}$	14	14	14	14	14	14	4.6
ev^{OI}	14	14	14	14	14	14	14
ev^{NOI}	14	14	10.9	5.9	4.2	5.1	3.4
ev^{DEEB}	14	14	14	14	14	14	3.3
cv^{OI}	14	14	5.9	5.6	5.1	5.5	6.0
cv^{NOI}	14	14	14	14	5.9	5.0	5.9
cv^{DEEB}	14	14	14	14	14	7.0	6.7

Table 6
KHI, $A = 0.75$ – end time of the simulations.

Set	k						
	1	2	3	4	5	6	7
16×16 grid (rect)							
$cv2ev^{\mathrm{OI}}$	14	14	4.3	3.9	3.9	3.9	4.0
$cv2ev^{\mathrm{NOI}}$	14	14	14	4.3	3.7	3.9	3.8
$cv2ev^{\mathrm{DEEB}}$	14	14	14	5.5	14	3.9	3.8
ev^{OI}	14	14	14	14	4.6	4.7	4.3
ev^{NOI}	14	14	14	5.6	4.1	4.5	3.9
ev^{DEEB}	14	14	14	14	6.5	4.5	3.9
cv^{OI}	1.4	1.5	2.5	2.8	2.8	2.7	2.1
cv^{NOI}	1.5	1.2	2.4	2.9	1.6	3.3	3.2
cv^{DEEB}	14	2.4	2.8	3.2	1.7	3.3	3.2
32×32 grid (rect)							
$cv2ev^{\mathrm{OI}}$	14	14	4.8	4.6	3.8	3.7	3.6
$cv2ev^{\mathrm{NOI}}$	14	14	14	4.6	4.3	3.8	3.9
$cv2ev^{\mathrm{DEEB}}$	14	14	14	4.6	4.0	3.5	3.9
ev^{OI}	14	14	14	14	14	3.7	3.9
ev^{NOI}	14	14	14	6.5	5.0	3.7	3.9
ev^{DEEB}	14	14	14	14	14	3.7	3.9
cv^{OI}	2.8	2.7	3.1	3.1	3.4	3.3	3.4
cv^{NOI}	4.2	3.0	3.2	3.2	3.6	3.4	3.5
cv^{DEEB}	14	3.0	3.2	3.2	3.6	3.4	3.5

trade-off between spatial resolution and computational complexity. For the $A = 0.2$ case, all the approaches are able to reach the simulation end time, regardless of the DEEB, up to CFL = 0.5. The non-alteration of the CFL upper bound by the DEEB further emerges from the $A = 0.7$ results. Indeed, net of the robustness differences between the different sets, with or without DEEB: *i*) $cv2ev$ runs to simulation end time up to CFL = 0.3; *ii*) ev runs to simulation end time up to CFL = 0.4; *iii*) cv crashes from CFL = 0.2 onwards. To summarize, we can infer that utilizing the DEEB has no detectable adverse impact on the stability of the numerical schemes.

4.2.4. *The Richtmyer-Meshkov instability problem*

The Richtmyer–Meshkov Instability (RMI) [49,50] is a flow phenomenon occurring in engineering applications, like inertial confinement fusion, as well as in astrophysical problems, like supernovae detonation. The phenomenon is characterized by shock waves hitting a stratified fluid.

Table 7
KHI, $A = 0.75$ – end time of the simulations.

Set	k						
	1	2	3	4	5	6	7
16×16 grid (tri)							
$cv2ev^{\mathrm{OI}}$	0.0	4.9	3.2	3.6	3.6	2.8	3.1
$cv2ev^{\mathrm{NOI}}$	14	3.6	4.6	2.0	3.2	3.2	3.2
$cv2ev^{\mathrm{DEEB}}$	14	14	14	4.3	3.2	3.5	3.2
ev^{OI}	14	14	14	14	3.8	4.0	3.7
ev^{NOI}	14	3.7	4.6	2.0	3.2	3.4	3.2
ev^{DEEB}	14	14	14	4.5	3.2	3.5	3.2
cv^{OI}	1.3	1.1	1.7	1.9	2.4	2.0	2.5
cv^{NOI}	0.9	0.7	1.7	1.4	1.4	1.5	1.5
cv^{DEEB}	14	2.6	2.4	1.8	1.6	1.5	1.5
32×32 grid (tri)							
$cv2ev^{\mathrm{OI}}$	14	5.5	4.0	4.0	4.1	3.7	3.4
$cv2ev^{\mathrm{NOI}}$	14	4.0	2.1	3.8	3.7	3.7	3.5
$cv2ev^{\mathrm{DEEB}}$	14	14	14	4.4	3.7	3.7	3.5
ev^{OI}	14	14	14	5.0	4.4	4.0	4.0
ev^{NOI}	14	4.2	2.1	3.9	3.8	3.7	3.0
ev^{DEEB}	14	14	14	4.4	3.7	3.7	3.5
cv^{OI}	1.6	2.0	2.1	2.1	3.5	2.9	3.6
cv^{NOI}	1.1	2.1	2.9	2.1	2.4	3.0	2.9
cv^{DEEB}	14	3.2	3.3	2.2	2.4	2.9	2.9

