High order entropy stable discontinuous Galerkin methods on curvilinear meshes

Jesse Chan, Lucas Wilcox

Abstract

Things to include: entropy stability for curvilinear meshes, choosing geometric factors to ensure constant state preservation, high order accuracy.

1 Intro/layout

1. Global DG differentiation operators and flux differencing
2. Global DG operators for curvilinear meshes
3. Flux differencing with geometric averaging.

2 Flux differencing for affine meshes

Let $\boldsymbol{F}(\boldsymbol{U})$ denote the flux matrix whose rows are

$$\boldsymbol{F}(\boldsymbol{U}) = \begin{pmatrix} \boldsymbol{F}_x \\ \boldsymbol{F}_y \end{pmatrix}.$$

The conservation law of interest is the following

$$\frac{\partial \boldsymbol{U}}{\partial t} + \nabla \cdot \boldsymbol{F}(\boldsymbol{U}) = 0,$$

where the divergence is taken over each column of $\boldsymbol{F}(\boldsymbol{U})$.

Define the global DG divergence

$$\begin{aligned}\left(\nabla_h \cdot \boldsymbol{u}, vw\right)_\Omega = \sum_k \left(\nabla \cdot \Pi_N \boldsymbol{u}, vw\right)_{D^k} \\ + \frac{1}{2}\left\langle \boldsymbol{u}^+ - \Pi_N \boldsymbol{u}, vw\boldsymbol{n}\right\rangle_{\partial D^k} + \frac{1}{2}\left\langle \boldsymbol{u} - \Pi_N \boldsymbol{u}, \Pi_N\left(vw\right)\boldsymbol{n}\right\rangle_{\partial D^k}.\end{aligned}$$

This is equivalent to defining

$$\nabla_h \cdot \boldsymbol{u} = \sum_i D_h^i \boldsymbol{u}_i.$$

Let $\boldsymbol{f}_S^i(\boldsymbol{u}_L, \boldsymbol{u}_R)$ for $i = 1, \ldots, d$ denote the multi-dimensional entropy conservative flux of Tadmor [1], such that

$$\left(\boldsymbol{v}_L - \boldsymbol{v}_R\right)^T \boldsymbol{f}_S^i(\boldsymbol{u}_L, \boldsymbol{u}_R) = \psi_{i,L} - \psi_{i,R}$$

where ψ_i denotes the ith component of the entropy potential. We concatenate these components together into a matrix $\boldsymbol{f}_S$ whose ith row is $\boldsymbol{f}_S^i$

$$\boldsymbol{f}_S = \left[\boldsymbol{f}_S^1, \ldots, \boldsymbol{f}_S^d\right]^T.$$

Theorem 1. *The semi-discrete DG discretization*

$$\left(\frac{\partial \boldsymbol{u}}{\partial t} + 2\left.\left(\nabla_h \cdot \boldsymbol{f}_S(\boldsymbol{u},\boldsymbol{u}')\right)\right|_{\boldsymbol{x}'=\boldsymbol{x}}, \boldsymbol{w}\right)_{\Omega} = 0$$

is entropy conservative.

Proof. Take $\boldsymbol{w}=\boldsymbol{v}$. The time term reduces to $\frac{\partial S(\boldsymbol{u})}{\partial t}$ by the chain rule. The spatial term is then

$$\begin{aligned}
2\left(\left.\left(\nabla_h \cdot \boldsymbol{f}_S(\boldsymbol{u},\boldsymbol{u}')\right)\right|_{\boldsymbol{x}'=\boldsymbol{x}}, \boldsymbol{v}\right) &= \left(\left.\left(\nabla_h \cdot \boldsymbol{f}_S(\boldsymbol{u},\boldsymbol{u}')\boldsymbol{v}'\right)\right|_{\boldsymbol{x}'=\boldsymbol{x}}, 1\right) + \sum_{i=1}^{d}\left(\left.\left(D_h^i \boldsymbol{f}_S^i(\boldsymbol{u},\boldsymbol{u}')\right)\right|_{\boldsymbol{x}'=\boldsymbol{x}}, \boldsymbol{v}\right)\\
&= \sum_{i=1}^{d}\left\langle \boldsymbol{f}_S^i(\boldsymbol{u},\boldsymbol{u}), \boldsymbol{v}\boldsymbol{n}_i\right\rangle - \left(\left.\left(D_h^i\left(\left(\boldsymbol{v}'-\boldsymbol{v}\right)^T \boldsymbol{f}_S^i(\boldsymbol{u},\boldsymbol{u}')\right)\right)\right|_{\boldsymbol{x}'=\boldsymbol{x}}, 1\right)
\end{aligned}$$

