Comput. Methods Appl. Mech. Engrg. 253 (2013) 479–490

Contents lists available at SciVerse ScienceDirect

Comput. Methods Appl. Mech. Engrg.

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

Implementation of the entropy viscosity method with the discontinuous Galerkin method ☆

Valentin Zingan [b], Jean-Luc Guermond [a,*,1], Jim Morel [b], Bojan Popov [a]

[a] Department of Mathematics, Texas A&M University, 3368 TAMU, College Station, TX 77843, USA
[b] Department of Nuclear Engineering, Texas A&M University, 3368 TAMU, College Station, TX 77843, USA

ARTICLE INFO

Article history:
Received 19 January 2012
Received in revised form 11 August 2012
Accepted 17 August 2012
Available online 7 September 2012

Keywords:
Entropy viscosity
Conservation laws
Compressible flow
Euler equations
Finite elements
Stabilized finite element method

ABSTRACT

The notion of entropy viscosity method introduced in Guermond and Pasquetti [21] is extended to the discontinuous Galerkin framework for scalar conservation laws and the compressible Euler equations.

1. Introduction

The use of artificial viscosity to solve nonlinear conservation equations has been pioneered by von Neumann and Richtmyer [49] and has been popularized later by Smagorinsky [44] for Large Eddy Simulation purposes and by Ladyženskaja [30,29] for theoretical purposes in the analysis of the Navier–Stokes equations. With the early versions of artificial viscosities being overly dissipative, interest in these technique have faded over the years, especially in the Discontinuous Galerkin Finite Element Method (DGFEM) literature, where up-winding and limiters have been shown to be efficient and to yield high-order accuracy [43,15,13,12]. Despite their un-disputable success story, limiters have some disadvantages. For instance, as argued in [48, Section 3.5, 23], some limiters may not be consistent in the steady-state limit, and thus may sometimes lead to difficulties when trying to use time stepping schemes to reach steady-state solutions. Furthermore, with a few exceptions [15, Section 3.4], slope limiting is essentially a one dimensional concept that does not generalize easily to unstructured meshes in two and more dimensions. The theoretical understanding of the stability and convergence of limiters is currently restricted to uniform grids and scalar equations in one space dimension [32,35,40,51,52]. A true two-dimensional non-oscillatory reconstruction which could be applied to arbitrary unstructured meshes (without any additional post processing) seems to be available only in the piecewise linear case [9], and extensions to higher degree polynomial reconstructions do not seem to be evident at the present time. Note however that slope limiters can be interpreted in term of shock capturing mechanism and artificial viscosity as argued in Cockburn [10]. It is shown in particular in [10] that "shock capturing terms or generalized slope limiting procedures are different ways of incorporating the information of the dissipation effects" that is required to ensure convergence to the entropy solution. In other words, the effect of slope limiting can in general be compared to that of a nonlinear artificial viscosity. For the above reasons and the fact that artificial viscosities are easy to implement, the interest for artificial viscosity has lately been revived in the DG literature [7,5,23,38,48] and in the Continuous Galerkin (CG) literature as well [8,21].

A new technique for generating high-order numerical approximations for nonlinear conservation equations has recently been introduced in [21,20,22] using continuous finite elements and spectral elements. Contrary to the viscosity-based shock-capturing introduced in [24] and analyzed in [27,11], it has been shown in [8,21,22] that nonlinear viscosity does not need additional linear stabilization to work properly and be high-order. The main

☆ This material is based upon work supported by the Master Agreement No. C08-00353 with Lawrence Livermore National Laboratory, the NSF grant DMS-0811041, the AFOSR award FA9550-09-1-0424, and it is also partially supported by award KUS-C1-016-04, made by King Abdullah University of Science and Technology (KAUST).

* Corresponding author.
E-mail address: guermond@math.tamu.edu (J.-L. Guermond).
[1] On leave from CNRS, France.

0045-7825/$ - see front matter © 2012 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.cma.2012.08.018

stabilization mechanism in this method is a nonlinear dissipation proportional to the local size of an entropy production. For this reason the method is called entropy viscosity. It is reasonably argued in [5,38] that good artificial viscosities can be computed from measures of the local regularity of the solution. The same case is made in [7,23] by proposing to make the viscosity proportional to the local residual of the PDE. In this paper we take a slightly different route by proposing to use the local residual of an entropy equation to construct the artificial viscosity. One immediate consequence of this choice is that the viscosity is proportional to the entropy production, which is known to be large in shocks and to be zero in contact discontinuities. As a result, this strategy makes an automatic distinction between shocks and contact discontinuities, and this subtle distinction cannot be made by any of the two classes of methods mentioned above. We also think that using the residual of the conservation equation may be less robust than using the entropy residual. This argument is based on the observation that consistency requires the residual of the PDE to converge to zero in the distribution sense as the mesh-size goes to zero, whereas the very nature of entropy implies that the entropy residual converges to Dirac measures supported in the shocks. Therefore, the entropy residual focuses far better on shocks than the PDE residual, and it is in this sense that we claim that the PDE residual is less reliable than the entropy residual.

Although no convergence proof of the entropy viscosity method has been produced yet (only stability for nonlinear scalar conservations has been proven), the method has been shown in [21,20,22] to deliver high-order accuracy by testing it on a large variety of benchmark problems. The objective of the present paper is to extend the entropy viscosity method to discontinuous finite elements, which, to the best of our knowledge, has not been done yet. The convergence properties of the proposed extensions are investigated here numerically.

The paper is organized as follows. Notation and other preliminaries are introduced in Section 2. The DG discretization in space is presented in Section 3. The highlights in this section are the definition of the entropy viscosity and the entropy stability result proved in Proposition 3.3. Implementation details of the method including the time discretization are reported in Section 4. One key novelty with respect to [21,22] is that we show how the entropy viscosity can be computed on-the-fly. The capabilities of the method are illustrated numerically in Section 5. It is confirmed in this section (as was observed in [21,22]) that linear stabilization is not mandatory to make the method convergent optimally, i.e., the method still converges optimally when the inviscid numerical flux is centered. Sections 2–5 are restricted to scalar conservation equations. Extensions of the method to compressible fluid dynamics are introduced and tested numerically in Section 6. Conclusions are reported in Section 7.

2. Preliminaries

The objective of the section is to introduce notation and to formulate the problem we are interested in. We restrict ourselves to scalar conservation equations from Sections 2 to 5.

2.1. Scalar conservation equations

We are interested in approximating the solutions of scalar-valued conservation equations

$$\partial_t u + \nabla \cdot \boldsymbol{f}(u) = 0, \quad u(x,0) = u_0(x), \quad (x,t) \in \Omega \times \mathbb{R}_+, \tag{2.1}$$

where Ω is a domain in $\mathbb{R}^d$, d is the space dimension, and $\boldsymbol{f} \in \mathcal{C}^1(\mathbb{R};\mathbb{R}^d)$. For the sake of simplicity we assume that there are no issues with the boundary conditions; for instance, either the boundary conditions are periodic, or the initial data is compactly supported and we are interested in the solution before the domain of dependence of u_0 reaches the boundary of Ω.

It is known that the scalar-valued Cauchy Problem (2.1) may have infinitely many weak solutions, but only one of them is physical and satisfies the additional inequalities

$$\partial_t E(u) + \nabla \cdot \boldsymbol{F}(u) \leqslant 0, \tag{2.2}$$

for all convex functions $E \in \mathcal{C}^0(\mathbb{R};\mathbb{R})$, where $\boldsymbol{F}(u) := \int E'(v)\boldsymbol{f}'(v)\,\mathrm{d}v$ [28]. This physical solution is henceforth called the entropy solution. The function $E(u)$ is called entropy and $\boldsymbol{F}(u)$ is the associated entropy flux. The most well known pairs are the Kružkov pairs generated by $\{E(u) = |u-c|,\ c \in \mathbb{R}\}$. It is also known for strictly convex fluxes in one space dimension that if a weak solution satisfies one entropy inequality (2.2) (provided the entropy E is strictly convex), then it is the unique entropy solution and therefore all the entropy inequalities are satisfied [37].

2.2. Discontinuous finite elements

We consider a mesh family $\{\mathcal{T}_h\}_{h>0}$. Members of $\mathcal{T}_h$ are equivalently called elements or cells. The diameter of $K \in \mathcal{T}_h$ is denoted d_K and the diameter of the largest ball inscribed in K is ρ_K. The mesh family is assumed to be shape regular, meaning that the quantity $\sup_{h>0}\max_{K\in\mathcal{T}_h} d_K/\rho_K$ is finite, i.e., the elements are not too flat. For all $K \in \mathcal{T}_h$ the collection of elements in $\mathcal{T}_h$ that touch K is denoted Δ_K. We assume that the mesh family is locally quasi-uniform in the sense that the quantity $\sup_{h>0}\max_{K\in\mathcal{T}_h}(d_K/(\min_{K'\in\Delta_K} d_{K'}))$ is finite, i.e., all the elements that touch K have a diameter of order d_K.

The reference element is denoted $\widehat{K}$ and the map between $\widehat{K}$ and an arbitrary element $K \in \mathcal{T}_h$ is denoted $T_K : \widehat{K} \to K$. We define the scalar-valued finite element approximation space

$$X_h := \{v \in L^1(\Omega;\mathbb{R});\ \ v|_K \circ T_K \in Q_{\widehat{K}},\ \forall K \in \mathcal{T}_h\}, \tag{2.3}$$

where $Q_{\widehat{K}}$ is a polynomial space over $\widehat{K}$. Let $\mathbb{P}_l$ and $\mathbb{Q}_l$ denote the set of multivariate polynomials of total degree and partial degree at most l, respectively. We denote $k \geqslant 1$ the largest integer so that $\mathbb{P}_k \subset Q_{\widehat{K}}$. The integer k defines the approximation order of the space X_h. Denoting s_K the smallest distance between all possible pairs of vertices in K we define

$$h_K := \frac{s_K}{k}. \tag{2.4}$$

Note that the shape regularity of the mesh family implies that

$$c_i := \inf_{h>0}\inf_{\substack{v\neq 0\\ v\in X_h}} \frac{\int_K |v|^2\,\mathrm{d}K}{h_K \int_{\partial K} |v|^2\,\mathrm{d}\partial K} \tag{2.5}$$

is bounded away from zero. It is shown in [42] that $c_i \sim \frac{d_K}{k\rho_K} \sim \frac{d_K}{k^2 h_K}$.

Let $\mathcal{F}_h^i$ be the set of all the internal interfaces in the mesh $\mathcal{T}_h$. The reference face is denoted $\widehat{F}$ and the map between $\widehat{F}$ and an arbitrary face $F \in \mathcal{F}_h^i$ is denoted $T_F : \widehat{F} \to F$. Denoting by $K_{1,F} \in \mathcal{T}_h,\ K_{2,F} \in \mathcal{T}_h$ the two cells so that $F = K_{1,F} \cap K_{2,F}$, and assuming that v is a scalar- or vector-valued function over $K_{1,F} \cup K_{2,F}$ that is continuous over $K_{1,F}$ and $K_{2,F}$, we define

$$\{v\}(\boldsymbol{x}) := \frac{1}{2}(v|_{K_{1,F}}(\boldsymbol{x}) + v|_{K_{2,F}}(\boldsymbol{x})), \tag{2.6}$$

the average of v across F at $\boldsymbol{x}$.

The outward unit normal at the boundary of the cell K is denoted by $\boldsymbol{n}_K$. Assume that F is an interface between $K_{1,F}$ and $K_{2,F}$; then we construct the jump of a function v across F as follows:

$$[\![v]\!](\boldsymbol{x}) := 2\{v\boldsymbol{n}\}(\boldsymbol{x}) = v|_{K_{1,F}}(\boldsymbol{x})\boldsymbol{n}_{K_{1,F}}(\boldsymbol{x}) + v|_{K_{2,F}}(\boldsymbol{x})\boldsymbol{n}_{K_{2,F}}(\boldsymbol{x}). \tag{2.7}$$

The jump function $[\![v]\!]$ is vector-valued and parallel to the normal vector $\boldsymbol{n}_K$. For instance we have $[\![v]\!](\boldsymbol{x})\cdot\boldsymbol{n}_{K_{1,F}} = v|_{K_{1,F}}(\boldsymbol{x}) - v|_{K_{2,F}}(\boldsymbol{x})$ and $[\![v]\!](\boldsymbol{x})\cdot\boldsymbol{n}_{K_{2,F}} = v|_{K_{2,F}}(\boldsymbol{x}) - v|_{K_{1,F}}(\boldsymbol{x})$.

