AN ENTROPY STABLE OSCILLATION-FREE DISCONTINUOUS GALERKIN METHOD FOR HYPERBOLIC CONSERVATION LAWS

YONG LIU, JIANFANG LU, AND CHI-WANG SHU

Abstract. Entropy inequalities are crucial to the well-posedness of hyperbolic conservation laws, which help to select the physically meaningful one among the infinite many weak solutions. Recently, several high order discontinuous Galerkin (DG) methods satisfying entropy inequalities were proposed, see [7, 5, 8] and the references therein. However, high order numerical methods typically generate spurious oscillations in the presence of shock discontinuities. In this paper, we construct a high order entropy stable oscillation-free DG (ESOFDG) method for hyperbolic conservation laws. With some suitable modification on the high order damping term introduced in [27, 26], we are able to construct an OFDG scheme with dissipative entropy. It is challenging to make the damping term compatible with the current entropy stable DG framework, that is, the damping term should be dissipative for any given entropy function without compromising high order accuracy. The key ingredient is to utilize the convexity of the entropy function and the orthogonality of the projection. Then the proposed method maintains the same properties of conservation, error estimates and entropy dissipation as the original entropy stable DG method. Extensive numerical experiments are presented to validate the theoretical findings and the capability of controlling spurious oscillations of the proposed algorithm.

1. Introduction

Hyperbolic conservation laws have been studied over the centuries in the realm of gas dynamics of continuum physics. The general form of systems of conservation laws is

$$\begin{cases} \dfrac{\partial \mathbf{u}}{\partial t} + \displaystyle\sum_{m=1}^{d} \dfrac{\partial \mathbf{f}_m(\mathbf{u})}{\partial x_m} = \mathbf{0}, \quad (\mathbf{x}, t) \in \mathbb{R}^d \times (0, +\infty), \\ \mathbf{u} = \mathbf{g}, \quad \text{on } \mathbb{R}^d \times \{t = 0\} \end{cases} \tag{1.1}$$

where $\mathbf{u} = [u_1, \dots, u_n]^T$ is a vector of functions denoting the conservative variables, $\mathbf{f}_m = [f_m^1, \dots, f_m^n]^T$ is the vector flux function, and $\boldsymbol{g} : \mathbb{R}^d \to \mathbb{R}^n$ is the given initial condition. It is widely known that shock waves or contact discontinuities might be developed at finite time, regardless of the smoothness of the initial or boundary conditions. Therefore it is reasonable to seek for weak solutions and interpret (1.1) in the sense of distribution.

2010 *Mathematics Subject Classification.* 65M60.

Key words and phrases. Hyperbolic conservation laws; Entropy stability; Summation-by-parts; Discontinuous Galerkin method; Non-oscillatory.

J. Lu's research is partially supported by NSFC grant 11901213.

C.-W. Shu's research is partially supported by NSF grant DMS-2010107 and AFOSR grant FA9550-20-1-0055.

Definition 1.1. *A function* $\mathbf{u} \in L^\infty(\mathbb{R}^d \times (0,+\infty);\mathbb{R}^n)$ *is called a weak solution of* (1.1) *if it satisfies* (1.1) *in the sense of distributions:*

$$\int_0^{+\infty} \int_{\mathbb{R}^d} \Big(\mathbf{u} \cdot \frac{\partial \varphi}{\partial t} + \sum_{m=1}^{d} \mathbf{f}_m(\mathbf{u}) \cdot \frac{\partial \varphi}{\partial x_m}\Big)\, \mathrm{d}\mathbf{x}\mathrm{d}t + \int_{\mathbb{R}^d} \varphi \cdot \mathbf{g}\, \mathrm{d}\mathbf{x}\Big|_{t=0} = 0 \tag{1.2}$$

for all smooth test functions $\varphi : \mathbb{R}^d \times [0,+\infty) \to \mathbb{R}^n$ *with compact support.*

Unfortunately, the weak solution in Definition 1.1 turns out to be inadequate, as such weak solution would not be unique in general. In order to select the "physical relevant" solution among all weak solutions, it requires the weak solution to satisfy certain entropy criterion. To this end, we first introduce the entropy pairs.

Definition 1.2. *A pair of functions* $\big[U(\mathbf{u}), \mathbf{F}(\mathbf{u})\big]$ *with* $U : \mathbb{R}^n \to \mathbb{R}$, $\mathbf{F} = [F_1, \dots, F_d]^T : \mathbb{R}^n \to \mathbb{R}^d$ *is called an entropy pair for* (1.1) *if* $U(\mathbf{u})$ *is convex and* $\{F_m(\mathbf{u})\}_{m=1}^d$ *satisfy*

$$F_m'(\mathbf{u}) = U'(\mathbf{u})\mathbf{f}_m'(\mathbf{u}), \quad m = 1, \dots, d, \tag{1.3}$$

where $U'(\mathbf{u})$ *and* $F_m'(\mathbf{u})$ *are viewed as row vectors and* $\mathbf{f}_m'(\mathbf{u})$ *is the* $n \times n$ *Jacobian matrix.*

With the setup of entropy pairs, we now specify the additional admissibility condition in order to select the physically meaningful weak solution.

Definition 1.3. *A weak solution* $\mathbf{u}$ *of* (1.1) *is an entropy solution if the following inequality holds:*

$$\frac{\partial U(\mathbf{u})}{\partial t} + \sum_{m=1}^{d} \frac{\partial F_m(\mathbf{u})}{\partial x_m} \le 0 \tag{1.4}$$

for all entropy pairs $\big[U(\mathbf{u}), \mathbf{F}(\mathbf{u})\big]$ *in the weak sense, that is,*

$$\int_0^{+\infty} \int_{\mathbb{R}^d} \Big(U(\mathbf{u})\frac{\partial \phi}{\partial t} + \sum_{m=1}^{d} F_m(\mathbf{u})\frac{\partial \phi}{\partial x_m}\Big)\mathrm{d}\mathbf{x}\mathrm{d}t + \int_{\mathbb{R}^d} U(\mathbf{g})\phi(\mathbf{x},0)\, \mathrm{d}\mathbf{x} \ge 0 \tag{1.5}$$

for any $\phi \in C_c^\infty\big(\mathbb{R}^d \times [0,+\infty)\big)$, $\phi \ge 0$.

The existence and uniqueness of the entropy solution of (1.1) can be established for scalar conservation law ($n = 1$), and for one-dimensional systems ($d = 1$) with small initial variation. However, the global existence and uniqueness of the entropy solutions for general hyperbolic conservation laws remain open and a good mathematical understanding of (1.1) is largely unavailable at present. For more details on the theory of hyperbolic conservation laws, we refer the readers to [28, 12] and the references therein.

Despite that uniqueness might not be guaranteed under the entropy conditions given in Definition 1.4, in the numerical approximation of (1.1) one would still like to seek numerical schemes which satisfy the entropy condition on the discrete level. Such a property is referred as *entropy stability*. For the first order (finite volume) method, entropy stability analysis is well-developed based on Tadmor's entropy conservative fluxes and entropy stable fluxes [35, 36]. For the high order entropy stable finite volume methods, a notable result is the *TeCNO* scheme proposed by Fjordholm, Mishra and Tadmor [16], with the use of high order entropy conservative fluxes [29] and the sign property of the essentially non-oscillatory (ENO) reconstruction [17]. In recent years, there have been rapid developments on the entropy stable quadrature-based discontinuous Galerkin (DG) methods. In [7], Chen and Shu proposed an entropy stable DG scheme on unstructured simplex meshes, in which they introduced special Gauss-Lobatto type quadrature rules with collocated surface quadrature points and discrete operators with the multidimensional summation-by-parts (SBP) property [20, 15]. There are a few different entropy stable DG methods in the SBP framework, see e.g. [10, 11, 5, 6]. A comprehensive review of entropy stable DG methods for systems of conservation laws can be found in [8].

For hyperbolic conservation laws, high order linear numerical schemes often generate spurious oscillations near the discontinuities (the Gibbs phenomenon), which may cause accuracy contaminated in the smooth regions and less robustness of the schemes, and even worse blowups of the code. The entropy stable quadrature-based DG methods are not exempt from it either. Generally, there are two kinds of treatments to overcome such a difficulty. One is to apply slope limiters on the DG solutions, such as the total variation diminishing (TVD) limiters, total variation bounded (TVB) limiters [9], and weighted ENO (WENO) limiters [40], etc. The limiters work quite effectively for their simplicity, low computational cost and with little modification required on the original codes, thus favored by many researchers and engineers. Another treatment is to add artificial diffusion in the weak formulations, which is more convenient to perform theoretical analysis. However, it needs a subtle analysis to determine how much diffusion needs to be added [21]. With suitable artificial diffusion, the effect to suppress spurious oscillations is obvious.

The objective of this paper is to design a DG scheme with both entropy stability and oscillation-free properties. It seems quite challenging to obtain these two properties simultaneously. In fact, [7, Remark 4.4] states that it is hard to design entropy stable TVD/TVB limiters for hyperbolic systems. Very recently, we developed an approach to control the spurious oscillations by introducing damping in the DG formulations artificially [27, 26]. As demonstrated in [27], the proposed DG methods can not only control the spurious oscillations, but also preserve some basic properties of the standard DG methods such as conservation, optimal a priori error estimates and superconvergence, etc. In this paper, we are trying to accommodate the damping technique to the entropy stable quadrature-based DG methods. The newly added damping should also be entropy dissipative without comprising high order accuracy. To this end, we make use of the convexity of the entropy functions and take only one part of the local projections in [27]. In fact, we only take the projection orthogonal to the constant states so as to achieve the entropy stability. Compared with the damping terms in [27], the one proposed here abandons the original hierarchical structure and the damping coefficients are also adjusted. By a careful theoretical analysis, the proposed algorithm still maintains the properties of the standard entropy stable quadrature-based DG methods. Finally, we make investigations on a variety of numerical examples to show the good performance of the constructed schemes.

The rest of the paper is organized as follows. In Section 2, we briefly present some preliminary results, including continuous entropy analysis, quadrature rules on simplex elements and the corresponding SBP operators. In Section 3, we first review the matrix-vector form of the nodal DG method, and the entropy stable DG method with quadrature rules of collocated surface nodes, and then we propose the entropy stable oscillation-free DG (ESOFDG) method. Theoretical analysis of accuracy, conservation and entropy stability are also given. In Section 4, we conduct extensive numerical experiments such as accuracy tests, convex and nonconvex conservation laws, and several benchmark problems associated with compressible Euler equations. Concluding remarks are given in Section 5.

2. Preliminaries

In this section, we first present the derivation of the entropy inequality for (1.1) in the PDE level. Then we introduce the quadrature rules and the SBP operators [13, 14, 34], which mimic integration by parts at the discrete level. Next, we give a brief description of the nodal DG method. As we shall see later, the nodal DG method in the matrix-vector form is well suited in the SBP framework.

2.1. Continuous entropy analysis.

As mentioned before, the functions $\big[U(\mathbf{u}), \mathbf{F}(\mathbf{u})\big]$ satisfying $U(\mathbf{u})$ is convex and (1.3) is called an entropy pair. Given a strictly convex entropy function U, let $\mathbf{v} = U'(\mathbf{u})^T$ be the entropy variables. Then $\mathbf{v}'(\mathbf{u}) = U''(\mathbf{u})$ is symmetric positive-definite, and the mapping $\mathbf{u} \to \mathbf{v}$ is invertible. Now let us define the potential fluxes in the following:

$$\psi_m(\mathbf{v}) = \mathbf{v}^T \mathbf{f}_m(\mathbf{u}(\mathbf{v})) - F_m(\mathbf{u}(\mathbf{v})), \quad m = 1, \dots, d. \tag{2.1}$$

One can easily verify that

$$\psi'_m(\mathbf{v}) = \mathbf{f}_m(\mathbf{u}(\mathbf{v}))^T. \tag{2.2}$$

If the solutions to conservation laws (1.1) are smooth, then they should satisfy an additional entropy conservation law as below

$$0 = U'(\mathbf{u})\frac{\partial \mathbf{u}}{\partial t} + \sum_{m=1}^{d} U'(\mathbf{u})\mathbf{f}'_m(\mathbf{u})\frac{\partial \mathbf{u}}{\partial x_m} = \frac{\partial U}{\partial t} + \frac{\partial F_m}{\partial x_m}. \tag{2.3}$$

When the solutions have discontinuities, it is natural to require the entropy to be dissipative. This is how the definition of entropy condition in (1.4) comes from. Integrate (1.4) in space and assume $\mathbf{u}$ is compactly supported, we obtain the following inequality:

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_\Omega U(\mathbf{u})\,\mathrm{d}\mathbf{x} \le 0. \tag{2.4}$$

This means that the total amount of entropy is non-increasing with respect to time. The existence of entropy function is not that trivial to obtain. For scalar conservation laws ($n = 1$), any convex function U can be taken as an entropy function, with the entropy fluxes $F_m(u) = \int U'(u) f'_m(u)\,\mathrm{d}u$. However, for general systems, the existence of entropy function is no longer guaranteed, and both existence and uniqueness of entropy solutions are much more challenging. Fortunately, for most systems we are interested in, such as shallow water equations, compressible Euler equations, magnetohydrodynamic (MHD) equations, we are able to find the entropy functions with physical meaning. For more details readers can refer to [28] on entropy analysis of systems of conservation laws.

2.2. **Quadrature rules and SBP operators.** In this subsection, We briefly review the quadrature rules and SBP operators in order to rewrite the DG method under the SBP framework. To obtain SBP operators, the volume and surface quadrature rules should be carefully constructed. Suppose $\Omega \in \mathbb{R}^d$ is some polygonal computational domain, and $\mathcal{T}_h = \{T_\kappa\}_{\kappa=1}^{N_h}$ is some conforming partition of Ω, with h being the characteristic length of $\mathcal{T}_h$. We assume that each element T_κ is a simplex, so that ∂T_κ consists of $(d-1)$-dimensional simplex faces. We also assume the simplex meshes $\mathcal{T}_h$ are shape regular and quasi-uniform. The set of simplex faces is denoted by

$$\Gamma_h = \{e : e = \partial T_\kappa \cap \partial T_\iota,\ 1 \le \kappa, \iota \le N_h,\ \kappa \ne \iota\} \tag{2.5}$$

Given $T_\kappa \in \mathcal{T}_h$ and $e \in \Gamma_h$ such that $e \in \partial T_\kappa$, we use the notation $\mathbf{n}^{e\kappa} = [n_1^{e\kappa}, \dots, n_d^{e\kappa}]^T$ to represent the unit outward normal vector at e.

