ENTROPY STABLE GAUSS-LEGENDRE COLLOCATION METHODS

JESSE CHAN, DAVID C. DEL REY FERNANDEZ, MARK H. CARPENTER

Abstract. Here is the abstract.

1. Introduction. Time dependent nonlinear conservation laws are ubiquitous in computational fluid dynamics (CFD). These conservation laws include the compressible Euler and Navier-Stokes equations, which govern compressible fluid behavior within aerospace applications, as well as the equations governing coastal flooding and geophysical flows (shallow water), and those describing astrophysical phenomena (magnetohydrodynamics). The utility of these simulations depends both on the reliability and accuracy of numerical methods for the solution of the associated nonlinear partial differential equations (PDEs).

High order methods are increasingly of interest for these applications. Such methods are more accurate per degree of freedom than low order methods, while also possessing much smaller numerical dispersion and dissipation errors. This makes high order methods especially well suited to time-dependent simulations. In this work, we focus specifically on discontinuous Galerkin methods on unstructured quadrilateral and hexahedral meshes. These methods combine properties of high order approximations with the geometric flexibility of unstructured meshing.

However, high order methods are notorious for being more prone to instability compared to low order methods [1]. This instability is addressed through various stabilization techniques (e.g. artificial viscosity, filtering, slope limiting). However, these techniques often reduce accuracy to first or second order, and can prevent solvers from realizing the advantages of high order approximations. Moreover, it is often not possible to prove that a high order scheme does not blow up even in the presence of stabilization. This ambiguity can necessitate the re-tuning of stabilization parameters, as parameters which are both stable and accurate for one problem or discretization setting may provide either too little or too much numerical dissipation for another.

The instability of high order methods is rooted in the fact that discretizations of nonlinear conservation laws do not typically satisfy a discrete analogue of the conservation or dissipation of energy (entropy). While the numerical dissipation present for low order methods serves as a stabilizing mechanism, the low of numerical dissipation in high order methods reveals the lack of an inherent statement of stability. The dissipation of entropy serves as an energetic principle for nonlinear conservation laws [2], and requires the use of the chain rule in its proof. The instability of high order methods is tied to the fact that, when discretizing systems of nonlinear PDEs, the chain rule does not typically hold at the discrete level. The lack of a chain rule was circumvented by using a non-standard "flux differencing" formulation [3, 4, 5, 6], which is key to constructing semi-discretely entropy stable high order schemes on unstructured quadrilateral and hexahedral meshes. Flux differencing replaces the derivative of the nonlinear flux with the discrete differentiation of an auxiliary quantity. This auxiliary quantity is computed through the evaluation of a two-point entropy con-

servative flux [7] using pairs of solution values at quadrature points. These entropy stable schemes were later extended to non-tensor product elements using GLL-like quadrature points on triangles and tetrahedra [8, 9]. More recently, the construction of efficient entropy stable schemes was extended to more arbitrary choices of basis and quadrature [10, 11].

This paper focuses on the construction of efficient semi-discretely entropy stable collocation schemes based on Gauss nodes, using the framework introduced in [10, 11]. While entropy stable collocation schemes have been constructed on Gauss-like quadrature points without boundary nodes [12], the inter-element coupling terms for such schemes introduce significantly more communication and computation than collocation points which contain boundary nodes. This work introduces new inter-element coupling terms which avoid this all-to-all coupling, requiring only communication of face values between neighboring elements.

In Section 2, we briefly review the derivation of continuous entropy inequalities for systems of nonlinear conservation laws. In Section 3, we describe how to construct entropy stable discretizations of nonlinear conservation laws using different quadrature points on affine tensor product elements. In Section 4, we describe how to extend this construction to curvilinear elements, and Section 5 presents numerical results which confirm the high order accuracy and stability of the proposed method for smooth, discontinuous, and under-resolved (turbulent) solutions of the compressible Euler equations in two and three dimensions.

2. A brief review of entropy stability theory. We are interested in methods for the numerical solution of systems of conservation laws in d dimensions

$$\frac{\partial \boldsymbol{u}}{\partial t} + \sum_{i=1}^{d} \frac{\partial \boldsymbol{f}_i(\boldsymbol{u})}{\partial x_i} = 0, \tag{2.1}$$

where $\boldsymbol{u}$ denotes the conservative variables, $\boldsymbol{f}_i(\boldsymbol{u})$ are nonlinear fluxes, and x_i denotes the ith coordinate. Many physically motivated conservation laws admit a statement of stability involving a convex scalar entropy $S(\boldsymbol{u})$. We first define the entropy variables $\boldsymbol{v}(\boldsymbol{u})$ to be the gradient of the entropy $S(\boldsymbol{u})$ with respect to the conservative variables

$$\boldsymbol{v} = \frac{\partial S(\boldsymbol{u})}{\partial \boldsymbol{u}}.$$

For a convex entropy, $\boldsymbol{v}(\boldsymbol{u})$ defines an invertible mapping from conservative to entropy variables. We denote the inverse of this mapping (from entropy to conservative variables) by $\boldsymbol{u}(\boldsymbol{v})$.

At the continuous level, it can be shown (for example, in [2]) that vanishing viscosity solutions to (2.1) satisfy the strong form of an entropy inequality

$$\frac{\partial S(\boldsymbol{u})}{\partial t} + \sum_{i=1}^{d} \frac{\partial F_i(\boldsymbol{u})}{\partial x_i} \leq 0, \tag{2.2}$$

where F_i denotes the ith scalar entropy flux function and $\psi_i(\boldsymbol{u})$ denotes the ith entropy potential

$$F_i(\boldsymbol{u}) = \boldsymbol{v}^T \frac{\partial \boldsymbol{f}_i}{\partial x_i}, \qquad \psi_i(\boldsymbol{u}) = \boldsymbol{v}^T \boldsymbol{f}_i(\boldsymbol{u}) - F_i(\boldsymbol{u}).$$

Integrating (2.2) over a domain Ω and applying the divergence theorem yields an

averaged entropy inequality

$$\int_\Omega \frac{\partial S(\boldsymbol{u})}{\partial t} + \int_{\partial\Omega} \sum_{i=1}^{d} \boldsymbol{n}_i \left(\boldsymbol{v}^T \boldsymbol{f}_i(\boldsymbol{u}) - \psi_i(\boldsymbol{u}) \right) \leq 0, \tag{2.3}$$

where $\partial\Omega$ denotes the boundary of Ω and $\boldsymbol{n}_i$ denotes the ith component of the outward normal on $\partial\Omega$. Roughly speaking, this implies that the time rate of change of entropy is less than or equal to the flux of entropy through the boundary.

3. Entropy stable Gauss and Gauss-Legendre-Lobatto collocation methods. This focus of this paper is on entropy stable high order collocation methods which satisfy a semi-discrete version of the entropy inequality (2.3). These methods collocate the solution at some choice of collocation nodes, and are applicable to tensor product meshes consisting of quadrilateral and hexahedral elements.