We define the piecewise constant mesh-size function $h:\Omega\to\mathbb{R}$ so that $h(\boldsymbol{x}) = h_K$ for all $\boldsymbol{x}\in K$. Note that h is two-valued over $\mathcal{F}_h^i$.

3. Space approximation

We describe the DG approximation in space in this section.

3.1. DG space discretization

The main idea of the entropy viscosity method is to regularize (2.1) with a nonlinear dissipative term

$$\partial_t u + \nabla\cdot\boldsymbol{f}(u) - \nabla\cdot(\mu(\nabla u,\partial_t u)\nabla u) = 0,\quad u(x,0) = u_0(x),\quad (x,t)\in\Omega\times\mathbb{R}_+, \tag{3.1}$$

where the dissipation $\mu(\nabla u,\partial_t u)$ is proportional to an entropy production. Details on the construction of μ will be given later. It suffices to know for the time being that $\mu(\nabla u,\partial_t u)$ is zero if u is smooth so that (3.1) is consistent with (2.1).

Let us now recall how DG approximations of second-order PDE's are usually constructed. Let us test (3.1) with a discrete function $v_h\in X_h$ with support in one cell, say $K\in\mathcal{T}_h$, and let us integrate by parts

$$\int_K v_h\partial_t u\,\mathrm{d}K - \int_K \boldsymbol{f}(u)\cdot\nabla v_h\,\mathrm{d}K + \int_{\partial K} v_h\boldsymbol{f}(u)\cdot\boldsymbol{n}_K\,\mathrm{d}\partial K + \int_K \mu(\nabla u,\partial_t u)\nabla u\cdot\nabla v_h\,\mathrm{d}K - \int_{\partial K} v_h\mu(\nabla u,\partial_t u)\nabla u\cdot\boldsymbol{n}_K\,\mathrm{d}\partial K = 0. \tag{3.2}$$

We would like to replace the exact solution u by an approximate one, say u_h, but since $u_h\in X_h$ is discontinuous across the boundary of $K,\partial K$, the functions $\boldsymbol{f}(u_h)$, and $\mu(\nabla u,\partial_t u)\nabla u$ are multi-valued over ∂K. To obtain unique values on ∂K we replace the ambiguous functions by numerical fluxes $\hat{\boldsymbol{f}}$ and $\hat{\boldsymbol{g}}$, respectively (the numerical fluxes are defined in the next section). The DG formulation of (3.1) consists of seeking $u_h\in\mathcal{C}^1([0,T];X_h)$ with $u_h|_{t=0} = u_{0,h}$, where $u_{0,h}$ is an appropriate approximation of u_0, so that the following hold for all $t>0$, all $K\in\mathcal{T}_h$ and all functions $v_h\in X_h$ with support in K:

$$\int_K v_h\partial_t u_h\,\mathrm{d}K - \int_K \boldsymbol{f}(u_h)\cdot\nabla v_h\,\mathrm{d}K + \int_{\partial K} v_h\hat{\boldsymbol{f}}\cdot\boldsymbol{n}_K\,\mathrm{d}\partial K + \int_K \mu(\nabla u_h,\partial_t u_h)\nabla u_h\cdot\nabla v_h\,\mathrm{d}K - \int_{\partial K} v_h\hat{\boldsymbol{g}}\cdot\boldsymbol{n}_K\,\mathrm{d}\partial K = 0. \tag{3.3}$$

3.2. Definition of fluxes

Numerical fluxes can be defined in many ways. In the hyperbolic literature it is common to use the so-called upwind flux [41,33]. In the elliptic/parabolic literature the numerical flux associated with diffusion is defined in many (equivalent) ways depending on the personal taste of the authors (IPG [1], NIPG, LDG [14], BR [6], etc.). Both the hyperbolic and elliptic numerical fluxes can be defined in a unified way [2,15,17] as the sum of the average flux plus a stabilizing term proportional to the jump of the dependent variable. The choice we made is the following:

$$\hat{\boldsymbol{f}}(\{u_h\},[\![u_h]\!]) = \boldsymbol{f}(\{u_h\}) + \omega\,\alpha_F(u_h)[\![u_h]\!], \tag{3.4}$$

$$\hat{\boldsymbol{g}}(u_h) = \{\mu\nabla u_h\} - \delta\,\beta_F(u_h)[\![u_h]\!], \tag{3.5}$$

where $\omega\geqslant 0,\beta\geqslant 0$, and the maps $\alpha_F(u_h)$and $\beta_F(u_h)$ are singled-valued and defined in 3.6,3.7. Using jumps and averages ensures that the numerical fluxes are single-valued, thereby implying conservation. The purpose of the averages is to ensure consistency and that of the jumps to ensure stability.

For instance, assuming that the exact solution is smooth, we have $\hat{\boldsymbol{f}}(u,0) = \boldsymbol{f}(u)$ and $\hat{\boldsymbol{g}}(u) = \mu(\nabla u,\partial_t u)\nabla u$ which implies that (3.3) is consistent with (3.1), which in turn is consistent with (2.1) since $\mu(\nabla u,\partial_t u) = 0$ when u is smooth with respect to space and time.

The functions $\alpha_F(u_h)$and $\beta_F(u_h)$ are defined as follows:

$$\alpha_F(u_h)(\boldsymbol{x}) = \max_{|\xi-\{u_h\}(\boldsymbol{x})|\leqslant\frac{1}{2}|[\![u_h]\!](\boldsymbol{x})|}\frac{1}{2}|\boldsymbol{f}'(\xi)\cdot\boldsymbol{n}_K|, \tag{3.6}$$

$$\beta_F(u_h)(\boldsymbol{x}) = \left\{\frac{\mu}{h}\right\}(\boldsymbol{x}), \tag{3.7}$$

for every interface $F\in\mathcal{F}_h^i$ and every $\boldsymbol{x}\in F$. In general we take $\omega\in\{0,1\}$ and $\delta\in\{0,1\}$.

Remark 3.1. It is also possible to use the so-called Lax–Friedrichs/Rusanov flux to define $\hat{\boldsymbol{f}}(\{u_h\},[\![u_h]\!])$, meaning

$$\hat{\boldsymbol{f}}_{\mathrm{LFR}}(\{u_h\},[\![u_h]\!]) := \{\boldsymbol{f}(u_h)\} + \omega\alpha_F(u_h)[\![u_h]\!]. \tag{3.8}$$

We chose to work with (3.4) because the proof of monotonicity (see Lemma 3.2) is slightly simpler for (3.4) than for the Lax-Fredrichs/Rusanov flux.

Remark 3.2. Note that in the particular case of the linear transport equation, i.e., $\boldsymbol{f}(u) = u\boldsymbol{V}$, using $\omega = 1$ and (3.6) in either (3.4) or (3.8) is equivalent to using the upwind flux, since

$$\hat{\boldsymbol{f}}(\{u_h\},[\![u_h]\!])\cdot\boldsymbol{n}_K = \boldsymbol{V}\cdot\boldsymbol{n}_K\frac{u_h^i+u_h^e}{2} + \frac{1}{2}|\boldsymbol{V}\cdot\boldsymbol{n}_K|(u_h^i-u_h^e) = \begin{cases} u_h^e\boldsymbol{V}\cdot\boldsymbol{n}_K & \text{if } \boldsymbol{V}\cdot\boldsymbol{n}_K\leqslant 0,\\ u_h^i\boldsymbol{V}\cdot\boldsymbol{n}_K & \text{if } \boldsymbol{V}\cdot\boldsymbol{n}_K>0.\end{cases} \tag{3.9}$$

Remark 3.3. It is possible to symmetrize the viscous flux in the spirit of the IP method [1] to enforce the adjoint consistency as done in [23], but we did not feel the need to add this additional complexity. Actually the adjoint consistency [2] is not required since the viscous flux is consistent with 0. We could also have used a more sophisticated penalty term in the definition of the viscous flux in the spirit of LDG [14] or the Brezzi et al. method [2] to make the method work for all $\delta>0$, or we could also have used the so-called BRMPS method [7] to have the method work for all $\delta>1$. All these alternative methods involve additional computations that we think are not really needed, since the method has been observed to be stable with the pair $(\omega,\delta) = (1,0)$ and is proved in Lemma 3.4 to be stable with $\delta = 0$, if $\omega\geqslant 1+\frac{c_{\max}}{c_i}$.

3.3. Viscosity

The viscosity μ is defined to be piecewise constant over the mesh. Upon choosing one or more entropy functionals, say $E_1,\dots,E_{n_E}$, the entropy viscosity μ is defined as follows over each cell $K\in\mathcal{T}_h$:

$$\mu|_K = \min(c_{\max}h_K\max_{\boldsymbol{x}\in K}|\boldsymbol{f}'(u_h(\boldsymbol{x},t))|, c_E h_K^2\max(D_1(u_h),\dots,D_{n_E}(u_h))), \tag{3.10}$$

where $D_i(u_h) := \max(\max_{\boldsymbol{x}\in K}|R_i(u_h)|,\max_{\boldsymbol{x}\in\partial K}|J_i(u_h)|)/N_i, 1\leqslant i\leqslant n_E$, is defined as follows:

$$R_i(u_h) := \partial_t E_i(u_h) + \boldsymbol{f}'(u_h)\cdot\nabla E_i(u_h), \tag{3.11}$$

$$J_i(u_h) := h_K^{-1}\{\boldsymbol{f}'(u_h)\}\cdot[\![E_i(u_h)]\!], \tag{3.12}$$

$$N_i := \max_{\boldsymbol{x}\in\Omega}|E_i(u_h(\boldsymbol{x},t)) - \frac{1}{|\Omega|}\int_\Omega E_i(u_h)\,\mathrm{d}\Omega|. \tag{3.13}$$

R_i is the entropy residual associated with the entropy E_i, J_i is the entropy residual induced by the jump of entropy across the boundary ∂K, and N_i is a normalization factor.

We generally use only one entropy. For instance we use $n_E = 1$ and $E_1(u) = \frac{1}{2}(u - m_0)^2$ where $m_0 = \frac{1}{2}(\max_{\boldsymbol{x}\in\Omega} u_0(\boldsymbol{x}) + \min_{\boldsymbol{x}\in\Omega} u_0(\boldsymbol{x}))^2$ in the numerical tests reported in the rest of the paper, unless explicitly stated otherwise.

3.4. Entropy stability

The purpose of this section is to derive sufficient conditions on the numerical flux so that the resulting approximation satisfies a local entropy inequality in the spirit of [25], see (3.17).

Definition 3.1. A numerical flux $\widehat{\boldsymbol{G}} : \mathbb{R} \times \mathbb{R}^d \to \mathbb{R}^d$, consistent with $\boldsymbol{G}$, is said to be monotone if the following holds:

$$\left(\widehat{\boldsymbol{G}}(a,\boldsymbol{b}) - \boldsymbol{G}(\xi)\right)\cdot\boldsymbol{b} \geqslant 0, \quad \forall \xi, \quad |a-\xi| \leqslant \frac{1}{2}|\boldsymbol{b}|, \quad \forall a \in \mathbb{R}, \quad \forall \boldsymbol{b} \in span(\boldsymbol{n}_K). \tag{3.14}$$

Lemma 3.2. Assume that $\boldsymbol{f}$ is of class $\mathcal{C}^1$. The numerical fluxes (3.4) and (3.8) with (3.6) are monotone if $\omega \geqslant 1$. More precisely, upon setting either $\hat{\boldsymbol{h}} = \hat{\boldsymbol{f}}$ or $\hat{\boldsymbol{h}} = \hat{\boldsymbol{f}}_{\mathrm{LFR}}$, we have

$$\left(\hat{\boldsymbol{h}}(a,\boldsymbol{b}) - \boldsymbol{f}(\xi)\right)\cdot\boldsymbol{b} \geqslant \left(\omega\alpha_F - \max_{|a-\xi|\leqslant\frac{1}{2}|\boldsymbol{b}|}\frac{1}{2}|\boldsymbol{f}'(\xi)\cdot\boldsymbol{n}_K|\right)|\boldsymbol{b}|^2, \quad \forall a \in \mathbb{R}, \quad \forall \boldsymbol{b} \in span(\boldsymbol{n}_K). \tag{3.15}$$