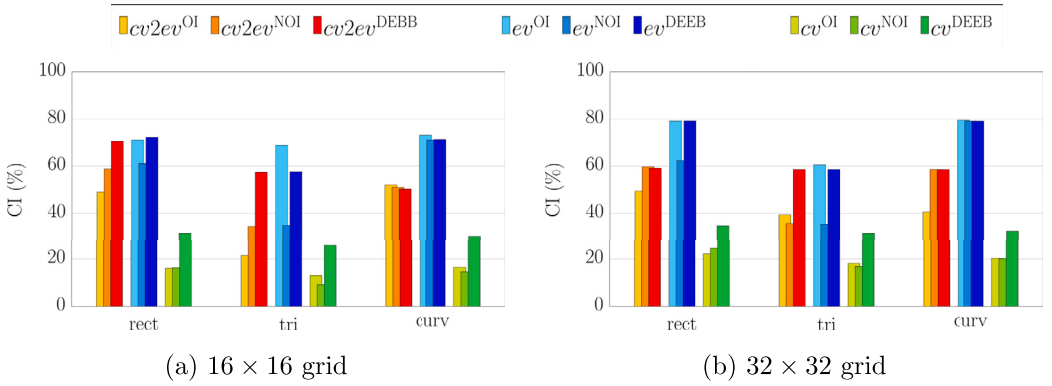


Fig. 10. KHI, $A = 0.75$ – bar charts for comparing the robustness of the various solution strategies. The completion index covers the range $k = \{1, \dots, 7\}$.

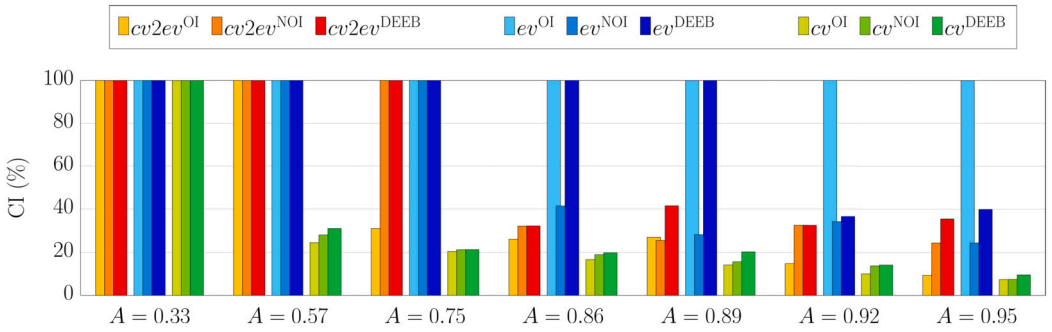


Fig. 11. KHI – completion index for increasingly higher Atwood numbers with $k = 3$ on the 32×32 grid (rect).

The flow is initialized as

$$\begin{aligned} u_i &= 0, \\ p &= 4.9 - 1.95\left[1 + \tanh\left(2\xi_1\right)\right], \\ \rho &= 4.22 - 1.985\left[2 + \tanh\left(2\xi_1\right) + \tanh\left(2\xi_2\right)\right], \end{aligned} \tag{48}$$

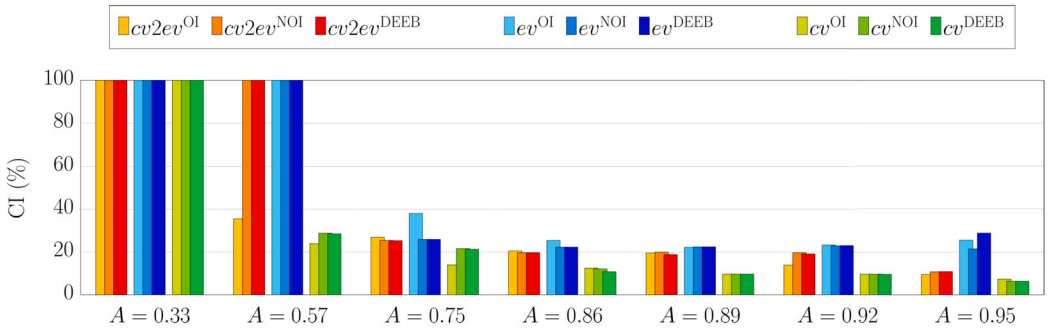


Fig. 12. KHI – completion index for increasingly higher Atwood numbers with $k = 7$ on the 16×16 grid (rect).