The boundary term is simplified by noting that $\boldsymbol{f}_S^i(\boldsymbol{u},\boldsymbol{u}) = \boldsymbol{f}^i(\boldsymbol{u})$. Applying properties of the entropy conservative flux yields

$$\left(\boldsymbol{v}'-\boldsymbol{v}\right)^T \boldsymbol{f}_S^i(\boldsymbol{u},\boldsymbol{u}') = \psi_i' - \psi_i,$$

which simplifies the final volume term to

$$\left(\left.\left(D_h^i\left(\psi_i'-\psi_i\right)\right)\right|_{\boldsymbol{x}'=\boldsymbol{x}}, 1\right) = \left\langle -\psi_i \boldsymbol{n}_i, 1\right\rangle.$$

□

3 Curved meshes

Let $\boldsymbol{G}$ denote the Jacobian of the geometric mapping

$$\boldsymbol{G}_{ij} = \frac{\partial \widehat{\boldsymbol{x}}_j}{\partial \boldsymbol{x}_i},$$

and let J denote the determinant of $\boldsymbol{G}$. One can show that

$$\widehat{\nabla}\cdot\left(J\boldsymbol{G}^T\right) = 0, \qquad \widehat{\nabla}\cdot\left(J\boldsymbol{G}^T\boldsymbol{u}\right) = \nabla\cdot\boldsymbol{u}$$

at the continuous level. We also have that the mapped normals obey the property

$$J\boldsymbol{G}\widehat{\boldsymbol{n}}\widehat{J}^f = \boldsymbol{n}J^f.$$

At the continuous level, the physical gradient and divergence satisfy

$$\left(\nabla u, \boldsymbol{v}\right)_{D^k} = \left(J\boldsymbol{G}\widehat{\nabla}u, \boldsymbol{v}\right)_{\widehat{D}}, \qquad \left(\nabla\cdot\boldsymbol{u}, v\right)_{D^k} = \left(\widehat{\nabla}\cdot\left(J\boldsymbol{G}^T u\right), \boldsymbol{v}\right)_{\widehat{D}},$$

as well as a corresponding integration by parts property.

3.1 Flux differencing for curvilinear meshes

We introduce a "reference" global DG derivative, which is defined (for $v \in V_h$ but $u, w \notin V_h$)

$$\left(\widehat{D}_h^i u, vw\right)_{\Omega} = \sum_k \left(\frac{\partial \Pi_N \boldsymbol{u}}{\partial \widehat{\boldsymbol{x}}_i}, vw\right)_{\widehat{D}} + \frac{1}{2}\left\langle \boldsymbol{u}^+ - \Pi_N\boldsymbol{u}, vw\widehat{\boldsymbol{n}}_i\right\rangle_{\partial\widehat{D}} + \frac{1}{2}\left\langle \boldsymbol{u} - \Pi_N\boldsymbol{u}, \Pi_N\left(vw\right)\widehat{\boldsymbol{n}}_i\right\rangle_{\partial\widehat{D}}.$$

We use this to define a reference DG divergence

$$\widehat{\nabla}_h \cdot \boldsymbol{u} = \sum_{i=1}^{d} \widehat{D}_h^i \boldsymbol{u}_i.$$

We next introduce two physical DG divergences based on the reference DG divergence. The a "conservative" DG divergence is defined using the reference DG divergence operator

$$\nabla_h^c \cdot \boldsymbol{u} = \widehat{\nabla}_h \cdot \left(J\boldsymbol{G}^T\boldsymbol{u}\right).$$

We also introduce the "non-conservative" divergence

$$\nabla_h^{nc} \cdot \boldsymbol{u} = \left(J\boldsymbol{G}\widehat{\nabla}_h\right) \cdot \boldsymbol{u} = \sum_{i=1}^{d}\sum_{j=1}^{d} J\boldsymbol{G}_{ij}\widehat{D}_h^j \boldsymbol{u}_i.$$

Unless $J\boldsymbol{G}$ is element-wise constant,

$$\widehat{\nabla}_h \cdot \left(J\boldsymbol{G}^T\boldsymbol{u}\right) \neq \left(J\boldsymbol{G}\widehat{\nabla}_h\right) \cdot \boldsymbol{u},$$

due to the presence of the projection operator Π_N in the definition of $\widehat{\nabla}_h$.