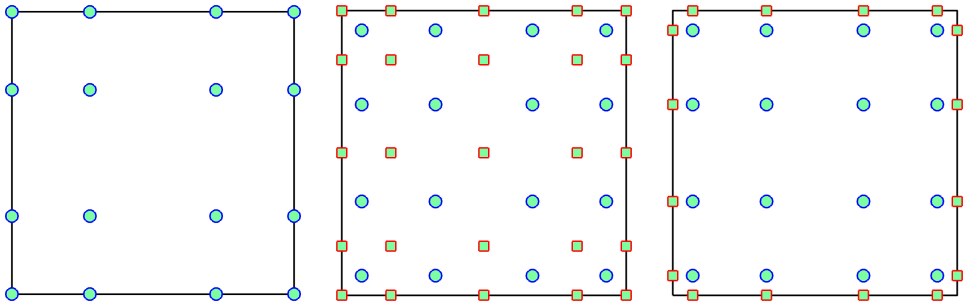
Entropy stable collocation methods have largely utilized Gauss-Legendre-Lobatto (GLL) nodes [3, 4, 5, 6], which contain points on the boundary. The popularity of GLL nodes can be attributed in part to a connection made in [13], where it was shown by Gassner that collocation DG discretizations based on GLL nodes could be recast in terms of summation-by-parts (SBP) operators. This equivalence allowed Gassner to leverage existing finite difference formulations to produce stable high order discretizations of the nonlinear Burgers' equation.

GLL quadratures contain boundary points, which greatly simplifies the construction of inter-element coupling terms for entropy stable collocation schemes. However, it is also known that the use of GLL quadrature within DG methods under-integrates the mass matrix, which can lead to solution "aliasing" and lower accuracy [14]. In this work, we explore entropy stable collocation schemes based on Gauss quadrature points instead of GLL points.

This comparison is motivated by the accuracy of each respective quadrature rule. While $(N+1)$-point GLL quadrature rules are exact for polynomial integrands of degree $(2N-1)$, $(N+1)$-point Gauss quadrature rules are exact for polynomials of degree $(2N+1)$. This higher accuracy of Gauss quadrature has been shown to translate to lower errors and slightly improved rates of convergence in simulations of wave propagation and fluid flow [15, 16, 17]. However, Gauss points have not been widely used to construct entropy stable discretizations due to the lack of efficient, stable, and high order accurate inter-element coupling terms, known as simultaneous approximation terms (SBP-SAT) in the finite difference literature [18, 12, 19]. SBP-SATs for Gauss points are non-compact, in the sense that they introduce all-to-all coupling between degrees of freedom on neighboring elements. This results in greater communication between elements, as well as a significantly larger number of two-point flux evaluations and floating point operations.

It is possible to realize the improved accuracy of Gauss points while avoiding non-compact SBP-SATs through a staggered grid formulation, where the solution is stored at Gauss nodes but interpolated to a set of higher degree $(N+2)$ GLL "flux" points for computation [14]. Because GLL nodes include boundary points, compact and high orderaccurate SBP-SAT terms can be constructed for the flux points. After performing computations on the flux points, the results are interpolated back to Gauss points and used to evolve the solution forward in time. Figure 1 shows an illustration of GLL, staggered grid, and Gauss point sets for a 2D quadrilateral element.

The following sections will describe how to construct efficient high order entropy stable schemes using Gauss points. These schemes are based on "decoupled" SBP operators introduced in [10, 11], which are applicable to general choices of basis and



(a) GLL nodes (b) Staggered grid nodes (c) Gauss nodes

Fig. 1: Examples of nodal sets under which efficient entropy stable schemes can be constructed. This work focuses on the construction of efficient and accurate SBP-SAT terms for Gauss nodal sets.

quadrature. By choosing a tensor product Lagrange polynomial basis and $(N+1)$ point Gauss quadrature rules, we recover a Gauss collocation scheme. The high order accuracy and entropy stability of this scheme are direct results of theorems presented in [10, 11]. However, we will also present a different proof of entropy stability in one dimension for completeness.

3.1. Gauss nodes and generalized summation by parts operators. We assume the solution is collocated at $(N+1)$ quadrature points x_i with associated quadrature weights w_i. We do not make any assumptions on the points, in order to accommodate both GLL and Gauss nodes using this notation. The collocation assumption is equivalent to approximating the solution using a degree N Lagrange basis $\ell_j(x)$ defined over the $(N+1)$ quadrature points.

Let $\boldsymbol{D}$ denote the nodal differentiation matrix, and let $\boldsymbol{V}_f$ denote the $(N+1)\times 2$ matrix which interpolates polynomials at Gauss nodes to values at endpoints. These two matrices are defined entrywise as

$$\boldsymbol{D}_{ij} = \left.\frac{\partial \ell_j}{\partial x}\right|_{x=x_i}, \qquad (\boldsymbol{V}_f)_{i1} = \ell_i(-1), \qquad (\boldsymbol{V}_f)_{i2} = \ell_i(1).$$

We also introduce the diagonal matrix of quadrature weights $\boldsymbol{W}_{ij} = \delta_{ij} w_i$. In this work, $\boldsymbol{W}$ is identical to the mass matrix, whose entries are evaluated as L^2 inner products of basis functions. This equality holds so long as the basis functions are Lagrange polynomials and integrals are approximated using quadrature (either GLL or Gauss). It can be show that the mass and differentiation matrices for Gauss nodes fall under the class of generalized SBP (GSBP) operators [20].

LEMMA 3.1. *$\boldsymbol{Q} = \boldsymbol{W}\boldsymbol{D}$ satisfies the generalized summation by parts property*

$$\boldsymbol{Q} = \boldsymbol{V}_f^T \boldsymbol{B} \boldsymbol{V}_f - \boldsymbol{Q}^T, \qquad \boldsymbol{B} = \begin{bmatrix} -1 & \\ & 1 \end{bmatrix}.$$

Proof. The proof is a direct restatement of integration by parts, and can be found in [21, 22, 23]. We reproduce it here for completeness. Let $\boldsymbol{u}, \boldsymbol{v}$ be two arbitrary

vectors, and let $u(x), v(x)$ denote the polynomials whose nodal values are given by $\boldsymbol{u}, \boldsymbol{v}$. Then,

$$\begin{aligned}\boldsymbol{v}^T \boldsymbol{Q}\boldsymbol{u} = \boldsymbol{v}^T \boldsymbol{W}\boldsymbol{D}\boldsymbol{u} &= \int_{-1}^{1} \frac{\partial u}{\partial x} v = uv|_{-1}^{1} - \int_{-1}^{1} u \frac{\partial v}{\partial x} \\ &= (\boldsymbol{V}_f \boldsymbol{v})^T \boldsymbol{B}\boldsymbol{V}_f \boldsymbol{u} - (\boldsymbol{D}\boldsymbol{v})^T \boldsymbol{W}\boldsymbol{u} = \boldsymbol{v}^T \boldsymbol{V}_f^T \boldsymbol{B}\boldsymbol{V}_f \boldsymbol{u} - \boldsymbol{v}^T \boldsymbol{Q}^T \boldsymbol{u},\end{aligned}$$

where we have used that $\boldsymbol{W}$ is diagonal and that $(N+1)$-point Gauss quadrature is exact for the above integrands. Taking $\boldsymbol{v} = \boldsymbol{e}_i$ and $\boldsymbol{u} = \boldsymbol{e}_j$ for $i, j = 1, \ldots, N+1$ (where $\boldsymbol{e}_i$ denotes the ith canonical vector) completes the proof. ∎

Lemma 3.1 holds for both GLL and Gauss nodes, and switching between these two nodal sets simply results in a redefinition of the matrices $\boldsymbol{D}, \boldsymbol{V}_f$. For example, because GLL nodes include boundary points, the interpolation matrix $\boldsymbol{V}_f$ reduces to a generalized permutation matrix which extracts the nodal values associated with the left and right endpoints.