The SBP operators of degree k requires at least degree $(2k-1)$ volume quadrature rules and at least degree $2k$ surface quadrature rules. For each simplex T_κ, $1 \le \kappa \le N_h$, suppose that there is an at least degree $(2k-1)$ quadrature rule on T_κ, associated with $\mathcal{N}_{Q,k}$ nodes $\{\mathbf{x}_j^\kappa\}_{j=1}^{\mathcal{N}_{Q,k}}$ and positive weights $\{\omega_j^\kappa\}_{j=1}^{\mathcal{N}_{Q,k}}$, $\mathcal{N}_{Q,k} \ge \mathcal{N}_{P,k}$ with

$$\mathcal{N}_{P,k} = \dim \mathcal{P}^k(\mathbb{R}^d) = \binom{k+d}{d}. \tag{2.6}$$

For each $e \in \Gamma_h$, we also choose some at least degree $2k$ (surface) quadrature rule on e, associated with $\mathcal{N}_{B,k}$ nodes $\{\mathbf{x}_s^e\}_{s=1}^{\mathcal{N}_{B,k}}$, and positive weights $\{\tau_s^e\}_{s=1}^{\mathcal{N}_{B,k}}$. For a scalar function u on Ω, we introduce the vector notation of nodal functions as follows:

$$\overrightarrow{u^\kappa} = [u(\mathbf{x}_1^\kappa), \dots, u(\mathbf{x}_{\mathcal{N}_{Q,k}}^\kappa)]^T, \quad \overrightarrow{u^e} = [u(\mathbf{x}_1^e), \dots, u(\mathbf{x}_{\mathcal{N}_{B,k}}^e)]^T. \tag{2.7}$$

Then we can define the continuous and discrete inner products on T_κ and e that

$$(u,v)_{T_\kappa} = \int_{T_\kappa} uv \,\mathrm{d}\mathbf{x}, \quad (u,v)_{T_\kappa,\omega} = \sum_{j=1}^{\mathcal{N}_{Q,k}} \omega_j^\kappa u(\mathbf{x}_j^\kappa) v(\mathbf{x}_j^\kappa) = \big(\overrightarrow{u^\kappa}\big)^T M_\kappa \overrightarrow{v^\kappa}, \tag{2.8}$$

$$\langle u,v\rangle_e = \int_e uv \,\mathrm{d}S, \quad \langle u,v\rangle_{e,\tau} = \sum_{s=1}^{\mathcal{N}_{B,k}} \tau_s^e u(\mathbf{x}_s^e) v(\mathbf{x}_s^e) = \big(\overrightarrow{u^e}\big)^T B^e \overrightarrow{v^e}, \tag{2.9}$$

where the volume mass matrix M_κ and the surfaces mass matrix B^e are diagonal matrices of quadrature weights:

$$M_\kappa = \mathrm{diag}\{\omega_1^\kappa, \dots, \omega_{\mathcal{N}_{Q,k}}^\kappa\}, \quad B^e = \mathrm{diag}\{\tau_1^e, \dots, \tau_{\mathcal{N}_{B,k}}^e\}. \tag{2.10}$$

Now let $\{\varphi_\ell(\mathbf{x})\}_{\ell=1}^{\mathcal{N}_{P,k}}$ be a set of basis functions of $\mathcal{P}^k(\mathbb{R}^d)$. Then we define the Vandermonde matrices, whose columns are nodal values of $\{\varphi_\ell(\mathbf{x})\}_{\ell=1}^{\mathcal{N}_{P,r}}$:

$$V_r^\kappa = \big[\overrightarrow{\varphi_1^\kappa}, \dots, \overrightarrow{\varphi_{\mathcal{N}_{P,r}}^\kappa}\big], \quad V_r^e = \big[\overrightarrow{\varphi_1^e}, \dots, \overrightarrow{\varphi_{\mathcal{N}_{P,r}}^e}\big], \quad 0 \le r \le k. \tag{2.11}$$

We also define $\mathcal{N}_{P,k} \times \mathcal{N}_{P,k}$ polynomial differential matrices $\hat{D}_m$ such that

$$\frac{\partial \varphi_\ell}{\partial x_m}(\mathbf{x}) = \sum_{r=1}^{\mathcal{N}_{P,k}} \hat{D}_{m,r\ell}\varphi_r(\mathbf{x}), \quad 1 \le m \le d\,.$$

Then $V_k^\kappa \hat{D}_m$ is the Vandermonde matrix of $\{\partial_{x_m}\varphi_\ell(\mathbf{x})\}_{\ell=1}^{\mathcal{N}_{P,k}}$ on T_κ. According to integration by parts and algebraic accuracy of $(\cdot,\cdot)_{T_\kappa,\omega}$ and $\langle\cdot,\cdot\rangle_{e,\tau}$, we can obtain the summation by parts property of *modal* matrices [8] and present it in the following:

$$\hat{M}_\kappa \hat{D}_m + \hat{D}_m^T \hat{M}_\kappa = \sum_{e\in\partial T_\kappa} n_m^{e\kappa} \hat{B}^e, \tag{2.12}$$

where $\hat{M}_\kappa$ and $\hat{B}^e$ are given as

$$\hat{M}_\kappa = (V_k^\kappa)^T M_\kappa V_k^\kappa, \quad \hat{B}^e = (V_k^e)^T B^e V_k^e. \tag{2.13}$$

To obtain *nodal* SBP property, we recall the definitions in [8] of *degree k difference matrix* D_m^κ (of the size $\mathcal{N}_{Q,k} \times \mathcal{N}_{Q,k}$) and extrapolation matrices $\{R^{e\kappa}\}_{e\in\partial T_\kappa}$ (of the size $\mathcal{N}_{B,k} \times \mathcal{N}_{Q,k}$), for which the following two conditions hold:

(i) Exactness: both D_m^κ and $R^{e\kappa}$ are exact for polynomials of degree $\le k$, i.e.

$$D_m^\kappa V_k^\kappa = V_k^\kappa \hat{D}_m, \quad R^{e\kappa} V_k^\kappa = V_k^e\,. \tag{2.14}$$

(ii) Summation-by-parts: let $S_m^\kappa = M^\kappa D_m^\kappa$ and $E^{e\kappa} = (R^{e\kappa})^T B^e R^{e\kappa}$, we have

$$S_m^\kappa + (S_m^\kappa)^T = M_\kappa D_m^\kappa + (D_m^\kappa)^T M_\kappa = \sum_{e\in\partial T_\kappa} n_m^{e\kappa} E^{e\kappa} = \sum_{e\in\partial T_\kappa} n_m^{e\kappa} (R^{e\kappa})^T B^e R^{e\kappa}. \tag{2.15}$$

With the help of the discrete inner product $(\cdot,\cdot)_{T_\kappa,\omega}$, we can also define the L^2 projection matrix [5] :

$$P_r^\kappa = (\hat{M}_\kappa^r)^{-1} (V_r^\kappa)^T M_\kappa, \quad 0 \le r \le k. \tag{2.16}$$

In particular, for $r = 0$, we have

$$\overrightarrow{c}^T M_\kappa (\overrightarrow{u^k} - V_0^\kappa P_0^\kappa \overrightarrow{u^k}) = 0, \quad \forall\, \overrightarrow{u^k} \in \mathbb{R}^{\mathcal{N}_{Q,k}}, \tag{2.17}$$

where $\overrightarrow{c} = [c, c, \ldots, c]^T \in \mathbb{R}^{\mathcal{N}_{Q,k}}$ is a constant vector. The existence of SBP difference matrices is ensured by the following theorem [7, 15, 20, 8]:

Theorem 2.1. *Assume that we have extrapolation matrices $R^{e\kappa}$ satisfying the exactness property. Then the difference matrices, given by the formula*

$$D_m^\kappa = \frac{1}{2}(M_\kappa)^{-1} \sum_{e\in\partial T_\kappa} n_m^{e\kappa} (R^{e\kappa} + V_k^e P_k^\kappa)^T B^e (R^{e\kappa} - V_k^e P_k^\kappa) + V_k^\kappa \hat{D}_m P_k^\kappa \tag{2.18}$$

also satisfy the exactness property and the SBP property.

For the choice of the extrapolation matrices $R^{e\kappa}$, we refer to Remark 3.2 in [8]. Now we also define the extended vector of nodal values to incorporate vector-valued functions $\mathbf{u}$:

$$\overrightarrow{\mathbf{u}^k} = \left[\mathbf{u}(\mathbf{x}_1^\kappa), \ldots, \mathbf{u}(\mathbf{x}_{\mathcal{N}_{Q,k}}^\kappa)\right]^T, \quad \overrightarrow{\mathbf{u}^e} = \left[\mathbf{u}(\mathbf{x}_1^e), \ldots, \mathbf{u}(\mathbf{x}_{\mathcal{N}_{B,k}}^e)\right]^T, \tag{2.19}$$

as well as the Kronecker products

$$\mathbf{M}_\kappa = M_\kappa \otimes I_n, \quad \mathbf{B}^e = B^e \otimes I_n, \quad \mathbf{D}_m^\kappa = D_m^\kappa \otimes I_n, \quad \mathbf{R}^{e\kappa} = R^{e\kappa} \otimes I_n,$$
$$\hat{\mathbf{M}}_\kappa = \hat{M}_\kappa \otimes I_n, \quad \hat{\mathbf{D}}_m = \hat{D}_m \otimes I_n, \quad \mathbf{V}_r^\kappa = V_r^\kappa \otimes I_n, \quad \mathbf{V}_r^e = V_r^e \otimes I_n.$$

We still have the following SBP properties

$$\mathbf{S}_m^\kappa = \mathbf{M}_\kappa \mathbf{D}_m^\kappa, \quad \mathbf{E}^{e\kappa} = (\mathbf{R}^{e\kappa})^T \mathbf{B}^e \mathbf{R}^{e\kappa}, \quad \mathbf{S}_m^\kappa + (\mathbf{S}_m^\kappa)^T = \sum_{e \in \partial T_\kappa} n_m^{e\kappa} \mathbf{E}^{e\kappa}. \tag{2.20}$$

3. High order entropy stable oscillation-free DG schemes

In this section, we proceed to construct the entropy stable OFDG schemes for (1.1). We first review the classic nodal DG methods and quadrature-based entropy stable DG methods with collocated surface nodes [7, 8]. The method successfully achieves entropy stability thanks to the SBP property of the corresponding discrete operators [13, 14, 34] and the *flux differencing* technique with entropy conservative fluxes [3, 7]. These two treatments are extremely important for they can recover the integration by parts and chain rule at the discrete level respectively. Now let us introduce the classic nodal DG method.

3.1. **Nodal DG schemes.** In this subsection, we recall the classic nodal DG schemes. Given polynomial degree $k \geq 0$, we define the DG finite element space:

$$\mathbf{W}_h^k := \{\mathbf{w}_h : \mathbf{w}_h^\kappa = \mathbf{w}_h|_{T_\kappa} \in [\mathcal{P}^k(T_\kappa)]^n, \ 1 \leq \kappa \leq N_h\}. \tag{3.1}$$

We seek $\mathbf{u}_h \in \mathbf{W}_h^k$ such that for each $\mathbf{w}_h \in \mathbf{W}_h^k$ and $1 \leq \kappa \leq N_h$, we have

$$\Big(\frac{\partial \mathbf{u}_h^\kappa}{\partial t}, \mathbf{w}_h^\kappa\Big)_{T_\kappa} - \sum_{m=1}^{d} \Big(\mathbf{f}_m(\mathbf{u}_h^\kappa), \frac{d\mathbf{w}_h^\kappa}{dx_m}\Big)_{T_\kappa} = - \sum_{e \in \partial T_\kappa} \big\langle \hat{\mathbf{f}}_{\mathbf{n}}(\mathbf{u}_h^\kappa, \mathbf{u}_h^{\tilde{\kappa}}), \mathbf{w}_h^\kappa \big\rangle_e, \tag{3.2}$$

where $\hat{\mathbf{f}}_{\mathbf{n}}$ is the interface numerical flux function, and $e = \partial T_\kappa \cap \partial T_{\tilde{\kappa}}$ is the interface of the elements T_κ and $T_{\tilde{\kappa}}$. Since $\{\varphi_\ell(\mathbf{x})\}_{\ell=1}^{\mathcal{N}_{P,k}}$ are basis functions of $\mathcal{P}^k(\mathbb{R}^d)$, we can expand $\mathbf{u}_h^\kappa$ and $\mathbf{w}_h^\kappa$ under $\{\varphi_\ell(\mathbf{x})\}_{\ell=1}^{\mathcal{N}_{P,k}}$:

$$\mathbf{u}_h^\kappa(\mathbf{x}, t) = \sum_{\ell=1}^{\mathcal{N}_{P,k}} \hat{\mathbf{u}}_\ell^\kappa(t) \varphi_\ell(\mathbf{x}), \quad \mathbf{w}_h^\kappa(\mathbf{x}, t) = \sum_{\ell=1}^{\mathcal{N}_{P,k}} \hat{\mathbf{w}}_\ell^\kappa(t) \varphi_\ell(\mathbf{x}).$$

We now replace the continuous inner products by the volume quadrature rule and the surface quadrature rule and rewrite (3.2) as follows:

$$\big(\overrightarrow{\hat{\mathbf{w}}^\kappa}\big)^T \hat{\mathbf{M}}_\kappa \frac{\mathrm{d}\overrightarrow{\hat{\mathbf{u}}^\kappa}}{\mathrm{d}t} - \sum_{m=1}^{d} \big(\mathbf{V}_k^\kappa \hat{\mathbf{D}}_m \overrightarrow{\hat{\mathbf{w}}^\kappa}\big)^T \mathbf{M}_\kappa \overrightarrow{\mathbf{f}_m^\kappa} = - \sum_{e \in \partial T_\kappa} \big(\mathbf{V}_k^e \overrightarrow{\hat{\mathbf{w}}^\kappa}\big)^T \mathbf{B}^e \overrightarrow{\mathbf{f}_{\mathbf{n}}^{e\kappa,*}}, \tag{3.3}$$

where

$$\overrightarrow{\hat{\mathbf{u}}^\kappa} = \big[\hat{\mathbf{u}}_1^\kappa, \dots, \hat{\mathbf{u}}_{\mathcal{N}_{P,k}}^\kappa\big]^T, \quad \overrightarrow{\hat{\mathbf{w}}^\kappa} = \big[\hat{\mathbf{w}}_1^\kappa, \dots, \hat{\mathbf{w}}_{\mathcal{N}_{P,k}}^\kappa\big]^T,$$
$$\overrightarrow{\mathbf{f}_m^\kappa} = \big[\mathbf{f}_m(\mathbf{u}_1^\kappa), \dots, \mathbf{f}_m(\mathbf{u}_{\mathcal{N}_{Q,k}^\kappa})\big]^T, \quad \overrightarrow{\mathbf{f}_{\mathbf{n}}^{e\kappa,*}} = \Big[\hat{\mathbf{f}}_{\mathbf{n}}(\mathbf{u}_1^{e\kappa}, \mathbf{u}_1^{e\tilde{\kappa}}), \dots, \hat{\mathbf{f}}_{\mathbf{n}}(\mathbf{u}_{\mathcal{N}_{B,k}}^{e\kappa}, \mathbf{u}_{\mathcal{N}_{B,k}}^{e\tilde{\kappa}})\Big]^T.$$