Instead of choosing one definition over the other, it was shown in [2] that stability is achieved when using the average of these two definitions of the physical gradient. The following lemma describes how to implement this within the flux differencing framework.

Lemma 1. *Let $\boldsymbol{u}^k(\boldsymbol{x},\boldsymbol{x}')$ be defined as*

$$\boldsymbol{u}^k(\boldsymbol{x},\boldsymbol{x}') = \{\{J\boldsymbol{G}^T\}\}\,\boldsymbol{u} = \left(\frac{J(\boldsymbol{x})\boldsymbol{G}^T(\boldsymbol{x}) + J(\boldsymbol{x}')\boldsymbol{G}^T(\boldsymbol{x}')}{2}\right)\boldsymbol{u}.$$

Applying the reference global DG divergence to $\boldsymbol{u}^k$ is equivalent to averaging both definitions of the global DG divergence

$$\left.\left(\widehat{\nabla}_h \cdot \boldsymbol{u}^k(\boldsymbol{x},\boldsymbol{x}')\right)\right|_{\boldsymbol{x}'=\boldsymbol{x}} = \frac{1}{2}\left(\widehat{\nabla}_h \cdot \left(J\boldsymbol{G}^T\boldsymbol{u}\right) + \left(J\boldsymbol{G}\widehat{\nabla}_h\right)\cdot\boldsymbol{u}\right) = \frac{1}{2}\left(\nabla_h^c \cdot \boldsymbol{u} + \nabla_h^{nc}\cdot\boldsymbol{u}\right).$$

Proof. TBD. □

We now have the following curvilinear generalization of Theorem 1

Theorem 2. *The semi-discrete DG discretization*

$$\left(\frac{\partial \boldsymbol{u}}{\partial t} + 2\left.\left(\widehat{\nabla}_h \cdot \boldsymbol{f}_S^k(\boldsymbol{u},\boldsymbol{u}')\right)\right|_{\boldsymbol{x}'=\boldsymbol{x}}, \boldsymbol{w}\right) = 0, \qquad \boldsymbol{f}_S^k(\boldsymbol{u}_L,\boldsymbol{u}_R) = \{\{J\boldsymbol{G}^T\}\}\,\boldsymbol{f}_S(\boldsymbol{u}_L,\boldsymbol{u}_R).$$

is entropy conservative on curvilinear meshes.

Proof. Take $\boldsymbol{w} = \boldsymbol{v}$. The time term is the same as before; however, we now have two spatial terms corresponding to the conservative and non-conservative DG divergences. Essentially, integrating the conservative divergence terms by parts results in volume contributions with the non-conservative divergence, and vice-versa. Contributions are swapped between each type of divergence, so that the last steps of the affine entropy conservation proof can be repeated for two volume terms (involving the conservative and non-conservative divergence, respectively).

□

3.2 Weight-adjusted DG

We define the curvilinear L^2 projection Π_N^k such that

$$\left(\Pi_N^k u, v\right)_{D^k} = \left(J\Pi_N^k u, v\right)_{\widehat{D}} = \left(uJ, v\right)_{\widehat{D}} = \left(u, v\right)_{D^k}.$$

Let the weight-adjusted projection operator P_N be defined as $P_N u = \Pi_N\left(\frac{1}{J}\Pi_N\left(uJ\right)\right)$. Note that P_N is self-adjoint with respect to the J-weighted L^2 inner product

$$\left(JP_N u, v\right) = \left(\Pi_N\left(\frac{1}{J}\Pi_N\left(uJ\right)\right), vJ\right) = \left(uJ, \Pi_N\left(\frac{1}{J}\Pi_N\left(vJ\right)\right)\right) = \left(uJ, P_N v\right).$$

Let $\Pi_N^k u$ be the L^2 projection of u with respect to the weighted (curvilinear) L^2 inner product. We observe in numerical experiments that for a fixed geometric mapping, $\left\|\Pi_N^k u - P_N u\right\|_{L^2(\Omega)} = O(h^{N+2})$. Because the difference between the L^2 and WADG projection is superconvergent, the results are indistinguishable for a fixed geometric mapping.

We can prove this bound using results from [3, 4]. We first define the operator T_w^{-1} as follows

$$\left(wT_w^{-1}u, v\right)_{\widehat{D}} = \left(u, v\right)_{\widehat{D}}, \qquad \forall v \in V_h.$$

This forms the basis of the weight-adjusted approximation of weighted L^2 inner products. The first theorem we need shows that $T_{1/J}^{-1}$ can be used to approximate weighted curvilinear L^2 inner products with order $(2N+2)$ accuracy.