3.2. Existing entropy stable SBP-SATs for generalized SBP operators. In this section, we will review the construction of semi-discretely entropy stable discretizations. Entropy stable discretizations can be constructed by first introducing an entropy conservative scheme, then adding appropriate interface dissipation to produce an entropy inequality. The construction of entropy conservative schemes relies on the existence of an two-point (dyadic) entropy conservative flux [7].

DEFINITION 3.2. *Let $\boldsymbol{f}_S(\boldsymbol{u}_L, \boldsymbol{u}_R)$ be a bivariate function which is symmetric and consistent with the flux function $\boldsymbol{f}(\boldsymbol{u})$*

$$\boldsymbol{f}_S(\boldsymbol{u}_L, \boldsymbol{u}_R) = \boldsymbol{f}_S(\boldsymbol{u}_R, \boldsymbol{u}_L), \qquad \boldsymbol{f}_S(\boldsymbol{u}, \boldsymbol{u}) = \boldsymbol{f}(\boldsymbol{u})$$

The numerical flux $\boldsymbol{f}_S(\boldsymbol{u}_L, \boldsymbol{u}_R)$ is entropy conservative if, for entropy variables $\boldsymbol{v}_L = \boldsymbol{v}(\boldsymbol{u}_L), \boldsymbol{v}_R = \boldsymbol{v}(\boldsymbol{u}_R)$, the Tadmor "shuffle" condition is satisfied

$$(\boldsymbol{v}_L - \boldsymbol{v}_R)^T \boldsymbol{f}_S(\boldsymbol{u}_L, \boldsymbol{u}_R) = (\psi_L - \psi_R), \qquad \psi_L = \psi(\boldsymbol{v}(\boldsymbol{u}_L)), \quad \psi_R = \psi(\boldsymbol{v}(\boldsymbol{u}_R)).$$

Here, ψ is the entropy potential.

Ask DCDR to fill this in [18, 12, 19].

3.3. Decoupled SBP operators. The main goal of this work is to circumvent the all-to-all coupling of degrees of freedom introduced by the SBP-SAT terms described in Section 3.2, and will do so through the use of "decoupled" SBP operators. Decoupled SBP operators were first introduced in [10] and used to construct entropy stable schemes on simplicial elements. These operators (and simplifications under a collocation assumption) are presented in a more general setting in Appendix A. In this section, decoupled SBP operators are introduced in one dimension for either GLL or Gauss nodal sets.

Let $\boldsymbol{Q} = \boldsymbol{W}\boldsymbol{D}$. The decoupled SBP operator $\boldsymbol{Q}_N$ is defined as the block matrix

$$\boldsymbol{Q}_N = \begin{bmatrix} \boldsymbol{Q} - \frac{1}{2}\boldsymbol{V}_f^T \boldsymbol{B}\boldsymbol{V}_f & \frac{1}{2}\boldsymbol{V}_f^T \boldsymbol{B} \\ -\frac{1}{2}\boldsymbol{B}\boldsymbol{V}_f & \frac{1}{2}\boldsymbol{B} \end{bmatrix}. \tag{3.1}$$

Lemma 3.1 and straightforward computations show that $\boldsymbol{Q}_N$ also satisfies the following SBP property

LEMMA 3.3. *Let $\boldsymbol{Q}_N$ be defined through (3.1). Then,*

$$\boldsymbol{Q}_N + \boldsymbol{Q}_N^T = \begin{bmatrix} \boldsymbol{0} & \\ & \boldsymbol{B} \end{bmatrix}.$$

The matrix $\boldsymbol{Q}_N$ acts not only on volume nodes, but on both volume and surface nodes. However, $\boldsymbol{Q}_N$ does not directly define an approximation to the derivative. Let $f(x), g(x)$ denote two functions, and let $\boldsymbol{f}, \boldsymbol{g}$ denote the values of f, g at interior nodal points. We also define vectors $\boldsymbol{f}_N, \boldsymbol{g}_N$ denoting the values of f, g at both interior and boundary points

$$\boldsymbol{f}_N = \begin{bmatrix} f(x_1) \\ \vdots \\ f(x_{N+1}) \\ f(-1) \\ f(1) \end{bmatrix} = \begin{bmatrix} \boldsymbol{f} \\ \boldsymbol{f}_f \end{bmatrix}, \qquad \boldsymbol{g}_N = \begin{bmatrix} g(x_1) \\ \vdots \\ g(x_{N+1}) \\ g(-1) \\ g(1) \end{bmatrix} = \begin{bmatrix} \boldsymbol{g} \\ \boldsymbol{g}_f \end{bmatrix}. \tag{3.2}$$

Then, a polynomial approximation to $f\frac{\partial g}{\partial x}$ can be computed using $\boldsymbol{Q}_N$. Let $\boldsymbol{u}$ denote the nodal values of the polynomial $u(x) \approx f\frac{\partial g}{\partial x}$. These coefficients are computed via

$$\boldsymbol{W}\boldsymbol{u} = \begin{bmatrix} \boldsymbol{I} \\ \boldsymbol{V}_f \end{bmatrix}^T \text{diag}\left(\boldsymbol{f}_N\right) \boldsymbol{Q}_N \boldsymbol{g}_N. \tag{3.3}$$

The expression (3.3) can be rewritten in "strong" form as follows

$$\begin{aligned} \boldsymbol{u} &= \boldsymbol{W}^{-1} \begin{bmatrix} \boldsymbol{I} \\ \boldsymbol{V}_f \end{bmatrix}^T \text{diag}\left(\boldsymbol{f}_N\right) \boldsymbol{Q}_N \boldsymbol{g}_N \\ &= \text{diag}\left(\boldsymbol{f}\right) \boldsymbol{D}\boldsymbol{g} + \frac{1}{2}\text{diag}\left(\boldsymbol{f}\right) \boldsymbol{W}^{-1}\boldsymbol{V}_f^T \boldsymbol{B}\left(\boldsymbol{g}_f - \boldsymbol{V}_f\boldsymbol{g}\right) \\ &\qquad + \frac{1}{2}\boldsymbol{W}^{-1}\boldsymbol{V}_f^T \boldsymbol{B}\text{diag}\left(\boldsymbol{f}_f\right)\left(\boldsymbol{g}_f - \boldsymbol{V}_f\boldsymbol{g}\right), \end{aligned}$$

where we have used the fact that diagonal matrices commute to simplify expressions. Thus, the decoupled SBP operator $\boldsymbol{Q}_N$ can be interpreted as adding boundary corrections to the GSBP operator $\boldsymbol{D}$ in a skew-symmetric fashion.

