AIAA AVIATION Forum
June 15-19, 2020, VIRTUAL EVENT
AIAA AVIATION 2020 FORUM

10.2514/6.2020-3033

Entropy-stable Galerkin difference discretization on unstructured grids

Ge Yan*, Sharanjeet Kaur †, Jason E. Hicken ‡, and Jeffrey W. Banks§
Rensselaer Polytechnic Institute, Troy, New York-12180

The paper presents a high-order, entropy-stable discretization of the Euler equations based on the discontinuous Galerkin difference (DGD) method. DGD methods are similar to high-order finite-volume schemes, but they are posed in a Galerkin finite-element framework. We show how DGD discretizations can be constructed from diagonal-norm summation-by-parts (SBP) discretizations. This connection to SBP operators allows us to leverage existing literature on SBP methods to easily prove the entropy-stability of the DGD method. Numerical experiments are provided that verify the high-order accuracy of the DGD scheme as well as its stability.

I. Introduction

Many computational fluid dynamics (CFD) researchers have advocated for high-order discretizations as a means of increasing the efficiency of flow simulations. However, despite their potential for improved accuracy, high-order methods are often less robust, which has discouraged practitioners from adopting them. To address this issue, there has been increasing interest in entropy-stable discretizations, which offer a promising framework to improve the robustness of high-order methods. This work proposes and investigates a particular entropy-stable discretization that aims to be both robust and efficient.

Efficiency is an interplay between accuracy and computational cost. To achieve high-order accuracy, we consider a variant of the Galerkin difference (GD) method proposed by Banks and Hagstrom [1]. Unlike conventional finite-element methods, which increase accuracy by introducing additional degrees of freedom on each element, GD methods achieve high-order accuracy by including neighboring degrees of freedom. The GD method was recently extended to two space dimensions using a tensor-product construction in [2], and [3] presents robust approaches to handling complex geometries with GD methods.

While the original GD formulation was based on piecewise continuous basis functions, the method has been generalized to discontinuous basis functions and applied to symmetric hyperbolic systems [4]. In the current work, we also use discontinuous basis functions. Furthermore, to apply the GD method on unstructured simplex meshes, we follow the approach used in [5] and employ the idea of patch reconstruction.

In order to construct an entropy-stable discretization, we take advantage of the relationship between GD operators and summation-by-parts (SBP) operators. SBP operators are finite-difference operators that mimic integration by parts [6]. This property allows one to easily construct energy-stable SBP discretizations of linear partial-differential equations; see for example, the reviews [7] and [8]. Perhaps more importantly, at least for the CFD community, SBP operators can be combined with entropy-conservative flux funcitons [9–11] to construct high-order entropy-stable discretizations of the Euler and Navier-Stokes equations [12–14].

As mentioned above, entropy-stable discretizations are seen as a potential way of addressing the issue of robustness for high-order methods. While classic finite element methods satisfy a discrete entropy inequality [15, 16], this inequality relies on exact integration of the semi-linear weak forms. Exact integration of these forms is difficult, if not impossible, to achieve with the nonlinear flux functions present in the compressible Euler and Navier-Stokes equations. In contrast, the entropy stability of SBP discretizations does not rely on exact integration, which is a significant advantage, especially for under-resolved flows. In this study, we leverage the entropy-stable SBP discretization proposed in [17] to construct an entropy-stable GD discretization.

*Graduate Student, Department of Mechanical, Aerospace and Nuclear Engineering
†Graduate Student, Department of Mechanical, Aerospace and Nuclear Engineering
‡Associate Professor, Department of Mechanical, Aerospace and Nuclear Engineering
§Associate Professor, Department of Mathematical Sciences

In summary, the goal of this work is two-fold. First, we want to demonstrate that entropy-stable GD discretizations can be constructed on unstructured grids. Second, we hypothesize that GD discretizations may offer some advantages in terms of efficiency compared to previously reported, entropy-stable SBP discretizations; thus, we conduct preliminary studies that test this hypothesis.

The rest of the paper is organized as follows. Section II shows how summation-by-parts and Galerkin difference methods are related by applying them to the constant-coefficient linear advection equation in one space dimension. Once this relationship is established, we proceed to the entropy-stable GD discretization of the Euler equations in Section III. In Section III, we also show that the GD discretization conserves entropy by leveraging existing SBP entropy-conservation theory. Section IV presents the numerical experiments to verify the properties of our discretization and compare it with a previously developed element-based SBP discretization. Finally, we conclude with a summary in Section V.

II. Summation-by-parts and Galerkin differences

This section reviews summation-by-parts (SBP) and discontinuous Galkerkin difference (DGD) discretizations, with the aim of illustrating the relationship between these schemes. To this end, we apply both discretizations to the linear advection equation in one dimension, since the relationship between SBP and DGD is straightforward for this simple model problem.

Consider the linear advection equation in one spatial dimension on the periodic domain $\Omega = [0, 1]$:

$$\begin{aligned} &\frac{\partial \mathcal{U}}{\partial t} + \frac{\partial \mathcal{U}}{\partial x} = 0, \qquad \forall\, x \in [0, 1], \\ &\mathcal{U}(x = 0, t) = \mathcal{U}(x = 1, t). \end{aligned} \tag{1}$$

For well-posedness, this partial differential equation (PDE) must be supplemented with an initial condition; however, for the purpose of illustrating the relationship between the SBP and DGD discretizations of (1), it suffices to ignore initial conditions for the time being.

A. SBP discretization of the linear advection equation

In one dimension, there are several possible ways we could discretize (1) using SBP operators. For instance, we could use a uniform grid and apply the classical SBP operators proposed by Kreiss and Scherer [6]. However, element-based, generalized-SBP operators [18] will prove more useful in order to relate SBP and DGD discretizations. Collocation spectral-element operators using the Legendre-Gauss or Legendre-Gauss-Lobbato nodes are examples of generalized SBP operators [19].

Before continuing, we briefly review the properties of generalized SBP operators. Suppose we want a difference operator $\mathsf{D}_\xi \in \mathbb{R}^{n\times n}$ that approximates the first derivative operator at a set of n nodes, $\Xi = \{\xi_i\}_{i=1}^n$, on the reference interval $[-1, 1]$. We can specify the accuracy of the difference operator precisely by indicating a set of linearly independent functions for which the operator is exact. For instance, a degree p operator is exact for all polynomials in $\mathbb{P}_p([-1, 1])$, where $\mathbb{P}_p([-1, 1])$ denotes the space of polynomials of degree p on the reference interval. Thus, if D_ξ is a degree p difference operator for the first derivative, then

$$\sum_{j=1}^{n} (\mathsf{D}_\xi)_{ij} \mathcal{P}(\xi_j) = \frac{\partial \mathcal{P}}{\partial \xi}(\xi_i), \qquad \forall\, \xi_i \in \Xi,\ \mathcal{P} \in \mathbb{P}_p([-1, 1]). \tag{2}$$

The matrix D_ξ is a degree p SBP (first-derivative) operator, if it satisfies the accuracy requirement (2) as well as the following additional requirements:

- The operator has the factorization $\mathsf{D}_\xi = \mathsf{H}_\xi^{-1}\mathsf{Q}_\xi$, where H_ξ is a symmetric positive definite matrix.
- $\mathsf{Q}_\xi = \mathsf{S}_\xi + \frac{1}{2}\mathsf{E}_\xi$, where $\mathsf{S}_\xi^T = -\mathsf{S}_\xi$ is a skew symmetric matrix, and the symmetric matrix $\mathsf{E}_\xi = \mathsf{E}_\xi^T$ satisfies

$$\sum_{i=1}^{n_\kappa} \sum_{j=1}^{n_\kappa} \mathcal{P}(\xi_i)(\mathsf{E}_\xi)_{ij} \mathcal{Q}(\xi_j) = \mathcal{P}(1)\mathcal{Q}(1) - \mathcal{P}(-1)\mathcal{Q}(-1),$$

  for all polynomials $\mathcal{P}, \mathcal{Q} \in \mathbb{P}_r([-1, 1])$, where $r \geq p$

The significance of these requirements is that they ensure that D_ξ mimics integration by parts in a discrete sense. For further details see [18]. Note that H_ξ is analogous to the mass matrix.

To use the SBP operator D_ξ to (semi-) discretize the linear advection PDE, we first partition the domain into K elements of uniform size, that is, $\Omega = \bigcup_{\kappa=1}^{K} \Omega_\kappa$ where $\Omega_\kappa = [(\kappa-1)h, \kappa h]$ and $h = 1/K$. Next, we multiply (1) by a test function and integrate over each element to obtain the weak formulation of the PDE: find $\mathcal{U} \in W(\Omega)$ such that

$$\sum_{\kappa=1}^{K} \int_{\Omega_\kappa} \mathcal{V} \frac{\partial \mathcal{U}}{\partial t}\, dx - \sum_{\kappa=1}^{K} \int_{\Omega_\kappa} \frac{\partial \mathcal{V}}{\partial x} \mathcal{U}\, dx + \sum_{\kappa=1}^{K} \oint_{\Gamma_\kappa} \mathcal{V}\mathcal{U}\, n_x d\Gamma = 0, \qquad \forall \mathcal{V} \in W(\Omega), \tag{3}$$

where $W(\Omega)$ is an appropriate function space. Note that we have used integration-by-parts, on each element, to arrive at the weak form.

We will discretize the weak form using so-called diagonal-norm SBP operators, that is, operators for which H_ξ is diagonal. For these operators the nodes Ξ and diagonal entries of H_ξ define a degree $q \geq 2p-1$ quadrature rule [20]. Furthermore, the matrix Q_ξ of diagonal-norm operators is a degree $2p-1$ exact discretization of the stiffness matrix [20]. Finally, when the first and last nodes in Ξ_κ coincide with the domain boundary in one dimension, E_ξ can be constructed to be an exact representation of the boundary integral.

