Entropy conserving implicit time integration in a Discontinuous Galerkin solver in entropy variables

Entropy conserving implicit time integration in a Discontinuous Galerkin solver in entropy variables

A. Colombo[a], A. Crivellini[b,*], A. Nigro[b]

[a] *University of Bergamo,*
Department of Engineering and Applied Sciences
[b] *Marche Polytechnic University,*
Department of Industrial Engineering and Mathematical Sciences

Abstract

This article presents a fully discrete entropy conserving/stable method based on a Discontinuous Galerkin (DG) discretization in entropy variables coupled with a modified Crank-Nicolson scheme. The entropy conserving time integration is inspired by the work of LeFloch [1], originally developed in the context of a Finite Volume method in conservative variables. This entropy conserving time integrator is here adapted to a DG discretization in entropy variables also demonstrating the fulfilment of entropy conservation regardless of the time step size and the type of elements used (quadrangular or triangular elements, possibly with curved edges). The performance of the implicit method will be demonstrated by computing several inviscid flow problems, *i.e.*, the convection of an isentropic vortex, the double shear layer, the Kelvin-Helmholtz instability, the shedding flow past a triangular wedge, the Sod shock tube, the receding flow and the Taylor-Green vortex.

Keywords: Discontinuous Galerkin, generalized Crank-Nicolson, entropy conserving/stable discretizations, entropy variables.

1. Introduction

This paper presents an entropy conserving method for the numerical solution of the Euler equations in the context of a high-order Discontinuous Galerkin (DG) discretization. Hyperbolic systems of partial differential equations (PDEs), such as the one describing the behaviour of compressible inviscid flows, can admit several

*Corresponding author: a.crivellini@staff.univpm.it, Tel +39 0712204436

weak solutions. To properly select the physical relevant one an entropy function can be defined. This function is constant where the solution is smooth and it can only increase across singularities such as shock waves. The idea to embed this physical constraint in the numerical method is not new, see for example the seminal works of Tadmor *et al.* [2, 3], but has received increasing attention in recent years, in particular when applied in the context of high-order DG methods. This class of numerical methods is very attractive for Computational Fluid Dynamics (CFD) thanks to the ability to get high-accuracy on unstructured and hybrid grids [4] and the favourable dissipation and dispersion proprieties that makes DG well suited to the scale-resolving simulation of turbulent flows, *e.g.*, [5, 6, 7].

DG discretizations with entropy conserving capabilities usually follow two approaches: *i*) the use of Summation by Parts (SBP) operators, split forms of the Euler equations and the conservative set of variables, see the works of Gassner et al., *e.g.*, [8, 9]; *ii*) the use of a symmetrized form of the governing equations together with the set of entropy variables. The latter method, employed in this work, does not introduce any limitation at the continuum level but it requires to "exactly" evaluate all the integrals part of the discrete form. This can be numerically achieved by means of *over-integration*, *i.e.*, by computing integrals using quadrature formulas with a degree of exactness large enough to make negligible the integration errors of non-polynomial functions, see [10]. When numerical methods that use a piecewise discontinuous representations of the solution are considered, properly designed numerical flux functions that guarantee entropy conservation/stability must be used, see for example [11, 12, 13]. Most of them were proposed in the context of low-order Finite Volume (FV) entropy conserving schemes. Space-time DG methods can also be considered to devise entropy conserving schemes. However, in the authors' opinion, such approach is impractical, as all the time slabs are linked together by a centred temporal state (numerical flux in time), *e.g*, see Friedrich *et al.* [14]. For this reason, the method of lines is here considered, which implies the use of numerical fluxes and time integration schemes both having entropy conserving properties.

While the development of specifically designed numerical fluxes has been the subject of several works, less attention has been devoted in literature to the development of entropy conserving time integrators. Recent articles from Lozano [15, 16] show that, for a generic entropy function, explicit and fully implicit Runge-Kutta schemes introduce spurious entropy. The papers also indicate the Backward-Euler method as an entropy stable scheme, *i.e.*, the entropy evolution in time is monotone. Gouasmi *et al.* [17] also show that for both the BDF2 and the explicit Leap-Frog methods it is difficult to determine a priori the sign of entropy pro-

duction, while the explicit Forward Euler scheme is entropy unstable. In Colombo *et al.* [10] no clear statement about the entropy production of the linearly implicit Runge-Kutta schemes of the Rosenbrock type was given. Note that, differently from [15, 16, 17], in [10] the spatial discretization is based on a DG method with entropy variables and not on more "standard" FV spatial approximations.

Up to the authors knowledge, the only entropy conserving scheme available in the literature is a modified version of the Crank-Nicolson method. This modified scheme is due to LeFloch *et al.* [1] and is often refereed as *"Generalized Crank-Nicolson"* method. The scheme was originally developed in the context of FV but is considered impractical as it requires numerical quadrature to assemble the modified intermediate state which substitutes the algebraic mean in the residuals vector evaluation of the standard scheme. Gouasmi *et al.* [17] proposed a computationally efficient implementation of the method by using theoretical arguments which are very similar to those used in the development of entropy conserving flux functions. It is worth mentioning that a similar idea was already proposed in Subbareddy and Candler [18] to obtain a fully discrete FV scheme capable of preserving kinetic energy.

The extension of these results to DG discretizations is not straightforward but it can be considered of great interest. In fact, an efficient and high-order entropy stable numerical framework that is also essentially dissipation-free in time, is an excellent candidate for explicit LES as it allows a sharp control on the amount of artificial dissipation added by the subgrid-scale model [19]. This paper will report in detail how to implement the entropy conserving time integrator scheme in a DG modal solver, in particular when entropy variables are used. The change of variables, in fact, involves a projection error that must be carefully taken into account. Theoretical and numerical proofs will demonstrate that the approach is entropy conserving in the sense that the L_2-projection of the entropy variables on the conservative ones results in a "global" entropy conservation up to machine precision irrespective of the time step size. The main features of the proposed method are: *i*) the use of a couple of L_2-projections between the entropy and the conservative variables and vice versa; *ii*) to retain the time-derivative of the conservative variables in the governing equations instead of using the time derivative of the entropy ones. This is done even though the solution is sought in terms of the degrees of freedom of the entropy variables, a fundamental difference from the approach followed by the authors in [10]. As in this work the conservation form of the Euler equations is considered, feature *ii*) guarantees a conservative discretization.

In this paper, together with the implementation details of the method, numerical results for several test cases will be presented to assess the performance of the

entropy conserving framework regardless of the number and the type of elements, the spatial order of accuracy and the time step size. Results will be also presented for the entropy stable method, obtained by considering properly designed numerical flux functions, and for "standard" time-integration schemes.

2. The governing equations

The set of equations governing the behaviour of inviscid flows, *i.e.*, the Euler equations, can be written for the compressible case as

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

where ρ is the fluid density, $\mathbf{u} = \{u_1, \dots, u_d\}$ the velocity vector, E and H the total energy and enthalpy, and $i, j = 1, \dots, d$, where d is the number of geometrical dimensions. For a perfect gas, the pressure p is given by $p = (\gamma - 1)\,\rho\left[E - (u_i u_i)/2\right]$, where $\gamma = c_p/c_v$ is the ratio of gas specific heats, here set to 1.4. In compact form the system (1) can be written as

$$\frac{\partial \mathbf{q}}{\partial t} + \frac{\partial \mathbf{F}_i(\mathbf{q})}{\partial x_i} = \mathbf{0}, \tag{2}$$

with implied summation on index i and where $\mathbf{q} = \{\rho, \rho u_i, \rho E\}$ is the vector of the conservative variables and $\mathbf{F} \in \mathbb{R}^{2+d} \otimes \mathbb{R}^d$ is the convective flux function.

2.1. The entropy framework

The concept of entropy conservation relies on the existence of a convex function, the *generalized entropy function* $\mathcal{S}$, and the scalar-valued *entropy flux* functions $\mathcal{U}_i$, such that the following compatibility conditions holds

$$\frac{\partial \mathcal{S}(\mathbf{q})}{\partial \mathbf{q}} \frac{\partial \mathbf{F}_i(\mathbf{q})}{\partial \mathbf{q}} = \frac{\partial \mathcal{U}_i(\mathbf{q})}{\partial \mathbf{q}} \qquad \text{with} \qquad i = 1, \dots, d. \tag{3}$$

By left-multiplying Eq. (2) by $\partial \mathcal{S}(\mathbf{q})/\partial \mathbf{q}$ we obtain

$$\frac{\partial \mathcal{S}(\mathbf{q})}{\partial t} + \frac{\partial \mathcal{U}_i(\mathbf{q})}{\partial x_i} = 0, \tag{4}$$

with implied summation on index i. Since entropy needs to be dissipated at shock waves, the above identity is replaced by the following inequality

$$\frac{\partial \mathcal{S}(\mathbf{q})}{\partial t} + \frac{\partial \mathcal{U}_i(\mathbf{q})}{\partial x_i} \leq 0, \tag{5}$$

that, assuming periodic boundary conditions, can be integrated in space giving

$$\frac{d}{dt} \int_{\mathbb{R}^d} \mathcal{S}(\mathbf{q}) d\mathbf{x} \leq 0. \tag{6}$$

Note that $\mathcal{S}$ differs from the common thermodynamic definition of entropy, which increases across shocks.

By assuming that $\mathcal{S}$ is strictly convex, the mapping $\mathbf{q} \to \partial \mathcal{S}(\mathbf{q})/\partial \mathbf{q}$ can be regarded as a change of variables from the conservative variables $\mathbf{q}$ to the entropy variables $\mathbf{v} = \partial \mathcal{S}(\mathbf{q})/\partial \mathbf{q}$. The system of governing equations (2) is symmetrized when rewritten in terms of entropy variables as

$$\frac{\partial \mathbf{q}(\mathbf{v})}{\partial \mathbf{v}} \frac{\partial \mathbf{v}}{\partial t} + \frac{\partial \mathbf{F}_i(\mathbf{q}(\mathbf{v}))}{\partial \mathbf{v}} \frac{\partial \mathbf{v}}{\partial x_i} = \mathbf{0}, \tag{7}$$

where $\partial \mathbf{q}(\mathbf{v})/\partial \mathbf{v}$ is positive-definite.

According to Huges *et al.* [20] the only *generalized entropy function-entropy flux pair*, up to a multiplicative constant, which symmetrizes the viscous term in the compressible Navier–Stokes equations is

$$\mathcal{S} = -\frac{\rho s}{\gamma - 1}, \quad \mathcal{U}_i = \mathcal{S} u_i, \quad \text{whith} \quad i = 1, \dots, d, \tag{8}$$

where $s = \ln(p\rho^{-\gamma})$ is the thermodynamic entropy. The corresponding set of entropy variables is

$$\mathbf{v} = \left\{ \frac{\gamma - s}{\gamma - 1} - \frac{\rho}{2p} |\mathbf{u}|^2, \ \frac{\rho u_i}{p}, \ -\frac{\rho}{p} \right\}. \tag{9}$$

Since $\partial \mathbf{q}(\mathbf{v})/\partial \mathbf{v}$ and $\partial \mathbf{F}_i(\mathbf{q}(\mathbf{v}))/\partial \mathbf{v}$ are symmetric, there exists an *entropy potential*, ϑ, *entropy flux potential*, ψ_i, *pair* such that $\partial \vartheta/\partial \mathbf{v} = \mathbf{q}(\mathbf{v})$ and $\partial \psi_i/\partial \mathbf{v} = \mathbf{F}_i(\mathbf{q}(\mathbf{v}))$. It can be easily proved that $\vartheta = \mathbf{v} \cdot \mathbf{q}(\mathbf{v}) - \mathcal{S}(\mathbf{q}(\mathbf{v}))$ and $\psi_i = \mathbf{v} \cdot \mathbf{F}_i(\mathbf{q}(\mathbf{v})) - \mathcal{U}_i(\mathbf{q}(\mathbf{v}))$,

and that they reduce to $\vartheta = \rho$ and $\psi_i = \rho u_i$ when the generalized entropy function of Eq. (8) is used.

In this work, following the seminal work of Huges *et al.* [20], the system of governing equations (7) is solved by directly approximating the entropy variables $\mathbf{v}$ in the discrete space. To fulfil entropy conservation at the discrete level, the approach assumes that integrals must be computed exactly [21]. However, this statement can be relaxed as shown in [10]. Indeed, numerical results demonstrate that if integrals are approximated by using "accurate enough" quadrature rules, the entropy conservation is verified.

3. The Generalized Crank-Nicolson scheme in the DG framework

In this work, when "standard" time integration schemes are considered, they are applied to the governing equations written in the following form

$$\frac{\partial \mathbf{q}(\mathbf{v})}{\partial \mathbf{v}} \frac{\partial \mathbf{v}}{\partial t} + \frac{\partial \mathbf{F}_i(\mathbf{q}(\mathbf{v}))}{\partial x_i} = \mathbf{0}. \tag{10}$$

By applying the classical Crank-Nicolson (SCN) scheme the system of semi-discrete governing equations becomes

$$\frac{\partial \mathbf{q}(\mathbf{v}^{n+1/2})}{\partial \mathbf{v}} \frac{\mathbf{v}^{n+1} - \mathbf{v}^n}{\Delta t} + \frac{\partial \mathbf{F}_i(\mathbf{q}(\mathbf{v}^{n+1/2}))}{\partial x_i} = \mathbf{0}, \tag{11}$$

where $\mathbf{v}^{n+1/2}$ is defined according to the algebraic mean of the working variables, *i.e.*, $1/2(\mathbf{v}^{n+1} + \mathbf{v}^n)$. Equation (11) is not in a conservative form due to the Jacobian of the change of variables, $\partial \mathbf{q}(\mathbf{v}^{n+1/2})/\partial \mathbf{v}$.

LeFloch *et al.* [1] designed the following entropy conserving time integration scheme

$$\frac{\mathbf{q}^{n+1} - \mathbf{q}^n}{\Delta t} + \frac{\partial \mathbf{F}_i(\mathbf{q}(\mathbf{v}^{n+1/2}))}{\partial x_i} = \mathbf{0}, \tag{12}$$

where

$$\mathbf{v}^{n+1/2}\left(\mathbf{q}^n, \mathbf{q}^{n+1}\right) = \int_0^1 \mathbf{v}\left((1-\chi)\mathbf{q}^n + \chi \mathbf{q}^{n+1}\right) d\chi. \tag{13}$$

The scheme was proposed in the context of FV entropy conserving schemes for non-linear systems of equations. Following the Tadmor [2] terminology this method will be referred as "Generalized Crank-Nicolson" (GCN). It is proven that when

GCN is coupled with an entropy conserving flux function, the resulting FV discretization is entropy conserving both in space and in time. However, it is not obvious how to extend this result to a high-order DG discretization in entropy variables. In fact, in Eq. (10) the set of working variables is not the conservative but the entropy one and the time derivative term is multiplied by the Jacobian matrix $\partial \mathbf{q}(\mathbf{v})/\partial \mathbf{v}$. Moreover, the projection between the entropy variables, $\mathbf{v}$, and the conservative ones, $\mathbf{q}$ (and vice versa) involves an approximation error. To solve these issues Eq. (7) has been recast in the traditional conservative form as

$$\frac{\partial \mathbf{q}(\mathbf{v})}{\partial t} + \frac{\partial \mathbf{F}_i(\mathbf{v})}{\partial x_i} = \mathbf{0}, \tag{14}$$

where, with an abuse of notation, $\mathbf{F}_i(\mathbf{v})$ is used in place of $\mathbf{F}_i(\mathbf{q}(\mathbf{v}))$. The time discretization of Eq. (14) gives

$$\frac{\mathbf{q}(\mathbf{v}^{n+1}) - \mathbf{q}(\mathbf{v}^n)}{\Delta t} + \frac{\partial \mathbf{F}_i(\mathbf{v}^{n+1/2})}{\partial x_i} = \mathbf{0}. \tag{15}$$

The system of governing equations is now discretized in space by multiplying Eq. (15) by an arbitrary smooth test function $\mathbf{w} = \{w_1, \dots, w_{d+2}\}$ and integrating by parts, to obtain the weak formulation of each $k-th$ scalar equation

$$\begin{aligned} \int_\Omega w_k \frac{q_k(\mathbf{v}^{n+1}) - q_k(\mathbf{v}^n)}{\Delta t} d\Omega &= \int_\Omega \frac{\partial w_k}{\partial x_i} F_{k,i}\left(\mathbf{v}^{n+1/2}\right) d\Omega \\ &- \int_{\partial\Omega} w_k F_{k,i}\left(\mathbf{v}^{n+1/2}\right) n_i d\sigma, \end{aligned} \tag{16}$$

where $\Omega \in \mathbb{R}^d$, with $d \in \{2, 3\}$, is the domain, $\partial\Omega$ its boundary and $\mathbf{n} = \{n_1, \dots, n_d\}$ the unit vector normal to the boundary. In this work, when used as a subscript, the symbol k is an index spanning the range $1, \dots, 2 + d$, otherwise it will represent the degree of polynomial functions.

