

On discretely entropy conservative and entropy stable discontinuous Galerkin methods

Jesse Chan^a,

^a*Department of Computational and Applied Mathematics, Rice University, 6100 Main St, Houston, TX, 77005*

Abstract

High order methods based on diagonal-norm summation by parts operators can be shown to satisfy a discrete conservation or dissipation of entropy for nonlinear systems of hyperbolic PDEs [1, 2]. These methods can also be interpreted as nodal discontinuous Galerkin methods with diagonal mass matrices [3, 4, 5, 6]. In this work, we describe how to construct discretely entropy conservative schemes to a more general class of DG methods using flux differencing, quadrature-based projections, and specific DG differentiation operators. This approach also recovers existing methods for Burgers' equation involving dense norm and generalized SBP operators without boundary nodes [7, 8, 9]. Numerical experiments confirm the stability and high order accuracy of the proposed methods for the one-dimensional compressible Euler equations.

1. Introduction

Numerical simulations in engineering increasingly require higher accuracy without sacrificing computational efficiency. Because they are more accurate than low order methods per degree of freedom for sufficiently regular solutions, high order methods provide one avenue towards improving fidelity in numerical simulations while maintaining reasonable computational costs. High order methods which can accomodate unstructured meshes are desirable for problems with complex geometries, and among such methods, high order discontinuous Galerkin (DG) methods are particularly well-suited to the solution of time-dependent hyperbolic problems on modern computing architectures [10, 11].

The accuracy of high order methods can be attributed in part to their low numerical dissipation and dispersion compared to low order schemes [12]. This accuracy has made them advantageous for the simulation of wave propagation [10, 13]. However, while high order methods can be applied in a stable manner to linear wave propagation problems, instabilities are observed when applying them to nonlinear hyperbolic problems. This is contrast to low order schemes, whose high numerical dissipation tends to apply a stabilizing effect [14]. As a result, most high order schemes for nonlinear conservation laws typically require additional stabilization procedures, including filtering [10], slope limiting [15], artificial viscosity [16], and polynomial de-aliasing through over-integration [17]. Moreover, stabilized numerical methods can still fail, requiring user intervention or heuristic modifications to achieve non-divergent solutions.

For linear wave propagation problems, semi-discretely energy stable numerical methods can be constructed, even in the presence of curvilinear coordinates or variable coefficients [18, 19, 20, 21]. This semi-discrete stability implies that, under a stable timestep restriction (CFL condition), discrete solutions do not suffer from non-physical growth in time. However, for nonlinear systems of conservation laws, traditional methods do not admit theoretical proofs of semi-discrete stability. This was addressed for low order methods with the introduction of discretely entropy conservative and entropy stable finite volume schemes by Tadmor [22]. These schemes rely on a specific entropy conservative flux which satisfies a condition involving the entropy variables and entropy potential, and were extended to low order finite volume methods on unstructured grids in [23]. High order entropy stable methods were also developed for structured grids in [24] based on an entropy conservative essentially non-oscillatory (ENO) reconstruction.

Email address: Jesse.Chan@caam.rice.edu (Jesse Chan)

The extension of entropy conservative schemes to unstructured high order methods was done much more recently for the compressible Euler and Navier-Stokes equations in [1, 2] based on a spectral collocation approach on tensor product elements, which can also be interpreted as a mass-lumped DG spectral element (DG-SEM) scheme. The proof of entropy conservation relies on the presence of a diagonal mass matrix, the summation by parts (SBP) property [25], and a concept referred to as flux differencing. Similar entropy-stable schemes have been constructed for the shallow water and MHD equations [4, 5, 26]. More recently, high order entropy conservative and entropy stable schemes were extended to unstructured triangular meshes in [6] using the split-form methodology of [3] and multi-dimensional SBP operators [27, 28].

To the authors knowledge, the construction of unstructured high order entropy conservative and entropy stable schemes has required diagonal norm SBP operators. We refer to DG methods with these properties as diagonal norm SBP-DG methods. Entropy stable high order finite element and DG methods which do not fall under the diagonal norm SBP-DG category have been proposed [29], but the proofs are often given at the continuous level, relying on exact integration or the chain rule, neither of which holds discretely.

On the other hand, while entropy conservative schemes can be constructed by combining flux differencing with diagonal norm SBP operators, this restricts methods to specific choices of basis and quadrature. Appropriate diagonal norm SBP operators are straightforward to construct on tensor product elements based on a DG-SEM discretization. Diagonal-norm SBP operators can also be constructed for triangles and tetrahedra [30, 27, 6]; however, the number of nodal points for such operators is typically greater than the dimension of the natural polynomial approximation space. Furthermore, to the author's knowledge, appropriate point sets have only been constructed for $N \leq 4$ in three dimensions [31], and the construction of high order diagonal norm SBP-DG methods has not yet been performed for uncommon elements such as pyramids [32].

This work focuses on the construction of entropy conservative high order DG schemes for systems of conservation laws. In order to generalize beyond diagonal norm SBP-DG methods, we will consider non-lumped (dense) mass matrices and more general quadrature rules (e.g. rules without boundary points, rules with more points than the dimension of the corresponding approximation space), which are also related to dense norm and generalized SBP operators [7, 8, 9, 33]. The work presented in this manuscript is intended to serve as a stepping stone to the construction of entropy stable DG methods on general elements in two and three dimensions. We will present proofs of discrete entropy stability in terms of integrals and L^2 projections, which we assume are computed using an appropriate quadrature rule. We stress that, while this work adopts a continuous notation involving integrals rather than discrete sums, the proofs rely only on properties of quadrature-based integration and quadrature-based L^2 projection.

The outline of the paper is as follows: Section 2 will briefly review the construction of entropy conservative diagonal norm SBP-DG methods on a single element. Section 3 will describe how to construct analogous entropy conservative methods on single element in a continuous setting. Section 4 will discuss extensions to multiple elements, including comparisons of different coupling terms and entropy stable fluxes. Finally, Section 5 presents numerical results which verify the high order accuracy and discrete entropy conservation of the proposed methods. We note that, for simplicity and clarity of presentation, the proposed methods, proofs, and numerical experiments are given in one dimension. However, the extension to multiple dimensions (including simplicial elements) is straightforward and will be described in an upcoming manuscript.

2. Entropy conservative diagonal norm SBP-DG methods

We will begin by reviewing continuous entropy theory and existing high order methods which are provably entropy conservative at the discrete level. These methods were introduced in [1, 2] for tensor product elements, though we adopt more general methods of proof introduced recently in [34, 6]. The proofs are based on diagonal norm SBP operators, which can be derived from a nodal discontinuous Galerkin framework through the collocation of nodal points and quadrature points [25].

2.1. Entropy stability for systems of hyperbolic PDEs

We consider systems of nonlinear conservation laws in one dimension with n variables

$$\frac{\partial \mathbf{u}}{\partial t} + \frac{\partial \mathbf{f}(\mathbf{u})}{\partial x} = 0, \quad \mathbf{u}(x, t) = (u_1(x, t), \dots, u_n(x, t)). \quad (1)$$

where the fluxes $\mathbf{f}(\mathbf{u})$ are smooth functions of the vector of conservative variables $\mathbf{u}(x, t)$. The Jacobian matrix $\mathbf{A}(\mathbf{u})$ is defined entrywise as

$$(\mathbf{A}(\mathbf{u}))_{ij} = \left(\frac{\partial \mathbf{f}(\mathbf{u})}{\partial u_j} \right)_i.$$

We are interested in systems for which there exists a convex entropy function $U(\mathbf{u})$ such that

$$U''(\mathbf{u})\mathbf{A}(\mathbf{u}) = (U''(\mathbf{u})\mathbf{A}(\mathbf{u}))^T. \quad (2)$$

For systems with convex entropy functions, one can define entropy variables $\mathbf{v} = U'(\mathbf{u})$. The convexity of the $U(\mathbf{u})$ guarantees that the mapping between conservative and entropy variables is invertible.

It can be shown (see, for example, [35]) that (2) is equivalent to the existence of an entropy flux function $F(\mathbf{u})$ such that

$$\mathbf{v}^T \frac{\partial \mathbf{f}}{\partial \mathbf{u}} = \frac{\partial F(\mathbf{u})}{\partial \mathbf{u}}^T.$$

The flux function and entropy flux function are further related through the so-called entropy potential ψ

$$\psi(\mathbf{v}) = \mathbf{v}^T \mathbf{f}(\mathbf{u}(\mathbf{v})) - F(\mathbf{u}(\mathbf{v})), \quad \psi'(\mathbf{v}) = \mathbf{f}(\mathbf{u}(\mathbf{v})).$$

When \mathbf{u} is smooth, multiplying (1) on the left by $\mathbf{v}^T = U'(\mathbf{u})^T$, applying the definition of the entropy flux and using the chain rule yields the conservation of entropy

$$\frac{\partial U(\mathbf{u})}{\partial t} + \frac{\partial F(\mathbf{u})}{\partial x} = 0.$$

More generally, it can be shown that physically relevant solutions of (1) (defined as the limiting solution for an appropriately defined vanishing viscosity) satisfy the entropy inequality

$$\frac{\partial U(\mathbf{u})}{\partial t} + \frac{\partial F(\mathbf{u})}{\partial x} \leq 0.$$

We assume now that the domain is the interval $[-1, 1]$. Integrating the entropy inequality over this interval and using the definition of the entropy potential then yields

$$\int_{-1}^1 \frac{\partial U(\mathbf{u})}{\partial t} + (\mathbf{v}^T \mathbf{f}(\mathbf{u}(\mathbf{v})) - \psi(\mathbf{v})) \Big|_{-1}^1 \leq 0. \quad (3)$$

The focus of this work is the construction of methods which satisfy a discrete analogue of this entropy inequality. We begin by reviewing existing discretely entropy conservative methods based on diagonal norm SBP operators.

2.2. Entropy conservative methods based on diagonal-norm SBP operators

Let $x_i \in [-1, 1]$ denote the $(N + 1)$ point Gauss-Lobatto-Legendre (GLL) quadrature rule with positive weights w_1, \dots, w_{N+1} , which integrates exactly polynomials of degree $(2N - 1)$. Let $\ell_i(x)$ denote the nodal Lagrange basis defined at GLL points, and define the differentiation matrix \mathbf{D} be defined as

$$D_{ij} = \frac{\partial \ell_j(x_i)}{\partial x}.$$

The matrix \mathbf{D} maps from nodal values of a polynomial to nodal values of its derivative. We also define the mass matrix \mathbf{M}

$$\mathbf{M}_{ij} = \int_{-1}^1 \ell_i(x) \ell_j(x) dx \approx \sum_k \ell_i(x_k) \ell_j(x_k) w_k = \delta_{ij} w_i,$$

where the equality holds only approximately due to the inexactness of GLL quadrature for polynomials of degree $2N$. For this choice of quadrature, the Kronecker property $\ell_j(x_i) = \delta_{ij}$ implies that \mathbf{M} is a diagonal matrix whose entries are simply the quadrature weights.

The mass and derivative matrices may be used to define the diagonal norm SBP operator $\mathbf{S} = \mathbf{M}\mathbf{D}$, with the summation-by-parts property

$$\mathbf{S} = \mathbf{B} - \mathbf{S}^T, \quad \mathbf{B}_{ij} = \begin{cases} -1 & i = j = 1 \\ 1 & i = j = N + 1 \\ 0 & \text{otherwise} \end{cases}.$$

This property is simply a restatement of integration by parts, using that the derivative of $\ell_j(x)$ is a polynomial of degree $(N - 1)$ and that GLL quadrature is exact for polynomials of degree $(2N - 1)$.

We seek a polynomial approximation to the solution $\mathbf{u}(x)$ of (1). A DG formulation in the “strong form” [10] is given as

$$\int_{-1}^1 \left(\frac{\partial \mathbf{u}}{\partial t} + \frac{\partial \mathbf{f}(\mathbf{u})}{\partial x} \right) \mathbf{w} + (\mathbf{f}^* - \mathbf{f}(\mathbf{u})) \mathbf{w}|_{-1}^1 = 0, \quad \forall \mathbf{w} \in V_h,$$

where $\mathbf{w}(x)$ is a vector-valued test function. Here, \mathbf{f}^* is some numerical flux used for the weak imposition of boundary or continuity conditions.

We derive a DG-SEM approximation by evaluating the above integrals using GLL quadrature. We furthermore assume that the vector solution is approximated using a nodal basis at GLL points

$$\mathbf{u}(x, t) = \sum_{j=1}^{N+1} \mathbf{u}_j(t) \ell_j(x).$$

The resulting semi-discrete system yields a nodal collocation scheme for each (vector-valued) degree of freedom \mathbf{u}_i

$$\frac{\partial \mathbf{u}_i}{\partial t} + \sum_{j=1}^{N+1} \left((\mathbf{D}_{ij} \otimes \mathbf{I}) \mathbf{f}(\mathbf{u}_j) + ((\mathbf{M}^{-1} \mathbf{B})_{ij} \otimes \mathbf{I}) (\mathbf{f}_j^* - \mathbf{f}(\mathbf{u}_j)) \right) = 0, \quad i = 1, \dots, N + 1,$$

where \mathbf{f}^* is the vector whose entries consist of evaluations of the numerical flux at boundary nodes and zeros at interior nodes.

This formulation conserves the quantities \mathbf{u}_i , but does not satisfy a discrete analogue of the statement of entropy conservation (3) except in specific cases. Entropy conservative schemes may be constructed based on the SBP property and a “flux differencing” approach [3] involving a definition of entropy conservative/stable fluxes given by Tadmor [22].

Definition 1. Let $\mathbf{f}_S(\mathbf{u}_L, \mathbf{u}_R)$ be a bivariate function which is symmetric and consistent with the flux function $\mathbf{f}(\mathbf{u})$

$$\mathbf{f}_S(\mathbf{u}_L, \mathbf{u}_R) = \mathbf{f}_S(\mathbf{u}_R, \mathbf{u}_L), \quad \mathbf{f}_S(\mathbf{u}, \mathbf{u}) = \mathbf{f}(\mathbf{u})$$

The numerical flux $\mathbf{f}_S(\mathbf{u}_L, \mathbf{u}_R)$ is entropy conservative if, for entropy variables $\mathbf{v}_L = \mathbf{v}(\mathbf{u}_L)$, $\mathbf{v}_R = \mathbf{v}(\mathbf{u}_R)$

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

Similarly, a flux $\mathbf{f}_S(\mathbf{u}_L, \mathbf{u}_R)$ is referred to as entropy stable if $(\mathbf{v}_L - \mathbf{v}_R)^T \mathbf{f}_S(\mathbf{u}_L, \mathbf{u}_R) \leq (\psi_L - \psi_R)$.