Proof. Consider first the flux (3.4). Using the mean value Theorem and the fact that $\boldsymbol{f}$ is of class $\mathcal{C}^1$, we infer that

$$\begin{aligned}\left(\hat{\boldsymbol{f}}(a,\boldsymbol{b}) - \boldsymbol{f}(\xi)\right)\cdot\boldsymbol{b} &= (\boldsymbol{f}(a) + \omega\alpha_F\boldsymbol{b} - \boldsymbol{f}(\xi))\cdot\boldsymbol{b}\\ &= (\boldsymbol{f}(a) - \boldsymbol{f}(\xi))\cdot\boldsymbol{b} + \omega\alpha_F|\boldsymbol{b}|^2\\ &= \boldsymbol{b}\cdot\boldsymbol{f}'(\zeta)(a-\xi) + \omega\alpha_F|\boldsymbol{b}|^2, \quad \text{for some} \quad \zeta\\ &\in (a,\xi) \geqslant -|\boldsymbol{b}||\boldsymbol{f}'(\zeta)\cdot\boldsymbol{n}_K||a-\xi| + \omega\alpha_F|\boldsymbol{b}|^2,\\ &\geqslant \left(\omega\alpha_F - \max_{|a-\xi|\leqslant\frac{1}{2}|\boldsymbol{b}|}\frac{1}{2}|\boldsymbol{f}'(\xi)\cdot\boldsymbol{n}_K|\right)|\boldsymbol{b}|^2.\end{aligned}$$

The conclusion is now a consequence of the definition of α_F, (3.6). The proof for the Lax-Freidrichs/Rusanov flux is similar. Using again the mean value Theorem and the fact that $\boldsymbol{f}$ is of class $\mathcal{C}^1$, we infer that

$$\begin{aligned}&\left(\hat{\boldsymbol{f}}_{\mathrm{LFR}}(a,\boldsymbol{b}) - \boldsymbol{f}(\xi)\right)\cdot\boldsymbol{b}\\ &= \left(\frac{1}{2}\left(\boldsymbol{f}\left(a+\frac{|\boldsymbol{b}|}{2}\right) - \boldsymbol{f}(\xi)\right) + \frac{1}{2}\left(\boldsymbol{f}\left(a-\frac{|\boldsymbol{b}|}{2}\right) - \boldsymbol{f}(\xi)\right) + \omega\alpha_F\boldsymbol{b}\right)\cdot\boldsymbol{b}\\ &= \frac{1}{2}\boldsymbol{b}\cdot\left(\boldsymbol{f}'(\zeta_1)\left(a+\frac{|\boldsymbol{b}|}{2}-\xi\right) + \boldsymbol{f}'(\zeta_2)\left(a-\frac{|\boldsymbol{b}|}{2}-\xi\right)\right) + \omega\alpha_F|\boldsymbol{b}|^2\end{aligned}$$

for some $\zeta_1 \in (a - \frac{|\boldsymbol{b}|}{2}, \xi)$ and some $\zeta_2 \in (\xi, a + \frac{|\boldsymbol{b}|}{2})$. This in turn implies that

$$\begin{aligned}\left(\hat{\boldsymbol{f}}_{\mathrm{LFR}}(a,\boldsymbol{b}) - \boldsymbol{f}(\xi)\right)\cdot\boldsymbol{b} &\geqslant -\frac{|\boldsymbol{b}|}{2}\max(|\boldsymbol{f}'(\zeta_1)\cdot\boldsymbol{n}_K|, |\boldsymbol{f}'(\zeta_2)\cdot\boldsymbol{n}_K|)\\ &\quad\times\left(\left|a+\frac{|\boldsymbol{b}|}{2}-\xi\right| + \left|a-\frac{|\boldsymbol{b}|}{2}-\xi\right|\right) + \omega\alpha_F|\boldsymbol{b}|^2,\\ &\geqslant \left(\omega\alpha_F - \max_{|a-\xi|\leqslant\frac{1}{2}|\boldsymbol{b}|}\frac{1}{2}|\boldsymbol{f}'(\xi)\cdot\boldsymbol{n}_K|\right)|\boldsymbol{b}|^2\end{aligned}$$

and we conclude again as above. □

Proposition 3.3. *Let $u_h \in \mathcal{C}^1([0,T]; X_h)$ solve (3.3). Assume that the numerical fluxes are defined by (3.4) or (3.8) and (3.5). The following discrete entropy inequality holds:*

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_K \frac{1}{2}u_h^2\,\mathrm{d}K + \int_{\partial K}\widehat{\boldsymbol{H}}\cdot\boldsymbol{n}_K\,\mathrm{d}\partial K \leqslant 0, \quad \forall K \in \mathcal{T}_h, \quad \forall t > 0, \tag{3.16}$$

where $\widehat{\boldsymbol{H}}$ is conservative and defined in (3.21), if the following holds:

$$\omega\alpha_F(u_h) + \delta\beta_F(u_h) - \max_{|\{u_h\}-\xi|\leqslant\frac{1}{2}[\![u_h]\!]|}\frac{1}{2}|\boldsymbol{f}'(\xi)\cdot\boldsymbol{n}_K| - \frac{1}{2c_i}\left\{\frac{\mu}{h}\right\} \geqslant 0. \tag{3.17}$$

Proof. Let us take $v_h = u_h$ in (3.3). Then

$$\begin{aligned}&\int_K \frac{1}{2}\partial_t|u_h|^2\,\mathrm{d}K - \int_K \boldsymbol{f}(u_h)\cdot\nabla u_h\,\mathrm{d}K + \int_{\partial K} u_h\hat{\boldsymbol{f}}\cdot\boldsymbol{n}_K\,\mathrm{d}\partial K\\ &\quad + \int_K \mu|\nabla u_h|^2\,\mathrm{d}K - \int_{\partial K} u_h\hat{\boldsymbol{g}}\cdot\boldsymbol{n}_K\,\mathrm{d}\partial K = 0.\end{aligned}$$

Let $\boldsymbol{K}(v) := \int_0^v \boldsymbol{f}(\tau)\,\mathrm{d}\tau$ be the anti-derivative of $\boldsymbol{f}$; then

$$\begin{aligned}\int_K \boldsymbol{f}(u_h)\nabla u_h\,\mathrm{d}K &= \int_K \boldsymbol{K}'(u_h)\nabla u_h\,\mathrm{d}K = \int_K \nabla\cdot(\boldsymbol{K}(u_h))\,\mathrm{d}K\\ &= \int_{\partial K}\boldsymbol{n}_K\cdot\boldsymbol{K}(u_h)\,\mathrm{d}\partial K.\end{aligned}$$

Let $\boldsymbol{x} \in \partial K$; we denote $u_h^i(\boldsymbol{x})$ the interior limit of u_h at $\boldsymbol{x}$ and $u_h^e(\boldsymbol{x})$ the exterior limit (i.e., $u_h^i(\boldsymbol{x}) = \lim_{\epsilon\to 0^-} u_h(\boldsymbol{x} + \epsilon\boldsymbol{n}_K)$ and $u_h^e(\boldsymbol{x}) = \lim_{\epsilon\to 0^+} u_h(\boldsymbol{x} + \epsilon\boldsymbol{n}_K)$). Using that $u_h^i\boldsymbol{n}_K = \{u_h\}\boldsymbol{n}_K + \frac{1}{2}[\![u_h]\!]$ we have

$$\begin{aligned}&-\int_K \boldsymbol{f}(u_h)\cdot\nabla u_h\,\mathrm{d}K + \int_{\partial K} u_h^i\hat{\boldsymbol{f}}\cdot\boldsymbol{n}_K\,\mathrm{d}\partial K\\ &= \int_{\partial K}\boldsymbol{n}_K\cdot(u_h^i\hat{\boldsymbol{f}} - \boldsymbol{K}(u_h^i))\,\mathrm{d}\partial K\\ &= \int_{\partial K}\boldsymbol{n}_K\cdot(\{u_h\}\hat{\boldsymbol{f}} - \{\boldsymbol{K}(u_h)\})\,\mathrm{d}\partial K\\ &\quad + \frac{1}{2}\int_{\partial K}([\![u_h]\!]\cdot\hat{\boldsymbol{f}} - \boldsymbol{n}_K\cdot(\boldsymbol{K}(u_h^i) - \boldsymbol{K}(u_h^e)))\,\mathrm{d}\partial K.\end{aligned}$$

Then using the mean-value Theorem, we obtain $\boldsymbol{K}(u_h^i) - \boldsymbol{K}(u_h^e)) = (u_h^i - u_h^e)\boldsymbol{f}(\xi)$ for some $\xi \in (u_h^i, u_h^e)$. The above equality can be re-written as follows:

$$\begin{aligned}&-\int_K \boldsymbol{f}(u_h)\cdot\nabla u_h\,\mathrm{d}K + \int_{\partial K} u_h^i\hat{\boldsymbol{f}}\cdot\boldsymbol{n}_K\,\mathrm{d}\partial K\\ &= \int_{\partial K}\boldsymbol{n}_K\cdot(\{u_h\}\hat{\boldsymbol{f}} - \{\boldsymbol{K}(u_h)\})\,\mathrm{d}\partial K + \frac{1}{2}\int_{\partial K}[\![u_h]\!]\cdot(\hat{\boldsymbol{f}} - \boldsymbol{f}(\xi))\,\mathrm{d}\partial K.\end{aligned}$$

Lemma 3.2 then implies

$$\begin{aligned}&-\int_K \boldsymbol{f}(u_h)\cdot\nabla u_h\,\mathrm{d}K + \int_{\partial K} u_h^i\hat{\boldsymbol{f}}\cdot\boldsymbol{n}_K\,\mathrm{d}\partial K\\ &\geqslant \int_{\partial K}\boldsymbol{n}_K\cdot(\{u_h\}\hat{\boldsymbol{f}} - \{\boldsymbol{K}(u_h)\})\,\mathrm{d}\partial K\\ &\quad + \frac{1}{2}\int_{\partial K}\left(\omega\alpha_F - \max_{|\{u_h\}-\xi|\leqslant\frac{1}{2}[\![u_h]\!]|}\frac{1}{2}|\boldsymbol{f}'(\xi)\cdot\boldsymbol{n}_K|\right)|[\![u_h]\!]|^2\,\mathrm{d}\partial K.\end{aligned} \tag{3.19}$$

Let us now focus on the viscous flux. Using again that $u_h^i\boldsymbol{n}_K = \{u_h\}\boldsymbol{n}_K + \frac{1}{2}[\![u_h]\!]$ we have

$$\begin{aligned}&\int_K \mu|\nabla u_h|^2\,\mathrm{d}K - \int_{\partial K} u_h\hat{\boldsymbol{g}}\cdot\boldsymbol{n}_K\,\mathrm{d}\partial K\\ &= \int_K \mu|\nabla u_h|^2\,\mathrm{d}K + \int_{\partial K}\{u_h\}(-\{\mu\nabla u_h\} + \delta\beta_F[\![u_h]\!])\\ &\quad\cdot\boldsymbol{n}_K\,\mathrm{d}\partial K\int_{\partial K}\left(-\frac{1}{2}[\![u_h]\!]\{\mu\nabla u_h\} + \frac{1}{2}\delta\beta_F|[\![u_h]\!]|^2\right)\mathrm{d}\partial K.\end{aligned}$$

The upon using the inequality $|ab| \leqslant \frac{\lambda}{2}a^2 + \frac{\lambda}{2}b^2$ with $\lambda = \frac{1}{2}$, together with the inverse inequality property (2.5) we infer that

$$\int_K \mu|\nabla u_h|^2\,\mathrm{d}K - \int_{\partial K} u_h\hat{\boldsymbol{g}}\cdot\boldsymbol{n}_K\,\mathrm{d}\partial K$$
$$\geqslant \int_{\partial K}\{u_h\}(-\{\mu\nabla u_h\}+\delta\beta_F[\![u_h]\!])\cdot\boldsymbol{n}_K\,\mathrm{d}\partial K + \int_K \frac{3}{4}\mu|\nabla u_h|^2\,\mathrm{d}K$$
$$+\frac{2c_i}{8}\int_{\partial K} h^i\mu^i|\nabla u_h^i|^2\,\mathrm{d}\partial K + \int_{\partial K}\left(\frac{1}{2}\delta\beta_F - \frac{1}{8c_i}\left(\frac{\mu^i}{h^i}+\frac{\mu^e}{h^e}\right)\right)|[\![u_h]\!]|^2\,\mathrm{d}\partial K$$
$$-\frac{c_i}{8}\int_{\partial K}(h^i\mu^i|\nabla u_h^i|^2 + h^e\mu^e|\nabla u_h^e|^2)\,\mathrm{d}\partial K,$$

which can be re-written

$$\int_K \mu|\nabla u_h|^2\,\mathrm{d}K - \int_{\partial K} u_h\hat{\boldsymbol{g}}\cdot\boldsymbol{n}_K\,\mathrm{d}\partial K$$
$$\geqslant \int_{\partial K}\left(\{u_h\}(\delta\beta_F[\![u_h]\!]-\{\mu\nabla u_h\}) + \frac{c_i}{8}[\![h\mu|\nabla u_h|^2]\!]\right)\cdot\boldsymbol{n}_K\,\mathrm{d}\partial K$$
$$+\int_K \frac{3}{4}\mu|\nabla u_h|^2\,\mathrm{d}K + \frac{1}{2}\int_{\partial K}\left(\delta\beta_F - \frac{1}{2c_i}\left\{\frac{\mu}{h}\right\}\right)|[\![u_h]\!]|^2\,\mathrm{d}\partial K. \quad (3.20)$$