Denoting by $\mathcal{K}_h = \{K\}$ a mesh of the domain made of elements K, the solution in conservative, $\mathbf{q}$, and entropy, $\mathbf{v}$, variables, together with the test function $\mathbf{w}$, are replaced with finite element approximations, $\mathbf{q}_h$ and $\mathbf{v}_h$, and a discrete test function $\mathbf{w}_h$, all of them belonging to the discrete polynomial space $[\mathbb{P}^k_d(\mathcal{K}_h)]^{2+d}$. Each component $q_{h,k}$ of the numerical solution can be expressed, in terms of the elements of the global vector $\mathbf{Q}$ of unknown degrees of freedom, as $q_{h,k} = \phi_j Q_{k,j}$, with $j = 1, \dots, \left(N_{dof}\right)\text{card}(\mathcal{K}_h)$, where $N_{dof} = \prod_{i=1,d}(k + i/i)$ and ϕ_j belongs to the set of orthogonal and hierarchical basis functions defined according to [4]. For the

sake of notation, the set $\mathcal{F}_h \overset{\text{def}}{=} \mathcal{F}_h^i \cup \mathcal{F}_h^b$ of mesh faces is defined, where $\mathcal{F}_h^b$ collects the faces lying on the boundary of Ω_h. For any $F \in \mathcal{F}_h^i$ there exist two elements $K^+, K^- \in \mathcal{K}_h$ such that $F \in \partial K^+ \cap \partial K^-$. For all $F \in \mathcal{F}_h$, $\mathbf{n}_F$ is the normal unit vector pointing from K^+ to K^-, for the sake of notation compactness, the subscript F will be dropped in the following. Since a function $w_h \in \mathbb{P}_d^k(\mathcal{K}_h)$ is double valued over an internal face $F \in \mathcal{F}_h^i$, the jump trace operator $[\![w_h]\!] \overset{\text{def}}{=} w_{h|K^+} - w_{h|K^-}$ is defined. This operator acts componentwise when applied to a vector. The DG discretization of the Euler equations results in seeking, for $k = 1, \ldots, 2+d$, the elements of $\mathbf{Q}$ such that

$$\begin{aligned}\sum_{K\in\mathcal{K}_h} \int_K \phi_i \phi_j \frac{Q_{k,j}^{n+1} - Q_{k,j}^n}{\Delta t} d\Omega = &\sum_{K\in\mathcal{K}_h} \int_K \frac{\partial \phi_i}{\partial x_l} F_{k,l}\left(\mathbf{v}_h^{n+1/2}\right) d\Omega \\ &- \sum_{F\in\mathcal{F}_h} \int_F [\![\phi_i]\!]\, \widehat{F}_k\left(\mathbf{v}_h^{\pm n+1/2}, \mathbf{n}\right) d\sigma.\end{aligned} \tag{17}$$

To demonstrate the conserving/stability properties of the present scheme we focus firstly on the discrete form of the unsteady term of Eq. (4) resulting from the LHS of (17). $Q_{k,j}^n$ and $Q_{k,j}^{n+1}$ are evaluated as the L_2-projection of the conservative variables (computed according to the entropy variables) on the DG polynomial space

$$Q_{k,i}^n = [(\mathbf{M}^K)^{-1}]_{i,j} \int_K \phi_j q_k(\mathbf{v}_h^n) d\Omega, \tag{18}$$

where $\mathbf{M}_{i,j}^K = \int_K \phi_i \phi_j d\Omega$ is the mass matrix of the element K. The projected conservative variables $\widehat{q} \in [\mathbb{P}_d^k(\mathcal{K}_h)]^{2+d}$ are then defined as

$$\widehat{q}_{h,k}^n = \phi_j Q_{k,j}^n. \tag{19}$$

Similarly,

$$V_{k,i}^{n+1/2} = [(\mathbf{M}^K)]_{i,j}^{-1} \int_K \phi_j v_k^{n+1/2} d\Omega, \tag{20}$$

and

$$v_{h,k}^{n+1/2} = \phi_j V_{k,j}^{n+1/2}. \tag{21}$$

When $v_{h,k}^{n+1/2} = \phi_i V_{k,i}^{n+1/2}$ are used as the test functions in place of ϕ_i in Eq. (17), for each $K \in \mathcal{K}_k$ the unsteady term can be re-formulated as

$$\begin{aligned}
&\int_K \phi_i V_{k,i}^{n+1/2} \phi_j \frac{Q_{k,j}^{n+1} - Q_{k,j}^{n}}{\Delta t} d\Omega = \left(\int_K \phi_i \phi_j d\Omega \right) V_{k,i}^{n+1/2} \frac{Q_{k,j}^{n+1} - Q_{k,j}^{n}}{\Delta t} = \\
&= \mathbf{M}_{j,i}^{K} [(\mathbf{M}^K)^{-1}]_{i,m} \left(\int_K \phi_m v_k^{n+1/2} d\Omega \right) \frac{Q_{k,j}^{n+1} - Q_{k,j}^{n}}{\Delta t} = \\
&= \delta_{j,m} \left(\int_K \phi_m v_k^{n+1/2} d\Omega \right) \frac{Q_{k,j}^{n+1} - Q_{k,j}^{n}}{\Delta t} = \int_K v_k^{n+1/2} \phi_j \frac{Q_{k,j}^{n+1} - Q_{k,j}^{n}}{\Delta t} d\Omega = \\
&= \int_K v_k^{n+1/2} \frac{\widehat{q}_{h,k}^{n+1} - \widehat{q}_{h,k}^{n}}{\Delta t} d\Omega,
\end{aligned} \tag{22}$$

where $\delta_{j,m}$ is the Kronecker delta function. The LeFloch's [1] identity states that

$$v_k^{n+1/2} \left(\widehat{q}_{h,k}^{n+1} - \widehat{q}_{h,k}^{n} \right) = \mathcal{S}(\widehat{\mathbf{q}}_h^{n+1}) - \mathcal{S}(\widehat{\mathbf{q}}_h^{n}), \tag{23}$$

and here it holds true pointwise if $\mathbf{v}^{n+1/2}$ is defined according to the projection of the entropy variables on the conservative set, *i.e.*, $\widehat{\mathbf{q}}_h^n$ and $\widehat{\mathbf{q}}_h^{n+1}$. In other words, Eq. (23) is valid when the intermediate state is

$$v_k^{n+1/2} \left(\widehat{\mathbf{q}}_h^n, \widehat{\mathbf{q}}_h^{n+1} \right) = \int_0^1 v_k \left((1-\chi) \widehat{\mathbf{q}}_h^n + \chi \widehat{\mathbf{q}}_h^{n+1} \right) d\chi. \tag{24}$$

For the sake of compactness, in the following, $v_k^{n+1/2} \left(\widehat{\mathbf{q}}_h^n, \widehat{\mathbf{q}}_h^{n+1} \right)$ will be denoted as $v_k^{n+1/2}$, accordingly, $\mathbf{v}^{n+1/2}$ was used in place of $\mathbf{v}^{n+1/2} \left(\mathbf{q}^n, \mathbf{q}^{n+1} \right)$. Equation (24) implies that the last integral of Eq. (22) is equivalent to

$$\int_K \frac{\mathcal{S}(\widehat{\mathbf{q}}_h^{n+1}) - \mathcal{S}(\widehat{\mathbf{q}}_h^{n})}{\Delta t} d\Omega, \tag{25}$$

which means that the PDE for the entropy evolution is implicitly discretized in a conservative form and the fully discrete scheme is entropy conserving. Moreover, since the conservative form of the governing equations is used, the scheme is fully conservative in time, *i.e.*, mass, momentum and energy are conserved at the discrete level. It is worth noting that $\mathcal{S}(\mathbf{q}(\mathbf{v}_h^n))$ is different from $\mathcal{S}(\widehat{\mathbf{q}}_h^n)$. In fact, the peculiarity of the scheme is that while $V_{k,j}^{n+1}$ are the DOFs, the scheme is entropy

conserving in the sense that the projection on the conservative set is entropy conserving, so the entropy solution is based on the $Q_{k,j}^{n+1}$ coefficients. However, during the simulations Eqs. (18) and (19) are only used for the evaluation of Eq. (24) since Eq. (22) is solved implicitly. Notice that, the L_2-projection procedure does no alter the order of accuracy of the method, neither in space nor in time.

From the implementation point of view, Eq. (24) requires numerical integration and to properly select the degree of exactness of the quadrature rules to guarantee entropy conservation, a value difficult to be estimated *a priori*. A viable alternative was proposed by Gouasmi *et al.* [17] who found a close form solution of Eq. (23). The idea behind this method is very similar to that used for the derivation of entropy conserving numerical flux functions. The intermediate state $\mathbf{v}^{n+1/2}$ of Gouasmi *et al.* is computed as

$$\begin{aligned} v_1^{n+1/2} &= \frac{1}{\gamma-1}\left(\gamma\frac{\overline{\rho}}{\overline{\rho^{ln}}} - \overline{s_h}\right) - \overline{u_{i,h}}v_{1+i}^{n+1/2} - \frac{1}{2}\overline{u_{i,h}u_{i,h}}\,v_{2+d}^{n+1/2}, \\ v_{1+i}^{n+1/2} &= -\overline{u_{i,h}}v_{2+d}^{n+1/2} \\ v_{2+d}^{n+1/2} &= -\frac{\overline{\rho_h}}{\overline{p_h^{\ln}}}, \end{aligned} \tag{26}$$

where $i = 1, \ldots, d$ and where $p_h = (\gamma-1)[(\rho E)_h - (\rho u_i)_h(\rho u_i)_h/(2\rho_h)]$, $u_{i,h} = (\rho u_i)_h/\rho_h$ and $s_h = \ln(p_h\rho_h^{-\gamma})$ are evaluated using the projected solutions $\widehat{\mathbf{q}}_h^n$ and $\widehat{\mathbf{q}}_h^{n+1}$. The arithmetic and logarithmic means are defined as

$$\overline{a_h} = \frac{a_h^{n+1} + a_h^n}{2}, \quad \overline{a_h^{\ln}} = \frac{a_h^{n+1} - a_h^n}{\ln(a_h^{n+1}) - \ln(a_h^n)}. \tag{27}$$

Gouasmi *et al.* [17] also proved that the resulting scheme is second-order accurate in time. In the following this method will be denoted as GCNG. Here, thanks to the use of orthonormal shape functions, see [4] for the details, the implementation strongly simplifies reducing the computational cost for the evaluation of the intermediate state $\mathbf{v}_h^{n+1/2}$.

Friedrich *et al.* [14] used a similar approach to derive an entropy conservative *temporal state* for their space-time DG algorithm. Their method takes advantage of the SBP property, it uses the conservative variable set and does not require *over-integration*. Their *temporal state* is essentially a central flux function defined in time that links together all the time slabs. The resulting space-time entropy

conservative scheme is not a real option since the only viable choice is a pure upwind *temporal state* which decouples the time slabs. However, in this case the scheme is "only" entropy stable.

Concerning the RHS of (17), the linear combination with the coefficients $V_{k,i}^{n+1/2}$, where $k = 1, \dots, 2+d$ and $i = 1, \dots, \left(N_{dof}\right) \text{card}(\mathcal{K}_h)$, gives

$$\sum_{K \in \mathcal{K}_h} \int_K \frac{\partial v_k^{n+1/2}}{\partial x_l} F_{k,l}\left(\mathbf{v}_h^{n+1/2}\right) d\Omega - \sum_{F \in \mathcal{F}_h} \int_F \left[\!\left[v_{h,k}^{n+1/2} \right]\!\right] \widehat{F}_k\left(\mathbf{v}_h^{\pm n+1/2}, \mathbf{n}\right) d\sigma. \quad (28)$$

By defining the potential flux as $\psi_l = v_k F_{k,l} - \mathcal{U}_l$, it can be shown that $\partial v_k/\partial x_l F_{k,l} = \partial \mathcal{U}_l/\partial x_l + \partial \psi_l/\partial x_l - v_k \partial F_{k,l}/\partial x_l$, cf. [21]. Moreover, since the compatibility condition holds, *i.e.*, $\partial \mathcal{U}_l/\partial x_l = v_k \partial F_{k,l}/\partial x_l$, it is possible to write $\partial v_k/\partial x_l F_{k,l} = \partial \psi_l/\partial x_l$. For the sake of notation compactness, in the above equation the dependence on $\mathbf{v}_h^{n+1/2}$ was omitted, and in the following the discrete potential flux $\psi_l(\mathbf{v}_h^{n+1/2})$ will be denoted as $\psi_{h,l}^{n+1/2}$. By substituting the previous result into Eq. (28) the following relation is obtained

$$\sum_{K \in \mathcal{K}_h} \int_K \frac{\partial \psi_{h,l}^{n+1/2}}{\partial x_l} d\Omega - \sum_{F \in \mathcal{F}_h} \int_F \left[\!\left[v_{h,k}^{n+1/2} \right]\!\right] \widehat{F}_k\left(\mathbf{v}_h^{\pm n+1/2}, \mathbf{n}\right) d\sigma, \quad (29)$$

which, by using the divergence theorem, becomes

$$\sum_{F \in \mathcal{F}_h} \int_F \left[\left[\!\left[\psi_{h,l}^{n+1/2} n_l \right]\!\right] - \left[\!\left[v_{h,k}^{n+1/2} \right]\!\right] \widehat{F}_k\left(\mathbf{v}_h^{\pm n+1/2}, \mathbf{n}\right) \right] d\sigma. \quad (30)$$

As an entropy conserving numerical flux function fulfils the following relation, cf. [21],

$$\left[\!\left[\psi_{h,l}^{n+1/2} n_l \right]\!\right] - \left[\!\left[v_{h,k}^{n+1/2} \right]\!\right] \widehat{F}_k\left(\mathbf{v}_h^{\pm n+1/2}, \mathbf{n}\right) = 0, \quad (31)$$

Eq. (30) is identically zero and the scheme is entropy conserving in space. Similarly, using an entropy stable numerical flux where

$$\left[\!\left[\psi_{h,l}^{n+1/2} n_l \right]\!\right] - \left[\!\left[v_{h,k}^{n+1/2} \right]\!\right] \widehat{F}_k\left(\mathbf{v}_h^{\pm n+1/2}, \mathbf{n}\right) \geq 0, \quad (32)$$

an entropy stable space DG discretization is obtained. Notice that, these results assume an exact evaluation of all the integrals. This result is not new, see [22], but it is here demonstrated on a different perspective. Together with Eq. (25) it proves

that the proposed method implicitly discretizes the entropy conservation/inequality of Eq. (6),

$$\sum_{K\in\mathcal{K}_h}\int_K \frac{\mathcal{S}(\widehat{\mathbf{q}}_h^{n+1})-\mathcal{S}(\widehat{\mathbf{q}}_h^{n})}{\Delta t}d\Omega = \sum_{F\in\mathcal{F}_h}\int_F \left[\left[\left[\psi_{h,l}^{n+1/2}n_l\right]\right]-\left[\left[v_{h,k}^{n+1/2}\right]\right]\widehat{F}_k\left(\mathbf{v}_h^{\pm n+1/2},\mathbf{n}\right)\right]d\sigma \geq 0. \tag{33}$$

Finally, the analogy between Eq. (31) and Eq. (23) highlights the similarity between the entropy conserving flux functions, $\widehat{\mathbf{F}}(\mathbf{v}^{\pm n+1/2},\mathbf{n})$, and the intermediate state in time, $\mathbf{v}^{n+1/2}$.