An entropy conservative diagonal norm SBP-DG method can then be derived by modifying the flux derivative as follows

$$\frac{\partial \mathbf{u}_i}{\partial t} + \sum_{j=1}^{N+1} \left((2\mathbf{D}_{ij} \otimes \mathbf{I}) \mathbf{f}_S(\mathbf{u}_i, \mathbf{u}_j) + \left((\mathbf{M}^{-1}\mathbf{B})_{ij} \otimes \mathbf{I} \right) (\mathbf{f}_j^* - \mathbf{f}(\mathbf{u}_j)) \right) = 0. \quad (4)$$

The replacement of the derivative of the flux function with flux differencing allows one to prove that (4) conserves a discrete entropy [34, 6]. We reproduce these proofs here, as we will refer to them in following sections.

Theorem 1. *The nodal collocation method (4) is discretely entropy conservative in the sense that*

$$\mathbf{1}^T \left(\mathbf{M} \frac{\partial U(\mathbf{u})}{\partial t} + \mathbf{B} (\mathbf{v}^T \mathbf{f}^* - \boldsymbol{\psi}) \right) = 0,$$

where $\mathbf{1}$ denotes the vector of ones and $\boldsymbol{\psi}_i = \psi(\mathbf{v}_i)$ is the entropy potential evaluated in terms of the nodal values \mathbf{u}_i .

Proof. In this proof, we assume the matrices \mathbf{M} , \mathbf{D} , \mathbf{B} are understood to act component-wise on the vector of solution degrees of freedom \mathbf{u} and drop the explicit Kronecker product for simplicity of notation. Let $\mathbf{v}_i = \mathbf{v}(\mathbf{u}_i)$ denote nodal values of the entropy variables. We multiply (4) by $(\mathbf{M}\mathbf{v})^T$. Using the fact that \mathbf{M} is diagonal, the time derivative term can be manipulated to

$$\mathbf{v}^T \mathbf{M} \frac{\partial \mathbf{u}}{\partial t} = \sum_{i=1}^{N+1} \mathbf{M}_{ii} \mathbf{v}_i^T \frac{\partial \mathbf{u}_i}{\partial t} = \sum_{i=1}^{N+1} \mathbf{M}_{ii} \frac{\partial U(\mathbf{u}_i)}{\partial t} = \mathbf{1}^T \mathbf{M} \frac{\partial U(\mathbf{u})}{\partial t},$$

where we have used continuity in time to arrive at the point-wise relation

$$\mathbf{v}^T \frac{\partial \mathbf{u}}{\partial t} = \frac{\partial U(\mathbf{u})}{\partial \mathbf{u}}^T \frac{\partial \mathbf{u}}{\partial t} = \frac{\partial U(\mathbf{u})}{\partial t}.$$

The spatial derivative term can be manipulated using the summation by parts property $\mathbf{M}\mathbf{D} = \mathbf{S} = \mathbf{B} - \mathbf{S}^T$

$$\begin{aligned} \sum_{i,j=1}^{N+1} \mathbf{v}_i^T 2\mathbf{M}\mathbf{D}_{ij} \mathbf{f}_S(\mathbf{u}_i, \mathbf{u}_j) &= \sum_{i,j=1}^{N+1} \mathbf{v}_i^T (\mathbf{S} + \mathbf{B} - \mathbf{S}^T)_{ij} \mathbf{f}_S(\mathbf{u}_i, \mathbf{u}_j) \\ &= \sum_{i,j=1}^{N+1} \mathbf{S}_{ij} \mathbf{v}_i^T \mathbf{f}_S(\mathbf{u}_i, \mathbf{u}_j) - \mathbf{S}_{ji} \mathbf{v}_i^T \mathbf{f}_S(\mathbf{u}_i, \mathbf{u}_j) + \mathbf{B}_{ij} \mathbf{v}_i^T \mathbf{f}_S(\mathbf{u}_i, \mathbf{u}_j). \end{aligned} \quad (5)$$

Rearranging indices and using symmetry of $\mathbf{f}_S(\mathbf{u}_i, \mathbf{u}_j)$ then yields that

$$\begin{aligned} \sum_{i,j=1}^{N+1} \mathbf{S}_{ij} \mathbf{v}_i^T \mathbf{f}_S(\mathbf{u}_i, \mathbf{u}_j) - \mathbf{S}_{ji} \mathbf{v}_i^T \mathbf{f}_S(\mathbf{u}_i, \mathbf{u}_j) &= \sum_{i,j=1}^{N+1} \mathbf{S}_{ij} (\mathbf{v}_i - \mathbf{v}_j)^T \mathbf{f}_S(\mathbf{u}_i, \mathbf{u}_j) \\ &= \sum_{i,j=1}^{N+1} \mathbf{S}_{ij} (\boldsymbol{\psi}_i - \boldsymbol{\psi}_j) = -\mathbf{1}^T \mathbf{S} \boldsymbol{\psi} = -\mathbf{1}^T \mathbf{B} \boldsymbol{\psi}, \end{aligned}$$

where we have used $\mathbf{S}\mathbf{1} = 0$ and the summation by parts property in the last line.

Using that \mathbf{B} is diagonal, combining the latter term in (5) with boundary terms involving the numerical flux yields

$$\sum_{i,j=1}^{N+1} \mathbf{B}_{ij} \mathbf{v}_i^T \mathbf{f}_S(\mathbf{u}_i, \mathbf{u}_j) + \mathbf{v}_i^T \mathbf{B}_{ij} (\mathbf{f}_j^* - \mathbf{f}(\mathbf{u}_j)) = \mathbf{1}^T \mathbf{B} (\mathbf{v}^T \mathbf{f}^*),$$

where we have used the consistency of $\mathbf{f}_S(\mathbf{u}, \mathbf{u}) = \mathbf{f}(\mathbf{u})$. Putting these components together results in a discrete statement of the conservation of entropy (3). \square

We note that, unlike the continuous proof of entropy conservation, the proof of discrete entropy conservation avoids the use of the chain rule, which does not hold in general at the discrete level. This proof of entropy conservation can also be extended beyond a single element by taking \mathbf{f}^* to be the entropy conservative flux $f_S(u_L, u_R)$ at the interface between two elements [2, 34, 6]. Similarly, employing an entropy stable flux at element interfaces results in an entropy stable method which satisfies a global entropy inequality.

It is important to emphasize that conservation of entropy does not, in general, imply stability of the numerical scheme unless the solution satisfies additional constraints. For example, the transformation between conservative and entropy variables for the compressible Euler and Navier-Stokes equations is well-defined only if the density and pressure are positive, and steps must be taken in any numerical scheme to guarantee that the discrete solution satisfies such constraints. For DG methods, this is most commonly done using bound and positivity-preserving limiters [36, 37].

3. Entropy conservative DG methods on a single element

In this section, we focus on constructing entropy conservative DG schemes for general choices of basis and quadrature. These include (as special cases) dense norm SBP operators and generalized SBP (GSBP) operators which do not contain boundary points. Difficulties in generalizing entropy conservative SBP-DG schemes present themselves primarily in two areas: the assumption of a diagonal mass matrix, which is used in the proof of entropy stability, and the generalization of flux differencing to a continuous formulation. We address both areas of difficulty by extending the entropy conservative DG-SEM scheme (4) from a discrete formulation to a continuous formulation involving flux differencing. We prove that the continuous scheme is entropy conservative (stable) on a single element in the following sections, and describe how to apply the proposed methods to produce energy conservative formulations of Burgers' equation as an illustrative example.

3.1. Notation

We assume that the domain Ω is a finite interval, which is decomposed into non-overlapping elements D^k of size h . We define the reference element $\hat{D} = [-1, 1]$, such that D^k is the image of \hat{D} under an affine mapping $x = \Phi^k(\hat{x})$. We define an approximation space using degree N polynomials on the reference element

$$P^N(\hat{D}) = \left\{ \hat{x}^i, \quad \hat{x} \in \hat{D}, \quad 0 \leq i \leq N \right\}.$$

where we define $N_p = \dim(P^N(\hat{D}))$ (with $N_p = N + 1$ in one dimension). We approximate the solution over a physical element by mapping $P^N(\hat{D})$ to D^k under Φ^k , and define the global approximation space V_h as the direct sum of local approximation spaces

$$P^N(D^k) = \Phi^k \circ P^N(\hat{D}), \quad V_h = \bigoplus_k P^N(D^k).$$

We define the L^2 norm and inner products over the element D^k and the surface of the element ∂D^k

$$(\mathbf{u}, \mathbf{v})_{D^k} = \int_{D^k} \mathbf{u} \cdot \mathbf{v} \, dx = \int_{\hat{D}} \mathbf{u} \cdot \mathbf{v} J^k \, d\hat{x}, \quad \|\mathbf{u}\|_{D^k}^2 = (\mathbf{u}, \mathbf{u})_{D^k}, \quad \langle \mathbf{u}, \mathbf{v} \rangle_{\partial D^k} = \int_{\partial D^k} \mathbf{u} \cdot \mathbf{v} \, dx,$$

where J^k is the determinant of the Jacobian of Φ^k . We also define the local L^2 projection operator $\Pi_N : L^2(D^k) \rightarrow P^N(D^k)$ such that

$$(\Pi_N u, v)_{D^k} = (u, v)_{D^k}, \quad \forall v \in P^N(D^k).$$

In one dimension, D^k is some interval and the L^2 inner product over the surface $\{-1, 1\}$ reduces to

$$\langle u, v \rangle_{\partial D^k} = u(x_L)v(x_L) + u(x_R)v(x_R).$$

However, the presented proofs in this work will not directly utilize this simplification in order to remain generalizable to higher dimensions in a straightforward manner.

In typical implementations, integrals and inner products are approximated using a quadrature rule consisting of $N_q \geq N_p$ points \hat{x}_i and weights w_i over the reference element \hat{D} , such that

$$(u, v)_{D^k} := \sum_{i=1}^{N_q} u(\hat{x}_i) v(\hat{x}_i) w_i J^k.$$

This also defines an analogous quadrature-based L^2 projection operator. We emphasize that, throughout this work, we will only use properties of quadrature-based inner products and projection operators. We assume only that the quadrature is sufficiently accurate to integrate degree $(2N - 1)$ polynomials, such that integration by parts holds for any two polynomials on the reference element, while allowing for quadratures with an arbitrary number of points.

Finally, we define the jump and average of discontinuous functions across element interfaces. Let $D^{k,+}$ denote the neighboring element of across a face f of D^k , and let u^+ denote the values of u on $D^{k,+}$. The jump of u across f on D^k is then defined as

$$[[u]] = u^+ - u, \quad \{\{u\}\} = \frac{u^+ + u}{2}.$$

The jump and average of vector fields are defined component-wise using the jumps and averages of components. When the face f lies on the boundary, we define $u^+ = u$ such that $[[u]] = 0$ and $\{\{u\}\} = u$ on $\partial\Omega$.

3.2. A continuous interpretation of flux differencing

In this section, we offer a continuous interpretation of flux differencing [1, 34, 6], which also encompasses the split form methodology proposed in [3]. This interpretation will guide the construction of entropy conservative DG schemes.

Assume that $\mathbf{f}_S(\mathbf{u}_L, \mathbf{u}_R)$ is the symmetric and consistent two-point flux defined in (1). Consider the bilinear function

$$\mathbf{f}_S(\mathbf{u}(x), \mathbf{u}(y)), \quad x, y \in \mathbb{R}.$$

Flux differencing can be interpreted as differentiating $\mathbf{f}_S(\mathbf{u}(x), \mathbf{u}(y))$ along one coordinate x , then evaluating the result with $y = x$

$$\frac{\partial \mathbf{f}(\mathbf{u}(x))}{\partial x} \implies 2 \left. \frac{\partial \mathbf{f}_S(\mathbf{u}(x), \mathbf{u}(y))}{\partial x} \right|_{y=x}. \quad (6)$$

It was shown in [6] that flux differencing provides a consistent approximation of the flux derivative $\frac{\partial \mathbf{f}(\mathbf{u})}{\partial x}$.

It is helpful to first understand flux differencing as a method for recovering split formulations. We illustrate this using the one-dimensional Burgers equation on domain $[-1, 1]$, which is given in conservative form as

$$\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = 0, \quad f(u) = \frac{1}{2}u^2.$$

Assuming smooth u , Burgers' equation conserves a square entropy $U(\mathbf{u}) = u^2/2$. This can be shown by multiplying by u , integrating over $[-1, 1]$, using the chain rule in time and space, and integrating by parts to arrive at

$$\frac{1}{2} \frac{\partial}{\partial t} \int_{-1}^1 u^2 = - \left. \frac{u^3}{3} \right|_{-1}^1. \quad (7)$$

An alternative rewriting of the Burgers' equation is the split formulation [25, 9], which may be also be recovered using flux differencing under the following two-point flux

$$f_S(u_L, u_R) = \frac{1}{6}(u_L^2 + u_L u_R + u_R^2).$$

Applying (6) yields the split form of Burgers' equation

$$2 \frac{\partial f_S(u(x), u(y))}{\partial x} \Big|_{y=x} = \frac{1}{3} \frac{\partial (u(x)^2 + u(x)u(y) + u(y)^2)}{\partial x} \Big|_{y=x} = \frac{1}{3} \left(\frac{\partial u^2}{\partial x} + u \frac{\partial u}{\partial x} \right).$$

We note that recovering the split-form of Burgers' equation relies on the property that

$$\frac{\partial (u(y)^2)}{\partial x} = u(y)^2 \frac{\partial (1)}{\partial x} = 0.$$

Thus, recovery of the split form can be achieved by using flux differencing in combination with any differential operator D such that $D1 = 0$.

