



# Entropy stable high order discontinuous Galerkin methods with suitable quadrature rules for hyperbolic conservation laws<sup>☆</sup>



Tianheng Chen, Chi-Wang Shu<sup>\*</sup>

Division of Applied Mathematics, Brown University, Providence, RI 02912, United States

## ARTICLE INFO

### Article history:

Received 28 January 2017

Received in revised form 7 May 2017

Accepted 13 May 2017

Available online 17 May 2017

### Keywords:

System of conservation laws

Entropy stability

Discontinuous Galerkin method

Summation-by-parts

## ABSTRACT

It is well known that semi-discrete high order discontinuous Galerkin (DG) methods satisfy cell entropy inequalities for the square entropy for both scalar conservation laws (Jiang and Shu (1994) [39]) and symmetric hyperbolic systems (Hou and Liu (2007) [36]), in any space dimension and for any triangulations. However, this property holds only for the square entropy and the integrations in the DG methods must be exact. It is significantly more difficult to design DG methods to satisfy entropy inequalities for a non-square convex entropy, and/or when the integration is approximated by a numerical quadrature. In this paper, we develop a unified framework for designing high order DG methods which will satisfy entropy inequalities for any given single convex entropy, through suitable numerical quadrature which is specific to this given entropy. Our framework applies from one-dimensional scalar cases all the way to multi-dimensional systems of conservation laws. For the one-dimensional case, our numerical quadrature is based on the methodology established in Carpenter et al. (2014) [5] and Gassner (2013) [19]. The main ingredients are summation-by-parts (SBP) operators derived from Legendre Gauss–Lobatto quadrature, the entropy conservative flux within elements, and the entropy stable flux at element interfaces. We then generalize the scheme to two-dimensional triangular meshes by constructing SBP operators on triangles based on a special quadrature rule. A local discontinuous Galerkin (LDG) type treatment is also incorporated to achieve the generalization to convection–diffusion equations. Extensive numerical experiments are performed to validate the accuracy and shock capturing efficacy of these entropy stable DG methods.

© 2017 Elsevier Inc. All rights reserved.

## 1. Introduction

In this paper, we will deal with the numerical approximation of systems of conservation laws in several space dimensions. The general form is

$$\frac{\partial \mathbf{u}}{\partial t} + \sum_{j=1}^d \frac{\partial \mathbf{f}_j(\mathbf{u})}{\partial x_j} = 0, \quad (\mathbf{x}, t) \in \mathbb{R}^d \times [0, \infty) \quad (1.1)$$

<sup>☆</sup> Research supported by ARO grant W911NF-16-1-0103 and NSF grant DMS-1418750.

<sup>\*</sup> Corresponding author.

E-mail addresses: [tianheng\\_chen@brown.edu](mailto:tianheng_chen@brown.edu) (T. Chen), [shu@dam.brown.edu](mailto:shu@dam.brown.edu) (C.-W. Shu).

where  $\mathbf{u} = [u^1, \dots, u^p]^T$  denotes the vector of state variables taking values in a convex set  $\Omega \in \mathbb{R}^p$ , and the functions  $\mathbf{f}_j = [f_j^1, \dots, f_j^p]^T$  are called the flux functions. For each  $1 \leq j \leq d$ , define the Jacobian matrix

$$A_j(\mathbf{u}) = \mathbf{f}'_j(\mathbf{u}) = \left\{ \frac{\partial f_j^i}{\partial u^k}(\mathbf{u}) \right\}_{1 \leq i, k \leq p} \quad (1.2)$$

Then the system (1.1) is called hyperbolic if  $A(\mathbf{u}, \mathbf{n}) = \sum_{j=1}^d n_j A_j(\mathbf{u})$  has  $p$  real eigenvalues and a complete set of eigenvectors for all  $\mathbf{u} \in \Omega$ ,  $\mathbf{n} \in \mathbb{R}^d$ .

It is well known that shock waves or contact discontinuities might develop at finite time even for smooth initial condition. Hence we have to interpret (1.1) in the sense of distribution and search for weak solutions. However, weak solutions are not necessarily unique. In order to select the “physically relevant” solution among all weak solutions, we usually use the following entropy functions as the admissibility criterion.

**Definition 1.1.** Assume that  $\Omega$  is convex. A convex function  $U : \Omega \rightarrow \mathbb{R}$  is called an entropy function for (1.1) if there exist  $d$  functions  $F_j : \Omega \rightarrow \mathbb{R}$ ,  $1 \leq j \leq d$ , called entropy fluxes, such that the following integrability condition holds

$$U'(\mathbf{u})\mathbf{f}'_j(\mathbf{u}) = F'_j(\mathbf{u}), \quad 1 \leq j \leq d \quad (1.3)$$

where  $U'(\mathbf{u})$  and  $F'_j(\mathbf{u})$  are viewed as row vectors.

In smooth regions, we can left-multiply  $U'(\mathbf{u})$  to (1.1) and obtain an extra conservation law for the entropy function

$$\frac{\partial U(\mathbf{u})}{\partial t} + \sum_{j=1}^d \frac{\partial F_j(\mathbf{u})}{\partial x_j} = 0 \quad (1.4)$$

Yet, at shock waves, we require the entropy to dissipate, which leads to the following definition of an entropy solution.

**Definition 1.2.** A weak solution  $\mathbf{u}$  of (1.1) is called an entropy solution if for all entropy functions  $U$ , we have

$$\frac{\partial U(\mathbf{u})}{\partial t} + \sum_{j=1}^d \frac{\partial F_j(\mathbf{u})}{\partial x_j} \leq 0 \quad (1.5)$$

in the sense of distribution.

Formally integrating the entropy condition (1.5) in space, we come up with the bound

$$\frac{d}{dt} \int_{\mathbb{R}^d} U(\mathbf{u}) d\mathbf{x} \leq 0 \quad (1.6)$$

That is, the total entropy is non-increasing with respect to time. If we further assume that  $U$  is uniformly convex, the above bound indeed implies an *a priori*  $L^2$  bound of the entropy solution [32]. For more details on the theory of systems of conservation laws, we refer the readers to [14,22,23] and the references therein.

Entropy conditions play an essential role in the well-posedness of hyperbolic conservation laws. It is natural to seek numerical schemes which satisfy a discrete version of (1.5) (and (1.6) if we impose periodic or compactly supported boundary conditions), i.e., entropy stable schemes. Entropy stability is the nonlinear analogue of the standard  $L^2$  stability discussed in [28], and can be translated to  $L^2$  stability for uniformly convex  $U$ .

Entropy stability analysis is well-developed for first order schemes. In the case of scalar conservation laws ( $p = 1$ ), monotone schemes were shown to be consistent with all entropy conditions and thus convergent to the unique entropy solution [13,31]. The convergence is guaranteed by the total variation diminishing (TVD) property [30] and Lax–Wendroff type argument [43]. Osher [46] established a more general class of schemes, called E-schemes, that preserve all entropy inequalities. Osher and Tadmor [47] also proved that E-schemes are in fact necessary for all entropy inequalities to be valid. As for systems ( $p > 1$ ), the Godunov type schemes introduced in [32] are entropy stable for all entropy functions.

Both monotone schemes and E-schemes are at most first order (spatially) accurate [31,46]. Therefore when designing high-order schemes, one usually expects entropy stability for only a *single* entropy function. In the realm of finite volume methods, Tadmor [56,57] built the framework of entropy conservative fluxes and entropy stable fluxes (for a given entropy function), and Lefloch, Mercier and Rohde [44] provided a procedure to compute high order accurate entropy conservative fluxes. Using these ingredients, along with the sign property of the essentially non-oscillatory (ENO) reconstruction [18], Fjordholm, Mishra and Tadmor [17] presented a version of ENO schemes, called *TeCNO*, that are entropy stable and arbitrarily order accurate. A second order generalization to higher dimensional unstructured meshes is proposed in [49]. Besides, let us

remark that Bouchut, Bourdaris and Perthame [4] gave a second order accurate scheme that satisfies all entropy inequalities. It does not contradict the argument by Osher and Tadmor since their scheme was not written in the standard finite volume form.

Another popular category of high order numerical schemes is the discontinuous Galerkin (DG) method developed in [10, 9,8,12]. Jiang and Shu [39] proved that the semi-discrete DG schemes satisfy a discrete entropy inequality for the square entropy for scalar conservation laws, in arbitrary dimension and on arbitrary triangulations, which is extended to symmetric systems by Hou and Liu [36]. However, these results are limited to the square entropy function ( $L^2$  norm) only, as the test functions must live in the space of numerical solutions and then  $U'(\mathbf{u})$  has to be linear. For systems whose Jacobian matrices are not symmetric, since the square function is not an entropy function, there is no entropy stability for the (unmodulated) DG methods. Moreover, all integrals in the DG formulation are assumed to be evaluated exactly for the proof of this entropy condition, which can be costly or even impossible to implement (e.g. in the case of Euler equations when the flux functions are rational functions of  $\mathbf{u}$ ). In practice one often uses quadrature rules and stability might be affected. An alternative approach, initiated by Hughes, Franca and Mallet [37], approximates the entropy variables  $\mathbf{v} = U'(\mathbf{u})^T$  in the discrete space. Then entropy stability is achieved for any given entropy function. The drawback of this approach is that it requires nonlinear solvers at each time step, even for explicit time discretization. Hence space-time DG formulation is often preferred [1,35]. In addition, this approach still assumes exact integration for the proof of entropy stability.

In recent years, there have been some developments on entropy stable DG type schemes directly built upon numerical integration. DG schemes can be recast into the nodal formulation after quadrature [41,33]. By choosing Gauss–Lobatto quadrature points, the resulting discrete operators satisfy the summation-by-parts (SBP) property [19]. Thanks to the SBP property, the nodal DG scheme can be adjusted to fulfill an arbitrary entropy condition, while conservation and high order accuracy are maintained. This adjustment is related to the splitting technique for the Burgers equation [16,19] and shallow water equations [20], but not equivalent to any kind of splitting for the Euler equations [15,5].

The main objective of this paper is to construct a unified framework of entropy stable high order nodal DG schemes. We start with an one-dimensional methodology, in which the entropy is conserved within elements, but dissipates at element interfaces. To be more precise, the single element discretization is based on entropy conservative fluxes, and the weak coupling between neighboring elements relies on entropy stable fluxes. Just like classical DG methods, we can apply a TVD/TVB limiter and/or a bound-preserving limiter to control oscillations and enhance robustness without violating the entropy condition. Next we will move to the extension to two-dimensional triangular meshes. The main difficulty is to find high order SBP operators on triangles. Inspired by [34], we will deduce the formulation of SBP operators by introducing a special quadrature rule. Even though we generally assume periodic or compactly supported boundary condition, we will prove that the standard reflecting technique is entropy stable at the wall boundary for the Euler equations. Finally, we will consider convection–diffusion equations, for which a nodal version of the local discontinuous Galerkin (LDG) method [11,6] will be included to handle the diffusive term while entropy stability still holds.

The rest of this paper is organized as follows. Section 2 is a brief tutorial on entropy analysis that is necessary for the subsequent sections. Section 3 presents the one-dimensional entropy stable nodal DG schemes as a reinterpretation of the methods in [5,19,20]. Compatibility with different limiters is also discussed. Section 4 is the major contribution of this paper, which provides SBP operators on triangles and entropy stable nodal DG schemes on triangular meshes. Entropy stability of wall boundary conditions will be proved. Section 5 explains an LDG type approach to convection–diffusion equations. Numerical experiments including smooth accuracy tests and discontinuous tests are reported in section 6. Concluding remarks are given in section 7. A few technical details are provided in the appendix.

## 2. More on entropy analysis

### 2.1. Symmetrization

We continue the entropy analysis in section 1. Define the entropy variables  $\mathbf{v} = U'(\mathbf{u})^T$ . If we assume that  $U$  is strictly convex, the mapping  $\mathbf{u} \mapsto \mathbf{v}$  is one-to-one and can be regarded as a change of variables. Setting  $\mathbf{g}_j(\mathbf{v}) = \mathbf{f}_j(\mathbf{u}(\mathbf{v}))$ , we rewrite the system (1.1) according to the entropy variables

$$\mathbf{u}'(\mathbf{v}) \frac{\partial \mathbf{v}}{\partial t} + \sum_{j=1}^d \mathbf{g}'_j(\mathbf{v}) \frac{\partial \mathbf{v}}{\partial x_j} = 0 \quad (2.1)$$

By strict convexity,  $\mathbf{u}'(\mathbf{v}) = (U''(\mathbf{u}))^{-1}$  is symmetric positive-definite. The following theorem tells us the symmetry of  $\mathbf{g}'_j(\mathbf{v})$  is equivalent to the existence of entropy function [24,45]. Proof can be found in [23].

**Theorem 2.1.** *A strictly convex function  $U$  serves as an entropy function if and only if  $\mathbf{u}'(\mathbf{v})$  is symmetric positive-definite and  $\mathbf{g}'_j(\mathbf{v})$  is symmetric for each  $1 \leq j \leq d$ . (2.1) is called the symmetrization of (1.1). Moreover,  $A(\mathbf{u}, \mathbf{n}) = \sum_{j=1}^d n_j \mathbf{f}_j(\mathbf{u}) = \sum_{j=1}^d n_j \mathbf{g}'_j(\mathbf{v}) \mathbf{v}'(\mathbf{u})$  is similar to*

$$\mathbf{v}'(\mathbf{u})^{\frac{1}{2}} \left( \sum_{j=1}^d n_j \mathbf{g}'_j(\mathbf{u}) \right) \mathbf{v}'(\mathbf{u})^{\frac{1}{2}},$$

which is another symmetric matrix. Hence existence of entropy function implies that (1.1) is hyperbolic.

Now since  $\mathbf{u}'(\mathbf{v})$  and  $\mathbf{g}'_j(\mathbf{v})$  are both symmetric, there exist functions  $\phi(\mathbf{v})$  and  $\psi_j(\mathbf{v})$ , called potential function and potential fluxes, such that

$$\phi'(\mathbf{v}) = \mathbf{u}(\mathbf{v})^T, \quad \psi'_j(\mathbf{v}) = \mathbf{g}_j(\mathbf{v})^T, \quad 1 \leq j \leq d \quad (2.2)$$

It is easy to verify that

$$\phi(\mathbf{v}) = \mathbf{u}(\mathbf{v})^T \mathbf{v} - U(\mathbf{u}(\mathbf{v})), \quad \psi_j(\mathbf{v}) = \mathbf{g}_j(\mathbf{v})^T \mathbf{v} - F_j(\mathbf{u}(\mathbf{v})) \quad (2.3)$$

Entropy conditions follow from the vanishing viscosity approach. Consider the following viscous perturbation of the system (1.1)

$$\frac{\partial \mathbf{u}_\varepsilon}{\partial t} + \sum_{j=1}^d \frac{\partial \mathbf{f}_j(\mathbf{u}_\varepsilon)}{\partial x_j} = \varepsilon \Delta \mathbf{u}_\varepsilon, \quad \varepsilon > 0 \quad (2.4)$$

By left-multiplying  $U'(\mathbf{u}_\varepsilon)$  to (2.4) and integrating by parts (formally),

$$\frac{\partial U(\mathbf{u}_\varepsilon)}{\partial t} + \sum_{j=1}^d \frac{\partial F_j(\mathbf{u}_\varepsilon)}{\partial x_j} = -\varepsilon \sum_{j=1}^d \frac{\partial \mathbf{u}_\varepsilon^T}{\partial x_j} U''(\mathbf{u}_\varepsilon) \frac{\partial \mathbf{u}_\varepsilon}{\partial x_j} \leq 0$$

Sending  $\varepsilon \rightarrow 0^+$  we recover the entropy condition (1.5). For some physical problems (e.g. compressible Navier–Stokes equations), it is necessary to look at the more general form of viscous perturbation

$$\frac{\partial \mathbf{u}_\varepsilon}{\partial t} + \sum_{j=1}^d \frac{\partial \mathbf{f}_j(\mathbf{u}_\varepsilon)}{\partial x_j} = \varepsilon \sum_{j,l=1}^d \frac{\partial}{\partial x_j} (C_{jl}(\mathbf{u}_\varepsilon) \frac{\partial \mathbf{u}_\varepsilon}{\partial x_l}) \quad (2.5)$$

where  $C_{jl}(\mathbf{u}_\varepsilon)$  are  $p \times p$  matrices. Let  $\mathbf{v}_\varepsilon = \mathbf{v}(\mathbf{u}_\varepsilon)$  and  $\widehat{C}_{jl}(\mathbf{v}_\varepsilon) = C_{jl}(\mathbf{u}_\varepsilon) \mathbf{u}'(\mathbf{v}_\varepsilon)$ . Then

$$\frac{\partial \mathbf{u}_\varepsilon}{\partial t} + \sum_{j=1}^d \frac{\partial \mathbf{f}_j(\mathbf{u}_\varepsilon)}{\partial x_j} = \varepsilon \sum_{j,l=1}^d \frac{\partial}{\partial x_j} (\widehat{C}_{jl}(\mathbf{v}_\varepsilon) \frac{\partial \mathbf{v}_\varepsilon}{\partial x_l}) \quad (2.6)$$

Left-multiplying  $U'(\mathbf{u}_\varepsilon) = \mathbf{v}_\varepsilon^T$  to (2.6) gives us

$$\frac{\partial U(\mathbf{u}_\varepsilon)}{\partial t} + \sum_{j=1}^d \frac{\partial F_j(\mathbf{u}_\varepsilon)}{\partial x_j} = -\varepsilon \sum_{j=1}^d \frac{\partial \mathbf{v}_\varepsilon^T}{\partial x_j} \widehat{C}_{jl}(\mathbf{v}_\varepsilon) \frac{\partial \mathbf{v}_\varepsilon}{\partial x_j}$$

In order to make the right hand side non-positive, we have to assume the following admissibility condition

$$\begin{bmatrix} \widehat{C}_{11}(\mathbf{v}_\varepsilon) & \cdots & \widehat{C}_{1d}(\mathbf{v}_\varepsilon) \\ \vdots & & \vdots \\ \widehat{C}_{d1}(\mathbf{v}_\varepsilon) & \cdots & \widehat{C}_{dd}(\mathbf{v}_\varepsilon) \end{bmatrix} \text{ is symmetric semi-positive-definite} \quad (2.7)$$

Therefore, the change of variables  $\mathbf{u} \mapsto \mathbf{v}$  should symmetrize the viscous term simultaneously.

## 2.2. Examples

Here we present some examples of hyperbolic conservation laws and the corresponding entropy function–entropy flux pairs and potential function–potential flux pairs. For simplicity we only focus on one-dimensional systems.

**Example 2.2.1.** The linear symmetric system is of the form

$$\frac{\partial \mathbf{u}}{\partial t} + \frac{\partial (A\mathbf{u})}{\partial x} = 0 \quad (2.8)$$

where  $A$  is a constant symmetric matrix. The standard energy  $U = \frac{1}{2} \mathbf{u}^T \mathbf{u}$  serves as an entropy function. Then  $\mathbf{v} = \mathbf{u}$  and



$$F = \frac{1}{2} \mathbf{u}^T A \mathbf{u}, \quad \phi = \frac{1}{2} \mathbf{u}^T \mathbf{u}, \quad \psi = \frac{1}{2} \mathbf{u}^T A \mathbf{u} \quad (2.9)$$

**Example 2.2.2.** The shallow water equations model water flows with a free surface under the influence of gravity. The governing equations (with flat bottom) are

$$\frac{\partial}{\partial t} \begin{bmatrix} h \\ hw \end{bmatrix} + \frac{\partial}{\partial x} \begin{bmatrix} hw \\ hw^2 + \frac{1}{2} gh^2 \end{bmatrix} = 0 \quad (2.10)$$

Here  $h$  and  $w$  are the water depth and velocity, and  $g$  stands for the gravity acceleration constant. In the absence of dry bed, the water depth is always positive and

$$\Omega = \{\mathbf{u} \in \mathbb{R}^2 : h > 0\} \quad (2.11)$$

The total (kinetic and potential) energy  $U = \frac{1}{2} hw^2 + \frac{1}{2} gh^2$  is a convex function of  $\mathbf{u} \in \Omega$  and serves as an entropy function with

$$\mathbf{v} = \begin{bmatrix} gh - \frac{1}{2} w^2 \\ w \end{bmatrix}, \quad F = \frac{1}{2} hw^3 + gh^2 w, \quad \phi = \frac{1}{2} gh^2, \quad \psi = \frac{1}{2} gh^2 w \quad (2.12)$$

**Example 2.2.3.** The Euler equations of gas dynamics are

$$\frac{\partial}{\partial t} \begin{bmatrix} \rho \\ \rho w \\ E \end{bmatrix} + \frac{\partial}{\partial x} \begin{bmatrix} \rho w \\ \rho w^2 + p \\ w(E + p) \end{bmatrix} = 0 \quad (2.13)$$

Here  $\rho$ ,  $w$  and  $p$  are the density, velocity and pressure of the gas.  $E$  is the total energy. In the case of polytropic ideal gas, the equation of state is

$$E = \frac{1}{2} \rho w^2 + \frac{p}{\gamma - 1} \quad (2.14)$$

where  $\gamma$  is ratio of specific heats.  $\gamma = 5/3$  for monatomic gas and  $\gamma = 7/5$  corresponds to diatomic molecules. Assume that there is no vacuum. Then density and pressure need to be positive and

$$\Omega = \{\mathbf{u} \in \mathbb{R}^3 : \rho > 0, p > 0\} = \{\mathbf{u} \in \mathbb{R}^3 : \rho > 0, (\gamma - 1)(E - \frac{(\rho w)^2}{2\rho}) > 0\} \quad (2.15)$$

We can verify that  $\Omega$  is a convex set and (2.13) is hyperbolic in  $\Omega$ . The physical specific entropy is  $s = \log(p\rho^{-\gamma})$ . Harten [29] proved that there exists a family of entropy pairs that are related to  $s$  and symmetrize (2.13). However, if we also want to symmetrize the viscous term in the compressible Navier–Stokes equations with heat conduction [37], the only choice of entropy pair satisfying (2.7) is

$$U = -\frac{\rho s}{\gamma - 1}, \quad F = -\frac{\rho w s}{\gamma - 1} \quad (2.16)$$

The corresponding entropy variables and potential function–potential flux pair are

$$\mathbf{v} = \begin{bmatrix} \frac{\gamma - s}{\gamma - 1} - \frac{\rho w^2}{2p} \\ \rho w/p \\ -\rho/p \end{bmatrix}, \quad \phi = \rho, \quad \psi = \rho w \quad (2.17)$$

### 3. Entropy stable high order nodal DG schemes in one dimension

In this section, we proceed to unravel the entropy stable nodal DG scheme for one-dimensional systems of conservation laws

$$\frac{\partial \mathbf{u}}{\partial t} + \frac{\partial \mathbf{f}(\mathbf{u})}{\partial x} = 0 \quad (3.1)$$

Let us make some standard assumptions. Firstly, we have periodic or compactly supported boundary conditions. Secondly, time is always continuous, so that we conduct semidiscrete analysis. Finally, the numerical solution is kept within the set  $\Omega$ . For instance, density and pressure are assumed to be positive for Euler equations.