After transforming the integrals in (3) to the reference interval $[-1, 1]$, we can approximate the weak form using the SBP operator $\mathsf{D}_\xi = \mathsf{H}_\xi^{-1}\mathsf{Q}_\xi$ as follows:

$$\sum_{\kappa=1}^{K} h \boldsymbol{v}_\kappa^T \mathsf{H}_\xi \frac{d\boldsymbol{u}_\kappa}{dt} - \sum_{\kappa=1}^{K} \boldsymbol{v}_\kappa^T \mathsf{Q}_\xi^T \boldsymbol{u}_\kappa + \sum_{\kappa=1}^{K} \boldsymbol{v}_\kappa^T \mathsf{E}_\xi \boldsymbol{u}_\kappa = 0, \qquad \forall \boldsymbol{v}_\kappa \in \mathbb{R}^{n_\kappa},$$

where $\boldsymbol{u}_\kappa \in \mathbb{R}^n$ and $\boldsymbol{v}_\kappa \in \mathbb{R}^n$ are the restriction of the solution and test function to the nodes of element κ. The weak form can also be written in terms of global vectors and matrices as follows:

$$\boldsymbol{v}^T \mathsf{H} \frac{d\boldsymbol{u}}{dt} - \boldsymbol{v}^T \mathsf{Q}_x^T \boldsymbol{u} + \boldsymbol{v}^T \mathsf{E}_x \boldsymbol{u} = 0, \qquad \forall\, \boldsymbol{v} \in \mathbb{R}^{K n_\kappa}, \tag{4}$$

where $\boldsymbol{u}^T = [\boldsymbol{u}_1^T, \boldsymbol{u}_2^T, \ldots, \boldsymbol{u}_K^T]$ is the concatenation of all the element-level solutions — similarly for the test function $\boldsymbol{v}$ — and the global SBP operators are block diagonal matrices, with K blocks, defined by

$$\mathsf{H} = h \begin{bmatrix} \mathsf{H}_\xi & & & \\ & \mathsf{H}_\xi & & \\ & & \ddots & \\ & & & \mathsf{H}_\xi \end{bmatrix}, \quad \mathsf{Q}_x = \begin{bmatrix} \mathsf{Q}_\xi & & & \\ & \mathsf{Q}_\xi & & \\ & & \ddots & \\ & & & \mathsf{Q}_\xi \end{bmatrix}, \quad \text{and} \quad \mathsf{E}_x = \begin{bmatrix} \mathsf{E}_\xi & & & \\ & \mathsf{E}_\xi & & \\ & & \ddots & \\ & & & \mathsf{E}_\xi \end{bmatrix}. \tag{5}$$

Finally, we emphasize that the SBP discretization needs to be modified in practice to couple the elements. Inter-element coupling can be addressed using interior penalties, which are called simultaneous approximation terms (SATs) in the SBP literature [21, 22]. We will consider such penalties when discretizing the Euler equation, but ignore them for now in order to avoid complicating the comparison with the Galerkin difference method.

B. DGD discretization of the linear advection equation

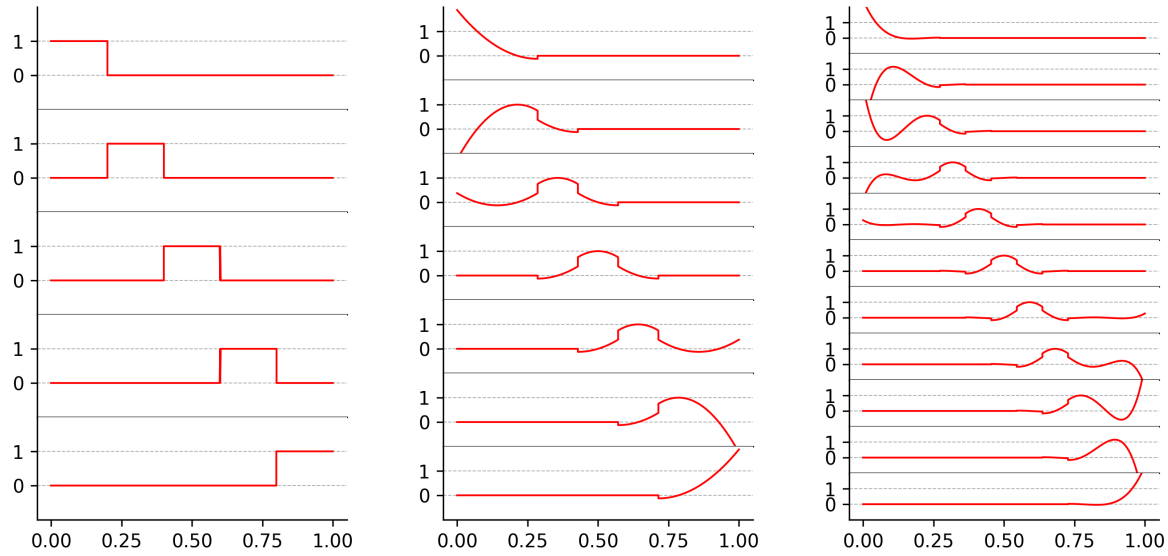
In this section we review discontinuous Galerkin difference (DGD) methods in the context of the linear advection equation. Note that high-order Galerkin difference (GD) methods were originally proposed in the context of continuous basis functions [1]. In the present work, we take advantage on the recent extension of GD methods to discontinuous basis functions [4, 5].

As before, consider the one dimensional domain $\Omega = [0, 1]$ discretized into a uniform mesh of K elements with subdomains $\Omega_\kappa = [(\kappa-1)h, \kappa h]$, $\kappa = 1, 2 \ldots, K$. The approximate DGD solution on element κ is given by

$$\tilde{\mathcal{U}}_\kappa(x) = \sum_{j \in S_\kappa} \tilde{u}_j \mathcal{P}_{\kappa,j}(x),$$

where $\tilde{u}_j$ denotes the discrete solution at the center of the jth element. The function $\mathcal{P}_{\kappa,j}(x) \in \mathbb{P}_p(\Omega_\kappa)$ is the pth order Lagrange interpolant that satisfies the interpolation conditions

$$\mathcal{P}_{\kappa,j}(x_i) = \begin{cases} 1, & x_i = x_j, \\ 0, & \text{otherwise}, \end{cases}$$



(a) $p = 0$ (b) $p = 2$ (c) $p = 4$

Fig. 1 Discontinuous Galerkin difference basis functions, for degrees $p = 0$, $p = 2$, and $p = 4$, on a uniform one dimensional grid. We have included more basis functions as p increases in order to clearly illustrate their behavior away from the boundary.

where $x_i = \left(i - \frac{1}{2}\right) h$ is the center of an element in the stencil S_κ of element κ. For an interior element, the stencil includes κ itself and q elements on either side of κ. Consequently, in one dimension on uniform meshes, it is reasonable to consider only even-order interpolants* of degree $p = 2q$. For elements near the boundary, we adopt a biased stencil with a sufficient number of neighbors to construct a degree p interpolant, although this is not the only possible boundary closure [1].

We can also define the DGD solution in terms of discontinuous basis functions. The basis function $\phi_j(x)$, corresponding to $\tilde{u}_j$, can be found by differentiating $\tilde{\mathcal{U}}_\kappa(x)$ with respect to $\tilde{u}_j$, since the interpolant is linear; doing so we find

$$\phi_j(x) = \begin{cases} \mathcal{P}_{\kappa,j}(x), & \text{if } x_\kappa < x < x_{\kappa+1}, \text{and } j \in S_\kappa, \\ 0, & \text{otherwise.} \end{cases}$$

Using this basis, we can express the DGD solution as a generalized linear model of the form

$$\tilde{\mathcal{U}}(x) = \sum_{j=1}^{K} \tilde{u}_j \phi_j(x).$$

Figure 1 illustrates the DGD basis functions of degree $p = 0$, $p = 2$, and $p = 4$. Note that for $p = 0$, the DGD basis is identical to a zeroth degree discontinuous Galerkin basis.

Using the definition of $\tilde{\mathcal{U}}(x)$ above, let us now consider the DGD discretization of the linear advection problem. Substituting the discrete solution into the weak form (3), and taking $\tilde{\mathcal{V}}(x) = \sum_{i=1}^{K} \tilde{v}_i \phi_i(x)$ as the test functions, we find

$$\tilde{\boldsymbol{v}}^T \tilde{\mathsf{H}} \frac{d\tilde{\boldsymbol{u}}}{dt} - \tilde{\boldsymbol{v}}^T \tilde{\mathsf{Q}}_x^T \tilde{\boldsymbol{u}} + \tilde{\boldsymbol{v}}^T \tilde{\mathsf{E}}_x \tilde{\boldsymbol{u}} = 0, \qquad \forall\, \tilde{\boldsymbol{v}} \in \mathbb{R}^K, \tag{6}$$

*The restriction to even-order interpolants is not justified in higher dimensions on unstructured grids, so we will consider both even- and odd-order interpolants in that setting.

where $\tilde{\boldsymbol{u}}^T = [\tilde{u}_1, \tilde{u}_2, \ldots, \tilde{u}_K]$ is the vector of solution coefficients, and the matrices in (6) are defined by

$$\tilde{\mathsf{H}}_{ij} = \sum_{\kappa=1}^{K} \int_{\Omega_\kappa} \phi_i \phi_j \, dx, \qquad (\tilde{\mathsf{Q}}_x^T)_{ij} = \sum_{\kappa=1}^{K} \int_{\Omega_\kappa} \frac{\partial \phi_i}{\partial x} \phi_j \, dx, \quad \text{and} \qquad (\tilde{\mathsf{E}}_x)_{ij} = \sum_{\kappa=1}^{K} \oint_{\Gamma_\kappa} \phi_i \phi_j \, n_x \, d\Gamma. \tag{7}$$

C. Relationship between the SBP and GD operators

In this section, we show how the GD matrices $\tilde{\mathsf{H}}$, $\tilde{\mathsf{Q}}_x$ and $\tilde{\mathsf{E}}_x$ are related to the corresponding SBP matrices. Consider the mass matrix, whose entries are

$$\tilde{\mathsf{H}}_{ij} = \sum_{\kappa=1}^{K} \int_{\Omega_\kappa} \phi_i \phi_j \, dx = \sum_{\kappa : i,j \in S_\kappa} \int_{\Omega_\kappa} \mathcal{P}_{\kappa,i} \mathcal{P}_{\kappa,j} \, dx,$$

where the latter sum is over all elements κ such that $i, j \in S_\kappa$. Next, since the product $\mathcal{P}_{\kappa,i}\mathcal{P}_{\kappa,j}$ is a degree $2p$ polynomial on element κ, we can replace its integral with a quadrature based on a $2p$-exact SBP norm[†]. To this end, let $\boldsymbol{p}_{\kappa,j} \in \mathbb{R}^n$ denote the Lagrange interpolant $\mathcal{P}_{\kappa,j}(x)$ evaluated at the SBP nodes Ξ of element κ in reference space. Then

$$\tilde{\mathsf{H}}_{ij} = \sum_{\kappa : i,j \in S_\kappa} h \boldsymbol{p}_{\kappa,i}^T \mathsf{H}_\xi \boldsymbol{p}_{\kappa,j},$$

and the global GD mass matrix can be written as

$$\tilde{\mathsf{H}} = \tilde{\mathsf{P}}^T \mathsf{H} \tilde{\mathsf{P}},$$

where, for example, the (rectangular) prolongation matrix $\tilde{\mathsf{P}} \in \mathbb{R}^{nK \times K}$ for a degree $p = 2$ scheme has the structure

$$\tilde{\mathsf{P}} = \begin{bmatrix} \boldsymbol{p}_{1,1} & \boldsymbol{p}_{1,2} & \boldsymbol{p}_{1,3} & & & & \\ \boldsymbol{p}_{2,1} & \boldsymbol{p}_{2,2} & \boldsymbol{p}_{2,3} & & & & \\ & \boldsymbol{p}_{3,2} & \boldsymbol{p}_{3,3} & \boldsymbol{p}_{3,4} & & & \\ & & \ddots & \ddots & \ddots & & \\ & & & \boldsymbol{p}_{K,K-2} & \boldsymbol{p}_{K,K-1} & \boldsymbol{p}_{K,K} \end{bmatrix}.$$

Similarly, the GD stiffness and boundary operators are given by

$$\tilde{\mathsf{Q}}_x = \tilde{\mathsf{P}}^T \mathsf{Q}_x \tilde{\mathsf{P}}, \quad \text{and} \qquad \tilde{\mathsf{E}}_x = \tilde{\mathsf{P}}^T \mathsf{E}_x \tilde{\mathsf{P}},$$

where Q_x and E_x were defined in (5).