3.3. Discrete differential operators and weighted integration by parts

While both the conservative and split forms of Burgers' equation are equivalent at the continuous level (assuming smoothness of u), they are not equivalent at the discrete level. Suppose now that the solution is approximated by $u \in P^N(\hat{D})$ on the reference element. Since $f(u) = u^2/2 \notin P^N(\hat{D})$ for general choices of u , it is not possible to differentiate $f(u)$ exactly without prior knowledge of the form of the nonlinearity. Indeed, it is more common to first project $f(u)$ to $P^N(\hat{D})$, then differentiate, such that the discrete scheme is equivalent to solving for $u \in P^N(\hat{D})$ such that

$$\int_{-1}^1 \left(\frac{\partial u}{\partial t} + \frac{\partial \Pi_N f(u)}{\partial x} \right) w \, dx = 0, \quad \forall w \in P^N(\hat{D}).$$

This scheme does not, in general, conserve the square entropy $u^2/2$.

We can derive a semi-discrete Galerkin scheme which conserves energy using flux differencing and a discrete differentiation operator D_N , which is inspired by the SBP operators introduced in [6]. Define $H^1(\hat{D})$ as the Sobolev space over \hat{D}

$$\|u\|_{H^1(\hat{D})}^2 = \|u\|_{L^2(\hat{D})}^2 + \left\| \frac{\partial u}{\partial x} \right\|_{L^2(\hat{D})}^2, \quad H^1(\hat{D}) = \left\{ u \in L^2(\hat{D}), \quad \|u\|_{H^1(\hat{D})} < \infty \right\}. \quad (8)$$

Because functions $u \in H^1(\hat{D})$ admit a well-defined boundary trace [38], we can define the following discrete differential operator as a map from H^1 to P^N :

Definition 2. Let $w(x)$ be a continuous and bounded function on \hat{D} . Define $D_N : H^1(\hat{D}) \rightarrow P^N(\hat{D})$ such that, for $u \in H^1(\hat{D})$,

$$\begin{aligned} (D_N u, vw)_{\hat{D}} &= \left(\frac{\partial \Pi_N u}{\partial x}, vw \right)_{\hat{D}} \\ &+ \frac{1}{2} \langle u - \Pi_N u, v w n_x \rangle_{\partial \hat{D}} + \frac{1}{2} \langle u - \Pi_N u, \Pi_N(vw) n_x \rangle_{\hat{D}}, \quad \forall v \in P^N(\hat{D}), \end{aligned} \quad (9)$$

where n_x is the x -component of the outward normal vector.

This operator satisfies a form of integration by parts, which is summarized by the following lemma:

Lemma 1. The linear operator D_N satisfies a “weighted” integration by parts property with respect the L^2 inner product and any bounded real-valued function $w(x)$

$$(D_N u, wv)_{\hat{D}} = \langle u, w v n_x \rangle_{\partial \hat{D}} - (u, D_N(wv))_{\hat{D}}.$$

Proof. Integrating the left hand side by parts gives

$$-\left(\frac{\partial \Pi_N(vw)}{\partial x}, u\right)_{\hat{D}} + \frac{1}{2} \langle u - \Pi_N u, vw n_x \rangle_{\partial \hat{D}} + \frac{1}{2} \langle u + \Pi_N u, \Pi_N(vw) n_x \rangle_{\partial \hat{D}}, \quad \forall v \in P^N(\hat{D}).$$

Rearranging boundary terms yields

$$-\left(\frac{\partial \Pi_N(vw)}{\partial x}, u\right)_{\hat{D}} + \frac{1}{2} \langle vw + \Pi_N(vw), u n_x \rangle_{\partial \hat{D}} - \frac{1}{2} \langle vw - \Pi_N(vw), \Pi_N u n_x \rangle_{\partial \hat{D}}.$$

Adding and subtracting $\langle vw, u n_x \rangle_{\partial \hat{D}}$ completes the proof. \square

Direct computations also show that $(D_N 1, vw)_{\hat{D}} = 0$ for all bounded weights $w(x)$ and $\forall v \in P^N(\hat{D})$. Thus, we can apply (6) with D_N to construct a discrete formulation for Burgers' equation

$$\left(\frac{\partial u}{\partial t}, v\right) + \frac{1}{3} (D_N u^2 + u D_N u, v) = \left(\frac{\partial u}{\partial t}, v\right)_{\hat{D}} + \frac{1}{3} (D_N u^2, v)_{\hat{D}} + \frac{1}{3} (D_N u, uv)_{\hat{D}}, \quad \forall v \in P^N(\hat{D}). \quad (10)$$

To show this formulation is discretely energy conservative, we take $v = u$ and use Lemma 1 to cancel volume terms, yielding

$$\frac{1}{2} \frac{\partial}{\partial t} (u, u)_{\hat{D}} = \frac{1}{2} \frac{\partial}{\partial t} \|u\|_{\hat{D}}^2 = -\frac{1}{3} \langle u^2, u \rangle_{\partial \hat{D}},$$

which is a discrete analogue of the conservation of energy (7) for smooth solutions. This is equivalent to the skew-symmetric split formulation described in [25]; however, as noted in [6], it can be derived from flux differencing without explicit knowledge of the skew form. We will show later in Section 4.2.2 that a natural extension of this formulation to multiple elements recovers the weak form of the scheme introduced in [9].

3.4. Conservation of entropy on one element

While flux differencing can be used to recover split formulations, it can also be used to construct entropy stable formulations which do not correspond to skew-symmetric splittings. In this section we will show that, for any sufficiently accurate quadrature rule and appropriate numerical flux function, it is possible to construct discretizations which satisfy a discrete equivalent of conservation of entropy (3).

For the proof, we will need the following assumption:

Assumption 1. *The numerical flux function \mathbf{f}_S admits an absolutely convergent expansion of the form*

$$\mathbf{f}_S(x, y) = \sum_{i=1}^{\infty} \mathbf{f}_i(x) \mathbf{g}_i(y),$$

where $\mathbf{f}_i, \mathbf{g}_i$ are real and bounded for appropriate values of x, y over each element.

The precise definition of what constitutes “appropriate values of x, y ” depends on the physical problem. For example, the flux functions for the compressible Euler equations require positivity constraints on the inputs, which are outlined in more detail in Section 5. This assumption is satisfied, for example, by \mathbf{f}_S which are real analytic and thus admit a Taylor expansion in the two arguments. In Appendix A, we argue that this assumption also holds for the flux functions \mathbf{f}_S considered in this work for the Burgers' and compressible Euler equations, assuming the solution satisfies physical bound constraints. However, we emphasize that this assumption is not required in the computation implementations of the proposed methods, and that it may also be possible to relax these conditions on \mathbf{f}_S .

We now introduce a discretization for the nonlinear system of conservation laws (1) on the reference element \hat{D} : find $\mathbf{u} \in P^N(\hat{D})$ such that

$$\left(\frac{\partial \mathbf{u}}{\partial t}, \mathbf{w}\right)_{\hat{D}} + \left((2D_N \mathbf{f}_S(\mathbf{u}_v(x), \mathbf{u}_v(y)))|_{y=x}, \mathbf{w}\right)_{\hat{D}} + \langle \mathbf{f}^* - \mathbf{f}(\mathbf{u}_v), \mathbf{w} n_x \rangle_{\partial \hat{D}} = 0, \quad \mathbf{w} \in P^N(\hat{D}), \quad (11)$$

where \mathbf{f}^* is a numerical flux used to impose boundary conditions weakly, and $\mathbf{u}_v(x), \mathbf{u}_v(y)$ are evaluations of \mathbf{u} in terms of the projections of entropy variables $\mathbf{v}(\mathbf{u}(x))$

$$\mathbf{u}_v(x) = \mathbf{u}((\Pi_N \mathbf{v})(x)). \quad (12)$$

Theorem 2. *Let Assumption 1 hold on \widehat{D} . The formulation (11) is semi-discretely entropy conservative in the sense that*

$$\left(\frac{\partial U(\mathbf{u})}{\partial t}, 1 \right)_{\widehat{D}} = \langle \psi, 1n_x \rangle_{\partial \widehat{D}} - \langle \mathbf{f}^*, \Pi_N \mathbf{v} n_x \rangle_{\partial \widehat{D}}.$$

Proof. We take the test function \mathbf{w} to be the projection of the entropy variables. On the left hand side, we may use the fact that, under a method of lines discretization,

$$\mathbf{u}(x, t) = \sum_{j=1}^{N_p} \mathbf{u}_j(t) \phi_j(x), \quad \frac{\partial \mathbf{u}(x, t)}{\partial t} \in P^N(\widehat{D}).$$

Thus, testing with $\Pi_N \mathbf{v}$ and assuming continuity in time yields

$$\left(\frac{\partial \mathbf{u}}{\partial t}, \Pi_N \mathbf{v}(\mathbf{u}) \right)_{\widehat{D}} = \left(\frac{\partial \mathbf{u}}{\partial t}, \mathbf{v}(\mathbf{u}) \right)_{\widehat{D}} = \left(\frac{\partial U(\mathbf{u})}{\partial t}, 1 \right)_{\widehat{D}}, \quad \forall \mathbf{u} \in P^N(\widehat{D}).$$

For the spatial derivative, we expand \mathbf{f}_S out using Assumption 1 and apply Lemma 1

$$\begin{aligned} & \left(\left(D_N \sum_{i=1}^{\infty} \mathbf{f}_i(\mathbf{u}_v(x)) \mathbf{g}_i(\mathbf{u}_v(y)) \right) \Big|_{y=x}, \Pi_N \mathbf{v}(x) \right)_{\widehat{D}} = \sum_{i=1}^{\infty} (D_N \mathbf{f}_i(\mathbf{u}_v), \mathbf{g}_i(\mathbf{u}_v) \Pi_N \mathbf{v})_{\widehat{D}} \\ &= \sum_{i=1}^{\infty} \langle \mathbf{f}_i(\mathbf{u}_v) \mathbf{g}_i(\mathbf{u}_v), \Pi_N \mathbf{v} n_x \rangle_{\partial \widehat{D}} - (\mathbf{f}_i(\mathbf{u}_v), D_N (\mathbf{g}_i \Pi_N \mathbf{v}))_{\widehat{D}} \\ &= \langle \mathbf{f}_S(\mathbf{u}_v, \mathbf{u}_v), \Pi_N \mathbf{v}(x) n_x \rangle_{\partial \widehat{D}} - \left((D_N (\mathbf{f}_S(\mathbf{u}_v(y), \mathbf{u}_v(x)) \Pi_N \mathbf{v}(x)))|_{y=x}, 1 \right)_{\widehat{D}} \\ &= \langle \mathbf{f}(\mathbf{u}_v), \Pi_N \mathbf{v}(x) n_x \rangle_{\partial \widehat{D}} - \left((D_N (\mathbf{f}_S(\mathbf{u}_v(x), \mathbf{u}_v(y)) \Pi_N \mathbf{v}(x)))|_{y=x}, 1 \right)_{\widehat{D}}, \end{aligned}$$

where we have used the linearity of D_N in the first step and the symmetry and consistency of \mathbf{f}_S in the third and last steps. Noting that

$$(D_N \mathbf{f}_S(\mathbf{u}_v(x), \mathbf{u}_v(y)))|_{y=x} \mathbf{v}(x) = (D_N (\mathbf{f}_S(\mathbf{u}_v(x), \mathbf{u}_v(y)) \mathbf{v}(y)))|_{y=x},$$

the volume terms can be manipulated to yield

$$\begin{aligned} & \left((D_N (\mathbf{f}_S(\mathbf{u}_v(x), \mathbf{u}_v(y)) (\Pi_N \mathbf{v}(y) - \Pi_N \mathbf{v}(x))))|_{y=x}, 1 \right)_{\widehat{D}} \\ &= \left((D_N (\psi(y) - \psi(x)))|_{y=x}, 1 \right)_{\widehat{D}} = (-D_N \psi(x), 1)_{\widehat{D}} = \langle -\psi, 1 \rangle_{\partial \widehat{D}}, \end{aligned}$$

where we have used Definition 1 in going from the first to second integral and Lemma 1 in the final step. \square

We see that, in the proof of Theorem 2, recovering the time derivative of entropy on the left hand side requires testing with the projection of the entropy variables. For diagonal norm SBP-DG methods, L^2 projection and nodal interpolation are equivalent, which implies that evaluating the flux \mathbf{f}_S using point-wise nodal values of the solution satisfies the entropy conservation condition given in Definition 1.

In general, the projection of the entropy variables does not agree with their point-wise evaluation. However, the entropy conservation condition given in Definition 1 holds so long as the flux is evaluated in a manner consistent with the projected entropy variables used to recover the time derivative of entropy. This motivates (12) as a non-standard way of evaluating the flux, which is the primary difference between the

construction of entropy conservative schemes for diagonal norm SBP-DG methods and more general DG methods.

Finally, we note that, while projected entropy variables can be used to build entropy conservative DG methods, the evaluation $\mathbf{u}(\Pi_N \mathbf{v})$ can be sensitive to differences between \mathbf{v} and $\Pi_N \mathbf{v}$, which are illustrated numerically and described in more detail in Section 5.1.5.

Remark. *In the proof of Theorem 2, Assumption 1 is used only to formally apply Lemma 1. The use of this assumption can be avoided by proving instead a discrete conservation of entropy in terms of a fixed quadrature rule and the quadrature-based matrices introduced in Section 4.3.*

4. Entropy conservative DG methods on multiple elements

In this section, we discuss the extension of entropy conservative DG schemes to multiple elements. The primary difficulty is the construction of appropriate interface (SBP-SAT) terms. Entropy stable formulations based on dense norm or GSBP operators require SBP-SAT terms which are either non-trivial to derive or have a wide stencil [39, 9]. We present an approach which instead utilizes a global approach to flux differencing to derive entropy conservative interface terms, and show that we can recover existing energy stable SBP-SAT terms for Burgers' equation.

4.1. Global DG differentiation operators

The construction of energy and entropy conservative DG-SBP methods requires the specification of appropriate SBP-SAT terms, which are often derived by reverse-engineering boundary terms such that interface contributions cancel [39, 33, 9]. We present an alternative systematic approach to the derivation of interface terms for DG formulations by combining flux differencing with global DG differential operators, which have been used to simplify the notation and analysis of DG formulations [40, 41, 42]. This approach does not require the specification of appropriate SBP-SAT terms in order to prove entropy conservation; rather, explicit expressions for SBP-SAT terms may be derived in a systematic manner using flux differencing and the definitions of global DG operators.