Our starting point is the classical DG scheme. Given a domain decomposition

$$x_{1/2} < x_{3/2} < \cdots < x_{N+1/2}, \quad I_i = [x_{i-1/2}, x_{i+1/2}], \quad \Delta x_i = x_{i+1/2} - x_{i-1/2}$$

and the discrete DG space of polynomial degree  $k$

$$\mathbf{V}_h^k = \{\mathbf{w}_h : \mathbf{w}_h|_{I_i} \in [\mathcal{P}^k(I_i)]^p, 1 \leq i \leq N\} \quad (3.2)$$

we seek  $\mathbf{u}_h \in \mathbf{V}_h^k$  such that for each  $\mathbf{w}_h \in \mathbf{V}_h^k$  and  $1 \leq i \leq N$ ,

$$\int_{I_i} \frac{\partial \mathbf{u}_h^T}{\partial t} \mathbf{w}_h dx - \int_{I_i} \mathbf{f}(\mathbf{u}_h)^T \frac{d\mathbf{w}_h}{dx} dx = -\widehat{\mathbf{f}}_{i+1/2}^T \mathbf{w}_h(x_{i+1/2}^-) + \widehat{\mathbf{f}}_{i-1/2}^T \mathbf{w}_h(x_{i-1/2}^+) \quad (3.3)$$

where  $\widehat{\mathbf{f}}_{i+1/2}$  is a single-valued numerical flux at the element interface, depending on the values of numerical solution from both sides

$$\widehat{\mathbf{f}}_{i+1/2} = \widehat{\mathbf{f}}(\mathbf{u}_h(x_{i+1/2}^-), \mathbf{u}_h(x_{i+1/2}^+)) \quad (3.4)$$

In general,  $\widehat{\mathbf{f}}_{i+1/2}$  is derived from some (exact or approximate) Riemann solver. (3.3) is usually called the *weak form*. We obtain the *strong form* after a simple integration by parts

$$\int_{I_i} \left( \frac{\partial \mathbf{u}_h^T}{\partial t} + \frac{\mathbf{f}(\mathbf{u}_h)^T}{\partial x} \right) \mathbf{w}_h dx = (\mathbf{f}(\mathbf{u}_h(x_{i+1/2}^-)) - \widehat{\mathbf{f}}_{i+1/2})^T \mathbf{w}_h(x_{i+1/2}^-) - (\mathbf{f}(\mathbf{u}_h(x_{i-1/2}^+)) - \widehat{\mathbf{f}}_{i-1/2})^T \mathbf{w}_h(x_{i-1/2}^+) \quad (3.5)$$

We are going to apply Legendre–Gauss–Lobatto quadrature rule with exactly  $k+1$  quadrature points to the two integrals in (3.3). Since the algebraic degree of accuracy is  $2k-1$ , the Gauss–Lobatto quadrature is not exact for the first integral, but is exact for the second term if  $\mathbf{f}$  is linear.

### 3.1. Gauss–Lobatto quadrature and summation-by-parts

Consider the reference element  $I = [-1, 1]$  associated with Gauss–Lobatto quadrature points

$$-1 = \xi_0 < \xi_1 < \dots < \xi_k = 1$$

and quadrature weights  $\{\omega_j\}_{j=0}^k$ . Define the Lagrangian (nodal) basis polynomials

$$L_j(\xi) = \prod_{\substack{l=0 \\ l \neq j}}^N \frac{\xi - \xi_l}{\xi_j - \xi_l}$$

such that  $L_j(\xi_l) = \delta_{jl}$ . Let  $\langle \cdot, \cdot \rangle$  and  $\langle \cdot, \cdot \rangle_\omega$  denote the continuous and discrete inner product

$$\langle u, v \rangle = \int_{-1}^1 u v d\xi, \quad \langle u, v \rangle_\omega = \sum_{j=0}^k \omega_j u(\xi_j) v(\xi_j)$$

The difference matrix  $D$  is set to be

$$D_{jl} = L'_l(\xi_j) \quad (3.6)$$

and the mass matrix  $M$  and stiffness matrix  $S$  are defined as

$$M_{jl} = \langle L_j, L_l \rangle_\omega = \omega_j \delta_{jl}, \text{ so that } M = \text{diag}\{\omega_0, \dots, \omega_k\} \quad (3.7)$$

$$S_{jl} = \langle L_j, L'_l \rangle_\omega = \langle L_j, L'_l \rangle \quad (3.8)$$

The discrete inner product contributes to a diagonal mass matrix, but also introduces some integration error. Such technique is typically termed *mass lumping*. On the other hand, the stiffness matrix is integrated exactly as  $L_j(\xi)L'_l(\xi)$  is of degree  $2k-1$ .

**Theorem 3.1** (Summation-by-parts property). Set the boundary matrix

$$B = \text{diag}\{-1, 0, \dots, 0, 1\} \quad (3.9)$$

Then we have

$$S = MD, \quad MD + D^T M = S + S^T = B \quad (3.10)$$

which is a discrete analogue of integration by parts.

**Proof.** Since  $S_{jl} = \sum_{j=0}^k \omega_r L_j(\xi_r) L'_l(\xi_r) = \omega_j L'_l(\xi_j) = M_{jj} D_{jl}$ , clearly  $S = MD$ . Moreover,

$$S_{jl} + S_{lj} = \langle L_j, L'_l \rangle + \langle L_l, L'_j \rangle = L_j(1) L_l(1) - L_j(-1) L_l(-1) = \delta_{kj} \delta_{kl} - \delta_{0j} \delta_{0l}$$

Hence  $B = S + S^T$ .  $\square$

**Theorem 3.2.** For each  $0 \leq j \leq k$  we have

$$\sum_{l=0}^k D_{jl} = \sum_{l=0}^k S_{jl} = 0, \quad \sum_{l=0}^k S_{lj} = \tau_j = \begin{cases} -1 & j=0 \\ 1 & j=k \\ 0 & 1 \leq j \leq k-1 \end{cases} \quad (3.11)$$

**Proof.** Since the sum of Lagrangian basis  $\sum_{l=0}^k L_l(\xi) = 1$ ,

$$\begin{aligned} \sum_{l=0}^k D_{jl} &= \sum_{l=0}^k L'_l(\xi_j) = 0, \quad \sum_{l=0}^k S_{jl} = \omega_j \sum_{l=0}^k D_{jl} = 0 \\ \sum_{l=0}^k S_{lj} &= \sum_{l=0}^k B_{jl} - \sum_{l=0}^k S_{jl} = \sum_{l=0}^k B_{jl} = B_{jj} = \tau_j \quad \square \end{aligned}$$

Using the matrices above, we are able to convert (3.3) into a compact matrix vector formulation based on nodal values. For clarity of notations we first work on scalar conservation laws. The weak form is

$$\int_{I_i} \frac{\partial u_h}{\partial t} w_h dx - \int_{I_i} f(u_h) \frac{dw_h}{dx} dx = -\widehat{f}_{i+1/2} w_h(x_{i+1/2}^-) + \widehat{f}_{i-1/2} w_h(x_{i-1/2}^+) \quad (3.12)$$

By the change of variables between  $I_i$  and the reference element  $I = [-1, 1]$

$$x_i(\xi) = \frac{1}{2}(x_{i-1/2} + x_{i+1/2}) + \frac{\xi}{2} \Delta x_i$$

the weak form on  $I$  is

$$\frac{\Delta x_i}{2} \int_I \frac{\partial u_h}{\partial t} w_h d\xi - \int_I f(u_h) \frac{dw_h}{d\xi} d\xi = -\widehat{f}_{i+1/2} w_h(x_i(1)) + \widehat{f}_{i-1/2} w_h(x_i(-1)) \quad (3.13)$$

Now we bring forth vector notations. Let  $\vec{u}^i$  denote the values of  $u_h$  at Gauss–Lobatto points

$$\vec{u}^i = [u_h(x_i(\xi_0)) \quad \cdots \quad u_h(x_i(\xi_k))]^T$$

Likewise, we can define  $\vec{w}^i$  and  $\vec{f}^i$

$$\vec{w}^i = [w_h(x_i(\xi_0)) \quad \cdots \quad w_h(x_i(\xi_k))]^T, \quad \vec{f}^i = [f(u_0^i) \quad \cdots \quad f(u_k^i)]^T$$

We also put the numerical fluxes into a vector

$$\vec{f}_*^i = [\widehat{f}_{i-1/2} \quad 0 \quad \cdots \quad 0 \quad \widehat{f}_{i+1/2}]^T$$

After applying Gauss–Lobatto quadrature, (3.13) becomes

$$\frac{\Delta x_i}{2} \vec{w}^{iT} M \frac{d\vec{u}^i}{dt} - (D\vec{w}^i)^T M \vec{f}^i = -\vec{w}^{iT} B \vec{f}_*^i \quad (3.14)$$

Since  $\vec{w}^i$  can be arbitrary,

$$\frac{\Delta x_i}{2} M \frac{d\vec{u}^i}{dt} - S^T \vec{f}^i = -B \vec{f}_*^i \quad (3.15)$$

which is the nodal DG formulation [33]. Using the SBP property (3.10), we can deduce another equivalent characterization, corresponding to the strong form (3.5).

$$\begin{aligned}\frac{\Delta x_i}{2} M \frac{d\vec{u}^i}{dt} + S \vec{f}^i &= B(\vec{f}^i - \vec{f}_*^i) \\ \frac{\Delta x_i}{2} \frac{d\vec{u}^i}{dt} + D \vec{f}^i &= M^{-1} B(\vec{f}^i - \vec{f}_*^i)\end{aligned}\quad (3.16)$$

It is also closely related to the spectral collocation method with a penalty type boundary treatment.

For systems, the nomenclature is essentially the same. The weak and strong nodal forms are

$$\frac{\Delta x_i}{2} \mathbf{M} \frac{d\vec{\mathbf{u}}^i}{dt} - \mathbf{S}^T \vec{\mathbf{f}}^i = -\mathbf{B} \vec{\mathbf{f}}_*^i \quad (3.17)$$

$$\frac{\Delta x_i}{2} \frac{d\vec{\mathbf{u}}^i}{dt} + \mathbf{D} \vec{\mathbf{f}}^i = \mathbf{M}^{-1} \mathbf{B}(\vec{\mathbf{f}}^i - \vec{\mathbf{f}}_*^i) \quad (3.18)$$

Everything is understood as a Kronecker product herein.

$$\vec{\mathbf{u}}^i = \begin{bmatrix} \mathbf{u}_h(x_i(\xi_0)) \\ \vdots \\ \mathbf{u}_h(x_i(\xi_k)) \end{bmatrix}, \quad \vec{\mathbf{f}}^i = \begin{bmatrix} \mathbf{f}(\mathbf{u}_0^i) \\ \vdots \\ \mathbf{f}(\mathbf{u}_k^i) \end{bmatrix}, \quad \vec{\mathbf{f}}_*^i = \begin{bmatrix} \widehat{\mathbf{f}}_{i-1/2} \\ \vdots \\ \widehat{\mathbf{f}}_{i+1/2} \end{bmatrix}$$

$$\mathbf{M} = M \otimes I_p, \quad \mathbf{D} = D \otimes I_p, \quad \mathbf{S} = S \otimes I_p, \quad \mathbf{B} = B \otimes I_p$$

These nodal DG forms do not satisfy any entropy condition, even for the square entropy function where we have entropy inequality for classical DG forms. However, due to the flexibility of nodal representation, we can modify these nodal forms to make them entropy stable. The key to this modification is the entropy conservative fluxes and entropy stable fluxes proposed by Tadmor [56,57], and defined as follows.

**Definition 3.1.** A consistent, symmetric two-point numerical flux  $\mathbf{f}_S(\mathbf{u}_L, \mathbf{u}_R)$  is entropy conservative for a given entropy function  $U$  if

$$(\mathbf{v}_R - \mathbf{v}_L)^T \mathbf{f}_S(\mathbf{u}_L, \mathbf{u}_R) = \psi_R - \psi_L \quad (3.19)$$

where  $\mathbf{v}_{L,R}$  and  $\psi_{L,R}$  are entropy variables and potential fluxes at the left and right states.

**Definition 3.2.** A consistent two-point numerical flux  $\widehat{\mathbf{f}}(\mathbf{u}_L, \mathbf{u}_R)$  is entropy stable for a given entropy function  $U$  if

$$(\mathbf{v}_R - \mathbf{v}_L)^T \widehat{\mathbf{f}}(\mathbf{u}_L, \mathbf{u}_R) - (\psi_R - \psi_L) \leq 0 \quad (3.20)$$

### 3.2. Single element: entropy conservative fluxes

The first step is to achieve internal entropy balance. We will concentrate on a single element and omit the superscript  $i$ . The modified scheme reads

$$\frac{\Delta x}{2} \frac{d\mathbf{u}_j}{dt} + 2 \sum_{l=0}^k D_{jl} \mathbf{f}_S(\mathbf{u}_j, \mathbf{u}_l) = \frac{\tau_j}{\omega_j} (\mathbf{f}_j - \mathbf{f}_{*,j}) \quad (3.21)$$

Here  $\mathbf{f}_S(\mathbf{u}_j, \mathbf{u}_l)$  is the symmetric entropy conservative flux for a given entropy function  $U$ . Notice that (3.18) can be written as

$$\frac{\Delta x}{2} \frac{d\mathbf{u}_j}{dt} + \sum_{l=0}^k D_{jl} \mathbf{f}(\mathbf{u}_l) = \frac{\tau_j}{\omega_j} (\mathbf{f}_j - \mathbf{f}_{*,j}) \quad (3.22)$$

Hence if we set  $\mathbf{f}_S(\mathbf{u}_j, \mathbf{u}_l) = \frac{1}{2}(\mathbf{f}(\mathbf{u}_j) + \mathbf{f}(\mathbf{u}_l))$ , we recover (3.18) ( $\sum_{l=0}^k D_{jl} \mathbf{f}(\mathbf{u}_l) = 0$ ). However, generally  $\frac{1}{2}(\mathbf{f}(\mathbf{u}_j) + \mathbf{f}(\mathbf{u}_l))$  is not entropy conservative.

The following theorem states that (3.21) is conservative, high order accurate and (internally) entropy conservative. The theorem is presented in [15]. We refine the proofs therein.

**Theorem 3.3.** If  $\mathbf{f}_S(\mathbf{u}_j, \mathbf{u}_l)$  is consistent and symmetric, then (3.21) is conservative and high order accurate. If we further assume that  $\mathbf{f}_S(\mathbf{u}_j, \mathbf{u}_l)$  is entropy conservative in the sense of (3.19), then (3.21) is also entropy conservative within a single element.

**Remark 3.1.** The conservation and entropy conservation of the scheme are both in the discrete sense. Specifically, the discrete integral of  $\mathbf{u}$  and  $U$  in the element are  $\sum_{j=0}^k \frac{\Delta x}{2} \omega_j \mathbf{u}_j$  and  $\sum_{j=0}^k \frac{\Delta x}{2} \omega_j U_j$ . As for accuracy, assume that  $\mathbf{u}$  is a smooth solution. Then the penalty term vanishes and we will show the truncation error at each collocation point is of  $k$ -th order

$$\frac{\partial \mathbf{f}(\mathbf{u})}{\partial x}(x(\xi_j)) - \frac{4}{\Delta x} \sum_{l=0}^k D_{jl} \mathbf{f}_S(\mathbf{u}(x(\xi_j)), \mathbf{u}(x(\xi_l))) = \mathcal{O}(\Delta x^k)$$

Notice that the truncation error is suboptimal, partly due to the fact that the Gauss–Lobatto quadrature is exact for polynomials of degree only up to  $2k - 1$ . In order to maintain optimal convergence, the algebraic degree of accuracy should be at least  $2k$  (consult [8]). We will see suboptimal convergence in some numerical tests.

**Proof.** Conservation:

$$\begin{aligned} \frac{d}{dt} \left( \sum_{j=0}^k \frac{\Delta x}{2} \omega_j \mathbf{u}_j \right) &= \sum_{j=0}^k \tau_j (\mathbf{f}_j - \mathbf{f}_{*,j}) - 2 \sum_{j=0}^k \sum_{l=0}^k S_{jl} \mathbf{f}_S(\mathbf{u}_j, \mathbf{u}_l) \\ &= \sum_{j=0}^k \tau_j (\mathbf{f}_j - \mathbf{f}_{*,j}) - \sum_{j=0}^k \sum_{l=0}^k (S_{jl} + S_{lj}) \mathbf{f}_S(\mathbf{u}_j, \mathbf{u}_l) \quad (\text{by symmetry}) \\ &= \sum_{j=0}^k \tau_j (\mathbf{f}_j - \mathbf{f}_{*,j}) - \sum_{j=0}^k \sum_{l=0}^k B_{jl} \mathbf{f}_S(\mathbf{u}_j, \mathbf{u}_l) \quad (\text{SBP property}) \\ &= \sum_{j=0}^k \tau_j (\mathbf{f}_j - \mathbf{f}_{*,j}) - \sum_{j=0}^k \tau_j \mathbf{f}(\mathbf{u}_j) = -(\mathbf{f}_{*,k} - \mathbf{f}_{*,0}) \end{aligned}$$

The only terms left are the interface numerical fluxes, which supports local conservation. It is also globally conservative since the interface numerical fluxes will cancel out when summing over elements.

**Accuracy:** let  $\tilde{\mathbf{f}}_S(x, y) = \mathbf{f}_S(\mathbf{u}(x), \mathbf{u}(y))$  and  $\tilde{\mathbf{f}}(x) = \mathbf{f}(\mathbf{u}(x))$ . Then  $\tilde{\mathbf{f}}_S$  is also symmetric and consistent in the sense that  $\tilde{\mathbf{f}}_S(x, x) = \tilde{\mathbf{f}}(x)$ . Hence

$$\frac{\partial \tilde{\mathbf{f}}}{\partial x}(x) = \frac{\partial \tilde{\mathbf{f}}_S}{\partial x}(x, x) + \frac{\partial \tilde{\mathbf{f}}_S}{\partial y}(x, x) = 2 \frac{\partial \tilde{\mathbf{f}}_S}{\partial y}(x, x)$$

Since the difference matrix  $D$  is exact for polynomials of degree up to  $k$ ,

$$\frac{4}{\Delta x} \sum_{l=0}^k D_{jl} \tilde{\mathbf{f}}_S(x(\xi_j), x(\xi_l)) = 2 \frac{\partial \tilde{\mathbf{f}}_S}{\partial y}(x(\xi_j), x(\xi_j)) + \mathcal{O}(\Delta x^k) = \frac{\partial \tilde{\mathbf{f}}}{\partial x}(x(\xi_j)) + \mathcal{O}(\Delta x^k)$$

Therefore the truncation error is  $\mathcal{O}(\Delta x^k)$  and the scheme is high order accurate.

**Entropy conservation:**

$$\frac{d}{dt} \left( \sum_{j=0}^k \frac{\Delta x}{2} \omega_j U_j \right) = \sum_{j=0}^k \frac{\Delta x}{2} \omega_j \mathbf{v}_j^T \frac{d\mathbf{u}_j}{dt} = \sum_{j=0}^k \tau_j \mathbf{v}_j^T (\mathbf{f}_j - \mathbf{f}_{*,j}) - 2 \sum_{j=0}^k \sum_{l=0}^k S_{jl} \mathbf{v}_j^T \mathbf{f}_S(\mathbf{u}_j, \mathbf{u}_l)$$

The second term is

$$\begin{aligned} \sum_{j=0}^k \sum_{l=0}^k (B_{jl} + S_{jl} - S_{lj}) \mathbf{v}_j^T \mathbf{f}_S(\mathbf{u}_j, \mathbf{u}_l) &= \sum_{j=0}^k \tau_j \mathbf{v}_j^T \mathbf{f}_j + \sum_{j=0}^k \sum_{l=0}^k S_{jl} (\mathbf{v}_j - \mathbf{v}_l)^T \mathbf{f}_S(\mathbf{u}_j, \mathbf{u}_l) \\ &= \sum_{j=0}^k \tau_j \mathbf{v}_j^T \mathbf{f}_j + \sum_{j=0}^k \sum_{l=0}^k S_{jl} (\psi_j - \psi_l) = \sum_{j=0}^k \tau_j (\mathbf{v}_j^T \mathbf{f}_j - \psi_j) \end{aligned}$$

Then

$$\frac{d}{dt} \left( \sum_{j=0}^k \frac{\Delta x}{2} \omega_j U_j \right) = \sum_{j=0}^k \tau_j (\psi_j - \mathbf{v}_j^T \mathbf{f}_{*,j}) = (\psi_k - \mathbf{v}_k^T \mathbf{f}_{*,k}) - (\psi_0 - \mathbf{v}_0^T \mathbf{f}_{*,0}) \quad (3.23)$$

We only have element boundary terms, so that the scheme is locally entropy conservative. The global entropy stability remains unclear and will be discussed later.  $\square$

In the scalar case, the entropy conservative flux is uniquely determined

$$f_S(u_L, u_R) = \begin{cases} \frac{\psi_R - \psi_L}{v_R - v_L} & u_L \neq u_R \\ f(u_L) & u_L = u_R \end{cases} \quad (3.24)$$

A prototype model is the Burgers equation with square entropy function where  $f = U = \frac{u^2}{2}$ . Then  $f_S(u_L, u_R) = \frac{1}{6}(u_L^2 + u_L u_R + u_R^2)$ , which actually corresponds to the nodal DG discretization of the canonical skew-symmetric splitting of Burgers equation [19,55].

For systems, (3.19) is underdetermined and  $\mathbf{f}_S(\mathbf{u}_L, \mathbf{u}_R)$  is not unique. A generic choice of entropy conservative flux is the following path integration [57].

$$\mathbf{f}_S(\mathbf{u}_L, \mathbf{u}_R) = \int_0^1 \mathbf{g}(\mathbf{v}_L + \lambda(\mathbf{v}_R - \mathbf{v}_L)) d\lambda \quad (3.25)$$

which may not have an explicit formula and can be computationally expensive. Fortunately, for many systems we are able to derive explicit entropy conservative fluxes that are easy to compute. Let us revisit the examples in Section 2.

**Example 3.2.1.** For a linear symmetric system, the entropy stable flux is simply the arithmetic mean

$$\mathbf{f}_S(\mathbf{u}_L, \mathbf{u}_R) = \frac{1}{2}(A\mathbf{u}_L + A\mathbf{u}_R) \quad (3.26)$$

Therefore, (3.18) is already locally entropy ( $L^2$ ) conservative.

**Example 3.2.2.** For shallow water equations, an explicit entropy conservative flux is

$$\mathbf{f}_S(\mathbf{u}_L, \mathbf{u}_R) = \left[ \begin{array}{c} \frac{1}{2}(h_L w_L + h_R w_R) \\ \frac{1}{4}(h_L w_L + h_R w_R)(w_L + w_R) + \frac{1}{2}gh_L h_R \end{array} \right] \quad (3.27)$$

It is also equivalent to the skew-symmetric splitting procedure in [20].