In summary, the GD discretization can be obtained from the SBP discretization by making the substitutions

$$\boldsymbol{u} \leftarrow \tilde{\mathsf{P}} \tilde{\boldsymbol{u}}, \quad \text{and} \qquad \boldsymbol{v} \leftarrow \tilde{\mathsf{P}} \tilde{\boldsymbol{v}}.$$

Performing these substitutions we obtain

$$\tilde{\boldsymbol{v}}^T \left(\tilde{\mathsf{P}}^T \mathsf{H} \tilde{\mathsf{P}} \right) \frac{d\tilde{\boldsymbol{u}}}{dt} - \tilde{\boldsymbol{v}}^T \left(\tilde{\mathsf{P}}^T \mathsf{Q}_x^T \tilde{\mathsf{P}} \right) \tilde{\boldsymbol{u}} + \tilde{\boldsymbol{v}}^T \left(\tilde{\mathsf{P}}^T \mathsf{E}_x \tilde{\mathsf{P}} \right) \tilde{\boldsymbol{u}} = 0, \qquad \forall \, \tilde{\boldsymbol{v}} \in \mathbb{R}^K.$$

This strategy for constructing GD discretizations is useful, because we can leverage existing theory from the SBP literature in order to prove properties about the discretization. For example, we will adopt similar substitutions to construct and analyze the entropy-stable GD discretization of the Euler equations, which is presented in the next section.

III. Entropy-stable Galerkin difference discretization

This section presents our entropy-stable DGD discretization of the Euler equations. We begin with a brief review of the Euler equations and their corresponding entropy function. Subsequently, in order to leverage existing theory for SBP discretizations, we describe an entropy-conservative SBP discretization of the Euler equations that is suitable for unstructured (simplex) grids. We then show how we convert the SBP discretization into the GD discretization, and introduce suitable entropy-stable dissipation. Finally, we conclude the section by describing the patch definition and reconstruction operator $\tilde{\mathsf{P}}$ for the multidimensional case.

[†]Recall that a diagonal-norm SBP operator defines a $2p-1$ exact quadrature in general, so the requirement for a $2p$ exact quadrature may require more SBP nodes than usual.

A. The Euler equations and entropy

The strong form of the two-dimensional Euler equations is

$$\frac{\partial \boldsymbol{\mathcal{U}}}{\partial t} + \frac{\partial \boldsymbol{\mathcal{F}}_x}{\partial x} + \frac{\partial \boldsymbol{\mathcal{F}}_y}{\partial y} = \mathbf{0}, \qquad \forall\, \boldsymbol{x} \in \Omega, \tag{8}$$

where, for the time being, the state variables are the conservative variables, $\boldsymbol{\mathcal{U}} = [\rho, \rho u, \rho v, e]^T$, and the flux vectors are

$$\boldsymbol{\mathcal{F}}_x = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho u v \\ (e+p)u \end{bmatrix}, \qquad \boldsymbol{\mathcal{F}}_y = \begin{bmatrix} \rho v \\ \rho v u \\ \rho v^2 + p \\ (e+p)v \end{bmatrix}.$$

The pressure is defined by the callorically perfect ideal gas law as $p = (\gamma - 1)[e - \frac{\rho}{2}(u^2 + v^2)]$, with $\gamma = 1.4$.

In addition to conserving mass, momentum, and energy, one can show that, in the absence of shocks, the Euler equations conserve entropy [23, 24]:

$$\frac{d}{dt}\int_\Omega \mathcal{S}\, d\Omega + \int_{\partial\Omega} \left(\mathcal{G}_x n_x + \mathcal{G}_y n_y\right)\, d\Gamma = 0, \tag{9}$$

where $\mathcal{S} \equiv -\rho s/(\gamma - 1)$ is the (mathematical) entropy, $s = \ln(p/\rho^\gamma)$ is the thermodynamic entropy, and the entropy fluxes in the x and y directions are $\mathcal{G}_x = u\mathcal{S}$ and $\mathcal{G}_y = v\mathcal{S}$, respectively. More generally, when shocks are present, the unique weak solution to the Euler equations dissipates (mathematical) entropy and the equality in (9) becomes an inequality.

$$\frac{d}{dt}\int_\Omega \mathcal{S}\, d\Omega + \int_{\partial\Omega} \left(\mathcal{G}_x n_x + \mathcal{G}_y n_y\right)\, d\Gamma \leq 0, \tag{10}$$

A discretization that mimics (9) is called entropy conservative, while a discretization that mimics (10) is called entropy stable. It is advantageous, for both physical and mathematical reasons, to use discretizations that are entropy stable. Physically, a discretization should respect (10) for consistency with the second-law of thermodynamics. Mathematically, a discretization that mimics the inequality (10) gains a mechanism for nonlinear stability [25]. Note that entropy-conservative schemes are valuable in this context because they provide a baseline scheme that can be readily made entropy-stable through the introduction of an appropriate numerical dissipation.

B. Entropy-conservative SBP discretization

Recently, Fisher [12] and his collaborators [14, 26], demonstrated how SBP operators can be combined with particular flux functions to achieve high-order, entropy conservative discretizations of the Euler equations. These initial contributions focused on tensor-product finite-difference and spectral element methods, but the approach was subsequently extended to more general elements [17, 27, 28]. In this work, we adopt the entropy conservative discretization presented in Crean *et al.* [17], which is listed below in its weak formulation.

$$\begin{aligned} \boldsymbol{v}_\kappa^T \bar{\mathsf{H}}_\kappa \frac{d\boldsymbol{u}_\kappa}{dt} + \boldsymbol{v}_\kappa^T \left[\bar{\mathsf{S}}_x \circ \mathsf{F}_x(\boldsymbol{u}_\kappa, \boldsymbol{u}_\kappa)\right]\mathbf{1} + \boldsymbol{v}_\kappa^T \left[\bar{\mathsf{S}}_y \circ \mathsf{F}_y(\boldsymbol{u}_\kappa, \boldsymbol{u}_\kappa)\right]\mathbf{1} \\ + \frac{1}{2}\sum_{\gamma \subset \partial\Omega_\kappa} \boldsymbol{v}_\kappa^T \left[\bar{\mathsf{E}}_n^{\kappa\nu} \circ \mathsf{F}_n(\boldsymbol{u}_\kappa, \boldsymbol{u}_\nu)\right]\mathbf{1} = \mathbf{0}, \qquad \forall\, \boldsymbol{v}_\kappa \in \mathbb{R}^{4n_k}. \end{aligned} \tag{11}$$

Equation (11) introduces a number of new notations that are defined below; however, these definitions are necessarily brief, so the interested reader is directed to [17] for additional details.

- As before, $\boldsymbol{u}_\kappa$ and $\boldsymbol{v}_\kappa$ are the solution and test vectors, respectively, restricted to element κ. Since there are four state variables at each node, these vectors have $4n_\kappa$ components.
- $\bar{\mathsf{H}}_\kappa = \mathsf{H}_\kappa \otimes \mathsf{I}_4$, where H_κ is the SBP norm matrix of element κ *in physical space*, I_4 is the 4×4 identity, and $\otimes$ denotes the Kronecker product.
- $\bar{\mathsf{S}}_x = \mathsf{S}_x \otimes \mathsf{I}_4$ and $\bar{\mathsf{S}}_y = \mathsf{S}_y \otimes \mathsf{I}_4$, where S_x and S_y are the skew-symmetric parts of Q_x and Q_y, respectively, on element κ *in physical space*.

- $\mathsf{F}_x(\boldsymbol{u}_\kappa, \boldsymbol{u}_\kappa)$ is a $4n_\kappa \times 4n_\kappa$ block matrix. The (i,j)th block is the 4×4 diagonal matrix

$$\text{diag}\left[\mathcal{F}_x^\star(\boldsymbol{u}_{\kappa,i}, \boldsymbol{u}_{\kappa,j})\right]$$

  where $\mathcal{F}_x^\star$ is a two-point, entropy-conservative flux function, and $\boldsymbol{u}_{\kappa,i}$ holds the conservative variables at node i. In this work we use the Ismail-Roe flux function [11] for $\mathcal{F}_x^\star$. A similar definition applies for $\mathsf{F}_y(\boldsymbol{u}_\kappa, \boldsymbol{u}_\kappa)$.
- $\bar{\mathsf{E}}_n^{\kappa\nu} = \mathsf{E}_n^{\kappa\nu} \otimes \mathsf{I}_4$ where $\mathsf{E}_n^{\kappa\nu} \equiv \mathsf{R}_{\gamma\kappa}^T \mathsf{B}_\gamma \mathsf{R}_{\gamma\nu} = -(\mathsf{E}_n^{\nu\kappa})^T$ and ν indicates the index of an element adjacent to κ on face γ. In this work we use SBP operators for which $\mathsf{R}_{\gamma\kappa}$ is a simple matrix of zeros and ones that extracts the degrees of freedom from $\boldsymbol{u}_\kappa$ that coincide with the quadrature nodes on face γ. Similarly, $\mathsf{R}_{\gamma\nu}$ extracts the degrees of freedom from the solution on the adjacent element ν, $\boldsymbol{u}_\nu$, that reside on the nodes of γ. Finally, B_γ is a diagonal matrix holding the (strictly positive) quadrature weights for face γ.
- $\mathsf{F}_n(\boldsymbol{u}_\kappa, \boldsymbol{u}_\nu)$ is similar to $\mathsf{F}_x(\boldsymbol{u}_\kappa, \boldsymbol{u}_\kappa)$, except it is based on the entropy-conservative flux function evaluated between pairs of nodes on the two elements κ and ν. Because of the particular form of $\mathsf{R}_{\gamma\kappa}$ used in this work, the flux function only needs to be computed for nodes of elements κ and ν that are coincident along the face γ.