We introduce the global L^2 inner product and the broken Sobolev space $H^1(\Omega_h)$

$$(u, v)_\Omega = \sum_k (u, v)_{D^k}, \quad H^1(\Omega_h) = \bigoplus_k H^1(D^k),$$

where the local Sobolev spaces $H^1(D^k)$ are defined analogously to (8). We adapt the construction of local differentiation operators introduced in Section 3.4 and [6] to construct a global DG differentiation operator:

Definition 3. Let $w(x)$ be a bounded function over each element D^k . Define $D_h^x : H^1(\Omega_h) \rightarrow V_h$ as the operator which satisfies

$$\begin{aligned} (D_h^x u, vw)_\Omega &= \sum_k \left(\frac{\partial \Pi_N u}{\partial x}, vw \right)_{D^k} \\ &+ \frac{1}{2} \langle u^+ - \Pi_N u, v w n_x \rangle_{\partial D^k} + \frac{1}{2} \langle u - \Pi_N u, \Pi_N(vw) n_x \rangle_{\partial D^k}, \quad \forall v \in V_h. \end{aligned} \quad (13)$$

Entropy conservative DG schemes can be constructed using the fact that D_h^x satisfies a *global* analogue of integration by parts:

Lemma 2. *Let D_h^x be defined as in (13), and define $v^+ = v$ on $\partial\Omega$. Then,*

$$(D_h^x u, vw)_\Omega = \langle u, v w n_x \rangle_{\partial\Omega} - (u, D_h^x(vw))_\Omega.$$

for $u \in H^1(\Omega_h)$ and $\forall v \in V_h$.

Proof. We may rewrite $(D_h^x u, vw)_\Omega$ using Definition 2 of D_N

$$(D_h^x u, vw)_\Omega = \sum_k (D_N u, vw)_{D^k} + \frac{1}{2} \langle \llbracket u \rrbracket, vwn_x \rangle_{\partial D^k \setminus \partial \Omega}.$$

Assuming D^k are affine mappings of \widehat{D} and applying Lemma 1 over each element yields

$$\begin{aligned} (D_h^x u, vw)_\Omega &= \sum_k (-u, D_N(vw))_{D^k} + \left\langle u + \frac{1}{2} \llbracket u \rrbracket, vwn_x \right\rangle_{\partial D^k} \\ &= \sum_k (-u, D_N(vw))_{D^k} + \langle \{u\}, vwn_x \rangle_{\partial D^k} \\ &= \sum_k (-u, D_N(vw))_{D^k} - \langle \llbracket vw \rrbracket, un_x \rangle_{\partial D^k \setminus \partial \Omega} + \langle vw, un_x \rangle_{\partial \Omega} \\ &= \langle vw, un_x \rangle_{\partial \Omega} - (u, D_h^x(vw))_\Omega. \end{aligned}$$

□

Remark. It is possible to define a simpler global DG differentiation operator based on the composition of a discrete differentiation operator and the L^2 projection. Let $D_h^x : V_h \rightarrow V_h$ be defined as

$$(D_h^x u, v)_\Omega = \sum_k \left(\frac{\partial u}{\partial x}, v \right)_{D^k} + \frac{1}{2} \langle \llbracket u \rrbracket, vn_x \rangle_{\partial D^k}, \quad \forall v \in V_h.$$

Then, using that the range of D_h^x is V_h , it is straightforward to show that $D_h^x \Pi_N$ satisfies a different discrete analogue of integration by parts

$$(D_h^x \Pi_N u, v)_\Omega = \langle \Pi_N u, vn_x \rangle_{\partial \Omega} - (D_h^x \Pi_N v, u)_\Omega, \quad \forall u, v \in L^2(\Omega).$$

Unfortunately, while (13) requires only boundary values of solutions on neighboring elements, this construction results in a wide stencil involving all degrees of freedom on neighboring elements due to the combination of $\llbracket u \rrbracket$ and the L^2 projection. This choice of global DG differentiation operator is analogous to the choice of SBP-SAT terms specified in [39] for SBP operators on simplicial elements without boundary nodes.

4.2. Conservation of entropy and entropy stability on multiple elements

Extending the entropy stable DG formulation (11) to multiple elements is done by combining flux differencing with the global DG differential operator D_h^x

$$\left(\frac{\partial \mathbf{u}}{\partial t}, \mathbf{w} \right)_\Omega + \left((2D_h^x \mathbf{f}_S(\mathbf{u}_v(x), \mathbf{u}_v(y)))|_{y=x}, \mathbf{w} \right)_\Omega + \langle \mathbf{f}^* - \mathbf{f}(\mathbf{u}_v), \mathbf{w}n_x \rangle_{\partial \Omega}, \quad (14)$$

where \mathbf{f}^* is some numerical flux used in the imposition of boundary conditions, and the global operator D_h^x is understood to be applied to each component of $\mathbf{f}_S(\mathbf{u}(x), \mathbf{u}(y))$. This formulation yields a theorem analogous to Theorem 2:

Theorem 3. Let Assumption 1 hold over each element D^k . The formulation (14) is globally entropy conservative in the sense that

$$\left(\frac{\partial U(\mathbf{u})}{\partial t}, 1 \right)_\Omega = \langle \psi, 1n_x \rangle_{\partial \Omega} - \langle \mathbf{f}^*, \Pi_N \mathbf{v}n_x \rangle_{\partial \Omega}.$$

Proof. The proof is identical to the proof of Theorem 2, except that Lemma 2 is used in place of Lemma 1. □

Remark. The presented analysis focuses on discrete entropy conservation in one dimension. However, the extension to two and three dimensions is straightforward (as shown in [6]), based on analogous definitions of global DG operators in higher dimensions. The construction of higher dimensional DG operators differ from the one-dimensional case in that the evaluation of surface integrals also requires appropriately accurate quadrature rules over the reference element, such that integration by parts holds for elements of the local approximation space.

Finally, we note that the analysis in previous sections has focused on the construction of entropy conservative schemes. However, entropy is only conserved for smooth solutions and should be dissipated away in the presence of discontinuities and shocks. To this end, one can construct discretely entropy stable schemes by adding additional dissipative terms (for example, by adding HLL or Lax-Friedrichs penalization [6] or matrix dissipation terms [43, 26]). For the numerical experiments presented here, we apply a Lax-Friedrichs penalization, augmenting the flux function at element interfaces with the additional term

$$\mathbf{f}_S(\mathbf{u}_L, \mathbf{u}_R) \Rightarrow \mathbf{f}_S(\mathbf{u}_L, \mathbf{u}_R) - \frac{\lambda}{2} \llbracket \mathbf{u}_v \rrbracket,$$

where λ is an estimate of the maximum eigenvalue of the flux Jacobian. For an appropriate approximation of λ , Lax-Friedrichs penalization dissipates entropy [6]. We note that, since conservation of entropy requires testing with the projection of the entropy variables, the jump term should involve the evaluations of the conservative variables in terms of the projected entropy variables $\mathbf{u}_v = \mathbf{u}(\Pi_N \mathbf{v})$ in order to guarantee entropy dissipation.

4.2.1. Local conservation

It is also straightforward to show that the formulation (14) is locally conservative. We follow [6] and take $\mathbf{w} = \mathbb{1}_{D^k}$ in the formulation (14) to yield

$$\left(\frac{\partial \mathbf{u}}{\partial t}, 1 \right)_{D^k} + \left((2D_h^x \mathbf{f}_S(\mathbf{u}_v(x), \mathbf{u}_v(y)))|_{y=x}, 1 \right)_{D^k} + \langle \mathbf{f}^* - \mathbf{f}(\mathbf{u}_v), 1n_x \rangle_{\partial\Omega \cap D^k} = 0.$$

Using Assumption 1, Lemmas 1 and 2, and the symmetry of \mathbf{f}_S , one can show that

$$\begin{aligned} \left((2D_h^x \mathbf{f}_S(\mathbf{u}_v(x), \mathbf{u}_v(y)))|_{y=x}, 1 \right)_{D^k} &= \sum_{i=1}^{\infty} (2D_N \mathbf{f}_i(\mathbf{u}_v), \mathbf{g}_i(\mathbf{u}_v))_{D^k} + \langle \llbracket \mathbf{f}_i(\mathbf{u}_v) \rrbracket, \mathbf{g}_i(\mathbf{u}_v)n_x \rangle_{\partial D^k} \\ &= \sum_{i=1}^{\infty} (D_N \mathbf{f}_i(\mathbf{u}_v), \mathbf{g}_i(\mathbf{u}_v))_{D^k} - (D_N \mathbf{f}_i(\mathbf{u}_v), \mathbf{g}_i(\mathbf{u}_v))_{D^k} \\ &\quad + \langle \mathbf{f}_i(\mathbf{u}_v), \mathbf{g}_i(\mathbf{u}_v)n_x \rangle_{\partial D^k} + \langle \mathbf{f}_i(\mathbf{u}_v^+) - \mathbf{f}_i(\mathbf{u}_v), \mathbf{g}_i(\mathbf{u}_v)n_x \rangle_{\partial D^k} \\ &= (D_N \mathbf{f}_S(\mathbf{u}_v(x), \mathbf{u}_v(y)), 1)_{D^k} - (D_N \mathbf{f}_S(\mathbf{u}_v(x), \mathbf{u}_v(y)), 1)_{D^k} \\ &\quad + \langle \mathbf{f}_S(\mathbf{u}_v^+, \mathbf{u}_v), 1n_x \rangle_{\partial D^k}. \end{aligned}$$

The volume terms cancel, leaving

$$\sum_k \left(\frac{\partial \mathbf{u}}{\partial t}, 1 \right)_{D^k} + \langle \mathbf{f}_S(\mathbf{u}_v^+, \mathbf{u}_v), 1n_x \rangle_{\partial D^k} + \langle \mathbf{f}^* - \mathbf{f}(\mathbf{u}_v), 1n_x \rangle_{\partial\Omega \cap D^k} = 0.$$

On the domain boundary $\partial\Omega$, $\mathbf{u}_v^+ = \mathbf{u}_v$ and $\mathbf{f}_S(\mathbf{u}_v^+, \mathbf{u}_v) = \mathbf{f}(\mathbf{u}_v)$ using the consistency of \mathbf{f}_S . This implies that the statement of local conservation replaces \mathbf{f}_S with the boundary condition numerical flux \mathbf{f}^* on element boundaries which intersect with the domain boundary.

We note that this definition of local conservation is given in terms of the conservative variables evaluated as a function of the projected entropy variables $\mathbf{u}(\Pi_N \mathbf{v})$. It is not immediately clear that this statement of local conservation satisfies the conditions of the classic Lax-Wendroff theorem. However, this form of local conservation does satisfy a generalized definition of local conservation, which is sufficient to guarantee convergence to a weak solution under mesh refinement [44].

4.2.2. Recovery of known SBP-SATs for Burgers' equation

To illustrate the application of the global flux differencing approach, we derive explicit expressions for an energy stable formulation for Burgers' equation. We will assume periodic boundary conditions for simplicity and use the energy conserving flux

$$f_S(u_L, u_R) = \frac{1}{6} (u_L^2 + u_L u_R + u_R^2).$$

We focus on the discretization of the spatial terms. Since the solution is itself the entropy variable for Burgers equation, we can apply flux differencing directly to the flux evaluated with u to yield the following contributions

$$\left(2D_h^x f_S(u(x), u(y))|_{y=x}, v \right) = \frac{1}{3} \left(D_h^x (u(x)^2 + u(x)u(y) + u(y)^2)|_{y=x}, v \right) = \frac{1}{3} (D_h^x (u^2) + u D_h^x u, v).$$

Expanding out the definition of $D_h^x (u^2)$ gives

$$\begin{aligned} (D_h^x (u^2), v)_\Omega &= \sum_k \left(\frac{\partial \Pi_N u^2}{\partial x}, v \right)_{D^k} + \frac{1}{2} \langle (u^+)^2 - \Pi_N u^2, v n_x \rangle_{\partial D^k} + \frac{1}{2} \langle u^2 - \Pi_N u^2, \Pi_N v n_x \rangle_{\partial D^k} \\ &= \sum_k \left(\frac{\partial \Pi_N u^2}{\partial x}, v \right)_{D^k} + \left\langle \frac{1}{2} ((u^+)^2 + u^2) - \Pi_N u^2, v n_x \right\rangle_{\partial D^k}. \end{aligned}$$

Similarly, expanding out $(u D_h^x u, v)_\Omega$ gives

$$\begin{aligned} (u D_h^x u, v)_\Omega &= (D_h^x u, uv)_\Omega = \sum_k \left(\frac{\partial u}{\partial x}, uv \right)_{D^k} + \frac{1}{2} \langle u^+ - \Pi_N u, uv n_x \rangle_{\partial D^k} + \frac{1}{2} \langle u - \Pi_N u, \Pi_N (uv) \rangle_{\partial D^k} \\ &= \sum_k \left(\frac{\partial u}{\partial x}, uv \right)_{D^k} + \frac{1}{2} \langle u^+ u - u^2, v n_x \rangle_{\partial D^k}. \end{aligned}$$

Summing these together yields explicit expressions for the DG weak formulation

$$\sum_k \left(\frac{\partial u}{\partial t} + \frac{1}{3} \left(\frac{\partial \Pi_N u^2}{\partial x} + u \frac{\partial u}{\partial x} \right), v \right)_{D^k} + \left\langle f_S(u^+, u) - \frac{1}{3} \Pi_N u^2 - \frac{1}{6} u^2, v n_x \right\rangle_{\partial D^k} = 0.$$

This formulation, which was derived through the use of flux differencing with the global DG operator (13), recovers the weak form of the split formulation and the correction terms introduced in [9] for Burgers' equation.¹

4.3. On implementation in terms of the flux function

While it is possible to use (14) to derive explicit DG weak formulations, this becomes impractical for complicated flux functions. Instead, we follow [3, 6] and seek an implementation of flux differencing and (14) which requires only the specification of $f_S(\mathbf{u}_L, \mathbf{u}_R)$ and local matrices.