**Example 3.2.3.** For Euler equations, Ismail and Roe [38] suggested the following affordable entropy conservative flux

$$\begin{aligned} f_S^1 &= \overline{z^2} (\overline{z^3})^{\log} \\ f_S^2 &= \frac{\overline{z^3}}{\overline{z^1}} + \frac{\overline{z^2}}{\overline{z^1}} f_S^1 \\ f_S^3 &= \frac{1}{2} \frac{\overline{z^2}}{\overline{z^1}} \left( \frac{\gamma + 1}{\gamma - 1} \frac{(\overline{z^3})^{\log}}{(\overline{z^1})^{\log}} + f_S^2 \right) \end{aligned} \quad (3.28)$$

where

$$\mathbf{z} = \begin{bmatrix} z^1 \\ z^2 \\ z^3 \end{bmatrix} = \sqrt{\frac{\rho}{p}} \begin{bmatrix} 1 \\ w \\ p \end{bmatrix}$$

$\overline{z^s}$  and  $(\overline{z^s})^{\log}$  are the arithmetic mean and the logarithmic mean

$$\overline{z^s} = \frac{1}{2}(z_L^s + z_R^s), \quad (\overline{z^s})^{\log} = \frac{z_R^s - z_L^s}{\log z_R^s - \log z_L^s}, \quad s = 1, 2, 3$$

Another entropy conservative flux, which also preserves kinetic energy, was recommended by Chandrashekar in [7]:

$$\begin{aligned} f_S^1 &= (\overline{\rho})^{\log} \overline{w} \\ f_S^2 &= \frac{\overline{\rho}}{2\overline{\beta}} + \overline{w} f_S^1 \\ f_S^3 &= \left( \frac{1}{2(\gamma - 1)(\overline{\beta})^{\log}} - \frac{1}{2} \overline{w^2} \right) f_S^1 + \overline{w} f_S^2 \end{aligned} \quad (3.29)$$

where

$$\beta = \frac{\rho}{2p}$$

Due to the presence of the logarithmic mean, these fluxes are no longer equivalent to any kind of splitting.

### 3.3. Multiple elements: entropy stable fluxes

The single element analysis is not enough in that we are left with the element boundary terms in (3.23). The next theorem establishes that entropy stable interface numerical fluxes guarantee non-positive interface entropy production rate.

**Theorem 3.4.** *If the numerical flux  $\widehat{\mathbf{f}}$  at the element interface is entropy stable, then the scheme (3.21) is entropy stable.*

**Proof.** According to (3.23), the entropy production rate at the interface is

$$(\psi_k^i - (\mathbf{v}_k^i)^T \widehat{\mathbf{f}}_{i+1/2}) - (\psi_0^{i+1} - (\mathbf{v}_0^{i+1})^T \widehat{\mathbf{f}}_{i+1/2}) = (\mathbf{v}_0^{i+1} - \mathbf{v}_k^i)^T \widehat{\mathbf{f}}(\mathbf{u}_k^i, \mathbf{u}_0^{i+1}) - (\psi_0^{i+1} - \psi_k^i)$$

which is non-positive as  $\widehat{\mathbf{f}}$  is entropy stable. By the assumption of periodic or compactly supported boundary condition, the whole scheme is entropy stable.  $\square$

**Remark 3.2.** Along the lines of [15], the entropy stable nodal DG scheme can be written in the finite volume manner

$$\frac{\Delta x_i}{2} \omega_j \frac{d\mathbf{u}_j^i}{dt} + (\mathbf{f}_{j+1/2}^i - \mathbf{f}_{j-1/2}^i) = 0 \quad (3.30)$$

where

$$\mathbf{f}_{j+1/2}^i = \begin{cases} \widehat{\mathbf{f}}_{i-1/2} & j = -1 \\ \widehat{\mathbf{f}}_{i+1/2} & j = k \\ 2 \sum_{l=0}^j \sum_{r=j+1}^k S_{lr} \mathbf{f}_S(\mathbf{u}_l^i, \mathbf{u}_r^i) & 0 \leq j \leq k-1 \end{cases} \quad (3.31)$$

The entropy stability is also transformed into

$$\frac{\Delta x_i}{2} \omega_j \frac{dU_j^i}{dt} + (F_{j+1/2}^i - F_{j-1/2}^i) \leq 0 \quad (3.32)$$

where

$$F_{j+1/2}^i = \begin{cases} \frac{1}{2}((\mathbf{v}_k^{i-1} + \mathbf{v}_0^i)^T \widehat{\mathbf{f}}_{i-1/2} - (\psi_k^{i-1} + \psi_0^i)) & j = -1 \\ \frac{1}{2}((\mathbf{v}_k^i + \mathbf{v}_0^{i+1})^T \widehat{\mathbf{f}}_{i+1/2} - (\psi_k^i + \psi_0^{i+1})) & j = k \\ \sum_{l=0}^j \sum_{r=j+1}^k S_{lr} ((\mathbf{v}_l^i + \mathbf{v}_r^i)^T \mathbf{f}_S(\mathbf{u}_l^i, \mathbf{u}_r^i) - (\psi_l + \psi_r)) & 0 \leq j \leq k-1 \end{cases} \quad (3.33)$$

A Lax–Wendroff type argument will yield that, if a sequence of numerical solutions whose mesh size tends to zero converges boundedly and a.e. to some function, then the function is a weak solution of (1.1) supporting the required entropy condition. This is enough to determine the entropy solution of scalar conservation laws with strictly convex flux functions [48].

One may be tempted to let  $\widehat{\mathbf{f}}$  be the entropy conservative flux, giving rise to an entropy conservative scheme. However, entropy should be dissipated at shock waves and entropy conservative schemes will produce strong oscillations near shocks. The construction of entropy stable fluxes can be divided into two categories. In [38,7,5,17], the authors build  $\widehat{\mathbf{f}}$  by adding some numerical diffusion operators, of Lax–Friedrichs type or Roe type, to the entropy conservative flux, so that the amount of entropy dissipation can be precisely determined. On the other hand, it has been known for decades that the widely used upwind numerical fluxes, including monotone fluxes for scalar conservation laws and Godunov-type fluxes for systems, are entropy stable. Here we will follow the latter approach because of other desirable properties of upwind fluxes (e.g. bound-preserving property).

**Theorem 3.5.** *In the scalar case, suppose  $\widehat{f}(u_L, u_R)$  is monotone such that  $\widehat{f}$  is a non-decreasing function of its first argument and a non-increasing function of its second argument. Then  $\widehat{f}$  is entropy stable.*

**Proof.** By the mean value theorem, there exists  $\tilde{v}$  between  $v_L$  and  $v_R$  such that

$$\psi_R - \psi_L = (v_R - v_L)g(\tilde{v}) = (v_R - v_L)f(u(\tilde{v}))$$

Since  $u(v)$  is an increasing function,  $u(\tilde{v})$  is also between  $u_L$  and  $u_R$ . By the monotonicity of  $\widehat{f}$  we have

$$(u_R - u_L)(\widehat{f}(u_L, u_R) - f(u(\tilde{v}))) \leq 0 \quad (3.34)$$

Consequently

$$(\psi_R - \psi_L) - \widehat{f}(u_L, u_R)(v_R - v_L) = (v_R - v_L)(f(u(\tilde{v})) - \widehat{f}(u_L, u_R)) \geq 0$$



We remark that (3.34) is exactly the characterization of the E-flux [46].  $\square$

For systems, most popular numerical fluxes rely on Riemann solvers, which exactly compute or approximate the solution of the Riemann problem

$$\begin{cases} \frac{\partial \mathbf{u}}{\partial t} + \frac{\partial \mathbf{f}(\mathbf{u})}{\partial x} = 0 \\ \mathbf{u}(x, 0) = \begin{cases} \mathbf{u}_L & x \leq 0 \\ \mathbf{u}_R & x > 0 \end{cases} \end{cases} \quad (3.35)$$

The solution of the Riemann problem is self-similar. We assume that our Riemann solver also has self-similar structure and is denoted by  $\mathbf{q}(x/t; \mathbf{u}_L, \mathbf{u}_R)$ . Let  $\lambda_L$  and  $\lambda_R$  be the leftmost and rightmost wave speed such that

$$\mathbf{q}(r; \mathbf{u}_L, \mathbf{u}_R) = \begin{cases} \mathbf{u}_L & r \leq \lambda_L \\ \mathbf{u}_R & r \geq \lambda_R \end{cases} \quad (3.36)$$

The Riemann solver should be conservative. For any  $S_L \leq \min\{\lambda_L, 0\}$  and  $S_R \geq \max\{\lambda_R, 0\}$ , integrating along the rectangle  $[S_L, S_R] \times [0, 1]$  yields

$$\int_{S_L}^{S_R} \mathbf{q}(r; \mathbf{u}_L, \mathbf{u}_R) dr - (S_R \mathbf{u}_R - S_L \mathbf{u}_L) + (\mathbf{f}_R - \mathbf{f}_L) = 0 \quad (3.37)$$

The Godunov-type flux follows from integration along  $[0, S_R] \times [0, 1]$  or  $[S_L, 0] \times [0, 1]$  [32]:

$$\widehat{\mathbf{f}}(\mathbf{u}_L, \mathbf{u}_R) = \mathbf{f}_R + \int_0^{S_R} \mathbf{q}(r; \mathbf{u}_L, \mathbf{u}_R) dr - S_R \mathbf{u}_R = \mathbf{f}_L - \int_{S_L}^0 \mathbf{q}(r; \mathbf{u}_L, \mathbf{u}_R) dr - S_L \mathbf{u}_L \quad (3.38)$$

The following theorem reveals the condition to make  $\widehat{\mathbf{f}}$  entropy stable.

**Theorem 3.6.** Assume that the Riemann solver also satisfies the entropy inequality such that for any  $S_L \leq \min\{\lambda_L, 0\}$  and  $S_R \geq \max\{\lambda_R, 0\}$ ,

$$\int_{S_L}^{S_R} U(\mathbf{q}(r; \mathbf{u}_L, \mathbf{u}_R)) dr - (S_R U_R - S_L U_L) + F_R - F_L \leq 0 \quad (3.39)$$

Then the corresponding Godunov-type flux is entropy stable.

**Proof.** By (3.38) and Jensen's inequality,

$$\begin{aligned} \int_0^{S_R} U(\mathbf{q}(r; \mathbf{u}_L, \mathbf{u}_R)) dr &\geq S_R U\left(\frac{1}{S_R} \int_0^{S_R} \mathbf{q}(r; \mathbf{u}_L, \mathbf{u}_R) dr\right) = S_R U\left(\mathbf{u}_R + \frac{1}{S_R} (\widehat{\mathbf{f}}(\mathbf{u}_L, \mathbf{u}_R) - \mathbf{f}_R)\right) \\ \int_{S_L}^0 U(\mathbf{q}(r; \mathbf{u}_L, \mathbf{u}_R)) dr &\geq -S_L U\left(-\frac{1}{S_L} \int_{S_L}^0 \mathbf{q}(r; \mathbf{u}_L, \mathbf{u}_R) dr\right) = -S_L U\left(\mathbf{u}_L + \frac{1}{S_L} (\widehat{\mathbf{f}}(\mathbf{u}_L, \mathbf{u}_R) - \mathbf{f}_L)\right) \end{aligned}$$

Summing them up and applying (3.39) gives

$$S_R (U(\mathbf{u}_R + \frac{1}{S_R} (\widehat{\mathbf{f}}(\mathbf{u}_L, \mathbf{u}_R) - \mathbf{f}_R)) - U_R) - S_L (U(\mathbf{u}_L + \frac{1}{S_L} (\widehat{\mathbf{f}}(\mathbf{u}_L, \mathbf{u}_R) - \mathbf{f}_L)) - U_L) + F_R - F_L \leq 0$$

We send  $S_R \rightarrow \infty$  and  $S_L \rightarrow -\infty$ . The first term converges to  $\mathbf{v}_R^T (\widehat{\mathbf{f}}(\mathbf{u}_L, \mathbf{u}_R) - \mathbf{f}_R)$  and the second term converges to  $\mathbf{v}_L^T (\widehat{\mathbf{f}}(\mathbf{u}_L, \mathbf{u}_R) - \mathbf{f}_L)$ . The inequality above simplifies to

$$\mathbf{v}_R^T (\widehat{\mathbf{f}}(\mathbf{u}_L, \mathbf{u}_R) - \mathbf{f}_R) - \mathbf{v}_L^T (\widehat{\mathbf{f}}(\mathbf{u}_L, \mathbf{u}_R) - \mathbf{f}_L) + F_R - F_L = (\mathbf{v}_R - \mathbf{v}_L)^T \widehat{\mathbf{f}}(\mathbf{u}_L, \mathbf{u}_R) - (\psi_R - \psi_L) \leq 0$$

which is exactly the condition of an entropy stable flux.  $\square$

The Riemann problem can be solved exactly for shallow water equations and Euler equations. The resulting numerical flux is called Godunov flux. Since the exact solutions satisfy entropy conditions, we immediately have the following corollary.

**Corollary 3.1.** *Godunov flux is entropy stable.*

The computation of exact Riemann solver often requires several Newton–Raphson iteration steps. Practically we resort to approximate Riemann solvers to reduce computational cost. A commonly used approximate Riemann solver is the HLL Riemann solver [29], which assumes a constant middle state. We first set values to  $\lambda_L$  and  $\lambda_R$ . Then

$$\mathbf{q}(r; \mathbf{u}_L, \mathbf{u}_R) = \begin{cases} \mathbf{u}_L & r \leq \lambda_L \\ \mathbf{u}_R & r \geq \lambda_R \\ \frac{\lambda_R \mathbf{u}_R - \lambda_L \mathbf{u}_L - (\mathbf{f}_R - \mathbf{f}_L)}{\lambda_R - \lambda_L} & \lambda_L < r < \lambda_R \end{cases} \quad (3.40)$$

Inserting (3.38), we obtain the HLL flux

$$\widehat{\mathbf{f}}(\mathbf{u}_L, \mathbf{u}_R) = \begin{cases} \mathbf{f}_L & \lambda_L \geq 0 \\ \mathbf{f}_R & \lambda_R \leq 0 \\ \frac{\lambda_R \mathbf{f}_L - \lambda_L \mathbf{f}_R + \lambda_L \lambda_R (\mathbf{u}_R - \mathbf{u}_L)}{\lambda_R - \lambda_L} & \lambda_L < 0 < \lambda_R \end{cases} \quad (3.41)$$

Note that the local Lax–Friedrichs flux is simply a special case of HLL flux by choosing  $\lambda_L = -\lambda$  and  $\lambda_R = \lambda$ . The HLL flux (and local Lax–Friedrichs flux) is entropy stable provided we approximate  $\lambda_L$  and  $\lambda_R$  properly.

**Corollary 3.2.** *If  $\lambda_L$  is not larger than the true leftmost wave speed and  $\lambda_R$  is not smaller than the true rightmost wave speed, the HLL flux is entropy stable.*

**Proof.** It suffices to prove (3.39). Since the approximate wave fan is larger than the true wave fan and the middle state is constant. The HLL Riemann solver is simply an average of the exact Riemann solver. Another application of Jensen’s inequality completes the proof.  $\square$

The computation of  $\lambda_L$  and  $\lambda_R$  is, however, not trivial. Simplistic approximation usually fails to bound the true wave speeds. Toro [59,60] recommends the two-rarefaction approximation, and Guermond and Popov [27] prove that the two-rarefaction approximated wave speeds indeed provide the correct bounds for Euler equations with  $1 \leq \gamma \leq 5/3$ . We can also prove the similar result for shallow water equations. More details on two-rarefaction approximation will be given in Appendix A.

**Remark 3.3.** Semidiscrete analysis is a crucial assumption. Fully discrete entropy stability analysis is available for first-order schemes, and implicit time integration [44]. The entropy stability of high-order schemes equipped with explicit time integration, such as strong stability preserving (SSP) Runge–Kutta methods [25,52], is still an open problem. There are positive results for the  $L^2$  stability of the Runge–Kutta DG discretization of linear advection equation [64], but the nonlinear (in the sense of both flux function and entropy function) analogue is difficult to prove.

### 3.4. Compatibility with limiters

As in the classical DG scheme, it is possible to design TVD/TVB limiter and/or bound-preserving limiter as an extra stabilizing mechanism. Limiters tend to squeeze the data towards the cell average, and hence make total entropy smaller. We formulate such intuition in the following lemma.

**Lemma 3.1.** *Suppose  $\alpha_j > 0$ ,  $\mathbf{u}_j \in \Omega$  for  $0 \leq j \leq k$  with  $\sum_{j=0}^k \alpha_j = 1$ . Define the average  $\bar{\mathbf{u}} = \sum_{j=0}^k \alpha_j \mathbf{u}_j$ . We modify these values without changing the average. That is, let  $\tilde{\mathbf{u}}_j = \bar{\mathbf{u}} + \theta_j(\mathbf{u}_j - \bar{\mathbf{u}})$  such that  $0 \leq \theta_j \leq 1$  and  $\sum_{j=0}^k \alpha_j \tilde{\mathbf{u}}_j = \bar{\mathbf{u}}$ . Then for any convex entropy function  $U$ , we have*

$$\sum_{j=0}^k \alpha_j U(\tilde{\mathbf{u}}_j) \leq \sum_{j=0}^k \alpha_j U(\mathbf{u}_j) \quad (3.42)$$

**Proof.** Since  $\sum_{j=0}^k \alpha_j \tilde{\mathbf{u}}_j = \sum_{j=0}^k \alpha_j (\bar{\mathbf{u}} + \theta_j(\mathbf{u}_j - \bar{\mathbf{u}})) = \bar{\mathbf{u}}$ ,

$$\sum_{j=0}^k \alpha_j (1 - \theta_j) \mathbf{u}_j = \left( \sum_{j=0}^k \alpha_j (1 - \theta_j) \right) \bar{\mathbf{u}}$$

By the convexity of  $U$ ,

$$U(\tilde{\mathbf{u}}_j) \leq \theta_j U(\mathbf{u}_j) + (1 - \theta_j) U(\bar{\mathbf{u}}), \quad \left( \sum_{j=0}^k \alpha_j (1 - \theta_j) \right) U(\bar{\mathbf{u}}) \leq \sum_{j=0}^k \alpha_j (1 - \theta_j) U(\mathbf{u}_j)$$

Hence

$$\begin{aligned} \sum_{j=0}^k \alpha_j U(\tilde{\mathbf{u}}_j) &\leq \sum_{j=0}^k \alpha_j (\theta_j U(\mathbf{u}_j) + (1 - \theta_j) U(\bar{\mathbf{u}})) = \sum_{j=0}^k \alpha_j \theta_j U(\mathbf{u}_j) + \left( \sum_{j=0}^k \alpha_j (1 - \theta_j) \right) U(\bar{\mathbf{u}}) \\ &\leq \sum_{j=0}^k \alpha_j \theta_j U(\mathbf{u}_j) + \sum_{j=0}^k \alpha_j (1 - \theta_j) U(\mathbf{u}_j) = \sum_{j=0}^k \alpha_j U(\mathbf{u}_j) \quad \square \end{aligned}$$

The bound-preserving limiter was developed by Zhang and Shu in [66,67] to maintain the physical bound  $\Omega$  of numerical approximations, such as the maximum principle for scalar conservation laws and positivity of density and pressure for Euler equations. This technique is constructed on Gauss–Lobatto nodes, so that it perfectly matches our nodal DG scheme. We will clarify the theoretical issues of bound-preserving limiter in Appendix B. In a nutshell, we compute the cell average  $\bar{\mathbf{u}}^i = \sum_{j=0}^k \frac{\omega_j}{2} \mathbf{u}_j^i$  and perform a simple linear limiting procedure with some  $0 \leq \theta \leq 1$  such that  $\tilde{\mathbf{u}}_j^i = \bar{\mathbf{u}}^i + \theta(\mathbf{u}_j^i - \bar{\mathbf{u}}^i) \in \Omega$ . Clearly, we have the following entropy stability result due to Lemma 3.1.

**Theorem 3.7.** *Bound-preserving limiter does not increase entropy.*

Bound-preserving limiter helps enhance robustness, but the solution profile may still contain oscillations. The TVD/TVB limiter is well suited for damping oscillations. For scalar conservation laws, the TVD type limiting procedure can be defined as

$$\begin{aligned} \tilde{u}_0^i &= \bar{u}^i - m(\bar{u}^i - u_0^i, \bar{u}^{i+1} - \bar{u}^i, \bar{u}^i - \bar{u}^{i-1}), \quad \tilde{u}_k^i = \bar{u}^i + m(u_k^i - \bar{u}^i, \bar{u}^{i+1} - \bar{u}^i, \bar{u}^i - \bar{u}^{i-1}) \\ \tilde{u}_j^i &= \bar{u}^i + \theta(u_j^i - \bar{u}^i) \text{ with } \theta = \frac{(\tilde{u}_0^i - \bar{u}^i) + (\tilde{u}_k^i - \bar{u}^i)}{(u_0^i - \bar{u}^i) + (u_k^i - \bar{u}^i)}, \quad 1 \leq j \leq k-1 \end{aligned}$$

We set  $\theta$  such that cell average does not change. The minmod function  $m$  is

$$m(a, b, c) = \begin{cases} s \min\{|a|, |b|, |c|\} & \text{if } s = \text{sign}(a) = \text{sign}(b) = \text{sign}(c) \\ 0 & \text{otherwise} \end{cases}$$

The TVB (total variation bounded) limiter is devised by replacing  $m$  with the modified minmod function  $\tilde{m}$  [50].

$$\tilde{m}(a, b, c) = \begin{cases} a & \text{if } |a| \leq Mh^2 \\ \text{sign}(a) \max\{|m(a, b, c)|, Mh^2\} & \text{if } |a| > Mh^2 \end{cases}$$

Here,  $h = \max_{1 \leq i \leq N} \Delta x_i$  and  $M$  is a parameter that has to be tuned adequately.

**Theorem 3.8.** *For scalar conservation laws, the TVD/TVB limiter mentioned above does not increase entropy.*

**Proof.** We only focus on the TVD limiter. The proof for the TVB limiter is exactly the same. Without loss of generality we assume that  $\bar{u}^i = 0$ . According to Lemma 3.1, we only need to show  $0 \leq \tilde{u}_j^i / u_j^i \leq 1$  for each  $0 \leq j \leq k$ . By the definition of minmod function,

$$\frac{\tilde{u}_0^i}{u_0^i} = -\frac{m(-u_0^i, -\bar{u}^{i-1}, \bar{u}^{i+1})}{u_0^i} \in [0, 1], \quad \frac{\tilde{u}_k^i}{u_k^i} = \frac{m(u_k^i, -\bar{u}^{i-1}, \bar{u}^{i+1})}{u_k^i} \in [0, 1]$$

It remains to prove that  $0 \leq \theta \leq 1$ . If  $u_0^i$  and  $u_k^i$  have the same sign, it is obvious. Otherwise we assume that  $u_0^i < 0$ ,  $u_k^i > 0$  and  $u_0^i + u_k^i \geq 0$ . Then  $\tilde{u}_0^i = -\min\{-u_0^i, (\bar{u}^{i-1})^-, (\bar{u}^{i+1})^+\}$  and  $\tilde{u}_k^i = \min\{u_k^i, (\bar{u}^{i-1})^-, (\bar{u}^{i+1})^+\}$ . It is easy to verify that  $0 \leq \tilde{u}_0^i + \tilde{u}_k^i \leq u_0^i + u_k^i$ . Other cases can be proved in a similar fashion.  $\square$

**Remark 3.4.** In general the TVD/TVB limiter for systems is not guaranteed to be entropy stable. The reason is that different components or characteristics are limited independently, which does not satisfy the assumption of Lemma 3.1 and the influence on total entropy is undecided. Certainly we could come up with a limiter that squeezes all components to the same degree, but it might be too restrictive.

**Remark 3.5.** There is still a gap in our result: entropy stability relies on semi-discrete analysis, while limiters can only be applied to fully discrete schemes. If we assume the fully discrete version of (3.21) is entropy stable, since limiters do not increase total entropy, the scheme modified by limiters is also entropy stable.