3.1. Implementation details

Numerical integration of Eq. (17) by means of suitable Gauss quadrature leads to the following discrete system of non-linear equations

$$\mathbf{M}\frac{\mathbf{Q}(\mathbf{V}^{n+1})-\mathbf{Q}(\mathbf{V}^{n})}{\Delta t}+\mathbf{R}(\mathbf{V}^{n+1/2})=\mathbf{0}, \tag{34}$$

where, according to Eq. (20) and Eq. (24) or Eq. (27), $\mathbf{V}^{n+1/2}$ is a non-linear function of $\mathbf{V}^{n+1}$ and $\mathbf{V}^{n}$.

Eq. (34) is here solved with a Newton-Kyrolv algorithm. For the sake of clarity, we will use here the notation $\mathbf{V}^{n+1/2}=\widehat{\mathbf{V}}(\mathbf{V},\mathbf{V}^n)$, where $\mathbf{V}$ can be the actual solution vector of the Newton algorithm, $\mathbf{V}^{(i)}$, or the solution vector at the end of the time step, $\mathbf{V}^{n+1}$. The, possibly inexact, i-th Newton step finds $\Delta\mathbf{V}^{(i)}=(\mathbf{V}^{(i+1)}-\mathbf{V}^{(i)})$ such that

$$\left(\frac{\mathbf{M}}{\Delta t}\frac{\partial\mathbf{Q}(\mathbf{V}^{(i)})}{\partial\mathbf{V}}+\frac{\partial\mathbf{R}}{\partial\widehat{\mathbf{V}}}\frac{\partial\widehat{\mathbf{V}}}{\partial\mathbf{V}}\right)\Delta\mathbf{V}^{(i)}=-\mathbf{M}\frac{\mathbf{Q}(\mathbf{V}^{(i)})-\mathbf{Q}(\mathbf{V}^{n})}{\Delta t}-\mathbf{R}, \tag{35}$$

where the dependence of $\mathbf{R}$ and $\partial\mathbf{R}/\partial\widehat{\mathbf{V}}$ on $\widehat{\mathbf{V}}(\mathbf{V}^{(i)},\mathbf{V}^n)$ was omitted for the sake of notation compactness.

The matrix $\partial\widehat{\mathbf{V}}/\partial\mathbf{V}$ is a block-diagonal matrix which couples all the DOFs of an element and, particularly for the GCN, is not trivial to be derived and implemented. For this reason, this contribution to the implicit operator was approximated as $(1/2)\mathbf{I}$, where $\mathbf{I}$ is the identity matrix. The idea was to simply handle the term as in the standard Cranck-Nicolson scheme. According to our numerical experiments, the impact of this approximation on the non-linear convergence was

small, *i.e.*, the number of non-linear steps required by GCN or GCNG to reach a full, machine precision, convergence was almost the same as for the SCN, which uses an exact Jacobian matrix. An example will be given in Sec. 4.1.

In all the implicit time integration schemes used in this paper, the Newton algorithm starts with $\mathbf{V}^{(1)} = \mathbf{V}^n$ and stops when $||\Delta\mathbf{V}^{(k)}||_{L2}$ reaches the tolerance ε_{tol}. For the solution of the linear systems arising from both implicit and linearly implicit Rosenbrock-type schemes the preconditioned GMRES solvers available in the PETSc library are used [23].

4. Numerical results

The performance of the entropy conserving Crank-Nicolson scheme named GCNG in Sec. 3, in terms of accuracy of the solution and conservation property, has been assessed and compared to other time integrators, by computing the following inviscid flow problems: *i*) the isentropic vortex; *ii*) the double shear layer; *iii*) the Kelvin-Helmholtz instability; *iv*) the shedding flow past a triangular wedge; *v*) the Sod shock tube; *vi*) the receding flow; *vii*) the Taylor-Green vortex. For the first flow problem, which has an exact analytical solution, the accuracy will be measured in terms of the η error, while the conservation properties, evaluated for several test cases, will be evaluated in terms of the ε error, which are defined as follows

$$\eta(\circ, \bullet_{ref}) = (\Omega_h)^{-1/2} \, ||\circ - \bullet_{ref}||_{L^2}, \tag{36}$$

$$\varepsilon(\circ, \bullet_{ref}) = \Omega_h^{-1} \left(\int_{\Omega_h} \circ \, d\Omega - \int_{\Omega_h} \bullet_{ref} \, d\Omega \right), \tag{37}$$

where $\circ$ and $\bullet_{ref}$ are the numerical and the "reference" solutions, respectively. The reference value is set equal to the L_2–projection of the initial solution on the DG polynomial space.

The results obtained with the GCN and/or the GCNG schemes will be compared with the computations from other time integrators. In particular, with the acronym BE we denote the first-order accurate Backward Euler scheme, with SCN the second-order accurate standard Crank-Nicolson scheme, with BDF2 the second-order accurate Backward Differentiation Formulae, with ROS`XY` the linearly-implicit Rosenbrock-type Runge-Kutta schemes, where `X` is the order of accuracy and `Y` is the number of stages, with RK35 the third-order accurate Strong Stability Preserving explicit Runge-Kutta scheme with 5 stages, with FE the first-order accurate

explicit Forward Euler scheme. Details on the implementation and the efficient use of Rosenbrock-type schemes in the DG framework can be found in [24, 25].

For all the simulations using implicit time integration schemes, to avoid any influence of the time integration error, the non-linear solver tolerance ε_{tol} has been set to 10^{-13}, whereas the linear solver tolerance has been set to 10^{-2} to speed-up the iterative process. For the Rosenbrock-type schemes the iterative linear solver tolerance was set equal to 10^{-13}.

To obtain a fully discrete entropy conserving/stable scheme, dedicated numerical flux functions with entropy conserving/stable properties have been implemented. In particular, the entropy conserving (EC) numerical flux of Ismail and Roe [11], the kinetic energy preserving and entropy-conservative (KEEC) flux of Chandrashekar [12], the entropy stable numerical flux based on the entropy-consistent dissipation of Roe (ES), named EC1 in [11], and the entropy stable Godunov flux computed from the exact solution of the Riemann problem (ERS) [26] have been used. Following the findings of Colombo et al. [10], when simulations are performed with the entropy preserving fluxes, *over-integration* must be considered to ensure entropy conservation. However, in the authors' experience, its role becomes less relevant for simulations of simple flow problems that uses the ES and ERS fluxes. In practice, all the computations shown in this paper were performed using *over-integration*, with the exception of the isentropic vortex and the double shear layer (on Cartesian grids) cases.

In the next sections, when reported, the elemental CFL number is defined as $\mathrm{CFL}_K = (|\mathbf{u}_a| + c_a)\Delta t/\Delta x_K$, where $c_a = \sqrt{\gamma p_a/\rho_a}$ is the speed of sound and $\Delta x_K = d(V_K/S_K)$ is a measure of the cell size where V_K and S_K are the cell volume (area when $d = 2$) and surface area (perimeter when $d = 2$), respectively. The a subscript in the CFL definition means that variables are computed from elemental mean values.

4.1. The isentropic vortex convection problem

The convection of an inviscid isentropic vortex [27, 28, 29] is here considered to assess the accuracy and the entropy conservation properties of the Crank-Nicolson scheme named GCNG in Sec. 3, using the GCN scheme as reference.

The initial flow condition is given by

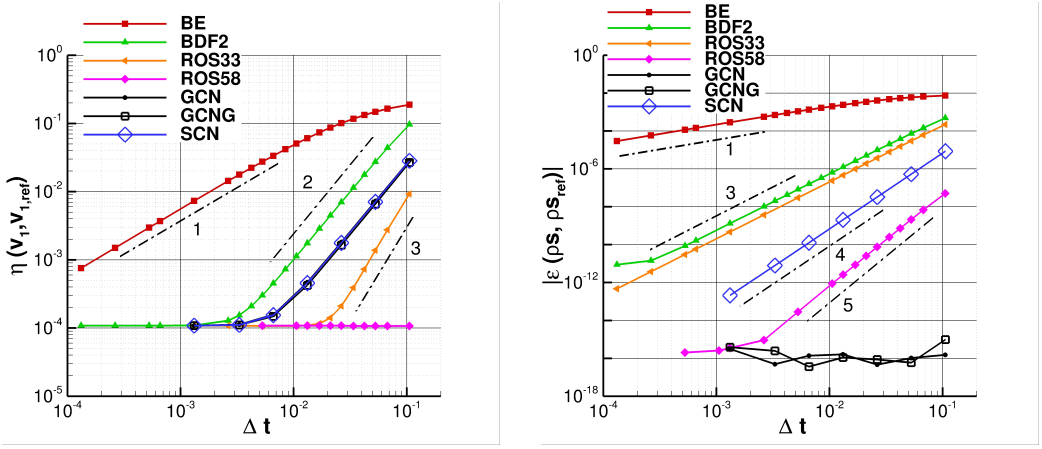
$$\begin{aligned} u_1 &= U - \frac{\alpha}{2\pi}\left(x_2 - \frac{L}{2}\right) e^{\beta\left(1-r^2\right)}, \\ u_2 &= U + \frac{\alpha}{2\pi}\left(x_1 - \frac{L}{2}\right) e^{\beta\left(1-r^2\right)}, \\ \Theta &= 1 - \frac{\alpha^2\left(\gamma-1\right)}{16\beta\gamma\pi^2} e^{2\beta\left(1-r^2\right)}, \\ p &= 1, \end{aligned} \tag{38}$$

where Θ is the temperature and the "free-stream" non dimensional velocity components are equal to $U = \sqrt{\gamma}$, corresponding to a Mach number $M_\infty \approx 1.4$. The symbol r is the distance of a generic point of the computational domain of coordinates, $\left(x_1, x_2\right)$, with respect to the vortex center, placed in the middle of the computational domain. The α and β values are set equal to 5 and $1/2$, respectively. The computational domain is $0 \leq x_i \leq L$, with $L = 10$, discretized with a fine and a coarse uniform mesh made of 50×50 and 25×25 quadrilateral elements, respectively. Boundary conditions are periodic and simulations are performed up to a final time T, corresponding to one period of vortex revolution.

Figure 1 shows a temporal convergence study performed using the fine grid, the $\mathbb{P}^3$ DG polynomial approximation and the EC flux. The largest time step size reported in the plots corresponds to CFL ≈ 3.8. Figure 1(a) shows the performance of the two entropy conserving Crank-Nicolson schemes compared with other implicit and linearly-implicit temporal integrator showing that all of them achieve their formal order of accuracy. Moreover, for each of the time step size considered, the schemes SCN, GCN and GCNG show the same η error values. As all the variables show similar behaviours, the η error for the first variable only is reported from now on. The ε error for ρs, reported in Fig. 1(b), shows that both the GCN and GCNG schemes reach the machine precision independently from the time step size, while all the other time integrators show convergence rates equal to or larger than the formal ones. The ε error is evaluated, for both the entropy conserving time integration schemes, by using the L_2-projection of the entropy variables on the conservative ones, cf. Sec. 3. Due to the definition of ε, see Eq. (37), the convergence rates can be mainly ascribed to the dissipation error. The measured rates are order 3 for BDF2, and order 4 for SCN, while BE, ROS33 and ROS58 show a rate equal to the theoretical one, *i.e.*, 1, 3 and 5, respectively. For the GCN scheme a Gaussian quadrature rule corresponding to a very high degree of exactness (27) was used to compute the intermediate state of Eq (13) in order to guarantee the entropy conservation independently from the time step size. Although a comparison

of the computational effort for the different time integrators is beyond the scope of this section, it is clear that the GCN always requires a larger CPU time than the GCNG. In fact, the GCNG replaces the explicit quadrature by the evaluation of much cheaper algebraic formulae. For example, when considering the small time step size $\Delta t = T/2\,560$, the use of a third-degree accurate quadrature formula allows to reach the zero machine precision value of $\mathcal{S}$ ($\mathbb{P}^3$, fine grid, EC flux). However, even using this low-degree rule, the GCN is ≈ 1.7 times slower than the GCNG. Increasing the degree of exactness to 27, the safe choice suitable for very large Δt sizes, the CPU time becomes ≈ 5.5 times larger.

When comparing the performance of the GCNG and SCN schemes, a similar number of inexact Newton's iterations, 7 vs. 6, and almost the same CPU time, the GCNG is 5% faster, is measured. These values prove that the the computational cost needed by the GCNG to compute $\mathbf{V}^{n+1/2}$, corresponding to the 12.6% of the overall CPU time, is counterbalanced by the assembly and inversion of the change-of-variables matrix used by the SCN scheme, cf. Eq. (11) and Eq. (15). Although these results may vary considerably, *e.g.*, depending on the Δt size and the degree of the polynomial approximation, the same conclusion can be drawn: the proposed GCNG scheme can be less computationally demanding than a standard method, such as the SCN.

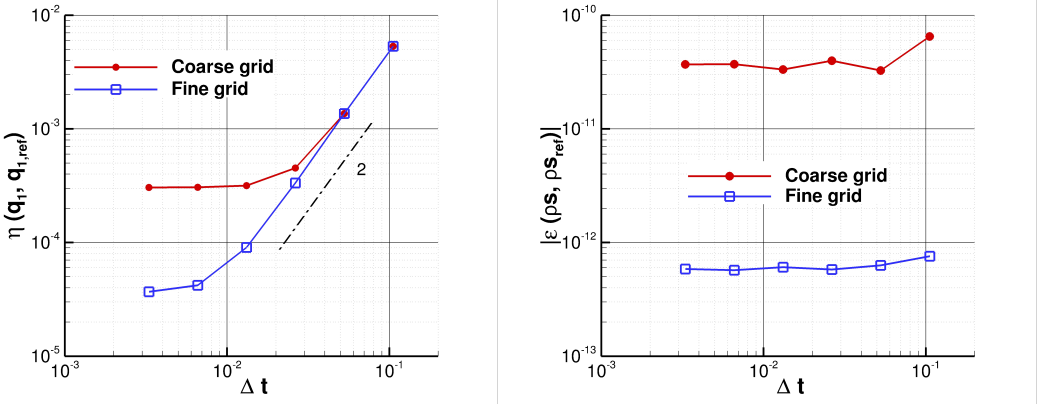


(a) $\eta\left(v_1, v_{1,ref}\right)$ (b) $\varepsilon\left(\rho s, \rho s_{ref}\right)$

Figure 1: Isentropic vortex problem – Time refinement study. Simulations performed on the fine grid using the $\mathbb{P}^3$ DG approximation, the EC numerical flux and several time integration schemes.