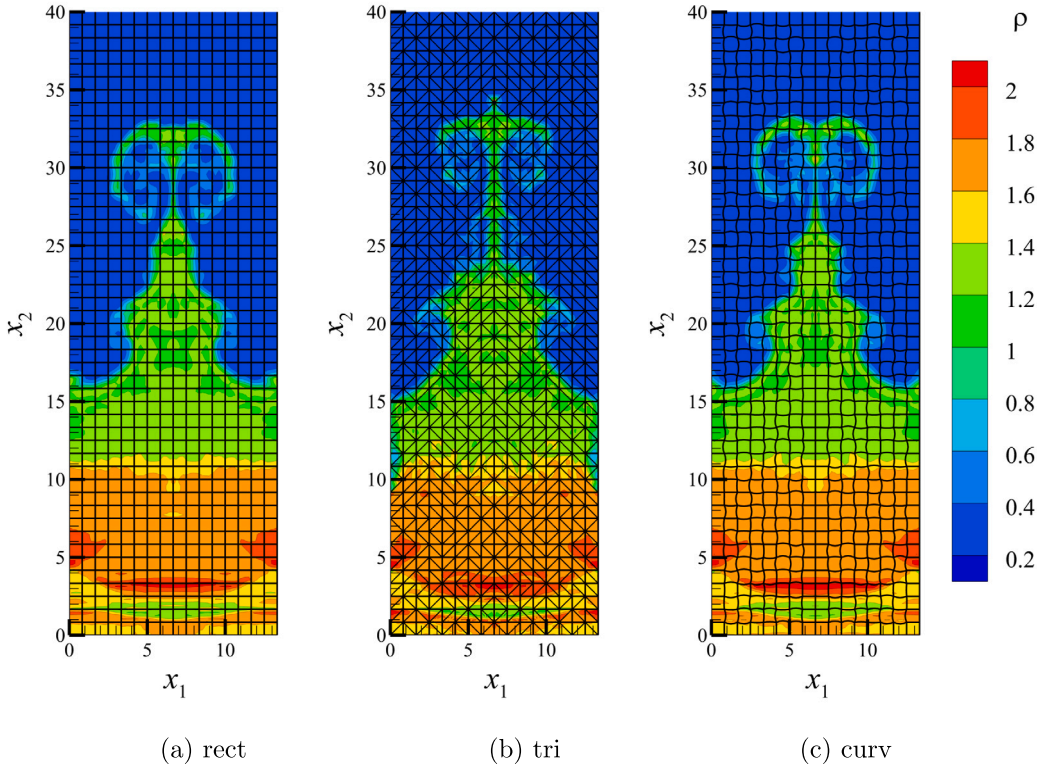


Fig. 13. RMI – density contours on the coarse grids at final time, ev^{DEEB}, $k = 4$ solution.

where

$$\begin{aligned} \xi_1 &= |x_2 - 4| - 2, \\ \xi_2 &= x_2 - \left[18 + 2\cos(6\pi x_1/40)\right]. \end{aligned} \tag{49}$$

Although the original computational domain is a $[0, 40/3] \times [0, 40]$ rectangle, in this work, simulations are run on half of the domain, i.e. $x_1 \in [0, 40/6]$, imposing symmetry boundary conditions to save computational time. The halved domain is discretized with a 8×48 (coarse) and a 16×96 (fine) grid, composed by the same three element shapes presented in Sec. 4.2.3. Fig. 13 shows the density contours on the various coarse meshes. The results have been mirrored with respect to a vertical axis passing through $x_1 = 40/6$ for visual clarity. All the computations are performed integrating the solution in time up to $t_f = 42$. As for the KHI flow problem, the robustness of the different strategies is investigated through the Completion Index (CI) for $k = \{1, \dots, 7\}$. The bar charts shown in Fig. 14, along with the data presented in Tables 8 and 9, confirm that the ev strategy is the most effective, followed by $cv2ev$ and with cv occupying the last place. Again, the trend observed for KHI is confirmed here, with DEEB giving a comparable performance to that provided by OI for ev and improving it for the $cv2ev$ set.

Table 8
RMI – end time of the simulations.

Set	k						
	1	2	3	4	5	6	7
8×48 grid (rect)							
$cv2ev^{\mathrm{OI}}$	42	42	42	8.5	8.5	8.5	7.1
$cv2ev^{\mathrm{NOI}}$	42	42	42	42	33.3	8.5	8.6
$cv2ev^{\mathrm{DEEB}}$	42	42	42	42	42	8.6	8.6
ev^{OI}	42	42	42	42	42	42	42
ev^{NOI}	42	42	42	42	33.2	42	24.5
ev^{DEEB}	42	42	42	42	42	42	42
cv^{OI}	42	6.6	6.5	4.2	3.3	2.9	2.5
cv^{NOI}	42	26.5	23.8	4.9	4.5	4.5	4.5
cv^{DEEB}	42	42	5.3	4.9	4.5	4.6	4.5
16×96 grid (rect)							
$cv2ev^{\mathrm{OI}}$	42	42	42	7.5	7.5	7.5	7.5
$cv2ev^{\mathrm{NOI}}$	42	42	42	42	26.6	8.7	8.7
$cv2ev^{\mathrm{DEEB}}$	42	42	42	42	42	7.9	7.7
ev^{OI}	42	42	42	42	42	42	42
ev^{NOI}	42	42	42	42	25.8	24.7	28.5
ev^{DEEB}	42	42	42	42	42	42	9.6
cv^{OI}	42	7.0	5.7	5.7	3.4	3.2	2.8
cv^{NOI}	42	7.0	6.0	5.7	5.3	4.5	4.5
cv^{DEEB}	42	42	5.8	5.2	4.8	4.5	4.5

Table 9
RMI – end time of the simulations.