#### 4. Generalization to triangular meshes

In this section, we move to the two-dimensional systems of conservation laws

$$\frac{\partial \mathbf{u}}{\partial t} + \frac{\partial \mathbf{f}_1(\mathbf{u})}{\partial x_1} + \frac{\partial \mathbf{f}_2(\mathbf{u})}{\partial x_2} = 0 \quad (4.1)$$

The one-dimensional framework can be directly applied to rectangular meshes through tensor product. The generalization to triangular meshes requires some extra effort in that we need to find high order SBP operators on triangles. We prove that the SBP property is related to a special quadrature rule of degree  $2k - 1$  on the triangle and of degree  $2k$  over edges. The mass matrix and boundary matrices come from quadrature weights and the difference matrices can be devised appropriately.

##### 4.1. SBP operators on triangles

The computational domain is divided into triangular elements. We assume periodic or compactly supported boundary condition, and that there is no hanging node in the triangular mesh. Without loss of generality we work on the reference element

$$T = \{\mathbf{x} : x_1 \geq 0, x_2 \geq 0, x_1 + x_2 \leq 1\} \quad (4.2)$$

Let  $k \in \mathbb{N}$  be the order of SBP operators.  $\mathcal{P}^k(T)$  is the set of polynomials of degree up to  $k$  restricted on  $T$ . The dimension of  $\mathcal{P}^k(T)$  is

$$n_k^* = \frac{(k+1)(k+2)}{2}$$

We aim to find a degree  $2k - 1$  quadrature rule associated with  $n_k$  nodes  $\{\mathbf{x}^j\}_{j=1}^{n_k}$  and positive weights  $\{\omega_j\}_{j=1}^{n_k}$ . Vector notations are again adopted. The restriction of function  $u$  on quadrature points is denoted by

$$\vec{u} = [u(\mathbf{x}^1) \quad \cdots \quad u(\mathbf{x}^{n_k})]^T$$

Let  $\{p_l(\mathbf{x})\}_{l=1}^{n_k^*}$  be a set of basis functions of  $\mathcal{P}^k(T)$ . We define the Vandermonde matrix  $V$  whose columns are the basis functions evaluated at nodes.

$$V = [\vec{p}_1 \quad \vec{p}_2 \quad \cdots \quad \vec{p}_{n_k^*}]$$

Likewise we introduce  $V_{x_1}$  and  $V_{x_2}$  whose columns are derivatives of the basis functions.

$$V_{x_1} = [\partial_{x_1} \vec{p}_1 \quad \partial_{x_1} \vec{p}_2 \quad \cdots \quad \partial_{x_1} \vec{p}_{n_k^*}], \quad V_{x_2} = [\partial_{x_2} \vec{p}_1 \quad \partial_{x_2} \vec{p}_2 \quad \cdots \quad \partial_{x_2} \vec{p}_{n_k^*}]$$

The  $k$ -th order SBP operators, constructed on nodes  $\{\mathbf{x}^j\}_{j=1}^{n_k}$ , are defined as follows. It is stronger than the definition in [34] as we also require diagonal boundary matrices.

**Definition 4.1.** Consider the diagonal mass matrix consisting of quadrature weights.

$$M = \text{diag}\{\omega_1, \cdots, \omega_{n_k}\} \quad (4.3)$$

Difference matrices  $D_1, D_2$  are  $k$ -th order SBP approximation of the gradient operator if

(i).  $D_1 \vec{p} = \partial_{x_1} \vec{p}$  and  $D_2 \vec{p} = \partial_{x_2} \vec{p}$  for any  $p \in \mathcal{P}^k(T)$ . In other words,

$$D_1 V = V_{x_1}, \quad D_2 V = V_{x_2} \quad (4.4)$$

(ii). Let  $S_1 = MD_1$  and  $S_2 = MD_2$  be the stiffness matrices. We have the SBP property

$$S_1 + S_1^T = B_1 = \text{diag}\{\tau_{1,1}, \cdots, \tau_{1,n_k}\}, \quad S_2 + S_2^T = B_2 = \text{diag}\{\tau_{2,1}, \cdots, \tau_{2,n_k}\} \quad (4.5)$$

$B_1$  and  $B_2$  are diagonal boundary matrices such that  $\tau_{1,j} = \tau_{2,j} = 0$  whenever  $\mathbf{x}^j \notin \partial T$ .

$B_1$  and  $B_2$  actually represent a quadrature rule over edges. The next theorem states that the algebraic degree of accuracy of the boundary quadrature rule is at least  $2k$ .

**Theorem 4.1.** Assume that there exist  $k$ -th order SBP difference matrices  $D_1, D_2$ , and boundary matrices  $B_1, B_2$ . Then for any  $p \in \mathcal{P}^{2k}(T)$ , we have

$$\sum_{j=0}^{n_k} \tau_{1,j} p(\mathbf{x}^j) = \int_{\partial T} p n_1 dS, \quad \sum_{j=0}^{n_k} \tau_{2,j} p(\mathbf{x}^j) = \int_{\partial T} p n_2 dS \quad (4.6)$$

where  $\mathbf{n} = [n_1 \ n_2]^T$  is the outer normal vector on  $\partial T$ .

**Proof.** For any  $p_1, p_2 \in \mathcal{P}^k(T)$ , integration by parts tells us

$$\int_T (p_1 \frac{\partial p_2}{\partial x_1} + p_2 \frac{\partial p_1}{\partial x_1}) d\mathbf{x} = \int_{\partial T} p_1 p_2 n_1 dS$$

Since the left hand side is an integration of a degree  $2k - 1$  polynomial, it is equal to the quadrature

$$\begin{aligned} \vec{p}_1^T M \partial_{x_1} \vec{p}_2 + \vec{p}_2^T M \partial_{x_1} \vec{p}_1 &= \vec{p}_1^T M D_1 \vec{p}_2 + \vec{p}_2^T M D_1 \vec{p}_1 \\ &= \vec{p}_1^T (S_1 + S_1^T) \vec{p}_2 = \vec{p}_1^T B_1 \vec{p}_2 = \sum_{j=1}^{n_k} \tau_{1,j} p_1(\mathbf{x}^j) p_2(\mathbf{x}^j) \end{aligned} \quad (4.7)$$

Hence the first equation of (4.6) holds for all  $p = p_1 p_2$  such that  $p_1, p_2 \in \mathcal{P}^k(T)$ . In particular it is satisfied by all monomials with degree up to  $2k$ , and so satisfied by all  $p \in \mathcal{P}^{2k}(T)$ . By the same token we can prove the second equation.  $\square$

**Corollary 4.1.** If  $B_1$  and  $B_2$  correspond to a degree  $2k$  boundary quadrature rule, then

$$V^T M V_{x_1} + V_{x_1}^T M V = V^T B_1 V, \quad V^T M V_{x_2} + V_{x_2}^T M V = V^T B_2 V \quad (4.8)$$

**Proof.** It is simply a rephrasing of (4.7).  $\square$

We now turn to the opposite direction. The following theorem guarantees the existence of SBP difference matrices as long as we have  $B_1$  and  $B_2$  satisfying (4.6). To the best of our knowledge, this is the first construction of triangular SBP operators with diagonal mass matrix and diagonal boundary matrices.

**Theorem 4.2.** Assume that  $n_k \geq n_k^*$  and  $V$  has linearly independent columns. If  $B_1$  and  $B_2$  correspond to a degree  $2k$  boundary quadrature rule, there exist  $k$ -th order difference matrices that meet the SBP property. To be more specific, if  $\{p_l(\mathbf{x})\}_{l=1}^{n_k^*}$  is an orthonormal set under discrete norm  $M$  such that  $V^T M V = I$ , then we can compute  $D_1$  and  $D_2$  by

$$D_1 = \frac{1}{2} (M^{-1} + V V^T) B_1 (I - V V^T M) + V_{x_1} V^T M \quad (4.9a)$$

$$D_2 = \frac{1}{2} (M^{-1} + V V^T) B_2 (I - V V^T M) + V_{x_2} V^T M \quad (4.9b)$$

**Proof.** It suffices to verify that the matrices given by (4.9) satisfy (4.4) and (4.5). For a more general basis set we can always orthonormalize it and apply (4.9). Since  $V^T M V = I$  and  $(I - V V^T M)V = 0$ ,  $D_1 V = V_{x_1}$  and  $D_2 V = V_{x_2}$ . As for the SBP property,

$$\begin{aligned} S_1 &= M D_1 = \frac{1}{2} (I + M V V^T) B_1 (I - V V^T M) + M V_{x_1} V^T M \\ &= \frac{1}{2} B_1 + \frac{1}{2} (M V V^T B_1 - B_1 V V^T M) + M (V_{x_1} V^T - \frac{1}{2} V V^T B_1 V V^T) M \end{aligned}$$

After summing  $S_1$  and its transpose, the first term becomes  $B_1$  and the second term vanishes. By (4.8), the third term is

$$M (V_{x_1} V^T + V V_{x_1}^T - V V^T B_1 V V^T) M = M ((I - V V^T M) V_{x_1} V^T + V V_{x_1}^T (I - M V V^T)) M$$

Since the columns of  $V_{x_1}$  are linear combinations of columns of  $V$ ,  $(I - V V^T M)V_{x_1} = 0$ , the third term also vanishes. Therefore  $S_1 + S_1^T = B_1$ . The proof of  $S_2 + S_2^T = B_2$  is the same.  $\square$

**Remark 4.1.** In the one-dimensional case,  $n_k = n_k^* = k + 1$  and  $V$  is invertible. We simply take  $D = V^{-1} V_x$ . Here we always need more than  $n_k^*$  nodes to accomplish the quadrature rule, which complicates the derivation of difference matrices.

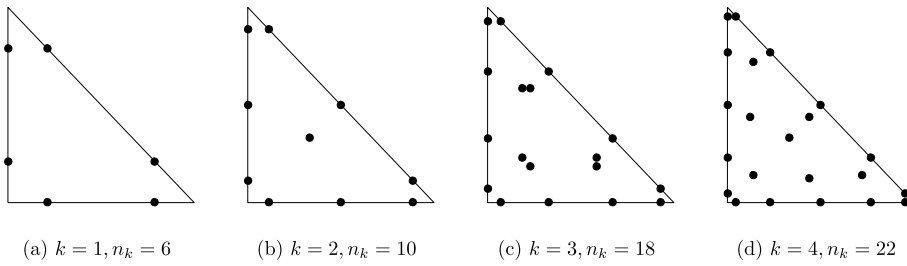


Fig. 4.1. Distributions of quadrature points on  $T$  with  $k = 1, 2, 3, 4$ .

Our remaining task is to find the quadrature rule that achieves interior and boundary accuracy simultaneously. For boundary accuracy, we put  $k + 1$  Legendre–Gauss points along each edge. Let  $\{\tau_j\}_{j=1}^{n_k}$  denote the Legendre–Gauss weights. Then the diagonal elements of boundary matrices are

$$\tau_{1,j} = \begin{cases} n_1(\mathbf{x}^j) \tau_j & \mathbf{x}^j \in \partial T \\ 0 & \mathbf{x}^j \notin \partial T \end{cases}, \quad \tau_{2,j} = \begin{cases} n_2(\mathbf{x}^j) \tau_j & \mathbf{x}^j \in \partial T \\ 0 & \mathbf{x}^j \notin \partial T \end{cases}$$

Let us summarize the prerequisites of the quadrature rule.

- It is symmetric so that adjacent elements can be glued together.
- The quadrature weights should be positive to make  $M$  positive-definite.
- It is exact for polynomials up to degree  $2k - 1$ .
- The quadrature points include  $k + 1$  Legendre–Gauss points on each edge.

Quadrature rules that meet these requirements are investigated in the literature. We use the software presented in [63] to obtain the rules of order  $k = 1, 2, 3$  and  $4$ .<sup>1</sup> The locations of quadrature points are illustrated in Fig. 4.1. For reference, we also list the coordinates of quadrature points and their quadrature weights in Appendix C.

**Remark 4.2.** The same requirements also arise in [68] where the authors tried to implement bound-preserving limiter on triangular meshes. They proposed a generic quadrature rule based on three warped transformation from a unit square to  $T$ . However,  $n_k = 3k(k + 1)$  for such technique, which is unnecessarily large.

To compute the difference matrices, we follow the standard practice in spectral element method and start with the set of orthonormal polynomials on triangle [40]. It is not an orthonormal basis under the discrete norm as the quadrature rule is not equal to exact integration for polynomials of degree  $2k$ . We still need to perform orthonormalization procedure. However, the condition number of the Vandermonde matrix will be small enough to prevent large error. For  $k = 1, 2$ , it is possible to use symbolic computation exclusively and compute the exact values of SBP matrices.

Finally, for a general triangle element  $\hat{T}$  such that the Jacobian matrix of affine mapping  $T \mapsto \hat{T}$  is denoted by  $J$ , the local SBP operators are

$$\begin{aligned} \hat{M} &= \det(J)M \\ \hat{D}_1 &= \frac{1}{\det(J)}(J_{22}D_1 - J_{21}D_2), \quad \hat{D}_2 = \frac{1}{\det(J)}(-J_{12}D_1 + J_{11}D_2) \\ \hat{B}_1 &= J_{22}B_1 - J_{21}B_2, \quad \hat{B}_2 = -J_{12}B_1 + J_{11}B_2 \end{aligned} \quad (4.10)$$

**Remark 4.3.** Conceptually, the SBP framework can be further generalized to higher dimensional simplex elements and even polygonal elements without any difficulty, as long as we find the quadrature rules, which could be a challenging task in practice.

#### 4.2. The entropy stable nodal DG schemes

With the SBP operators at hand, we are ready to mimic the procedure in Section 3 and develop high order entropy stable nodal DG schemes on triangular meshes. Analogously, we define the two-dimensional entropy conservative fluxes and entropy stable fluxes.

<sup>1</sup> <http://lsec.cc.ac.cn/phg/download/quadrule.tar.bz2>.

**Definition 4.2.** Consistent, symmetric numerical fluxes  $\mathbf{f}_{1,S}(\mathbf{u}_j, \mathbf{u}_l)$  and  $\mathbf{f}_{2,S}(\mathbf{u}_j, \mathbf{u}_l)$  are entropy conservative for a given entropy function  $U$  if

$$(\mathbf{v}_l - \mathbf{v}_j)^T \mathbf{f}_{1,S}(\mathbf{u}_j, \mathbf{u}_l) = \psi_{1,l} - \psi_{1,j}, \quad (\mathbf{v}_l - \mathbf{v}_j)^T \mathbf{f}_{2,S}(\mathbf{u}_j, \mathbf{u}_l) = \psi_{2,l} - \psi_{2,j} \quad (4.11)$$

**Definition 4.3.** Given a normal vector  $\mathbf{n} \in \mathbb{R}^2$ , a directional numerical flux  $\widehat{\mathbf{f}}(\mathbf{u}, \mathbf{u}^{\text{out}}, \mathbf{n})$  is consistent if

$$\widehat{\mathbf{f}}(\mathbf{u}, \mathbf{u}, \mathbf{n}) = n_1 \mathbf{f}_1(\mathbf{u}) + n_2 \mathbf{f}_2(\mathbf{u}) \quad (4.12)$$

It is called conservative if

$$\widehat{\mathbf{f}}(\mathbf{u}^{\text{out}}, \mathbf{u}, -\mathbf{n}) = -\widehat{\mathbf{f}}(\mathbf{u}, \mathbf{u}^{\text{out}}, \mathbf{n}) \quad (4.13)$$

A consistent and conservative directional numerical flux is entropy stable for a given entropy function  $U$  if

$$(\mathbf{v}^{\text{out}} - \mathbf{v})^T \widehat{\mathbf{f}}(\mathbf{u}, \mathbf{u}^{\text{out}}, \mathbf{n}) - (\psi_{\mathbf{n}}^{\text{out}} - \psi_{\mathbf{n}}) \leq 0, \quad \text{where } \psi_{\mathbf{n}} = n_1 \psi_1 + n_2 \psi_2 \quad (4.14)$$

Here the “out” superscript refers to the value from the other side of the interface.

Computationally efficient entropy conservative fluxes can be described in the same manner as in Section 3.2. The directional numerical fluxes correspond to directional Riemann solvers with the flux function  $\mathbf{f}_{\mathbf{n}}$ . As a consequence, the upwind numerical fluxes are still entropy stable. Since two-dimensional shallow water equations and Euler equations are rotationally invariant, the one-dimensional Riemann solvers can be directly used.

For clarity of notations, we explain the entropy stable nodal DG scheme on the reference element. Numerical solution collocated at the  $n_k$  quadrature points will be evolved. Let  $\bar{\mathbf{u}}$  denote the numerical solution and  $\bar{\mathbf{f}}_*$  stand for the vector of interface fluxes:

$$\mathbf{f}_{*,j} = \begin{cases} \widehat{\mathbf{f}}(\mathbf{u}_j, \mathbf{u}_j^{\text{out}}, \mathbf{n}(\mathbf{x}^j)) & \mathbf{x}^j \in \partial T \\ 0 & \mathbf{x}^j \notin \partial T \end{cases}$$

Similar to (3.21), the entropy stable nodal DG scheme is given by

$$\frac{d\mathbf{u}_j}{dt} + 2 \sum_{l=1}^{n_k} D_{1,jl} \mathbf{f}_{1,S}(\mathbf{u}_j, \mathbf{u}_l) + 2 \sum_{l=1}^{n_k} D_{2,jl} \mathbf{f}_{2,S}(\mathbf{u}_j, \mathbf{u}_l) = \frac{1}{\omega_j} (\tau_{1,j} \mathbf{f}_{1,j} + \tau_{2,j} \mathbf{f}_{2,j} - \tau_j \mathbf{f}_{*,j}) \quad (4.15)$$

The main properties of the scheme are outlined in the following theorem. We will omit most parts of the proof since it is almost the same as its one-dimensional counterpart.

**Theorem 4.3.** Assume that  $\mathbf{f}_{1,S}$  and  $\mathbf{f}_{2,S}$  are symmetric and consistent, and that  $\widehat{\mathbf{f}}$  is conservative and consistent. Then (4.15) is conservative and  $k$ -th order accurate. If we further assume that  $\mathbf{f}_{1,S}$  and  $\mathbf{f}_{2,S}$  are entropy conservative, and that  $\widehat{\mathbf{f}}$  is entropy stable, (4.15) is entropy conservative within single element and entropy stable across interfaces.

**Proof.** The proof of accuracy is the same as Theorem 3.3. As for conservation and entropy stability, we have

$$\frac{d}{dt} \left( \sum_{j=1}^{n_k} \omega_j \mathbf{u}_j \right) = - \sum_{j=1}^{n_k} \tau_j \mathbf{f}_{*,j} \quad (4.16)$$

and

$$\frac{d}{dt} \left( \sum_{j=1}^{n_k} \omega_j U_j \right) = \sum_{j=1}^{n_k} (\tau_{1,j} \psi_{1,j} + \tau_{2,j} \psi_{2,j} - \tau_j \mathbf{v}_j^T \mathbf{f}_{*,j}) = \sum_{j=1}^{n_k} \tau_j (\psi_{\mathbf{n},j} - \mathbf{v}_j^T \mathbf{f}_{*,j}) \quad (4.17)$$

Hence the scheme is locally conservative and entropy conservative. Since  $\widehat{\mathbf{f}}(\mathbf{u}_j, \mathbf{u}_j^{\text{out}}, \mathbf{n})$  and  $\widehat{\mathbf{f}}(\mathbf{u}_j^{\text{out}}, \mathbf{u}_j, -\mathbf{n})$  cancel out, the scheme is also globally conservative. The entropy production rate at interface  $\mathbf{x}^j$  is

$$\tau_j (\mathbf{v}_j^{\text{out}} - \mathbf{v}_j)^T \widehat{\mathbf{f}}(\mathbf{u}_j, \mathbf{u}_j^{\text{out}}, \mathbf{n}) - \tau_j (\psi_{\mathbf{n},j}^{\text{out}} - \psi_{\mathbf{n},j}) \leq 0$$

Therefore entropy is dissipated at the interface.  $\square$

**Remark 4.4.** The bound-preserving limiter can again be imposed naturally without affecting entropy stability. However, it is hard to design entropy stable TVD/TVB limiters.



**Remark 4.5.** The link between the entropy stable nodal DG scheme and the classical DG scheme seems vague due to the fact that the degree of freedom ( $n_k$ ) is larger than the dimension of the underlying polynomial basis ( $n_k^*$ ). We can build the bridge by considering the virtual element framework [2,3]. Let  $V^k(T)$  be a local space containing  $\mathcal{P}^k(T)$  such that  $\dim V^k(T) = n_k$ .  $\{L_l(\mathbf{x})\}_{l=1}^{n_k}$  is the set of Lagrangian basis functions such that  $L_l \in V^k(T)$  and  $L_l(\mathbf{x}^j) = \delta_{jl}$ . We define the discrete inner product  $\langle \cdot, \cdot \rangle_\omega$  corresponding to  $M$ , and discrete bilinear forms  $\langle \cdot, \cdot \rangle_{\tau_1}$  and  $\langle \cdot, \cdot \rangle_{\tau_2}$  corresponding to  $B_1$  and  $B_2$ .  $\Pi_\omega^k$  is set to be the  $L^2$  projection to  $\mathcal{P}^k(T)$  under  $\langle \cdot, \cdot \rangle_\omega$ . Recalling (4.9), we can characterize the stiffness matrices as

$$S_{1,jl} = \frac{1}{2} \langle (I + \Pi_\omega^k) l_j, (I - \Pi_\omega^k) l_l \rangle_{\tau_1} + \langle \Pi_\omega^k l_j, \frac{\partial (\Pi_\omega^k l_l)}{\partial x_1} \rangle_\omega \quad (4.18a)$$

$$S_{2,jl} = \frac{1}{2} \langle (I + \Pi_\omega^k) l_j, (I - \Pi_\omega^k) l_l \rangle_{\tau_2} + \langle \Pi_\omega^k l_j, \frac{\partial (\Pi_\omega^k l_l)}{\partial x_2} \rangle_\omega \quad (4.18b)$$

If we choose  $\mathbf{f}_{1,S}(\mathbf{u}_j + \mathbf{u}_l) = \frac{1}{2}(\mathbf{f}_1(\mathbf{u}_j) + \mathbf{f}_1(\mathbf{u}_l))$  and  $\mathbf{f}_{2,S}(\mathbf{u}_j + \mathbf{u}_l) = \frac{1}{2}(\mathbf{f}_2(\mathbf{u}_j) + \mathbf{f}_2(\mathbf{u}_l))$ , (4.15) turns into the nodal DG scheme

$$\mathbf{M} \frac{d\tilde{\mathbf{u}}}{dt} - \mathbf{S}_1^T \tilde{\mathbf{f}}_1 - \mathbf{S}_2^T \tilde{\mathbf{f}}_2 = -\mathbf{B} \tilde{\mathbf{f}}_*, \quad \mathbf{B} = B \otimes I_p, \quad B = \text{diag}\{\tau_1, \dots, \tau_{n_k}\} \quad (4.19)$$

with virtual element type stiffness matrices.

#### 4.3. Wall boundary condition of Euler equations

So far we have always assumed periodic or compactly supported boundary condition. There is a need to investigate the solid wall boundary condition of Euler equations. We will prove that the commonly used mirror state treatment is entropy stable. This subsection extends the one-dimensional analysis in [54].