Figure 2 shows a time refinement study for both the coarse and fine grids using the $\mathbb{P}^3$ approximation. In particular, Fig. 2(a) displays η computed for the first

conservative variable, q_1, obtained trough the L_2–projection of the entropy set on the conservative one. For large time step sizes the computations on the two meshes show the same error values and achieve the formal order of accuracy of the scheme. When the time step is reduced the temporal error dominates the spatial one and different plateau values are reached. Notice that, the minimum error for q_1 on the fine mesh is lower than the one for v_1 shown in Fig. 1(a). The error on entropy in Fig. 2(b) is directly computed from the working variables and, as expected, does not verify the entropy preservation, although using a fully conserving scheme, cf. Sec. 3.

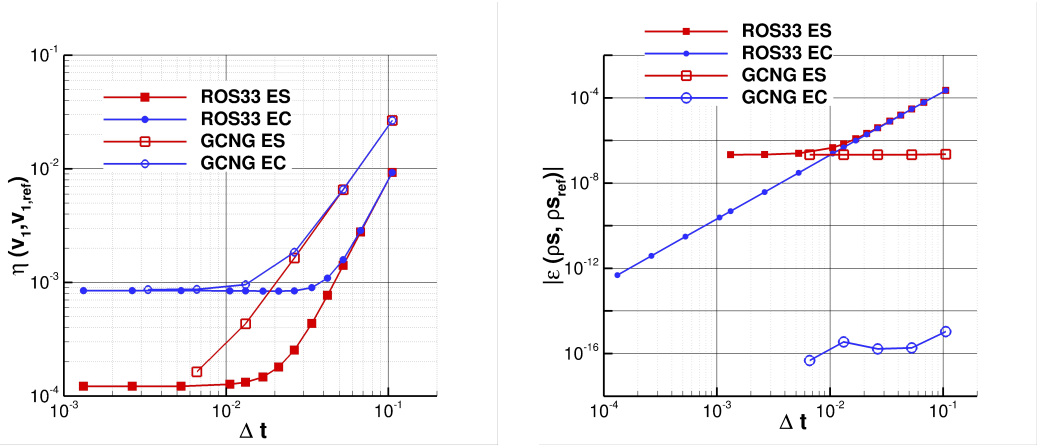


(a) $\eta\left(q_1, q_{1,ref}\right)$ (b) $\varepsilon\left(\rho s, \rho s_{ref}\right)$

Figure 2: Isentropic vortex problem – Time refinement study. Simulations performed on the fine and the coarse grid using the $\mathbb{P}^3$ DG approximation, the EC numerical flux and the GCNG time integration scheme.

Figure 3 reports a time refinement study performed on the coarse grid using the $\mathbb{P}^3$ DG approximation and the entropy conserving and entropy stable numerical fluxes, EC and ES, respectively. In this plot the results from the GCNG method are compared with those from the ROS33 scheme. The simulations performed with the ES flux do not use *over-integration*. In fact, for this flow problem, it was observed that even using a large number of quadrature points leads to the same results. As expected, in Fig. 3, when the spatial error overwhelms the temporal one, the same plateau value for η is obtained by the different time integrators for a given numerical flux. Although for a large enough time step size the ROS33 scheme reaches a significantly lower η error level than the GCNG, the linearly-implicit scheme cannot guarantee the entropy conservation for an arbitrary step

size but only for very small values, *i.e.*, when the temporal error is negligible with respect to the spatial one. On the contrary, the GCNG scheme reaches both the plateau values for ε, $\approx 10^{-7}$ for the ES flux and the machine precision for the EC flux, independently from the time step size, thus demonstrating the entropy preserving capability of the time integrator.



(a) $\eta\left(v_1, v_{1,ref}\right)$ (b) $\varepsilon\left(\rho s, \rho s_{ref}\right)$

Figure 3: Isentropic vortex problem – Time refinement study. Simulations performed on the coarse grid using the $\mathbb{P}^3$ DG approximation, the EC and ES numerical fluxes and the ROS33 and the GCNG time integration schemes.

Based upon to the findings of this section, among the two generalized CN methods considered in this work, only the GCNG scheme was used for the computation of the other test cases as it combines the entropy-conserving property with an affordable implementation suitable for production runs.

4.2. The inviscid double shear layer

The inviscid double shear layer [30, 31] flow problem is used to assess the conservation properties of entropy, kinetic energy and enstrophy. The kinetic energy is defined as $\kappa = \rho u_i u_i/2$, and the enstrophy as $\zeta = \omega^2$, where ω is the vorticity. Long time simulations performed with the GCNG time integration scheme, using different time step sizes and the EC and ES numerical fluxes, have been considered.

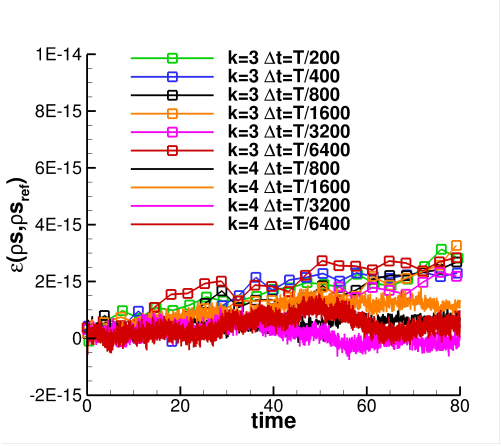
The initial flow condition is given by

$$
\begin{aligned}
u_1 &= \begin{cases} U \tanh\left[\left(x_2 - \pi/2\right)/\delta_1\right] & \text{if } x_2 \leq \pi, \\ U \tanh\left[\left(3\pi/2 - x_2\right)/\delta_1\right] & \text{if } x_2 > \pi, \end{cases} \\
u_2 &= U\delta_2 \sin(x_1), \\
p &= 1, \\
\rho &= 1,
\end{aligned}
\tag{39}
$$

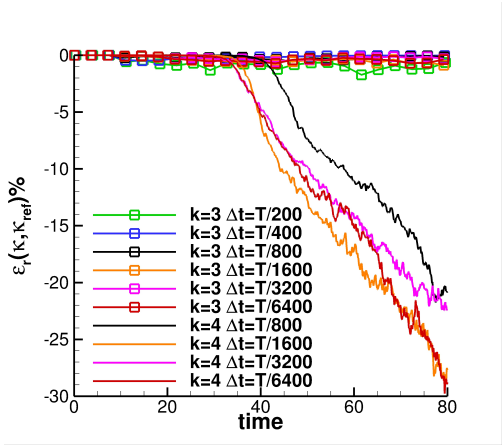
where $U = M_\infty\sqrt{\gamma}$, $\delta_1 = \pi/15$ and $\delta_2 = 0.05$, and, to obtain a nearly incompressible flow, $M_\infty = 0.01$. The computational domain $0 \leq x_i \leq 2\pi L$, with $L = 1$, has been discretized by: *i*) a 8×8 uniform Cartesian mesh; *ii*) an anisotropic and not uniform mesh made of 176 triangular elements; *iii*) a 8×8 mesh made of quadrangular elements with curved parabolic edges. At all the boundaries periodic conditions are imposed and the simulations are advanced in time up to $T = 80T_c$, where $T_c = L/U$ is the convective time. The T_c value is ten times larger than the one usually used in the literature.

Figure 4 shows the evolution in time of the errors on entropy, kinetic energy and enstrophy for the Cartesian grid using the GCNG time integration scheme coupled with the EC numerical flux and several time step sizes for the $\mathbb{P}^{3,4}$ DG approximations. To quantify the errors related to enstrophy and kinetic energy conservation, their relative percentage variation with respect to the reference value, *i.e.*, $\varepsilon_r(\zeta, \zeta_{ref})\%$ and $\varepsilon_r(\kappa, \kappa_{ref})\%$, is monitored. Figure 4(a) shows an error on entropy close to machine precision for any time step value, thus confirming the entropy conserving capability of the GCNG scheme. Plots 4(b) and 4(c) reveal that, for long time simulations, the even polynomial degree approximations show a significant dissipation for both κ and ζ when compared to the odd case, this odd/even effect was already observed and commented by the authors in [10]. This behaviour contributes to a worse stability of $\mathbb{P}^4$ than $\mathbb{P}^3$ computations, as testified by the divergent simulations for $\Delta t = T/200$ and $T/400$ (not shown in the plots). Note that, the simulations performed with $\Delta t = T/800$, which is the larger time step size for which both the odd and even DG approximations lead to a convergent solution, corresponds to CFL ≈ 25.

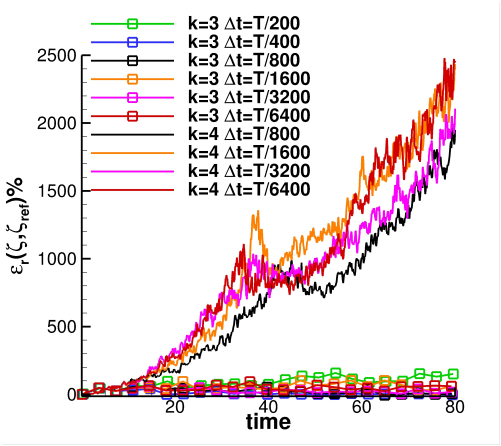
Furthermore, Fig. 4(b) shows that the relative percentage errors on kinetic energy for the $\mathbb{P}^3$ solutions are almost constant for the different time step sizes considered. Differently, Fig. 4(c) shows that, for the same DG approximation $\mathbb{P}^3$, the relative percentage error on enstrophy is greater influenced by the time step size, with error levels that are in the range $[2\%, 65\%]$, except for the 160% value obtained for the largest $\Delta t = T/200$ size.



(a) $\varepsilon\left(\rho s, \rho s_{ref}\right)$



(b) $\varepsilon_r\left(\kappa, \kappa_{ref}\right)\%$



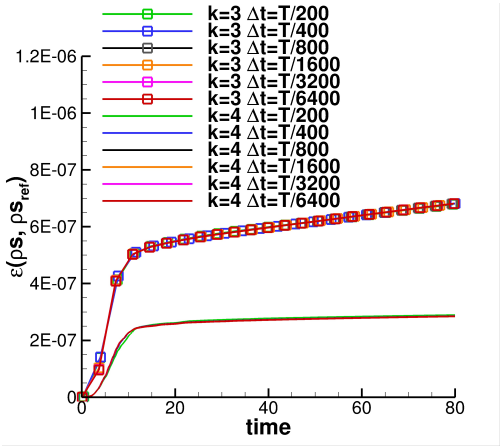
(c) $\varepsilon_r\left(\zeta, \zeta_{ref}\right)\%$

Figure 4: Double shear layers problem – Time refinement study. Simulations performed on the Cartesian grid using the $\mathbb{P}^{3,4}$ DG approximations, the EC numerical flux and the GCNG time integration scheme with different Δt values.

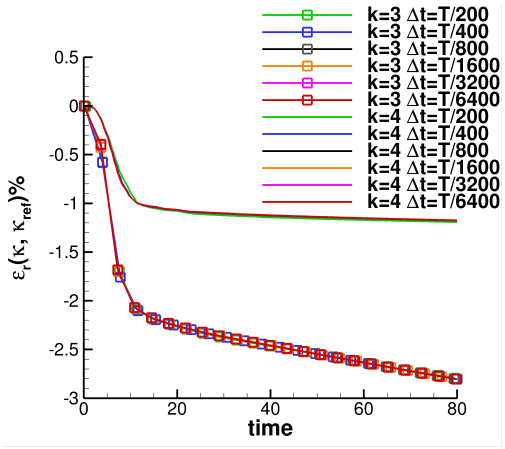
The same refinement study was performed by using the ES flux and results are summarized in Fig. 5. For the Cartesian grids only, entropy-stable simulations were performed without *over-integration* but verifying that a greater number of quadrature points led to the same results. As expected, cf. [10], computations are not affected by an odd/even effect and the $k = 4$ approximation shows, especially for entropy and kinetic energy, better conservation properties than the $k = 3$ case. Differently from the EC case, the use of the ES flux allowed to successfully perform

the computation for the $T/\Delta t = 200$ and $T/\Delta t = 400$ values.

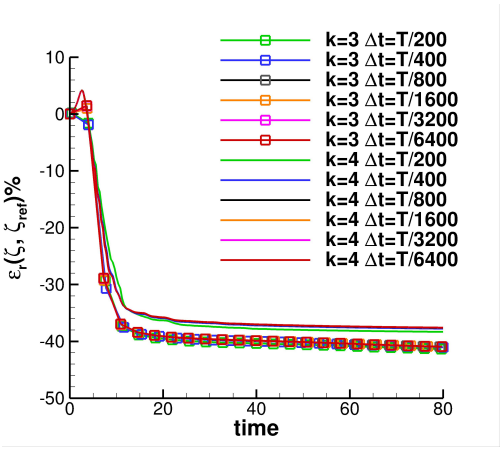
When using the ES flux combined with the GCNG time integrator, the conservation properties for a given DG approximation are roughly the same for all the time step sizes considered, thus further confirming the good properties of the GCNG. The pressure contours shown in Fig. 6 highlights the different level of accuracy achieved for the $\mathbb{P}^4$ DG approximation by using two different time steps sizes, *i.e.*, $T/\Delta t = 200$ and $T/\Delta t = 6\,400$.



(a) $\varepsilon\left(\rho s, \rho s_{ref}\right)$

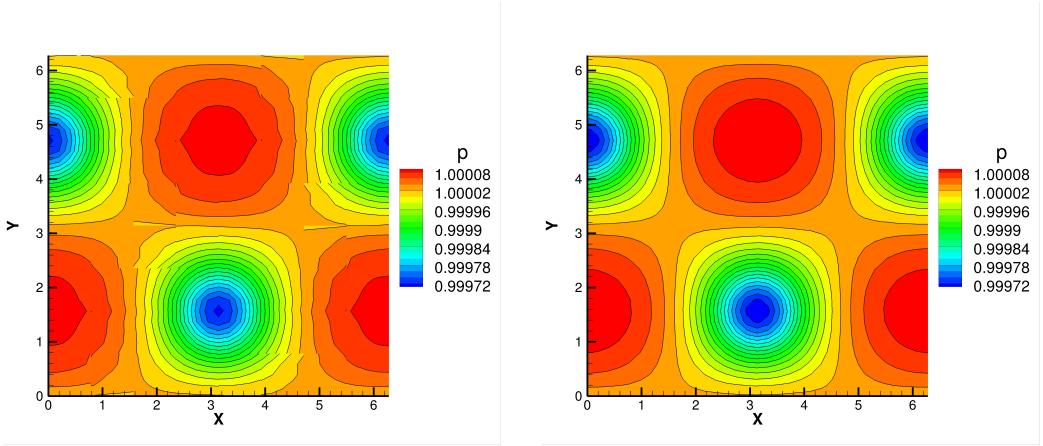


(b) $\varepsilon_r\left(\kappa, \kappa_{ref}\right)\%$



(c) $\varepsilon_r\left(\zeta, \zeta_{ref}\right)\%$

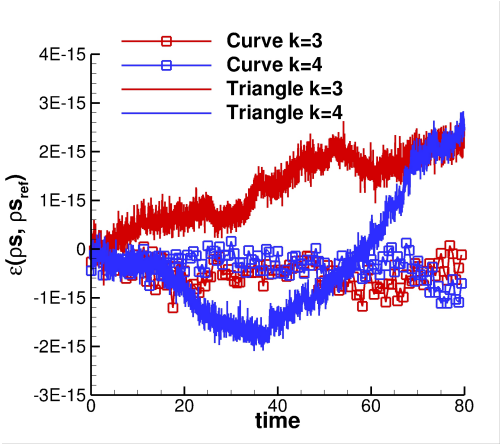
Figure 5: Double shear layers problem – Time refinement study. Simulations performed on the Cartesian grid using the $\mathbb{P}^{3,4}$ DG approximations, the ES numerical flux and the GCNG time integration scheme with different Δt values.