It was shown in [10] that $\boldsymbol{u}$ is a high order accurate approximation to the quantity $f\frac{\partial g}{\partial x}$. Both the generalized SBP operator $\boldsymbol{D}$ and the expression in (3.3) involving the decoupled SBP operator recover exact derivatives of high order polynomials. However, when applied to non-polynomial functions, the decoupled SBP operator $\boldsymbol{Q}_N$ improves accuracy near the boundaries. Figure 2 illustrates this by using both operators to approximate the derivative of a Gaussian e^{-4x^2}. The decoupled SBP operator results in an improved approximation near the boundary.

3.4. An entropy stable Gauss collocation scheme based on decoupled SBP operators. We can now construct an entropy conservative Gauss collocation scheme with compact SBP-SATs using decoupled SBP operators. Let $\boldsymbol{u}_N$ and $\boldsymbol{v}(\boldsymbol{u}_N)$ denote the values of the conservative variables and entropy variables at Gauss points, respectively. We define approximate values $\widetilde{\boldsymbol{u}}_f$ of the conservative variables by interpolating the values of the entropy variables to element boundaries

$$\widetilde{\boldsymbol{u}}_f = \boldsymbol{u}(\boldsymbol{v}_f), \qquad \boldsymbol{v}_f = \boldsymbol{V}_f \boldsymbol{v}(\boldsymbol{u}_N).$$

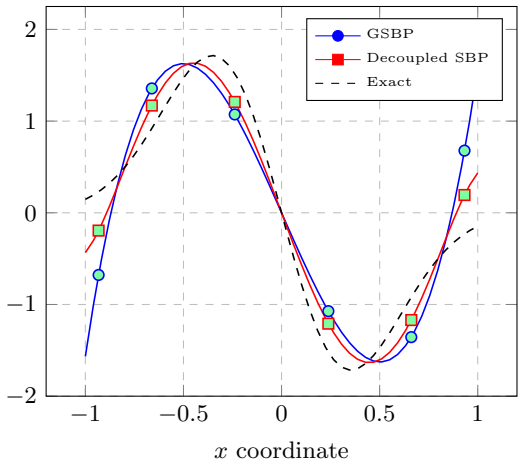


Fig. 2: Degree $N = 5$ approximations of derivatives of a Gaussian e^{-4x^2} using the generalized SBP operator $\boldsymbol{D}$ and the decoupled SBP operator $\boldsymbol{Q}_N$ via (3.3).

We now define $\widetilde{\boldsymbol{u}}$ as the concatenated vector of solution values at Gauss points and the "entropy-projected conservative variables" $\widetilde{\boldsymbol{u}}_f$. We also define $\boldsymbol{F}_S$ as the matrix of evaluations of the two-point flux $\boldsymbol{f}_S$ at combinations of values of $\widetilde{\boldsymbol{u}}$

$$(\boldsymbol{F}_S)_{ij} = \boldsymbol{f}_S\left(\widetilde{\boldsymbol{u}}_i, \widetilde{\boldsymbol{u}}_j\right), \qquad \widetilde{\boldsymbol{u}} = \begin{bmatrix} \boldsymbol{u}_N \\ \widetilde{\boldsymbol{u}}_f \end{bmatrix}.$$

THEOREM 3.4. *Let $\circ$ denote the Hadamard product, and let $\widetilde{\boldsymbol{u}}_f^+$ denote values of the entropy-projected conservative variables on neighboring elements. Then, the formulation*

$$\boldsymbol{W}\frac{\mathrm{d}\boldsymbol{u}_N}{\mathrm{dt}} + 2\begin{bmatrix} \boldsymbol{I} \\ \boldsymbol{V}_f \end{bmatrix}^T \left(\boldsymbol{Q}_N \circ \boldsymbol{F}_S\right)\boldsymbol{1} + \boldsymbol{V}_f^T\boldsymbol{B}\left(\boldsymbol{f}_S\left(\widetilde{\boldsymbol{u}}_f^+, \widetilde{\boldsymbol{u}}_f\right) - \boldsymbol{f}(\boldsymbol{u})\right) = 0 \tag{3.4}$$

satisfies the quadrature form of the semi-discrete conservation of entropy

$$\boldsymbol{1}^T\boldsymbol{W}\frac{\mathrm{d}S(\boldsymbol{u}_N)}{\mathrm{dt}} + \boldsymbol{1}^T\boldsymbol{B}\left(\boldsymbol{v}_f^T\boldsymbol{f}(\widetilde{\boldsymbol{u}}_f) - \psi_i\left(\widetilde{\boldsymbol{u}}_f\right)\right) = 0.$$

Proof. Theorem 3.4 follows from choosing either GLL or Gauss quadratures in Theorem 4 of [10]. We provide a separate illustrative proof on a two-element mesh. The extension to multiple elements is straightforward.

First, we rewrite formulation (3.4) in a skew-symmetric form. Using Lemma 3.3 and the diagonal nature of $\boldsymbol{B}$, (3.4) is equivalent to solving

$$\boldsymbol{W}\frac{\mathrm{d}\boldsymbol{u}_N}{\mathrm{dt}} + \begin{bmatrix} \boldsymbol{I} \\ \boldsymbol{V}_f \end{bmatrix}^T \left(\left(\boldsymbol{Q}_N - \boldsymbol{Q}_N^T\right) \circ \boldsymbol{F}_S\right)\boldsymbol{1} + \boldsymbol{V}_f^T\boldsymbol{B}\boldsymbol{f}_S\left(\widetilde{\boldsymbol{u}}_f^+, \widetilde{\boldsymbol{u}}_f\right) = 0 \tag{3.5}$$ ∎

Add two-element case.

A semi-discrete entropy inequality can be enforced by adding appropriate interface dissipation terms, such as Lax-Friedrichs or matrix dissipation [24]. We note that these terms must be computed in terms of $\widetilde{\boldsymbol{u}}_f$ in order to ensure a discrete dissipation of entropy [8, 10].

4. Extension to higher dimensions and curved meshes. Theorem 3.4

Describe GCL using GLL nodes [11]

Retaining entropy stability on curved meshes requires satisfaction of a discrete geometric conservation law (GCL) [4, 6, 9, 11]. For two-dimensional isoparametric mappings, the GCL is automatically satisfied. However, in three dimensions, the GCL is not guaranteed to be satisfied at the discrete level, due to the fact that geometric terms are polynomials of degree higher than N. Modifications to the computation of geometric terms are required to ensure that the GCL is satisfied at the discrete level.