Consider the two-dimensional Euler equation

$$\frac{\partial}{\partial t} \begin{bmatrix} \rho \\ \rho w_1 \\ \rho w_2 \\ E \end{bmatrix} + \frac{\partial}{\partial x} \begin{bmatrix} \rho w_1 \\ \rho w_1^2 + p \\ \rho w_1 w_2 \\ w_1(E + p) \end{bmatrix} + \frac{\partial}{\partial y} \begin{bmatrix} \rho w_2 \\ \rho w_1 w_2 \\ \rho w_2^2 + p \\ w_2(E + p) \end{bmatrix} = 0 \quad (4.20)$$

Here,  $\mathbf{w} = [w_1 \ w_2]^T$  is the velocity field. The equation of state is

$$E = \frac{1}{2} \rho (w_1^2 + w_2^2) + \frac{p}{\gamma - 1} \quad (4.21)$$

The entropy function, entropy variables and potential fluxes are given by

$$U = -\frac{\rho s}{\gamma - 1}, \quad \mathbf{v} = \begin{bmatrix} \frac{\gamma - s}{\gamma - 1} - \frac{\rho(w_1^2 + w_2^2)}{2p} \\ \rho w_1 / p \\ \rho w_2 / p \\ -\rho / p \end{bmatrix}, \quad \psi_1 = \rho w_1, \quad \psi_2 = \rho w_2 \quad (4.22)$$

At the wall boundary, we prescribe the no penetration condition; that is,

$$w_{\mathbf{n}} = w_1 n_1 + w_2 n_2 = 0 \quad (4.23)$$

Suppose that we have a numerical state  $\mathbf{u}$  on the solid wall. In order to weakly impose the no penetration condition, we have to provide an artificial state  $\mathbf{u}^{\text{out}}$  on the other side of the interface, and compute the numerical flux  $\hat{\mathbf{f}}(\mathbf{u}, \mathbf{u}^{\text{out}}, \mathbf{n})$ . Let  $w_{\mathbf{n}^\perp} = n_2 w_1 - n_1 w_2$ . The reflecting technique introduces a mirror state such that

$$\rho^{\text{out}} = \rho, \quad p^{\text{out}} = p, \quad w_{\mathbf{n}}^{\text{out}} = -w_{\mathbf{n}}, \quad w_{\mathbf{n}^\perp}^{\text{out}} = w_{\mathbf{n}^\perp} \quad (4.24)$$

The following theorem affirms the entropy stability of the reflecting technique.

**Theorem 4.4.** If  $\hat{\mathbf{f}}(\mathbf{u}, \mathbf{u}^{\text{out}}, \mathbf{n})$  is Godunov flux or HLL flux and  $\mathbf{u}^{\text{out}}$  is taken to be the mirror state (4.24), then such boundary treatment is entropy stable.

**Proof.** According to (4.17), we need to prove that the entropy production rate at the interface

$$\psi_{\mathbf{n}} - \mathbf{v}^T \hat{\mathbf{f}}(\mathbf{u}, \mathbf{u}^{\text{out}}, \mathbf{n})$$

is non-positive. By rotational symmetry, it is enough to consider the vertical wall  $x_1 = 0$ . Then  $\mathbf{n} = [1 \ 0]^T$  and

$$\mathbf{u} = [\rho \quad \rho w_1 \quad \rho w_2 \quad E]^T, \quad \mathbf{u}^{\text{out}} = [\rho \quad -\rho w_1 \quad \rho w_2 \quad E]^T$$

The numerical flux simply solves the one-dimensional Riemann problem in  $x$  direction. The exact Riemann solver will give a middle state  $\mathbf{u}^*$  such that  $w_1^* = 0$ . Hence the Godunov flux is

$$\widehat{\mathbf{f}}(\mathbf{u}, \mathbf{u}^{\text{out}}, \mathbf{n}) = \mathbf{f}_1(\mathbf{u}^*) = [0 \quad p^* \quad 0 \quad 0]^T \quad (4.25)$$

For the HLL Riemann solver, the two-rarefaction approximation yields  $\lambda_L = -\lambda$  and  $\lambda_R = \lambda$ . Then we actually have the local Lax–Friedrichs flux

$$\widehat{\mathbf{f}}(\mathbf{u}, \mathbf{u}^{\text{out}}, \mathbf{n}) = \frac{1}{2}(\mathbf{f}_1(\mathbf{u}) + \mathbf{f}_1(\mathbf{u}^{\text{out}})) - \frac{\lambda}{2}(\mathbf{u}^{\text{out}} - \mathbf{u}) = [0 \quad p + \lambda \rho w_1 \quad 0 \quad 0]^T \quad (4.26)$$

In both cases only the second component of  $\widehat{\mathbf{f}}$  is nonzero. On the other hand, since

$$\mathbf{v} = \left[ \frac{\gamma-s}{\gamma-1} - \frac{\rho(w_1^2 + w_2^2)}{2p} \quad \frac{\rho w_1}{p} \quad \frac{\rho w_2}{p} \quad -\frac{\rho}{p} \right]^T, \quad \psi_{\mathbf{n}} = \rho w_1$$

$$\mathbf{v}^{\text{out}} = \left[ \frac{\gamma-s}{\gamma-1} - \frac{\rho(w_1^2 + w_2^2)}{2p} \quad -\frac{\rho w_1}{p} \quad \frac{\rho w_2}{p} \quad -\frac{\rho}{p} \right]^T, \quad \psi_{\mathbf{n}}^{\text{out}} = -\rho w_1$$

we can easily verify that

$$\psi_{\mathbf{n}} - \mathbf{v}^T \widehat{\mathbf{f}}(\mathbf{u}, \mathbf{u}^{\text{out}}, \mathbf{n}) = (\mathbf{v}^{\text{out}})^T \widehat{\mathbf{f}}(\mathbf{u}, \mathbf{u}^{\text{out}}, \mathbf{n}) - \psi_{\mathbf{n}}^{\text{out}} = \frac{1}{2}((\mathbf{v}^{\text{out}} - \mathbf{v})^T \widehat{\mathbf{f}}(\mathbf{u}, \mathbf{u}^{\text{out}}, \mathbf{n}) - (\psi_{\mathbf{n}}^{\text{out}} - \psi_{\mathbf{n}}))$$

It is non-positive due to the entropy stability of Godunov flux and HLL flux.  $\square$

## 5. Generalization to convection–diffusion equations

In this section, we consider the entropy stable discretization of convection dominated convection–diffusion equations in two space dimensions. Recalling (2.6), we use the second derivatives of entropy variables to represent the diffusion term:

$$\frac{\partial \mathbf{u}}{\partial t} + \sum_{j=1}^2 \frac{\partial}{\partial x_j} (\mathbf{f}_j(\mathbf{u}) - \sum_{l=1}^2 \widehat{c}_{jl}(\mathbf{v}) \frac{\partial \mathbf{v}}{\partial x_l}) = 0 \quad (5.1)$$

where

$$\begin{bmatrix} \widehat{c}_{11}(\mathbf{v}) & \widehat{c}_{12}(\mathbf{v}) \\ \widehat{c}_{21}(\mathbf{v}) & \widehat{c}_{22}(\mathbf{v}) \end{bmatrix}$$

should be symmetric semi-positive-definite to ensure entropy dissipation. The convective part will be handled in the same way as section 4. For the diffusive part, we present an approach closely resembling the LDG method of Cockburn and Shu [11], with provable entropy stability. The treatment on Cartesian meshes was recognized in [5].

We rewrite (5.1) as the mixed formulation

$$\frac{\partial \mathbf{u}}{\partial t} + \sum_{j=1}^2 \frac{\partial}{\partial x_j} (\mathbf{f}_j(\mathbf{u}) - \mathbf{q}_j) = 0, \quad \mathbf{q}_j = \sum_{l=1}^2 \widehat{c}_{jl}(\mathbf{v}) \boldsymbol{\theta}_l, \quad \boldsymbol{\theta}_l = \frac{\partial \mathbf{v}}{\partial x_l} \quad (5.2)$$

Let  $\mathbf{Q} = [\mathbf{q}_1 \quad \mathbf{q}_2]$  and  $\Theta = [\boldsymbol{\theta}_1 \quad \boldsymbol{\theta}_2]$ . The LDG type approach evolves the nodal discretization of  $\mathbf{u}$  and  $\Theta$  simultaneously. The coupling between adjoining elements are achieved by  $\widehat{\mathbf{f}}(\mathbf{u}, \mathbf{u}^{\text{out}}, \mathbf{n})$  and single-valued numerical fluxes of  $\mathbf{v}$  and  $\mathbf{q}$ :

$$\widehat{\mathbf{v}} = \widehat{\mathbf{v}}(\mathbf{v}, \mathbf{v}^{\text{out}}), \quad \widehat{\mathbf{Q}} = [\widehat{\mathbf{q}}_1 \quad \widehat{\mathbf{q}}_2] = \widehat{\mathbf{Q}}(\mathbf{v}, \mathbf{v}^{\text{out}}, \mathbf{Q}, \mathbf{Q}^{\text{out}}) \quad (5.3)$$

Once again  $\bar{\mathbf{u}}$ ,  $\bar{\boldsymbol{\theta}}_1$  and  $\bar{\boldsymbol{\theta}}_2$  denote the numerical solutions collocated at the SBP nodes in the reference element.  $\bar{\mathbf{q}}_1$  and  $\bar{\mathbf{q}}_2$  are given by

$$\mathbf{q}_{j,r} = \sum_{l=1}^2 \widehat{c}_{jl}(\mathbf{v}_r) \boldsymbol{\theta}_{l,r}, \quad 1 \leq r \leq n_k$$

Additionally, we also let  $\bar{\mathbf{v}}_*$ ,  $\bar{\mathbf{q}}_{1,*}$  and  $\bar{\mathbf{q}}_{2,*}$  describe the vectors of corresponding numerical fluxes. The nodal version of the LDG scheme is

$$\frac{d\bar{\mathbf{u}}}{dt} + \text{C.D.T} - \sum_{j=1}^2 \mathbf{D}_j \bar{\mathbf{q}}_j = \text{C.B.T} - \sum_{j=1}^2 \mathbf{M}^{-1} \mathbf{B}_j (\bar{\mathbf{q}}_j - \bar{\mathbf{q}}_{j,*}) \quad (5.4a)$$

$$\bar{\boldsymbol{\theta}}_l - \mathbf{D}_l \bar{\mathbf{v}} = -\mathbf{M}^{-1} \mathbf{B}_l (\bar{\mathbf{v}} - \bar{\mathbf{v}}_*), \quad l = 1, 2 \quad (5.4b)$$

where C.D.T and C.B.T are the convective difference terms and convective boundary terms in (4.15). We can also write down the component-wise formulation:

$$\frac{d\mathbf{u}_r}{dt} + 2 \sum_{j=1}^2 \sum_{s=1}^{n_k} D_{j,rs} \mathbf{f}_{j,s}(\mathbf{u}_r, \mathbf{u}_s) - \sum_{j=1}^2 \sum_{s=1}^{n_k} D_{j,rs} \mathbf{q}_{j,s} = \sum_{j=1}^2 \frac{\tau_{j,r}}{\omega_r} (\mathbf{f}_{j,r} - \mathbf{q}_{j,r} + \mathbf{q}_{j,*,r}) - \frac{\tau_r}{\omega_r} \mathbf{f}_{*,r} \quad (5.5a)$$

$$\theta_{l,r} - \sum_{s=1}^{n_k} D_{l,rs} \mathbf{v}_s = \frac{\tau_{l,r}}{\omega_r} (\mathbf{v}_{*,r} - \mathbf{v}_r), \quad l = 1, 2 \quad (5.5b)$$

As indicated in the next theorem, the conventional LDG fluxes will lead to an entropy stable scheme.

**Theorem 5.1.** We introduce the average form  $\{\cdot\}$  and the jump form  $[\cdot]$  over the element boundary with outer normal vector  $\mathbf{n} = [n_1 \ n_2]^T$ .

$$\begin{aligned} \{\mathbf{v}\} &= \frac{1}{2}(\mathbf{v} + \mathbf{v}^{out}), \quad \{Q\} = \frac{1}{2}(Q + Q^{out}) \\ [\mathbf{v}] &= (\mathbf{v} - \mathbf{v}^{out})\mathbf{n}^T, \quad [Q] = (Q - Q^{out})\mathbf{n} \end{aligned} \quad (5.6)$$

Given parameters  $\alpha \geq 0$  and  $\beta \in \mathbb{R}^2$ , if we use the LDG fluxes

$$\widehat{\mathbf{v}}(\mathbf{v}, \mathbf{v}^{out}) = \{\mathbf{v}\} + [\mathbf{v}]\beta, \quad \widehat{Q}(\mathbf{v}, \mathbf{v}^{out}, Q, Q^{out}) = \{Q\} - [Q]\beta^T - \alpha[\mathbf{v}] \quad (5.7)$$

Then the nodal scheme (5.4) is entropy stable.

**Proof.** We multiply (5.4a) by  $\vec{\mathbf{v}}^T \mathbf{M}$  and (5.4b) by  $\vec{\mathbf{q}}_l^T \mathbf{M}$ , and sum them up. The convective part is already entropy stable. The remaining terms are

$$\begin{aligned} & - \sum_{l=1}^2 \vec{\mathbf{q}}_l^T \mathbf{M} \vec{\theta}_l + \sum_{l=1}^2 (\vec{\mathbf{v}}^T \mathbf{M} D_l \vec{\mathbf{q}}_l + \vec{\mathbf{q}}_l^T \mathbf{M} D_l \vec{\mathbf{v}} - \vec{\mathbf{v}}^T \mathbf{B}_l (\vec{\mathbf{q}}_l - \vec{\mathbf{q}}_{l,*}) - \vec{\mathbf{q}}_l^T \mathbf{B}_l (\vec{\mathbf{v}} - \vec{\mathbf{v}}_*)) \\ & = - \sum_{l=1}^2 \vec{\mathbf{q}}_l^T \mathbf{M} \vec{\theta}_l + \sum_{l=1}^2 (\vec{\mathbf{v}}^T \mathbf{B}_l \vec{\mathbf{q}}_{l,*} + \vec{\mathbf{q}}_l^T \mathbf{B}_l \vec{\mathbf{v}}_* - \vec{\mathbf{v}}^T \mathbf{B}_l \vec{\mathbf{q}}_l) \end{aligned}$$

The first sum is the interior contribution, it is non-positive since

$$- \sum_{l=1}^2 \vec{\mathbf{q}}_l^T \mathbf{M} \vec{\theta}_l = - \sum_{r=1}^{n_k} \omega_r \left( \sum_{l=1}^2 \mathbf{q}_{l,r}^T \theta_{l,r} \right) = - \sum_{r=1}^{n_k} \omega_r \left( \sum_{j=1}^2 \sum_{l=1}^2 \theta_{j,r}^T \widehat{C}_{jl}(\mathbf{v}_r) \theta_{l,r} \right) \leq 0$$

The boundary contribution reduces to

$$\sum_{r=1}^{n_k} \sum_{l=1}^2 \tau_{l,r} (\mathbf{v}_r^T \mathbf{q}_{l,*,r} + \mathbf{q}_{l,r}^T \mathbf{v}_{*,r} - \mathbf{v}_r^T \mathbf{q}_{l,r}) = \sum_{r=1}^{n_k} \tau_r \left( \sum_{l=1}^2 n_{l,r} (\mathbf{v}_r^T \mathbf{q}_{l,*,r} + \mathbf{q}_{l,r}^T \mathbf{v}_{*,r} - \mathbf{v}_r^T \mathbf{q}_{l,r}) \right) \equiv \sum_{r=1}^{n_k} \tau_r A_r$$

If  $\mathbf{x}^r \in \partial T$ , we add the corresponding terms from the other side of the interface. The contribution at  $\mathbf{x}^r$  is

$$\begin{aligned} A_r + A_r^{out} &= \sum_{l=1}^2 n_{l,r} [(\mathbf{v}_r - \mathbf{v}_r^{out})^T \widehat{\mathbf{q}}_l(\mathbf{v}_r, \mathbf{v}_r^{out}, Q_r, Q_r^{out}) \\ & \quad + (\mathbf{q}_{l,r} - \mathbf{q}_{l,r}^{out})^T \widehat{\mathbf{v}}(\mathbf{v}, \mathbf{v}^{out}) - (\mathbf{v}_r^T \mathbf{q}_{l,r} - (\mathbf{v}_r^{out})^T \mathbf{q}_{l,r}^{out})] \end{aligned} \quad (5.8)$$

Due to the identity

$$\mathbf{v}_r^T \mathbf{q}_{l,r} - (\mathbf{v}_r^{out})^T \mathbf{q}_{l,r}^{out} = \frac{1}{2}(\mathbf{v}_r + \mathbf{v}_r^{out})^T (\mathbf{q}_{l,r} - \mathbf{q}_{l,r}^{out}) + \frac{1}{2}(\mathbf{v}_r - \mathbf{v}_r^{out})^T (\mathbf{q}_{l,r} + \mathbf{q}_{l,r}^{out})$$

rearranging the terms in (5.8) and plugging (5.7) yields

$$\begin{aligned} A_r + A_r^{out} &= \text{tr}([\mathbf{v}_r]^T \widehat{Q}(\mathbf{v}_r, \mathbf{v}_r^{out}, Q_r, Q_r^{out})) + [Q_r]^T \widehat{\mathbf{v}}(\mathbf{v}_r, \mathbf{v}_r^{out}) - \{\mathbf{v}_r\}^T [Q_r] - \text{tr}([\mathbf{v}_r]^T \{Q_r\}) \\ &= -\alpha \text{tr}([\mathbf{v}_r]^T [\mathbf{v}_r]) \leq 0 \end{aligned}$$

Therefore the boundary contribution is also non-positive and our nodal LDG scheme is entropy stable.  $\square$

**Remark 5.1.** Both  $\alpha$  and  $\beta$  may be a function of  $\mathbf{x}$ . We can also replace  $\alpha$  by a symmetric positive-definite  $p \times p$  matrix.

**Table 6.1**

**Example 6.1.1:** accuracy test of the one-dimensional linear advection equation associated with initial data  $u(x, 0) = \sin^4(x)$  and exponential entropy function at  $t = 2\pi$ .

$k$	$N$	$L^1$ error	Order	$L^2$ error	Order	$L^\infty$ error	Order
2	20	7.030e-2	–	3.347e-2	–	2.688e-2	–
	40	5.363e-3	3.712	2.669e-3	3.649	2.340e-3	3.522
	80	4.575e-4	3.551	2.205e-4	3.598	1.846e-4	3.664
	160	4.414e-5	3.374	2.230e-5	3.305	2.582e-5	2.838
	320	4.745e-6	3.218	2.595e-6	3.103	3.626e-6	2.832
	640	5.485e-7	3.113	3.181e-7	3.028	4.794e-7	2.919
3	20	3.097e-3	–	1.514e-3	–	1.890e-3	–
	40	1.675e-4	4.208	8.672e-5	4.126	1.359e-4	3.798
	80	1.053e-5	3.993	5.372e-6	4.013	8.928e-6	3.928
	160	6.571e-7	4.002	3.354e-7	4.001	5.664e-7	3.978
	320	4.107e-8	4.000	2.096e-8	4.000	3.553e-8	3.995
	640	2.553e-8	5.000	1.151e-8	5.000	9.454e-9	5.028
4	10	2.608e-2	–	1.178e-2	–	8.580e-3	–
	20	8.325e-4	4.969	3.763e-4	4.969	3.497e-4	4.617
	40	2.623e-5	4.988	1.179e-5	4.997	9.860e-6	5.149
	80	8.170e-7	5.004	3.683e-7	5.000	3.084e-7	4.999
	160	2.553e-8	5.000	1.151e-8	5.000	9.454e-9	5.028
	320	7.945e-9	5.000	3.817e-9	5.000	2.981e-9	5.000

## 6. Numerical experiments

In this section, we test the performance of the entropy stable nodal DG schemes (3.21) and (4.15). One-dimensional tests are performed on uniform grids and two-dimensional tests are performed on unstructured triangular meshes generated by Gmsh<sup>2</sup> [21]. The schemes are integrated in time with third order SSP Runge–Kutta method (given in Appendix B). Unless otherwise pointed out, Godunov flux will be employed at element interfaces. For Euler equations, the ratio of specific heat  $\gamma$  is taken to be 7/5, and the entropy conservative flux (3.29) will be used as it seems to give better results than (3.28).

### 6.1. Smooth tests

Various test problems with smooth solutions are presented to validate the accuracy of the scheme. We would like to compute on elements of degree  $k = 2, 3, 4$ . If  $k = 2$ , we set the CFL number to be 0.15; otherwise we will let  $\Delta t = \text{CFL} \cdot h^{(k+1)/3}$  where  $h$  is the characteristic length of the mesh, so that time error will be dominated by space error.

**Example 6.1.1.** We solve the one-dimensional linear advection equation

$$\frac{\partial u}{\partial x} + \frac{\partial u}{\partial t} = 0, \quad x \in [0, 2\pi]$$

with periodic boundary condition and initial data  $u(x, 0) = \sin^4(x)$ . The exact solution is  $u(x, t) = \sin^4(x - t)$ . The entropy function in this case is the exponential function  $U = e^u$ , and the entropy conservative flux is given by

$$f_S(u_L, u_R) = \frac{(u_R - 1)e^{u_R} - (u_L - 1)e^{u_L}}{e^{u_R} - e^{u_L}}, \quad \text{if } u_L \neq u_R$$

When  $|u_L - u_R|$  is small, such formula suffers from round-off effect. Instead, we should use Taylor's expansion to approximate the numerator and the denominator. Numerical errors and orders of convergence of the entropy stable nodal DG scheme with  $k = 2, 3, 4$  are listed in Table 6.1. The scheme is evolved up to  $t = 2\pi$ . We observe optimal convergence for all values of  $k$ , better than the prediction of truncation error analysis. Probably the reason is that Gauss–Lobatto quadrature is exact for the linear convective part.

**Example 6.1.2.** Next we consider the one-dimensional Burgers equation

$$\frac{\partial u}{\partial t} + \frac{\partial(u^2/2)}{\partial x} = 0, \quad x \in [0, 2\pi]$$

with periodic boundary condition and initial data  $u(x, 0) = 0.5 + \sin x$ . The exact solution can be obtained by tracing back characteristic lines. We choose square entropy function  $U = u^2/2$ . Then the entropy stable nodal DG scheme is equivalent to the skew-symmetric splitting. In Table 6.2, we present the errors at  $t = 0.5$  when the solution is still smooth. It is evident that the convergence rate is slightly below optimal, especially for the  $L^\infty$  error. However, when  $k = 3$  we still have optimal convergence.

<sup>2</sup> <http://gmsh.info/>.

**Table 6.2**

**Example 6.1.2:** accuracy test of the one-dimensional Burgers equation associated with initial data  $u(x, 0) = 0.5 + \sin x$  and square entropy function at  $t = 0.5$ .