In [17], it was shown that (11) satisfies semi-discrete conservation of entropy, which takes the following form on a periodic domain:

$$\sum_{\kappa=1}^{K} \mathbf{1}^T \mathsf{H}_\kappa \frac{d\boldsymbol{s}_\kappa}{dt} = 0, \tag{12}$$

where $\boldsymbol{s}_\kappa \in \mathbb{R}^{n_\kappa}$ is the entropy $\mathcal{S}$ evaluated at the SBP nodes of element κ. We now set out to construct a DGD discretization that also conserves entropy, and thus inherits a form of nonlinear stability.

C. Entropy-conservative GD discretization

As described earlier for the linear advection equation, we can construct a GD scheme from an SBP scheme by setting $\boldsymbol{u} = \tilde{\mathsf{P}}\tilde{\boldsymbol{u}}$ and $\boldsymbol{v} = \tilde{\mathsf{P}}\tilde{\boldsymbol{v}}$. While this strategy can be applied to any linear PDE, the approach must be modified somewhat for the Euler equations, if we are to retain entropy conservation.

For each element κ, let S_κ denote the set of neighboring elements used to interpolate to κ. Furthermore, let $\tilde{\mathsf{P}}_\kappa$ be the degree p exact prolongation matrix that interpolates from the barycenters of the elements in S_κ to the SBP nodes of element κ. The procedure for selecting elements for S_κ and constructing $\tilde{\mathsf{P}}_\kappa$ is described in Section III.E.

The key to constructing an entropy-stable DGD scheme is to ensure that the conservative variables are consistent with the entropy variables at the nodes of the SBP discretization (11), where the entropy variables are defined by

$$\mathcal{W}(\mathcal{U}) \equiv \frac{\partial \mathcal{S}}{\partial \mathcal{U}} = \left[\tfrac{\gamma-s}{\gamma-1} - \tfrac{1}{2}\tfrac{\rho}{p}(u^2+v^2), \quad \tfrac{\rho u}{p}, \quad \tfrac{\rho v}{p}, \quad -\tfrac{\rho}{p}\right]^T.$$

Note that the entropy variable mapping is one-to-one [10], so we can equivalently write $\mathcal{U}(\mathcal{W})$. Returning to the construction of the DGD scheme, there are two ways to achieve consistency between the conservative and entropy variables at the level of the SBP nodes.

1) Rather than interpolating the conservative variables from the DGD degrees of freedom to the SBP nodes using $\tilde{\mathsf{P}}_\kappa$, we instead convert to the entropy variables and interpolate these. Thus, if $\tilde{\boldsymbol{u}}$ is the vector of conservative variables at the barycenter of the elements, then the conservative variables at the nodes of element κ in the SBP discretization (11) are defined by

$$\boldsymbol{u}_\kappa(\tilde{\boldsymbol{u}}) = \boldsymbol{u}(\tilde{\mathsf{P}}_\kappa \tilde{\boldsymbol{w}}(\tilde{\boldsymbol{u}})),$$

   where $\tilde{\boldsymbol{w}}(\tilde{\boldsymbol{u}})$ is the vector of entropy variables at the element barycenters. In words, the interpolation proceeds by converting from conservative variables to entropy variables at the GD degrees of freedom, interpolating the entropy variables to the nodes of κ using $\tilde{\mathsf{P}}_\kappa$, and then converting back to conservative variables on κ. A similar approach was used in [29, 30] for staggered-grid SBP discretizations and in [28] for entropy-stable DG methods.
2) Alternatively, we can simply adopt the entropy variables as the state variables. That is, the vector of unknowns that we solve for is $\tilde{\boldsymbol{w}}$. In this approach, the conservative variables at the SBP nodes of element κ are found using the mapping

$$\boldsymbol{u}_\kappa(\tilde{\boldsymbol{w}}) = \boldsymbol{u}(\tilde{\mathsf{P}}_\kappa \tilde{\boldsymbol{w}}).$$

We have chosen the second approach for this work. Using the entropy variables eliminates some computational overhead and was more straightforward to implement with `mfem` [31], the finite-element library used to implement

the discretizations. On the other hand, the second approach has implications for fully discrete entropy-stability and conservation, which are the focus of ongoing research.

Let $\tilde{\boldsymbol{u}}_\kappa = \boldsymbol{u}_\kappa(\tilde{\boldsymbol{w}})$ and $\tilde{\boldsymbol{v}}_\kappa = \tilde{\mathsf{P}}_\kappa \tilde{\boldsymbol{v}}$ be shorthand for the DGD trial and test functions mapped to element κ of the SBP discretization. Then the entropy conservative DGD discretization is given by

$$\begin{aligned}\tilde{\boldsymbol{v}}_\kappa^T \bar{\mathsf{H}}_\kappa \frac{d\tilde{\boldsymbol{u}}_\kappa}{dt} + \tilde{\boldsymbol{v}}_\kappa^T \left[\bar{\mathsf{S}}_x \circ \mathsf{F}_x(\tilde{\boldsymbol{u}}_\kappa, \tilde{\boldsymbol{u}}_\kappa)\right] \mathbf{1} + \tilde{\boldsymbol{v}}_\kappa^T \left[\bar{\mathsf{S}}_y \circ \mathsf{F}_y(\tilde{\boldsymbol{u}}_\kappa, \tilde{\boldsymbol{u}}_\kappa)\right] \mathbf{1} \\ + \frac{1}{2} \sum_{\gamma \subset \partial\Omega_\kappa} \tilde{\boldsymbol{v}}_\kappa^T \left[\bar{\mathsf{E}}_n^{\kappa\nu} \circ \mathsf{F}_n(\tilde{\boldsymbol{u}}_\kappa, \tilde{\boldsymbol{u}}_\nu)\right] \mathbf{1} = \mathbf{0}, \qquad \forall\, \tilde{\boldsymbol{v}} \in \mathbb{R}^{4K}.\end{aligned} \tag{13}$$

We can easily show that (13) is entropy conservative on a periodic domain by leveraging the entropy-conservativeness of the underlying SBP discretization. For example, if we replace the generic test function $\tilde{\boldsymbol{v}}$ with the entropy variables at the element barycenters, $\tilde{\boldsymbol{w}}$, then we get the following result:

$$\begin{aligned}&\tilde{\boldsymbol{w}}_\kappa^T \bar{\mathsf{H}}_\kappa \frac{d\tilde{\boldsymbol{u}}_\kappa}{dt} + \tilde{\boldsymbol{w}}_\kappa^T \left[\bar{\mathsf{S}}_x \circ \mathsf{F}_x(\tilde{\boldsymbol{u}}_\kappa, \tilde{\boldsymbol{u}}_\kappa)\right] \mathbf{1} + \tilde{\boldsymbol{w}}_\kappa^T \left[\bar{\mathsf{S}}_y \circ \mathsf{F}_y(\tilde{\boldsymbol{u}}_\kappa, \tilde{\boldsymbol{u}}_\kappa)\right] \mathbf{1} \\ &\quad + \frac{1}{2} \sum_{\gamma \subset \partial\Omega_\kappa} \tilde{\boldsymbol{w}}_\kappa^T \left[\bar{\mathsf{E}}_n^{\kappa\nu} \circ \mathsf{F}_n(\tilde{\boldsymbol{u}}_\kappa, \tilde{\boldsymbol{u}}_\nu)\right] \mathbf{1} \\ &\quad = \sum_{\kappa=1}^{K} \mathbf{1}^T \mathsf{H}_\kappa \frac{d\tilde{\boldsymbol{s}}_\kappa}{dt} \\ &\quad = \mathbf{0},\end{aligned}$$

where $\tilde{\boldsymbol{s}}_\kappa = \boldsymbol{s}(\tilde{\mathsf{P}}_\kappa \tilde{\boldsymbol{w}})$ is the entropy $\mathcal{S}$ at the SBP nodes of element κ based on the interpolated entropy variables. To arrive at the above result, we used the consistency between the conservative and entropy variables at the SBP nodes — that is, $\tilde{\boldsymbol{u}}_\kappa = \boldsymbol{u}(\tilde{\boldsymbol{w}}_\kappa)$ — and the equality (12).

D. Entropy-stable dissipation

When discontinuities are present in the flow, the mathematical entropy $\mathcal{S}$ should decrease across shocks and the relevant global entropy condition is given by (10). Even for isentropic flows, adding entropy dissipation to the discretization is often necessary to ensure optimal rates of convergence in the solution error. To address both of these motivations, entropy-stable dissipation is added to the discretizations considered in this work.

First, consider the baseline SBP discretization (11). On each element κ, we introduce the following Lax-Friedichs-type dissipation based on the jump in entropy-variables along the edge of each face.

$$\sum_{\gamma \in \partial\Omega_k} (\bar{\mathsf{R}}_{\gamma\kappa} \boldsymbol{v}_\kappa)^T \mathsf{B}_\gamma \Lambda_\gamma (\bar{\mathsf{R}}_{\gamma\kappa} \boldsymbol{w}_\kappa - \bar{\mathsf{R}}_{\gamma\nu} \boldsymbol{w}_\nu), \tag{14}$$

where $\boldsymbol{w}_\kappa = \boldsymbol{w}(\boldsymbol{u}_\kappa)$ are the entropy variables at the nodes of element κ, and $\boldsymbol{w}_\nu = \boldsymbol{w}(\boldsymbol{u}_\nu)$ are the entropy variables at the nodes of element ν. Recall that, for the particular operators used in this work, $\bar{\mathsf{R}}_{\gamma\kappa}$ simply picks out the values from $\boldsymbol{u}_\kappa$ (or, in this case, $\boldsymbol{w}_\kappa$) where the element nodes and face nodes coincide. Thus, $(\bar{\mathsf{R}}_{\gamma\kappa} \boldsymbol{w}_\kappa - \bar{\mathsf{R}}_{\gamma\nu} \boldsymbol{w}_\nu)$ is merely the difference in the entropy variables over the quadrature nodes of face γ.

The matrix Λ_γ is a block diagonal matrix with $4n_\gamma$ rows and columns, where n_γ is the number of nodes along face γ. The ith block of this matrix is given by (in Matlab-like notation)

$$\Lambda_\gamma(4(i-1):4i, 4(i-1):4i) = \left[|\lambda_{\max}| \frac{\partial \boldsymbol{\mathcal{U}}}{\partial \boldsymbol{\mathcal{W}}}\right]_{ii}, \qquad \forall i = 1, 2, \ldots, n_\gamma,$$

where $\lambda_{\max}$ is the spectral radius of the flux Jacobian in the normal direction at node i. The spectral radius and matrix $\partial \boldsymbol{\mathcal{U}} / \partial \boldsymbol{\mathcal{W}}$ are evaluated using a simple average of the two states on either side of the face at node i.