For general SBP operators, the discrete GCL can be enforced through the solution of a local least squares problem [9]. We take an alternative approach and construct geometric terms using the approach of Kopriva [25]. This takes advantage of the fact that the collocation methods constructed in this work correspond to polynomial discretizations. Kopriva's construction is based on rewriting the geometric terms as the curl of some auxiliary quantity. By interpolating this auxiliary quantity prior to applying the curl,

To summarize, entropy stability curved meshes

1. Construct polynomial approximations of the geometric terms using interpolation at GLL nodes [25].
2. Interpolate the geometric terms to volume and surface quadrature points.
3. Split form.

5. Numerical results. The compressible Euler equations in d dimensions are given as follows:

$$\frac{\partial \rho}{\partial t} + \sum_{j=1}^{d} \frac{\partial \left(\rho \boldsymbol{u}_j\right)}{\partial x_j} = 0,$$

$$\frac{\partial \rho \boldsymbol{u}_i}{\partial t} + \sum_{j=1}^{d} \frac{\partial \left(\rho \boldsymbol{u}_i \boldsymbol{u}_j + p\delta_{ij}\right)}{\partial x_j} = 0, \qquad i = 1, \ldots, d$$

$$\frac{\partial E}{\partial t} + \sum_{j=1}^{d} \frac{\partial \left(\boldsymbol{u}_j (E+p)\right)}{\partial x_j} = 0.$$

Here, ρ is density, $\boldsymbol{u} = (\boldsymbol{u}_1, \ldots, \boldsymbol{u}_d)$ is the vector of velocities, and E is the total energy. The pressure p and specific internal energy ρe are given by

$$p = (\gamma - 1)\left(E - \frac{1}{2}\rho \left|\boldsymbol{u}\right|^2\right), \qquad \rho e = E - \frac{1}{2}\rho \left|\boldsymbol{u}\right|^2, \qquad \left|\boldsymbol{u}\right|^2 = \left(\sum_{j=1}^{d} \boldsymbol{u}_j^2\right).$$

There exists an infinite family of suitable convex entropies for the compressible Euler equations [26]. However, there is only a single unique entropy which appropriately treats the viscous heat conduction term in the compressible Navier-Stokes equations [27]. This entropy $S(\boldsymbol{u})$ is given in d dimensions by

$$S(\boldsymbol{u}) = -\frac{\rho s}{\gamma - 1},$$

where $s = \log\left(\frac{p}{\rho^\gamma}\right)$ is the physical specific entropy. The entropy variables in d

dimensions are given by

$$v_1 = \frac{\rho e(\gamma + 1 - s) - E}{\rho e},$$
$$v_{1+i} = \frac{\rho \boldsymbol{u}_i}{\rho e}, \qquad i = 1, \ldots, d,$$
$$v_{d+2} = -\frac{\rho}{\rho e},$$

while the conservation variables in terms of the entropy variables are given by

$$\rho = -(\rho e) v_{d+2},$$
$$\rho \boldsymbol{u}_i = (\rho e) v_{1+i}, \qquad i = 1, \ldots, d$$
$$E = (\rho e)\left(1 - \frac{\sum_{j=1}^{d} v_{1+j}^2}{2 v_{d+2}}\right),$$

where the quantities ρe and s in terms of the entropy variables are

$$\rho e = \left(\frac{(\gamma - 1)}{(-v_{d+2})^{\gamma}}\right)^{1/(\gamma-1)} e^{\frac{-s}{\gamma - 1}}, \qquad s = \gamma - v_1 + \frac{\sum_{j=1}^{d} v_{1+j}^2}{2 v_{d+2}}.$$

Explicit expressions for entropy conservative numerical fluxes in two dimensions are given by Chandrashekar [28]

$$f^1_{1,S}(\boldsymbol{u}_L, \boldsymbol{u}_R) = \{\{\rho\}\}^{\log} \{\{u\}\}, \qquad f^1_{2,S}(\boldsymbol{u}_L, \boldsymbol{u}_R) = \{\{\rho\}\}^{\log} \{\{v\}\},$$
$$f^2_{1,S}(\boldsymbol{u}_L, \boldsymbol{u}_R) = f^1_{1,S} \{\{u\}\} + p_{\text{avg}}, \qquad f^2_{2,S}(\boldsymbol{u}_L, \boldsymbol{u}_R) = f^1_{2,S} \{\{u\}\},$$
$$f^3_{1,S}(\boldsymbol{u}_L, \boldsymbol{u}_R) = f^2_{2,S}, \qquad f^3_{2,S}(\boldsymbol{u}_L, \boldsymbol{u}_R) = f^1_{2,S} \{\{v\}\} + p_{\text{avg}},$$
$$f^4_{1,S}(\boldsymbol{u}_L, \boldsymbol{u}_R) = (E_{\text{avg}} + p_{\text{avg}}) \{\{u\}\}, \qquad f^4_{2,S}(\boldsymbol{u}_L, \boldsymbol{u}_R) = (E_{\text{avg}} + p_{\text{avg}}) \{\{v\}\},$$

where we have defined the auxiliary quantities

$$p_{\text{avg}} = \frac{\{\{\rho\}\}}{2\{\{\beta\}\}}, \qquad E_{\text{avg}} = \frac{\{\{\rho\}\}^{\log}}{2\{\{\beta\}\}^{\log}(\gamma - 1)} + \frac{\|\boldsymbol{u}\|^2_{\text{avg}}}{2},$$
$$\|\boldsymbol{u}\|^2_{\text{avg}} = 2(\{\{u\}\}^2 + \{\{v\}\}^2) - \left(\{\{u^2\}\} + \{\{v^2\}\}\right).$$

Expressions for entropy conservative numerical fluxes for the three-dimensional compressible Euler equations can also be explicitly written as

$$\boldsymbol{f}_{1,S} = \begin{pmatrix} \{\{\rho\}\}^{\log} \{\{u\}\} \\ \{\{\rho\}\}^{\log} \{\{u\}\}^2 + p_{\text{avg}} \\ \{\{\rho\}\}^{\log} \{\{u\}\} \{\{v\}\} \\ \{\{\rho\}\}^{\log} \{\{u\}\} \{\{w\}\} \\ (E_{\text{avg}} + p_{\text{avg}}) \{\{u\}\} \end{pmatrix}, \qquad \boldsymbol{f}_{2,S} = \begin{pmatrix} \{\{\rho\}\}^{\log} \{\{v\}\} \\ \{\{\rho\}\}^{\log} \{\{u\}\} \{\{v\}\} \\ \{\{\rho\}\}^{\log} \{\{v\}\}^2 + p_{\text{avg}} \\ \{\{\rho\}\}^{\log} \{\{v\}\} \{\{w\}\} \\ (E_{\text{avg}} + p_{\text{avg}}) \{\{v\}\} \end{pmatrix},$$
$$\boldsymbol{f}_{3,S} = \begin{pmatrix} \{\{\rho\}\}^{\log} \{\{w\}\} \\ \{\{\rho\}\}^{\log} \{\{u\}\} \{\{w\}\} \\ \{\{\rho\}\}^{\log} \{\{v\}\} \{\{w\}\} \\ \{\{\rho\}\}^{\log} \{\{w\}\}^2 + p_{\text{avg}} \\ (E_{\text{avg}} + p_{\text{avg}}) \{\{w\}\} \end{pmatrix}.$$