The conclusion follows upon defining the following conservative flux:

$$\widehat{\boldsymbol{H}}(u_h) := \{u_h\}\hat{\boldsymbol{f}} - \{\boldsymbol{K}(u_h)\} + \{u_h\}(\delta\beta_F[\![u_h]\!]-\{\mu\nabla u_h\}) + \frac{c_i}{8}[\![h\mu|\nabla u_h|^2]\!], \quad (3.21)$$

and combining (3.19) and (3.20) into (3.18). □

Remark 3.4. Observe that the flux $\widehat{\boldsymbol{H}}$ is consistent with the entropy flux $\boldsymbol{F}(u) := \int_0^u \boldsymbol{f}'(\xi)E'(\xi)\,\mathrm{d}\xi$ associated with the entropy $E(u) := \frac{1}{2}u^2$. Actually, if u is a smooth function so that $\mu(\nabla u, \partial_t u) = 0$, $[\![u]\!] = 0$, $\{u\} = u$, we have $\hat{\boldsymbol{f}}(\{u\}, \nu[\![u]\!]) = \boldsymbol{f}(u)$, $\{\boldsymbol{K}(u)\} = \boldsymbol{K}(u)$ and

$$\widehat{\boldsymbol{H}}(u) = u\boldsymbol{f} - \boldsymbol{K}(u) = u\boldsymbol{f}(u) - \int_0^u \boldsymbol{f}(\tau)\,\mathrm{d}\tau = \int_0^u \boldsymbol{f}'(\tau)\tau\,\mathrm{d}\tau$$
$$= \int_0^u \boldsymbol{f}'(\tau)E'(\tau)\,\mathrm{d}\tau = \boldsymbol{F}(u).$$

Lemma 3.4. *Assuming β_F is defined by (3.7), the inequality (3.17) holds under each of the following conditions:*

(i) *α_F is defined by (3.6) and $\omega = 1$ and $\delta \geqslant \frac{1}{2c_i}$.*
(ii) *α_F is defined by $\alpha_F(u_h)|_F = \frac{1}{2}\max_{\boldsymbol{x}\in K_{1,F}\cup K_{2,F}}|\boldsymbol{f}'(u_h(\boldsymbol{x},t))|$, where recall that $F = K_{1,F}\cap K_{2,F}$, μ is defined by (3.10) and $\omega \geqslant 1 + \frac{c_{\max}}{c_i}$, $\delta = 0$.*

Proof. Statement (i) is evident. Statement (ii) is just a consequence of the definition of α_F together with the observation that $\{\mu/h\} \leqslant c_{\max}\max_{\boldsymbol{x}\in K_{1,F}\cup K_{2,F}}|\boldsymbol{f}'(u_h(\boldsymbol{x},t))|$. □

Remark 3.5. We have observed numerically that the proposed algorithm performs very well with the following pairs $(\omega,\delta)\in\{(1,1),(1,0),(0,1)\}$ and we conjecture that $\omega+\delta\geqslant 1$ and $\omega,\ \delta\geqslant 0$ should be a sufficient condition.

Remark 3.6. Note that α_F depends on the information on the approximate solution on the two neighboring cells in a way that is similar to what is done for the so-called Local-Lax–Friedrichs flux, see item (ii) from Lemma 3.4, and (4.4).

4. Implementation details

We give implementation details on the method in this section.

Table 4.1
Butcher tableaux for the explicit SSP RK3 (left) and RK4 (right) methods.

0			
1	1		
$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{4}$	
	$\frac{1}{6}$	$\frac{1}{6}$	$\frac{2}{3}$

0				
$\frac{1}{2}$	$\frac{1}{2}$			
$\frac{1}{2}$	0	$\frac{1}{2}$		
1	0	0	1	
	$\frac{1}{6}$	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{6}$

4.1. Time integration

Upon choosing a basis for X_h, say $\{\psi_i\}_{1\leqslant i\leqslant I}$, and denoting $U(t)\in\mathbb{R}^I$ the coordinate vector of $u_h(\cdot,t)$ in this basis, the semi-discrete problem (3.3) can be re-written as follows:

$$M\partial_t U = S(U(t)), \quad (4.1)$$

where the components of the nonlinear map $S:\mathcal{C}^1([0,T];\mathbb{R}^I)\to\mathcal{C}^0([0,T];\mathbb{R}^I)$ are defined as follows:

$$S_i(U(t)) = \int_K \boldsymbol{f}(u_h)\cdot\nabla\psi_i\,\mathrm{d}K - \int_{\partial K}\psi_i\hat{\boldsymbol{f}}\cdot\boldsymbol{n}_K\,\mathrm{d}\partial K$$
$$-\int_K \mu(\nabla u_h,\partial_t u_h)\nabla u_h\cdot\nabla\psi_i\,\mathrm{d}K + \int_{\partial K}\psi_i\hat{\boldsymbol{g}}\cdot\boldsymbol{n}_K\,\mathrm{d}\partial K, \quad (4.2)$$

for all $i\in\{1,\dots,I\}$, where K is the cell supporting ψ_i. Initialization of (4.1) is done by setting $U(0) = U_{0,h}$ where $U_{0,h}$ is the coordinate vector of $u_{0,h}$.

The time stepping to solve (4.1) is done using an explicit Runge–Kutta method. We have tested various techniques of second-, third- and fourth-order. All work similarly. Most of the tests reported in the present paper have been done using either the strongly-stable explicit Runge–Kutta algorithm RK3 described in [18] or the standard RK4 method. The Butcher tableaux of these two methods are shown in Table 4.1. We have not noticed significant differences in behavior of the entropy viscosity method when using either SSP RK3 or RK4, besides the fact that RK4 is more accurate than SSP RK3 on problems with smooth solutions.

One remaining detail that needs to be settled is the actual computation of the viscosity. We have chosen to make the viscosity explicit and it is assumed to be constant over the time interval $(t^n, t^n+\tau^n)$. It is computed by using the past values of u_h. More precisely, let u_h^n and t^n, τ^n, be the current solution, time, and time step, and let u_h^{n-1}, t^{n-1}, τ^{n-1}, u_h^{n-2}, t^{n-2}, and τ^{n-2} be the solution, time, and time step from the previous two times. The entropy residual is assumed to be constant over the time interval $(t^n, t^n+\tau^n)$ and is computed as follows:

$$R(u_h) = a_{n-2}E(u_h^{n-2}) + a_{n-1}E(u_h^{n-1}) + a_nE(u_h^n) + b_{n-1}\boldsymbol{f}'(u_{h,i}^{n-1})\nabla E(u^{n-1})$$
$$+ b_n\boldsymbol{f}'\nabla E(u_h^n). \quad (4.3)$$

The coefficients a_{n-2}, a_{n-1}, a_n and b_{n-1}, b_n depend on the time-stepping technique. The coefficients corresponding to the Crank-Nicolson and BDF2 schemes are shown in Table 4.2. $R(u_h)$ is initialized to 0 when the values of u_h^{n-2} and u_h^{n-1} are not available. The jump term (3.12) is computed by using u_h^n, i.e., $J(u_h) = h_K^{-1}\{\boldsymbol{f}'(u_h^n)\}\cdot[\![E(u_h^n)]\!]$.

The above approximation of the entropy residual implies that the entropy viscosity is of order $(\tau^2+h^p)h^2$. In particular the

Table 4.2
CN and BDF2 coefficients.

	a_{n-2}	a_{n-1}	a_n	b_{n-1}	b_n
CN	0	$-\frac{1}{\tau^{n-1}}$	$\frac{1}{\tau^{n-1}}$	$\frac{1}{2}$	$\frac{1}{2}$
BDF2	$\frac{\tau^{n-1}}{\tau^{n-2}(\tau^{n-2}+\tau^{n-1})}$	$-\frac{\tau^{n-2}+\tau^{n-1}}{\tau^{n-2}\tau^{n-1}}$	$\frac{\tau^{n-2}+2\tau^{n-1}}{\tau^{n-1}(\tau^{n-2}+\tau^{n-1})}$	0	1

second-order approximation of the time derivative implies that the entropy viscosity if fourth-order in smooth regions, i.e., this technique limits the overall accuracy of the method to fourth-order. This is not a issue if the time stepping to solve (4.1) is fourth-order or less or if the polynomial degree of the space approximation is at most 3. If necessary, it is possible to increase the consistency order of the method by using a higher-order approximation of the time derivative in (4.3).

4.2. Implementation using quadratures

All the integrals over interfaces and cells are approximated using quadratures. The quadrature points are defined on the reference interface $\widehat{F}$, say $\hat{\boldsymbol{x}}_1^\partial,\dots,\hat{\boldsymbol{x}}_S^\partial$, and reference element $\widehat{K}$, say $\hat{\boldsymbol{x}}_1,\dots,\hat{\boldsymbol{x}}_V$, and mapped to the current interface F and element K by the map T_F and T_K, respectively. We denote $G_F := \{T_F(\hat{\boldsymbol{x}}_1^\partial),\dots,T_F(\hat{\boldsymbol{x}}_S^\partial)\}$ the set of quadrature points over F and $G_K := \{T_K(\hat{\boldsymbol{x}}_1),\dots,T_K(\hat{\boldsymbol{x}}_V)\}$ the set of quadrature points over K.

The function α_F is piecewise constant over each interface F and defined as follows in the code that we used to run the numerical simulations reported in the rest of the paper:

$$\alpha_F(u_h)|_F = \max_{\boldsymbol{x}\in G_{K_{1,F}}\cup G_{K_{2,F}}} \frac{1}{2}|\boldsymbol{f}'(u_h(\boldsymbol{x},t))|. \tag{4.4}$$

The definition of $\beta_F(u_h)$ remains unchanged, i.e., see (3.7). We take $\omega = 1$ and $\delta = 1$, unless specified otherwise.

The viscosity μ is defined as follows over each cell $K \in \mathcal{T}_h$:

$$\mu|_K = \min(c_{\max}h_K \max_{\boldsymbol{x}\in G_K}|\boldsymbol{f}'(u_h(\boldsymbol{x},t))|, c_E h_K^2 \max(D_1(u_h),\dots,D_{n_E}(u_h))), \tag{4.5}$$

where the quantities $D_i(u_h),\ 1 \leqslant i \leqslant n_E$, are defined as follows:

$$D_i(u_h) := \max\left(\max_{\boldsymbol{x}\in G_K}|R_i(u_h)|, \max_{F\subset\partial K}\max_{\boldsymbol{x}\in G_F}|J_i(u_h)|\right)\frac{1}{N_i}. \tag{4.6}$$

The volume and interface residuals R_i and J_i are defined in (3.11) and (3.12). The normalization coefficient N_i is computed as follows:

$$N_i := \max_{fracK\in\mathcal{T}_h}\max_{\boldsymbol{x}\in G_K}\left|E_i(u_h(\boldsymbol{x},t)) - \frac{1}{|\Omega|}\int_\Omega E_i(u_h)\,\mathrm{d}\Omega\right|. \tag{4.7}$$

4.3. The viscosity parameters

As already mentioned in Remark 3.5, the proposed algorithm performs equally well with the following pairs $(\omega,\delta) \in \{(1,1),(1,0),(0,1)\}$. The largest CFL that can be reached with the pair $(\omega,\delta) = (1,1)$ is slightly smaller than with the pairs $(\omega,\delta) \in \{(1,0),(0,1)\}$ due to the combined explicit diffusion induced by both ω and δ being nonzero.

We have observed that the method gives satisfactory results by systemically using $c_E = 1$. We have also observed that $c_{\max} = \frac{1}{2k}$ is always a good choice. For instance, assuming that the space dimension is one and $f(u) = Vu\boldsymbol{e}_x$ (i.e., linear transport), setting $c_{\max} = \frac{1}{2}$ is exactly equivalent to first-order up-winding if the transport term is approximated with uniform centered second-order finite differences $(k = 1)$, thereby justifying the scaling $c_{\max} = 0.5/k$ for $k = 1$. It is sometimes possible to be greedy when solving linear problems by using a smaller value for $c_{\max}$. Lowering $c_{\max}$ usually allows for a larger CFL.