The dissipation (14) is added to the other terms on the left-hand side of (11), and this additional term ensures that the total entropy is non-increasing. This claim is easy to show on a periodic domain by replacing $\boldsymbol{v}_\kappa$ with $\boldsymbol{w}_\kappa$ in (14) and

summing over all elements:

$$\begin{aligned}
&\sum_{\kappa=1}^{K} \sum_{\gamma \in \partial\Omega_k} (\bar{\mathsf{R}}_{\gamma\kappa} \boldsymbol{w}_\kappa)^T \mathsf{B}_\gamma \Lambda_\gamma (\bar{\mathsf{R}}_{\gamma\kappa} \boldsymbol{w}_\kappa - \bar{\mathsf{R}}_{\gamma\nu} \boldsymbol{w}_\nu) \\
&\quad = \left[\cdots (\bar{\mathsf{R}}_{\gamma\kappa} \boldsymbol{w}_\kappa)^T \mathsf{B}_\gamma \Lambda_\gamma (\bar{\mathsf{R}}_{\gamma\kappa} \boldsymbol{w}_\kappa - \bar{\mathsf{R}}_{\gamma\nu} \boldsymbol{w}_\nu) + \cdots + (\bar{\mathsf{R}}_{\gamma\nu} \boldsymbol{w}_\nu)^T \mathsf{B}_\gamma \Lambda_\gamma (\bar{\mathsf{R}}_{\gamma\nu} \boldsymbol{w}_\nu - \bar{\mathsf{R}}_{\gamma\kappa} \boldsymbol{w}_\kappa) \right] \\
&\quad = \sum_{\gamma} (\bar{\mathsf{R}}_{\gamma\kappa} \boldsymbol{w}_\kappa - \bar{\mathsf{R}}_{\gamma\nu} \boldsymbol{w}_\nu)^T \mathsf{B}_\gamma \Lambda_\gamma (\bar{\mathsf{R}}_{\gamma\kappa} \boldsymbol{w}_\kappa - \bar{\mathsf{R}}_{\gamma\nu} \boldsymbol{w}_\nu) \\
&\quad \geq 0,
\end{aligned}$$

where we used the positive-definiteness and symmetry of the Hessian $\partial \boldsymbol{\mathcal{U}} / \partial \boldsymbol{\mathcal{W}}$ and the quadrature weight matrix B_γ. Note that the sum in the second-last line above is a sum over all faces γ in the mesh.

We can adapt the dissipation (14) to the DGD discretization in exactly the same way we adapted the entropy-conservative SBP discretization (11). That is, we need only replace $\boldsymbol{v}_\kappa$, $\boldsymbol{w}_\kappa$, and $\boldsymbol{w}_\nu$ with their interpolated values to obtain the following expression for the DGD dissipation on element κ:

$$\sum_{\gamma \in \partial\Omega_k} (\bar{\mathsf{R}}_{\gamma\kappa} \tilde{\boldsymbol{v}}_\kappa)^T \mathsf{B}_\gamma \Lambda_\gamma (\bar{\mathsf{R}}_{\gamma\kappa} \tilde{\boldsymbol{w}}_\kappa - \bar{\mathsf{R}}_{\gamma\nu} \tilde{\boldsymbol{w}}_\nu), \tag{15}$$

where, as before, $\tilde{\boldsymbol{v}}_\kappa = \tilde{\mathsf{P}}_\kappa \tilde{\boldsymbol{v}}$ are the interpolated test functions, and $\tilde{\boldsymbol{w}}_\kappa = \tilde{\mathsf{P}}_\kappa \tilde{\boldsymbol{w}}$ are the entropy variables interpolated to element κ. The DGD dissipation (15) retains the entropy stability of the baseline SBP dissipation, because of the symmetry of the operator.

E. Patch definition and reconstruction operator

To implement the DGD method using the SBP discretization, we need high-order interpolation operators defined on patches of elements. To construct these operators on unstructured triangular grids we rely on the patch reconstruction approach proposed by Li *et al.* [5].

To illustrate the reconstruction procedure, consider a two dimensional domain $\Omega \subset \mathbb{R}^2$ and its mesh $\mathcal{T}_h$ with K triangles. Here $h = \max_{\kappa \in \tau_h} h_\kappa$ is the maximum element diameter in the mesh $\mathcal{T}_h$. As described earlier, the DGD solution on each element is associated with the element's barycenter, although other sampling locations could be considered. For each element $\kappa \in \mathcal{T}_h$, an element patch S_κ is constructed containing element κ itself and some neighboring elements. We use $|S_\kappa|$ to denote the number of elements in each patch S_κ and X_κ the coordinates of the barycenters in the patch S_κ.

Before describing how we determine the elements in patch S_κ, let us define the prolongation/reconstruction operator $\tilde{\mathsf{P}}_\kappa$. Consider a polynomial $\mathcal{P} \in \mathbb{P}_p(\Omega)$, and let $\boldsymbol{p}_{\text{patch}}$ denote this polynomial evaluated at the coordinates in X_k and let $\boldsymbol{p}_\kappa$ denote the same polynomial evaluated at the SBP quadrature nodes Ξ_κ of element κ. Then the prolongation operator must satisfy

$$\tilde{\mathsf{P}}_\kappa \boldsymbol{p}_{\text{patch}} = \boldsymbol{p}_\kappa,$$

for all $\mathcal{P} \in \mathbb{P}_p(\Omega)$. This accuracy condition can be expressed concisely as

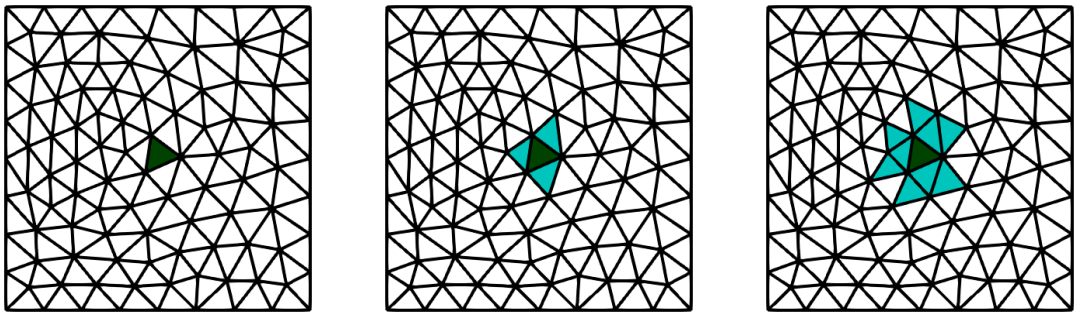
$$\tilde{\mathsf{P}}_\kappa \mathsf{V} = \mathsf{V}_\kappa \tag{16}$$

where $\mathsf{V} \in \mathbb{R}^{|S_\kappa| \times n^*}$ denotes a basis for $\mathbb{P}_p(\Omega)$ evaluated at the patch barycenters, X_κ, and $\mathsf{V}_\kappa \in \mathbb{R}^{n_\kappa \times n^*}$ is the basis evaluated at the SBP nodes of κ. Note that $n^* = (p+2)(p+1)/2$ is the dimension of $\mathbb{P}_p(\Omega)$.

The matrix equation (16) consists of $n_\kappa n^*$ equations. The unknowns are the entries in $\tilde{\mathsf{P}}_\kappa$, of which there are $n_\kappa |S_\kappa|$. Therefore, a necessary (but not sufficient) condition for there to be a solution to (16) is that the number of elements in the patch, $|S_\kappa|$, is greater than the number of basis functions n^*. In this work we ensure that $|S_\kappa| \geq n^*$ and check that V has full rank, and then we find the minimium-norm solution to (16).

To ensure a sufficient number of elements are in S_κ, the patches are constructed as follows. Consider the recursive set definition

$$\begin{aligned}
S_\kappa^0 &= \{\kappa\}, \\
S_\kappa^j &= S_\kappa^{j-1} \cup N(S_\kappa^{j-1}),
\end{aligned}$$



(a) Initial element in patch ($p = 0$) (b) Patch for $p = 1$ (c) Patch for $p = 2$

Fig. 2 Patch construction

where $N(S_\kappa^{j-1})$ are the face-adjacent neighbors of all elements in the set S_κ^{j-1} in the sense of graph theory. In other words, S_κ^j consists of all the elements in S_κ^{j-1} plus all those elements that are adjacent along the face to any element in S_κ^{j-1}. The element patch S_κ is defined as the smallest index j such that $|S_\kappa^j| \geq n^*$. Figure 2 illustrates the patches corresponding to $\tilde{\mathsf{P}}_\kappa$ for degree $p = 0$, $p = 1$ and $p = 2$ reconstructions.

IV. Numerical Experiments

This section presents numerical experiments intended to verify the DGD discretization described in Section III. To verify the accuracy of the discretization, the steady isentropic vortex problem is solved using the DGD discretization and the results are compared against those obtained using the discontinuous SBP discretization. Subsequently, the entropy stability of the DGD discretization is demonstrated using an unsteady isentropic vortex. We also use the unsteady vortex to examine the spectra of the DGD operators.

A. Steady isentropic vortex problem

The two-dimensional isentropic vortex is a simple flow consisting of circular streamlines and radially varying density and pressure. It is often used to verify accuracy because it has an analytical solution. Specifically, the exact solution for the two-dimensional steady vortex problem is defined as

$$\begin{aligned} \rho(r) &= \rho_i \left[1 + \frac{\gamma - 1}{2} M_i^2 \left(1 - \frac{r_i^2}{r^2} \right) \right]^{\frac{1}{\gamma - 1}}, & u(r,\theta) &= \rho \sqrt{\frac{\gamma p}{\rho}} M_a \sin\theta, \\ v(r,\theta) &= -\rho \sqrt{\frac{\gamma p}{\rho}} M_a \cos\theta, & e(r,\theta) &= \frac{p}{\gamma - 1} + \frac{1}{2} \gamma p M_a^2, \end{aligned} \tag{17}$$

where r is the radial polar coordinate, and $r_i = 1$ is the reference radius. The density and Mach number at r_i are given by $\rho_i = 1$ and $M_i = 0.5$, respectively. Here, u, v, e are calculated using the isentropic gas relations and M_a is the local mach number given by

$$M_a = \sqrt{\frac{2}{\gamma - 1} \left[\left(\frac{\rho_i}{\rho} \right)^{\gamma - 1} \left(1 + \frac{1}{2} (\gamma - 1) M_i^2 \right) - 1 \right]}.$$

The domain for the steady-vortex verification is a quarter annulus: $\Omega = \{(r,\theta) \mid 1 \leq r \leq 3,\ 0 \leq \theta \leq \pi/2\}$. The mesh is created by generating an $N \times N$ quadrilateral mesh in polar-coordinate space and then splitting the quadrilaterals into triangles. The triangles are then mapped to physical space using an isoparametric mapping of degree $p + 1$ for an SBP/DGD discretization of degree p. Additional details regarding the mesh for the isentropic-vortex case can be found in [32]. A sample mesh for $N = 10$ is shown in figure 3(a), and the density obtained using the SBP entropy-stable

discretization is shown in Figure 3(b). The slip-wall boundary condition is applied along the inner radius at $r = 1$, and the exact solution is applied to incoming characteristics on the remaining boundaries using the Roe numerical-flux function. Finally, note that entropy variables are adopted as the state variables for both the DGD and D-SBP discretizations[‡].