We begin by rewriting the global DG differentiation operator D_h^x using the lifting operator $L : V_h \rightarrow V_h$ [10, 41]

$$(Lu, v)_{D^k} = \langle u, v \rangle_{\partial D^k}, \quad u \in V_h, \quad \forall v \in V_h.$$

The lift operator can be used to rewrite surface terms which involve the volume L^2 projection

$$\langle u, \Pi_N(vw) \rangle_{\partial D^k} = (Lu, \Pi_N(vw))_{D^k} = (Lu, vw)_{D^k}, \quad u, v \in V_h,$$

¹Entropy conservative SBP-SAT were introduced for generalized SBP operators and the compressible Euler equations in [33]; however, we have not yet determined if the combination of flux differencing with (13) recovers these fluxes as well.

where $w(x)$ is smooth and bounded over D^k . Then, the global DG operator D_h^x can be rewritten as

$$(D_h^x u, v)_\Omega = \sum_k \left(\frac{\partial \Pi_N u}{\partial x} + \frac{1}{2} L(u - \Pi_N u) n_x, v \right)_{D^k} + \frac{1}{2} \langle u^+ - \Pi_N u, v n_x \rangle_{\partial D^k}. \quad (15)$$

Applying Assumption 1, we can rewrite

$$\begin{aligned} \left((D_h^x \mathbf{f}_S(\mathbf{u}_v(x), \mathbf{u}_v(y)))|_{y=x}, \mathbf{w} \right)_\Omega &= \sum_k \sum_{i=1}^{\infty} \left(\frac{\partial \Pi_N \mathbf{f}_i(\mathbf{u}_v)}{\partial x}, \mathbf{g}_i(\mathbf{u}_v) \mathbf{w} \right)_{D^k} \\ &+ \sum_k \sum_{i=1}^{\infty} \frac{1}{2} \langle \mathbf{f}_i(\mathbf{u}_v^+) \mathbf{g}_i(\mathbf{u}_v) - \mathbf{g}_i(\mathbf{u}_v) \Pi_N \mathbf{f}_i(\mathbf{u}_v), \mathbf{w} n_x \rangle_{\partial D^k} \\ &+ \sum_k \sum_{i=1}^{\infty} \frac{1}{2} (\mathbf{g}_i(\mathbf{u}_v) L(\mathbf{f}_i(\mathbf{u}_v^+) - \Pi_N \mathbf{f}_i(\mathbf{u}_v)) n_x, \mathbf{w})_{D^k}, \end{aligned}$$

where $\mathbf{w} n_x$ denotes component-wise multiplication of \mathbf{w} by n_x , the x -component of the unit outward normal. Moving the sum inside and simplifying then reduces this expression to

$$\begin{aligned} \left((D_h^x \mathbf{f}_S(\mathbf{u}_v(x), \mathbf{u}_v(y)))|_{y=x}, \mathbf{w} \right)_\Omega &= \sum_k \left(\frac{\partial \Pi_N \mathbf{f}_S(\mathbf{u}_v(x), \mathbf{u}_v(y))}{\partial x} \Big|_{y=x}, \mathbf{w} \right)_{D^k} \\ &+ \frac{1}{2} \langle \mathbf{f}_S(\mathbf{u}_v^+, \mathbf{u}_v) - \Pi_N^x \mathbf{f}_S(\mathbf{u}_v(x), \mathbf{u}_v(y))|_{y=x}, \mathbf{w} n_x \rangle_{\partial D^k} \\ &+ \frac{1}{2} \left((L^x(\mathbf{f}_S(\mathbf{u}_v(x), \mathbf{u}_v(y)) - \Pi_N^x \mathbf{f}_S(\mathbf{u}_v(x), \mathbf{u}_v(y))) n_x)|_{y=x}, \mathbf{w} \right)_{D^k}, \end{aligned} \quad (16)$$

where the superscripts L^x, Π_N^x are used to emphasize that the lift and volume projection operators are applied in x but not in y . This can be used to practically evaluate $(D_h^x \mathbf{f}_S(\mathbf{u}_v(x), \mathbf{u}_v(y)))|_{y=x}$ using only reference operators and evaluations of \mathbf{f}_S . We note that in general, the boundary terms in (16) interact with volume terms through the L^2 projection and the lift operator. This can be simplified for diagonal norm SBP operators with boundary nodes, using the equivalence between L^2 projection and interpolation at nodal points for diagonal norm nodal SBP operators.

We now discuss how to recover a computational form of flux differencing analogous to the approaches presented in [3, 6]. We will describe how to compute the three non-standard terms which appear within (16):

$$\left(\frac{\partial \Pi_N f_S(u(x), u(y))}{\partial x} \Big|_{y=x}, w \right)_{\hat{D}} \quad (17)$$

$$\langle f_S(u^+, u) - \Pi_N^x f_S(u(x), u(y))|_{y=x}, w n_x \rangle_{\partial \hat{D}} \quad (18)$$

$$\left((L^x(f_S(u(x), u(y)) - \Pi_N^x f_S(u(x), u(y))) n_x)|_{y=x}, w \right)_{\hat{D}}. \quad (19)$$

where we have restricted ourselves to the reference element \hat{D} and a scalar variable $u(x)$ and flux function f_S for simplicity. Extending to vector variables and flux functions requires repeating the same procedures component-by-component, and extending to affinely mapped elements D^k involves scaling by constant geometric factors.

We first introduce matrices defined over the reference element $\hat{D} = [-1, 1]$. Let $\mathbf{W} \in \mathbb{R}^{N_q \times N_q}$ denote the diagonal matrix whose entries are quadrature weights

$$\mathbf{W}_{ij} = \begin{cases} w_i & i = j \\ 0 & \text{otherwise.} \end{cases}$$

Then, assuming $u(x) \in P^N(\hat{D})$, it can be represented in some polynomial basis $\phi_j(\hat{x})$ of degree N and dimension N_p

$$P^N(\hat{D}) = \text{span} \{ \phi_i(\hat{x}) \}_{i=1}^{N_p}.$$

We can define the quadrature interpolation matrix \mathbf{V}_q

$$(\mathbf{V}_q)_{ij} = \phi_j(\hat{x}_i), \quad 1 \leq j \leq N_p, \quad 1 \leq i \leq N_q.$$

Given \mathbf{V}_q , we introduce the mass matrix $\mathbf{M} = \mathbf{V}_q^T \mathbf{W} \mathbf{V}_q$, as well as the quadrature-based L^2 projection operator \mathbf{P}_q

$$\mathbf{P}_q = \mathbf{M}^{-1} \mathbf{V}_q^T \mathbf{W},$$

which maps a function (in terms of its evaluation at quadrature points) to coefficients of the basis $\phi_j(x)$. Next, let $\hat{\mathbf{D}}$ denote the differentiation matrix defined implicitly through

$$u(x) = \sum_{j=1}^{N_p} \mathbf{u}_j \phi_j(\hat{x}), \quad \frac{\partial u}{\partial \hat{x}} = \sum_{j=1}^{N_p} (\hat{\mathbf{D}} \mathbf{u})_j \phi_j(\hat{x})$$

In other words, $\hat{\mathbf{D}}$ maps basis coefficients of some function $u \in V_h$ to coefficients of its derivative on $[-1, 1]$.

We now describe how to evaluate the volume term (17) using these matrices.² We note that this procedure only evaluates the variational formulation. To use this evaluation within an explicit time-stepping scheme, the additional inversion of a local mass matrix is required to compute the right hand side contribution over each element.

Let \mathbf{F}_S denote the $N_q \times N_q$ matrix constructed by evaluating f_S at volume quadrature points

$$(\mathbf{F}_S)_{ij} = f_S(u(x_i), u(x_j)).$$

Then, integrating (17) using quadrature is equivalent to computing

$$\left(\frac{\partial \Pi_N f_S(u(x), u(y))}{\partial x} \Big|_{y=x}, w \right)_{D^k} = \mathbf{w}^T \mathbf{V}_q^T \text{diag} \left(\mathbf{W} \mathbf{V}_q \hat{\mathbf{D}} \mathbf{P}_q \mathbf{F}_S \right),$$

where \mathbf{w} denote the vector of coefficients for the weighting function $w(x)$.

The diagonal operation can be replaced by a more convenient evaluation. Let $\mathbf{A} \circ \mathbf{B}$ denote the Hadamard product of two matrices $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{N_q \times N_q}$, such that

$$(\mathbf{A} \circ \mathbf{B})_{ij} = \mathbf{A}_{ij} \mathbf{B}_{ij}, \quad 1 \leq i, j \leq N_q.$$

The Hadamard product of two matrices can also be used to conveniently represent the diagonal of a product of matrices [45]:

$$\text{diag}(\mathbf{A} \mathbf{B}^T) = (\mathbf{A} \circ \mathbf{B}) \mathbf{1}.$$

Using this property and the symmetry of \mathbf{F}_S , we can rewrite the diagonal operation as follows

$$\text{diag}(\mathbf{W} \mathbf{V}_q \mathbf{D}_q \mathbf{P}_q \mathbf{F}_S) = ((\mathbf{W} \mathbf{V}_q \mathbf{D}_q \mathbf{P}_q) \circ \mathbf{F}_S) \mathbf{1}.$$

²We can also use this matrix notation to directly show conservation of entropy. For example, the time term in the statement of discrete entropy conservation or dissipation can be derived as follows: let $\mathbf{u}_q, \mathbf{v}_q$ denote the conservation and entropy variables evaluated at quadrature points. Then, multiplying with the transpose of $\mathbf{v} = \mathbf{P}_q \mathbf{v}_q$ yields

$$(\mathbf{P}_q \mathbf{v}_q)^T \mathbf{M} \frac{\partial \mathbf{u}}{\partial t} = \mathbf{v}_q^T \mathbf{W} \mathbf{V}_q \mathbf{M}^{-1} \mathbf{M} \frac{\partial \mathbf{u}}{\partial t} = \mathbf{1}^T \mathbf{W} \text{diag}(\mathbf{v}_q) \frac{\partial \mathbf{u}_q}{\partial t} = \mathbf{1}^T \mathbf{W} \frac{\partial U(\mathbf{u}_q)}{\partial t},$$

where we have assumed continuity in time, and have used that \mathbf{W} is diagonal.

A computational advantage of evaluation in terms of the Hadamard product is that \mathbf{F}_S does not need to be formed explicitly.

The same approach may also be applied to (18) and (19). We focus first on (18) and introduce matrices used to compute boundary terms. Let \mathbf{V}_f denote the matrix which interpolates to boundary values³

$$(\mathbf{V}_f)_{1j} = \phi_j(-1), \quad (\mathbf{V}_f)_{2j} = \phi_j(1), \quad 1 \leq j \leq N_p.$$

Furthermore, let \mathbf{W}_f denote the diagonal matrix of surface quadrature weights. In one dimension, this is simply the 2×2 identity matrix. We now define \mathbf{F}_S^f to be the matrix constructed by evaluating \mathbf{F}_S at both volume quadrature points x_i and boundary points x_j^f

$$\left(\mathbf{F}_S^f\right)_{ij} = f_S\left(u(x_i), u\left(x_j^f\right)\right), \quad 1 \leq i \leq N_q, \quad j = 1, 2.$$

We also define \mathbf{F}_S^+ to be the vector of evaluations of the flux function $f_S(u^+, u)$ at the boundary points of D^k . Using these matrices and properties of Hadamard products, we can then evaluate

$$\begin{aligned} \left\langle f_S(u^+, u) - \Pi_N^x f_S(u(x), u(y))|_{y=x}, w n_x \right\rangle_{\partial D^k} &= \mathbf{w}^T \mathbf{V}_f^T \mathbf{W}_f \text{diag}(n_x) \left(\mathbf{F}_S^+ - \text{diag}\left(\mathbf{V}_f \mathbf{P}_q \mathbf{F}_S^f\right) \right) \\ &= \mathbf{w}^T \mathbf{V}_f^T \mathbf{W}_f \text{diag}(n_x) \left(\mathbf{F}_S^+ - \left((\mathbf{V}_f \mathbf{P}_q) \circ \left(\mathbf{F}_S^f \right)^T \right) \mathbf{1} \right). \end{aligned}$$

This step can be interpreted as coupling together evaluations of the flux function at volume and boundary points. We note that this is the only step in evaluating (16) which requires non-local data (to compute \mathbf{F}_S^+).

In order to evaluate the final term (19), we need to introduce the quadrature-based lift matrix

$$\mathbf{L} = \mathbf{M}^{-1} \mathbf{V}_f^T \mathbf{W}_f,$$

which (as the name implies) “lifts” a function from the boundary (in terms of its evaluation at surface quadrature points) to coefficients of a basis defined on the interior of the element. Then, we have that

$$\begin{aligned} &\left((L^x(f_S(u(x), u(y)) - \Pi_N^x f_S(u(x), u(y))) n_x)|_{y=x}, w \right)_{D^k} \\ &= \mathbf{w}^T \mathbf{V}_q^T \mathbf{W} \text{diag}\left(\mathbf{V}_q \mathbf{L} \text{diag}(n_x) \left(\left(\mathbf{F}_S^f \right)^T - \mathbf{V}_f \mathbf{P}_q \mathbf{F}_S^f \right) \right) \\ &= \mathbf{w}^T \mathbf{V}_q^T \mathbf{W} \left((\mathbf{V}_q \mathbf{L} \text{diag}(n_x)) \circ \mathbf{F}_S^f - (\mathbf{V}_q \mathbf{L} \text{diag}(n_x) \mathbf{V}_f \mathbf{P}_q) \circ \mathbf{F}_S \right) \mathbf{1}. \end{aligned}$$

This expression is similar to the evaluation of (18) in that it couples together volume and boundary evaluations of the flux function, but in the reverse order.

It is worth pointing out that these sets of expressions can be simplified; for example, straightforward computations show that the $\mathbf{W} \mathbf{V}_q \mathbf{L} = (\mathbf{W}_f \text{diag}(n_x) \mathbf{V}_f \mathbf{P}_q)^T$, and the computation of the volume term $((\mathbf{V}_q \mathbf{D}_q \mathbf{P}_q) \circ \mathbf{F}_S) \mathbf{1}$ and the surface term $((\mathbf{V}_q \mathbf{L} \text{diag}(n_x) \mathbf{V}_f \mathbf{P}_q) \circ \mathbf{F}_S) \mathbf{1}$ can be combined into one single operation. Furthermore, for specific choices of basis and quadrature, one can eliminate certain terms altogether. For example, the co-location of nodal and quadrature points (assuming that the dimension of the approximation space is identical to the number of quadrature points) reduces $\mathbf{V}_q, \mathbf{P}_q$ to identity matrices. Furthermore, if quadrature points coincide with boundary points (as with DG-SEM methods), the lift matrix \mathbf{L} is zero except for entries which correspond to those boundary points. These assumptions greatly simplify both the formulation and implementation of entropy conservative/stable DG methods, and will be discussed more thoroughly in a future manuscript.