Remark 4.1. (DG0) We can take $c_{\max} = 0$ (together with $\omega = 1$ and $\delta = 0$) in the particular case of piecewise constant approximation (i.e., $k = 0$) since it is known that the upwind DG0 flux is equivalent to first-order viscosity which is exactly the amount of dissipation that is required.

5. Numerical illustrations

We illustrate the capabilities of the above technique by solving various scalar conservation laws. The two-dimensional problems are solved using the deal.II finite element library [4,3].

5.1. One-dimensional transport with smooth data

We consider the linear transport problem

$$\partial_t u + \partial_x u = 0, \quad \forall (x,t) \in (0,1)\times\mathbb{R}_+, \quad u|_{t=0} = u_0(x) \quad \forall x \in (0,1), \tag{5.1}$$

with periodic boundary conditions and with the following smooth initial data $u_0(x) = \sin(2\pi x)$. The time stepping is done with the RK4 method. The computations are done with the following parameters: $c_{\max} = 0.5/k,\ c_E = 1, \omega = 1,\ \delta = 1$ and CFL $= 0.5$. We did not obtain significantly different results by using $\delta = 0$ instead of $\delta = 1$.

The solution is computed up to time $T = 1$ on various uniform meshes for polynomial degrees $1, 2$ and 3. We report the errors measured in the L^1- and L^2-norm at $T = 1$ in Table 5.1. The observed convergence rates are $2, 3$ and 4 for both norms, respectively. The method gives optimal convergence orders in one space dimension.

5.1.1. One-dimensional transport with non-smooth data

We now evaluate the performance of the method with non smooth data. We consider again the linear transport Eq. (5.1) with periodic boundary conditions, but this time the initial data is

$$u_0(x) = \begin{cases} 1 & \text{if } 0.25 \leqslant x \leqslant 0.75, \\ 0 & \text{otherwise.} \end{cases} \tag{5.2}$$

Table 5.1
Convergence test for one-dimensional linear transport with smooth data, $\mathbb{P}_1$, $\mathbb{P}_2$, and $\mathbb{P}_3$ finite elements.

h	dofs	L^1-error	Rate	L^2-error	Rate
(a) $\mathbb{P}_1$					
2e−01	10	6.355e−01	–	6.951e−01	–
1e−01	20	5.325e−01	0.26	5.917e−01	0.23
5e−02	40	2.276e−01	1.23	2.573e−01	1.20
2.5e−02	80	3.725e−02	2.61	4.276e−02	2.59
1.25e−02	160	5.198e−03	2.84	6.023e−03	2.83
6.25e−03	320	5.155e−04	3.33	6.097e−04	3.30
3.125e−03	640	6.864e−05	2.91	8.338e−05	2.87
1.5625e−03	1280	9.741e−06	2.82	1.262e−05	2.73
7.8125e−04	2560	1.631e−06	2.58	2.248e−06	2.49
3.90625e−04	5120	3.435e−07	2.25	4.712e−07	2.25
1.953125e−04	10,240	8.068e−08	2.09	1.090e−07	2.11
9.765625e−05	20,480	1.977e−08	2.03	2.710e−08	2.01
(b) $\mathbb{P}_2$					
2e−01	15	5.570e−01	–	6.159e−01	–
1e−01	30	2.854e−01	0.97	3.217e−01	0.94
5e−02	60	1.771e−02	4.01	2.026e−02	3.99
2.5e−02	120	8.131e−04	4.45	8.920e−04	4.51
1.25e−02	240	7.221e−05	3.49	8.117e−05	3.46
6.25e−03	480	7.442e−06	3.28	8.367e−06	3.28
3.125e−03	960	8.357e−07	3.16	9.433e−07	3.15
1.5625e−03	1920	9.859e−08	3.08	1.117e−07	3.08
7.8125e−04	3840	1.196e−08	3.04	1.359e−08	3.04
3.90625e−04	7680	1.483e−09	3.01	1.687e−09	3.01
(c) $\mathbb{P}_3$					
2e−01	20	4.610e−01	–	5.121e−01	–
1e−01	40	2.405e−01	0.94	2.678e−01	0.94
5e−02	80	3.817e−02	2.66	4.319e−02	2.63
2.5e−02	160	5.457e−04	6.13	6.347e−04	6.09
1.25e−02	320	2.850e−05	4.26	3.247e−05	4.29
6.25e−03	640	1.496e−06	4.25	2.026e−06	4.00
3.125e−03	1280	9.021e−08	4.05	1.279e−07	3.98
1.5625e−03	2560	5.554e−09	4.02	8.050e−09	3.99
7.8125e−04	5120	3.502e−10	3.99	5.036e−10	4.00

Table 5.2
Convergence test for one-dimensional linear transport with non-smooth data. Convergence rates for L^1- and L^2-norms vs. polynomial degrees.

Polyn. degree	1	2	3	4	5
Rate (L^1-norm)	0.76	0.82	0.87	0.91	0.94
Rate (L^2-norm)	0.38	0.42	0.45	0.46	0.48

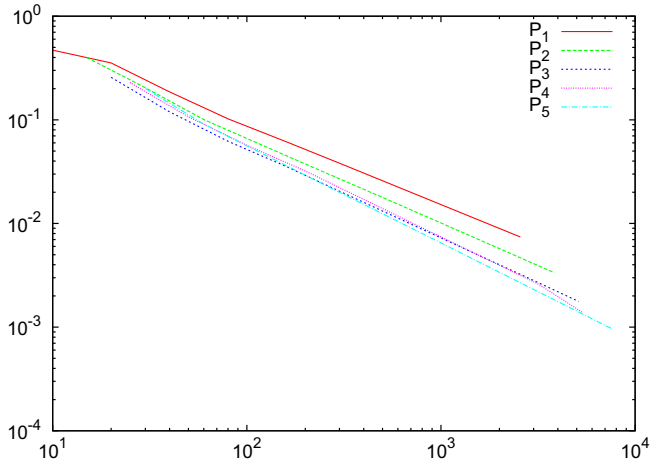


Fig. 5.1. Linear transport with smooth data. L^1-norm of error at $T = 1$ vs. nb. dofs for $\mathbb{P}_1, \ldots, \mathbb{P}_5$.

The time stepping is done with the RK4 method. The computations are done with the following parameters: $c_{\max} = 0.5/k$, $c_E = 1$, $\omega = 1$, $\delta = 1$ and CFL $= 0.5$.

The computations are done on 9 uniform meshes going from 5 cells up to $1280 = 2^8 \times 5$ cells. The errors in the L^1- and L^2-norm are computed at $T = 1$. The method is tested with polynomials in $\mathbb{P}_1, \ldots, \mathbb{P}_5$. The mean value of the observed convergence orders are reported in Table 5.2. The convergence rates in the L^1- and L^2-norm are compatible with the estimates $\frac{k+\frac{1}{2}}{k+1}$ and $\frac{1}{2}\frac{k+\frac{1}{2}}{k+1}$, respectively. Recall that, assuming the time to be continuous, the standard a priori error estimate for the DG approximation of (5.1) is $\|u - u_h\|_{L^\infty((0,T);L^2)} \leqslant ch^{k+\frac{1}{2}}\|u_0\|_{H^{k+1}}$. Since the initial data (5.2) is in $H^{\frac{1}{2}-\epsilon}$ for all $\epsilon > 0$, the real method of interpolation implies that $\|u - u_h\|_{L^2} \leqslant c'h^{(\frac{1}{2}-\epsilon)\frac{k+\frac{1}{2}}{k+1}}$ (see for instance [34] or [46, Chapter 22]). It is possible to get rid of the arbitrary number ϵ by using the Besov spaces $B^s_{2,\infty}$. A similar argument holds for the error in the L^1-norm by interpolating between $W^{k+1,1}$ and $L^1 := W^{0,1}$ using the fact that u_0 is in $W^{1-\epsilon,1}$ for all $\epsilon > 0$. In conclusion the higher the polynomial degree the more accurate the method. This test shows that higher-order methods perform better than lower-order methods even on non-smooth solutions. This statement is reinforced by looking at Fig. 5.1. We show in this figure the L^1-norm of the error at $T = 1$ as a function of the total number of degrees of freedom (dofs) for the $\mathbb{P}_1, \ldots, \mathbb{P}_5$ approximations. For any given number of degrees of freedom the error is a decreasing function of the polynomial degree k. Note that the $\mathbb{P}_3$, $\mathbb{P}_4$ and $\mathbb{P}_5$ errors almost coincide because the accuracy is limited to fourth-order by our using RK4.

5.2. Two-dimensional tests

We consider the domain $\Omega = \{\boldsymbol{x} \in \mathbb{R}^2, |\boldsymbol{x}| < 1\}$, the vector field $\boldsymbol{V}(\boldsymbol{x}) = 2\pi(-y, x)$, where $\boldsymbol{x} = (x, y)$, and the two-dimensional transport problem

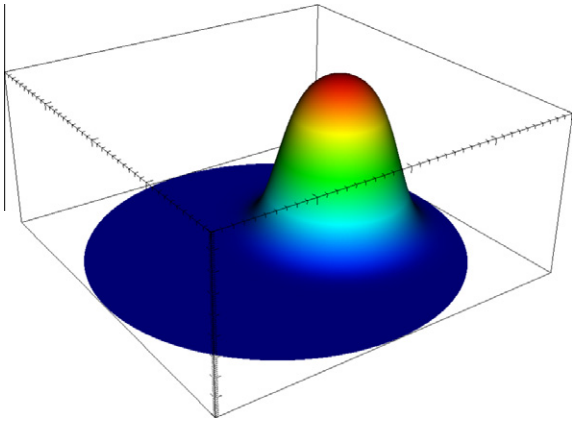


Fig. 5.2. Linear transport with smooth data. Graph of $\mathbb{Q}_3$ solution at $T = 1$.

Table 5.3
Convergence test with $\mathbb{Q}_2$ finite elements.

h	dofs	L^1-norm	Rate	L^2-norm	Rate
(a) Upwing flux					
0.2929	180	2.685e−01	–	2.482e−01	–
0.1768	720	6.570e−02	2.03	7.464e−02	1.73
0.0937	2880	8.774e−03	2.90	1.162e−02	2.68
0.0496	11,520	7.655e−04	3.52	1.137e−03	3.35
0.0252	46,080	5.721e−05	3.74	1.132e−04	3.33
0.0128	184,320	5.699e−06	3.33	1.685e−05	2.81
0.0064	737,280	7.101e−07	3.01	2.261e−06	2.90
(b) Centered flux					
0.2929	180	2.633e−01	–	2.415e−01	–
0.1768	720	9.589e−02	1.46	9.972e−02	1.28
0.0937	2880	1.895e−02	2.34	2.299e−02	2.12
0.0496	11,520	2.036e−03	3.22	2.469e−03	3.22
0.0252	46,080	1.895e−04	3.43	1.961e−04	3.65
0.0128	184,320	1.826e−05	3.38	1.604e−05	3.61

$$\partial_t u + \nabla \cdot (\boldsymbol{V}u) = 0, \tag{5.3}$$

$$u(\boldsymbol{x}, 0) = \frac{1}{2}\left(1 - \tanh\left(\frac{(\boldsymbol{x} - \boldsymbol{r}_0)^2}{a^2} - 1\right)\right), \tag{5.4}$$

with $a = 0.3$ and $\boldsymbol{r}_0 = (0.4, 0)$. Note that $\nabla \cdot \boldsymbol{V} = 0$ and the boundary of Ω is a characteristics boundary (i.e., $\boldsymbol{V} \cdot \boldsymbol{n}|_{\partial\Omega} = 0$). The graph of the solution at $T = 1$ using $\mathbb{Q}_3$ approximation is shown in Fig. 5.2.

Two convergence tests are done using $\mathbb{Q}_2$ discontinuous finite elements on various grids composed of $\mathbb{Q}_6$ quadrangles (meaning that the mappings $T_K : \hat{K} \to K$ are $\mathbb{Q}_6$). In the first series of tests we use the upwind flux in (3.4), i.e., $\omega = 1$, and in the second test we use the centered flux, i.e., $\omega = 0$. In both cases we take $\delta = 1$ in (3.5). The other parameters are set as follows: $c_E = 0.5$, $c_{\max} = 0.1/k$, $\delta = 1.0$, CFL $= 0.25$. The time stepping is done using RK4. The goal of these tests is to evaluate the importance of the definition of the inviscid flux (3.4).