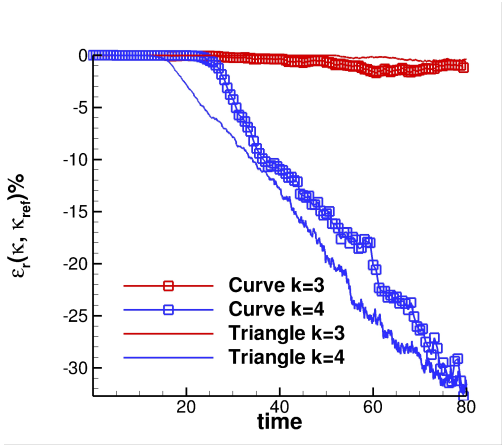
(a) $T/\Delta t = 200$ (b) $T/\Delta t = 6\,400$

Figure 6: Double shear layers problem – Pressure contours at time $T = 80T_c$ computed on the Cartesian grid using the $\mathbb{P}^4$ DG approximation, the ES flux and the GCNG time integration scheme with different Δt values.

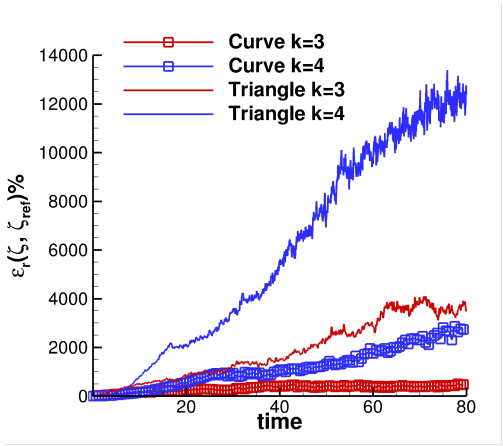
To demonstrate the geometrical flexibility of the proposed method, Fig. 7 and Fig. 8 show the results for the EC and the ES fluxes, respectively, using meshes made of triangular elements and quadrilateral elements with curved edges. The grids used, superimposed to the pressure contours, are shown in Fig. 9. The simulation that uses triangular elements is performed with CFL ≈ 20, while, for the grid with curved edges, this value is ≈ 17. The numerical experiments essentially confirm the findings obtained for the Cartesian mesh case. It can be stated that the method preserves its conservation properties also on unstructured meshes and without the need of any special treatment.



(a) $\varepsilon\left(\rho s, \rho s_{ref}\right)$

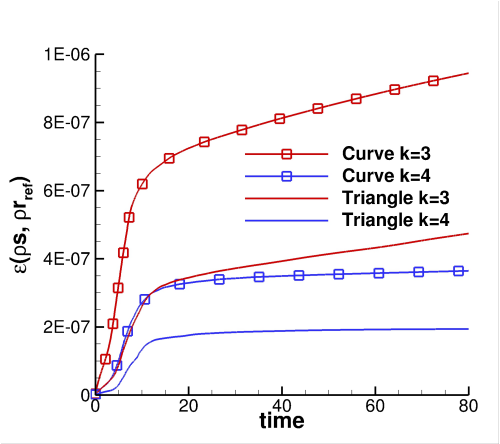


(b) $\varepsilon_r\left(\kappa, \kappa_{ref}\right)\%$

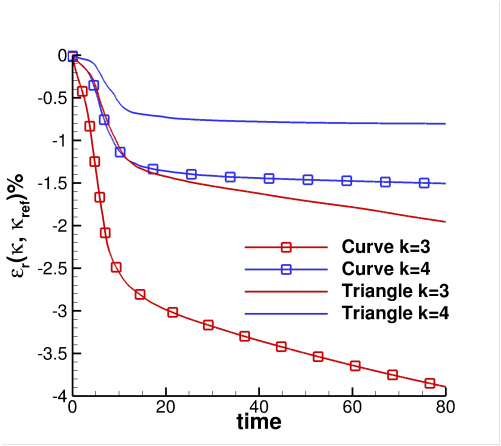


(c) $\varepsilon_r\left(\zeta, \zeta_{ref}\right)\%$

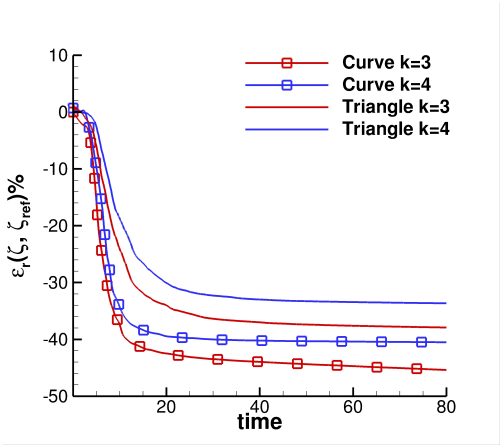
Figure 7: Double shear layers problem – Simulations performed on the meshes made of triangular elements (Triangle) and quadrilateral elements with curved edges (Curve), using the $\mathbb{P}^{3,4}$ DG approximations, the EC numerical flux and the GCNG time integration scheme with $T/\Delta t = 3200$.



(a) $\varepsilon\left(\rho s, \rho s_{ref}\right)$

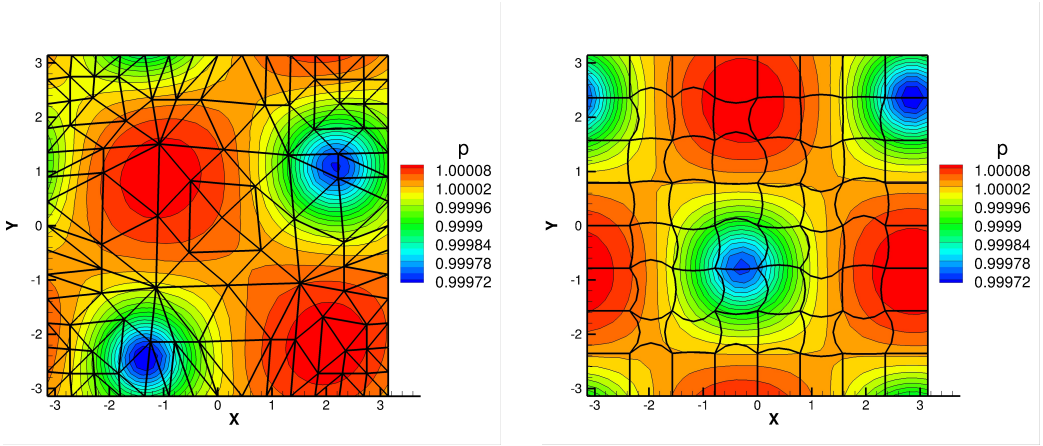


(b) $\varepsilon_r\left(\kappa, \kappa_{ref}\right)\%$



(c) $\varepsilon_r\left(\zeta, \zeta_{ref}\right)\%$

Figure 8: Double shear layers problem – Simulations performed on the meshes made of triangular elements (Triangle) and quadrilateral elements with curved edges (Curve), using the $\mathbb{P}^{3,4}$ DG approximations, the ES numerical flux and the GCNG time integration scheme with $T/\Delta t = 3200$.



(a) Triangular elements

(b) Curved quadrangular elements

Figure 9: Double shear layers problem – Pressure contours at time $T = 80T_c$. Simulations performed on the meshes made of triangular elements and quadrilateral elements with curved edges using the $\mathbb{P}^4$ DG approximation, the ES flux and the GCNG time integration scheme with $T/\Delta t = 3200$.

4.3. The Kelvin-Helmholtz instability problem

This section deals with the two-dimensional Kelvin-Helmholtz instability (KHI) problem parametrized in [32] as a function of the Atwood number $A = (\rho_2 - \rho_1)/(\rho_2 + \rho_1)$. As in the work of Chan et al. [32] an entropy stable DGSEM method (seventh degree polynomial approximation) was observed to begin to crash for $A \approx 0.3$, to demonstrate the robustness of the present method a value slightly larger was used in the following computations, *i.e.*, $A = 1/3$. For the sake of comparison with the reference paper, a similar set-up was considered using the entropy stable ERS flux together with a 32×32 Cartesian uniform grid and polynomial approximations up to $\mathbb{P}^6$.

KHI-type flow problems are known to be very sensitive to initial conditions as well as to the numerical resolution and possible small perturbations. In fact, the flow evolution is characterized by the generation of small structures becoming smaller and smaller when increasing the discretization accuracy.

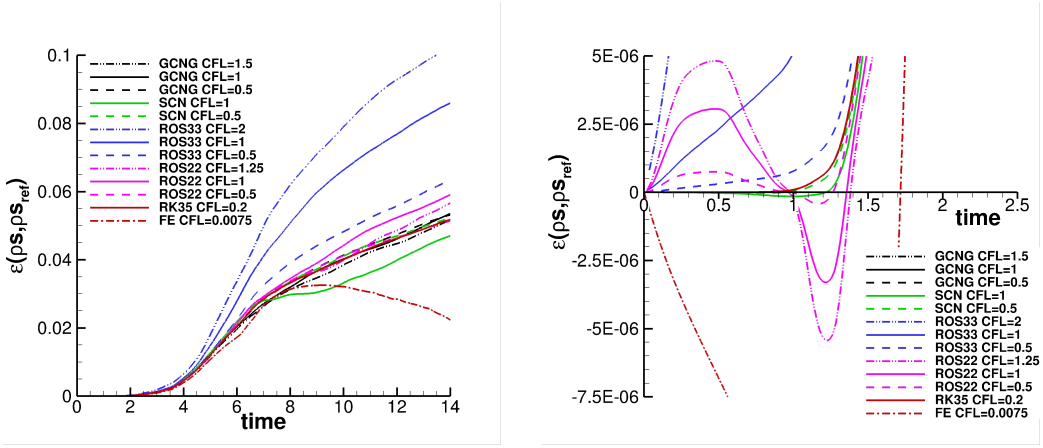
It is worth noting that this test case can be considered as a strongly compressible version of the double shear layer problem with a non-uniform density initialization and a much higher velocity magnitude (maximum Mach number equal to 0.6 for $t = 0$). Although the initial condition can be recast in a form similar to Eq. (39),

the domain size, $-L \leq x_i \leq L$ with $L = 1$ and the same definitions reported in [32] are here used

$$
\begin{aligned}
u_1 &= U\left(f(x_2) - \frac{1}{2}\right), \\
u_2 &= \frac{U}{10}\sin(2\pi x_1), \\
p &= 1, \\
\rho &= \rho_1 + f(x_2)\left(\rho_2 - \rho_1\right),
\end{aligned}
\tag{40}
$$

where $f(x_2) = 1/2\left[\tanh(15x_2 + 7.5) - \tanh(15x_2 - 7.5)\right]$, $U = \sqrt{p/\rho_1}$ and $\rho_1 = 1$.

Computations have been integrated in time up to $T = 14L/U$, a value larger than the final time used in [32]. As the dynamics of the problem is quite fast, the mesh is uniform and the Mach number is high, the computational efficiency of implicit methods was expected not to be very high. Nevertheless, for the $k = 6$ case, the use of GCNG allows to march in time with a CFL number 7.5 times larger than the stability limit, *i.e.* 0.2, of the explicit RK35 scheme, see [33], here considered as a reference solution. Figure 10(a) reports the $\varepsilon(\rho s, \rho s_{ref})$ value as a function of time. For each time integration scheme solutions are shown up to the largest admissible CFL_{max} value. For the present numerical set-up and among all the considered time integrators, *i.e.*, FE, ROS22, ROS33, CNS, GCNG, the ROS33 is the most stable one with a $\mathrm{CFL}_{max} = 2$. For this value, the same scheme also shows the largest production of entropy on the whole time interval. Concerning the GCNG, the method outperforms both the SCN ($\mathrm{CFL}_{max} = 1$) and the ROS22 ($\mathrm{CFL}_{max} = 1.25$) schemes and results, with $\mathrm{CFL}_{max} = 1.5$, in a $\varepsilon(\rho s, \rho s_{ref})$ profile very close to the reference value, the RK35 with $\mathrm{CFL}_{max} = 0.2$, or to the result obtained with a smaller time step size (CFL= 0.5). Moreover, the SCN and ROS22 schemes proved to be entropy unstable, see the detail in Fig. 10(b). Note that, even if the ROS33 scheme seams here stable, we are not able to guarantee its entropy stability. For example, even the ROS58 scheme proves to be entropy unstable, for large time steps, in Figure 17 of [10]. According to the theory, the stability limit of the Forward Euler is very low and the method exhibits a nonphysical large drop of entropy for $t > 8$, see Fig. 10(a). These results highlight how an entropy stable spatial discretization alone may not guarantee an overall fully discrete entropy stable method. In fact, entropy production in space can be overwhelmed by the entropy destruction in time.



(a) (b)

Figure 10: Kelvin-Helmholtz instability problem – Comparison of the $\varepsilon\left(\rho s, \rho s_{ref}\right)$ evolution obtained with different time integrators and different CFL values. Simulations performed using the $\mathbb{P}^6$ DG approximation and the ERS numerical flux.

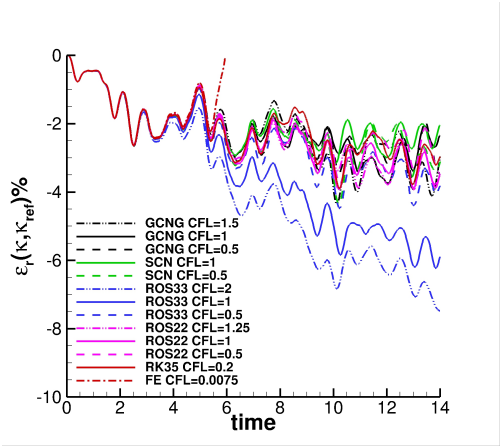
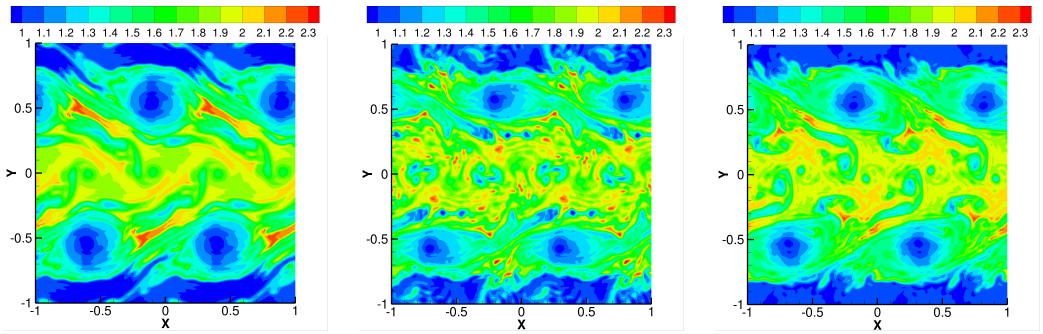


Figure 11: Kelvin-Helmholtz instability problem – Comparison of the $\varepsilon_r\left(\kappa, \kappa_{ref}\right)$ % evolution obtained with different time integrators and different CFL values. Simulations performed using the $\mathbb{P}^6$ DG approximation and the ERS numerical flux.