Figure 4 compares the L^2 density error calculated from the DGD discretizations, for $1 \leq p \leq 4$, as a function of element size h. The L^2 density error is defined by

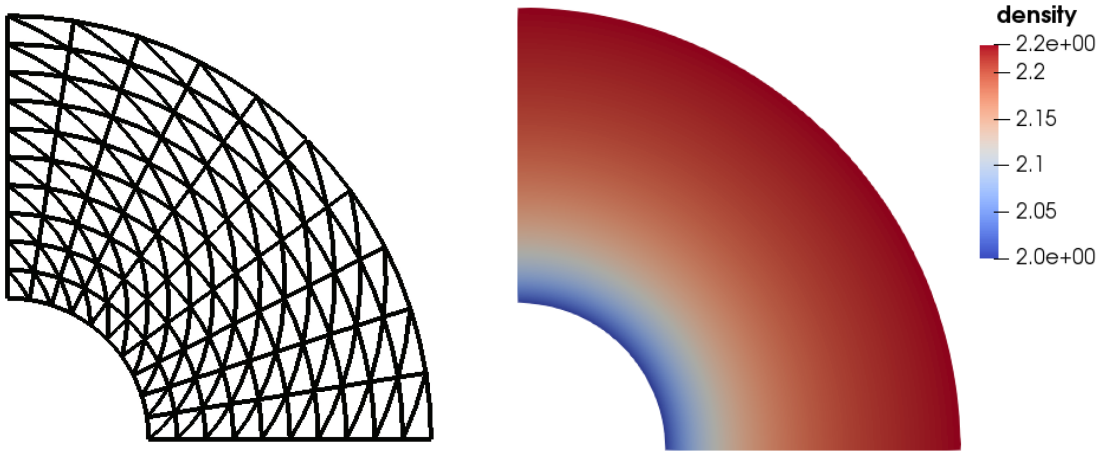
$$L^2 \text{ Error} = \sqrt{\sum_{\kappa=1}^{K} (\rho_\kappa - \rho_\kappa^{\text{exact}})^T \mathsf{H}_\kappa (\rho_\kappa - \rho_\kappa^{\text{exact}})}$$

where ρ_κ is the density — as a function of the entropy variables — on the nodes of element κ obtained from either the DGD or D-SBP discretizations. Recall that H_κ is the SBP norm matrix which defines a quadrature rule. The results show that the D-SBP errors appear to approach the optimal $p + 1$ rate asymptotically for all polynomial degrees under consideration. For the DGD errors, the degree $p = 3$ and $p = 4$ appear to be converging at close to optimal rates, while the $p = 1$ and $p = 2$ schemes are super- and sub-optimal, respectively.

Accuracy versus element size provides an incomplete picture of the efficiency of the D-SBP and DGD schemes, since the D-SBP scheme has more degrees of freedom on the same mesh. In order to compare the two discretizations more fairly, we have plotted the L^2 density error versus the number of degrees of freedom in Figure 5. For the DGD discretization the number of degrees of freedom is simply the total number of elements, K. For the D-SBP scheme the number of degrees of freedom is K times the number of nodes per element.

The results in Figure 5 reveal that the DGD discretization generally outperforms the D-SBP discretization when error is measured in terms of degrees of freedom. In particular, while errors from the two discretizations are similar for the lowest order $p = 1$ schemes, the DGD errors are significantly smaller for the higher order operators.

We emphasize that, while the number of degrees of freedom provides a better cost metric than element size, it is by no means perfect. In particular, the number of degrees of freedom does not reflect the potential computational saving of the DGD scheme due to its better spectral radius and conditioning; see, for example, the DGD spectra presented in the next section. A more thorough comparison is the subject of on-going work.



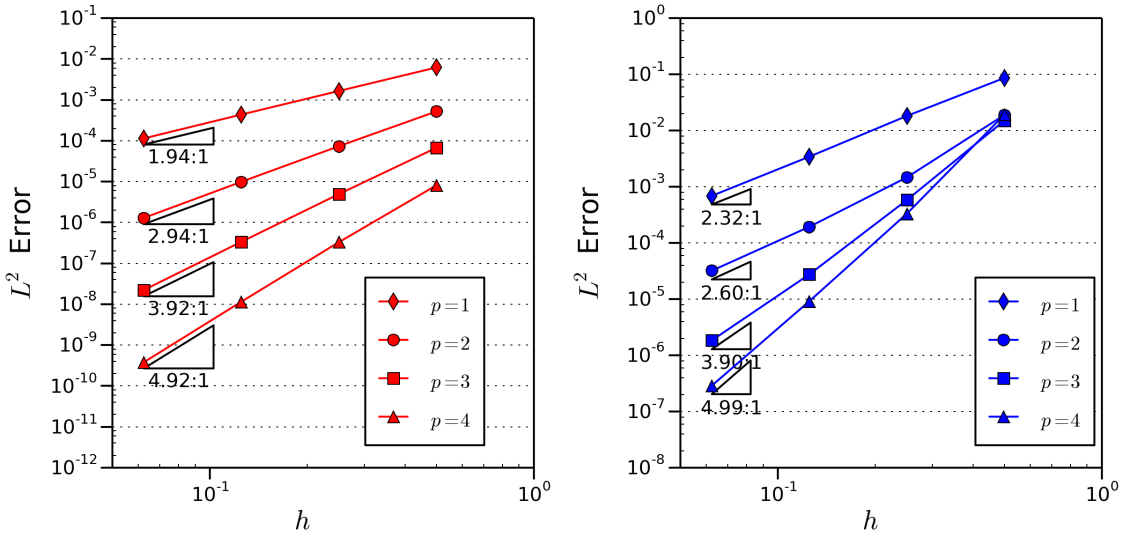
(a) 2D curvilinear mesh (b) density for $p = 3$, $N = 16$

Fig. 3 Sample mesh and solution for the steady-vortex problem

B. Unsteady vortex problem

In this section we apply the DGD method to the two-dimensional unsteady isentropic vortex problem to verify that the scheme is entropy conserative, if no dissipation is present, and entropy stable, if dissipation is present. The analytical

[‡]For steady problems, there is no difference between using the conservative and entropy variables for the D-SBP discretization.



(a) D-SBP (b) DGD

Fig. 4 L^2 **solution error for the D-SBP and DGD schemes versus element size**

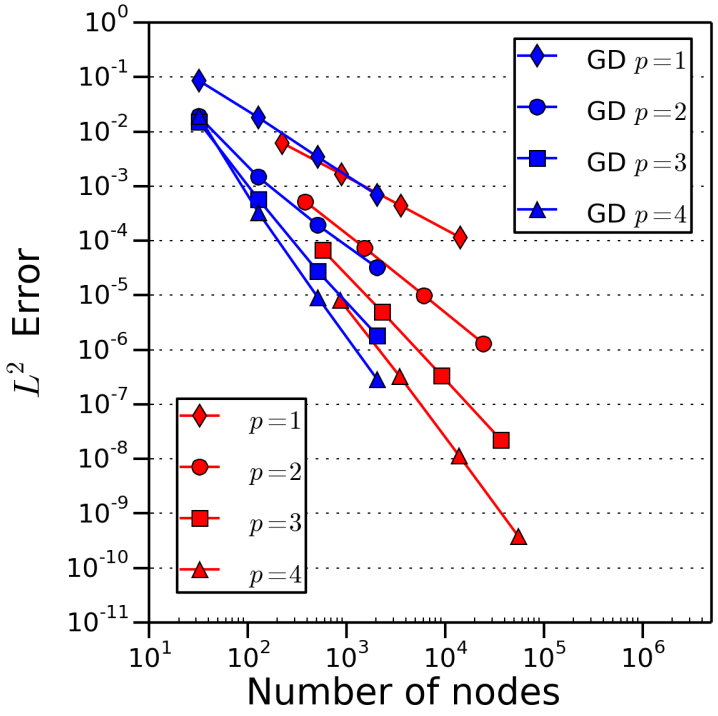


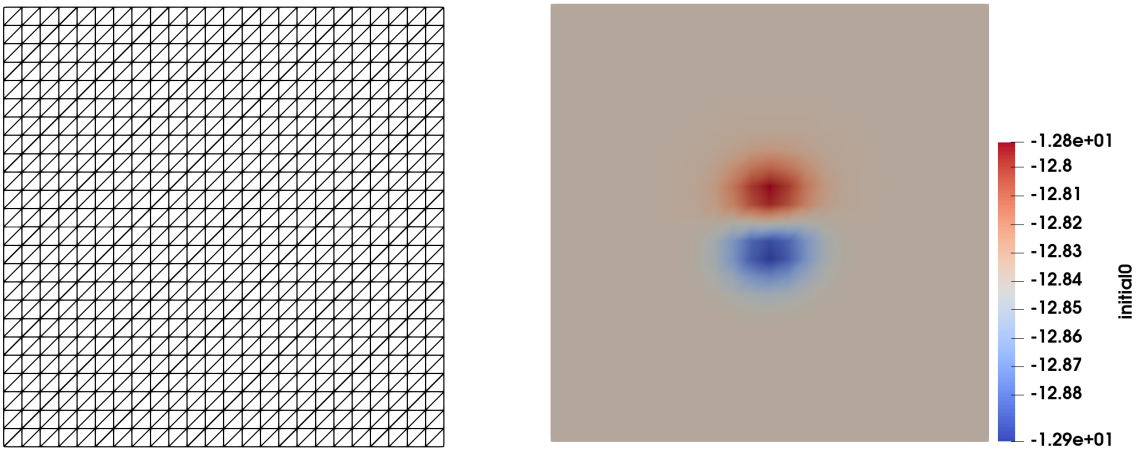
Fig. 5 L^2 **error versus degree-of-freedoms for the DGD and D-SBP discretizations**

solution is defined as

$$\rho = \left(1 - \frac{\epsilon^2(\gamma-1)M^2}{8\pi^2}\exp\left(f(x,y,t)\right)\right)^{\frac{1}{\gamma-1}}, \qquad u = k\rho\left(1 - \frac{\epsilon y}{2\pi}\exp\left(\frac{f(x,y,t)}{2}\right)\right),$$
$$v = k\rho\frac{\epsilon x}{2\pi}\exp\left(\frac{f(x,y,t)}{2}\right), \qquad e = \frac{p}{\gamma-1} + \frac{1}{2\rho}(u^2+v^2), \tag{18}$$

where $f(x,y,t) = 1 - ((x - x_0 - t)^2 + (y - y_0)^2)$, and the vortex is initially centered at $(x_0, y_0) = (0.5, 0.5)$. The Mach number is $M = 0.5$, and $\epsilon = 1.0$ is the vortex strength. The constant k is a scaling factor that controls the vortex speed, and in the following simulations its value is 15.