$k$	$N$	$L^1$ error	Order	$L^2$ error	Order	$L^\infty$ error	Order
2	40	1.320e-3	–	1.178e-3	–	3.269e-3	–
	80	2.071e-4	2.672	2.284e-4	2.366	7.923e-4	2.045
	160	3.162e-5	2.711	4.316e-5	2.404	2.078e-4	1.931
	320	4.724e-6	2.743	7.979e-6	2.435	5.100e-5	2.026
	640	6.911e-7	2.773	1.450e-6	2.460	1.290e-5	1.983
	1280	9.930e-8	2.799	2.606e-7	2.477	3.209e-6	2.008
3	40	4.344e-5	–	4.566e-5	–	1.658e-4	–
	80	3.348e-6	3.698	3.703e-6	3.624	1.610e-5	3.364
	160	2.344e-7	3.836	2.771e-7	3.740	1.306e-6	3.624
	320	1.577e-8	3.894	1.950e-8	3.829	9.301e-8	3.812
	640	1.036e-9	3.928	1.336e-9	3.868	6.252e-9	3.895
4	20	6.782e-5	–	6.319e-5	–	1.525e-4	–
	40	2.630e-6	4.688	2.849e-6	4.471	1.126e-5	3.760
	80	1.067e-7	4.624	1.374e-7	4.375	7.149e-7	3.977
	160	4.203e-9	4.666	6.385e-9	4.427	4.342e-8	4.041
	320	1.576e-10	4.737	2.858e-10	4.481	2.620e-9	4.050

**Table 6.3**

**Example 6.1.3:** accuracy test of the two-dimensional linear advection equation associated with initial data  $u(\mathbf{x}, 0) = \sin(2\pi x_1) \sin(2\pi x_2)$  and square entropy function at  $t = 0.2$ .

$k$	$h$	$L^1$ error	Order	$L^2$ error	Order	$L^\infty$ error	Order
2	1/8	3.380e-3	–	6.000e-3	–	6.890e-2	–
	1/16	5.032e-4	2.748	9.868e-4	2.604	1.809e-2	1.930
	1/32	6.170e-5	3.028	1.213e-4	3.024	2.292e-3	2.981
	1/64	7.916e-6	2.962	1.551e-5	2.967	3.387e-4	2.758
	1/128	9.890e-7	3.001	1.926e-6	3.010	4.419e-5	2.938
	1/256	1.244e-7	2.991	2.414e-7	2.996	5.929e-6	2.898
3	1/8	2.329e-4	–	4.375e-4	–	8.752e-3	–
	1/16	2.114e-5	3.461	3.536e-5	3.629	8.228e-4	3.411
	1/32	1.790e-6	3.562	2.810e-6	3.654	6.194e-5	3.731
	1/64	1.429e-7	3.647	2.210e-7	3.668	4.310e-6	3.845
	1/128	1.063e-8	3.748	1.658e-8	3.737	3.183e-7	3.759
	1/256	7.341e-10	3.856	1.160e-9	3.838	2.194e-8	3.859
4	1/8	1.295e-5	–	2.230e-5	–	6.184e-4	–
	1/16	4.534e-7	4.837	9.969e-7	4.483	6.627e-5	3.222
	1/32	1.528e-8	4.891	2.824e-8	5.141	1.401e-6	5.564
	1/64	4.923e-10	4.956	8.940e-10	4.982	6.046e-8	4.535
	1/128	1.547e-11	4.992	2.773e-11	5.011	1.897e-9	4.994

**Example 6.1.3.** We continue to solve some two-dimensional smooth test cases. The first example is the two-dimensional linear advection equation

$$\frac{\partial u}{\partial t} + \frac{\partial u}{\partial x_1} + \frac{\partial u}{\partial x_2} = 0, \quad \mathbf{x} \in [0, 1]^2$$

with periodic boundary condition and initial data  $u(\mathbf{x}, 0) = \sin(2\pi x_1) \sin(2\pi x_2)$ , and square entropy function  $U = u^2/2$ . The exact solution is  $u(\mathbf{x}, t) = u(x_1 - t, x_2 - t, 0)$ . We test the two-dimensional entropy stable nodal DG scheme on a hierarchy of unstructured triangular meshes. Errors and orders of convergence at  $t = 0.2$  are shown in Table 6.3. Once again we obtain optimal convergence.

**Example 6.1.4.** We consider the two-dimensional Burgers equation

$$\frac{\partial u}{\partial t} + \frac{\partial u^2}{\partial x_1} + \frac{\partial u^2}{\partial x_2} = 0, \quad \mathbf{x} \in [0, 1]^2$$

with periodic boundary condition and initial data  $u(\mathbf{x}, 0) = 0.5 \sin(2\pi(x_1 + x_2))$ , and square entropy function  $U = u^2/2$ . Exact solution follows from the solution of one-dimensional Burgers equation of  $\eta = x_1 + x_2$ . The entropy stable nodal DG scheme is evolved up to  $t = 0.05$  when the solution is still smooth. Errors and orders of convergence are displayed in Table 6.4. The results are similar to its one-dimensional counterpart. Convergence rate is below optimal.

**Table 6.4**

**Example 6.1.4:** accuracy test of the two-dimensional Burgers equation associated with initial data  $u(\mathbf{x}, 0) = 0.5 \sin(2\pi(x_1 + x_2))$  and square entropy function at  $t = 0.05$ .

$k$	$h$	$L^1$ error	Order	$L^2$ error	Order	$L^\infty$ error	Order
2	1/16	1.354e-3	–	3.275e-3	–	5.954e-2	–
	1/32	2.394e-4	2.500	7.046e-4	2.217	1.646e-2	1.855
	1/64	3.900e-5	2.618	1.406e-4	2.325	4.894e-3	1.750
	1/128	5.773e-6	2.756	2.456e-5	2.518	1.269e-3	1.948
	1/256	8.431e-7	2.776	4.109e-6	2.579	2.413e-4	2.394
3	1/16	1.890e-4	–	6.252e-4	–	1.968e-2	–
	1/32	2.482e-5	2.929	1.058e-4	2.563	4.859e-3	2.018
	1/64	2.327e-6	3.415	1.106e-5	3.258	7.311e-4	2.733
	1/128	2.065e-7	3.494	1.158e-6	3.255	1.195e-4	2.613
	1/256	1.898e-8	3.444	1.236e-7	3.229	1.299e-5	3.202
4	1/16	3.740e-5	–	1.454e-4	–	6.039e-3	–
	1/32	2.787e-6	3.746	1.427e-5	3.349	1.068e-3	2.500
	1/64	1.348e-7	4.370	7.651e-7	4.221	8.839e-5	3.595
	1/128	5.566e-9	4.598	3.722e-8	4.362	6.398e-6	3.788
	1/256	2.293e-10	4.602	1.696e-9	4.456	3.059e-7	4.387

**Table 6.5**

**Example 6.1.5:** accuracy test of isentropic vortex problem for two-dimensional Euler equations at  $t = 1$ . Results of the density are tabulated.

$k$	$h$	$L^1$ error	Order	$L^2$ error	Order	$L^\infty$ error	Order
2	10/8	2.299e-1	–	6.053e-2	–	8.735e-2	–
	10/16	4.204e-2	2.451	1.223e-2	2.307	2.957e-2	1.563
	10/32	6.598e-3	2.671	1.918e-3	2.673	5.162e-3	2.518
	10/64	9.330e-4	2.822	2.688e-4	2.835	1.064e-3	2.279
	10/128	1.273e-4	2.873	3.609e-5	2.897	1.717e-4	2.631
	10/256	1.652e-5	2.947	4.779e-6	2.917	2.280e-5	2.913
3	10/8	4.344e-2	–	1.160e-2	–	2.960e-2	–
	10/16	3.976e-3	3.450	1.155e-3	3.327	4.271e-3	2.793
	10/32	3.632e-4	3.453	1.030e-4	3.487	2.652e-4	4.009
	10/64	3.041e-5	3.578	8.538e-6	3.593	4.557e-5	2.541
	10/128	2.536e-6	3.584	7.148e-7	3.578	3.793e-6	3.587
	10/256	1.990e-7	3.672	5.670e-8	3.656	2.720e-7	3.802
4	10/8	7.754e-3	–	2.136e-3	–	8.836e-3	–
	10/16	3.941e-4	4.298	1.308e-4	4.030	5.582e-4	3.985
	10/32	1.546e-5	4.672	4.858e-6	4.750	2.549e-5	4.452
	10/64	5.620e-7	4.782	1.806e-7	4.749	1.680e-6	3.923
	10/128	2.020e-8	4.798	6.433e-9	4.812	8.998e-8	4.223

**Example 6.1.5 (Isentropic vortex).** The last smooth test case is the isentropic vortex advection problem for the two-dimensional Euler equations, taken from Shu [51]. The computational domains is  $[0, 10]^2$  and the initial condition is given by

$$w_1(\mathbf{x}, 0) = 1 - (x_2 - y_2)\phi(r), \quad w_2(\mathbf{x}, 0) = 1 + (x_1 - y_1)\phi(r)$$

$$T(\mathbf{x}, 0) = 1 - \frac{\gamma - 1}{2\gamma}\phi(r)^2, \quad \rho(\mathbf{x}, 0) = T^{\frac{1}{\gamma-1}}, \quad p(\mathbf{x}, 0) = T^{\frac{\gamma}{\gamma-1}}$$

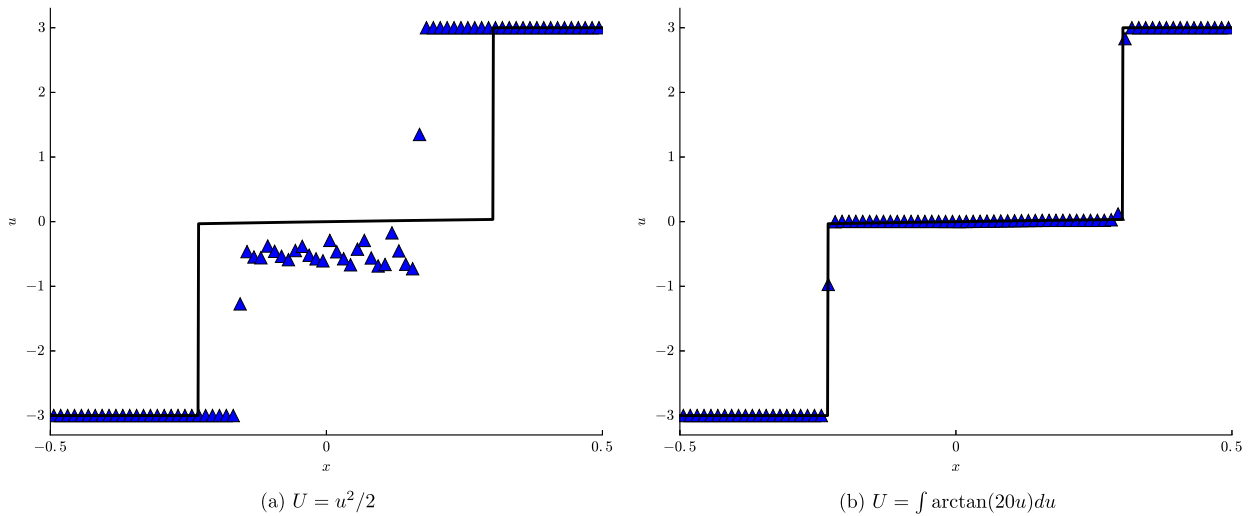
where  $(y_1, y_2)$  is the initial center of the vortex and

$$\phi(r) = \varepsilon e^{\alpha(1-r^2)}, \quad r = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2}$$

The parameters are  $\varepsilon = \frac{5}{2\pi}$ ,  $\alpha = 0.5$  and  $(y_1, y_2) = (5, 5)$ . The vortex will be advected in the diagonal direction and the exact solution is  $\mathbf{u}(\mathbf{x}, t) = \mathbf{u}(x_1 - t, x_2 - t, 0)$ . We use the exact solution to prescribe boundary conditions. Table 6.5 summarizes errors and orders of convergence of the density at  $t = 1$ . Here the convergence rate is also slightly below optimal, but better than Burgers equation. It is probably due to the linear behavior of exact solution.

## 6.2. Discontinuous tests

Discontinuous test problems are provided to illustrate shock-capturing capability. We will only show the numerical solutions of schemes with  $k = 2$  and CFL number 0.15. The bound-preserving limiter can be added to make the scheme robust. Specifically, for Euler equations, it is named the positivity-preserving limiter to prevent negative density or negative pressure. However, due to the lack of entropy stable non-oscillatory limiters for systems or in two dimensions (we only prove



**Fig. 6.1.** Example 6.2.1. Numerical solution of the Riemann problem of Buckley–Leverett equation at  $t = 1$  with the square entropy function and an ad hoc entropy function. Computational domain  $[-0.5, 0.5]$  is decomposed into  $N = 80$  cells. Bound-preserving limiter is used. The solid line represents the exact entropy solution and the triangle symbols are cell averages.

the entropy stability of one-dimensional TVD/TVB limiter for scalar equations), there are still spurious oscillations in some test results.

**Example 6.2.1.** We consider the following Riemann problem of Buckley–Leverett equation

$$\frac{\partial u}{\partial t} + \frac{\partial}{\partial x} \left( \frac{4u^2}{4u^2 + (1-u)^2} \right) = 0, \quad u(x, 0) = \begin{cases} -3 & \text{if } x < 0 \\ 3 & \text{if } x \geq 0 \end{cases}$$

The exact entropy solution contains two shock waves connected by a flat rarefaction wave that is close to 0. For such a nonconvex flux function, the choice of entropy function will affect the performance of numerical scheme substantially. We first test the scheme with square entropy function  $U = u^2/2$ . The computational domain is  $[-0.5, 0.5]$  and the end time  $t = 1$ . We also apply the bound-preserving limiter with  $\Omega = [-3, 3]$ . The numerical solution on 80 cells is plotted in the left panel of Fig. 6.1. Evidently it does not agree with the entropy solution. Then we try an ad hoc entropy function  $U = \int \arctan(20u) du$ . The entropy variable  $v = \arctan(20u)$ , which emphasizes the states near  $u = 0$ . In fact it can be viewed as a mollified version of the Kruzhkov's entropy function [42]  $U = |u|$ . The numerical solution with the same setting is depicted in the right panel of Fig. 6.1. The result is quite satisfactory thanks to the carefully chosen entropy function.

**Example 6.2.2 (Sod's shock tube).** It is a classical Riemann problem of one-dimensional Euler equations. The computational domain is  $[-0.5, 0.5]$  and the initial condition is

$$(\rho, w, p) = \begin{cases} (1, 0, 1) & \text{if } x < 0 \\ (0.125, 0, 0.1) & \text{if } x \geq 0 \end{cases}$$

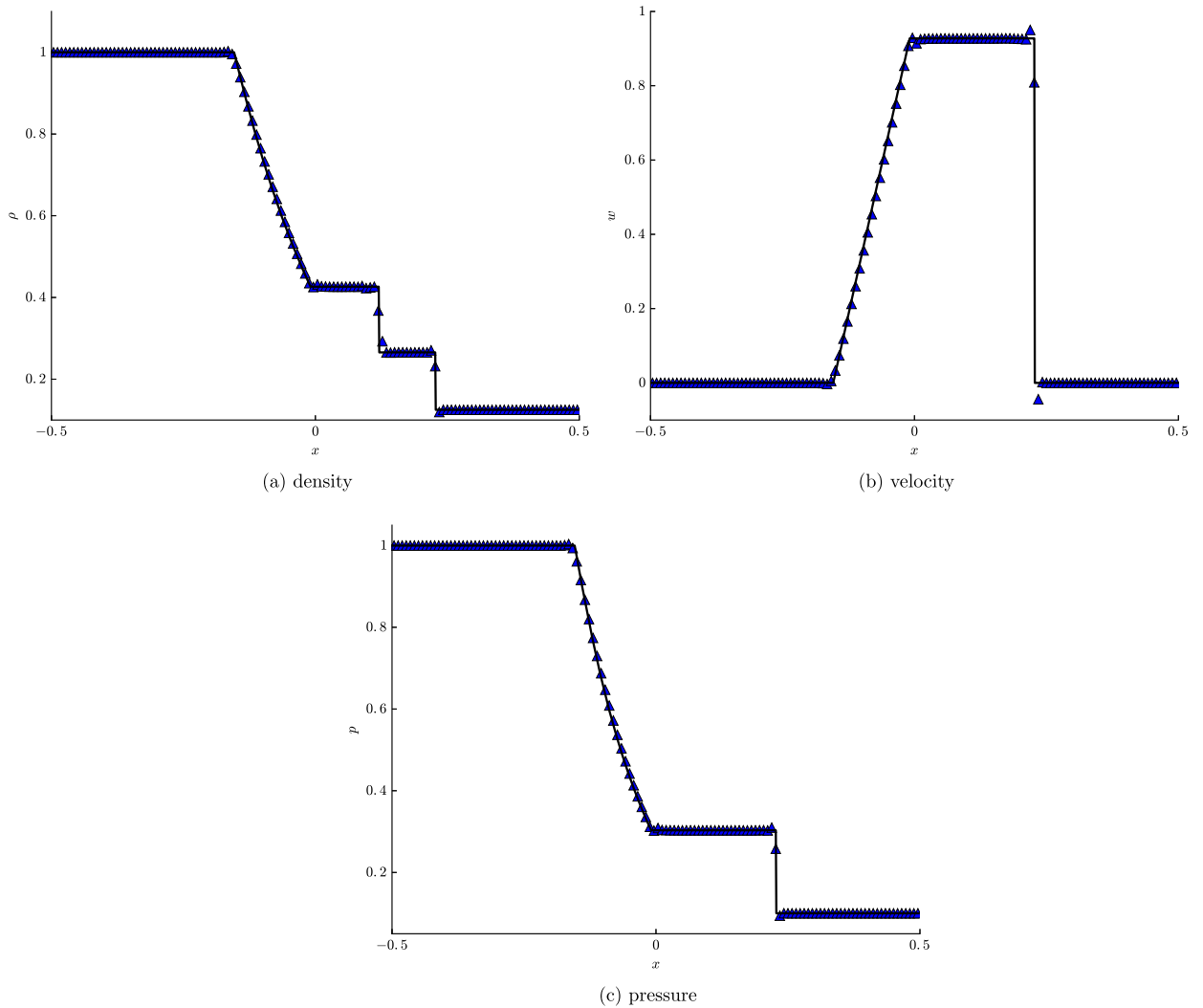
The exact solution contains a left rarefaction wave, a right shock wave and a middle contact discontinuity. The classical DG scheme tends to blow up due to emergence of negative density or negative pressure unless we apply positivity-preserving limiter or TVD/TVB limiter. The entropy stable nodal DG scheme, on the other hand, can be evolved without any limiter. Fig. 6.2 illustrates the profiles of density, velocity and pressure at  $t = 0.13$  with 130 cells. All waves are resolved correctly despite some slight oscillations at the right shock wave. Entropy stability contributes to a more robust scheme for this test problem.

**Example 6.2.3 (Sine-shock interaction).** This benchmark test problem of one-dimensional Euler equations was given by Shu and Osher in [53]. The solution has complicated structure in that it contains both strong and weak shock waves and highly oscillatory smooth waves. The computational domain is  $[-5, 5]$  and the initial condition is

$$(\rho, w, p) = \begin{cases} (3.857143, 2.629369, 10.3333) & \text{if } x < -4 \\ (1 + 0.2 \sin(5x), 0, 1) & \text{if } x \geq -4 \end{cases}$$

We compute the reference solution using a first order scheme on a very fine mesh with 80000 cells. Once again the classical DG scheme suffers from negative pressure or negative density, while the entropy stable nodal DG scheme works without any





**Fig. 6.2.** Example 6.2.2: Numerical solution of Sod's shock tube problem at  $t = 0.13$  with 130 cells. We do not apply any limiter. The solid line represents the exact entropy solution and the triangle symbols are cell averages.

limiter. The plots of density, velocity and pressure at  $t = 1.8$  with 150 cells are displayed in Fig. 6.3. The scheme performs well despite some minor oscillations.

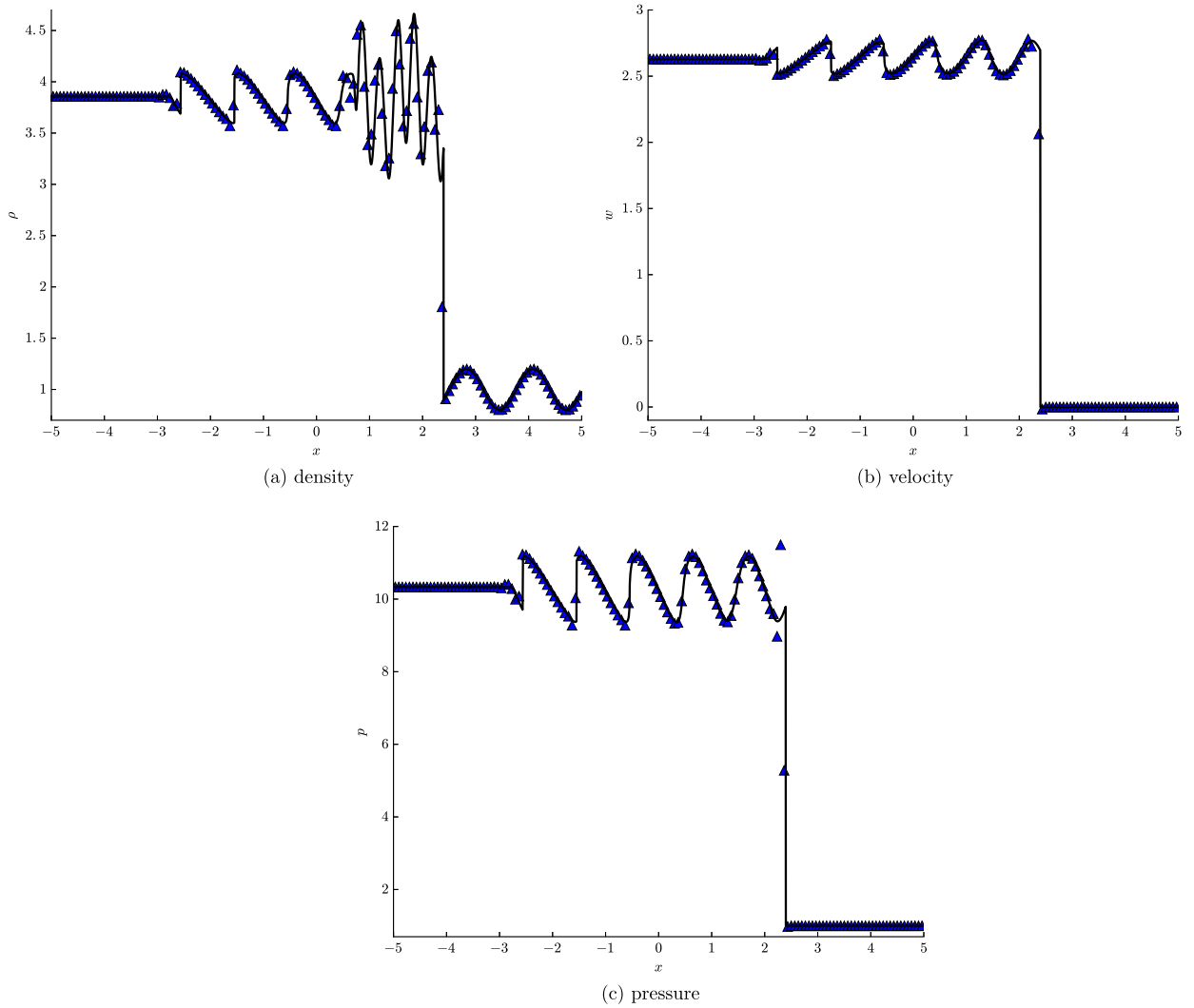
**Example 6.2.4** (Two-dimensional Riemann problem). We solve the Riemann problem of the two-dimensional Burgers equation

$$\frac{\partial u}{\partial t} + \frac{\partial u^2}{\partial x_1} + \frac{\partial u^2}{\partial x_2} = 0, \quad \mathbf{x} \in [0, 1]^2$$

subject to the initial condition

$$u(\mathbf{x}, 0) = \begin{cases} 0.25 & \text{if } x_1 < 0.5 \text{ and } x_2 < 0.5 \\ -0.1 & \text{if } x_1 < 0.5 \text{ and } x_2 \geq 0.5 \\ 0.4 & \text{if } x_1 \geq 0.5 \text{ and } x_2 < 0.5 \\ -0.5 & \text{if } x_1 \geq 0.5 \text{ and } x_2 \geq 0.5 \end{cases}$$