Multiplying (3.3) by the matrix $\mathbf{V}_k^\kappa$ from the left, with the relation $\mathbf{V}_k^\kappa(\hat{\mathbf{M}}_\kappa)^{-1} = (\mathbf{M}_\kappa)^{-1}(\mathbf{P}_k^\kappa)^T$ we obtain the nodal formulation:

$$\frac{\mathrm{d}\overrightarrow{\mathbf{u}^\kappa}}{\mathrm{d}t} - (\mathbf{M}_\kappa)^{-1}\sum_{m=1}^{d}(\mathbf{D}_m^\kappa)^T\,\mathbf{M}_\kappa\overrightarrow{\mathbf{f}_m^\kappa} = -(\mathbf{M}_\kappa)^{-1}\sum_{e\in\partial T_\kappa}(\mathbf{R}^{e\kappa})^T\,\mathbf{B}^e\overrightarrow{\mathbf{f}_\mathbf{n}^{e\kappa,*}}, \tag{3.4}$$

by choosing $\mathbf{D}_m^\kappa = \mathbf{V}_k^\kappa\hat{\mathbf{D}}_m\mathbf{P}_k^\kappa$ and $\mathbf{R}^{e\kappa} = \mathbf{V}_k^e\mathbf{P}_k^\kappa$ and $\overrightarrow{\mathbf{u}^\kappa}$ is given in (2.19). According to the SBP property (2.20), we have the equivalent strong nodal DG formulation:

$$\frac{\mathrm{d}\overrightarrow{\mathbf{u}^\kappa}}{\mathrm{d}t} + \sum_{m=1}^{d}\mathbf{D}_m^\kappa\overrightarrow{\mathbf{f}_m^\kappa} = (\mathbf{M}_\kappa)^{-1}\sum_{e\in\partial T_\kappa}\big(\mathbf{R}^{e\kappa}\big)^T\mathbf{B}^e\big(\overrightarrow{\mathbf{f}_\mathbf{n}^{e\kappa}} - \overrightarrow{\mathbf{f}_\mathbf{n}^{e\kappa,*}}\big), \tag{3.5}$$

where $\overrightarrow{\mathbf{f}_m^{e\kappa}}$ and $\overrightarrow{\mathbf{f}_\mathbf{n}^{e\kappa}}$ are the vectors of extrapolated nodal values on each face $e \in \partial T_\kappa$ that

$$\overrightarrow{\mathbf{f}_m^{e\kappa}} = \mathbf{R}^{e\kappa}\overrightarrow{\mathbf{f}_m^\kappa}, \quad \overrightarrow{\mathbf{f}_\mathbf{n}^{e\kappa}} = \sum_{m=1}^{d} n_m^{e\kappa}\overrightarrow{\mathbf{f}_m^{e\kappa}}. \tag{3.6}$$

An interpretation of the link between the modal and nodal DG formulations was given in [8, Remark 3.4].

3.2. **Entropy stable DG schemes with collocated surface nodes.** Now we continue to introduce the entropy stable nodal DG formulation based on (3.4). The nodal DG formulation (3.4) does not satisfy any entropy inequality because the second term on the left hand side does not satisfy the discrete chain rule. To recover the chain rule on the discrete level, we replace the difference term in (3.5) with high order difference operation of entropy conservative fluxes. This technique is termed by flux differencing which is crucial to the entropy balance within an element. A deeper look into the flux differencing technique was given in [8, Appendix A]. The modified DG formulation with collocated surface nodes [7] reads

$$\frac{\mathrm{d}\overrightarrow{\mathbf{u}^\kappa}}{\mathrm{d}t} + 2\sum_{m=1}^{d}\mathbf{D}_m^\kappa \circ \mathbf{F}_{m,S}\big(\overrightarrow{\mathbf{u}^\kappa}, \overrightarrow{\mathbf{u}^\kappa}\big)\overrightarrow{\mathbf{1}^\kappa} = (\mathbf{M}_\kappa)^{-1}\sum_{e\in\partial T_\kappa}(\mathbf{R}^{e\kappa})^T\mathbf{B}^e\big(\overrightarrow{\mathbf{f}_\mathbf{n}^{e\kappa}} - \overrightarrow{\mathbf{f}_\mathbf{n}^{e\kappa,*}}\big) \tag{3.7}$$

where $\circ$ denotes the Hadamard (pointwise) product of vectors and matrices, and $\mathbf{F}_{m,S}(\cdot,\cdot)$ is the matrix of pairwise combinations of entropy conservative fluxes [22, 4, 10, 11]:

$$\mathbf{F}_{m,S}(\overrightarrow{\mathbf{u}_L}, \overrightarrow{\mathbf{u}_R}) = \begin{bmatrix} \mathrm{diag}\big(\mathbf{f}_{m,S}(\mathbf{u}_{L,1}, \mathbf{u}_{R,1})\big) & \cdots & \mathrm{diag}\big(\mathbf{f}_{m,S}(\mathbf{u}_{L,1}, \mathbf{u}_{R,\mathcal{N}_R})\big) \\ \vdots & \ddots & \vdots \\ \mathrm{diag}\big(\mathbf{f}_{m,S}(\mathbf{u}_{L,\mathcal{N}_L}, \mathbf{u}_{R,1})\big) & \cdots & \mathrm{diag}\big(\mathbf{f}_{m,S}(\mathbf{u}_{L,\mathcal{N}_L}, \mathbf{u}_{R,\mathcal{N}_R})\big) \end{bmatrix}, \tag{3.8}$$

for $\overrightarrow{\mathbf{u}_L} \in \mathbb{R}^{n\mathcal{N}_L}$ and $\overrightarrow{\mathbf{u}_R} \in \mathbb{R}^{n\mathcal{N}_R}$. Note that this method requires the collocated surface quadrature nodes ($\{\mathbf{x}_x^e\}_{s=1}^{\mathcal{N}_{B,k}}$ is a subset of $\{\mathbf{x}_j^\kappa\}_{j=1}^{\mathcal{N}_{Q,k}}$ for $e \in \partial T_\kappa$), thus $R^{e\kappa}$ is a simple restriction onto e and diagonal. In [7], the authors proved the nodal DG scheme (3.7) is conservative, entropy stable and high order accurate, under the assumptions that $\mathbf{f}_{m,S}$ is entropy conservative, and $\hat{\mathbf{f}}_\mathbf{n}$ is entropy stable for any given entropy function.

3.3. **Entropy stable oscillation-free DG schemes.** Despite the fact that DG scheme (3.7) possesses so many good properties, it cannot eliminate numerical oscillations near the discontinuities. To control the non-physical oscillations, we adopt the recent developed damping technique in [27, 26] and construct a suitable damping term for (3.7). Note that the added damping term should not destroy the good properties of the entropy stable DG schemes, such as conservation, entropy stability, and high order accuracy. By a careful construction, we obtain the following nodal DG schemes:

$$\begin{aligned}\frac{\mathrm{d}\overrightarrow{\mathbf{u}^{\kappa}}}{\mathrm{d}t} &+ 2\sum_{m=1}^{d} \mathbf{D}_m^{\kappa} \circ \mathbf{F}_{m,S}\big(\overrightarrow{\mathbf{u}^{\kappa}}, \overrightarrow{\mathbf{u}^{\kappa}}\big)\overrightarrow{\mathbf{1}^{\kappa}} \\ &= (\mathbf{M}_{\kappa})^{-1} \sum_{e\in\partial T_{\kappa}} (\mathbf{R}^{e\kappa})^T \mathbf{B}^e \Big(\overrightarrow{\mathbf{f}_{\mathbf{n}}^{e\kappa}} - \overrightarrow{\mathbf{f}_{\mathbf{n}}^{e\kappa,*}}\Big) - \sigma_{\kappa}(\mathbf{u})\big(\overrightarrow{\mathbf{u}^{\kappa}} - \mathbf{V}_0^{\kappa}\mathbf{P}_0^{\kappa}\overrightarrow{\mathbf{u}^{\kappa}}\big).\end{aligned} \tag{3.9}$$

The damping coefficient $\sigma_\kappa(\mathbf{u})$ is defined as follows

$$\sigma_{\kappa}(\mathbf{u}) = \max_{1\le s\le n} \left(\sum_{\ell=0}^{1} \frac{h_{T_\kappa}^{2\ell}}{\ell+1} \sum_{|\alpha|=\ell} \frac{1}{N_e} \sum_{\upsilon\in\partial T_{\kappa}} [\![\big(\mathbf{L}\partial^{\alpha}\mathbf{u}\big)_s|_{\upsilon}]\!]^2 \right)^{\frac{1}{2}}, \tag{3.10}$$

where the vector α is the multiindex of order $|\alpha| = \alpha_1 + \ldots + \alpha_d$, and $\partial^\alpha \omega = \partial_{x_1}^{\alpha_1} \cdots \partial_{x_d}^{\alpha_d}\omega$. The matrix $\mathbf{L}$ comes from the characteristic decomposition such that

$$\sum_{i=1}^{d} n_i \mathbf{f}_i'(\bar{\mathbf{u}}) = \mathbf{L}^{-1}\boldsymbol{\Lambda}\mathbf{L}$$

with $\mathbf{n} = [n_1, \ldots, n_d]^T$ is the unit outward normal and $\bar{\mathbf{u}}$ is some average of $\mathbf{u}$ on the point $\upsilon \in \partial T_\kappa$. For more details, we refer the readers to [26]. Since we only have the values of the entropy variables on nodal points, we first project the $\mathbf{u}$ into the finite element space to obtain the coefficients of polynomials, $\overrightarrow{\hat{\mathbf{u}}^{\kappa}} = \mathbf{P}_k^{\kappa}\overrightarrow{\mathbf{u}^{\kappa}}$, then we can obtain the derivatives of $\mathbf{u}$. For the jump of the function ω on the vertex $\upsilon \in T_\kappa$, we only consider the adjacent neighbors of element T_κ. For example, see Fig. 1, the adjacent neighbors of element K are K_1, K_2, K_3, thus $N_e = 3$. Under the same assumptions of entropy stable nodal DG methods in [7, 8], we have our main theorem as follows:

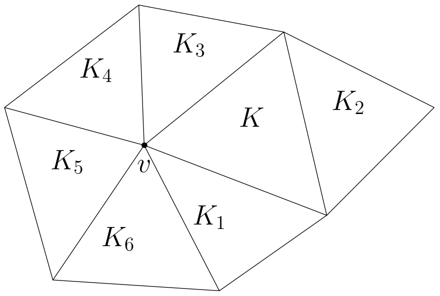


FIGURE 1. Illustrating graph for the jumps in the damping coefficient σ_κ defined in (3.10).

Theorem 3.1. *The scheme* (3.9) *is consistent in the sense that for a smooth solution* $\mathbf{u}$ *of* (1.1) *and a smooth entropy variable* $\mathbf{v}$ *with respect to* U*, the local truncation error is* *of high order:*

$$
\begin{aligned}
&\frac{\mathrm{d}\mathbf{u}_j^\kappa}{\mathrm{d}t} + 2\sum_{m=1}^{d}\sum_{i=1}^{\mathcal{N}_{Q,k}} D^\kappa_{m,ji}\mathbf{f}_{m,S}(\mathbf{u}_j^\kappa, \mathbf{u}_i^\kappa) - \sum_{e\in\partial T_\kappa}\sum_{s=1}^{\mathcal{N}_{B,k}} R^{e\kappa}_{sj}\frac{\mathcal{T}_s^e}{\omega_j^\kappa}\big(\mathbf{f}_{\mathbf{n}}^{e\kappa} - \hat{\mathbf{f}}_{\mathbf{n}}(\mathbf{u}_s^{e\kappa}, \mathbf{u}_s^{e\tilde{\kappa}})\big) \\
&\quad + \sigma_\kappa(\mathbf{u})\Big(\mathbf{u}_j^\kappa - \sum_{i=1}^{\mathcal{N}_{Q,k}}(V_0^\kappa P_0^\kappa)_{ji}\mathbf{u}_i^\kappa\Big) = \mathcal{O}(h^k), \quad j = 1, \ldots, \mathcal{N}_{Q,k}.
\end{aligned} \tag{3.11}
$$

It is also conservative and entropy stable with respect to U in that

$$
\frac{\mathrm{d}}{\mathrm{d}t}\bigg(\sum_{\kappa=1}^{N_h}\Big(\overrightarrow{\mathbf{1}^k}\Big)^T\mathbf{M}_\kappa\overrightarrow{\mathbf{u}^k}\bigg) = 0, \quad \frac{\mathrm{d}}{\mathrm{d}t}\bigg(\sum_{\kappa=1}^{N_h}\Big(\overrightarrow{1^k}\Big)^T M_\kappa\overrightarrow{U^k}\bigg) \le 0\,. \tag{3.12}
$$

Proof. Since $\mathbf{u}$ is a smooth function, we have $\sigma_\kappa^\ell(\mathbf{u}) = 0$, then from the proof in [8, Theorem 4.1], we obtain (3.11). By (2.17), we have

$$
\begin{aligned}
&\sum_{\kappa=1}^{N_h}\Big(\overrightarrow{\mathbf{1}^k}\Big)^T\mathbf{M}_\kappa\sigma_\kappa(\mathbf{u})\big(\overrightarrow{\mathbf{u}^k} - \mathbf{V}_0^\kappa\mathbf{P}_0^\kappa\overrightarrow{\mathbf{u}^k}\big) \\
&\qquad = \sum_{\kappa=1}^{N_h}\sigma_\kappa(\mathbf{u})\big(\overrightarrow{\mathbf{1}^k}\big)^T\mathbf{M}_\kappa\big(\overrightarrow{\mathbf{u}^k} - \mathbf{V}_0^\kappa\mathbf{P}_0^\kappa\overrightarrow{\mathbf{u}^k}\big) = 0\,.
\end{aligned} \tag{3.13}
$$