³In generalizing to higher dimensions, \mathbf{V}_f would be the matrix which interpolates to quadrature points on the boundary.

5. Numerical experiments

In this section, we illustrate the entropy conservation and accuracy of the proposed schemes for the one dimensional compressible Euler equations. All numerical experiments utilize the fourth order five-stage low-storage Runge-Kutta method Carpenter and Kennedy [46]. Following the derivation of stable timestep restrictions in [47], we define the timestep Δt to be

$$\Delta t = \text{CFL} \times \frac{h}{C_N}, \quad C_N = \frac{(N+1)^2}{2},$$

where C_N is the one-dimensional constant in the trace inequality [48], and CFL is a user-defined constant.

We note that the numerical implementations used here are oblivious to the choice of basis, and the discretization is specified completely by the choice of quadrature. For example, when GLL quadratures are used, an entropy conservative/stable DG-SEM discretization is recovered [2, 6, 34], and when a Gauss quadrature with $(N+1)$ points is used, generalized SBP-DG methods are recovered [9, 33].

5.1. Compressible Euler equations

The one-dimensional compressible Euler equations, which correspond to the inviscid limit of the compressible Navier-Stokes equations, are given as follows:

$$\begin{aligned} \frac{\partial \rho}{\partial t} + \frac{\partial (\rho u)}{\partial x} &= 0, \\ \frac{\partial \rho u}{\partial t} + \frac{\partial (\rho u^2 + p)}{\partial x} &= 0, \\ \frac{\partial E}{\partial t} + \frac{\partial (u(E + p))}{\partial x} &= 0. \end{aligned}$$

We assume an ideal gas, such that the pressure satisfies the constitutive relation $p = (\gamma - 1) (E - \frac{1}{2} \rho u^2)$, where $\gamma = 1.4$ is the ratio of specific heat for a diatomic gas.

The choice of convex entropy for the Euler equations is non-unique [49]. However, a unique entropy can be chosen by restricting to choices of entropy variables which symmetrize the viscous heat conduction term in the compressible Navier-Stokes equations [29]. This leads to $U(\mathbf{u})$ of the form

$$U(\mathbf{u}) = -\frac{\rho s}{\gamma - 1},$$

where $s = \log\left(\frac{p}{\rho^\gamma}\right)$ is the physical specific entropy. The entropy variables under this choice of entropy are then

$$v_1 = \frac{\gamma - s}{\gamma - 1} - \frac{\rho u^2}{2p}, \quad v_2 = \frac{\rho u}{p}, \quad v_3 = -\frac{\rho}{p}.$$

The inverse mapping is given by

$$\rho = -(\rho e)v_3, \quad m = (\rho e)v_2, \quad E = (\rho e) \left(1 - \frac{v_2^2}{2v_3}\right),$$

where ρe and s in terms of the entropy variables are

$$\rho e = \left(\frac{(\gamma - 1)}{(-v_3)^\gamma}\right)^{1/(\gamma-1)} e^{\frac{-s}{\gamma-1}}, \quad s = \gamma - v_1 + \frac{v_2^2}{2v_3}.$$

In order for the entropy $U(\mathbf{u})$ to be well defined, we require the assumption that the discrete density and pressure solutions are bounded away from zero

$$\rho \geq \rho_0 > 0, \quad p \geq p_0 > 0. \quad (20)$$

This can be achieved using positivity-preserving limiters [36, 37]. However, these have not been implemented in our numerical simulations.

Examples of entropy conservative flux functions can be found in [50, 43]. In this work, we utilize the flux function $\mathbf{f}_S(\mathbf{u}_L, \mathbf{u}_R)$ introduced by Chandreshekar [43], whose components are given as

$$\begin{aligned} f_S^1(\mathbf{u}_L, \mathbf{u}_R) &= \{\{\rho\}\}^{\log} \{\{u\}\} \\ f_S^2(\mathbf{u}_L, \mathbf{u}_R) &= \frac{\{\{\rho\}\}}{2 \{\{\beta\}\}} + \{\{u\}\} f_S^1 \\ f_S^3(\mathbf{u}_L, \mathbf{u}_R) &= f_S^1 \left(\frac{1}{2(\gamma-1) \{\{\beta\}\}^{\log}} - \frac{1}{2} \{\{u^2\}\} \right) + \{\{u\}\} f_S^2, \end{aligned}$$

where we have introduced the inverse temperature β

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

and the logarithmic mean

$$\{\{u\}\}^{\log} = \frac{u_L - u_R}{\log u_L - \log u_R}.$$

We note that, because the direct evaluation of the logarithmic mean is numerically sensitive for $u_L \approx u_R$, when $|u_L - u_R| < \epsilon$ we switch to evaluation using a high order accurate expansion introduced by Ismail and Roe [50].

These fluxes are both entropy conservative and kinetic energy preserving. Unlike the shallow water equations, these entropy conservative fluxes do not correspond to stable split formulations of the Euler equations. Thus, the compressible Euler equations serve as a test of the flux differencing formulation, as entropy stability cannot be achieved through skew-symmetry.

We also present results which utilize the dissipative local Lax-Friedrichs flux described in Section 4.2, where the value of λ is estimated by

$$\lambda = \max_{\mathbf{u}^+, \mathbf{u}} \{|u| + c\}, \quad c = \sqrt{\frac{\gamma p}{\rho}}.$$

We will refer to the combination of the entropy conservative flux with Lax-Friedrichs dissipation as the “Lax-Friedrichs” flux.

5.1.1. Smooth entropy wave solution

We begin by verifying the high order accuracy of the proposed methods using a periodic entropy wave solution

$$\rho(x, t) = 2 + \sin(\pi(x - t)), \quad u(x, t) = 1, \quad p(x, t) = 1.$$

We compute the L^2 error in the conservative variables

$$\|\mathbf{u} - \mathbf{u}_h\|_{L^2}^2 = \|\rho - \rho_h\|_{L^2}^2 + \|m - m_h\|_{L^2}^2 + \|E - E_h\|_{L^2}^2$$

at final time $T = .7$ using both the entropy conservative and local Lax-Friedrichs fluxes and a CFL of .125. For these experiments, we compare three quadrature rules:

1. the $(N + 1)$ point Gauss-Legendre-Lobatto rule (referred to as “GLL”),
2. a $(N + 1)$ point Gauss quadrature rule (referred to as “GQ- $(N + 1)$ ”),
3. an over-integrated $(N + 2)$ point Gauss quadrature rule (referred to as “GQ- $(N + 2)$ ”).

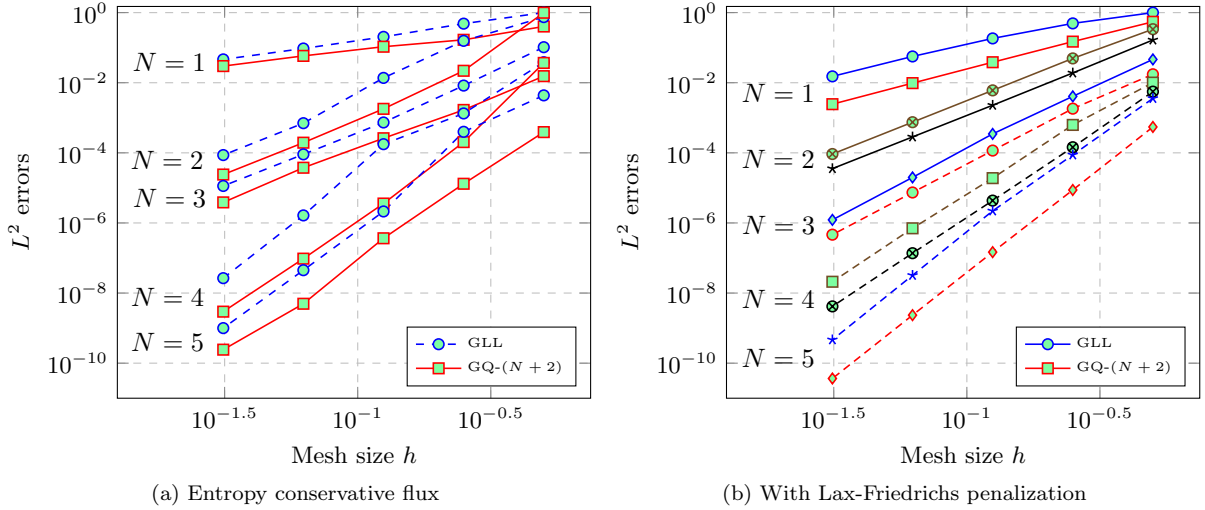


Figure 1: L^2 errors under mesh refinement for entropy conservative and Lax-Friedrichs fluxes under both Gauss-Legendre-Lobatto (GLL) and over-integrated $(N + 2)$ point Gauss quadrature (GQ- $(N + 2)$).

	$N = 1$	$N = 2$	$N = 3$	$N = 4$	$N = 5$
GLL (entropy conservative)	1.0672	3.6561	3.0086	6.3486	5.5278
GQ- $(N + 2)$ (entropy conservative)	0.9175	3.1132	3.0388	5.1221	5.2779
GLL (Lax-Friedrichs)	1.8887	3.0164	4.0241	5.0650	6.1095
GQ- $(N + 2)$ (Lax-Friedrichs)	1.9950	3.0018	3.9980	5.0419	6.0034

Table 1: Computed asymptotic convergence rates of L^2 errors under mesh refinement.

The L^2 error is evaluated using a more accurate $N + 5$ point Gauss quadrature rule.

Figure 1 shows computed L^2 errors under mesh refinement for both GLL and GQ- $(N + 2)$ quadrature rules with and without Lax-Friedrichs penalization. We do not compare against the $(N + 1)$ point Gauss quadrature rule or quadrature rules with $N_q > (N + 2)$, as the L^2 errors are very similar to the GQ- $(N + 2)$ case. Computed asymptotic convergence rates are reported in Table 1. It can be observed that, for both GLL and GQ- $(N + 2)$ quadratures, the entropy conservative flux exhibits suboptimal convergence rates for odd orders, while the inclusion of Lax-Friedrichs penalization restores the optimal $O(h^{N+1})$ convergence rate for both quadrature choices.

5.1.2. Discontinuous profile on a periodic domain

Next, we examine the discrete evolution of entropy by evolving a discontinuous initial profile to final time $T = 2$ on the domain $[-1, 1]$. We initialize the density and velocity to be

$$\rho(x, t) = \begin{cases} 3 & |x| < 1/2 \\ 2 & \text{otherwise,} \end{cases} \quad u(x, t) = 0, \quad p(x, t) = \rho^\gamma. \quad (21)$$

Periodic boundary conditions are enforced in order to examine the evolution of entropy over longer time periods. Figure 2a shows ρ, u at time $T = 1/10$ using both entropy conservative and Lax-Friedrichs fluxes (referred to in the figure as “EC” and “LF”, respectively) and GQ- $(N + 2)$ quadrature. As expected, the entropy conservative flux results in spurious high frequency oscillations, which are significantly damped under Lax-Friedrichs penalization.

We next examine the change in entropy over time. While the proof of conservation of entropy holds at the semi-discrete level, it does not take into account the time discretization and thus does not hold at the

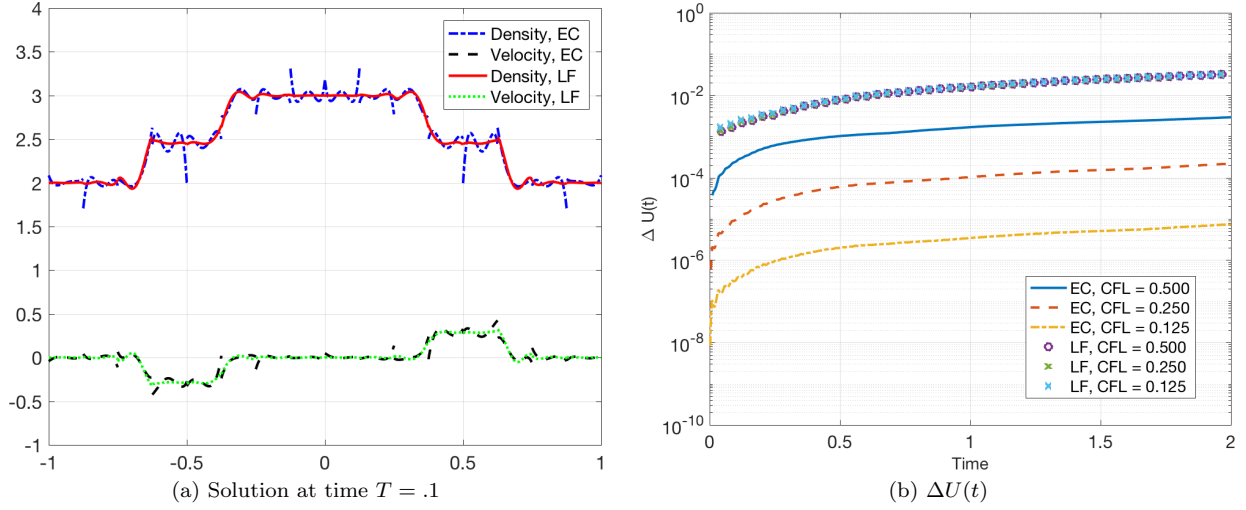


Figure 2: Solution snapshot and change in entropy $\Delta U(t)$ for both entropy conservative (EC) and Lax-Friedrichs (LF) fluxes.

fully discrete level. This is reflected in the observation that the numerical change in entropy

$$\Delta U(t) = U(\mathbf{u}(x, t)) - U(\mathbf{u}(x, 0))$$

increases as t increases. However, numerical experiments in [4] suggest that the discrete change in entropy over time should converge to zero as the timestep decreases.

Figure 2b shows the evolution of $\Delta U(t)$ to final time $T = 2$ for both entropy conservative and Lax-Friedrichs fluxes at various CFL numbers. For the entropy conservative flux, we observe that $\Delta U(t)$ decreases as the CFL and timestep Δt decrease. This is expected, since the discrete time problem should converge to the continuous semi-discrete problem (for which $\Delta U(t) = 0$) as $\Delta t \rightarrow 0$. We also observe that $\Delta U(t)$ does not change significantly as a function of the CFL for the Lax-Friedrichs flux, indicating that the non-zero change in entropy in this case is due to the effect of the dissipative flux rather than the time discretization.