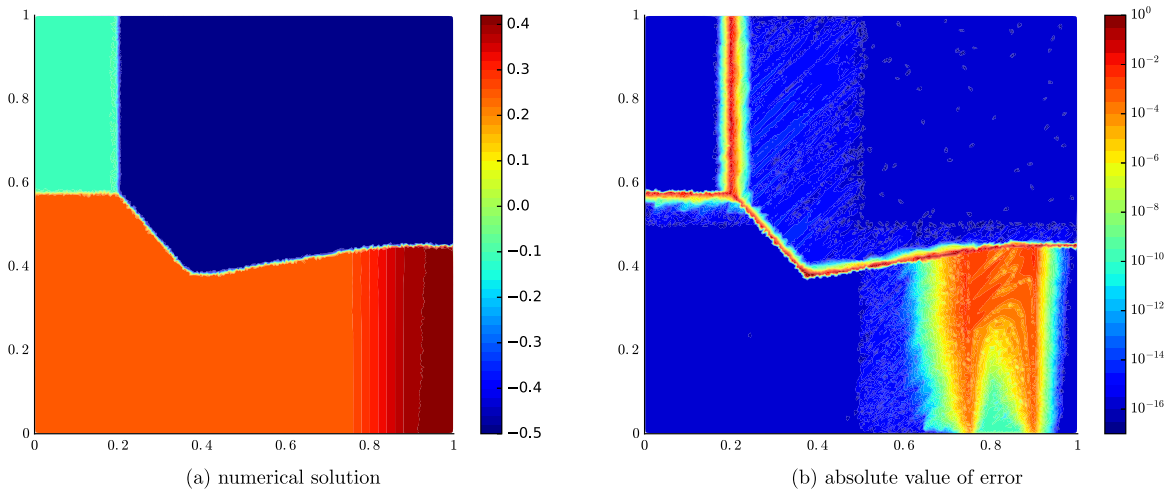
The exact solution for  $t > 0$  is as follows [61,26]



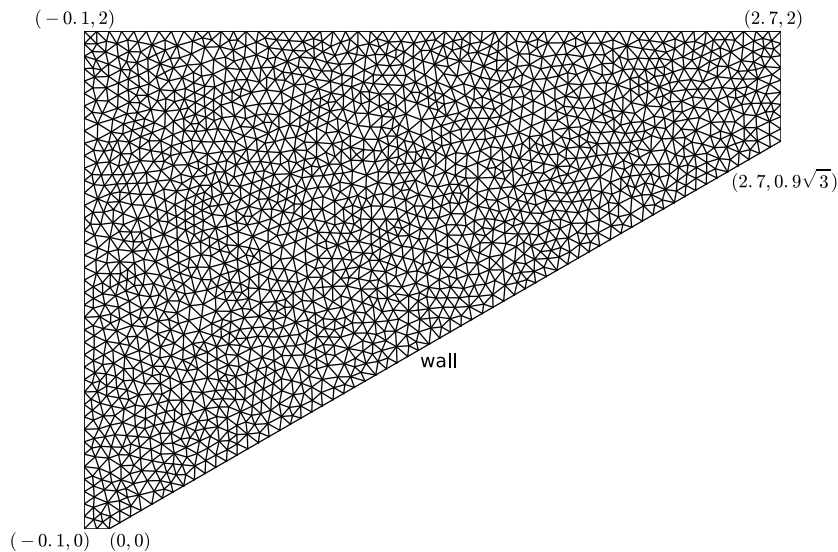
**Fig. 6.3.** Example 6.2.3: Numerical solution of sine-shock interaction test problem at  $t = 1.8$  with 150 cells. We do not apply any limiter. The solid line represents the reference solution computed with 80000 cells and the triangle symbols are cell averages.

$$u(\mathbf{x}, t) = \begin{cases} 0.25 & \text{if } x_1 < \frac{1}{2} - \frac{3t}{5} \text{ and } x_2 < \frac{1}{2} + \frac{t}{30} \\ -0.1 & \text{if } x_1 < \frac{1}{2} - \frac{3t}{5} \text{ and } x_2 \geq \frac{1}{2} + \frac{t}{30} \\ 0.25 & \text{if } \frac{1}{2} - \frac{3t}{5} \leq x_1 < \frac{1}{2} - \frac{t}{4} \text{ and } x_2 < \frac{-8x_1}{7} + \frac{15}{14} - \frac{15t}{28} \\ -0.5 & \text{if } \frac{1}{2} - \frac{3t}{5} \leq x_1 < \frac{1}{2} - \frac{t}{4} \text{ and } x_2 \geq \frac{-8x_1}{7} + \frac{15}{14} - \frac{15t}{28} \\ 0.25 & \text{if } \frac{1}{2} - \frac{t}{4} \leq x_1 < \frac{1}{2} + \frac{t}{2} \text{ and } x_2 < \frac{x_1}{6} + \frac{5}{12} - \frac{5t}{24} \\ -0.5 & \text{if } \frac{1}{2} - \frac{t}{4} \leq x_1 < \frac{1}{2} + \frac{t}{2} \text{ and } x_2 \geq \frac{x_1}{6} + \frac{5}{12} - \frac{5t}{24} \\ \frac{2x_1 - 1}{4t} & \text{if } \frac{1}{2} + \frac{t}{2} \leq x_1 < \frac{1}{2} + \frac{4t}{5} \text{ and } x_2 < x_1 - \frac{5}{18t}(x_1 + t - \frac{1}{2})^2 \\ -0.5 & \text{if } \frac{1}{2} + \frac{t}{2} \leq x_1 < \frac{1}{2} + \frac{4t}{5} \text{ and } x_2 \geq x_1 - \frac{5}{18t}(x_1 + t - \frac{1}{2})^2 \\ 0.4 & \text{if } x_1 \geq \frac{1}{2} + \frac{4t}{5} \text{ and } x_2 < \frac{1}{2} - \frac{t}{10} \\ -0.5 & \text{if } x_1 \geq \frac{1}{2} + \frac{4t}{5} \text{ and } x_2 \geq \frac{1}{2} - \frac{t}{10} \end{cases}$$

We choose the square entropy function  $U = u^2/2$  and run the entropy stable nodal DG scheme up to  $t = 0.5$  on a triangular mesh with  $h = 1/128$ . The bound-preserving limiter with  $\Omega = [-0.5, 0.4]$  is also imposed. The numerical result is shown in the left panel of Fig. 6.4, and the absolute value error is also plotted in the right panel where we use logarithmic scale and



**Fig. 6.4.** Example 6.2.4: Numerical solution and error of a Riemann problem of two-dimensional Burgers equation at  $t = 0.5$  on a mesh with  $h = 1/128$ . Entropy function is  $U = u^2/2$  and bound-preserving limiter is used. Error is shown in logarithmic scale.



**Fig. 6.5.** Example 6.2.5: illustration of the computational domain and the unstructured mesh with  $h = 1/20$ .

values less than  $10^{-16}$  are transformed to  $10^{-16}$ . The scheme successfully captures the correct profile. Error is very small unless near shock waves.

**Example 6.2.5 (Double Mach reflection).** This famous test problem of two-dimensional Euler equations was proposed by Woodward and Colella in [62] and has been intensively studied in the last few decades. It involves a Mach 10 shock which makes a  $60^\circ$  angle with a reflecting wall. The undisturbed air ahead of the shock has a density of 1.4 and a pressure of 1. Usually people solve the problem with rectangular computational domain and horizontal wall. Here we use the flexibility of the triangular mesh to consider the original physical problem with a horizontally moving shock and a wall inclined with a  $30^\circ$  angle (e.g. [58]). We illustrate the computational domain and the unstructured mesh with  $h = 1/20$  in Fig. 6.5. Initially the shock is positioned at  $x_1 = 0$ . Inflow/outflow boundary conditions are prescribed for the left and right boundaries, and at the top boundary the flow values are set to describe the exact motion of shock.

The entropy stable nodal DG scheme is implemented with positivity-preserving limiter and local Lax–Friedrichs flux. We do not use Godunov flux since the exact Riemann solver at element interface sometimes contains vacuum state. The plots of density and pressure at  $t = 0.2$  with mesh size  $h = 1/240$  are given in Fig. 6.6. Similar to the observations in [65], the solution is more oscillatory than results obtained via WENO scheme or DG scheme with TVD/TVB limiter, but it also catches some interesting features such as the small roll-ups due to Kelvin–Helmholtz instability, which indicates low numerical dissipation of our scheme.

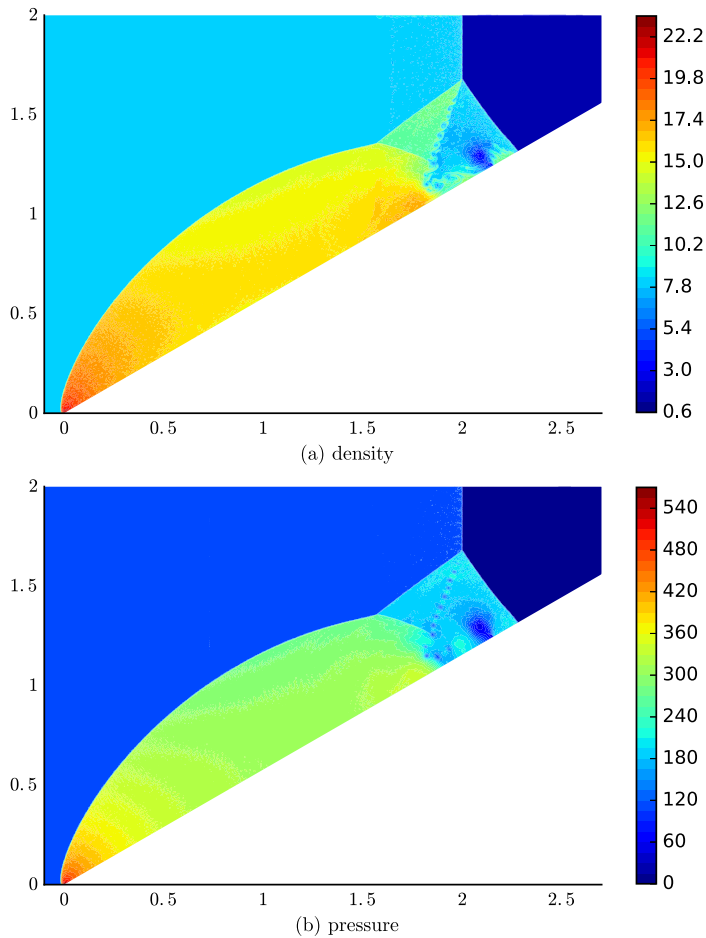


Fig. 6.6. Example 6.2.5: profiles of density and pressure at  $t = 0.2$  on a mesh with  $h = 1/240$ . 40 equally spaced contour levels are used for both plots.

**Example 6.2.6 (Shock diffraction).** A shock wave diffracting at a sharp corner is another popular test problem for two-dimensional Euler equations. In [12,67] the results of a Mach 5.09 shock diffracting at a  $90^\circ$  edge are presented. Here we would like to study a Mach 10 shock diffracting at a  $120^\circ$  degree [68]. The computational domain and the triangular mesh with  $h = 1/4$  are demonstrated in Fig. 6.7. The shock is initially located at  $x_1 = 3.4$  and  $6 \leq x_2 \leq 11$ , moving into undisturbed air with a density of 1.4 and a pressure of 1. Boundary conditions are inflow at the left/top boundary (in accordance with the exact shock motion), and outflow at the right/bottom boundary.

We still use positivity-preserving limiter and local Lax–Friedrichs interface flux. The contour plots of density and pressure at  $t = 0.9$  with mesh size  $h = 1/40$  are depicted in Fig. 6.8. The result is comparable to the one in [68] despite some oscillations and overshoots near the shock wave.

## 7. Concluding remarks

In this paper, we construct a (formally) high order entropy stable nodal DG scheme for systems of conservation laws. It does not require exact integration and can be stable with respect to an arbitrary entropy function. Therefore the limitations of the Jiang–Shu cell entropy inequality are circumvented. Our scheme has good flexibility in that it is compatible with

- (i) bound-preserving limiter and (one-dimensional scalar) TVD/TVB limiter.
- (ii) unstructured triangular meshes, and potentially simplex meshes and polygonal meshes.
- (iii) convection–diffusion equations through an LDG type approach.
- (iv) reflecting wall boundary conditions of Euler equations.

The entropy stability is guaranteed by three main ingredients: high order SBP operators, entropy conservative fluxes and entropy stable fluxes. The usual nodal DG scheme can be recovered if we take the entropy conservative flux to be the arithmetic mean. The major obstacle to exceeding one-dimensional framework is the construction of multi-dimensional SBP

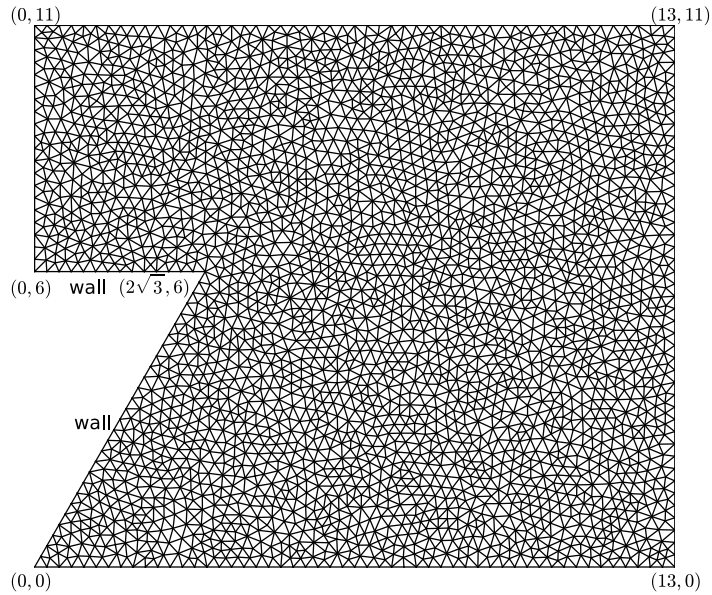


Fig. 6.7. Example 6.2.6: illustration of the computational domain and the unstructured mesh with  $h = 1/4$ .

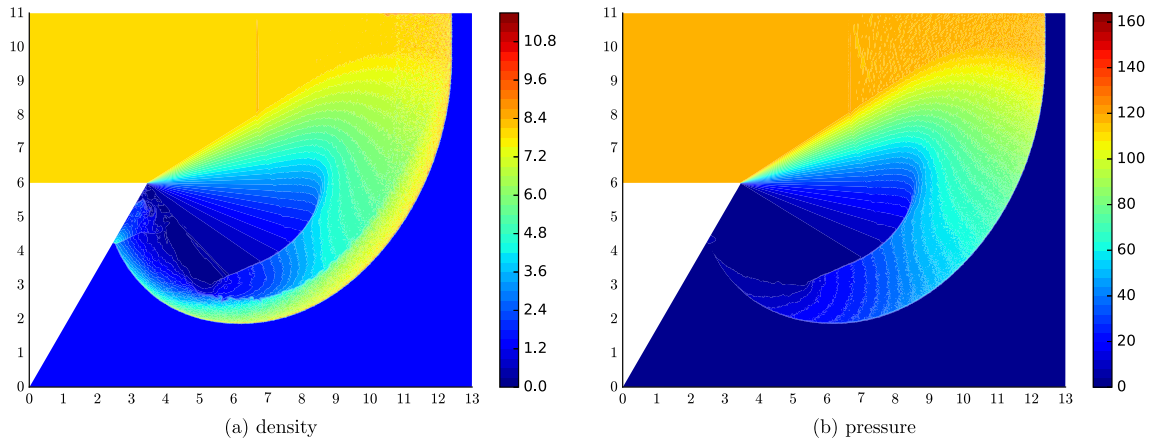


Fig. 6.8. Example 6.2.6: profiles of density and pressure at  $t = 0.9$  on a mesh with  $h = 1/20$ . 40 equally spaced contour levels are used for both plots.

operators with diagonal mass matrix and diagonal boundary matrices. We achieve this by finding a special quadrature rule and the formula of difference matrices (4.9).

We perform a large number of numerical tests whose results are comparable to existing schemes. In some cases the entropy stable nodal DG scheme shows better robustness and potential of computing physically correct solution if we choose a different entropy function. However, we should also point out some disadvantages of our scheme:

- (i) The quadrature rule is of degree  $2k - 1$ . We detect reduced orders of convergence in nonlinear smooth tests.
- (ii) For triangular meshes, the degree of freedom on each element is larger than the dimension of polynomial space, which is computationally more expensive than the classic DG scheme.
- (iii) The stabilization due to entropy dissipation at element interfaces is not enough. There are evident oscillations in some profiles of Euler equations.

The investigation of entropy stable oscillation control mechanism, such as entropy stable limiters for systems, and introduction of artificial viscosity, is a possible direction of our future study.

## Appendix A. Two-rarefaction approximation

For Euler equations, the solution of the Riemann problem consists of three characteristic waves. The left wave and right wave are either rarefaction fans or shocks, and the middle wave is a contact discontinuity. The pressure is continuous across

the contact discontinuity and thus constant in the middle region, denoted by  $p^*$ . We find the exact value of  $p^*$  by solving the following equation.

$$\varphi(p^*, p_L, \rho_L) + \varphi(p^*, p_R, \rho_R) + w_R - w_L = 0 \quad (\text{A.1})$$

where

$$\varphi(p^*, p, \rho) = \begin{cases} \varphi_r(p^*, p, \rho) = \frac{2a}{\gamma-1} \left( \left( \frac{p^*}{p} \right)^{(\gamma-1)/2\gamma} - 1 \right) & \text{if } p^* \leq p \text{ (rarefaction wave)} \\ \varphi_s(p^*, p, \rho) = \frac{p^* - p}{\sqrt{(\rho/(\gamma-1)p^* + (\gamma+1)p)/2}} & \text{if } p^* > p \text{ (shock wave)} \end{cases} \quad (\text{A.2})$$

and  $a = \sqrt{\gamma p / \rho}$  is the sound speed.  $\varphi$  is a continuous, strictly increasing and concave function of  $p^*$  (see [60]), so that we can use Newton–Raphson iteration to find the unique root. Once we have  $p^*$ , the leftmost and rightmost wave speeds are given by

$$\lambda_L = w_L - a_L q(p^*, p_L), \quad \lambda_R = w_R + a_R q(p^*, p_R) \quad (\text{A.3})$$

such that

$$q(p^*, p) = \begin{cases} 1 & \text{if } p^* \leq p \\ \sqrt{1 + \frac{\gamma+1}{2\gamma} \left( \frac{p^*}{p} - 1 \right)} & \text{if } p^* > p \end{cases} \quad (\text{A.4})$$

The following inequality is proved in [27].

**Theorem A.1.** If  $1 < \gamma \leq 5/3$ ,  $\varphi_s(p^*, p, \rho) \geq \varphi_r(p^*, p, \rho)$  for  $p^* > p$ .

**Proof.** Substitute  $x = (p^*/p)^{(\gamma-1)/2\gamma}$ . Then

$$\varphi_r(p^*, p, \rho) = \frac{2a}{\gamma-1}(x-1), \quad \varphi_s(p^*, p, \rho) = \frac{a}{\gamma} \frac{x^{2\gamma/(\gamma-1)} - 1}{\sqrt{(\gamma-1)/2\gamma + ((\gamma+1)/2\gamma)x^{2\gamma/(\gamma-1)}}}$$

Let  $\alpha = 2\gamma/(\gamma-1) \in [5, \infty)$ . We need to show that

$$\left( \frac{x^\alpha - 1}{x - 1} \right)^2 \geq \alpha + \alpha(\alpha-1)x^\alpha, \quad \text{for } x > 1$$

Rearranging the term yields

$$\left( \frac{x^\alpha - 1}{x - 1} - \frac{1}{2}\alpha(\alpha-1)(x-1) \right)^2 \geq \alpha^2 + \frac{1}{4}\alpha^2(\alpha-1)^2(x-1)^2 \quad (\text{A.5})$$

By Taylor's expansion

$$\frac{x^\alpha - 1}{x - 1} \geq \alpha + \frac{1}{2}\alpha(\alpha-1)(x-1) + \frac{1}{6}\alpha(\alpha-1)(\alpha-2)(x-1)^2$$

Inserting this inequality, we have

$$\begin{aligned} \left( \frac{x^\alpha - 1}{x - 1} - \frac{1}{2}\alpha(\alpha-1)(x-1) \right)^2 &\geq \left( \alpha + \frac{1}{6}\alpha(\alpha-1)(\alpha-2)(x-1)^2 \right)^2 \\ &\geq \alpha^2 + \frac{1}{3}\alpha^2(\alpha-1)(\alpha-2)(x-1)^2 \end{aligned}$$

Since  $\alpha \geq 5$ ,  $(\alpha-2)/3 \geq (\alpha-1)/4$  and so (A.5) is valid. We note that in most physical applications  $\gamma$  does fall into the range  $(1, 5/3]$  ( $5/3$  for monatomic gas and  $7/5$  for diatomic gas).  $\square$

Invoking Newton–Raphson iteration during all flux computations can be time-consuming. The two-rarefaction approximation assumes that the left wave and the right wave are both rarefaction waves, and provides an explicit formula of  $p^*$ ,  $\lambda_L$  and  $\lambda_R$ . Thanks to (A.1), the approximated wave speeds bound the true wave speeds. Then we can take these wave speeds to construct entropy stable HLL flux (or local Lax–Friedrichs flux).

**Theorem A.2.** The two-rarefaction approximation solves the equation

$$\varphi_r(p_{tr}^*, p_L, \rho_L) + \varphi_r(p_{tr}^*, p_R, \rho_R) + w_R - w_L = 0 \quad (\text{A.6})$$

The explicit solution is

$$p_{tr}^* = \left( \frac{a_L + a_R + (\gamma - 1)(w_L - w_R)/2}{a_L/p_L^{(\gamma-1)/2\gamma} + a_R/p_R^{(\gamma-1)/2\gamma}} \right)^{2\gamma/(\gamma-1)} \quad (A.7)$$

The approximated wave speeds are

$$\lambda_{tr,L} = w_L - a_L q(p_{tr}^*, p_L), \quad \lambda_{tr,R} = w_R + a_R q(p_{tr}^*, p_R) \quad (A.8)$$

Then  $q_{tr}^* \geq q^*$ ,  $\lambda_{tr,L} \leq \lambda_L$  and  $\lambda_{tr,R} \geq \lambda_R$ .