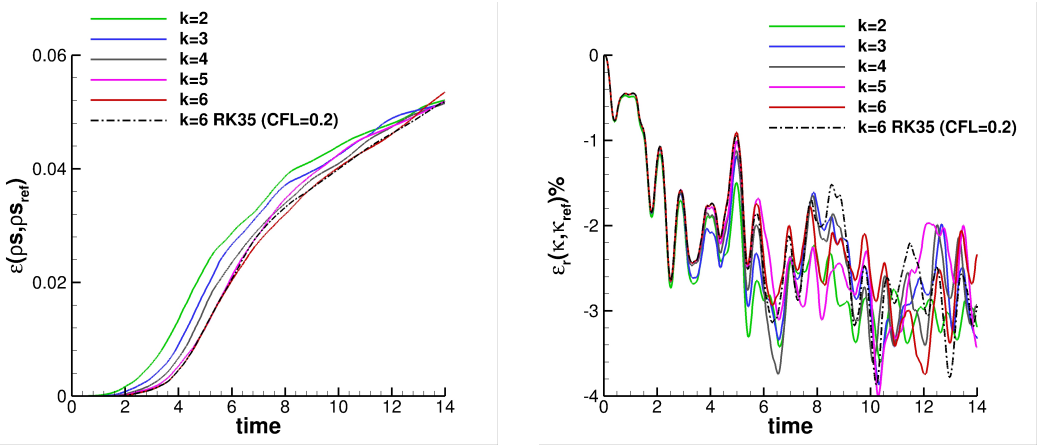
(a) ROS33, CFL = 1 (b) GCNG, CFL = 1 (c) RK35, CFL = 0.2

Figure 12: Kelvin-Helmholtz instability problem – Density contours at the final time $T = 14L/U$. Simulations performed using the $\mathbb{P}^6$ DG approximation, the ERS numerical flux and different time integration schemes.

As our objective is to develop a numerical method well suited for the under-resolved simulation of multiscale phenomena, *e.g.*, in a LES context, it is of particular interest to investigate the kinetic energy conservation properties of the scheme [19]. As expected, the evolution in time of kinetic energy follows a trend similar to entropy, see Fig. 11, where the $\varepsilon(\kappa, \kappa_{ref})\%$ profile for the GCNG is closer to the reference curves than the ROS33, which significantly dissipates energy, especially for large CFL values. As regards the FE, the nonphysical raise of kinetic energy, shown in Fig. 11, dramatically spoils the solution accuracy, and corresponds to a drop in entropy, as observed in Fig. 10.

Notice that, although the theoretical order of accuracy of ROS33 is larger than GCNG, *i.e.*, 3 vs. 2, the density contours shown in Figs. 12(a) and 12(b) reveal that the GCNG delivers a solution containing smaller scales than the ROS33. Obviously, this behavior does not mean that the GCNG solution is in absolute more accurate, see as a reference the solution obtained with the RK35 scheme in Fig. 12(c). However, it highlights the enhanced conservation proprieties of the GCNG, with respects to other schemes, even when using an entropy stable spatial discretization.

Finally, Fig. 13 proves that when using the GCNG (CFL= 1) the entropy and kinetic energy time history of lower polynomial degree disctretizations is close to the $k = 6$ case, although entropy starts to be produced earlier. This behavior is due to the discontinituies of the solution at the mesh faces, which become larger when the numerical accuracy is lowered, as they are the only source of entropy when the ERS flux is coupled with GCNG.



(a) $\varepsilon\left(\rho s, \rho s_{ref}\right)$ (b) $\varepsilon_r\left(\kappa, \kappa_{ref}\right)\%$

Figure 13: Kelvin-Helmholtz instability problem – Time evolution of $\varepsilon\left(\rho s, \rho s_{ref}\right)$ and $\varepsilon_r\left(\kappa, \kappa_{ref}\right)\%$. Simulations performed using the $\mathbb{P}^{2\rightarrow 6}$ DG approximations, the ERS numerical flux and, if not otherwise indicated, the GCNG time integration scheme.

4.4. The shedding flow past a triangular wedge

In this section, a simple test case where an inviscid unsteady flow interacts with boundary conditions is presented. The scope of this flow problem is to testify that no particular numerical treatment is needed at the boundary when entropy variables are used. Inspired by [34, 35], the inviscid flow around a triangular wedge is considered for the free-stream condition $M_\infty = 0.2$. The triangular body immediately generates vortices at sharp corners, even under an inviscid condition, resulting in a vortex shedding behind the wedge.

A natural approach to impose boundary conditions in a DG framework is their weak enforcement, [5], where properly defined states are used, directly or together with the internal states, to compute the numerical fluxes at the boundary faces. These boundary states must be defined according to the condition type and be consistent with the physical flux. In this flow problem inlet/outlet characteristic-based conditions are imposed at left/right boundary, symmetry conditions on the top and bottom boundaries, and the wedge surface is treated as a slip wall.

The computational domain consists of an equilateral triangle with side $L = 1$, placed on the centerline of a rectangular box of size $26L \times 10L$ at a distance $6L$ from the inlet (left) boundary. The simulations were performed using the ERS flux with the $\mathbb{P}^4$ approximation on an unstructured mesh made of 5 407 triangular elements with linear edges. The solution was advanced in time using the GCNG

scheme with a time step size equal to $\Delta t = 10^{-2} T_c$ with $T_c = L/U$, where U is the the free-stream velocity magnitude. During iterations, this time step size corresponds to roughly a CFL value of ≈ 2.3. Figure 14 shows the density contours for two different time levels, $T = 30T_c$ and $T = 50T_c$. A density variation of roughly the 30% is observed for this compressible flow problem. For the $T = 30T_c$ snapshot the mesh has been also superimposed on half of the domain for the sake of completeness. Once generated, the eddies are convected downstream, slightly diffused by the quite coarse mesh, and finally interact with the outlet boundary with only a moderate and expected distortion of the flow structure.

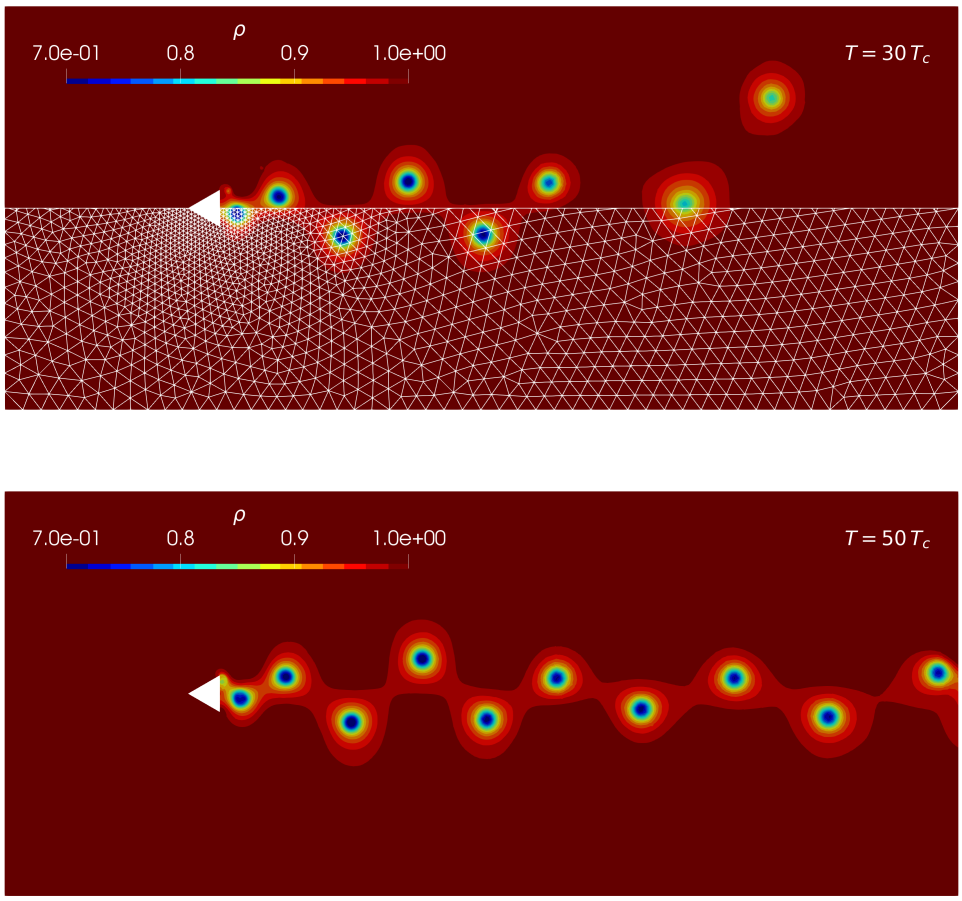


Figure 14: Inviscid flow past a wedge – Density contours, $T = 30T_c$ (top) and $T = 50T_c$ (bottom). Simulation performed on a grid made of 5 407 triangular elements using the $\mathbb{P}^4$ DG approximation, the ERS numerical flux and the GCNG time integration scheme.

4.5. The receding flow problem

In this section a flow problem where rarefaction is generated by two inviscid flows receding one from each other is considered. This test case has been extensively studied by Liou [36, 37] and used by Gouasmi et al. in their presentation of an entropy conserving time integration scheme [17]. The initial condition is given by two constant states separated by a discontinuity

$$(p, \rho, u) = \begin{cases} (2, 1, -0.4) & \text{if } -0.5 \leqslant x \leqslant 0, \\ (2, 1, 0.4) & \text{if } 0 < x \leqslant 0.5, \end{cases} \tag{41}$$

note that these values correspond to the ones used in [17]. This test case exhibits a non-physical temperature rise (overheating) with a spurious entropy generation at the origin that cannot be fixed by simply refining the spatial discretization. Liou proposed to cure this problem by replacing the energy conservation equation with a transport equation for the specific entropy [37]. Unfortunately, this approach does not guarantee for the total energy conservation. Gouasmi et al. demonstrated that this spurious entropy rise is also observed in a fully (both in space and in time) entropy conserving numerical scheme [17].

The solutions, here computed up to $T = 0.18$, have been integrated in time with different values for the time-step size $\Delta t = \{10^{-3}, 5 \cdot 10^{-4}, 10^{-4}\}$ and comparing the results of the GCNG with the second-order Rosenbrock-type scheme of Iannelli and Baker (ROS22) [38, 24]. Taking advantage of the symmetry condition set at $x = 0$ the computational domain $x = [0, 0.5]$ is discretized with 50 elements and Dirichlet conditions applied at the right boundary. The temperature profile, with a detail of the symmetry region, is shown for the fully entropy conserving scheme (GCNG and KEEC) and different values of the time step size, $\Delta t = \{10^{-3}, 5 \cdot 10^{-4}, 10^{-4}\}$, in Fig. 15. It is worth mentioning that in the authors' experience the EC and KEEC flux functions deliver very similar results on several flow cases [10].

As expected, due to the essentially non-dissipative nature of the scheme, a solution affected by non-physical oscillations is predicted regardless of the time step size, cf. [10].

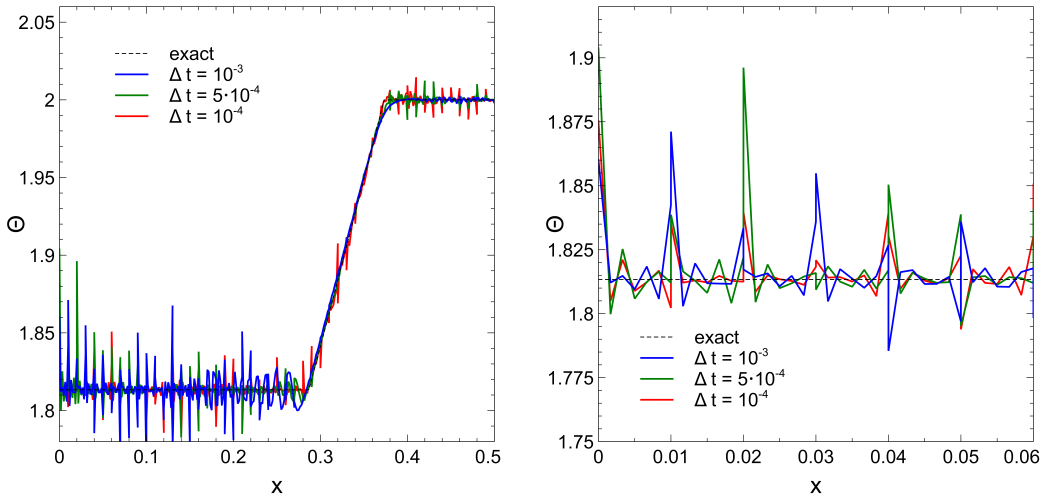
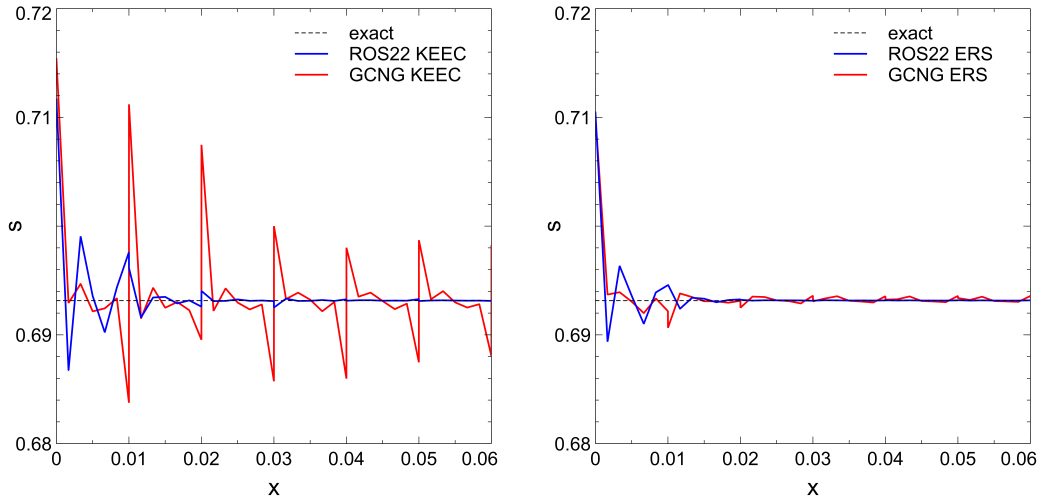


Figure 15: Receding flow problem – Temperature profiles with detail of the symmetry region (right) at $T = 0.18$. Simulations performed on the grid composed by 50 elements using the $\mathbb{P}^6$ DG approximation, the KEEC numerical flux and the GCNG time integration scheme with different values of the time step size ($\Delta t = \{10^{-3}, 5 \cdot 10^{-4}, 10^{-4}\}$).