Note that $\mathbf{V}_0^\kappa = \overrightarrow{\mathbf{1}^k}$, together with the conservative form of (3.7), implies the conservative property of (3.9). For the entropy stable property, we have

$$
\frac{\mathrm{d}}{\mathrm{d}t}\bigg(\sum_{\kappa=1}^{N_h}\Big(\overrightarrow{1^k}\Big)^T M_\kappa\overrightarrow{U^k}\bigg) = \sum_{\kappa=1}^{N_h}(\overrightarrow{\mathbf{v}^k})^T\mathbf{M}_\kappa\frac{\mathrm{d}}{\mathrm{d}t}\overrightarrow{\mathbf{u}^k}. \tag{3.14}
$$

Since $\overrightarrow{U'}\big(\mathbf{V}_0^\kappa\mathbf{P}_0^\kappa\overrightarrow{\mathbf{u}^k}\big)$ is a constant vector for each entropy variable, by (2.17), we have

$$
\begin{aligned}
&-\sum_{\kappa=1}^{N_h}\big(\overrightarrow{\mathbf{v}^k}\big)^T\mathbf{M}_\kappa\sigma_\kappa(\mathbf{u})\big(\overrightarrow{\mathbf{u}^k} - \mathbf{V}_0^\kappa\mathbf{P}_0^\kappa\overrightarrow{\mathbf{u}^k}\big) \\
&= -\sum_{\kappa=1}^{N_h}\sigma_\kappa^\ell(\mathbf{u})\Big(\overrightarrow{U'}\big(\overrightarrow{\mathbf{u}^k}\big) - \overrightarrow{U'}\big(\mathbf{V}_0^\kappa\mathbf{P}_0^\kappa\overrightarrow{\mathbf{u}^k}\big)\Big)^T\mathbf{M}_\kappa\big(\overrightarrow{\mathbf{u}^k} - \mathbf{V}_0^\kappa\mathbf{P}_0^\kappa\overrightarrow{\mathbf{u}^k}\big) \quad (\text{by } (3.13))
\end{aligned} \tag{3.15}
$$

By using the following facts of entropy $U''(\mathbf{u}) \ge 0$, we have

$$
\big(U'(\mathbf{u}_1) - U'(\mathbf{u}_2)\big)\cdot(\mathbf{u}_1 - \mathbf{u}_2) \ge 0 \quad \forall\,\mathbf{u}_1,\,\mathbf{u}_2. \tag{3.16}
$$

Since $\omega_j^\kappa \ge 0$, we obtain

$$
\big(U'(\mathbf{u}(\mathbf{x}_j^\kappa)) - U'((\mathbf{V}_0^\kappa\mathbf{P}_0^\kappa\overrightarrow{\mathbf{u}^k})_j)\big)^T\big(\mathbf{u}(\mathbf{x}_j^\kappa) - (\mathbf{V}_0^\kappa\mathbf{P}_0^\kappa\overrightarrow{\mathbf{u}^k})_j\big)\omega_j^\kappa \ge 0, \quad \forall\, 1 \le j \le \mathcal{N}_{Q,k}. \tag{3.17}
$$

Finally, by above inequality together with the entropy stable property of (3.7) in [7], we obtain (3.12). □

Remark 3.1. *From the proof of Theorem 3.1, we can see that a more natural idea is to construct the damping term with the entropy variables, that is $\sigma_\kappa(\mathbf{u})\big(\overrightarrow{\mathbf{v}^k} - \mathbf{V}_0^\kappa\mathbf{P}_0^\kappa\overrightarrow{\mathbf{v}^k}\big)$. We can still have the conservative and entropy dissipative property, however numerical* *investigations indicate this choice does not control the spurious oscillations well, thus it is not adopted.*

Remark 3.2. *We note that the damping term in* (3.9) *is different from* [26]*. We only used the projection* $\mathbf{P}_0^\kappa$*. Since the* $U'(\mathbf{u})$ *is not a linear function with respect to* $\mathbf{u}$ *for most entropy functions (except the square entropy), the* $U'\big(\mathbf{V}_r^\kappa \mathbf{P}_r^\kappa \overrightarrow{\mathbf{u}^\kappa}\big)$*,* $r \geq 1$*, can not be values of some polynomial at nodes. Thus, we can not use the* L^2 *projection property to obtain the entropy stability. The damping coefficient* (3.10) *is also reconstructed accordingly.*

Remark 3.3. *Since the damping term in* (3.9) *does not take effect on the evolution scheme of the cell average, the positive preserving limiter, which was developed by Zhang and Shu in* [38, 39] *and does not increase entropy* [7, Theorem 3.7]*, can also be applied to ESOFDG scheme* (3.9)*. It is worth noting that we do not use any limiters to demonstrate the robustness of the ESOFDG method in the numerical experiments.*

3.4. **Entropy stable OFDG method on general set of nodes.** Next we introduce three approaches to obtain the entropy stable DG methodology for arbitrary volume and surface quadrature rules. The first one is called the hybridized SBP operators approach in [5, 6] by Chan. The key idea is to combine volume nodes and surface nodes together to obtain the hybridized SBP operators. The second approach is called the global SBP operators approach, which was found by Crean et al. in [10, 11]. The key idea is to view the nodal values on different elements as a whole, grouping them into a single global vector. Then the global SBP operators should be constructed and the entropy dissipation function with respect to U can ensure the entropy stable property. The third approach is to enforce the entropy balance directly [1], in which the method was written in the more general residual distribution framework. A simple linear correction term would be added in the original nodal DG scheme to obtain the entropy stable property. For the general cases, we present these schemes as follows:

$$\frac{\mathrm{d}\overrightarrow{\mathbf{u}^\kappa}}{\mathrm{d}t} = \mathbf{r}^\kappa\big(\overrightarrow{\mathbf{u}^g}\big), \tag{3.18}$$

such that

$$\sum_{\kappa=1}^{N_h} \big(\overrightarrow{\mathbf{1}^\kappa}\big)^T \mathbf{M}_\kappa \mathbf{r}^\kappa\big(\overrightarrow{\mathbf{u}^g}\big) = 0, \quad \sum_{\kappa=1}^{N_h} \big(\overrightarrow{\mathbf{v}^\kappa}\big)^T \mathbf{M}_\kappa \mathbf{r}^\kappa\big(\overrightarrow{\mathbf{u}^g}\big) \leq 0. \tag{3.19}$$

The corresponding entropy stable OFDG scheme is written as

$$\frac{\mathrm{d}\overrightarrow{\mathbf{u}^\kappa}}{\mathrm{d}t} = \mathbf{r}^\kappa\big(\overrightarrow{\mathbf{u}^g}\big) - \sigma_\kappa(\mathbf{u})\big(\overrightarrow{\mathbf{u}^\kappa} - \mathbf{V}_0^\kappa \mathbf{P}_0^\kappa \overrightarrow{\mathbf{u}^\kappa}\big). \tag{3.20}$$

Theorem 3.2. *Under the same assumption in the entropy stable nodal DG scheme* (3.18)*, the scheme* (3.20) *is conservative and entropy stable.*

Proof. The proof is similar to Theorem 3.1 and thus omitted. □

4. Numerical experiments

In this section we show some numerical results to justify the good performance of the proposed algorithm. In one-dimensional problems, we use the three-point Gauss-Lobatto quadrature for $k = 1$, and four-point Gauss-Lobatto quadrature for $k = 2$ and five-point Gauss-Lobatto quadrature for $k = 3$. We also use its tensor-product for two-dimensional problems. In the figures below, for simplicity we only plot the cell averages within each cell instead of showing the full polynomial. We also use the classic fourth order Runge-Kutta method as our time-stepping method. In several numerical examples we plot the total entropy against time by the formulation

$$\text{total entropy} = \sum_{\kappa} \sum_{j=1}^{\mathcal{N}_{Q,k}} \omega_j^{\kappa} U\big(\mathbf{u}_h^{\kappa}(\mathbf{x}_j^{\kappa}, t_n)\big)$$

at time level $t = t_n$. The entropy stable fluxes are chosen as the local Lax-Friedrichs fluxes throughout this paper. Specifically, for one-dimensional compressible Euler equations with $\mathbf{u} = [\rho, \rho u, E]^T$, $\mathbf{f}(\mathbf{u}) = [\rho u, \rho u^2 + p, u(E+p)]^T$, we use the entropy function and entropy variables as

$$U = -\frac{\rho s}{\gamma - 1}, \quad \mathbf{v} = \left[\frac{\gamma - s}{\gamma - 1} - \frac{\rho u^2}{2p},\ \rho u/p,\ -\rho/p\right]^T, \tag{4.1}$$

where $s = \log(p\rho^{-\gamma})$ is the physical specific entropy, and the entropy conservative flux is

$$\mathbf{f}_S(\overrightarrow{\mathbf{u}_L}, \overrightarrow{\mathbf{u}_R}) = \Big[(\bar{\rho})^{\log}\bar{u},\ (\bar{\rho})^{\log}\bar{u}^2 + \tilde{p},\ \bar{u}\big(\tfrac{(\tilde{p})^{\log}}{\gamma-1} + \tilde{\tilde{E}}\big)\Big]^T \tag{4.2}$$

where $\bar{z} = \dfrac{1}{2}(z_L + z_R)$, $(\bar{z})^{\log} = \dfrac{z_R - z_L}{\log z_R - \log z_L}$ and

$$\beta = \frac{\rho}{2p}, \quad \tilde{p} = \frac{\bar{\rho}}{2\bar{\beta}}, \quad (\tilde{p})^{\log} = \frac{(\bar{\rho})^{\log}}{2(\bar{\beta})^{\log}}, \quad \tilde{\tilde{E}} = \frac{1}{2}(\bar{\rho})^{\log}(2\bar{u}^2 - \overline{u^2}) + \tilde{p}.$$

For two-dimensional compressible Euler equations with $\mathbf{u} = [\rho, \rho u, \rho v, E]^T$, $\mathbf{f}_1(\mathbf{u}) = [\rho u, \rho u^2 + p, \rho uv, u(E+p)]^T$, $\mathbf{f}_2(\mathbf{u}) = [\rho v, \rho uv, \rho v^2 + p, v(E+p)]^T$, we use the entropy function and entropy variables as

$$U = -\frac{\rho s}{\gamma - 1}, \quad \mathbf{v} = \left[\frac{\gamma - s}{\gamma - 1} - \frac{\rho(u^2 + v^2)}{2p},\ \rho u/p,\ \rho v/p,\ -\rho/p\right]^T, \tag{4.3}$$

and the entropy conservative fluxes are given as

$$\mathbf{f}_{1,S}(\overrightarrow{\mathbf{u}_L}, \overrightarrow{\mathbf{u}_R}) = \begin{bmatrix} (\bar{\rho})^{\log}\bar{u} \\ (\bar{\rho})^{\log}\bar{u}^2 + \tilde{p} \\ (\bar{\rho})^{\log}\bar{u}\bar{v} \\ \bar{u}\big(\frac{(\tilde{p})^{\log}}{\gamma-1} + \tilde{\tilde{E}}\big) \end{bmatrix}, \quad \mathbf{f}_{2,S}(\overrightarrow{\mathbf{u}_L}, \overrightarrow{\mathbf{u}_R}) = \begin{bmatrix} (\bar{\rho})^{\log}\bar{v} \\ (\bar{\rho})^{\log}\bar{u}\bar{v} \\ (\bar{\rho})^{\log}\bar{v}^2 + \tilde{p} \\ \bar{v}\big(\frac{(\tilde{p})^{\log}}{\gamma-1} + \tilde{\tilde{E}}\big) \end{bmatrix}, \tag{4.4}$$

where the notations are the same as in one-dimensional case except

$$\tilde{\tilde{E}} = \frac{1}{2}(\bar{\rho})^{\log}\big(2(\bar{u}^2 + \bar{v}^2) - \overline{u^2 + v^2}\big) + \widetilde{p}.$$

For more details we refer the readers to [4, 7, 5]. The ratio of the specific heat γ is taken to be 1.4 for air unless specified otherwise.

4.1. One-dimensional problems.

Example 1. *We firstly consider the linear scalar conservation laws that $u_t + u_x = 0$ with periodic boundary condition. Two initial conditions are taken as follows:*

(a) *The smooth case: $u_0(x) = \sin(\pi x)^2 + 1$, $x \in (-1, 1)$.*

(b) *The non-smooth case:*

$$u_0(x) = \begin{cases} \sin(\pi x), & -0.5 \le x \le 0.5, \\ 0, & \text{otherwise.} \end{cases}$$

The computational domain is $(-1, 1)$ and the final time is taken as $T = 1.2$ and $T = 5.0$ for case (a) and (b) respectively. Both of these two cases are using the square entropy function $U(u) = u^2/2$.

From Table 1, we observe the optimal convergence rate for the smooth solution in Example 1. Throughout this paper, we just use the high order numerical quadrature rule to approximate the errors in L^2 norm instead of integrating them exactly. In Fig. 2(a), we can clearly see the spurious oscillation is controlled well and we also plot the discrete entropy against time in Fig. 2(b). Although we have not proved the entropy stability of the fully discrete schemes with the explicit Runge-Kutta time discretization method, we are still able to observe the phenomenon of dissipative entropy.

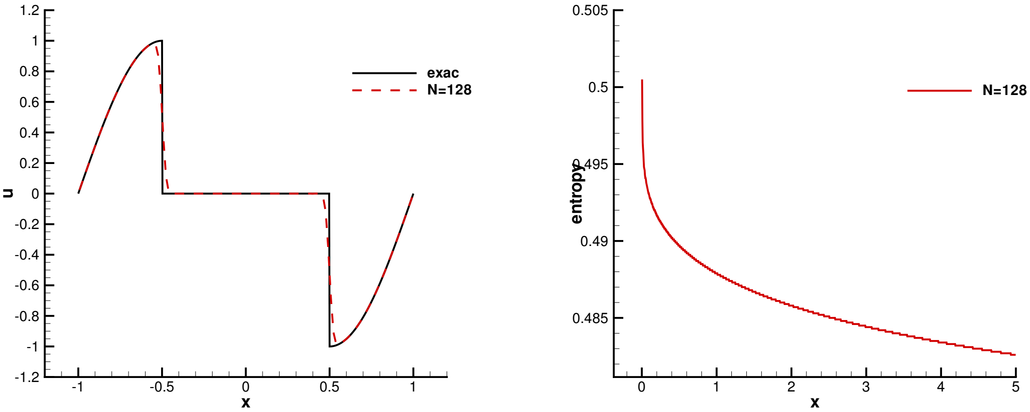
Table 1. Errors and orders of the case (a) in Example 1 with the final time $T = 1.2$.