**Proof.** By Theorem A.1,  $\varphi_r \leq \varphi$  for all  $p > 0$ . As both  $\varphi$  and  $\varphi_r$  are strictly increasing,  $p_{tr}^* \geq p^*$ .  $q$  is also an increasing function of  $p^*$ . Hence  $\lambda_{tr,L} \leq \lambda_L$  and  $\lambda_{tr,R} \geq \lambda_R$ .  $\square$

The same argument also works for shallow water equations. We will omit the details, only giving the key inequality without proof. The exact Riemann solver reduces to the equation

$$\varphi(h^*, h_L) + \varphi(h^*, h_R) + w_R - w_L = 0 \quad (A.9)$$

where

$$\varphi(h^*, h) = \begin{cases} \varphi_r(h^*, h) = 2(\sqrt{gh^*} - \sqrt{gh}) & \text{if } h^* \leq h \\ \varphi_s(h^*, h) = (h^* - h)\sqrt{\frac{1}{2}g\frac{h^*+h}{h^*h}} & \text{if } h^* > h \end{cases} \quad (A.10)$$

When  $h^* > h$ , it is easy to prove that

$$\varphi_s(h^*, h) \geq \varphi_r(h^*, h) \quad (A.11)$$

Therefore two-rarefaction approximation will produce proper wave speeds.

## Appendix B. Bound-preserving limiter

We introduce the superscript  $n$  to represent the current time step. For the sake of simplicity we shall consider Euler forward time discretization and assume uniform grid. Let  $\lambda = \Delta t/\Delta x$  be the ratio of time step and mesh size. Let us start with the first order scheme

$$\mathbf{u}^{i,n+1} = H(\mathbf{u}^{i-1,n}, \mathbf{u}^{i,n}, \mathbf{u}^{i+1,n}; \lambda) = \mathbf{u}^{i,n} - \lambda(\widehat{\mathbf{f}}(\mathbf{u}^{i,n}, \mathbf{u}^{i+1,n}) - \widehat{\mathbf{f}}(\mathbf{u}^{i-1,n}, \mathbf{u}^{i,n})) \quad (B.1)$$

It is bound-preserving if  $\mathbf{u}^{i,n}, \mathbf{u}^{i,n}, \mathbf{u}^{i+1,n} \in \Omega$  implies  $\mathbf{u}^{i,n+1} \in \Omega$  provided that  $\lambda \leq \lambda_0$  for some  $\lambda_0 > 0$ . We will see that the upwind numerical fluxes in section 3.3 also correspond to bound-preserving first order schemes.

**Theorem B.1.** For scalar conservation laws, if  $\widehat{f}$  is monotone and Lipschitz continuous of both arguments, and  $\Omega = [m, M]$  for  $m, M \in \mathbb{R}$ , the corresponding first order scheme is bound-preserving.

**Proof.** Since  $\widehat{f}$  is monotone,  $H$  is non-decreasing with respect to  $u^{i,n-1}$  and  $u^{i,n+1}$ . Let  $L$  be the Lipschitz constant of  $\widehat{f}$ . Then  $H$  is also a non-decreasing function of  $u^{i,n}$  provided that  $\lambda \leq \frac{1}{2L}$ . Now if  $u^{i-1,n}, u^{i,n}, u^{i+1,n} \in [m, M]$ ,

$$u^{i,n+1} \geq H(m, m, m; \lambda) = m, \quad u^{i,n+1} \leq H(M, M, M; \lambda) = M$$

We see that  $H$  is bound-preserving with  $\lambda_0 = \frac{1}{2L}$ .  $\square$

**Theorem B.2.** For systems, if the exact Riemann solver is bound-preserving (e.g. no dry bed for shallow water equations or no vacuum for Euler equations), then the Godunov scheme and HLL scheme are bound-preserving.

**Proof.** First order schemes can be regarded as an averaging procedure of Riemann solvers when  $\lambda$  is small enough such that waves originated from different interfaces do not intersect. Since  $\Omega$  is a convex set, the Godunov scheme is bound-preserving. The HLL scheme is also bound-preserving due to the fact that the HLL Riemann solver is another average of the exact Riemann solver.  $\square$

High order schemes are generally not bound-preserving. However, we can still make sure that the cell average at next time step is in  $\Omega$ . The next theorem paves the path for high order bound-preserving limiter. It can be formulated in a more general manner, but we stay within the context of entropy stable nodal DG scheme.

**Theorem B.3.** For the entropy stable nodal DG scheme whose underlying first order scheme is bound preserving, if  $\mathbf{u}_j^{i,n} \in \Omega$  for each  $1 \leq i \leq N$  and  $0 \leq j \leq k$ , we have  $\bar{\mathbf{u}}^{i,n+1} \in \Omega$  under the CFL condition  $\lambda \leq \frac{\omega_0}{2} \lambda_0$ .



**Table C.1**

Quadrature rules for SBP operators with  $k = 1, 2, 3, 4$ . Exact values of abscissas and weights are available for  $k = 1, 2$ , so that we can use symbolic computation to derive SBP operators.

Orbit	Abscissas	Weight
$S_{111}$	$(0, \frac{1}{2} - \frac{\sqrt{3}}{6})$	$\frac{1}{12}$
(a) $k = 1, n_k = 6$		
Orbit	Abscissas	Weight
$S_3$	$\frac{1}{3}$	$\frac{9}{40}$
$S_{12}$	$\frac{1}{2}$	$\frac{1}{20}$
$S_{111}$	$(0, \frac{1}{2} - \frac{\sqrt{15}}{10})$	$\frac{1}{48}$
(b) $k = 2, n_k = 10$		
Orbit	Abscissas	Weight
$S_{111}$	$(0, 0.330009478207572)$	$0.0202282703414950$
$S_{111}$	$(0, 0.0694318442029737)$	$0.00754950743628280$
$S_{111}$	$(0.1870738791912771, 0.5841571139756569)$	$0.0555555555555556$
(c) $k = 3, n_k = 18$		
Orbit	Abscissas	Weight
$S_3$	$\frac{1}{3}$	$0.0455499555988567$
$S_{12}$	$\frac{1}{2}$	$0.00926854241697489$
$S_{12}$	$0.4384239524408185$	$0.0623683661448868$
$S_{12}$	$0.1394337314154536$	$0.0527146648104222$
$S_{111}$	$(0, 0.230765344947159)$	$0.0102652298402145$
$S_{111}$	$(0, 0.046910077030668)$	$0.00330065754050081$
(d) $k = 4, n_k = 22$		

**Proof.** Since the scheme is conservative,

$$\begin{aligned}
 \bar{\mathbf{u}}^{i,n+1} &= \bar{\mathbf{u}}^{i,n} - \lambda(\widehat{\mathbf{f}}(\mathbf{u}_k^{i,n}, \mathbf{u}_0^{i+1,n}) - \widehat{\mathbf{f}}(\mathbf{u}_k^{i-1,n}, \mathbf{u}_0^{i,n})) \\
 &= \sum_{j=0}^k \frac{\omega_j}{2} \mathbf{u}_j^{i,n} - \lambda(\widehat{\mathbf{f}}(\mathbf{u}_k^{i,n}, \mathbf{u}_0^{i+1,n}) - \widehat{\mathbf{f}}(\mathbf{u}_k^{i-1,n}, \mathbf{u}_0^{i,n})) \\
 &= \sum_{j=1}^{k-1} \frac{\omega_j}{2} \mathbf{u}_j^{i,n} + \frac{\omega_0}{2} H(\mathbf{u}_k^{i-1,n}, \mathbf{u}_0^{i,n}, \mathbf{u}_k^{i,n}, \frac{2\lambda}{\omega_0}) + \frac{\omega_0}{2} H(\mathbf{u}_0^{i,n}, \mathbf{u}_k^{i,n}, \mathbf{u}_0^{i+1,n}, \frac{2\lambda}{\omega_0})
 \end{aligned}$$

If  $\lambda \leq \frac{\omega_0}{2} \lambda_0$ , the last two terms are in  $\Omega$ . Then  $\bar{\mathbf{u}}^{i,n+1} \in \Omega$  as it is a convex combination of elements in  $\Omega$ .  $\square$

The bound-preserving limiter is a simple linear scaling procedure  $\tilde{\mathbf{u}}_j^{i,n} = \bar{\mathbf{u}}_j^{i,n} + \theta^{i,n}(\mathbf{u}_j^{i,n} - \bar{\mathbf{u}}_j^{i,n})$  to enforce  $\tilde{\mathbf{u}}_j^{i,n} \in \Omega$ . It can be enforced as long as  $\bar{\mathbf{u}}^{i,n} \in \Omega$ . Roughly speaking, for each  $0 \leq j \leq k$  we compute

$$\theta_j^{i,n} = \max\{s \in [0, 1] : \bar{\mathbf{u}}_j^{i,n} + s(\mathbf{u}_j^{i,n} - \bar{\mathbf{u}}_j^{i,n}) \in \Omega\}$$

Then we simply let  $\theta^{i,n} = \min_{0 \leq j \leq k} \theta_j^{i,n}$ . A combination of mathematical induction and [Theorem B.3](#) tell us that we can apply such limiter at each time step, leading to a robust scheme whose numerical solution never goes out of  $\Omega$ . For implementation details and the proof that bound-preserving limiter is genuinely high order accurate, one may check the papers by Zhang and Shu [\[66,67\]](#).

Finally, the magic of SSP time discretization enables us to go beyond Euler forward time stepping. In this paper, we use the third order SSP Runge–Kutta method. For an ODE system  $\mathbf{u}_t = L\mathbf{u}$ , the three stages at the  $n$ -th time step are

$$\mathbf{u}^{(1)} = \mathbf{u}^n + \Delta t L(\mathbf{u}^n) \quad (\text{B.2a})$$

$$\mathbf{u}^{(2)} = \frac{3}{4} \mathbf{u}^n + \frac{1}{4} (\mathbf{u}^{(1)} + \Delta t L(\mathbf{u}^{(1)})) \quad (\text{B.2b})$$

$$\mathbf{u}^{n+1} = \frac{1}{3} \mathbf{u}^n + \frac{2}{3} (\mathbf{u}^{(2)} + \Delta t L(\mathbf{u}^{(2)})) \quad (\text{B.2c})$$

Since it is a convex combination of Euler forward steps, all the previous analyses are still valid.

**Table C.2**  
Symmetry orbits on a triangle.

Orbit	Barycentric coordinates	# of points
$S_3(\frac{1}{3})$	$(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$	1
$S_{21}(\alpha)$	permutation of $(\alpha, \alpha, 1 - 2\alpha)$	3
$S_{111}(\alpha, \beta)$	permutation of $(\alpha, \beta, 1 - \alpha - \beta)$	6

## Appendix C. Quadrature rules on a triangle

The special quadrature rules designed for triangular SBP operators are listed in Table C.1. As indicated in [63], we divide the quadrature points into symmetry orbits. The orbit  $S_3$  only includes one point, the barycenter of the triangle. The three points in  $S_{21}$  are determined by a single abscissa, and the six points in  $S_{111}$  are determined by two abscissas. Table C.2 shows the idea of symmetry orbits.

## References

- [1] T.J. Barth, Numerical methods for gasdynamic systems on unstructured meshes, in: *An Introduction to Recent Developments in Theory and Numerics for Conservation Laws*, in: *Lecture Notes in Computational Science and Engineering*, vol. 5, Springer, 1999, pp. 195–285.
- [2] L. Beirão da Veiga, F. Brezzi, A. Cangiani, G. Manzini, L. Marini, A. Russo, Basic principles of virtual element methods, *Math. Models Methods Appl. Sci.* 23 (2013) 199–214.
- [3] L. Beirão da Veiga, F. Brezzi, L. Marini, A. Russo, The hitchhiker's guide to the virtual element method, *Math. Models Methods Appl. Sci.* 24 (2014) 1541–1573.
- [4] F. Bouchut, C. Bourdarias, B. Perthame, A MUSCL method satisfying all the numerical entropy inequalities, *Math. Comput.* 65 (1996) 1439–1461.
- [5] M.H. Carpenter, T.C. Fisher, E.J. Nielsen, S.H. Frankel, Entropy stable spectral collocation schemes for the Navier–Stokes equations: discontinuous interfaces, *SIAM J. Sci. Comput.* 36 (2014) B835–B867.
- [6] P. Castillo, B. Cockburn, I. Perugia, D. Schötzau, An a priori error analysis of the local discontinuous Galerkin method for elliptic problems, *SIAM J. Numer. Anal.* 38 (2000) 1676–1706.
- [7] 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.
- [8] B. Cockburn, S. Hou, C.-W. Shu, The Runge–Kutta local projection discontinuous Galerkin finite element method for conservation laws. IV: The multidimensional case, *Math. Comput.* 54 (1990) 545–581.
- [9] B. Cockburn, S.-Y. Lin, C.-W. Shu, TVB Runge–Kutta local projection discontinuous Galerkin finite element method for conservation laws. III: One-dimensional systems, *J. Comput. Phys.* 84 (1989) 90–113.
- [10] B. Cockburn, C.-W. Shu, TVB Runge–Kutta local projection discontinuous Galerkin finite element method for conservation laws. II: General framework, *Math. Comput.* 52 (1989) 411–435.
- [11] B. Cockburn, C.-W. Shu, The local discontinuous Galerkin method for time-dependent convection–diffusion systems, *SIAM J. Numer. Anal.* 35 (1998) 2440–2463.
- [12] B. Cockburn, C.-W. Shu, The Runge–Kutta discontinuous Galerkin method for conservation laws. V: multidimensional systems, *J. Comput. Phys.* 141 (1998) 199–224.
- [13] M.G. Crandall, A. Majda, Monotone difference approximations for scalar conservation laws, *Math. Comput.* 34 (1980) 1–21.
- [14] C.M. Dafermos, *Hyperbolic Conservation Laws in Continuum Physics*, Grundlehren der Mathematischen Wissenschaften, vol. 325, Springer, 2010.
- [15] T.C. Fisher, M.H. Carpenter, High-order entropy stable finite difference schemes for nonlinear conservation laws: finite domains, *J. Comput. Phys.* 252 (2013) 518–557.
- [16] T.C. Fisher, M.H. Carpenter, J. Nordström, N.K. Yamaleev, C. Swanson, Discretely conservative finite-difference formulations for nonlinear conservation laws in split form: theory and boundary conditions, *J. Comput. Phys.* 234 (2013) 353–375.
- [17] U.S. Fjordholm, S. Mishra, E. Tadmor, Arbitrarily high-order accurate entropy stable essentially nonoscillatory schemes for systems of conservation laws, *SIAM J. Numer. Anal.* 50 (2012) 544–573.
- [18] U.S. Fjordholm, S. Mishra, E. Tadmor, ENO reconstruction and ENO interpolation are stable, *Found. Comput. Math.* 13 (2013) 139–159.
- [19] G.J. Gassner, A skew-symmetric discontinuous Galerkin spectral element discretization and its relation to SBP–SAT finite difference methods, *SIAM J. Sci. Comput.* 35 (2013) A1233–A1253.
- [20] G.J. Gassner, A.R. Winters, D.A. Kopriva, A well balanced and entropy conservative discontinuous Galerkin spectral element method for the shallow water equations, *Appl. Math. Comput.* 272 (2016) 291–308.
- [21] C. Geuzaine, J.-F. Remacle, Gmsh: a 3-D finite element mesh generator with built-in pre- and post-processing facilities, *Int. J. Numer. Methods Eng.* 79 (2009) 1309–1331.
- [22] E. Godlewski, P.-A. Raviart, *Hyperbolic Systems of Conservation Laws*, Mathématiques & Applications, Ellipses, 1991.
- [23] E. Godlewski, P.-A. Raviart, *Numerical Approximation of Hyperbolic Systems of Conservation Laws*, Appl. Math. Sci., vol. 118, Springer, 2013.
- [24] S.K. Godunov, An interesting class of quasilinear systems, *Dokl. Akad. Nauk SSSR* 139 (1961) 521–523.
- [25] S. Gottlieb, C.-W. Shu, E. Tadmor, Strong stability-preserving high-order time discretization methods, *SIAM Rev.* 43 (2001) 89–112.
- [26] J.-L. Guermond, R. Pasquetti, B. Popov, Entropy viscosity method for nonlinear conservation laws, *J. Comput. Phys.* 230 (2011) 4248–4267.
- [27] J.-L. Guermond, B. Popov, Fast estimation from above of the maximum wave speed in the Riemann problem for the Euler equations, *J. Comput. Phys.* 321 (2016) 908–926.
- [28] B. Gustafsson, H.-O. Kreiss, J. Oliger, *Time Dependent Problems and Difference Methods*, John Wiley & Sons, 1995.
- [29] A. Harten, On the symmetric form of systems of conservation laws with entropy, *J. Comput. Phys.* 49 (1983) 151–164.
- [30] A. Harten, High resolution schemes for hyperbolic conservation laws, *J. Comput. Phys.* 49 (1983) 357–393.
- [31] A. Harten, J.M. Hyman, P.D. Lax, B. Keyfitz, On finite-difference approximations and entropy conditions for shocks, *Commun. Pure Appl. Math.* 29 (1976) 297–322.
- [32] A. Harten, P.D. Lax, B. Van Leer, On upstream differencing and Godunov-type schemes for hyperbolic conservation laws, *SIAM Rev.* 25 (1983) 35–61.
- [33] J.S. Hesthaven, T. Warburton, *Nodal Discontinuous Galerkin Methods: Algorithms, Analysis, and Applications*, Springer, 2007.
- [34] J.E. Hicken, D.C. Del Rey Fernández, D.W. Zingg, Multidimensional summation-by-parts operators: general theory and application to simplex elements, *SIAM J. Sci. Comput.* 38 (2016) A1935–A1958.

- [35] A. Hildebrand, S. Mishra, Entropy stable shock capturing space–time discontinuous Galerkin schemes for systems of conservation laws, *Numer. Math.* 126 (2014) 103–151.
- [36] S. Hou, X.-D. Liu, Solutions of multi-dimensional hyperbolic systems of conservation laws by square entropy condition satisfying discontinuous Galerkin method, *J. Sci. Comput.* 31 (2007) 127–151.
- [37] T.J. Hughes, L. Franca, M. Mallet, A new finite element formulation for computational fluid dynamics: I. Symmetric forms of the compressible Euler and Navier–Stokes equations and the second law of thermodynamics, *Comput. Methods Appl. Mech. Eng.* 54 (1986) 223–234.
- [38] F. Ismail, P.L. Roe, Affordable, entropy-consistent Euler flux functions II: entropy production at shocks, *J. Comput. Phys.* 228 (2009) 5410–5436.
- [39] G.S. Jiang, C.-W. Shu, On a cell entropy inequality for discontinuous Galerkin methods, *Math. Comput.* 62 (1994) 531–538.
- [40] G. Karniadakis, S. Sherwin, *Spectral/hp Element Methods for Computational Fluid Dynamics*, Oxford University Press, 2013.
- [41] D.A. Kopriva, G. Gassner, On the quadrature and weak form choices in collocation type discontinuous Galerkin spectral element methods, *J. Sci. Comput.* 44 (2010) 136–155.
- [42] S.N. Kruzhkov, First order quasilinear equations in several independent variables, *Mat. Sb.* 123 (1970) 228–255.
- [43] P. Lax, B. Wendroff, Systems of conservation laws, *Commun. Pure Appl. Math.* 13 (1960) 217–237.
- [44] P.G. Lefloch, J.-M. Mercier, C. Rohde, Fully discrete, entropy conservative schemes of arbitrary order, *SIAM J. Numer. Anal.* 40 (2002) 1968–1992.
- [45] M.S. Mock, Systems of conservation laws of mixed type, *J. Differ. Equ.* 37 (1980) 70–88.
- [46] S. Osher, Riemann solvers, the entropy condition, and difference approximations, *SIAM J. Numer. Anal.* 21 (1984) 217–235.
- [47] S. Osher, E. Tadmor, On the convergence of difference approximations to scalar conservation laws, *Math. Comput.* 50 (1988) 19–51.
- [48] E.Y. Panov, Uniqueness of the solution of the Cauchy problem for a first order quasilinear equation with one admissible strictly convex entropy, *Math. Notes* 55 (1994) 517–525.
- [49] D. Ray, P. Chandrashekar, U.S. Fjordholm, S. Mishra, Entropy stable scheme on two-dimensional unstructured grids for Euler equations, *Commun. Comput. Phys.* 19 (2016) 1111–1140.
- [50] C.-W. Shu, TVB uniformly high-order schemes for conservation laws, *Math. Comput.* 49 (1987) 105–121.
- [51] C.-W. Shu, Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws, in: B. Cockburn, C. Johnson, C.-W. Shu, E. Tadmor (Eds.), *Advanced Numerical Approximation of Nonlinear Hyperbolic Equations*, in: A. Quarteroni (Ed.), *Lecture Notes in Mathematics*, vol. 1697, Springer, Berlin, 1998, pp. 325–432.
- [52] C.-W. Shu, S. Osher, Efficient implementation of essentially non-oscillatory shock-capturing schemes, *J. Comput. Phys.* 77 (1988) 439–471.
- [53] C.-W. Shu, S. Osher, Efficient implementation of essentially non-oscillatory shock-capturing schemes, II, *J. Comput. Phys.* 83 (1989) 32–78.
- [54] M. Svärd, H. Özcan, Entropy-stable schemes for the Euler equations with far-field and wall boundary conditions, *J. Sci. Comput.* 58 (2014) 61–89.
- [55] E. Tadmor, Skew-selfadjoint form for systems of conservation laws, *J. Math. Anal. Appl.* 103 (1984) 428–442.
- [56] E. Tadmor, The numerical viscosity of entropy stable schemes for systems of conservation laws. I, *Math. Comput.* 49 (1987) 91–103.
- [57] E. Tadmor, Entropy stability theory for difference approximations of nonlinear conservation laws and related time-dependent problems, *Acta Numer.* 12 (2003) 451–512.
- [58] S. Tan, C.-W. Shu, Inverse Lax–Wendroff procedure for numerical boundary conditions of conservation laws, *J. Comput. Phys.* 229 (2010) 8144–8166.
- [59] E.F. Toro, *Shock-Capturing Methods for Free-Surface Shallow Flows*, John Wiley, 2001.
- [60] E.F. Toro, *Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction*, Springer, 2013.
- [61] D. Wagner, The Riemann problem in two space dimensions for a single conservation law, *SIAM J. Math. Anal.* 14 (1983) 534–559.
- [62] P. Woodward, P. Colella, The numerical simulation of two-dimensional fluid flow with strong shocks, *J. Comput. Phys.* 54 (1984) 115–173.
- [63] L. Zhang, T. Cui, H. Liu, A set of symmetric quadrature rules on triangles and tetrahedra, *J. Comput. Math.* 27 (2009) 89–96.
- [64] Q. Zhang, C.-W. Shu, Stability analysis and a priori error estimates of the third order explicit Runge–Kutta discontinuous Galerkin method for scalar conservation laws, *SIAM J. Numer. Anal.* 48 (2010) 1038–1063.
- [65] X. Zhang, On positivity-preserving high order discontinuous Galerkin schemes for compressible Navier–Stokes equations, *J. Comput. Phys.* 328 (2017) 301–343.
- [66] X. Zhang, C.-W. Shu, On maximum-principle-satisfying high order schemes for scalar conservation laws, *J. Comput. Phys.* 229 (2010) 3091–3120.
- [67] X. Zhang, 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.
- [68] X. Zhang, Y. Xia, C.-W. Shu, Maximum-principle-satisfying and positivity-preserving high order discontinuous Galerkin schemes for conservation laws on triangular meshes, *J. Sci. Comput.* 50 (2012) 29–62.