The unsteady isentropic vortex problem is solved on the square domain $\Omega = \{(x,y)|x \in [0,1], y \in [0,1]\}$ with periodic boundary conditions applied on all four edges. The analytical solution is imposed as the initial condition. All simulations are run for a period of $T = 1/k$ time units, which allows the vortex to return to its initial position of $(0.5, 0.5)$. A sample mesh and the initial condition are shown in Figures 6(a) and 6(b), respectively.



(a) Example mesh

(b) Initial condition corresponding to first entropy variable

Fig. 6 Mesh and initial condition for the unsteady vortex study

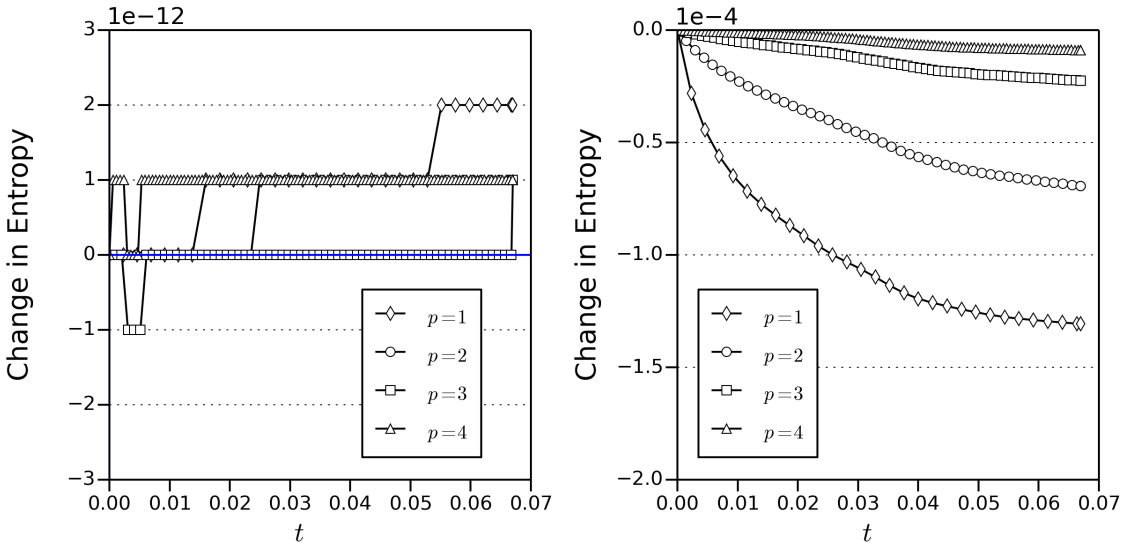
The simulations use the relaxation Runge-Kutta (RRK) [33] variant of the implicit midpont method to discretize the time derivative. RRK methods respect the entropy conservation/stability of the spatial discretization. Furthermore, our choice of the implicit midpoint method allows us to adopt a relatively large CFL number of 10.

Both entropy-conservative and entropy-stable DGD schemes were considered for this study in order to verify that total entropy is conserved and dissipated, respectively. Figure 7(a) shows the time history of the spatially integrated entropy, $\sum_\kappa \mathbf{1}^T \mathsf{H}_\kappa \tilde{\boldsymbol{s}}_k$, for the entropy-conservative scheme. The figure shows that entropy is conserved up to an error of approximately 10^{-12}, which is consistent with the tolerances used in the iterative solvers. Figure 7(b) plots the analogous results for the entropy-stable DGD scheme. In this case, the entropy decreases monotonically, as predicted by the theory. Moreover, we see that less entropy is dissipated as the solution degree p increases.

We conclude our study of the unsteady vortex by presenting the spectra of the DGD operators for this problem. While the eigenvalues are not strictly relevant to the nonlinear entropy-stability of the DGD discretizations, the spectra do provide insight into the potential efficiency of the schemes. Figures 8(a) and 8(b) show the eigenvalue distributions of the matrices $\mathsf{H}^{-1}\mathsf{J}$ for the entropy-conservative and -stable schemes, respectively, where J is the Jacobian of the spatial residual and H is the Jacobian of the temporal term. To normalize the results, the eigenvalues are scaled by the spectral radius of the corresponding $p = 1$ operator.

The most notable feature of the eigenvalues is that the spectral radius does not grow significantly as the polynomial degree increases. This suggests that a high-order DGD scheme can take large time steps relative to DG-type discretizations. Granted, a disadvantage of the DGD discreization is that its mass matrix is neither diagonal nor block-diagonal, which diminishes the appeal of explicit time marching schemes. Note that, while efficient solvers are available for the mass matrix in tensor-product DGD schemes, this is not the case for the unstructured DGD scheme considered here.

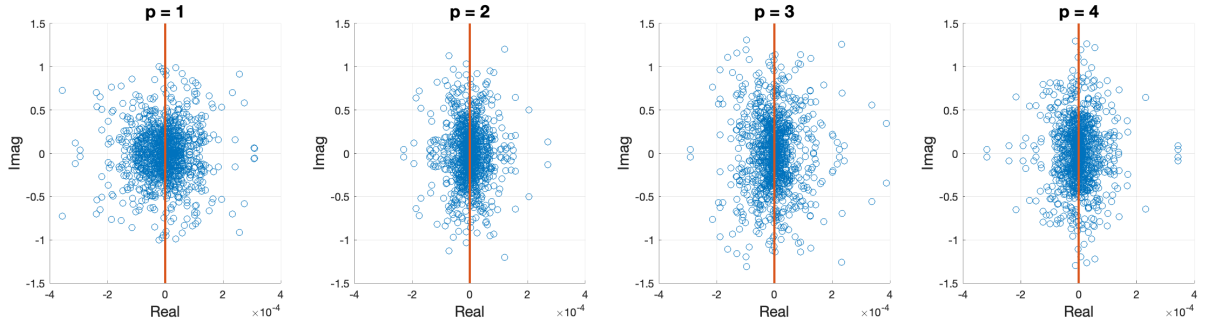
Finally, the careful reader will notice that the real part of some eigenvalues in figures 8(a) and 8(b) is positive. According to linear stability, this would imply that the discretizations are unstable. However, this illustrates one of the limitations of linear theory and does not contradict nonlinear entropy stability theory.



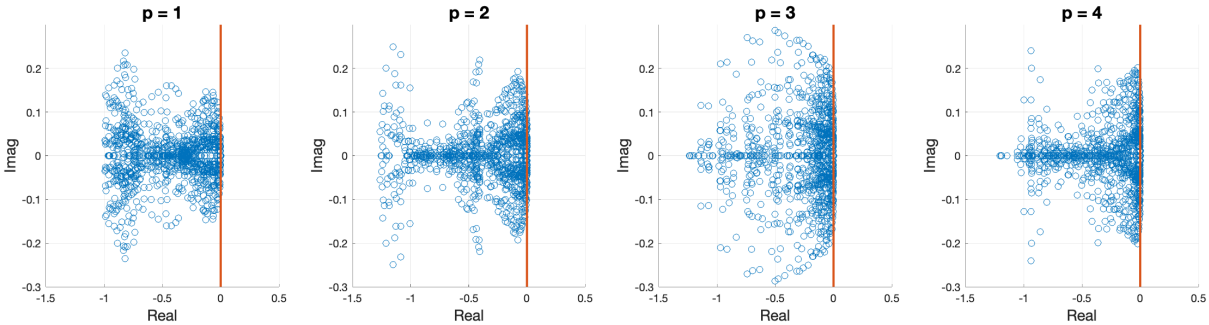
(a) Entropy conservative scheme

(b) Entropy stable scheme

Fig. 7 Change in mathematical entropy



(a)



(b)

Fig. 8 Normalized eigenvalues from the entropy-conservative (upper row) and entropy stable (lower row) DGD discretizations

V. Conclusions

We have presented entropy-conservative and entropy-stable discontinuous Galerkin difference (DGD) discretizations of the Euler equations on unstructured grids. To construct the DGD discretizations and establish their properties, we leveraged existing SBP theory and used high-order projection operators to map the DGD test and trial functions from the element centers to the nodes of the SBP discretization.

The accuracy of the DGD discretizations was verified using the steady isentropic vortex. Using the steady vortex, we also demonstrated that the DGD discretization was more efficient than the baseline D-SBP discretization when the L^2 solution error was measured versus number of degrees of freedom.

We solved the unsteady isentropic vortex in order to verify the stability properties of the DGD discretizations. In particular, we showed that entropy is conserved by the entropy-conservative DGD schemes, up to the accuracy of the iterative methods used during each time step. Similarly, entropy was shown to decrease monotonically for the entropy-stable DGD schemes. We also used the unsteady-vortex problem to investigate the spectra of the DGD operators. The spectra have attractive distributions from the perspective of time-step restrictions, with limited dependence on operator order of accuracy.

Acknowledgments

G. Yan was supported by the National Science Foundation under Grant No. 1554253, and S. Kaur was supported by the National Science Foundation under Grant No. 1825991. The authors gratefully acknowledge this support. We also thank RPI's Scientific Computation Research Center for the use of computer facilities.

References

[1] Banks, J., and Hagstrom, T., "On Galerkin Difference Methods," *Journal of Computational Physics*, Vol. 313, 2016. https://doi.org/10.1016/j.jcp.2016.02.042.

[2] Banks, J. W., Hagstrom, T., and Jacangelo, J., "Galerkin Differences for acoustic and elastic wave equations in two space dimensions," *Journal of Computational Physics*, Vol. 372, 2018, pp. 864–892. https://doi.org/10.1016/j.jcp.2018.06.029.