Set	k						
	1	2	3	4	5	6	7
8×48 grid (tri)							
$cv2ev^{\mathrm{OI}}$	42	42	42	7.3	2.2	2.1	2.0
$cv2ev^{\mathrm{NOI}}$	42	42	42	1.9	1.9	1.3	1.2
$cv2ev^{\mathrm{DEEB}}$	42	42	42	36.7	18.7	7.0	7.0
ev^{OI}	42	42	42	42	42	38.8	41.4
ev^{NOI}	42	42	34.5	1.9	1.9	1.3	1.3
ev^{DEEB}	42	42	42	42	30.1	18.6	7.3
cv^{OI}	4.2	5.3	1.8	1.7	1.6	1.3	1.3
cv^{NOI}	42	27.3	1.9	1.9	1.6	1.5	1.3
cv^{DEEB}	42	42	5.8	5.6	1.8	1.5	1.4
16×96 grid (tri)							
$cv2ev^{\mathrm{OI}}$	42	42	42	6.1	1.9	1.8	1.5
$cv2ev^{\mathrm{NOI}}$	42	42	27.7	2.0	1.4	1.3	1.3
$cv2ev^{\mathrm{DEEB}}$	42	42	42	42	7.2	6.4	5.4
ev^{OI}	42	42	42	42	42	32.9	7.2
ev^{NOI}	42	42	24.0	2.0	1.4	1.3	1.3
ev^{DEEB}	42	42	42	42	25.4	7.2	6.4
cv^{OI}	24.1	2.2	1.6	1.5	1.3	1.3	1.3
cv^{NOI}	26.0	20.5	1.9	2.0	1.5	1.3	1.3
cv^{DEEB}	42	7.0	5.4	3.4	2.1	1.4	1.4

4.2.5. *A few remarks*

According to the results presented in the previous sections, a substantial difference in the impact of over-integration is observed, resulting in increased robustness for ev but potentially weakening both $cv2ev$ and cv, see Figs. 9 and 14. This contrast can be explained by considering that although over-integration is always helpful in terms of under-resolution, the positivity of thermodynamic variables is only controlled when entropy variables are used. Indeed, for an ideal gas ($p=\rho RT$), the temperature reads $T=-1/(Rv_{2+d})$, meaning that v_{2+d} should cross a vertical asymptote in the $T-v_{2+d}$ plane to result in negative values of T. Moreover, pressure is defined as an exponential function of the entropy variables

$$p=\left(-\frac{1}{v_{2+d}}\right)^{\frac{\gamma}{\gamma-1}}\exp\left(\frac{\gamma}{1-\gamma}+v_1-\frac{1}{2}\frac{v_{i+1}v_{i+1}}{v_{2+d}}\right)\quad i=1,\dots,d,$$

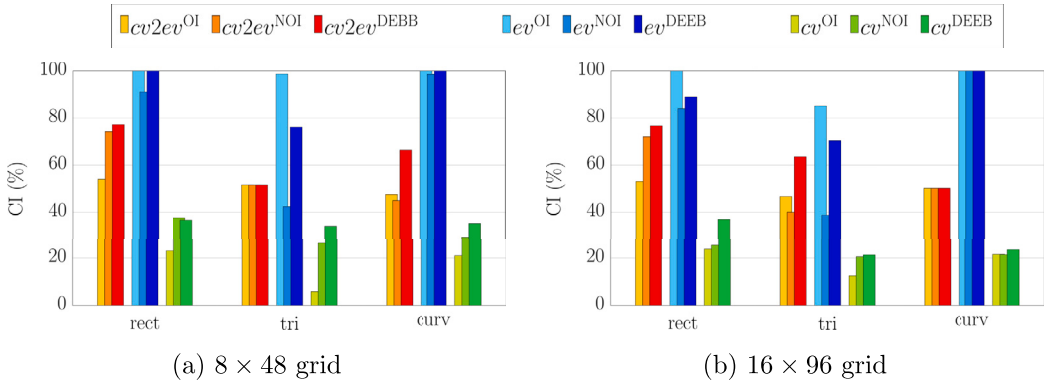


Fig. 14. RMI – bar charts for comparing the robustness of the various solution strategies. The completion index covers the range $k = \{1, \dots, 7\}$.

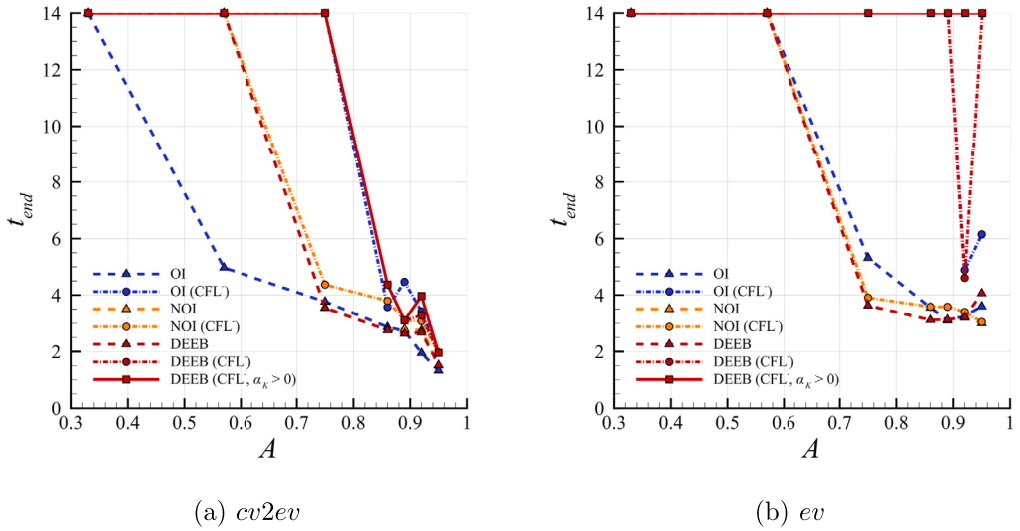


Fig. 15. KHI – end time for $k = 7$ computations on the 16×16 grid (rect) and different Atwood numbers. Results tabulated for the CFL value 0.1, the lowered value $0.002 \leq \text{CFL}^- \leq 0.1$, and the CFL^- value coupled with the lower bound on the DEEB correction (CFL^-, $\alpha_K > 0$).

and, as long as v_{2+d} is negative, the state is physically consistent. In practice, the intrinsic properties of the entropy set support a sort of "weak" positivity constraint on thermodynamic variables. On the contrary, the conservative variables have no built-in restrictions on the sign of thermodynamic properties. Thus, with increased quadrature points because of over-integration, there is a higher possibility of encountering a point where the solution becomes nonphysical and diverges. This, of course, can also happen when adopting the entropy projection strategy.