with auxiliary quantities

$$p_{\text{avg}} = \frac{\{\{\rho\}\}}{2\,\{\{\beta\}\}}, \qquad E_{\text{avg}} = \frac{\{\{\rho\}\}^{\log}}{2(\gamma-1)\,\{\{\beta\}\}^{\log}} + \frac{1}{2}\,\{\{\rho\}\}^{\log}\,\|\boldsymbol{u}\|^2_{\text{avg}}$$
$$\|\boldsymbol{u}\|^2_{\text{avg}} = 2(\{\{u\}\}^2 + \{\{v\}\}^2 + \{\{w\}\}^2) - \left(\{\{u^2\}\} + \{\{v^2\}\} + \{\{w^2\}\}\right).$$

In all problems, we estimate the timestep size dt using J, J^f, and degree-dependent L^2 trace constants C_N

$$dt = C_{\text{CFL}}\frac{h}{C_N}, \qquad h = \frac{\left\|J^f\right\|_{L^\infty}}{\|J^{-1}\|_{L^\infty}},$$

where C_{CFL} is some user-defined CFL constant. For isotropic elements, the ratio of J to J^f scales as the mesh size h, while C_N captures the dependence of the maximum stable timestep on the polynomial degree N. For hexahedral elements, C_N varies depending on the choice of quadrature. It was shown in [17] that

$$C_N = \begin{cases} d\frac{N(N+1)}{2} & \text{for GLL nodes} \\ d\frac{(N+1)(N+2)}{2} & \text{for Gauss nodes} \end{cases}.$$

Thus, based on this rough estimate of the maximum stable timestep, GLL collocation schemes should be able to take a timestep which is roughly $(1+2/N)$ times larger than the maximum stable timestep for Gauss collocation schemes. We do not account for this discrepancy in this work, and instead set the timestep for both GLL and Gauss collocation schemes based on the more conservative Gauss collocation estimate of dt.

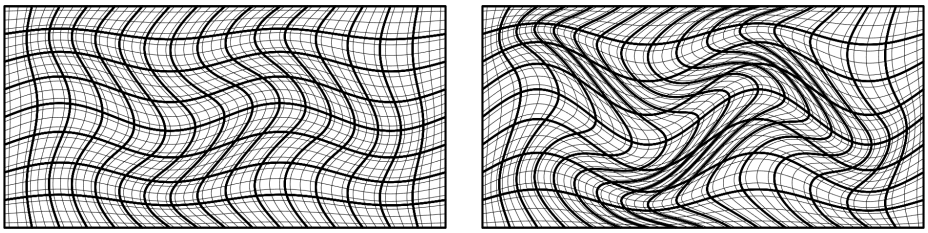
5.1. 2D isentropic vortex problem. We begin by examining high order convergence of the proposed methods in two dimensions using the isentropic vortex problem [29, 12]. The analytical solution is

$$\rho(\boldsymbol{x},t) = \left(1 - \frac{\frac{1}{2}(\gamma-1)(\beta e^{1-r(\boldsymbol{x},t)^2})^2}{8\gamma\pi^2}\right)^{\frac{1}{\gamma-1}}, \qquad p = \rho^\gamma, \tag{5.1}$$
$$u(\boldsymbol{x},t) = 1 - \frac{\beta}{2\pi}e^{1-r(\boldsymbol{x},t)^2}(y-y_0), \qquad v(\boldsymbol{x},t) = \frac{\beta}{2\pi}e^{1-r(\boldsymbol{x},t)^2}(y-y_0),$$

where u, v are the x and y velocity and $r(\boldsymbol{x},t) = \sqrt{(x-x_0-t)^2+(y-y_0)^2}$. Here, we take $x_0 = 5, y_0 = 0$ and $\beta = 5$.

We solve on a periodic rectangular domain $[0,20]\times[-5,5]$ until final time $T = 5$, and compute errors over all all solution fields. For a degree N approximation, we approximate the L^2 error using an $(N+2)$ point Gauss quadrature rule. We also examine the influence of element curvature for both GLL and Gauss collocation schemes by examining L^2 errors on a sequence of moderately and heavily warped curvilinear meshes (see Figure 3). The following results are computed using $C_{\text{CFL}} = 1/2$ and Lax-Friedrichs interface dissipation.

5.2. 3D isentropic vortex problem. As in two dimensions, we test the accuracy of the proposed scheme using an isentropic vortex solution adapted to three dimensions. We take the solution to be the extruded 2D vortex propagating in the x_2



(a) Moderately warped mesh (b) Heavily warped mesh

Fig. 3: Examples of moderate and heavily warped meshes for $N = 4, K = 16$.

direction, whose analytic expression is derived from [30]

$$\begin{aligned}
\rho(\boldsymbol{x},t) &= \left(1 - \frac{(\gamma-1)}{2}\Pi^2\right)^{\frac{1}{\gamma-1}} \\
\boldsymbol{u}(\boldsymbol{x},t) &= \Pi \boldsymbol{r}, \\
E(\boldsymbol{x},t) &= \frac{p_0}{\gamma-1}\left(1 - \frac{\gamma-1}{2}\Pi^2\right)^{\frac{\gamma}{\gamma-1}} + \frac{\rho}{2}\left|\boldsymbol{u}\right|.
\end{aligned}$$

where $\boldsymbol{u} = (u, v, w)^T$ is the velocity vector and

$$\Pi = \Pi_{\max} e^{\frac{1-\boldsymbol{r}^T\boldsymbol{r}}{2}}, \qquad \boldsymbol{r} = \begin{pmatrix} -(x_2 - c_2 - t) \\ x_1 - c_1 \\ 0 \end{pmatrix}.$$

In this problem, we take $c_1 = c_2 = 5$, $p_0 = 1/\gamma$, and $\Pi_{\max} = 0.4$. The problem is solved on the domain $[0, 10] \times [0, 20] \times [0, 10]$. As in the 2D case, we also examine the effect of curvilinear mesh warping. As in [11], we construct a curved warping of the initial Cartesian mesh by mapping nodes on each hexahedron to warped nodal positions $(\widetilde{x_1}, \widetilde{x_2}, \widetilde{x_3})$ through the transformation