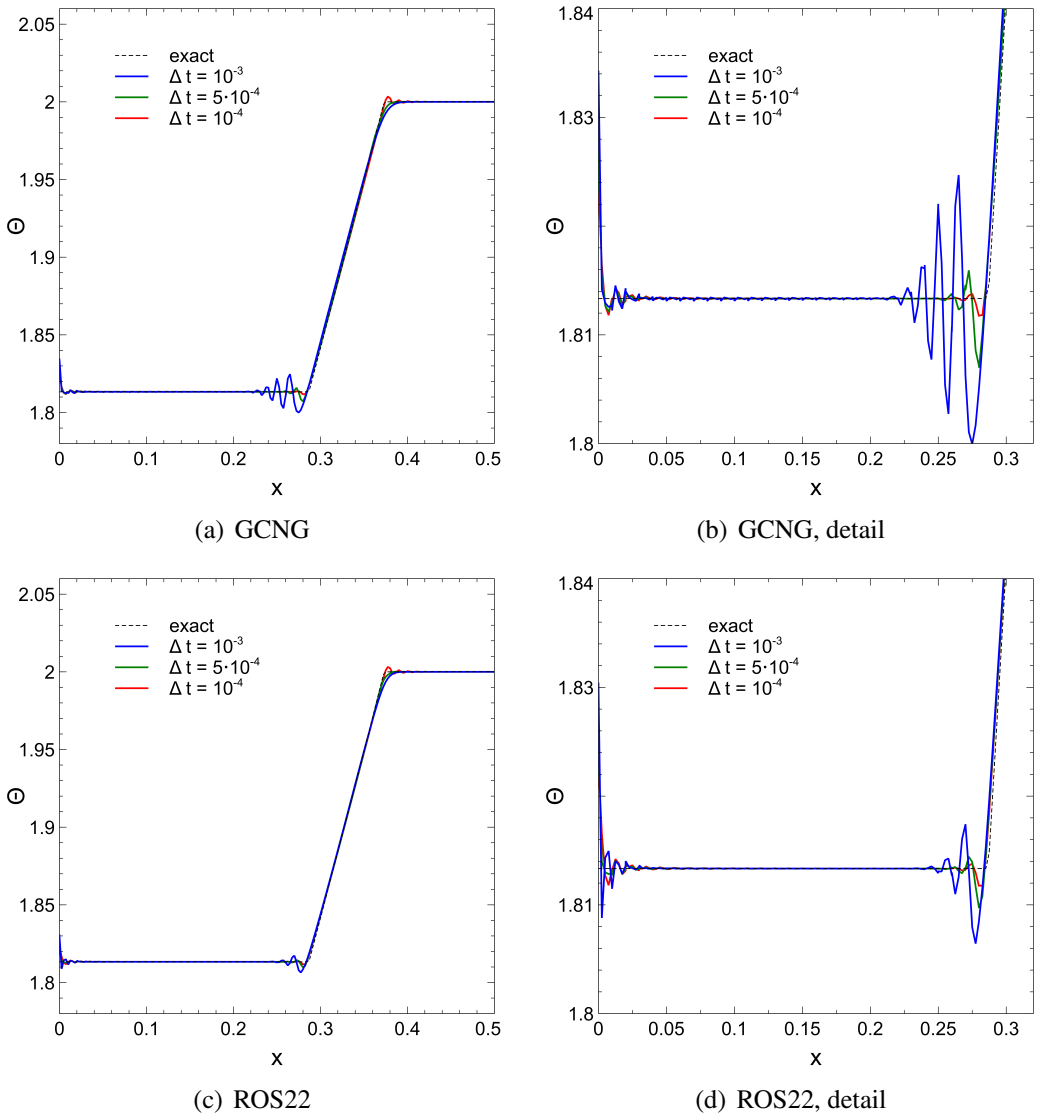
Figure 16(a) shows a detail of the oscillating entropy profile for the $\Delta t = 10^{-4}$ value and the $\mathbb{P}^6$ approximation. Fluctuations strongly reduce when an entropy stable spatial discretization (ERS) and/or the non-entropy-conserving linearly-implicit ROS22 scheme are used.



(a) KEEC (b) ERS

Figure 16: Receding flow problem – Detail of the specific entropy profiles for the receding flow problem at $T = 0.18$. Simulations performed on the grid composed by 50 elements using the $\mathbb{P}^6$ DG approximation, the KEEC and ERS numerical fluxes and the GCNG and ROS22 time integration schemes with $\Delta t = 10^{-4}$).

Temperature profiles are shown in Fig. 17 for the $\mathbb{P}^4$ approximation and the different values of the time step size when using the ERS flux function together with the GCNG or the ROS22 time integrators. Oscillations are mainly concentrated at the symmetry plane, at the foot of the expansion and reduce when reducing the time step size.



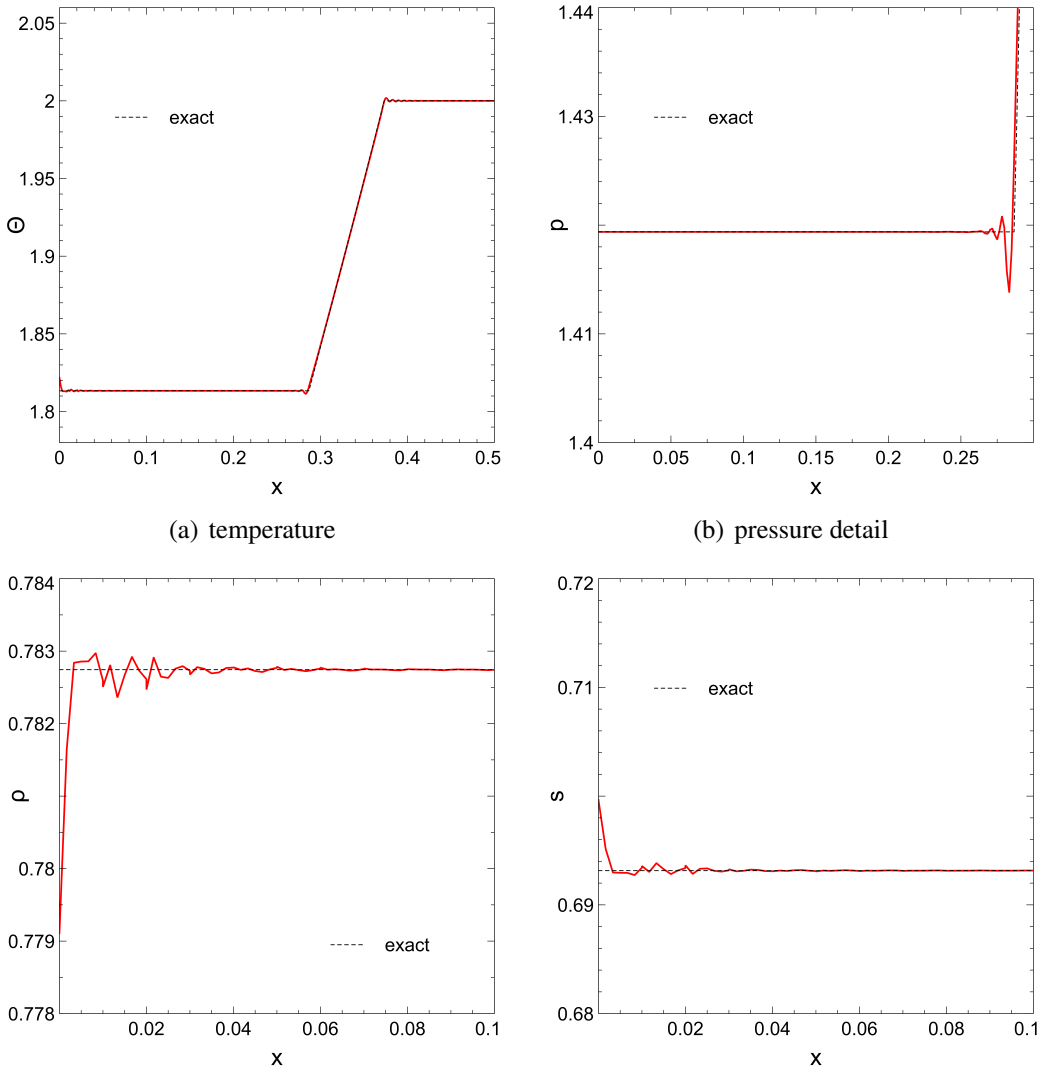
(a) GCNG (b) GCNG, detail

(c) ROS22 (d) ROS22, detail

Figure 17: Receding flow problem – Temperature profiles at $T = 0.18$. Simulations performed on the grid composed by 50 elements using the $\mathbb{P}^4$ DG approximation, the ERS numerical flux and the GCNG and ROS22 time integration schemes with different values of the time step size ($\Delta t = \{10^{-3}, 5 \cdot 10^{-4}, 10^{-4}\}$).

Focusing on the use of the GCNG scheme and the ERS flux, when increasing the accuracy of the spatial discretization from $\mathbb{P}^4$ to $\mathbb{P}^6$, fluctuations reduces as shown Figure 18. Although pressure is well resolved (Fig. 18(b)), overheating develops with a slightly under-estimated density value (Fig. 18(c)) and a spurious

entropy generation (Fig. 18(d)) at symmetry. This corresponds to the behaviour observed by Gouasmi et al. in [17].



(a) temperature

(b) pressure detail

(c) density detail

(d) entropy detail

Figure 18: Receding flow problem – Temperature profile for the receding flow problem at $T = 0.18$ with detail of the symmetry region for pressure, density and entropy. Simulations performed on the grid composed by 50 elements using the $\mathbb{P}^6$ DG approximation, the ERS numerical flux and the GCNG time integration scheme with $\Delta t = 10^{-4}$).

4.6. The Sod shock tube problem

The Sod Shock tube is a Riemann problem for the Euler equations with an initial condition defined as

$$(p, \rho, u) = \begin{cases} (1, 1, 0) & \text{if } -0.5 \leqslant x \leqslant 0, \\ (0.1, 0.125, 0) & \text{if } 0 < x \leqslant 0.5. \end{cases} \tag{42}$$

The solution, here computed up to $T = 0.2/\sqrt{\gamma}$, is made of a left-moving rarefaction wave, a right-moving contact discontinuity and a right-moving shock wave. The computational domain is spatially discretized with 100 elements and symmetry conditions are applied at boundaries. Computations have been performed with the ERS flux, using the $\mathbb{P}^4$ approximation and integrating the solution in time with 2 000 time steps and the GCNG or the ROS22 [38, 24] schemes.

In high-order spatial discretizations flow discontinuities give rise to spurious oscillations that cause stability issues and a loss of accuracy. To cure this problem several strategies to stabilize the solution have been proposed in the literature, *e.g.*, [39, 40, 41, 42]. As in this work the focus is on time integration, to avoid any possible interaction of a discontinuity control algorithm with the solution evolution, we opted not to use any shock-capturing approach during computations. This choice leads to spurious oscillations at discontinuities that also propagate to regions where the solution is almost constant, as shown by the density, temperature and velocity profiles in Fig. 19.

The results from the ROS22 and the GCNG schemes (same order of accuracy) show an overall similar behaviour and demonstrate that the present method can be considered robust even in presence of flow discontinuities and when an entropy conserving time integration scheme is employed.

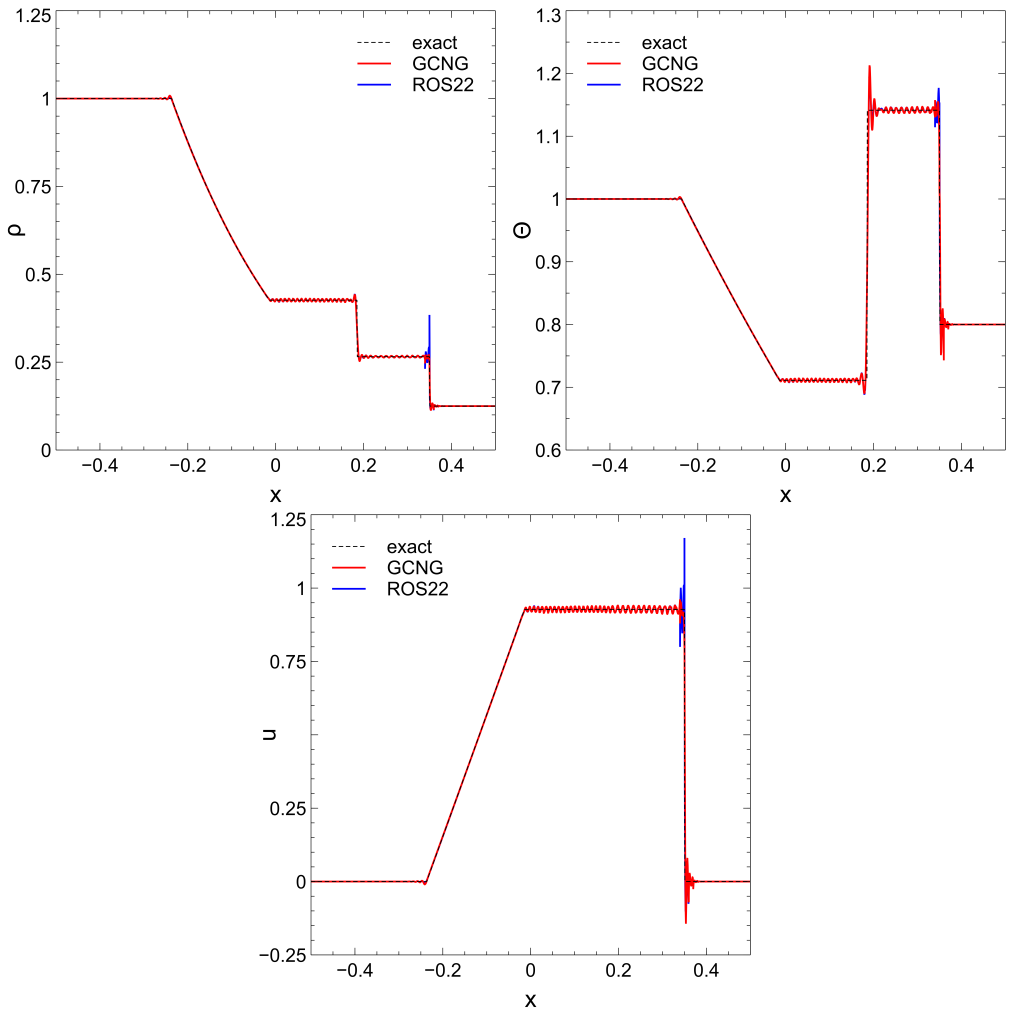


Figure 19: Sod shock tube – Density, temperature and velocity profiles at $T = 0.2/\sqrt{\gamma}$. Simulations performed on the grid composed by 100 elements using the $\mathbb{P}^4$ DG approximation, the ERS numerical flux and the GCNG and ROS22 time integration schemes.