This section also compares the performance of the proposed strategies to the work of Chan et al. [15]. Their paper considers both a nodal and a modal version of an entropy conserving/stable Discontinuous Galerkin Spectral Element Method (DGSEM) developed using multidimensional SBP operators, flux differencing, and tensor product basis. A direct comparison is not straightforward since differences are not only in spatial discretization but also in time integration, having them implemented an optimized, adaptive fourth-order nine-stage explicit Runge-Kutta scheme. Although their modal discretization could appear superior to any of our strategies, the performance gap can decrease and even reverse with minor tuning, such as adjusting the CFL value. By progressively decreasing it, down to a minimum value 0.002, the number of ev and $cv2ev$ simulations which successfully complete significantly raises. As a further improvement, the fix proposed by Abgrall et al. in [16] can be also considered, which retains α_K only if it is greater than zero. This means applying the correction only when the numerical solution evolves against the physics, i.e. "destroying" thermodynamic entropy. The enhanced robustness of these variants may be appreciated for the KHI case at different Atwood numbers in Fig. 15. When directly using the entropy variables with over-integration (ev^{OI}) and lowering the CFL value to 0.05, the KHI simulation can also be completed for the very high Atwood number 0.98 on the 32×32 mesh of quadrilaterals using $k = 3$, even without restricting α_K to only positive values. Interestingly, when using DEEB correction (ev^{DEEB}), the same simulation runs without needing to lower the CFL value from 0.1. The corresponding density contours are illustrated in Fig. 16. The ev^{DEEB} strategy proves to be as robust as the solver proposed by Chan et al. [15] by computing the $A = 0.75$ case on the 16×16 grid with $k = 10$. Notice that, as the present method uses a Pascal basis, a higher polynomial degree approximation is needed to reasonably match the number of DOFs with the seventh-order tensor basis approximation used by Chan et al. [15]. To summarize, the findings indicate that a "standard"

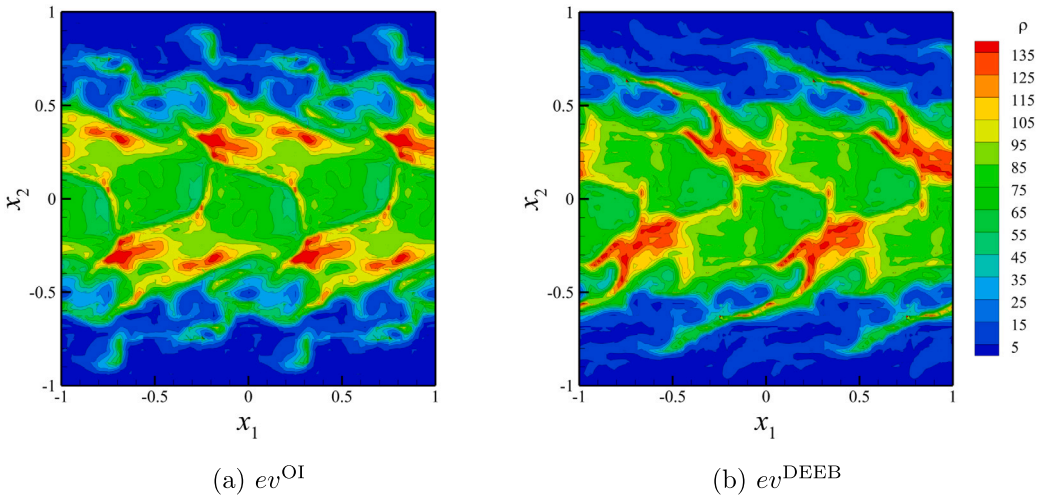


Fig. 16. KHI – density contours at final time for $k = 3$ computations at $A = 0.98$ on the 32×32 grid (rect).

modal DG method utilizing entropy variables and possibly the DEEB correction can compete with some of the newer DG variants, i.e. [15]. Specifically, for problems that include small-scale features and large variations in thermodynamic variables, our $cv2ev^{\mathrm{DEEB}}$ implementation shows similar robustness to the nodal DGSEM. Additionally, the modal variant of the DGSEM [15] with an entropy projection aligns with our ev^{DEEB} version.

4.2.6. GCN results

The SSPRK35 time integrator is not an entropy-conserving/stable scheme, and explicit methods are generally deemed to produce spurious entropy, as stated by Lozano in [20]. This could be the reason behind the simulation crashes observed in the previous sections. To further investigate this behavior the $cv2ev^{\mathrm{DEEB}}$ approach is combined with the GCN entropy-conserving implicit time integrator.