	$k=1$		$k=2$		$k=3$	
N	$\|u-u_h\|_{L^2}$	order	$\|u-u_h\|_{L^2}$	order	$\|u-u_h\|_{L^2}$	order
16	1.819E-02	–	1.148E-03	–	2.545E-04	–
32	3.277E-03	2.473	8.899E-05	3.690	9.937E-06	4.679
64	6.291E-04	2.381	9.234E-06	3.269	3.605E-07	4.785
128	1.363E-04	2.207	1.081E-06	3.095	1.315E-08	4.777
256	3.238E-05	2.073	1.326E-07	3.027	5.482E-10	4.584
512	7.978E-06	2.021	1.649E-08	3.007	2.799E-11	4.292

Example 2. *In this example, we consider the Burgers' equation that $u_t + \left(\frac{u^2}{2}\right)_x = 0$ with periodic boundary condition. The initial condition is $u_0(x) = 2\sin(x) + 1$, $x \in (0, 2\pi)$. We adopt $U(u) = 0.1e^u + 0.45u^2$ as our entropy function and compute the solution at different final time $T = 0.3$ and $T = 5$. Note that the exact solution $u(x, t)$ stays smooth when $T = 0.3$ so we can test the accuracy of the ESOFDG scheme. When $T = 5$, the exact solution develops an discontinuity and spurious oscillations may occur for classic high order DG schemes.*

For nonlinear scalar equations, we also observe optimal convergence for $k = 1, 2, 3$ in Table 2. When $T = 5$, a shock has appeared due to the nonlinearity, and we observe the ESOFDG scheme captures the shock well without visible oscillations in Fig. 3.

Figure 2. The numerical solution and entropy of the case (b) in Example 1 with final time $T = 5.0$, $k = 2, N = 128$.



(a) Numerical solution.

(b) Entropy against time.

Table 2. Errors and orders in Example 2 with the final time $T = 0.3$.

	$k=1$		$k=2$		$k=3$	
N	$\|u-u_h\|_{L^2}$	order	$\|u-u_h\|_{L^2}$	order	$\|u-u_h\|_{L^2}$	order
16	1.852E-02	–	3.125E-03	–	6.374E-04	–
32	4.631E-03	1.999	3.381E-04	3.028	5.770E-05	3.466
64	1.163E-03	1.993	4.326E-05	3.147	3.899E-06	3.888
128	2.901E-04	2.003	5.258E-06	3.040	2.033E-07	4.261
256	7.244E-05	2.002	6.504E-07	3.015	1.010E-08	4.331
512	1.810E-05	2.001	8.102E-08	3.005	5.488E-10	4.202

Example 3. *In the following we consider the Riemann problem* [24] *for the one-dimensional nonconvex scalar hyperbolic conservation law $u_t + f(u)_x = 0$ with*

$$f(u) = \begin{cases} \dfrac{u(1-u)}{4}, & u < \dfrac{1}{2}, \\ \dfrac{u^2}{2} - \dfrac{u}{2} + \dfrac{3}{16}, & u \geq \dfrac{1}{2}. \end{cases} \tag{4.5}$$

And the initial condition is

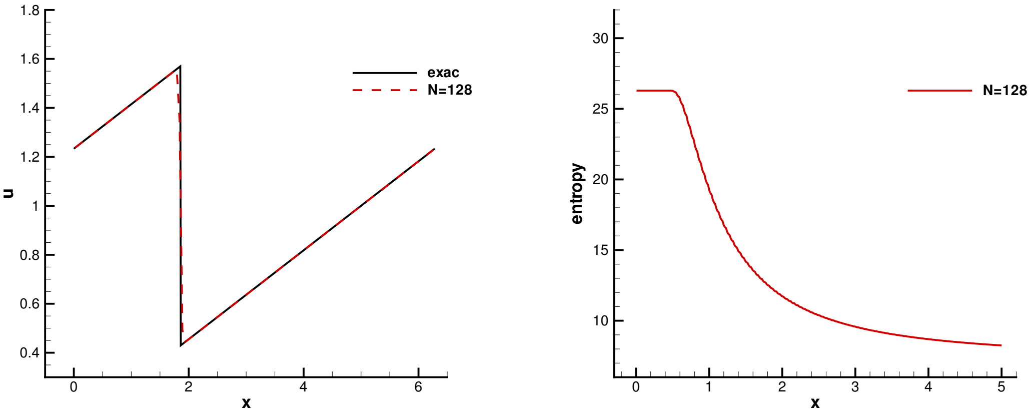
$$u(x,0) = \begin{cases} u_l, & u < \dfrac{1}{4}, \\ u_r, & u \geq \dfrac{1}{4}. \end{cases}$$

We test two cases in the following:

(i) $u_l = 0$, $u_r = 1$, *and the final time is* $T = 1$;

(ii) $u_l = 1$, $u_r = 0$, *and the final time is* $T = 2$.

Figure 3. The numerical solution and entropy in Example 2 with the final time $T = 5$, $k = 2, N = 128$.



(a) Numerical solution.

(b) Entropy against time.

The computational domain is $\Omega = (0, 1)$. We take three kinds of entropy functions as follows:

(a) $U(u) = \frac{1}{2}u^2$;

(b) $U(u) = 0.1e^u + 0.45u^2$;

(c) $U(u) = u \arctan(20u) - \frac{1}{40}\log(1 + 400u^2)$.

Nonconvex hyperbolic conservation laws are very challenging in computation, because if their numerical schemes are not carefully constructed, they may fail to converge to the unique entropy solution or may be too slow to converge that would require impractically fine meshes [24]. In Fig. 4 and 5, we observe the ESOFDG scheme works well for all three kinds of entropy functions. The discrete entropy against time is also plotted for different initial conditions. In Fig. 5, the entropy increases for the reason that the boundary terms do not vanish and we can only obtain entropy stability instead of entropy dissipation that

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\sum_{\kappa=1}^{N_h}\left(\overrightarrow{1^k}\right)^T M_\kappa \overrightarrow{U^k}\right) \leq C$$

where C is some positive constant.

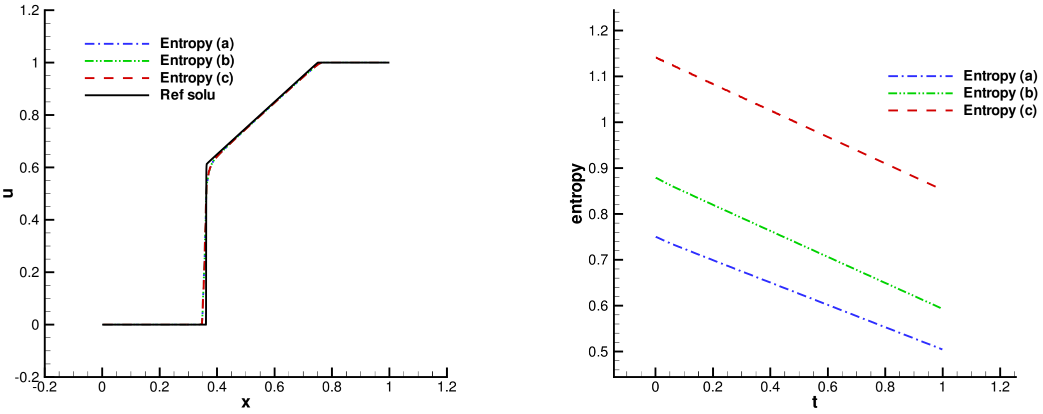
Example 4. *Consider a Riemann problem for one-dimensional Buckley-Leverett equation with the flux function defined as*

$$f(u) = \frac{4u^2}{4u^2 + (1-u)^2}, \tag{4.6}$$

and the initial condition is given as

$$u(x, 0) = \begin{cases} u_l, & x \leq 0, \\ u_r, & x > 0. \end{cases} \tag{4.7}$$

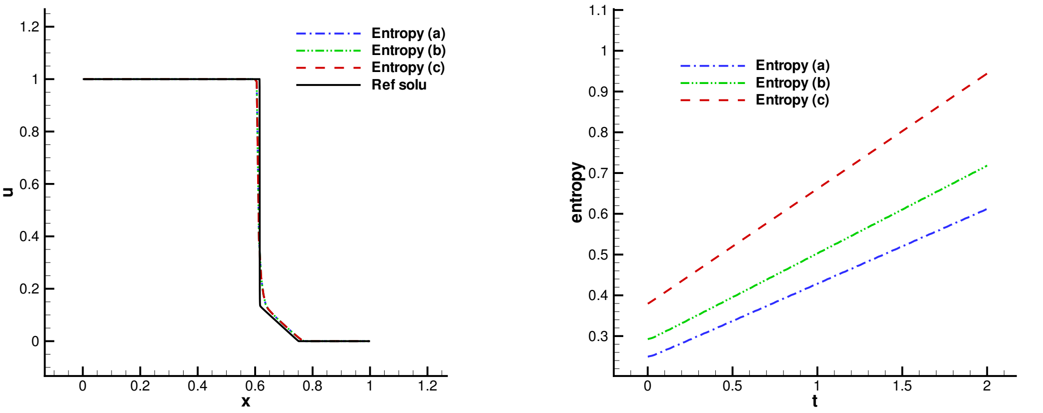
FIGURE 4. The numerical solution and entropy in Example 3 with I.C. (i), $k = 2, N = 128$.



(a) Numerical solution.

(b) Entropy against time.

FIGURE 5. The numerical solution and entropy in Example 3 with I.C. (ii), $k = 2, N = 128$.



(a) Numerical solution.

(b) Entropy against time.

We consider two kinds of initial conditions as follows:

(i) $u_l = 2, \quad u_r = -2$;

(ii) $u_l = -3, \quad u_r = 3$.

We take the computational domain $\Omega = (-4, 4)$, *and the final time* $T = 10$. *We take three kinds of entropy functions as follows:*

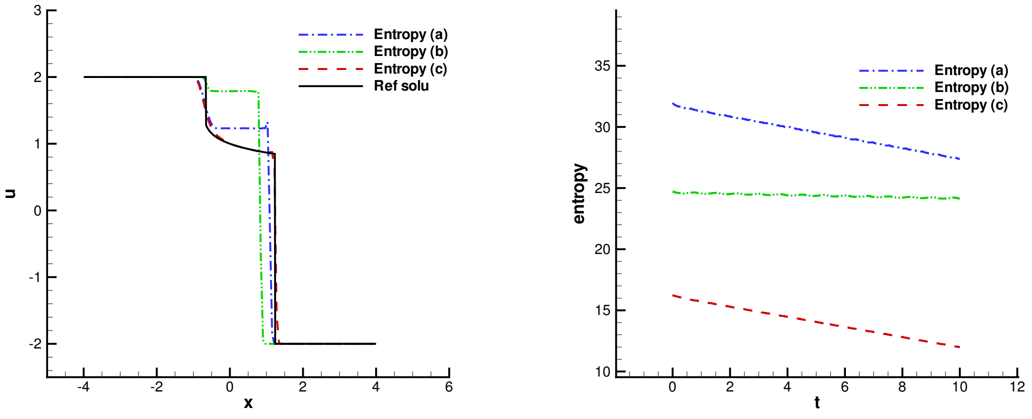
(a) $U(u) = \dfrac{1}{2}u^2$;

(b) $U(u) = u \arctan(20u) - \dfrac{1}{40}\log(1 + 400u^2)$;

(c) $U(u) = (u-1)\arctan(u-1) - \dfrac{1}{2}\log(u^2 - 2u + 2)\,.$

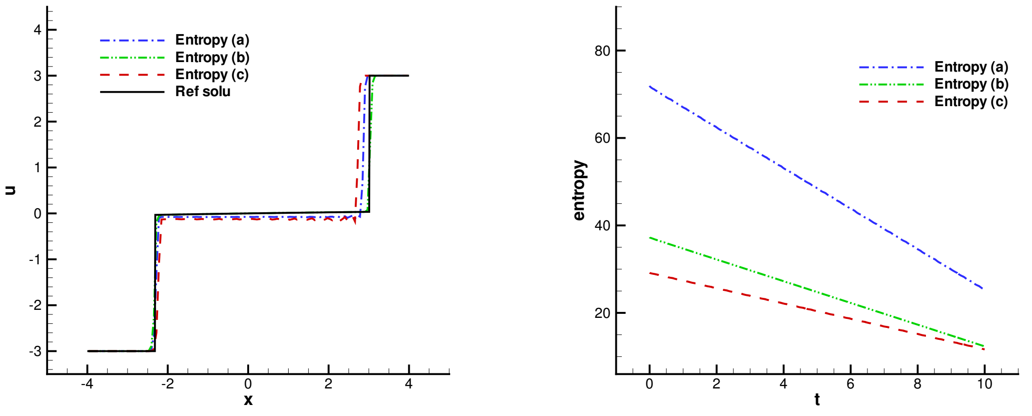
In Fig. 6, we observe that numerical solutions of entropies (a) and (b) do not agree with the entropy solution. But the numerical solution of entropy (c) has satisfactory results thanks to the carefully chosen entropy function. For the initial condition (ii), the numerical solution obtained by using entropy (b) is better than the other two, see Fig. 7. The similar observations are also mentioned in [7].

Figure 6. The numerical solution and entropy in Example 4 with I.C. (i), $k = 2, N = 128$.



(a) Numerical solution. (b) Entropy against time with I.C. (i).

Figure 7. The numerical solution and entropy in Example 4 with I.C. (ii), $k = 2, N = 128$.



(a) Numerical solution. (b) Entropy against time with I.C. (ii).