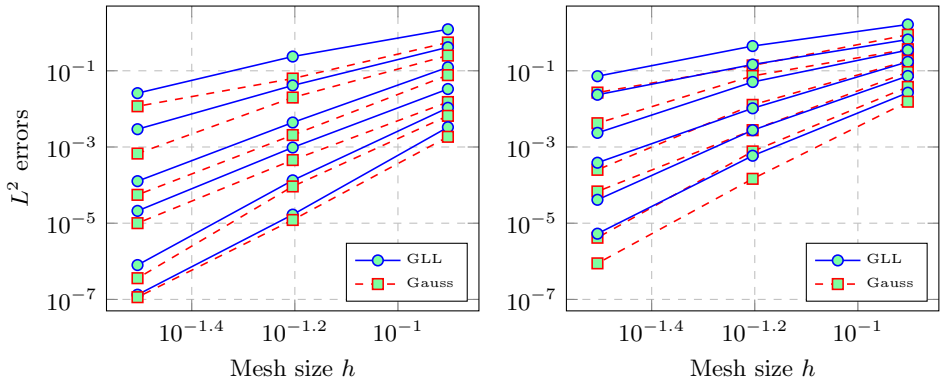
$$\widetilde{\boldsymbol{x}} = \boldsymbol{x} + \frac{1}{2}\sin\left(\pi\frac{(x_1-5)}{5}\right)\sin\left(2\pi\frac{(x_2-10)}{10}\right)\sin\left(\pi\frac{(x_3-5)}{5}\right).$$

5.3. Shock-vortex interaction.

5.4. Inviscid Taylor-Green vortex. We conclude by investigating the behavior of entropy stable Gauss collocation schemes for the inviscid Taylor–Green vortex [31, 5, 9]. This problem is posed on the periodic box $[-\pi, \pi]^3$, with initial conditions

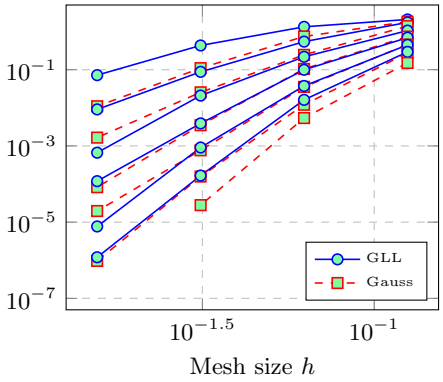
$$\begin{aligned}
\rho &= 1 \\
u &= \sin(x_1)\cos(x_2)\cos(x_3), \\
v &= -\cos(x_1)\sin(x_2)\cos(x_3), \\
w &= 0, \\
p &= \frac{100}{\gamma} + \frac{1}{16}\left(\cos(2x_1) + \cos(2x_2)\right)\left(2 + \cos(2x_3)\right).
\end{aligned}$$

The Taylor–Green vortex is used to study the transition and decay of turbulence [32]. Under an inviscid assumption, the Taylor–Green vortex develops successively smaller



(a) Affine mesh

(b) Moderately warped mesh



(c) Heavily warped mesh

Fig. 4: L^2 errors for the 2D isentropic vortex at time $T = 5$ for degree $N = 2, \ldots, 7$ GLL and Gauss collocation schemes.

scales. As a result, the solution is guaranteed to contain under-resolved features after a sufficiently large time. We study the evolution of the kinetic energy $\kappa(t)$

$$\kappa(t) = \frac{1}{|\Omega|} \int_\Omega \rho \boldsymbol{u} \cdot \boldsymbol{u} \, \mathrm{d}\boldsymbol{x},$$

as well as the kinetic energy dissipation rate $-\frac{\partial \kappa}{\partial t}$, which is approximated by differencing $\kappa(t)$.

6. Conclusion. blah

Appendix A. Decoupled SBP operators.

For general choices of quadrature and basis, decoupled projection operators involve a volume quadrature interpolation matrix $\boldsymbol{V}_q$, a face quadrature interpolation matrix $\boldsymbol{V}_f$, and a quadrature-based L^2 projection matrix $\boldsymbol{P}_q$. Let $\{\phi_j\}_{j=1}^{N_p}$ denote a set of N_p basis functions, and let $\{\boldsymbol{x}_i, \boldsymbol{w}_i\}_{i=1}^{N_q}$ denote a set of N_q volume quadrature

points and weights in d dimensions. Then, $\boldsymbol{V}_q, \boldsymbol{V}_f$ are given as

$$(\boldsymbol{V}_q)_{ij} = \phi_j(\boldsymbol{x}_i), \quad 1 \leq i, j \leq N_q,$$
$$(\boldsymbol{V}_f)_{ij} = \phi_j\left(\boldsymbol{x}_i^f\right), \quad 1 \leq i \leq N_q, \quad 1 \leq j \leq N_q^f.$$

These matrices can be used to define the quadrature-based L^2 projection matrix $\boldsymbol{P}_q$. Let $\boldsymbol{W}$ denote the diagonal matrix of quadrature weights. Then,

$$\boldsymbol{M} = \boldsymbol{V}_q^T \boldsymbol{W} \boldsymbol{V}_q, \qquad \boldsymbol{P}_q = \boldsymbol{M}^{-1} \boldsymbol{V}_q^T \boldsymbol{W}.$$

Let $\boldsymbol{D}^i$ now denote a modal differentiation matrix with respect to the ith coordinate, which maps coefficients in the basis ϕ_j to coefficients of the ith derivative. By composing this matrix with interpolation and projection matrices, one can define differencing operators $\boldsymbol{D}_q^i = \boldsymbol{V}_q \boldsymbol{D}^i \boldsymbol{P}_q$ which map values at quadrature points to values of approximate derivatives at quadrature points. It was shown in [10] that $\boldsymbol{Q}^i = \boldsymbol{W}\boldsymbol{D}_q^i$ satisfies a generalized SBP property involving the face interpolation and projection matrices $\boldsymbol{V}_f, \boldsymbol{P}_q$.

The decoupled SBP operator $\boldsymbol{Q}_N^i$ is then given as

$$\boldsymbol{Q}_N^i = \begin{bmatrix} \boldsymbol{Q}^i - \frac{1}{2}\left(\boldsymbol{V}_f \boldsymbol{P}_q\right)^T \boldsymbol{W}_f \mathrm{diag}\left(\boldsymbol{n}_i\right) \boldsymbol{V}_f \boldsymbol{P}_q & \frac{1}{2}\left(\boldsymbol{V}_f \boldsymbol{P}_q\right)^T \boldsymbol{W}_f \mathrm{diag}\left(\boldsymbol{n}_i\right) \\ -\frac{1}{2}\boldsymbol{W}_f \mathrm{diag}\left(\boldsymbol{n}_i\right) \boldsymbol{V}_f \boldsymbol{P}_q & \frac{1}{2}\boldsymbol{W}_f \mathrm{diag}\left(\boldsymbol{n}_i\right) \end{bmatrix}.$$

A straightforward computation shows that $\boldsymbol{Q}_N^i$ satisfies an SBP property [10].

It is worth noting that the form of $\boldsymbol{Q}_N^i$ does not depend on the choice of basis. So long as the approximation space spanned by the basis ϕ_j does not change, the domain and range of $\boldsymbol{Q}_N^i$ depend solely on the choice of volume and surface quadrature points.