Concerning the KHI flow problem, the GCN allows the use of significantly larger CFL values for low Atwood numbers. We indeed report, for the polynomial degree $k = 3$ on the coarse 16×16 Cartesian grid, a CFL ≈ 50 at $A = 0.2$ against a maximum value CFL $= 0.5$ for the explicit scheme, see Sec. 4.2.3. As A increases, i.e. for larger values of the instability-driving density gradient, the CFL values for the GCN have to be reduced to the same level as those set for explicit computations. For $A = 0.7$ the maximum value of CFL for the implicit integrator is reduced to 1.2, while the upper limit of SSPRK35 for the same case is 0.3. When further increasing the Atwood number and the level of spatial resolution, an even stronger reduction in the stability limit is observed. To allow the GCN to run up to end time at $A = 0.75$ with $k = 7$ on the Cartesian 16×16 mesh, the CFL have to be reduced to 0.05, with the SSPRK35 scheme running at CFL $= 0.025$. A similar behavior is observed for the RMI problem. On the Cartesian 8×48 grid, using a fixed CFL $= 0.1$ value, the GCN successfully complete the simulation up to $k = 6$, whereas the explicit scheme diverges for $k > 5$. However, for $k = 7$ both time integrators can run up to end time only if CFL is reduced to 0.05.

In the Authors opinion, the comparison between SSPRK and GCN shows that explicit time integration results in minimal production of spurious entropy when the time step size is small enough. This is because the entropy conservation error for SSPRK35 is $\mathcal{O}(\Delta t^3)$, cf. Sec. 4.1, thus preventing the spurious production in time of S to overwhelm or balance the S destruction proper of the spatial discretization. This observation, coupled with the strong time-step limitation for high Atwood numbers, which affects both explicit and implicit patterns, suggests that simulation crashes are to be attributed to the occurrence of negative values of thermodynamic variables. It should be noted that even though GCN is an entropy-preserving integrator, it cannot guarantee the positivity of thermodynamic states. In some cases, reducing the time step size is the only option to avoid negative values.

4.3. Computational efficiency

In the previous section the use of entropy projection and DEEB have been shown to be in favor of computational robustness. The total CPU time taken to simulate the KHI with $k = \{1, \dots, 7\}$ is measured to estimate the computational efficiency of different solution strategies. To ensure the completion of the simulations for each polynomial degree, the low Atwood number $A = 0.2$ is considered. Table 10 and Fig. 17 show the overhead to assemble and invert $\mathbf{P}(\mathbf{v}_h)$ for the set ev, which is $\mathcal{O}(k^{3d})$, cf. Sec. 3.3. The comparison between $cv2ev^{\mathrm{DEEB}}$ and cv^{NOI} reveals that the strategy based on direct use of conservative variables is subjected to a 40% slowdown when implementing entropy projection and DEEB correction. Roughly half of this overhead can be attributed to the entropy projection while the rest is likely due to the more complicated evaluation of the residual vector because of the mathematical form of the entropy variables. Assembling the DEEB correction comes at a negligible expense, thanks to the orthonormal basis detailed in Sec. 3.2. The relative cost of projecting variables and assembling the DEEB term decreases when discretizing the Navier-Stokes equations, as additional terms (viscous fluxes) need to be evaluated in the residual function. As expected, over-integration results in

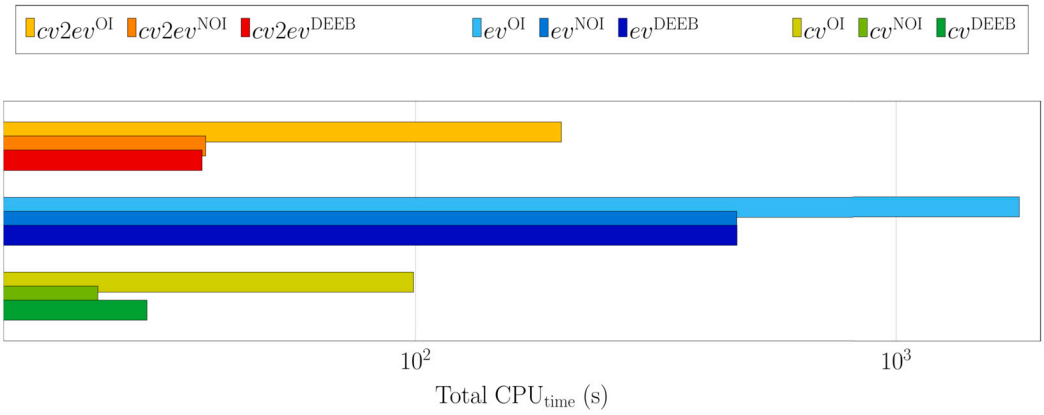


Fig. 17. KHI, $A = 0.2$ – total CPU time for $k = \{1, \dots, 7\}$ on the 16×16 Cartesian mesh.

Table 10
KHI, $A = 0.2$ – ratio between the CPU time of the different strategies and of $cv2ev^{DEEB}$ on the 16×16 Cartesian mesh. The last column considers the overall CPU time for $k = 1, \dots, 7$.

$CPU_{time}(\circ)/CPU_{time}(\bullet)$	$k = 1$	$k = 4$	$k = 7$	**Total**
$cv^{NOI}/cv2ev^{DEEB}$	0.8	0.6	0.6	**0.6**
$cv^{DEEB}/cv2ev^{DEEB}$	0.8	0.6	0.8	**0.8**
$cv2ev^{NOI}/cv2ev^{DEEB}$	1.0	1.0	1.0	**1.0**
$cv2ev^{OI}/cv2ev^{DEEB}$	16.1	6.9	3.5	**5.6**
$ev^{DEEB}/cv2ev^{DEEB}$	1.4	6.3	21.0	**13.0**
$ev^{OI}/cv2ev^{DEEB}$	28.2	43.1	59.6	**50.3**

a significant increase in CPU time. Although Table 10 may suggest that this cost decreases as k increases, this is an effect of setting an arbitrarily fixed degree of exactness, i.e. 30.