Example 5. *Now, we consider one-dimensional hyperbolic systems. As an example, we consider two well-known Riemann problems for one-dimensional Euler equations. Both of them have the following Riemann type initial conditions:*

$$\boldsymbol{U}(x,0) = \begin{cases} \boldsymbol{U}_L, & x < 0, \\ \boldsymbol{U}_R, & x > 0. \end{cases}$$

The first test case is Sod's problem [33]*. The initial conditions are*

$$[\rho_L, u_L, p_L]^T = [1, 0, 1]^T, \quad [\rho_R, u_R, p_R]^T = [0.125, 0, 0.1]^T.$$

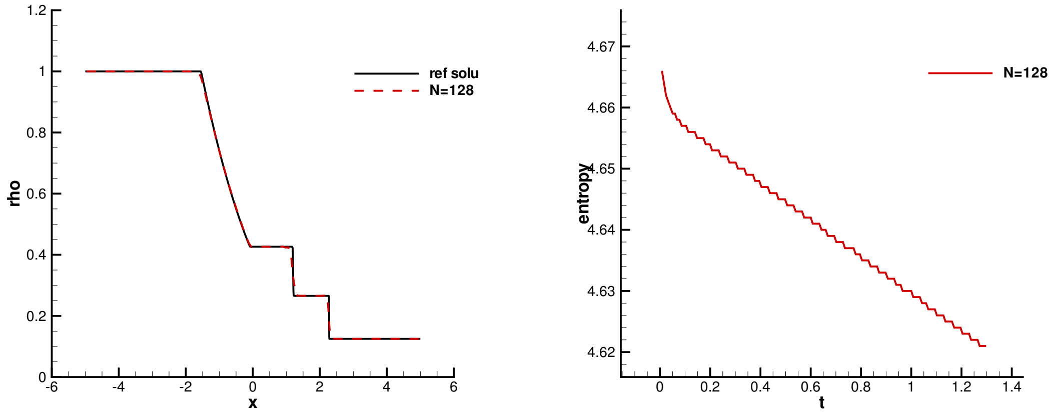
The second one is Lax's problem [25]*. The initial conditions are*

$$[\rho_L, u_L, p_L]^T = [0.445, 0.698, 3.528]^T, \quad [\rho_R, u_R, p_R]^T = [0.5, 0, 0.571]^T.$$

The computational domain $\Omega = (-5, 5)$ and the terminal time $T = 1.3$.

Next, we consider the compressible Euler systems in one dimension. The density of the numerical solutions are shown in Fig. 8 and 9. The numerical solutions of both Sod's problem and Lax' problem have good performances without obvious oscillations.

FIGURE 8. The density profile and entropy for Sod's problem in Example 5 with the final time $T = 1.3$, $k = 2, N = 128$.



(a) Density profile.

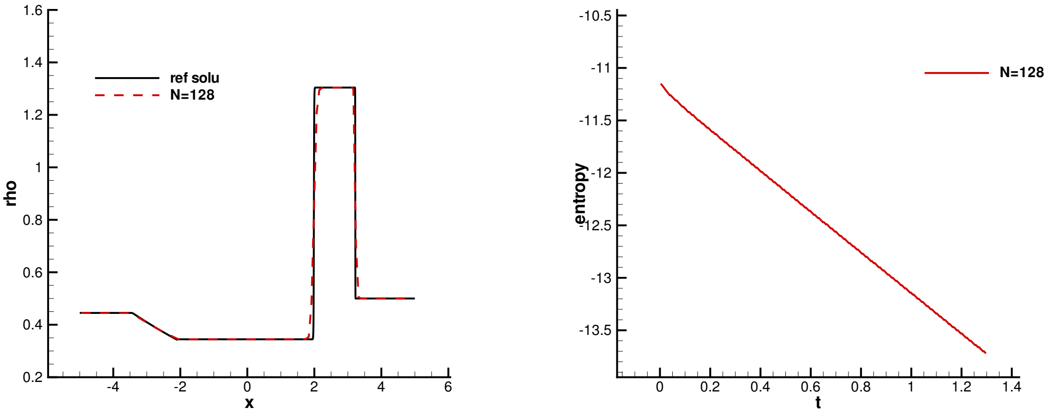
(b) Entropy against time.

Example 6. *Now we consider the Shu-Osher's problem* [32]*. This example describes the interaction between a right-moving Mach = 3 shock and sine waves in density. It is often used to test high order numerical schemes because both shocks and complicated smooth flow structures co-exist. The initial conditions are given as*

$$\begin{aligned} &\rho(x,0) = 3.857143, \quad u(x,0) = 2.629369, \quad p(x,0) = 10.33333, && x < -4, \\ &\rho(x,0) = 1 + 0.2\sin(5x), \quad u(x,0) = 0, \quad p(x,0) = 1, && x > -4. \end{aligned}$$

The computational domain is $\Omega = (-5, 5)$ and the final time is $T = 1.8$.

FIGURE 9. The density profile and entropy for Lax's problem in Example 5 with the final time $T = 1.3$, $k = 2, N = 128$.

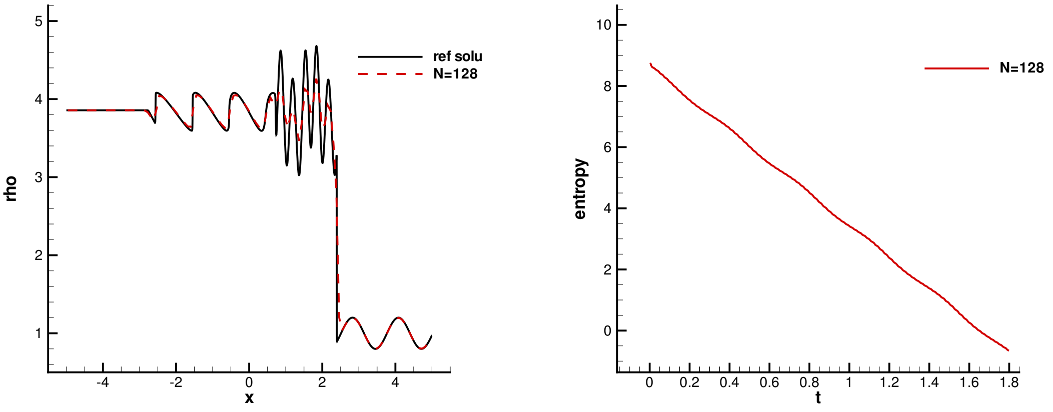


(a) Density profile.

(b) Entropy against time.

The plots of density with 128 cells are displayed in Fig. 10(a) and the discrete entropy is also plotted in Fig. 10(b). We again observe the entropy decreases as time evolves. The ESOFDG scheme has good performances in this example compared with [32] .

FIGURE 10. The density profile and entropy for Shu-Osher's problem in Example 6 with the final time $T = 1.8$, $k = 2, N = 128$.



(a) Density profile.

(b) Entropy against time.

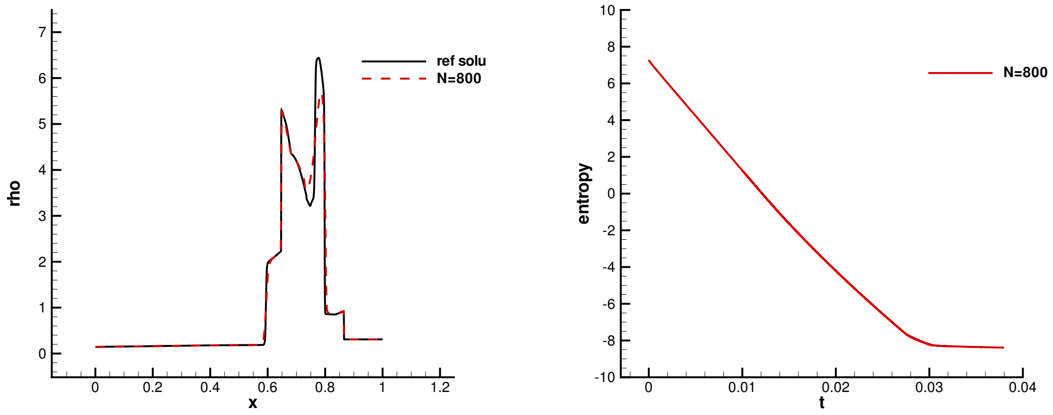
Example 7. *We consider here the interaction of two blast waves* [37]. *This problem involves multiple reflections of shocks and rarefaction waves off the walls. The initial* *conditions are given as*

$$\rho(x,0)=1, \quad u(x,0)=1, \quad p(x,0)=\begin{cases} 10^3, & 0<x<0.1,\\ 10^{-2}, & 0.1<x<0.9,\\ 10^2, & 0.9<x<1. \end{cases}$$

The computational domain is $\Omega=(0,1)$ and the reflective boundary conditions are imposed on both left and right boundaries. The final time is $T=0.038$.

This example easily generates negative density and negative pressure numerically if no oscillation control procedure is used. The classic DG scheme tends to blow up due to the occurrence of the negative density and negative pressure. The ESOFDG scheme, on the other hand, could proceed without using any limiters. Fig. 11 shows the profile of density at $T=0.038$ with 800 cells. All shocks and structures are resolved correctly without obvious oscillations.

FIGURE 11. The density profile and entropy for *two blast waves* problem in Example 7 with the final time $T=0.038$, $k=2, N=800$.



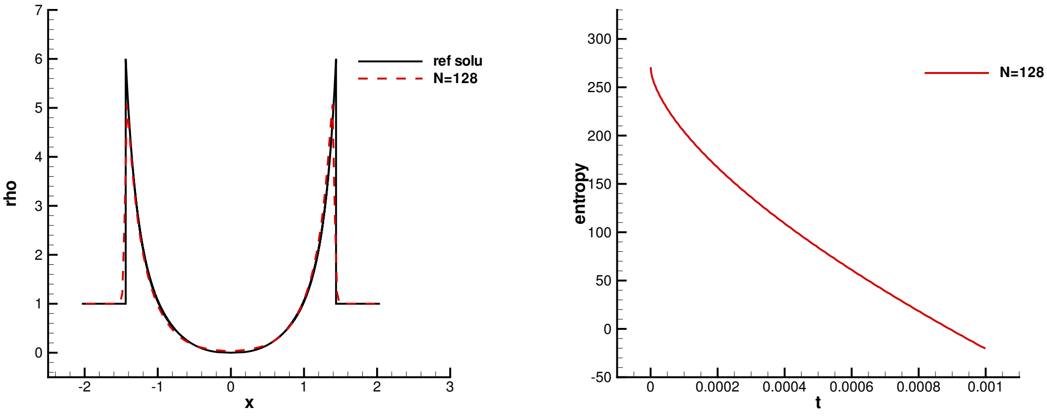
(a) Density profile.

(b) Entropy against time.

Example 8. *We consider the one-dimensional Sedov point blast problem* [39] *which models the expanding wave by an intense explosion in the perfect gas. The authors successfully computed this problem by using both the positivity preserving limiter and TVB limiter in* [39]*. The initial conditions are $\rho(x,0)=1$, $u(x,0)=0$ and $E(x,0)=10^{-12}$ everywhere except $E(x,0)=E_0/h_0$, $E_0=3,200,000$ in the center cell, h_0 is the length of the center cell. The computational domain is $\Omega=(-2,2)$ and the final time is $T=10^{-3}$. The formula of the exact solution can be found in* [31]*.*

In Fig. 12, we show the profile of density with 128 cells at time $T=10^{-3}$ for the one-dimensional Sedov point blast problem in Example 8. The ESOFDG scheme gives satisfactory numerical results and we again observe that the discrete entropy decreases with time during the simulation, which indicates the fully discrete scheme is also entropy stable.

FIGURE 12. The density profile and entropy for *Sedov point blast* problem in Example 8 with the final time $T = 10^{-3}$, $k = 2, N = 128$.



(a) Density profile.

(b) Entropy against time.

4.2. **Two-dimensional problems.**

Example 9. *Consider the two-dimensional linear scalar conservation law*

$$u_t + u_x + u_y = 0, \quad (x, y) \in \Omega$$

with periodic boundary condition. We consider two initial conditions in the following.

(a) *The smooth case:* $u_0(x, y) = \sin(2x)\cos(2y) + 0.5$, $\Omega = (0, \pi) \times (0, \pi)$. *The final time* $T = 1.2$.

(b) *The non-smooth case:*

$$u_0(x, y) = \begin{cases} 1, & r \le \dfrac{1}{8}\left(3 + 3^{\sin 5\theta}\right), \\ 0, & \text{elsewhere,} \end{cases}$$

where (r, θ) *are the polar coordinates. The computational domain* $\Omega = (-1, 1) \times (-1, 1)$. *The final time is* $T = 1.8$.

We use the entropy function $U(u) = 0.1e^u + 0.45u^2$ *for both cases.*

In Table 3, we report the errors and convergence orders of the numerical solution in L^2 norm for $k = 1, 2, 3$ in Example 9. We observe the $(k + 1)$th order of convergence for all k, which is better than the prediction of truncation error analysis. For non-smooth solutions, we observe the ESOFDG scheme sharply captures the interfaces of the shock in Fig. 13.

Example 10. *Consider the following two-dimensional Burgers' equation*

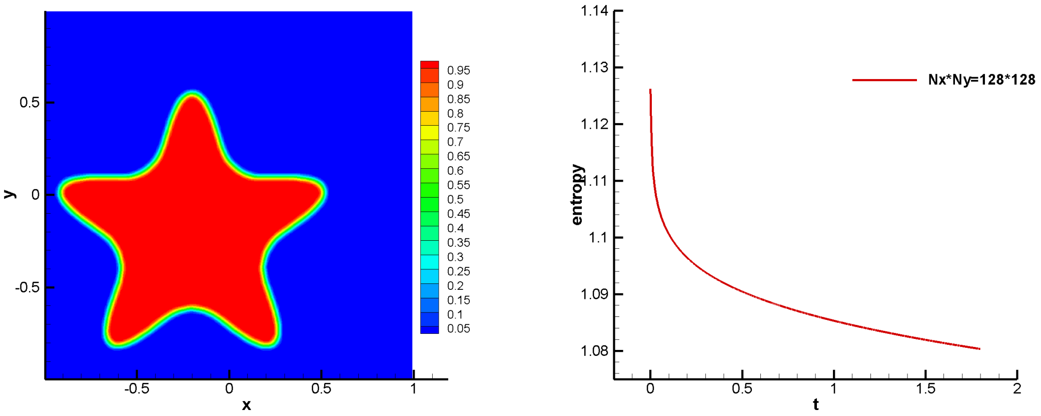
$$u_t + \left(\frac{u^2}{2}\right)_x + \left(\frac{u^2}{2}\right)_y = 0, \quad (x, y) \in \Omega.$$

with the two cases in the following:

TABLE 3. Errors and orders of the case (a) in Example 9 with the final time $T = 1.2$.

	$k = 1$		$k = 2$		$k = 3$	
N	$\|u - u_h\|_{L^2}$	order	$\|u - u_h\|_{L^2}$	order	$\|u - u_h\|_{L^2}$	order
16×16	1.903E-02	–	1.177E-03	–	8.232E-05	–
32×32	4.065E-03	2.227	9.589E-05	3.618	3.106E-06	4.728
64×64	9.671E-04	2.072	9.579E-06	3.323	1.402E-07	4.469
128×128	2.386E-04	2.019	1.108E-06	3.112	7.651E-09	4.196
256×256	5.946E-05	2.005	1.355E-07	3.031	4.589E-10	4.059
512×512	1.485E-05	2.001	1.685E-08	3.008	2.841E-11	4.014

FIGURE 13. The numerical solution and entropy of the case (b) in Example 9 with final time $T = 1.8$, $k = 2$, $N_x \times N_y = 128 \times 128$.