The errors measured at $T = 1$ in the L^1- and L^2-norm together with the respective convergence orders are shown in Table 5.3(a) and Table 5.3(b). It is remarkable that the convergence order is 3 irrespective of the definition of the flux. This test confirms that the present DG implementation of the entropy viscosity method works properly by using centered fluxes.

5.2.1. Convergence tests for two-dimensional inviscid Burgers equation

We consider the inviscid Burgers equation in two space dimensions. Let $\boldsymbol{V} = (1, 1)$ be a constant vector field and consider the conservation equation

$$\partial_t u + \nabla \cdot \left(\frac{1}{2}u^2\boldsymbol{V}\right) = 0, \tag{5.5}$$

Table 5.4
Convergence test for two-dimensional inviscid Burgers equation, $T = 0.5$, $\mathbb{Q}_1$, $\mathbb{Q}_2$, and $\mathbb{Q}_3$ discontinuous FEs.

h	dofs	L^1-error	Rate	L^2-error	Rate
(a) $\mathbb{Q}_1$					
0.5000	16	5.845e−01	–	6.527e−01	–
0.2500	64	2.788e−01	1.07	3.782e−01	0.79
0.1250	256	1.127e−01	1.31	2.328e−01	0.70
0.0625	1024	6.415e−02	0.81	1.762e−01	0.40
0.0312	4096	3.174e−02	1.02	1.248e−01	0.50
0.0156	16,384	1.697e−02	0.90	9.238e−02	0.43
0.0078	65,536	8.506e−03	1.00	6.678e−02	0.47
(b) $\mathbb{Q}_2$					
0.5000	36	4.843e−01	–	5.596e−01	–
0.2500	144	1.500e−01	1.69	2.601e−01	1.11
0.1250	576	7.441e−02	1.01	1.911e−01	0.44
0.0625	2304	3.702e−02	1.01	1.370e−01	0.48
0.0312	9216	1.948e−02	0.93	1.010e−01	0.44
0.0156	36,864	9.718e−03	1.00	7.157e−02	0.50
0.0078	147,456	5.118e−03	0.93	5.308e−02	0.43
(c) $\mathbb{Q}_3$					
0.5000	64	4.510e−01	–	5.126e−01	–
0.2500	256	1.237e−01	1.87	2.332e−01	1.14
0.1250	1024	5.628e−02	1.14	1.638e−01	0.51
0.0625	4096	2.945e−02	0.93	1.223e−01	0.42
0.0312	16,384	1.445e−02	1.03	8.728e−02	0.49
0.0156	65,536	7.618e−03	0.92	6.395e−02	0.45
0.0078	262,144	3.731e−03	1.03	4.527e−02	0.50

subject to the following initial condition

$$u(x,y,0) = u_0(x,y) = \begin{cases} -0.2 & \text{if } x < 0.5 \quad \text{and} \quad y > 0.5; \\ -1 & \text{if } x > 0.5 \quad \text{and} \quad y > 0.5; \\ 0.5 & \text{if } x < 0.5 \quad \text{and} \quad y < 0.5; \\ 0.8 & \text{if } x > 0.5 \quad \text{and} \quad y < 0.5. \end{cases} \tag{5.6}$$

The entropy pair that we choose for this problem is $(E(u) = \frac{1}{2}u^2,\ \boldsymbol{F}(u) = \frac{1}{3}u^3\boldsymbol{v})$. The solution is computed at $T = 0.5$ to facilitate comparisons with [9,26]. The parameters for this computation are $c_E = 1.0,\ c_{\max} = 0.25/k,\ \omega = 1,\ \delta = 1.0$, CFL $= 0.25$.

The exact solution to this problem for $t > 0$ is given in [22, Section 4.1]. We compute the L^1- and L^2-norm of the error and the corresponding convergence rate at $T = 0.5$ on various uniform meshes composed of squares. The results for $\mathbb{Q}_1$, $\mathbb{Q}_2$ and $\mathbb{Q}_3$, approximations are shown in Table 5.4. We observe that the convergence rates are 1 and $\frac{1}{2}$ in the L^1- and L^2-norm, respectively and are independent of the polynomial degree of the approximation.

The graph of the solution at $T = 0.5$ is shown in the left panel in Fig. 5.3 and the entropy viscosity filed is shown in the right panel. The entropy viscosity focuses in the shock as anticipated.

6. Extension to compressible gas dynamics

We extend the above technique to the equations of compressible gas dynamics in this section.

6.1. Details on implementation

We consider the Euler equations for a perfect gas written in conservative form

$$\partial_t \boldsymbol{c} + \nabla \cdot (\boldsymbol{f}(\boldsymbol{c})) = 0, \quad \boldsymbol{c} = \begin{pmatrix} \rho \\ \boldsymbol{m} \\ E \end{pmatrix}, \quad \boldsymbol{f}(\boldsymbol{c}) = \begin{pmatrix} \boldsymbol{m} \\ \boldsymbol{m} \otimes \frac{\boldsymbol{m}}{\rho} + p\mathbb{I} \\ \frac{\boldsymbol{m}}{\rho}(E+p) \end{pmatrix}, \tag{6.1}$$

where the independent variables are the density ρ, the momentum vector field $\boldsymbol{m}$ and the total energy E. The velocity vector field $\boldsymbol{u}$ is defined by $\boldsymbol{u} := \boldsymbol{m}/\rho$. The symbol $\mathbb{I}$ denotes the identity matrix in $\mathbb{R}^d$. The pressure is expressed via the equation of state of ideal gases:

$$p = \rho T, \quad \text{with} \quad T = (\gamma - 1)e, \quad e = \left(\frac{E}{\rho} - \frac{1}{2}\boldsymbol{u}^2\right), \tag{6.2}$$

where γ is the adiabatic constant and T is the temperature. The Euler system is known to have a physical entropy functional

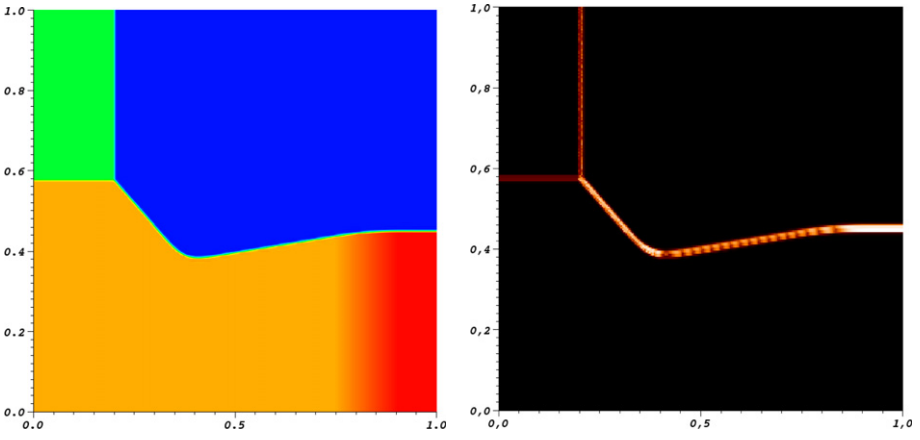
$$S(p,\rho) = \frac{\rho}{\gamma - 1}\log(p/\rho^\gamma), \tag{6.3}$$

that satisfies the following inequality

$$\partial_t S + \nabla \cdot (\boldsymbol{u}S) \geqslant 0. \tag{6.4}$$

The above inequality becomes an equality if all the fields are smooth.

The solution technique is almost identical to what we have described in Section 4 for scalar conservation equations, the only significant difference being in the adopted regularization of (6.1). Following [21,22], we replace the Euler system by the Navier–Stokes system where the viscosity and the thermal diffusivity are proportional to the entropy residual. Setting



(a) Solution (b) Viscosity

Fig. 5.3. Two-dimensional Burgers equation, $T = 0.5$, $\mathbb{Q}_3$ polynomials.

Table 5.5
Data for one-dimensional Riemann problems from [47].

Test	T	x_0	ρ_l	u_l	p_l	ρ_r	u_r	p_r
1	0.200	0.300	1.000	0.750	1.000	0.125	0.000	0.100
2	0.012	0.500	1.000	0.000	1000.	1.000	0.000	0.010
3	0.035	0.400	5.99924	19.5975	460.894	5.99242	−6.19633	46.095

$\nabla^s \boldsymbol{u} := \frac{1}{2}(\nabla \boldsymbol{u} + (\nabla \boldsymbol{u})^T)$, the inviscid flux $\boldsymbol{f}$ in (6.1) is augmented with the following viscous flux:

$$\boldsymbol{g}(\boldsymbol{c}) = -\begin{pmatrix} \nu \nabla \log \rho \\ \mu \nabla^s \boldsymbol{u} \\ \mu \nabla^s \boldsymbol{u} \cdot \boldsymbol{u} + \kappa \nabla e \end{pmatrix}. \tag{6.5}$$

Note that the Navier–Stokes viscous fluxes are compatible with the entropy inequality (6.4). Note also that the presence of an artificial mass diffusion may be objectionable since this term is not present in the Navier–Stokes system. This term must be understood only as a numerical device whose objective is to stabilize the mass conservation equation. Virtually all numerical schemes induce some kind of dissipation on the mass conservation in one way or another. For instance the viscous flux associated with the Lax–Friedrichs schemes with $\mathbb{P}_0$ approximation in one space dimension is

$$\boldsymbol{g}_{\mathrm{LR}}(\boldsymbol{c}) = -(|u| + \sqrt{\gamma T})h \begin{pmatrix} \nabla \rho \\ \nabla \boldsymbol{m} \\ \nabla E \end{pmatrix}. \tag{6.6}$$

See the appendix of [39] for more details on this issue.

The viscosity μ is assumed to be piecewise constant over the mesh; it is defined as follows over each cell $K \in \mathcal{T}_h$:

$$\mu|_K = \min(c_{\max} h_K \max_{\boldsymbol{x} \in K}(\rho_h(|u_h(\boldsymbol{x}, t)| + \sqrt{\gamma T_h})),\ c_E h_K^2 D), \tag{6.7}$$

where $D(u_h) := \max(\max_{\boldsymbol{x} \in K} |R(u_h)|, \max_{\boldsymbol{x} \in \partial K} |J(u_h)|)$ is defined as follows:

$$R(u_h) := \partial_t S(p_h, \rho_h) + \nabla \cdot \boldsymbol{u}_h S(p_h, \rho_h)), \tag{6.8}$$

$$J(u_h) := h_K^{-1} |\boldsymbol{n} \cdot [\![\boldsymbol{u}_h S(p_h, \rho_h)]\!]|. \tag{6.9}$$

R is the entropy residual and J is the entropy residual induced by the jump of the entropy across the boundary ∂K. Note that no normalization is necessary, since D has the right dimension. The artificial thermal diffusivity and the mass conservation viscosity are defined as follows[2]

$$\kappa|_K = \mathcal{P}_T \mu|_K, \quad \nu|_K = \mathcal{P}_\rho \mu|_K. \tag{6.10}$$

The DG implementation of the viscous flux (6.5) is analogous to what we have done in (3.5) for scalar conservation equations. We omit the details for brevity. We have observed that one important aspect of the implementation is that the pair (ω, δ) must be so that $\omega > 0$. For all the numerical tests reported hereafter we used $\omega = 1$ and $\delta = 1$.

6.2. One-dimensional Riemann problem

We now consider three classical one-dimensional Riemann problems taken from [47]. These tests are defined by the set of data reported in Table 5.5. The indices l and r designate the left ($x \leqslant x_0$) and right ($x \geqslant x_0$) states, respectively. The computational domain is $\Omega = (0, 1)$ and the final time of the computation is denoted T.

[2] During the review of the manuscript, the authors [19] have determined that using $\nu_K = \frac{\gamma-1}{\gamma}\kappa|_K$, i.e., $\mathcal{P}_\rho = \frac{\gamma-1}{\gamma}\mathcal{P}_T$, implies that the regularized system satisfies a minimum principle on the specific entropy (see Tadmor [45] for an argumentation on the importance of the minimum principle on the specific entropy). The computations reported in the present paper have not been re-done with this scaling.

Table 5.6
Entropy viscosity parameters for the one-dimensional Riemann problems.

Test	CFL	$c_{\max}$	C_E	$\mathcal{P}_\rho$	$\mathcal{P}_T$	ω	δ
1	0.33	0.5/k	1.0	0.15	0.15	1	1
2	0.33	0.5/k	1.0	0.15	0.15	1	1
3	0.33	0.5/k	1.0	0.15	0.15	1	1

We take $\gamma = 1.4$ for heat capacity ratio in all these tests. The computations are done using $\mathbb{P}_3$ discontinuous finite elements on a uniform mesh composed of 200 cells and the time stepping is done with RK4. The computational parameters are reported in Table 5.6.