Assuming that the solution is represented using degree N Lagrange polynomials at $(N+1)$ quadrature points, these matrices reduce down to

$$(\boldsymbol{V}_q)_{ij} = \delta_{ij}, \qquad \boldsymbol{M} = \boldsymbol{W}, \qquad \boldsymbol{P}_q = \boldsymbol{M}^{-1} \boldsymbol{V}_q^T \boldsymbol{W} = \boldsymbol{I}.$$

Todo: add.

REFERENCES

[1] Zhijian J Wang, Krzysztof Fidkowski, Rémi Abgrall, Francesco Bassi, Doru Caraeni, Andrew Cary, Herman Deconinck, Ralf Hartmann, Koen Hillewaert, Hung T Huynh, et al. High-order CFD methods: current status and perspective. *International Journal for Numerical Methods in Fluids*, 72(8):811–845, 2013.

[2] Constantine M Dafermos. *Hyperbolic conservation laws in continuum physics*. Springer, 2005.

[3] Travis C Fisher and Mark H Carpenter. High-order entropy stable finite difference schemes for nonlinear conservation laws: Finite domains. *Journal of Computational Physics*, 252:518–557, 2013.

[4] Mark H Carpenter, Travis C Fisher, Eric J Nielsen, and Steven H Frankel. Entropy Stable Spectral Collocation Schemes for the Navier–Stokes Equations: Discontinuous Interfaces. *SIAM Journal on Scientific Computing*, 36(5):B835–B867, 2014.

[5] Gregor J Gassner, Andrew R Winters, and David A Kopriva. Split form nodal discontinuous Galerkin schemes with summation-by-parts property for the compressible Euler equations. *Journal of Computational Physics*, 327:39–66, 2016.

[6] Gregor J Gassner, Andrew R Winters, Florian J Hindenlang, and David A Kopriva. The BR1 scheme is stable for the compressible Navier–Stokes equations. *Journal of Scientific Computing*, pages 1–47, 2017.

[7] Eitan Tadmor. The numerical viscosity of entropy stable schemes for systems of conservation laws. I. *Mathematics of Computation*, 49(179):91–103, 1987.

[8] Tianheng Chen and Chi-Wang Shu. Entropy stable high order discontinuous Galerkin methods with suitable quadrature rules for hyperbolic conservation laws. *Journal of Computational Physics*, 345:427–461, 2017.

[9] Jared Crean, Jason E Hicken, David C Del Rey Fernández, David W Zingg, and Mark H Carpenter. Entropy-stable summation-by-parts discretization of the Euler equations on general curved elements. *Journal of Computational Physics*, 356:410–438, 2018.

[10] Jesse Chan. On discretely entropy conservative and entropy stable discontinuous Galerkin methods. *Journal of Computational Physics*, 362:346 – 374, 2018.

[11] Jesse Chan and Lucas C Wilcox. Discretely entropy stable weight-adjusted discontinuous galerkin methods on curvilinear meshes. *arXiv preprint arXiv:1805.10934*, 2018.

[12] Jared Crean, Jason E Hicken, David C Del Rey Fernández, David W Zingg, and Mark H Carpenter. High-Order, Entropy-Stable Discretizations of the Euler Equations for Complex Geometries. In *23rd AIAA Computational Fluid Dynamics Conference*. American Institute of Aeronautics and Astronautics, 2017.

[13] Gregor J Gassner. A skew-symmetric discontinuous Galerkin spectral element discretization and its relation to SBP-SAT finite difference methods. *SIAM Journal on Scientific Computing*, 35(3):A1233–A1253, 2013.

[14] Matteo Parsani, Mark H Carpenter, Travis C Fisher, and Eric J Nielsen. Entropy Stable Staggered Grid Discontinuous Spectral Collocation Methods of any Order for the Compressible Navier–Stokes Equations. *SIAM Journal on Scientific Computing*, 38(5):A3129–A3162, 2016.

[15] David A Kopriva and Gregor Gassner. On the quadrature and weak form choices in collocation type discontinuous Galerkin spectral element methods. *Journal of Scientific Computing*, 44(2):136–155, 2010.

[16] Florian Hindenlang, Gregor J Gassner, Christoph Altmann, Andrea Beck, Marc Staudenmaier, and Claus-Dieter Munz. Explicit discontinuous Galerkin methods for unsteady problems. *Computers & Fluids*, 61:86–93, 2012.

[17] Jesse Chan, Zheng Wang, Axel Modave, Jean-Francois Remacle, and T Warburton. GPU-accelerated discontinuous Galerkin methods on hybrid meshes. *Journal of Computational Physics*, 318:142–168, 2016.

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

[19] David C Del Rey Fernández, Jason E Hicken, and David W Zingg. Simultaneous approximation terms for multi-dimensional summation-by-parts operators. *Journal of Scientific Computing*, 75(1):83–110, 2018.

[20] David C Del Rey Fernández, Pieter D Boom, and David W Zingg. A generalized framework for nodal first derivative summation-by-parts operators. *Journal of Computational Physics*, 266:214–239, 2014.

[21] Sigrun Ortleb. Kinetic energy preserving DG schemes based on summation-by-parts operators on interior node distributions. *PAMM*, 16(1):857–858, 2016.

[22] Sigrun Ortleb. A Kinetic Energy Preserving DG Scheme Based on Gauss–Legendre Points. *Journal of Scientific Computing*, 71(3):1135–1168, 2017.

[23] Hendrik Ranocha. Generalised summation-by-parts operators and variable coefficients. *Journal of Computational Physics*, 362:20 – 48, 2018.

[24] Andrew R Winters, Dominik Derigs, Gregor J Gassner, and Stefanie Walch. A uniquely defined entropy stable matrix dissipation operator for high Mach number ideal MHD and compressible Euler simulations. *Journal of Computational Physics*, 332:274–289, 2017.

[25] David A Kopriva. Metric identities and the discontinuous spectral element method on curvilinear meshes. *Journal of Scientific Computing*, 26(3):301–327, 2006.

[26] Amiram Harten. On the symmetric form of systems of conservation laws with entropy. *Journal of computational physics*, 49(1):151–164, 1983.

[27] Thomas JR Hughes, LP Franca, and M Mallet. A new finite element formulation for computational fluid dynamics: I. Symmetric forms of the compressible Euler and Navier-Stokes equations and the second law of thermodynamics. *Computer Methods in Applied Mechanics and Engineering*, 54(2):223–234, 1986.

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

[29] Chi-Wang Shu. Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws. In *Advanced numerical approximation of nonlinear hyperbolic equations*, pages 325–432. Springer, 1998.

[30] David M Williams and Antony Jameson. Nodal points and the nonlinear stability of high-order methods for unsteady flow problems on tetrahedral meshes. In *21st AIAA Computational Fluid Dynamics Conference*. American Institute of Aeronautics and Astronautics, June 2013.

[31] Geoffrey I Taylor and Albert E Green. Mechanism of the production of small eddies from large ones. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 158(895):499–521, February 1937.

[32] James DeBonis. Solutions of the Taylor-Green vortex problem using high-resolution explicit finite difference methods. In *51st AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*. American Institute of Aeronautics and Astronautics, January 2013.