We also numerically evaluate the spatial formulation tested against the projected entropy variables

$$\delta(t) = \left| \left((D_h^x \mathbf{f}_S(\mathbf{u}_v(x), \mathbf{u}_v(y)))|_{y=x}, \Pi_N \mathbf{v} \right)_\Omega \right|, \quad 0 \leq t \leq T.$$

We utilize the non-dissipative entropy conservative flux and evolve the initial profile (21) until time $T = 1$ using a GQ- $(N + 2)$ quadrature rule. From Theorem 3, we expect $\delta_{\max} = \max_{t \in (0, T)} \delta$ to be machine zero. In practice, we have found that δ_{\max} depends on the tolerance ϵ used in evaluation of the logarithmic mean. For a simulation to time $T = 4$ using $N = 4$ and $K = 16$ with a CFL of $1/2$, we observe that using $\epsilon = 10^{-2}$ (as recommended in [50]) results in $\delta_{\max} = O(10^{-10})$. Decreasing ϵ to 10^{-3} reduces δ_{\max} to $O(10^{-14})$, and decreasing ϵ further to 1×10^{-4} reduces δ_{\max} to 10^{-15} . Smaller values of ϵ do result in observable changes to δ_{\max} , and we do not observe any significant dependence of δ_{\max} on other discretization parameters.

Next, in order to illustrate the importance of evaluating the flux in terms of the projected entropy variables $\Pi_N \mathbf{v}$, we compute $\delta(t)$ while evaluating the flux function \mathbf{f}_S directly in terms of the conservative variables $\mathbf{u}(x)$ and in terms of the projected entropy variables $\mathbf{u}((\Pi_N \mathbf{v})(x))$. It can be observed from Figure 3 that $\delta(t)$ is near machine precision when evaluating the flux in terms of the projected entropy variables. When evaluating the flux function directly in terms of the conservative variables, $\delta(t)$ begins near 10^{-4} but grows steadily, blowing up exponentially near $t = 1$.

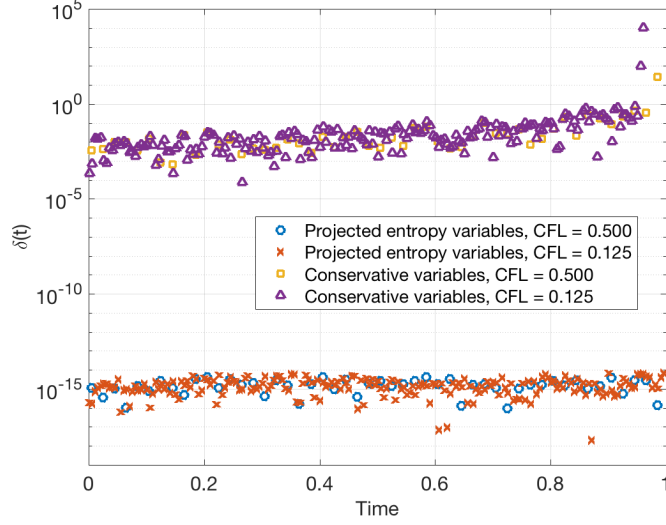


Figure 3: Comparison of $\delta(t)$ when evaluating the flux function f_S using conservative variables and projected entropy variables.

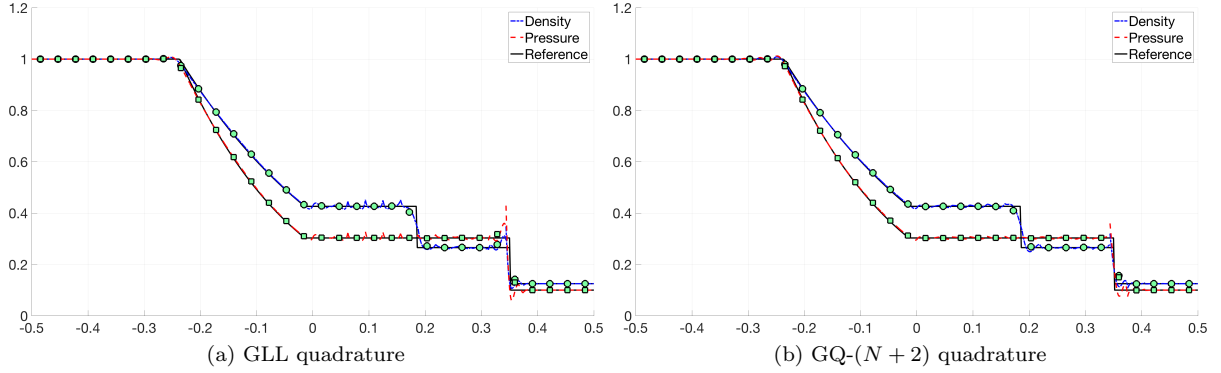


Figure 4: Density and pressure solutions for the Sod shock tube at $T = .2$ for $N = 4$ and $K = 32$ elements. Cell averages are overlaid as filled circles.

5.1.3. Sod shock tube

We now examine the behavior of the proposed DG methods for some common one-dimensional test problems. We begin with the Sod shock tube, which is posed on the domain $[-1/2, 1/2]$ with initial conditions

$$\rho = \begin{cases} 1 & x < 0 \\ .125 & x \geq 0, \end{cases} \quad u = 0, \quad p = \begin{cases} 1 & x < 0 \\ .1 & x \geq 0. \end{cases}$$

Boundary conditions are enforced by taking the external value \mathbf{u}^+ in the numerical flux to be that of the initial condition at $x = \pm 1$. The solution develops a left-moving rarefaction, as well as a right moving shock wave and a contact discontinuity. We simulate the solution until time $T = .2$, without the use of positivity preserving or TVD-type limiters. For all choices of quadrature tested, the solution diverges when using the entropy conservative flux, which is a result of oscillations in the solution and density and temperature becoming negative. This is remedied when using the dissipative Lax-Friedrichs flux, for which we do not observe blowup of the solution.

Figure 4 shows both the exact solution and the computed density and pressure along with their cell averages. These results were obtained using a CFL of .125 and GLL and GQ- $(N + 2)$ quadrature rules.

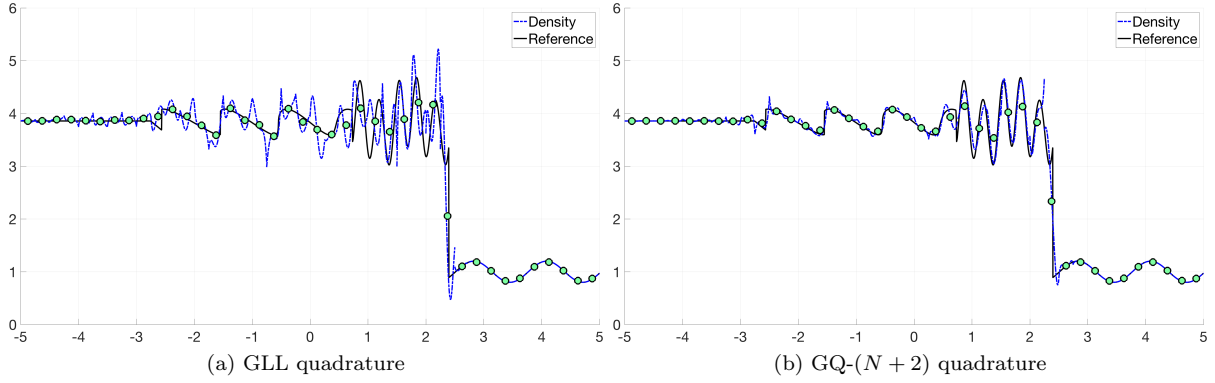


Figure 5: Density solutions for the sine-shock interaction problem at $T = 1.8$ for $N = 4$ and $K = 40$ elements. Cell averages are overlaid as filled circles.

For both quadratures, the cell averages agree relatively well with the exact solution. Both solutions contain spurious oscillations, though the oscillations under GQ- $(N + 2)$ quadrature appear qualitatively smoother and smaller.

5.1.4. Sine-shock interaction

The next benchmark problem we consider is the sine-shock interaction problem, which is posed on domain $[-5, 5]$ with initial conditions

$$\rho(x, 0) = \begin{cases} 3.857143 & x < -4 \\ 1 + .2 \sin(5x) & x \geq -4, \end{cases} \quad u(x, 0) = \begin{cases} 2.629369 & x < -4 \\ 0 & x \geq -4, \end{cases} \quad p(x, 0) = \begin{cases} 10.3333 & x < -4 \\ 1 & x \geq -4. \end{cases}$$

We simulate the solution using different quadrature rules with $N = 4$, $K = 40$ elements, and a Lax-Friedrichs flux. No TVD or positivity-preserving limiters are applied. A smaller CFL of .05 is used, and is necessary to avoid solution divergence when using GQ- $(N + 2)$ quadrature. It should be pointed out that, for GLL quadrature, it is possible to use a larger CFL of .125 without observing solution blowup. The reason for this discrepancy is the sensitivity of the evaluation $\mathbf{u}(\Pi_N \mathbf{v})$ when $\Pi_N \mathbf{v}$ differs significantly from \mathbf{v} , and is described in more detail in Section 5.1.5.

Figure 5 shows snapshots of the density at final time $T = 1.8$ for both GLL and GQ- $(N + 2)$ quadrature, along with a reference solution computed using a 5th order WENO scheme with 25000 cells [51]. For both GLL and GQ- $(N + 2)$ quadrature, the cell averages are close to the reference solution, though the solutions still contain spurious oscillations resulting from the presence of shocks and discontinuities. However, as with the Sod shock tube, we observe that these oscillations are smoother and smaller in amplitude for the choice of GQ- $(N + 2)$ quadrature.

5.1.5. Sensitivity of evaluation in terms of the projected entropy variables

The numerical experiments in previous sections suggest that solutions computed using Gauss quadrature rules can be more accurate than those computed using GLL quadratures. However, we also observe that, for the sine-shock interaction problem, a larger CFL of .125 can be taken when using GLL quadrature, whereas a much smaller CFL of .05 is required to prevent solution blowup when using GQ- $(N + 1)$ and GQ- $(N + 2)$ quadratures. For GQ- $(N + 1)$ and GQ- $(N + 2)$ quadratures, solution spikes can occur when evaluating the conservative variables in terms of the projected entropy variables at surface points. The use of GLL quadrature avoids this phenomena because of two factors: the equivalence between interpolation and projection under a $(N + 1)$ point quadrature rule, and the presence of boundary points in GLL quadrature.

Figure 6 shows snapshots of different variables for $N = 2$ and $K = 20$ elements at the fifth Runge-Kutta stage of the first timestep (just prior to the detection of negative density and pressure values) using

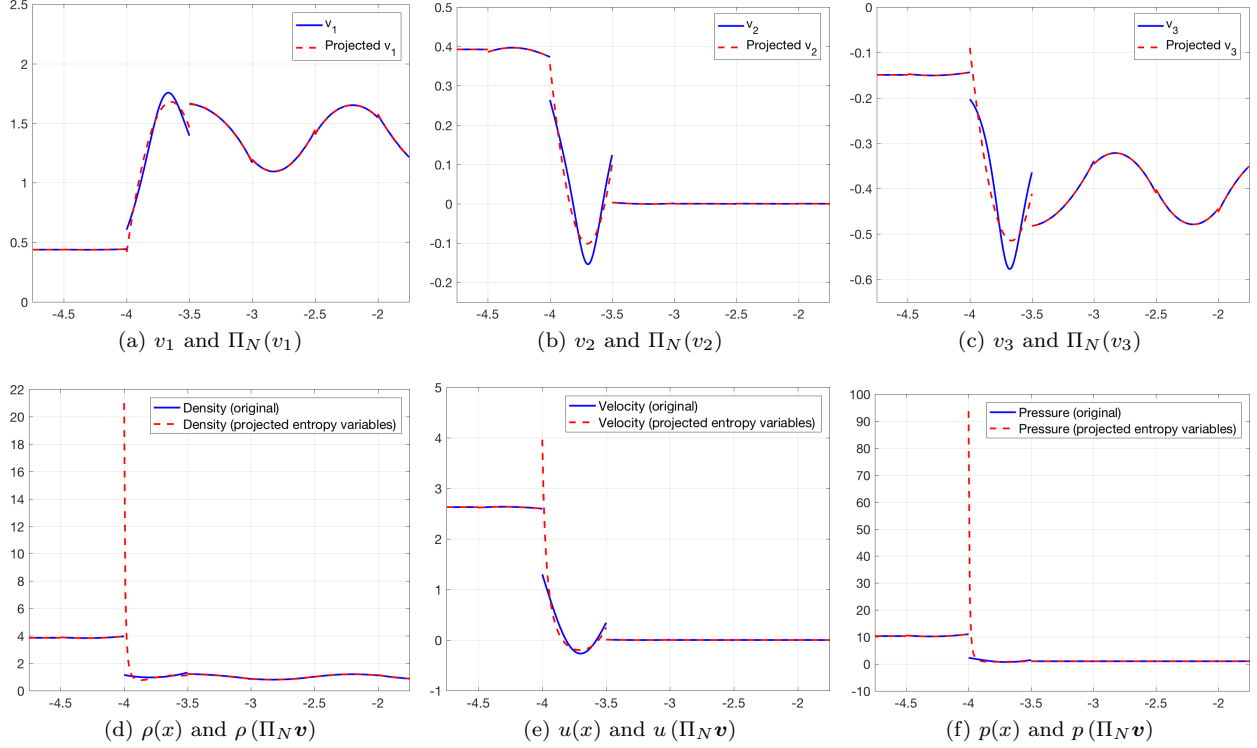


Figure 6: Top row: comparisons of entropy variables (evaluated directly as functions of conservative variables) and projected entropy variables (using $N = 2$, $K = 20$ elements, and a GQ- $(N + 2)$ quadrature rule) for the sine-shock interaction before the detection of negative density and pressures. Bottom row: comparisons of the conservative variables and their evaluations in terms of the projected entropy variables.

a GQ- $(N + 2)$ quadrature rule. Discrepancies between the evaluated and projected values of the entropy variables v_1, v_2, v_3 at element boundaries are present, and while these discrepancies are not extremely large, they produce large spikes in the conservative variables due to the sensitivity of the nonlinear evaluation $\mathbf{u}(\Pi_N \mathbf{v})$. In particular, because v_3 appears in the denominators of entropy and ρe (as functions of the entropy variables), values of v_3 near zero result in large values of density and energy. These spikes result in large oscillations in the solution if the CFL is too large, which eventually cause the density and pressure to become negative at quadrature or boundary points.