The results for each test case are shown in Figs. 6.1, 6.2, 6.3. We observe that the results are non-oscillatory even though we used cubic elements. We have verified that the computed solutions are the entropy solutions.

6.3. Contact vs. shocks

We have mentioned in the introduction that we expect the entropy residual to "make an automatic distinction between shocks and contact discontinuities". We illustrate this conjecture by solving again the one-dimensional Riemann problem called Test 1 in the above section. We show in Fig. 6.4 the (piece-wise linear reconstruction of) coefficient $\kappa(x)$ at the end of the simulation on various uniform meshes of mesh size $h_i = 2^{1-i} \times 10^{-2}$, $i = 1, \ldots 8$ using piecewise linear polynomials. The distribution is normalized by its maximum value in the shock. We observe that the ratio of the artificial diffusion in the contact discontinuity to that in the shock goes to zero as the mesh size goes to zero. The observed convergence rate in the contact discontinuity is $\mathcal{O}(h^{\frac{5}{4}})$, thereby supporting the above conjecture.

6.4. Two-dimensional tests

The method is now tested in two space dimensions on two classical benchmark problems.

6.4.1. Riemann 12

We consider Problem 12 from [36]. It is a two-dimensional Riemann problem developing complex structures involving shocks and contacts. The computational domain is $\Omega = (0, 1)^2$. The heat capacity ratio is $\gamma = 1.4$ and the initial data is

$$\begin{aligned} &p = 1, \quad \rho = 4/5, \quad \boldsymbol{u} = (0, 0), \quad 0 < x < 1/2, \quad 0 < y < 1/2, \\ &p = 1, \quad \rho = 1, \quad \boldsymbol{u} = (3/\sqrt{17}, 0), \quad 0 < x < 1/2, \quad 1/2 < y < 1, \\ &p = 1, \quad \rho = 1, \quad \boldsymbol{u} = (0., 3/\sqrt{17}), \quad 1/2 < x < 1, \quad 0 < y < 1/2, \\ &p = 2/5, \quad \rho = 17/32, \quad \boldsymbol{u} = (0, 0), \quad 1/2 < x < 1, \quad 0.5 < y < 1. \end{aligned} \tag{6.11}$$

Due to the finite speed of propagation of perturbations, the solution of the problem in $(0, 1)^2$ is identical to the restriction to $(0, 1)^2$ of the solution to the Riemann problem in $\mathbb{R}^2$ up to time $t^* := \frac{s}{2(s^2+0.6)} > 0.32$, where $s = \frac{3}{\sqrt{17}}$.

The control parameters of the entropy viscosity are $c_E = 1$, $c_{\max} = 0.5/k$, $\omega = 1$, $\delta = 1$, $\mathcal{P}_\rho = 0.25$, $\mathcal{P}_T = 0.25$. The time stepping is done with RK4 with CFL $= 0.25$. The computations

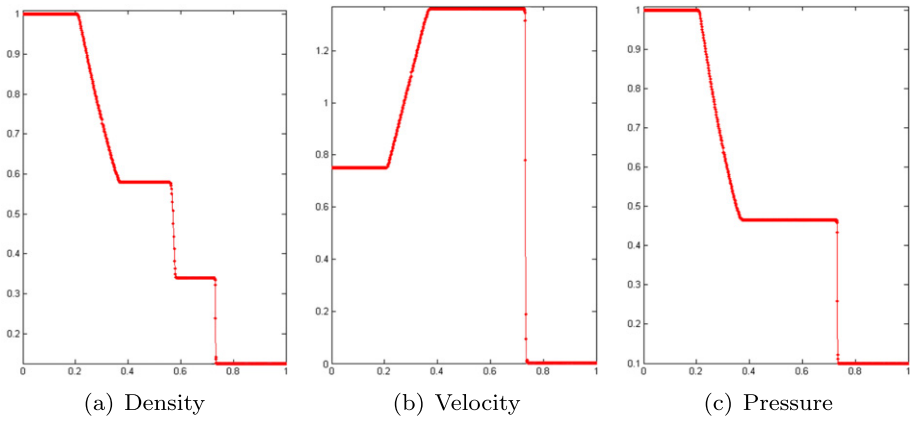


Fig. 6.1. Test 1. Density (left), velocity (center), pressure (right).

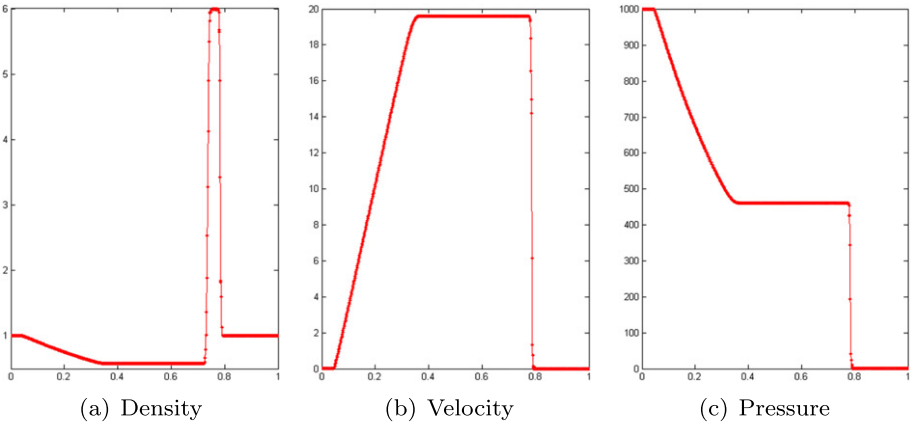


Fig. 6.2. Test 2. Density (left), velocity (center), pressure (right).

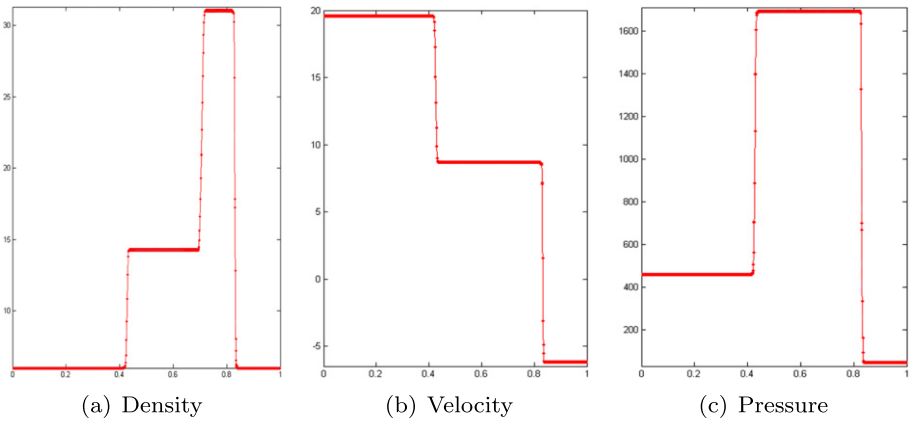


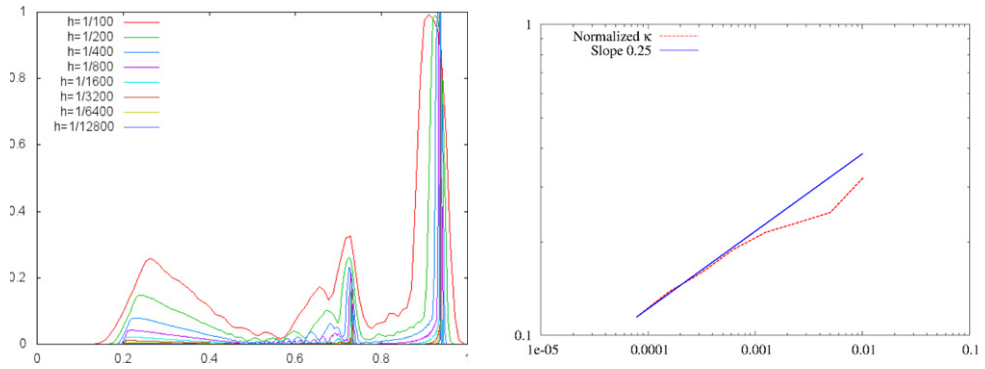
Fig. 6.3. Test 3. Density (left), velocity (center), pressure (right).

are done with $\mathbb{Q}_1, \mathbb{Q}_2$ and $\mathbb{Q}_3$ discontinuous finite elements on a grid composed of $16384 = 128^2$ quadrangular cells. The total number of scalar degrees of freedom for the $\mathbb{Q}_1, \mathbb{Q}_2$ and $\mathbb{Q}_3$ approximations are $65536, 146456, 262144$, respectively, i.e., $4^7 \times (k+1)$.

We show in Fig. 6.5 the density field at $T = 0.2 < T^*$ for the $\mathbb{Q}_1, \mathbb{Q}_2$ and $\mathbb{Q}_3$ approximations. The results compare well with those from [36]. The shocks and the fine structures that develop behind them are very well described. The method behaves well as the polynomial degree of the approximation increases.

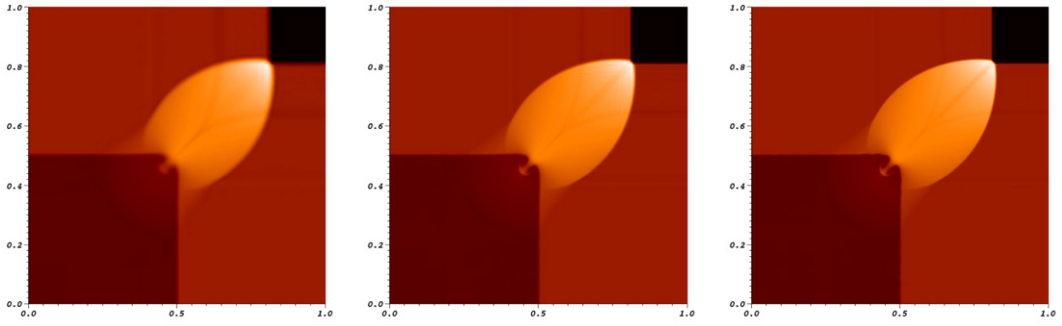
6.4.2. Mach 3 step

We finish the series of numerical examples by considering the flow past a forward facing step in a wind tunnel at Mach 3. This benchmark test has been proposed in [16] and has been popular-



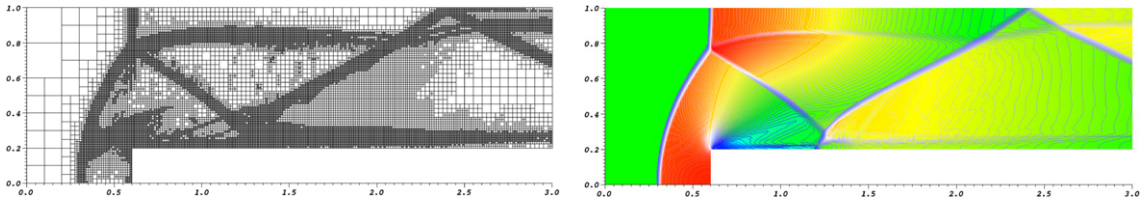
(a) Normalized κ for various mewh-sizes (b) normalized $\kappa_{\max}$ in contact vs.mesh-size

Fig. 6.4. Artificial thermal diffusivity for Test 1 at $T = 0.25$, normalized by its maximum value in the shock.



(a) $\mathbb{Q}_1$ (b) $\mathbb{Q}_2$ (c) $\mathbb{Q}_3$

Fig. 6.5. Riemann Problem 12 at $T = 0.2$. $\mathbb{Q}_1$ (left), $\mathbb{Q}_2$ (center), $\mathbb{Q}3$ (right), 128×128 cells.



(a) Mesh (b) density

Fig. 6.6. Mach 3 flow past a forward facing step at $T = 4$, $\mathbb{Q}_1$ approximation, 25,500 cells. Mesh (left), density (right).

ized by Woodward and Colella's extensive study [50]. The geometry of the domain is shown in Fig. 6.6. The initial data and inflow boundary conditions are specified in terms of the primitive variables

$$\left.\begin{array}{l}(\rho,\boldsymbol{u},p)^T(x,y,0)\\(\rho,\boldsymbol{u},p)^T(0,y,t)\end{array}\right\} = (1.4,(3.0,0.0),1.0)^T. \tag{6.12}$$

The outflow boundary at $\{x = 3\}$ is free. The slip condition $\boldsymbol{u}\cdot\boldsymbol{n} = 0$ is specified on the solid wall of the tunnel, $\boldsymbol{n}$ being the unit outward normal on $\partial\Omega$.