5. Conclusions

The article proposes an efficient and highly-accurate modal DG method for the discretization of the Euler equation able to guarantee the physical evolution of entropy at the discrete level. The scheme solves the conservative variables but takes advantage from entropy projection [45] for the spatial discretization assembly and of an elemental correction term designed by Abgrall [1] to directly enforce entropy conservation and avoid over-integration. For the temporal discretization, an explicit Strong Stability Preserving Runge-Kutta scheme and an entropy-conserving implicit time integrator, i.e. the Generalized Crank–Nicolson, are considered, showing that the former can conserve entropy in the limit of $\Delta t \to 0$ while the latter results in an entropy-conserving scheme regardless of time-step and mesh size. A detailed comparison is presented between the performance and accuracy of the proposed method and the direct use of conservative or entropy variables. Coupled with the explicit temporal integrator, the method is found to be an excellent compromise between the inherent robustness of entropy variables and the low computational complexity of conservative variables, due to the use of an orthonormal basis defined on the mesh space. The numerical results also show that neither entropy projection nor DEEB spoil the theoretical spatial/temporal convergence rate.

Future work will be devoted to the extension of the current DG framework to the discretization of the Navier-Stokes equations and its application to the scale-resolving simulation of turbulent flows. The implementation of entropy-stable shock capturing techniques and positive preserving limiters for the thermodynamic variables will be also investigated to tackle transonic flows problems. Finally, GPU implementation will be considered to take advantage of the low computational complexity of the method, paving the way to the full exploitation of the new generation of High-Performance Computing platforms.

CRediT authorship contribution statement

Luca Alberti: Data curation, Formal analysis, Investigation, Software, Validation, Visualization, Writing – original draft, Writing – review & editing, Methodology. **Emanuele Carnevali:** Data curation, Formal analysis, Investigation, Methodology, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. **Alessandro Colombo:** Data curation, Formal analysis, Investigation, Methodology, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. **Andrea Crivellini:** Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Software, Supervision, Validation, Visualization, Writing – original draft, Writing – review & editing.

Declaration of competing interest

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

Data availability

Data will be made available on request.

Acknowledgement

We acknowledge Dr. Samuele Capuzzi for contributing to developing the numerical framework and setting up the NWP flow cases.

References

[1] 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.
[2] T.J. 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 (2) (1986) 223–234.
[3] M.S. Mock, Systems of conservation laws of mixed type, J. Differ. Equ. 37 (1) (1980) 70–88.
[4] J.E. Hicken, D.C. Del Rey Fernández, D.W. Zingg, Multidimensional summation-by-parts operators: general theory and application to simplex elements, SIAM J. Sci. Comput. 38 (4) (2016) A1935–A1958.
[5] M. Carpenter, T. Fisher, E. Nielsen, M. Parsani, M. Svärd, N. Yamaleev, Entropy stable summation-by-parts formulations for compressible computational fluid dynamics, in: Handbook of Numerical Analysis, vol. 17, Elsevier, 2016, pp. 495–524.
[6] 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.
[7] 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 (3) (2013) A1233–A1253.
[8] T. Chen, C.-W. Shu, Review of entropy stable discontinuous Galerkin methods for systems of conservation laws on unstructured simplex meshes, CSIAM Trans. Appl. Math. 1 (1) (2020) 1–52.
[9] 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.
[10] 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.
[11] J. Chan, On discretely entropy conservative and entropy stable discontinuous Galerkin methods, J. Comput. Phys. 362 (2018) 346–374.
[12] J. Chan, Skew-symmetric entropy stable modal discontinuous Galerkin formulations, J. Sci. Comput. 81 (2019) 459–485.
[13] 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.
[14] 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.
[15] 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. (2022) 356.
[16] 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.
[17] E. Gaburro, P. Öffner, M. Ricchiuto, D. Torlo, High order entropy preserving ADER-DG schemes, Appl. Math. Comput. 440 (2023) 127644.
[18] R.J. Spiteri, S.J. Ruuth, A new class of optimal high-order strong-stability-preserving time discretization methods, SIAM J. Numer. Anal. 40 (2) (2002) 469–491.
[19] 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.
[20] C. Lozano, Entropy production by explicit Runge–Kutta schemes, J. Sci. Comput. 76 (1) (2018) 521–564.
[21] P.D. Lax, Hyperbolic Systems of Conservation Laws and the Mathematical Theory of Shock Waves, SIAM, 1973.
[22] K.O. Friedrichs, P.D. Lax, Systems of conservation equations with a convex extension, Proc. Natl. Acad. Sci. USA 68 (8) (1971) 1686–1688.
[23] A. Harten, On the symmetric form of systems of conservation laws with entropy, J. Comput. Phys. 49 (1983).
[24] A. Harten, P.D. Lax, C.D. Levermore, W.J. Morokoff, Convex entropies and hyperbolicity for general Euler equations, SIAM J. Numer. Anal. 35 (6) (1998) 2117–2127.
[25] E. Tadmor, Entropy functions for symmetric systems of conservation laws, J. Math. Anal. Appl. 122 (2) (1987) 355–359.
[26] E. Tadmor, The numerical viscosity of entropy stable schemes for systems of conservation laws. I, Math. Comput. 49 (179) (1987) 91–103.
[27] E. Tadmor, Entropy stability theory for difference approximations of nonlinear conservation laws and related time-dependent problems, Acta Numer. 12 (2003) 451–512.
[28] F. Ismail, P.L. Roe, Affordable, entropy-consistent Euler flux functions II: entropy production at shocks, J. Comput. Phys. 228 (15) (2009) 5410–5436.
[29] J. Gottlieb, C.P. Groth, Assessment of Riemann solvers for unsteady one-dimensional inviscid flows of perfect gases, J. Comput. Phys. 78 (2) (1988) 437–458.
[30] 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.
[31] A. Gouasmi, S.M. Murman, K. Duraisamy, Entropy conservative schemes and the receding flow problem, J. Sci. Comput. 78 (2019) 971–994.
[32] P.G. Lefloch, J.-M. Mercier, C. Rohde, Fully discrete, entropy conservative schemes of arbitrary order, SIAM J. Numer. Anal. 40 (5) (2002) 1968–1992.
[33] G.J. Gassner, A.D. Beck, On the accuracy of high-order discretizations for underresolved turbulence simulations, Theor. Comput. Fluid Dyn. 27 (3) (2013) 221–237.
[34] R.M. Kirby, G.E. Karniadakis, De-aliasing on non-uniform grids: algorithms and applications, J. Comput. Phys. 191 (1) (2003) 249–264.
[35] G. Mengaldo, D. De Grazia, D. Moxey, P. Vincent, S. Sherwin, Dealiasing techniques for high-order spectral element methods on regular and irregular grids, J. Comput. Phys. 299 (2015) 56–81.