(a) Numerical solution.

(b) Entropy against time.

(a) *The initial condition is* $u_0(x, y) = \sin(\pi(x + y))$ *and periodic boundary condition. The computational domain is* $\Omega = (0, 2) \times (0, 2)$*, and we take the final time that* $T = 0.2$.

(b) *Riemann problem* [18] *The initial condition is*

$$u_0(x, y) = \begin{cases} -\dfrac{1}{5}, & \textit{if } x < \dfrac{1}{2},\ y > \dfrac{1}{2}, \\ -1, & \textit{if } x \geq \dfrac{1}{2},\ y > \dfrac{1}{2}, \\ \dfrac{1}{2}, & \textit{if } x < \dfrac{1}{2},\ y \leq \dfrac{1}{2}, \\ \dfrac{4}{5}, & \textit{if } x \geq \dfrac{1}{2},\ y \leq \dfrac{1}{2}. \end{cases}$$

The exact solution for $t > 0$ *can be found in* [18]*. The computational domain is* $\Omega = (0, 1) \times (0, 1)$ *and the final time is* $T = 0.5$.

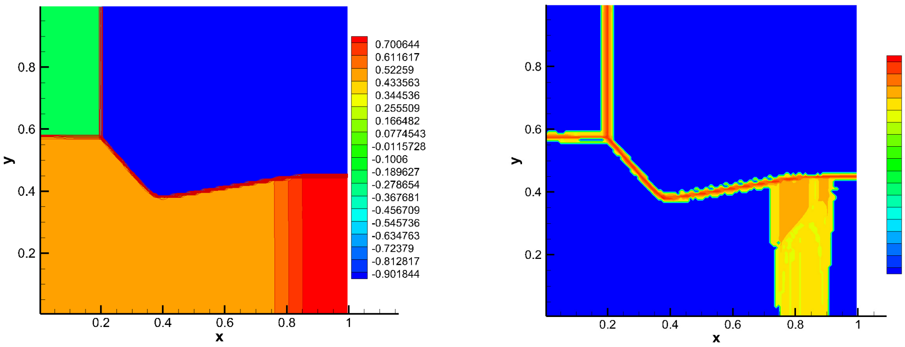
We take $U(u) = \cosh(u)$ *as the entropy function for both cases.*

For the case (a) in Example 10, the errors and orders are shown in Table 4 and the convergence rates are optimal at least for $k = 1, 2$. The numerical solution of case (b) in Example 10 is displayed in the left panel of Fig. 14, and the absolute value error is plotted in the right panel where it uses logarithmic scale. We can see the numerical error is small away from the shock waves.

TABLE 4. Errors and orders of the case (a) in Example 10 with the final time $T = 0.2$.

	$k=1$		$k=2$		$k=3$	
N	$\|u-u_h\|_{L^2}$	order	$\|u-u_h\|_{L^2}$	order	$\|u-u_h\|_{L^2}$	order
16×16	1.462E-02	–	2.319E-03	–	4.225E-04	–
32×32	3.723E-03	1.973	3.161E-04	2.875	3.199E-05	3.723
64×64	9.336E-04	1.996	4.090E-05	2.950	2.568E-06	3.639
128×128	2.341E-04	1.996	5.305E-06	2.947	2.347E-07	3.452
256×256	5.900E-05	1.988	7.063E-07	2.909	2.229E-08	3.397
512×512	1.500E-05	1.975	9.575E-08	2.883	2.092E-09	3.413

FIGURE 14. The numerical solution and entropy of the case (b) in Example 10 with final time $T = 0.5$, $k = 2, N_x \times N_y = 256 \times 256$.



(a) Numerical solution.

(b) Absolute value of error

Example 11. *We now test the Riemann problem proposed in* [30] *which is very challenging to many high-order numerical schemes for the reason that the solution has a two-dimensional composite wave structure. We have the flux functions as*

$$\boldsymbol{f}(u) = [\sin u, \cos u]^T, \tag{4.8}$$

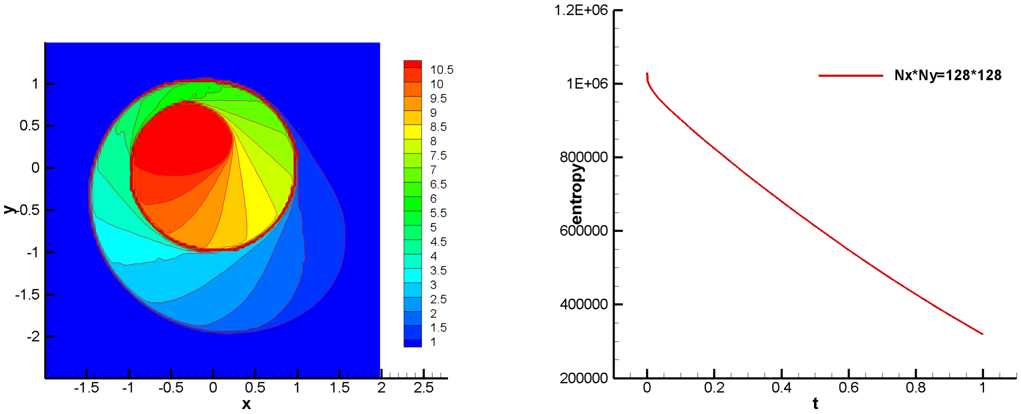
and the initial condition is given as

$$u(x, y, 0) = \begin{cases} \dfrac{7\pi}{2}, & x^2 + y^2 < \dfrac{1}{2}, \\ \dfrac{\pi}{4}, & \text{otherwise}. \end{cases} \tag{4.9}$$

The computational domain is $\Omega = (-2,2) \times (-2.5, 1.5)$*, and the final time is* $T = 1$*. The entropy function is* $U(u) = \cosh(u)$*.*

The numerical result of Example 11 is shown in Fig. 15. With this suitable entropy function, the ESOFDG scheme provides satisfactory results. We again observe that the discrete entropy monotonically decays which indicates the fully discrete scheme is entropy stable.

FIGURE 15. The numerical solution and entropy in Example 11 with final time $T = 1$, $k = 2, N_x \times N_y = 128 \times 128$.



(a) Numerical solution.

(b) Entropy against time.

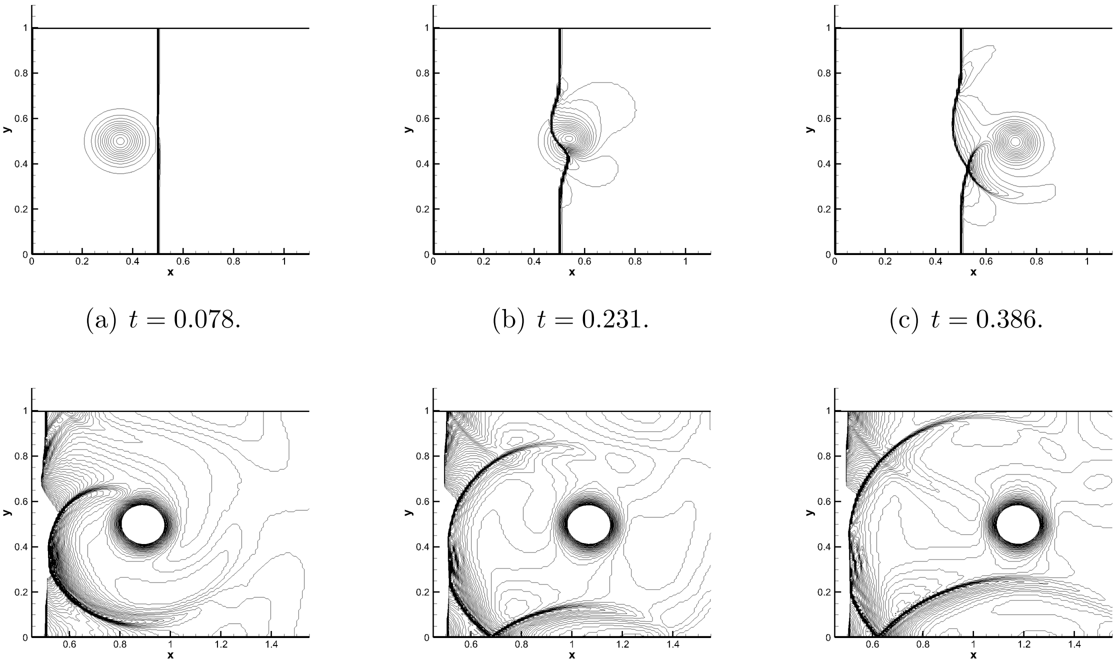
Example 12. *In this example, we consider the shock vortex interactions in two dimensions* [2, 23]*. A stationary Mach 1.1 shock is positioned at* $x = 0.5$*, perpendicular to the* x*-axis. Its left state is* $[\rho, u, v, p]^T = [1, 1.1\sqrt{\gamma}, 0, 1]^T$*. An isentropic vortex perturbation centered at* (x_c, y_c) *is added to the velocity* (u, v)*, temperature* $(T = p/\rho)$ *and entropy* $(S = ln(p/\rho^\gamma))$ *of the flow, given in the following:*

$$[\delta u, \delta v]^T = \frac{\varepsilon}{r_c} e^{\alpha(1-\tau^2)}[-\bar{y}, \bar{x}]^T,\ \delta T = -\frac{(\gamma-1)\varepsilon^2}{4\alpha\gamma} e^{2\alpha(1-\tau^2)},\ \delta S = 0, \tag{4.10}$$

where $[\bar{x}, \bar{y}]^T = [x - x_c, y - y_c]^T$*,* $r = (\bar{x}^2 + \bar{y}^2)^{1/2}$ *and* $\tau = r/r_c$*. The parameters are taken in the same way as in* [23] *that* $[x_c, y_c]^T = [0.25, 0.5]^T$*,* $\varepsilon = 0.3$*,* $r_c = 0.05$ *and* $\alpha = 0.204$*. The computational domain is taken as* $(0, 2) \times (0, 1)$ *and the final time is* $T = 0.8$*. The left and right boundary conditions are inflow and outflow respectively, and reflecting boundary conditions are imposed on the upper and lower boundaries.*

In Fig. 16, we plot the vortex interacting with the stationary shock wave at different time. Since we change the direction of the perturbation of the velocity compared with that in [23], we can see the solution at $t = 0.8$ in which one branch of the shock bifurcations has reached the bottom boundary and has been reflected. The ESOFDG scheme captures the reflection well and the results are comparable to those in [23].

Figure 16. Pressure contours of the two-dimensional shock vortex interaction problem in Example 12, $k=2$, $N_x \times N_y = 256 \times 128$. 30 contours: (a) $t=0.078$; (b) $t=0.231$; (c) $t=0.386$. 90 contours from 1.19 to 1.37: (d) $t=0.544$; (e) $t=0.701$; (f) $t=0.8$.



(a) $t=0.078$.

(b) $t=0.231$.

(c) $t=0.386$.

(d) $t=0.544$.

(e) $t=0.701$.

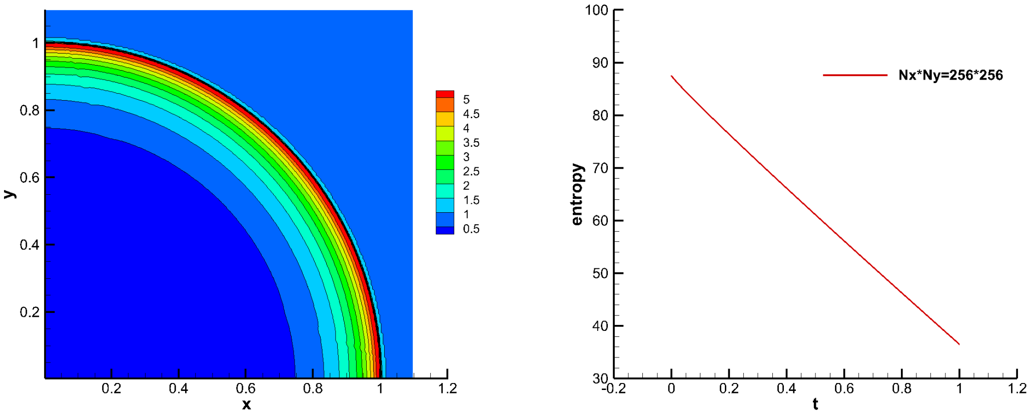
(f) $t=0.8$.

Example 13. *Now let us consider the two-dimensional Sedov point blast problem* [39]. *The initial states are* $\rho(x,y,0)=1$, $u(x,y,0)=v(x,y,0)=0$ *and* $E(x,y,0)=10^{-12}$ *everywhere except* $E(x,0)=E_0/S_0$ *in the lower left corner cell, where* $E_0=0.244816$ *and* S_0 *is the area of the lower left corner cell. The computational domain is* $\Omega=(0,1.1)\times(0,1.1)$ *and the final time is* $T=1$. *The numerical boundary treatment is to extend the DG solutions of* ρ, v, E *as even functions and* u *as an odd function with respect to the left boundary, and extend the DG solutions of* ρ, u, E *as even functions and* v *as an odd function with respect to the bottom boundary. The formula of the exact solution can also be found in* [31].

In Fig. 17, we show the density profiles of the two-dimensional Sedov point blast problem in Example 13. Same as in the one-dimensional problem, a typical low density would appear along with shock discontinuities. We can see the ESOFDG scheme works well without using any limiters.

Example 14. *Now let us consider the double Mach reflection problem* [37]. *Initially, a Mach* 10 *shock attacks the horizontal wall with a* 60° *angle. The reflecting wall lies at*

FIGURE 17. Density contour of the two-dimensional Sedov point blast problem and entropy against time in Example 13 with final time $T = 1$, $k = 2, N_x \times N_y = 256 \times 256$.



(a) Numerical solution.