4.7. The inviscid Taylor-Green vortex

The three-dimensional weakly compressible inviscid Taylor-Green vortex [43] is here considered as a representative problem to evaluate the performance of the GCNG scheme for the scale-resolving simulation of turbulent flows. The initial dimensionless condition is given by

$$
\begin{aligned}
u_1 &= U \sin\left(\frac{x_1}{L}\right)\cos\left(\frac{x_2}{L}\right)\cos\left(\frac{x_3}{L}\right),\\
u_2 &= -U \cos\left(\frac{x_1}{L}\right)\sin\left(\frac{x_2}{L}\right)\cos\left(\frac{x_3}{L}\right),\\
u_3 &= 0,\\
p &= 1 + \frac{U^2}{16}\left[\cos\left(\frac{2x_1}{L}\right) + \cos\left(\frac{2x_2}{L}\right)\right]\left[\cos\left(\frac{2x_3}{L}\right) + 2\right],\\
\rho &= 1,
\end{aligned}
\tag{43}
$$

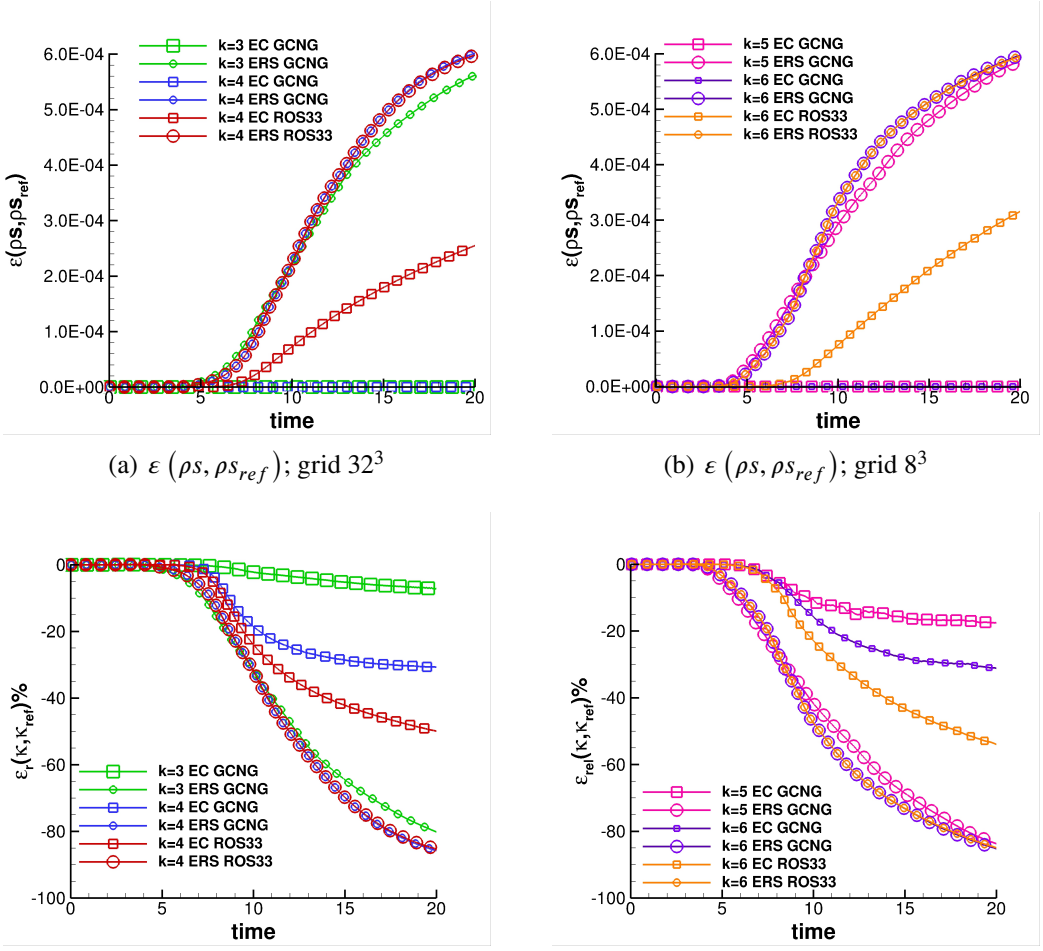
where $U = M_\infty\sqrt{\gamma}$, $L = 1$ and the "free-stream" Mach number is $M_\infty = 0.1$.

The flow problem is solved on the periodic cube $0 \le x_i \le 2\pi L$ using two Cartesian grids made of 8^3 (coarse) and 32^3 (fine) elements, respectively. The simulations have been integrated in time up to $T = 20T_c$ with CFL ≈ 1.4, where $T_c = L/U$ is the convective time. The inviscid nature of this test case makes it very interesting because an infinite range of scales develop and the numerical solution is under-resolved by definition. Figures 20(a) and 20(b) show the time history of $\varepsilon\left(\rho s, \rho s_{ref}\right)$ computed with the fine and the coarse grid, respectively, and several DG approximations. These plots distinctly show that a "true" entropy preserving simulation is possible only if the two entropy conserving schemes, EC in space and GCNG in time, are used together. It is worth noting that the computations performed by coupling the ROS33 scheme with the EC or the ERS numerical fluxes deliver a final value of $\varepsilon(\rho s, \rho s_{ref})$ of the same order of magnitude (10^{-4}). This result suggests that the evolution of ρs is significantly affected by the time integration scheme. However, surprisingly, the ERS results seem almost insensitive to the choice of the time integrator.

These findings demonstrate the difficulty in *a priori* identifying the trend of entropy when a not provable entropy conserving scheme is used, here the ROS33. In fact, in the authors' experience, for a schemes like the ROS33, the entropy production, and possible destruction, depends on the specific features of the numerical solution.

Figures 20(c) and 20(d) report the time history of the relative percentage variation of kinetic energy and confirm that this quantity is much better conserved when the fully entropy conserving scheme uses odd DG approximations ($\mathbb{P}^{3,5}$) rather than even ones ($\mathbb{P}^{4,6}$), cf. Sec. 4.2. Moreover, although the ROS33 is formally one order more accurate than the GCNG, it is observed that κ is better preserved by the last scheme. In fact, as already noted, the conservation of entropy has a positive

influence in the preservation of kinetic energy (κ should be perfectly conserved for $M_\infty \to 0$).



(a) $\varepsilon\left(\rho s, \rho s_{ref}\right)$; grid 32^3

(b) $\varepsilon\left(\rho s, \rho s_{ref}\right)$; grid 8^3

(c) $\varepsilon_r\left(k, k_{ref}\right)$ %; grid 32^3

(d) $\varepsilon_r\left(k, k_{ref}\right)$ %; grid 8^3

Figure 20: Taylor-Green vortex problem – Time evolution of $\varepsilon\left(\rho s, \rho s_{ref}\right)$ and $\varepsilon_r\left(\kappa, \kappa_{ref}\right)$ %. Simulations performed on the fine grid using the $\mathbb{P}^{3,4}$ DG approximations (left column) and on the coarse grid using the $\mathbb{P}^{5,6}$ DG approximations (right column), the EC and ERS numerical fluxes and the GCNG and ROS33 time integration schemes.

Conclusion

In this article a fully discrete entropy conserving/stable numerical method for the solution of the Euler equations has been presented. The method uses a DG spa-

tial discretization and a modified Crank-Nicolson time integration scheme. The entropy conserving time integrator, originally proposed in the context of FV schemes, was adapted to a DG discretization in entropy variables theoretically demonstrating the fulfilment of entropy conservation regardless of the time step size. The conservation and stability properties have been numerically corroborated by computing several unsteady compressible flow problems, also considering different types of, possibly curved, mesh cells.

Future work will be devoted to the solution of the Navier–Stokes equations using the present entropy stable numerical framework with the purpose of addressing scale–resolving simulations of turbulent flows, *e.g.*, DNS, LES.

References

[1] P. G. LeFloch, J. M. Mercier, C. Rohde, Fully discrete, entropy conservative schemes of arbitrary order, SIAM Journal on Numerical Analysis 40 (5) (2003) 1968–1992.

[2] E. Tadmor, Entropy stability theory for difference approximations of nonlinear conservation laws and related time-dependent problems, Acta Numerica 12 (2003) 451–512.

[3] U. S. Fjordholm, S. Mishra, E. Tadmor, Arbitrarily high-order accurate entropy stable essentially nonoscillatory schemes for systems of conservation laws, SIAM Journal on Numerical Analysis 50 (2) (2012) 544–573.

[4] F. Bassi, L. Botti, A. Colombo, D. A. Di Pietro, P. Tesini, On the flexibility of agglomeration based physical space discontinuous Galerkin discretizations, Journal of Computational Physics 231 (1) (2012) 45–65.

[5] F. Bassi, L. Botti, A. Colombo, A. Crivellini, A. Ghidoni, F. Massa, On the development of an implicit high-order discontinuous Galerkin method for DNS and implicit LES of turbulent flows, European Journal of Mechanics, B/Fluids 55 (2016) 367–379. doi:10.1016/j.euromechflu.2015.08.010.

[6] C. de Wiart, K. Hillewaert, L. Bricteux, G. Winckelmans, Implicit LES of free and wall-bounded turbulent flows based on the discontinuous Galerkin/symmetric interior penalty method, Int. J. Numer. Methods Fluids 78 (6) (2015) 335–354. doi:10.1002/fld.4021.

[7] M. de la Llave Plata, V. Couaillier, M.-C. le Pape, On the use of a high-order discontinuous Galerkin method for DNS and LES of wall-bounded turbulence, Computers and Fluids 176 (2018) 320–337. doi:10.1016/j.compfluid.2017.05.013.

[8] G. J. Gassner, A kinetic energy preserving nodal discontinuous Galerkin spectral element method, International Journal for Numerical Methods in Fluids 76 (1) (2014) 28–50.

[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, Journal of Computational Physics 327 (2016) 39 – 66.

[10] A. Colombo, A. Crivellini, A. Nigro, On the entropy conserving/stable implicit DG discretization of the Euler equations in entropy variables, Computers & Fluids 232 (2022) 105198. doi:https://doi.org/10.1016/j.compfluid.2021.105198.

[11] F. Ismail, P. Roe, Affordable, entropy-consistent Euler flux functions II: Entropy production at shocks, Journal of Computational Physics 228 (15) (2009) 5410–5436.

[12] P. Chandrashekar, Kinetic energy preserving and entropy stable finite volume schemes for compressible Euler and Navier-Stokes equations, Communications in Computational Physics 14 (5) (2013) 1252–1286.

[13] H. Ranocha, G. J. Gassner, Preventing pressure oscillations does not fix local linear stability issues of entropy-based split-form high-order schemes (2020). arXiv:2009.13139.

[14] L. Friedrich, G. Schnücke, A. R. Winters, D. C. D. R. Fernández, G. J. Gassner, M. H. Carpenter, Entropy Stable Space–Time Discontinuous Galerkin Schemes with Summation-by-Parts Property for Hyperbolic Conservation Laws, Journal of Scientific Computing 80 (1) (2019) 175–222.

[15] C. Lozano, Entropy production by explicit Runge–Kutta schemes, Journal of Scientific Computing 76 (1) (2018) 521–564.

[16] C. Lozano, Entropy production by implicit Runge–Kutta schemes, Journal of Scientific Computing 79 (3) (2019) 1832–1853.

[17] A. Gouasmi, S. M. Murman, K. Duraisamy, Entropy conservative schemes and the receding flow problem, Journal of Scientific Computing 78 (2) (2019) 971–994. doi:10.1007/s10915-018-0793-8.

[18] P. K. Subbareddy, G. V. Candler, A fully discrete, kinetic energy consistent finite-volume scheme for compressible flows, Journal of Computational Physics 228 (5) (2009) 1347–1364. doi:https://doi.org/10.1016/j.jcp.2008.10.026.

[19] D. Flad, G. Gassner, On the use of kinetic energy preserving DG-schemes for large eddy simulation, Journal of Computational Physics 350 (2017) 782–795.

[20] T. Hughes, L. 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, Computer Methods in Applied Mechanics and Engineering 54 (2) (1986) 223 – 234.

[21] T. Chen, C.-W. Shu, Entropy stable high order discontinuous Galerkin methods with suitable quadrature rules for hyperbolic conservation laws, Journal of Computational Physics 345 (2017) 427 – 461.

[22] T. Chen, C.-W. Shu, Review of Entropy Stable Discontinuous Galerkin Methods for Systems of Conservation Laws on Unstructured Simplex Meshes, CSIAM Transactions on Applied Mathematics 1 (1) (2020) 1–52.

[23] S. Balay, S. Abhyankar, M. F. Adams, J. Brown, P. Brune, K. Buschelman, L. Dalcin, A. Dener, V. Eijkhout, W. D. Gropp, D. Kaushik, M. G. Knepley, D. A. May, L. C. McInnes, R. T. Mills, T. Munson, K. Rupp, P. Sanan, B. F. Smith, S. Zampini, H. Zhang, H. Zhang, PETSc Web page (2018).
URL http://www.mcs.anl.gov/petsc

[24] F. Bassi, L. Botti, A. Colombo, A. Ghidoni, F. Massa, Linearly implicit Rosenbrock-type Runge-Kutta schemes applied to the Discontinuous Galerkin solution of compressible and incompressible unsteady flows, Computers and Fluids 118 (2015) 305–320. doi:10.1016/j.compfluid.2015.06.007.

[25] G. Noventa, F. Massa, S. Rebay, F. Bassi, A. Ghidoni, Robustness and efficiency of an implicit time-adaptive discontinuous Galerkin solver for un-

steady flows, Computers & Fluids 204 (2020) 104529. doi:https://doi.org/10.1016/j.compfluid.2020.104529.

[26] J. Gottlieb, C. Groth, Assessment of Riemann solvers for unsteady one-dimensional inviscid flows of perfect gases, Journal of Computational Physics 78 (2) (1988) 437–458.

[27] H. Yee, N. Sandham, M. Djomehri, Low dissipative high order shock-capturing methods using characteristic-based filters, Journal of Computational Physics 150 (1999) 199–238.

[28] C. Hu, C. Shu, Weighted essentially non-oscillatory schemes on triangular meshes, Journal of Computational Physics 150 (1999) 97–127.

[29] L. Wang, D. Mavriplis, Implicit solution of the unsteady euler equations for high-order accurate discontinuous Galerkin discretizations, Journal of Computational Physics 225 (2007) 1994–2015.

[30] J. B. Bell, P. Colella, H. M. Glaz, A second-order projection method for the incompressible Navier-Stokes equations, Journal of Computational Physics 85 (2) (1989) 257–283.

[31] M. Tavelli, M. Dumbser, A staggered space–time discontinuous Galerkin method for the incompressible Navier–Stokes equations on two-dimensional triangular meshes, Computers & Fluids 119 (2015) 235–249.

[32] 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, Frontiers in Physics 10.

[33] R. J. Spiteri, S. J. Ruuth, A new class of optimal high-order strong-stability-preserving time discretization methods, SIAM Journal on Numerical Analysis 40 (2) (2002) 469–491.

[34] L. Wang, D. J. Mavriplis, Adjoint-based *h-p* adaptive discontinuous Galerkin methods for the 2D compressible Euler equations, J. Comput. Phys. 228 (20) (2009) 7643–7661.

[35] H. Luo, H. Segawa, M. R. Visbal, An implicit discontinuous Galerkin method for the unsteady compressible Navier–Stokes equations, Computers & Fluids 53 (2012) 133–144. doi:https://doi.org/10.1016/j.compfluid.2011.10.009.

[36] M.-S. Liou, Why is the overheating problem difficult: The role of entropy, in: 21st AIAA Computational Fluid Dynamics Conference. doi:10.2514/6.2013-2697.

[37] M.-S. Liou, The root cause of the overheating problem, in: 23rd AIAA Computational Fluid Dynamics Conference. doi:10.2514/6.2017-4107.

[38] G. Iannelli, A. Baker, A stiffly-stable implicit Runge–Kutta algorithm for CFD applications, AIAA Paper 88-0416.

[39] P.-O. Persson, J. Peraire, Sub-cell shock capturing for discontinuous Galerkin methods, in: 44th AIAA Aerospace Sciences Meeting and Exhibit, 2006, p. 112.

[40] R. Hartmann, Adaptive discontinuous Galerkin methods with shock-capturing for the compressible Navier-Stokes equations, International Journal for Numerical Methods in Fluids 51 (9-10) (2006) 1131–1156.

[41] F. Bassi, L. Botti, A. Colombo, A. Crivellini, N. Franchina, A. Ghidoni, S. Rebay, Very high-order accurate discontinuous Galerkin computation of transonic turbulent flows on aeronautical configurations, in: N. Kroll, H. Bieler, H. Deconinck, V. Couaillier, H. van der Ven, K. Sørensen (Eds.), ADIGMA - A European Initiative on the Development of Adaptive Higher-Order Variational Methods for Aerospace Applications, Vol. 113 of Notes on Numerical Fluid Mechanics and Multidisciplinary Design, Springer Berlin Heidelberg, 2010, pp. 25–38.

[42] J. Zeifang, A. Beck, A data-driven high order sub-cell artificial viscosity for the discontinuous Galerkin spectral element method, Journal of Computational Physics 441.

[43] C.-W. Shu, W.-S. Don, D. Gottlieb, O. Schilling, L. Jameson, Numerical Convergence Study of Nearly Incompressible, Inviscid Taylor–Green Vortex Flow, Journal of Scientific Computing 24 (2005) 1–27.