[36] A.R. Winters, R.C. Moura, G. Mengaldo, G.J. Gassner, S. Walch, J. Peiro, S.J. Sherwin, A comparative study on polynomial dealiasing and split form discontinuous Galerkin schemes for under-resolved turbulence computations, J. Comput. Phys. 372 (2018) 1–21.
[37] H.C. Yee, N.D. Sandham, M.J. Djomehri, Low-dissipative high-order shock-capturing methods using characteristic-based filters, J. Comput. Phys. 150 (1) (1999) 199–238.
[38] C. Hu, C.-W. Shu, Weighted essentially non-oscillatory schemes on triangular meshes, J. Comput. Phys. 150 (1) (1999) 97–127.
[39] L. Wang, D.J. Mavriplis, Implicit solution of the unsteady Euler equations for high-order accurate discontinuous Galerkin discretizations, J. Comput. Phys. 225 (2) (2007) 1994–2015.
[40] A. Robert, Bubble convection experiments with a semi-implicit formulation of the Euler equations, J. Atmos. Sci. 50 (13) (1993) 1865–1873.
[41] F. Giraldo, M. Restelli, A study of spectral element and discontinuous Galerkin methods for the Navier–Stokes equations in nonhydrostatic mesoscale atmospheric modeling: equation sets and test cases, J. Comput. Phys. 227 (8) (2008) 3849–3877.
[42] D. Ghosh, E.M. Constantinescu, Well-balanced, conservative finite difference algorithm for atmospheric flows, AIAA J. 54 (4) (2016) 1370–1385.
[43] L. Yelash, A. Müller, M. Lukáčová-Medvid'ová, F. Giraldo, V. Wirth, Adaptive discontinuous evolution Galerkin method for dry atmospheric flow, J. Comput. Phys. 268 (2014) 106–133.
[44] D.S. Abdi, L.C. Wilcox, T.C. Warburton, F.X. Giraldo, A GPU-accelerated continuous and discontinuous Galerkin non-hydrostatic atmospheric model, Int. J. High Perform. Comput. Appl. 33 (1) (2019) 81–109.
[45] X. Chen, N. Andronova, B.V. Leer, J.E. Penner, J.P. Boyd, C. Jablonowski, S.-J. Lin, A control-volume model of the compressible Euler equations with a vertical Lagrangian coordinate, Mon. Weather Rev. 141 (7) (2013) 2526–2544.
[46] J.E. Guerra, P.A. Ullrich, A high-order staggered finite-element vertical discretization for non-hydrostatic atmospheric models, Geosci. Model Dev. 9 (5) (2016) 2007–2029.
[47] A. Navas-Montilla, I. Echeverribar, A family of well-balanced WENO and TENO schemes for atmospheric flows, J. Comput. Phys. 489 (2023) 112273.
[48] M. Andrés-Carcasona, M. Soria, E. García-Melendo, A. Miró, Toward a quantitative understanding of Robert's rising thermal bubble as a benchmark for 3D atmospheric codes, Mon. Weather Rev. 151 (2) (2023) 509–522.
[49] R.L. Holmes, G. Dimonte, B. Fryxell, M.L. Gittings, J.W. Grove, M. Schneider, D.H. Sharp, A.L. Velikovich, R.P. Weaver, Q. Zhang, Richtmyer–Meshkov instability growth: experiment, simulation and theory, J. Fluid Mech. 389 (1999) 55–79.
[50] Y. Zhou, R.J. Williams, P. Ramaprabhu, M. Groom, B. Thornber, A. Hillier, W. Mostert, B. Rollin, S. Balachandar, P.D. Powell, et al., Rayleigh–Taylor and Richtmyer–Meshkov instabilities: a journey through scales, Physica D 423 (2021) 132838.