We solve this problem by using $\mathbb{Q}_1$ discontinuous finite elements and an adaptive mesh refinement strategy built in deal.II. The local error indicator is $h_K \int_{\partial K} |\{\partial_n \rho\}|^2 \, \mathrm{d}\partial K$ and we limit the number of refinement levels to 7. The refining and coarsening is done so that 30% of the cells with the largest error indicators are refined whereas 10% of the cells with the smallest error indicators are coarsen. Mesh adaption is done at every time step. The time stepping in done with RK4 at CFL $= 0.2$. The parameters of the entropy viscosity are $c_E = 1.0$, $c_{\max} = 0.5/k$, $\omega = 1$, $\delta = 1$, $\mathcal{P}_\rho = 0.05$, $\mathcal{P}_T = 0.1$. This example is included in the present paper as an illustration that adaptive refinement can be used in conjunction with the proposed technique. Our purpose is not to present a refined solution.

The mesh obtained at $T = 4$ is shown in Fig. 6.6 and the density field is shown in 6.6(b). This solutions agrees, at least in the eyeball norm, with other reference solutions that can be found in the literature. The contact discontinuity emerging from the three-shock interaction point is present in both simulations and is captured satisfactorily, considering the small number of elements used in this computation (25500 cells). We observe a spurious Mach stem at the bottom wall. This artifact is due to the entropy layer produced at the bottom wall by the corner singularity. The phenomenon is well explained in [15, p. 217]. It can be removed by modifying the scheme in some ad hoc fashion close to the corner, aggressively refining the mesh at the corner, or simply regularizing the corner.

7. Conclusions

We have extended the notion of entropy viscosity to the discontinuous Galerkin framework for scalar conservation laws and the compressible Euler equations. The method is easy to implement on unstructured grids with arbitrary polynomial approximations. Limiters and high-order polynomial reconstructions are avoided by using a viscous stabilization that is compatible with the entropy inequality. The method has been shown to behave well on many benchmark tests. We have observed that the DG method is at least as efficient as CG and spectral element versions of the entropy viscosity method on the same class of problems. The use of DG elements provides a natural setting for coupling the hydrodynamics and radiation transport equations because DG FEM methods are vastly superior to CG methods for radiative transfer calculations [31].

References

[1] D. Arnold, An interior penalty finite element method with discontinuous elements, SIAM J. Numer. Anal. 19 (1982) 742–760.
[2] D. Arnold, F. Brezzi, B. Cockburn, L. Marini, Unified analysis of discontinuous Galerkin methods for elliptic problems, SIAM J. Numer. Anal. 39 (5) (2001/02) 1749–1779.
[3] W. Bangerth, R. Hartmann, G. Kanschat, deal.II – a general purpose object oriented finite element library, ACM Trans. Math. Softw. 33 (4) (2007) 24/1–24/27.
[4] W. Bangerth, G. Kanschat, deal.II Differential Equations Analysis Library, Technical Reference. Available from: <http://www.dealii.org>.
[5] G.E. Barter, D.L. Darmofal, Shock capturing with PDE-based artificial viscosity for DGFEM: part I formulation, J. Comput. Phys. 229 (5) (2010) 1810–1827.
[6] F. Bassi, S. Rebay, A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier–Stokes equations, J. Comput. Phys. 131 (2) (1997) 267–279.
[7] F. Bassi, S. Rebay, G. Mariotti, S. Pedinotti, M. Savini, A high-order accurate discontinuous finite element method for inviscid and viscous turbomachinery flows, in: R. Decuypere, G. Dibelius (Eds.), Proceedings of the 2nd European Conference on Turbomachinery Fluid Dynamics and Thermodynamics, 1997, pp. 99–109.
[8] E. Burman, On nonlinear artificial viscosity discrete maximum principle and hyperbolic conservation laws, BIT 47 (4) (2007) 715–733.
[9] I. Christov, B. Popov, New non-oscillatory central schemes on unstructured triangulations for hyperbolic systems of conservation laws, J. Comput. Phys. 227 (11) (2008) 5736–5757.
[10] B. Cockburn, Devising discontinuous Galerkin methods for non-linear hyperbolic conservation laws, J. Comput. Appl. Math. 128 (1–2) (2001) 187–204 (Numerical analysis 2000, vol. VII, Partial differential equations).
[11] B. Cockburn, P.-A. Gremaud, Error estimates for finite element methods for scalar conservation laws, SIAM J. Numer. Anal. 33 (2) (1996) 522–554.
[12] B. Cockburn, C. Johnson, C. Shu, E. Tadmor, Advanced numerical approximation of nonlinear hyperbolic equations, Lecture Notes in Mathematics, vol. 1697, Springer, 1998.
[13] B. Cockburn, G. Karniadakis, C. Shu, Discontinuous Galerkin methods – theory, computation and applications, Lecture Notes in Computer Science and Engineering, vol. 11, Springer, 2000.
[14] B. Cockburn, C. Shu, The local discontinuous Galerkin method for time-dependent convection–diffusion systems, SIAM J. Numer. Anal. 35 (1998) 2440–2463.
[15] B. Cockburn, C.-W. Shu, The Runge–Kutta discontinuous Galerkin method for conservation laws V. Multidimensional systems, J. Comput. Phys. 141 (2) (1998) 199–224.
[16] A.F. Emery, An evaluation of several differencing methods for inviscid fluid flow problems, J. Comput. Phys. 2 (1968) 306–331.
[17] A. Ern, J.-L. Guermond, Discontinuous Galerkin methods for Friedrichs' systems, I. General theory, SIAM J. Numer. Anal. 44 (2) (2006) 753–778.
[18] S. Gottlieb, C.-W. Shu, E. Tadmor, Strong stability-preserving high-order time discretization methods, SIAM Rev. 43 (1) (2001) 89–112 (electronic).
[19] J. Guermond, B. Popov, Viscous regularization of the euler equations and minimum entropy principle, in preparation.
[20] J.-L. Guermond, On the use of the notion of suitable weak solutions in CFD, Int. J. Nmer. Methods Fluids 57 (2008) 1153–1170.
[21] J.-L. Guermond, R. Pasquetti, Entropy-based nonlinear viscosity for fourier approximations of conservation laws, C.R. Math. Acad. Sci. Paris 346 (2008) 801–806.
[22] J.-L. Guermond, R. Pasquetti, B. Popov, Entropy viscosity method for nonlinear conservation laws, J. Comput. Phys. 230 (2011) 4248–4267.
[23] R. Hartmann, Adaptive discontinuous Galerkin methods with shock-capturing for the compressible Navier–Stokes equations, Internat. J. Numer. Methods Fluids 51 (9-10) (2006) 1131–1156.
[24] T. Hughes, M. Mallet, A new finite element formulation for computational fluid dynamics IV: a discontinuity-capturing operator for multidimensional advective–diffusive systems, Comput. Methods Appl. Mech. Engrg. 58 (1986) 329–336.
[25] G.S. Jiang, C.-W. Shu, On a cell entropy inequality for discontinuous Galerkin methods, Math. Comput. 62 (206) (1994) 531–538.
[26] G.-S. Jiang, E. Tadmor, Nonoscillatory central schemes for multidimensional hyperbolic conservation laws, SIAM J. Sci. Comput. 19 (6) (1998) 1892–1917 (electronic).
[27] C. Johnson, A. Szepessy, On the convergence of a finite element method for a nonlinear hyperbolic conservation law, Math. Comput. 49 (180) (1987) 427–444.
[28] S.N. Kružkov, First order quasilinear equations with several independent variables, Mat. Sb. (N.S.) 81 (123) (1970) 228–255.
[29] O. Ladyženskaja, Modification of the Navier–Stokes equations for large velocity gradients, in: O. Ladyženskaja (Ed.), Seminars in Mathematics, Boundary Value Problems of Mathematical Physics and Related Aspects of Function Theory, Part II, vol. 7, V.A. Stheklov Mathematical Institute, New York, NY, 1970 (Consultant Bureau).
[30] O. Ladyženskaja, New equations for the description of motion of viscous incompressible fluids and solvability in the large of boundary value problems for them, in: O. Ladyženskaja (Ed.), Proceedings of the Stheklov Institute of Mathematics, number 102 (1967), Boundary Value Problems of Mathematical Physics, V, American Mathematical Society (AMS), Providence, RI, 1970.
[31] E. Larsen, J.E. Morel, Advances in discrete-ordinates methodology, in: Y. Azmy, E. Sartori (Eds.), Nuclear Computational Science: A Century in Review, Springer, New York, 2010, pp. 1–82 (Chapter 1).
[32] P.G. LeFloch, J.-G. Liu, Generalized monotone schemes, discrete paths of extrema, and discrete entropy conditions, Math. Comput. 68 (227) (1999) 1025–1055.
[33] P. Lesaint, P.-A. Raviart, On a finite element method for solving the neutron transport equation, in: Mathematical Aspects of Finite Elements in Partial Differential Equations, Publication No. 33. Math. Res. Center, University of Wisconsin-Madison/Academic Press, New York, 1974, pp. 89–123.
[34] J.-L. Lions, J. Peetre, Sur une classe d'espaces d'interpolation, Inst. Hautes Études Sci. Publ. Math. 19 (1964) 5–68.
[35] P.-L. Lions, P.E. Souganidis, Convergence of MUSCL and filtered schemes for scalar conservation laws and Hamilton–Jacobi equations, Numer. Math. 69 (4) (1995) 441–470.
[36] R. Liska, B. Wendroff, Comparison of several difference schemes on 1D and 2D test problems for the Euler equations, SIAM J. Sci. Comput. 25 (3) (2003) 995–1017 (electronic).
[37] E.Y. Panov, Uniqueness of the solution of the Cauchy problem for a first-order quasilinear equation with an admissible strictly convex entropy, Mat. Zametki 55 (5) (1994) 116–129, 15.
[38] P.-O. Persson, J. Peraire, Sub-cell shock capturing for discontinuous Galerkin methods, in: 44th AIAA Aerospace Sciences Meeting, vol. AIAA-2006-112, AIAA, 2006.
[39] B. Perthame, C.-W. Shu, On positivity preserving finite volume schemes for Euler equations, Numer. Math. 73 (1) (1996) 119–130.
[40] B. Popov, O. Trifonov, One-sided stability and convergence of the Nessyahu–Tadmor scheme, Numer. Math. 104 (4) (2006) 539–559.
[41] W. Reed, T. Hill, Triangular mesh methods for the neutron transport equation, Technical Report LA-UR-73-479, Los Alamos Scientific Laboratory, Los Alamos, NM, 1973.
[42] C. Schwab, *p*- and *hp*-Finite Element Methods, Numerical Mathematics and Scientific Computation, Theory and Applications in Solid and Fluid Mechanics, The Clarendon Press/Oxford University Press, New York, 1998.
[43] C.-W. Shu, S. Osher, Efficient implementation of essentially nonoscillatory shock-capturing schemes, J. Comput. Phys. 77 (2) (1988) 439–471.
[44] J. Smagorinsky, General circulation experiments with the primitive equations, part I: the basic experiment, Monthly Wea. Rev. 91 (1963) 99–152.
[45] E. Tadmor, A minimum entropy principle in the gas dynamics equations, Appl. Numer. Math. 2 (3-5) (1986) 211–219.
[46] L. Tartar, An introduction to Sobolev spaces and interpolation spaces, Lecture Notes of the Unione Matematica Italiana, vol. 3, Springer, 2007.
[47] E.F. Toro, Riemann Solvers and Numerical Methods for Fluid Dynamics, second ed., Springer-Verlag, Berlin, 1999.
[48] J.J.W. van der Vegt, H. van der Ven, Space-time discontinuous Galerkin finite element method with dynamic grid motion for inviscid compressible flows: I. General formulation, J. Comput. Phys. 182 (2) (2002) 546–585.
[49] J. von Neumann, R.D. Richtmyer, A method for the numerical calculation of hydrodynamic shocks, J. Appl. Phys. 21 (1950) 232–237.
[50] P. Woodward, P. Colella, The numerical simulation of two-dimensional fluid flow with strong shocks, J. Comput. Phys. 54 (1) (1984) 115–173.
[51] H. Yang, On wavewise entropy inequalities for high-resolution schemes. I. The semidiscrete case, Math. Comput. 65 (213) (1996) 45–67. S1–S13.
[52] H. Yang, On wavewise entropy inequality for high resolution schemes. II. Fully discrete MUSCL schemes with exact evolution in small time, SIAM J. Numer. Anal. 36 (1) (1999) 1–31 (electronic).