Qualitatively similar behavior is observed when using a GQ- $(N + 1)$ quadrature rule, though negative density and pressure values occur at a slightly later time. In this case, the L^2 projection reduces to interpolation at interior Gauss points. This still implies that $\mathbf{u}(\Pi_N \mathbf{v})$ does not necessarily agree with \mathbf{u} at element boundaries, and this discrepancy again leads to spikes in the conservative variables similar to those presented in Figure 6. These experiments suggest that the lack of control of boundary values in $\Pi_N \mathbf{v}$ leads to large oscillations or spikes in the boundary values of $\mathbf{u}(\Pi_N \mathbf{v})$. The presence of boundary nodes in GLL quadratures avoids this issue: $\mathbf{u}(\Pi_N \mathbf{v})$ and \mathbf{u} agree at element boundaries due to the fact that quadrature-based L^2 projection under a GLL quadrature is equivalent to interpolation at GLL points.

This scenario illustrates some of the more subtle differences between the use of GLL and GQ quadratures. We note that bound-preserving and TVD limiters [36, 37] are typically implemented in practice to detect and control spikes of the kind observed in these numerical experiments. However, slope-limiting the conservative variables \mathbf{u} may still result in spikes being generated when evaluating $\mathbf{u}(\Pi_N \mathbf{v})$ (unless the slope is set to zero such that \mathbf{u} is constant). A more nuanced approach will likely require modifying the conservative variables \mathbf{u} based on the projected entropy variables $\Pi_N \mathbf{v}$ to control oscillations while maintaining entropy conservation. These strategies will be the focus in future work.

6. Conclusions

This work presents a generalization of discretely entropy conservative methods from diagonal-norm SBP-DG methods to a more general class of high order DG methods, which include dense norm SBP operators and over-integrated quadrature rules. The resulting schemes satisfy a discrete conservation of entropy while maintaining a reasonable computational cost and small stencil. Numerical results indicate that the resulting methods deliver high order accuracy for smooth solutions while improving stability for under-resolved and shock solutions compared to non-entropy conservative and non-entropy stable solvers. We also describe differences between the sensitivity of methods based on Gauss-Lobatto quadrature and methods based on Gauss quadrature rules.

We note that, while entropy conservative and entropy stable schemes improve the robustness of solvers for nonlinear hyperbolic conservation laws, they do not address problems such as spurious oscillations in high order approximations of shock solutions or positivity preservation of density and pressure variables [6]. These issues can be addressed through the use of regularization (e.g. filtering, artificial viscosity) and/or limiting. However, this can lead to a reliance on regularization techniques as ad-hoc stabilization mechanisms. Addressing this issue requires constructing methods which avoid the need for excessive regularization and limiting in the pursuit of stability, and is a significant motivation for the development of entropy conservative and entropy stable discretizations.

Finally, we note that the analysis and numerical experiments in this work are presented in one dimension for clarity and conciseness of presentation. This work is intended as a foundation on which higher dimensional schemes can be constructed. Future work will focus on generalizations to higher dimensions (including simplicial and pyramidal elements), as well as convergence and error analysis.

7. Acknowledgments

The author thanks Lucas Wilcox, Andrew Winters, David M. Williams, Weifeng Qiu, and Paul Hand for helpful discussions. Jesse Chan is supported by the National Science Foundation under awards DMS-1719818 and DMS-1712639.

Appendix A. Assumptions on the flux function

In this work, we assume that the flux function \mathbf{f}_S admits an absolutely convergent expansion of the form

$$\mathbf{f}_S(x, y) = \sum_{i=1}^{\infty} \mathbf{f}_i(x) \mathbf{g}_i(y),$$

where $\mathbf{f}_i, \mathbf{g}_i$ are real and bounded for appropriate values of x, y . This holds, for example, for analytic \mathbf{f}_S , for which the Taylor series can be rearranged to yield the form of the desired expansion. This assumption is trivially satisfied for the Burgers' and shallow water equations, as the flux functions consist only of finite sums of products of variables. However, for the compressible Euler equations, this expansion is less clear due to the presence of the logarithmic mean in the entropy conservative Chandreshekar flux and the presence of inverses of averages and logarithmic means of the variable $\beta = \rho/(2p)$.

We first address the logarithmic mean by noting that it can be expanded out into series form using multi-index notation [52]. Let $\mu = (\mu_1, \mu_2)$ be positive integers and define $|\mu| = \mu_1 + \mu_2$; then, the logarithmic mean can be rewritten as

$$\frac{x - y}{\log(x) - \log(y)} = \sum_{n=0}^{\infty} \frac{x_n}{n!}, \quad x_n = \sum_{|\mu|=n} \frac{\log(x)^{\mu_1} \log(y)^{\mu_2}}{n+1},$$

where division by $(n+1)$ averages over the $(n+1)$ different permutations of $\mu_1, \mu_2 \geq 0$ which sum to n . Since x_n is the average of degree n powers of $\log(x), \log(y)$, we have that

$$\left| \frac{x_n}{n!} \right| \leq \frac{\max |\log(x), \log(y)|^n}{n!}.$$

The latter term is simply the n th term of the exponential Taylor series evaluated with the argument $\max |\log(x), \log(y)|$. Because the exponential Taylor series converges absolutely everywhere, the series for the logarithmic mean also converges absolutely for x, y bounded away from 0 and ∞ (i.e. x, y are in the logarithmic interval of convergence).

The entropy conservative Chandreshekar flux also requires multiplication by the terms

$$\frac{1}{\{\{\beta\}\}}, \quad \frac{1}{\{\{\beta\}\}^{\log}}, \quad \beta = \frac{\rho}{2p}.$$

These terms are analytic if $\rho, p, \{\{\beta\}\}, \{\{\beta\}\}^{\log}$ are bounded away from zero as assumed in (20). Because the composition, sum, and product of analytic functions are also analytic, the Chandreshekar flux also is analytic under positivity assumptions.

Finally, we note that the application of the operators D_N, D_h^x to convergent infinite sums should be rigorously justified. This requires D_N, D_h^x to be bounded operators mapping from a broken Sobolev space to P^N or V_h , which can be shown using boundedness of local L^2 projection operators and discrete inverse and trace inequalities [47].

References

- [1] 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.
- [2] 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.
- [3] 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.
- [4] Gregor J Gassner, Andrew R Winters, and David A Kopriva. A well balanced and entropy conservative discontinuous Galerkin spectral element method for the shallow water equations. *Applied Mathematics and Computation*, 272:291–308, 2016.
- [5] Niklas Wintermeyer, Andrew R Winters, Gregor J Gassner, and David A Kopriva. An entropy stable nodal discontinuous Galerkin method for the two dimensional shallow water equations on unstructured curvilinear meshes with discontinuous bathymetry. *Journal of Computational Physics*, 340:200–242, 2017.
- [6] 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.
- [7] 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.
- [8] Hendrik Ranocha, Philipp Öffner, and Thomas Sonar. Summation-by-parts operators for correction procedure via reconstruction. *Journal of Computational Physics*, 311:299–328, 2016.
- [9] Hendrik Ranocha, Philipp Öffner, and Thomas Sonar. Extended skew-symmetric form for summation-by-parts operators and varying Jacobians. *Journal of Computational Physics*, 342:13–28, 2017.
- [10] Jan S Hesthaven and Tim Warburton. *Nodal discontinuous Galerkin methods: algorithms, analysis, and applications*, volume 54. Springer, 2007.

- [11] Andreas Klöckner, Tim Warburton, Jeff Bridge, and Jan S Hesthaven. Nodal discontinuous Galerkin methods on graphics processors. *Journal of Computational Physics*, 228(21):7863–7882, 2009.
- [12] Mark Ainsworth. Dispersive and dissipative behaviour of high order discontinuous Galerkin finite element methods. *Journal of Computational Physics*, 198(1):106–130, 2004.
- [13] Lucas C Wilcox, Georg Stadler, Carsten Burstedde, and Omar Ghattas. A high-order discontinuous Galerkin method for wave propagation through coupled elastic–acoustic media. *Journal of Computational Physics*, 229(24):9373–9396, 2010.
- [14] 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.
- [15] Lilia Krivodonova. Limiters for high-order discontinuous Galerkin methods. *Journal of Computational Physics*, 226(1):879–896, 2007.
- [16] Per-Olof Persson and Jaime Peraire. Sub-cell shock capturing for discontinuous Galerkin methods. *AIAA*, 112, 2006.
- [17] Robert M Kirby and George Em Karniadakis. De-aliasing on non-uniform grids: algorithms and applications. *Journal of Computational Physics*, 191(1):249–264, 2003.
- [18] T Warburton. A low-storage curvilinear discontinuous Galerkin method for wave problems. *SIAM Journal on Scientific Computing*, 35(4):A1987–A2012, 2013.
- [19] Jesse Chan, Russell J Hewett, and T Warburton. Weight-adjusted discontinuous Galerkin methods: wave propagation in heterogeneous media. *arXiv preprint arXiv:1608.01944*, 2016.
- [20] Jesse Chan, Russell J Hewett, and T Warburton. Weight-adjusted discontinuous Galerkin methods: curvilinear meshes. *arXiv preprint arXiv:1608.03836*, 2016.
- [21] Jesse Chan. Weight-adjusted discontinuous Galerkin methods: matrix-valued weights and elastic wave propagation in heterogeneous media. *arXiv preprint arXiv:1701.00215*, 2017.
- [22] Eitan Tadmor. The numerical viscosity of entropy stable schemes for systems of conservation laws. I. *Mathematics of Computation*, 49(179):91–103, 1987.
- [23] Deep Ray, Praveen Chandrashekar, Ulrik S Fjordholm, and Siddhartha Mishra. Entropy stable scheme on two-dimensional unstructured grids for Euler equations. *Communications in Computational Physics*, 19(5):1111–1140, 2016.
- [24] Ulrik S Fjordholm, Siddhartha Mishra, and Eitan Tadmor. Arbitrarily high-order accurate entropy stable essentially nonoscillatory schemes for systems of conservation laws. *SIAM Journal on Numerical Analysis*, 50(2):544–573, 2012.
- [25] 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.
- [26] 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.
- [27] Jason E Hicken, David C Del Rey Fernandez, and David W Zingg. Multidimensional summation-by-parts operators: General theory and application to simplex elements. *SIAM Journal on Scientific Computing*, 38(4):A1935–A1958, 2016.

- [28] 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.
- [29] 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.
- [30] MJS Chin-Joe-Kong, WA Mulder, and M Van Veldhuizen. Higher-order triangular and tetrahedral finite elements with mass lumping for solving the wave equation. *Journal of Engineering Mathematics*, 35(4):405–426, 1999.
- [31] Elena Zhebel, Sara Minisini, Alexey Kononov, and Wim A Mulder. A comparison of continuous mass-lumped finite elements with finite differences for 3-D wave propagation. *Geophysical Prospecting*, 62(5):1111–1125, 2014.
- [32] Jesse Chan and T Warburton. Orthogonal bases for vertex-mapped pyramids. *SIAM Journal on Scientific Computing*, 38(2):A1146–A1170, 2016.
- [33] Hendrik Ranocha. Comparison of some entropy conservative numerical fluxes for the Euler equations. *arXiv preprint arXiv:1701.02264*, 2017.
- [34] Gregor J Gassner, Andrew R Winters, Florian J Hindenlang, and David A Kopriva. The BR1 Scheme is Stable for the Compressible Navier-Stokes Equations. *arXiv preprint arXiv:1704.03646*, 2017.
- [35] Michael S Mock. Systems of conservation laws of mixed type. *Journal of Differential equations*, 37(1):70–88, 1980.
- [36] Xiangxiong Zhang and Chi-Wang Shu. On positivity-preserving high order discontinuous Galerkin schemes for compressible Euler equations on rectangular meshes. *Journal of Computational Physics*, 229(23):8918–8934, 2010.
- [37] Xiangxiong Zhang, Yinhua Xia, and Chi-Wang Shu. Maximum-principle-satisfying and positivity-preserving high order discontinuous Galerkin schemes for conservation laws on triangular meshes. *Journal of Scientific Computing*, 50(1):29–62, 2012.
- [38] Susanne Brenner and Ridgway Scott. *The mathematical theory of finite element methods*, volume 15. Springer Science & Business Media, 2007.
- [39] David C Fernández, Jason E Hicken, and David W Zingg. Simultaneous approximation terms for multi-dimensional summation-by-parts operators. *arXiv preprint arXiv:1605.03214*, 2016.
- [40] Jan S Hesthaven and T Warburton. High-order nodal discontinuous Galerkin methods for the Maxwell eigenvalue problem. *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, 362(1816):493–524, 2004.
- [41] Daniele Antonio Di Pietro and Alexandre Ern. *Mathematical aspects of discontinuous Galerkin methods*, volume 69. Springer Science & Business Media, 2011.
- [42] Junping Wang and Xiu Ye. A weak Galerkin finite element method for second-order elliptic problems. *Journal of Computational and Applied Mathematics*, 241:103–115, 2013.
- [43] 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.

- [44] Cengke Shi and Chi-Wang Shu. On local conservation of numerical methods for conservation laws. *Computers and Fluids*, 2017.
- [45] George PH Styan. Hadamard products and multivariate statistical analysis. *Linear algebra and its applications*, 6:217–240, 1973.
- [46] Mark H Carpenter and Christopher A Kennedy. Fourth-order $2n$ -storage Runge-Kutta schemes. Technical Report NASA-TM-109112, NAS 1.15:109112, NASA Langley Research Center, 1994.
- [47] 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.
- [48] T Warburton and Jan S Hesthaven. On the constants in hp -finite element trace inverse inequalities. *Computer methods in applied mechanics and engineering*, 192(25):2765–2773, 2003.
- [49] Amiram Harten. On the symmetric form of systems of conservation laws with entropy. *Journal of computational physics*, 49(1):151–164, 1983.
- [50] Farzad Ismail and Philip L Roe. Affordable, entropy-consistent Euler flux functions II: Entropy production at shocks. *Journal of Computational Physics*, 228(15):5410–5436, 2009.
- [51] Chi-Wang Shu. High order weighted essentially nonoscillatory schemes for convection dominated problems. *SIAM review*, 51(1):82–126, 2009.
- [52] Seppo Mustonen. Logarithmic mean for several arguments. <http://www.survo.fi/papers/logmean.pdf>, 2002.