(b) Entropy against time.

the bottom of domain starting from $x = 1/6$. The initial conditions are given as follows.

$$[\rho, u, v, p]^T = \begin{cases} \left[8, 8.25\cos\left(\frac{\pi}{6}\right), -8.25\sin\left(\frac{\pi}{6}\right), 116.5\right]^T, & x < \frac{1}{6} + \frac{y}{\sqrt{3}}, \\ [1.4, 0, 0, 1]^T, & x > \frac{1}{6} + \frac{y}{\sqrt{3}}. \end{cases}$$

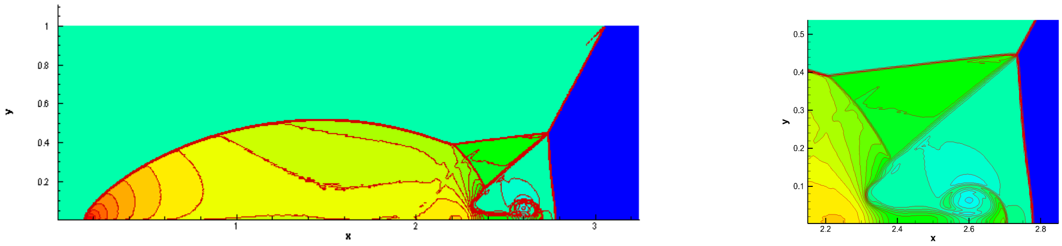
The computational domain is $(0, 4) \times (0, 1)$ and the final time is taken to be 0.2. We have inflow boundary conditions for the left boundary and outflow boundary condition at the right boundary. For the bottom boundary, the exact post-shock condition is imposed for the part from $x = 0$ to $x = 1/6$ and a reflective boundary condition is used for the rest. For the upper boundary, the post-shock condition is imposed for the part from $x = 0$ to $x = 1/6 + (1 + 20t)/\sqrt{3}$ and the pre-shock condition is used for the rest.

In Fig. 18, we plot the density contours of the double Mach reflection problem in Example 14 with $k = 2$ on a grid with $h_x = h_y = 1/256$. We can see the flow structure are resolved very clearly, and no instability occurs for our proposed algorithm.

Example 15. *In the last example, we test the high Mach number astrophysical jets problem* [19, 39]*. The code could easily blow up since the negative pressure and density could easily appear during numerical computation. Conventionally, a positivity preserving limiter was developed to preserve the positivity of the relevant physical quantities in* [38, 39]*. Now we compute the high Mach number astrophysical jets without using any positivity preserving limiter. We consider two cases: $Mach = 80$ and $Mach = 2000$ in the following. Note that the heat capacity ratio $\gamma = 5/3$ in this example.*

(a) *For the Mach 80 problem, the jet initially locates at $y \in (-0.05, 0.05)$, $x = 0$, with the physical values $[\rho, u, v, p]^T = [5, 30, 0, 0.4127]^T$ and the ambient gas is $[\rho, u, v, p]^T =$*

FIGURE 18. Density contours of double Mach reflection at $t = 0.2$ in Example 14, 30 contour lines from 1.731 to 20.92, $k = 2$, $h_x = h_y = 1/256$.



(a) Density profile.

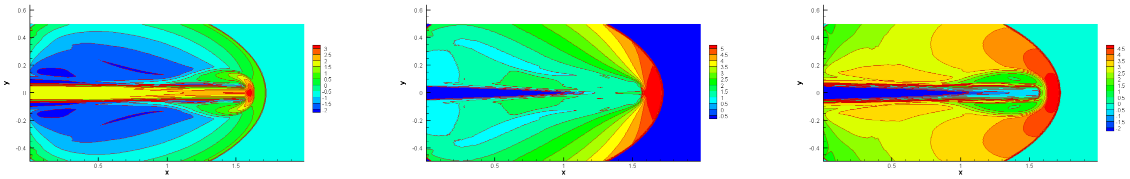
(b) Mach stem in (a).

$[0.5, 0, 0, 0.4127]^T$. *The computational domain is* $(0, 2) \times (-0.5, 0.5)$ *and the terminal time is* 0.07. *The boundary conditions of the rest boundaries are outflow.*

(b) *For the Mach 2000 problem, the jet initially locates at* $y \in (-0.05, 0.05)$, $x = 0$. *The physical values of the jet are* $[\rho, u, v, p]^T = [5, 800, 0, 0.4127]^T$ *and the ambient gas is* $[\rho, u, v, p]^T = [0.5, 0, 0, 0.4127]^T$. *The computational domain is* $(0, 1) \times (-0.25, 0.25)$ *and the terminal time is* 0.001. *The boundary conditions of the rest boundaries are outflow.*

In Fig. 19 and 20, we show the density, pressure and temperature contours of Mach 80 and Mach 2000 astrophysical jets in Example 15, $k = 2$, $N_x \times N_y = 320 \times 160$. The ESOFDG scheme successfully obtain satisfactory results compared to the results in [39] without any occurrence of instability.

FIGURE 19. *High Mach astrophysical jets* in Example 15, Mach $= 80$, $k = 2$, $N_x \times N_y = 320 \times 160$. Scales are logarithmic.



(a) Density contour.

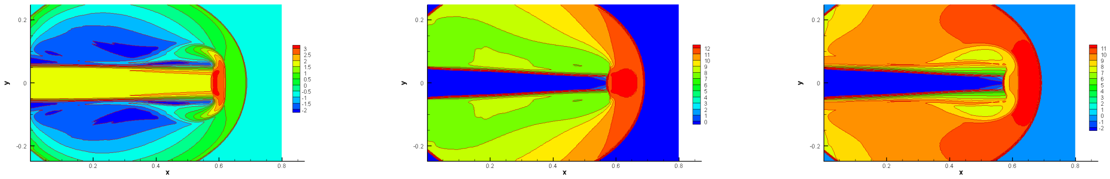
(b) Pressure contour.

(c) Temperature contour.

5. CONCLUDING REMARKS

In this paper, we propose an entropy stable oscillation-free discontinuous Galerkin method for hyperbolic conservation laws. The entropy stable DG method [7, 8] has attracted much attention from the date of its birth. Several key ingredients like summation-by-parts operators, flux differencing technique and entropy conservative fluxes and entropy stable fluxes are incorporated in the nodal DG formulation. This also brings challenges if one wants to apply the damping technique [27, 26] to the current entropy stable DG framework. Thanks to the

FIGURE 20. *High Mach astrophysical jets* in Example 15, Mach = 2000, $k = 2$, $N_x \times N_y = 320 \times 160$. Scales are logarithmic.



(a) Density contour. (b) Pressure contour. (c) Temperature contour.

convexity of the entropy functions and the orthogonality of the projection, we are able to construct a damping term similar to the original ones. This indicates the constructed scheme would preserve several properties such as conservation, entropy stability and high order accuracy of the entropy stable DG method, in the meantime it can also suppress the spurious oscillations as demonstrated in the numerical tests. We are aware that currently there is no theoretical analysis of the oscillation control mechanism with the damping term, and this is a possible investigation direction of our future study.

REFERENCES

[1] R. Abgrall, A general framework to construct schemes satisfying additional conservation relations. Application to entropy conservative and entropy dissipative schemes, *J. Comput. Phys.*, 372 (2018), 640 – 666.

[2] J. Casper, Finite-volume implementation of high-order essentially nonoscillatory schemes in two dimensions, *AIAA Journal*, 30 (1992), 2829 – 2835.

[3] M.H. Carpenter, T.C. Fisher, E.J. Nielsen and S.H. Frankel, Entropy stable spectral collocation schemes for the NavierStokes equations: discontinuous interfaces, *SIAM J. Sci. Comput.*, 36 (2014), B835 – B867.

[4] P. Chandrashekar, Kinetic energy preserving and entropy stable finite volume schemes for compressible Euler and Navier-Stokes equations, *Commun. Comput. Phys.*, 14 (2013), 1252 – 1286.

[5] J. Chan, On discretely entropy conservative and entropy stable discontinuous Galerkin methods, *J. Comput. Phys.*, 362 (2018), 346 – 374.

[6] J. Chan, D.C. Del Rey Fernández and M.H. Carpenter, Efficient entropy stable Gauss collocation methods, *SIAM J. Sci. Comput.*, 41 (2019), A2938 – A2966.

[7] T. Chen and C.-W. Shu, Entropy stable high order discontinuous Galerkin methods with suitable quadrature rules for hyperbolic conservation laws, *J. Comput. Phys.*, 345 (2017), 427 – 461.

[8] T. Chen and C.-W. Shu, Review of entropy stable discontinuous Galerkin methods for systems of conservation laws on unstructured simplex meshes, *CSIAM Trans. Appl. Math.*, 1 (2020), 1 – 52.

[9] B. Cockburn and C.-W. Shu, TVB Runge-Kutta local projection discontinuous Galerkin finite element method for scalar conservation laws II: General framework, *Math. Comp.*, 52 (1989), 411 – 435.

[10] J. Crean, J.E. Hicken, D.C. Del Rey Fernández, D.W. Zingg and M.H. Carpenter, High order, entropy-stable discretizations of the Euler equations for complex geometries, In 23rd AIAA Computational Fluid Dynamics Conference. American Institute of Aeronautics and Astronautics, 2017.

[11] J. Crean, J.E. Hicken, D.C. Del Rey Fernández, D.W. Zingg and M.H. Carpenter, Entropy stable summation-by-parts discretization of the Euler equations on general curved elements, *J. Comput. Phys.*, 356 (2018), 410 – 438.

[12] C.M. Dafermos, Hyperbolic Conservation Laws in Continuum Physics, vol. 325, Springer, 2010.

[13] D.C. Del Rey Fernández, P.D. Boom and D.W. Zingg, A generalized framework for nodal first derivative summation-by-parts operators, *J. Comput. Phys.*, 266 (2014), 214 – 239.

[14] D.C. Del Rey Fernández, J.E. Hicken and D.W. Zingg, Review of summation-by-parts operators with simultaneous approximation terms for the numerical solution of partial differential equations, *Computers & Fluids*, 95 (2014), 171 – 196.

[15] D.C. Del Rey Fernández, J.E. Hicken and D.W. Zingg, Simultaneous approximation terms for multi-dimensional summation-by-parts operators, *J. Sci. Comput.*, 75 (2018), 83 – 110.

[16] U.S. Fjordholm, S. Mishra and E. Tadmor, Arbitrarily high-order accurate entropy stable essentially nonoscillatory schemes for systems of conservation laws, *SIAM J. Numer. Anal.*, 50 (2012), 544 – 573.

[17] U.S. Fjordholm, S. Mishra and E. Tadmor, ENO reconstruction and ENO interpolation are stable, *Found. Comput. Math.*, 13 (2013), 139 – 159.

[18] J.L. Guermond, R. Pasquetti and B. Popov, Entropy viscosity method for nonlinear conservation laws , *J. Comput. Phys.*, 230 (2011), 4248 – 4267.

[19] Y. Ha, C. Gardner, A. Gelb and C.-W. Shu, Numerical simulation of high Mach number astrophysical jets with radiative cooling, *J. Sci. Comput.*, 24 (2005), 597 – 612.

[20] J.E. Hicken, D.C. Del Rey Fernández and D.W. Zingg, Multidimensional summation-by-parts operators: general theory and application to simplex elements, *SIAM J. Sci. Comput.*, 38 (2016), A1935 – A1958.

[21] A. Hiltebrand and S. Mishra, Entropy stable shock capturing space time discontinuous Galerkin schemes for systems of conservation laws, *Numer. Math.*, 126 (2014), 103 – 151.

[22] F. Ismail and P.L. Roe, Affordable, entropy-consistent Euler flux functions II: Entropy production at shocks, *J. Comput. Phys.*, 228 (2009), 5410 – 5436.

[23] G.-S. Jiang and C.-W. Shu, Efficient implementation of weighted ENO schemes, *J. Comput. Phys.*, 126 (1996), 202 – 228.

[24] A. Kurganov, G. Petrova and B. Popov, Adaptive semi-discrete central-upwind schemes for nonconvex hyperbolic conservation laws, *SIAM J. Sci. Comput.*, 29 (2007), 2381 – 2401.

[25] P.D. Lax, Weak solutions of nonlinear hyperbolic equations and their numerical computation, *Comm. Pure Appl. Math.*, 7 (1954), 159 – 193.

[26] Y. Liu, J. Lu and C.-W. Shu, An essentially oscillation-free discontinuous Galerkin method for hyperbolic systems, *SIAM J. Sci. Comput.*, 44 (2022), A230 – A259.

[27] J. Lu, Y. Liu and C.-W. Shu, An oscillation-free discontinuous Galerkin method for scalar hyperbolic conservation laws, *SIAM J. Numer. Anal.*, 59 (2021), 1299 – 1324.

[28] E. Lodlewski and P.-A. Raviart, Hyperbolic Systems of Conservation Laws, Ellipses, 1991.

[29] P.G. Lefloch, J.-M. Mercier and C. Rohde, Fully discrete, entropy conservative schemes of arbitrary order, *SIAM J. Numer. Anal.*, 40 (2002), 1968 – 1992.

[30] A. kurganov, G. Petrova and B. Popov, Adaptive semidiscrete central-upwind schemes for nonconvex hyperbolic conservation laws, *SIAM J. Sci. Comput.*, 29 (2007), 2381 – 2401.

[31] L.I. Sedov, Similarity and Dimensional Methods in Mechanics, Academic Press, New York, 1959.

[32] C.-W. Shu and S. Osher, Efficient implementation of essentially nonoscillatory shock-capturing schemes. II, *J. Comput. Phys.*, 83 (1989), 32 – 78.

[33] G.A. Sod, A survey of several finite difference methods for systems of nonlinear hyperbolic conservation laws, *J. Comput. Phys.*, 27 (1978), 1 – 31.

[34] M. Svärd and J. Nordström, Review of summation-by-parts schemes for initial-boundary-value problems, *J. Sci. Comput.*, 58 (2014), 61 – 89.

[35] E. Tadmor, The numerical viscosity of entropy stable schemes for systems of conservation laws. I., *Math. Comp.*, 49 (1987), 91 – 103.

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

[37] P. Woodward and P. Colella, The numerical simulation of two-dimensional fluid flow with strong shocks, *J. Comput. Phys.*, 54 (1984), 115 – 173.

[38] X. Zhang and C.-W. Shu, On maximum-principle-satisfying high order schemes for scalar conservation laws, *J. Comput. Phys.*, 229 (2010), 3091 – 3120.

[39] X. Zhang and C.-W. Shu, On positivity-preserving high order discontinuous Galerkin schemes for compressible Euler equations on rectangular meshes, *J. Comput. Phys.*, 229 (2010), 8918 – 8934.

[40] X. Zhong and C.-W. Shu, A simple weighted essentially nonoscillatory limiter for Runge-Kutta discontinuous Galerkin methods, *J. Comput. Phys.*, 232 (2013), 397 – 415.

LSEC, Institute of Computational Mathematics, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing 100190, P.R. China.
E-mail address: yongliu@lsec.cc.ac.cn

School of Mathematics, South China University of University, Canton, Guangdong 510641, China.
E-mail address: jflu@scut.edu.cn.

Division of Applied Mathematics, Brown University, Providence, RI 02912, USA
E-mail address: chi-wang_shu@brown.edu