[3] Kozdon, J. E., Wilcox, L. C., Hagstrom, T., and Banks, J. W., "Robust approaches to handling complex geometries with Galerkin difference methods," *Journal of Computational Physics*, Vol. 392, 2019, pp. 483 – 510. https://doi.org/https://doi.org/10.1016/j.jcp.2019.04.031, URL http://www.sciencedirect.com/science/article/pii/S0021999119302773.

[4] Hagstrom, T., Banks, J. W., Buckner, B. B., and Juhnke, K., "Discontinuous Galerkin Difference Methods for Symmetric Hyperbolic Systems," *Journal of Scientific Computing*, Vol. 81, No. 3, 2019, pp. 1509–1526.

[5] Li, R., Ming, P., Sun, Z., and Yang, Z., "An Arbitrary-Order Discontinuous Galerkin Method with One Unknown Per Element," *Journal of Scientific Computing*, Vol. 80, No. 1, 2019, pp. 268–288. https://doi.org/10.1007/s10915-019-00937-y, URL https://doi.org/10.1007/s10915-019-00937-y.

[6] Kreiss, H. O., and Scherer, G., "Finite element and finite difference methods for hyperbolic partial differential equations," *Mathematical Aspects of Finite Elements in Partial Differential Equations*, edited by C. de Boor, Mathematics Research Center, the University of Wisconsin, Academic Press, 1974.

[7] Svärd, M., and Nordström, J., "Review of summation-by-parts schemes for initial-boundary-value problems," *J. Comput. Physics*, Vol. 268, 2014, pp. 17–38.

[8] Del Rey Fernández, D. C., Hicken, J. E., and Zingg, D. W., "Review of summation-by-parts operators with simultaneous approximation terms for the numerical solution of partial differential equations," *Computers & Fluids*, Vol. 95, 2014, pp. 171–196. https://doi.org/10.1016/j.compfluid.2014.02.016, URL http://dx.doi.org/10.1016/j.compfluid.2014.02.016.

[9] Tadmor, E., "The Numerical Viscosity of Entropy Stable Schemes for Systems of Conservation Laws. I," *Mathematics of Computation*, Vol. 49, No. 179, 1987, pp. 91–103. URL http://www.jstor.org/stable/2008251.

[10] Tadmor, E., "Entropy stability theory for difference approximations of nonlinear conservation laws and related time-dependent problems," *Acta Numerica*, Vol. 12, 2003, pp. 451–512. https://doi.org/10.1017/s0962492902000156, URL http://dx.doi.org/10.1017/s0962492902000156.

[11] Ismail, F., and Roe, P. L., "Affordable, entropy-consistent Euler flux functions II: Entropy production at shocks," *Journal of Computational Physics*, Vol. 228, No. 15, 2009, pp. 5410–5436. https://doi.org/10.1016/j.jcp.2009.04.021, URL http://dx.doi.org/10.1016/j.jcp.2009.04.021.

[12] Fisher, T. C., "High-order L2 stable multi-domain finite difference method for compressible flows," Ph.D. thesis, 2012. URL http://libproxy.rpi.edu/login?url=https://search.proquest.com/docview/1221239207?accountid=28525, copyright - Database copyright ProQuest LLC; ProQuest does not claim copyright in the individual underlying works; Last updated - 2019-11-01.

[13] Fisher, T. C., and Carpenter, M. H., "High-order entropy stable finite difference schemes for nonlinear conservation laws: Finite domains," *Journal of Computational Physics*, Vol. 252, 2013, pp. 518–557. https://doi.org/10.1016/j.jcp.2013.06.014, URL http://dx.doi.org/10.1016/j.jcp.2013.06.014.

[14] Fisher, T. C., Carpenter, M. H., Nordström, J., Yamaleev, N. K., and Swanson, C., "Discretely conservative finite-difference formulations for nonlinear conservation laws in split form: Theory and boundary conditions," *Journal of Computational Physics*, Vol. 234, 2013, pp. 353–375. https://doi.org/10.1016/j.jcp.2012.09.026, URL http://dx.doi.org/10.1016/j.jcp.2012.09.026.

[15] Hughes, T. J. R., Franca, L. P., and Mallet, M., "A new finite element formulation for computational fluid dynamics: I. symmetric forms of the compressible Euler and Navier-Stokes equations and the second law of thermodynamics," *Computer Methods in Applied Mechanics and Engineering*, Vol. 54, No. 2, 1986, pp. 223–234. https://doi.org/10.1016/0045-7825(86)90127-1, URL http://dx.doi.org/10.1016/0045-7825(86)90127-1.

[16] Jiang, G., and Shu, C.-W., "On a Cell Entropy Inequality for Discontinuous Galerkin Methods," *Math. Comput.*, Vol. 62, No. 206, 1994, pp. 531–538. https://doi.org/10.2307/2153521, URL http://dx.doi.org/10.2307/2153521.

[17] Crean, J., Hicken, J. E., Del Rey Fernández, D. C., Zingg, D. W., and Carpenter, M. H., "Entropy-stable summation-by-parts discretization of the Euler equations on general curved elements," *Journal of Computational Physics*, Vol. 356, 2018, pp. 410–438. https://doi.org/10.1016/j.jcp.2017.12.015, URL http://dx.doi.org/10.1016/j.jcp.2017.12.015.

[18] Del Rey Fernández, D. C., Boom, P. D., and Zingg, D. W., "A Generalized Framework for Nodal First Derivative Summation-By-Parts Operators," *Journal of Computational Physics*, Vol. 266, No. 1, 2014, pp. 214–239.

[19] Gassner, G. J., "A Skew-Symmetric Discontinuous Galerkin Spectral Element Discretization and Its Relation to SBP-SAT Finite Difference Methods," *SIAM Journal on Scientific Computing*, Vol. 35, No. 3, 2013, pp. A1233–A1253. https://doi.org/10.1137/120890144, URL http://dx.doi.org/10.1137/120890144.

[20] Hicken, J. E., and Zingg, D. W., "Summation-by-parts operators and high-order quadrature," *Journal of Computational and Applied Mathematics*, Vol. 237, No. 1, 2013, pp. 111–125. https://doi.org/10.1016/j.cam.2012.07.015, URL http://dx.doi.org/10.1016/j.cam.2012.07.015.

[21] Carpenter, M. H., Gottlieb, D., and Abarbanel, S., "Time-stable boundary conditions for finite-difference schemes solving hyperbolic systems: methodology and application to high-order compact schemes," *Journal of Computational Physics*, Vol. 111, No. 2, 1994, pp. 220–236. https://doi.org/10.1006/jcph.1994.1057, URL http://dx.doi.org/10.1006/jcph.1994.1057.

[22] Carpenter, M. H., Nordström, J., and Gottlieb, D., "Revisiting and extending interface penalties for multi-domain summation-by-parts operators," *Journal of Scientific Computing*, Vol. 45, No. 1, 2009, pp. 118–150. https://doi.org/10.1007/s10915-009-9301-5, URL http://dx.doi.org/10.1007/s10915-009-9301-5.

[23] Lax, P., *Hyperbolic Systems of Conservation Laws and the Mathematical Theory of Shock Waves*, Society for Industrial and Applied Mathematics, 1973. https://doi.org/10.1137/1.9781611970562, URL http://epubs.siam.org/doi/abs/10.1137/1.9781611970562.

[24] Tadmor, E., "Skew-selfadjoint form for systems of conservation laws," *Journal of Mathematical Analysis and Applications*, Vol. 103, No. 2, 1984, pp. 428–442. https://doi.org/10.1016/0022-247x(84)90139-2, URL http://dx.doi.org/10.1016/0022-247x(84)90139-2.

[25] Dafermos, C. M., *Hyperbolic Conservation Laws in Continuum Physics*, Vol. 325, Springer Berlin Heidelberg, Berlin, Heidelberg, 2010. https://doi.org/10.1007/978-3-642-04048-1, URL http://dx.doi.org/10.1007/978-3-642-04048-1.

[26] Fisher, T. C., and Carpenter, M. H., "High-order entropy stable finite difference schemes for nonlinear conservation laws: Finite domains," *Journal of Computational Physics*, Vol. 252, 2013, pp. 518–557. https://doi.org/10.1016/j.jcp.2013.06.014, URL http://dx.doi.org/10.1016/j.jcp.2013.06.014.

[27] Chen, T., and Shu, C.-W., "Entropy stable high order discontinuous Galerkin methods with suitable quadrature rules for hyperbolic conservation laws," *Journal of Computational Physics*, Vol. 345, 2017, pp. 427–461. https://doi.org/10.1016/j.jcp.2017.05.025, URL http://dx.doi.org/10.1016/j.jcp.2017.05.025.

[28] Chan, J., "On Discretely Entropy conservative and Entropy Stable Discontinuous Galerkin Methods," *Journal of Computational Physics*, Vol. 362, 2018, pp. 346–374. https://doi.org/10.1016/j.jcp.2018.02.033.

[29] Parsani, M., Carpenter, M. H., Fisher, T. C., and Nielsen, E. J., "Entropy Stable Staggered Grid Discontinuous Spectral Collocation Methods of any Order for the Compressible Navier–Stokes Equations," *SIAM Journal on Scientific Computing*, Vol. 38, No. 5, 2016, pp. A3129–A3162. https://doi.org/10.1137/15m1043510, URL http://dx.doi.org/10.1137/15m1043510.

[30] Fernández, D. C. D. R., Crean, J., Carpenter, M. H., and Hicken, J. E., "Staggered-grid entropy-stable multidimensional summation-by-parts discretizations on curvilinear coordinates," *Journal of Computational Physics*, Vol. 392, 2019, pp. 161–186. https://doi.org/10.1016/j.jcp.2019.04.029.

[31] Kolev, T., and Dobrev, V., "Modular Finite Element Methods (MFEM)," [Computer Software] https://doi.org/10.11578/dc.20171025.1248, jun 2010. URL https://doi.org/10.11578/dc.20171025.1248.

[32] Hicken, J. E., "Entropy-Stable, High-Order Summation-by-Parts Discretizations Without Interface Penalties," *Journal of Scientific Computing*, Vol. 82, No. 2, 2020, p. 50. https://doi.org/10.1007/s10915-020-01154-8, URL https://doi.org/10.1007/s10915-020-01154-8.

[33] Ranocha, H., Sayyari, M., Dalcin, L., Parsani, M., and Ketcheson, D. I., "Relaxation Runge–Kutta Methods: Fully Discrete Explicit Entropy-Stable Schemes for the Compressible Euler and Navier–Stokes Equations," *SIAM Journal on Scientific Computing*, Vol. 42, No. 2, 2020, pp. A612–A638.