Theorem 3. *Let $u \in W^{N+1,2}\left(D^k\right)$ and $v \in V_h\left(D^k\right)$. Then,*

$$\begin{aligned}&\left|\left(u, vJ\right)_{\widehat{D}} - \left(T_{1/J}^{-1}u, v\right)_{\widehat{D}}\right| \leq \\ &Ch^{2N+2}\left\|J\right\|_{L^\infty(D^k)}\left\|\frac{1}{J}\right\|_{L^\infty(D^k)}^2\left\|J\right\|_{W^{N+1,\infty}(D^k)}^2\left\|u\right\|_{W^{N+1,2}(D^k)}\left\|v\right\|_{W^{N+1,2}(D^k)}.\end{aligned}$$

Proof. The proof involves straightforward adaptations of Theorem 4, Theorem 5, and Theorem 6 in [3] to the reference element $\widehat{D}$. ☐

The next result we need is a generalized inverse inequality.

Lemma 2. *Let $v \in P^N\left(D^k\right)$, and let $h = \operatorname{diam}\left(D^k\right)$. Then,*

$$\left\|v\right\|_{W^{N+1,2}(D^k)} \leq C_N h^{-N}\left\|v\right\|_{L^2(D^k)}.$$

where C_N is independent of h.

Proof. The result is the consequence of a scaling argument and a Rayleigh quotient bound involving the largest eigenvalue of the generalized eigenvalue problem

$$\boldsymbol{K}_N\boldsymbol{u} = \lambda\boldsymbol{M}\boldsymbol{u},$$

where $\boldsymbol{M}$ is the L^2 mass matrix over D^k and $\boldsymbol{K}_N$ is the Gram matrix corresponding to the Sobolev inner product for $W^{N+1,2}\left(D^k\right)$. We note that the constant C_N depends on the largest eigenvalue, which in turn depends on the order N and dimension d. ☐

We can now prove that $P_N u$ is superconvergent to the curvilinear L^2 projection $\Pi_N^k u$

Theorem 4. *Let $u \in W^{N+1,2}\left(D^k\right)$. The difference between the L^2 projection $\Pi_N^k u$ and the weight-adjusted projection $P_N u$ is*

$$\left\|\Pi_N^k u - P_N u\right\|_{L^2(D^k)} \leq C \left\|J\right\|_{L^\infty(D^k)} \left\|\frac{1}{J}\right\|^2_{L^\infty(D^k)} \left\|J\right\|^2_{W^{N+1,\infty}(D^k)} h^{N+2} \left\|u\right\|_{W^{N+1,2}(D^k)}.$$

where C is a mesh-independent constant.

Proof. Let $\Pi_N^k u$ be the L^2 projection over the curved element D^k, such that

$$\left(J\Pi_N^k u, v\right) = \left(uJ, v\right), \qquad \forall v \in V_h.$$

By definition, $P_N u$ satisfies a similar property

$$\left(T^{-1}_{1/J} P_N u, v\right) = \left(uJ, v\right), \qquad \forall v \in V_h.$$

Then, we have that

$$\left\|\Pi_N^k u - P_N u\right\|^2_{L^2(D^k)} = \left(\Pi_N^k u - P_N u, vJ\right)_{\widehat{D}}, \qquad v = \Pi_N^k u - P_N u.$$

Because $v \in P^N\left(D^k\right)$, we can also evaluate the squared error as

$$\begin{aligned}\left\|\Pi_N^k u - P_N u\right\|^2_{L^2(D^k)} &= \left|\left(\Pi_N^k, vJ\right)_{\widehat{D}} - \left(P_N u, vJ\right)_{\widehat{D}}\right| \\ &= \left|\left(u, vJ\right)_{\widehat{D}} - \left(P_N u, vJ\right)_{\widehat{D}}\right| = \left|\left(T^{-1}_{1/J} P_N u, v\right)_{\widehat{D}} - \left(JP_N u, v\right)_{\widehat{D}}\right|.\end{aligned}$$

Applying Theorem 3 and Lemma 2 then yields that

$$\begin{aligned}\left\|\Pi_N^k u - P_N u\right\|^2_{L^2(D^k)} &\leq Ch^{2N+2} C_J \left\|u\right\|_{W^{N+1,2}(D^k)} \left\|v\right\|_{W^{N+1,2}(\widehat{D})} \\ &\leq Ch^{N+2} C_J \left\|u\right\|_{W^{N+1,2}(D^k)} \left\|v\right\|_{L^2(D^k)},\end{aligned}$$

where C_J is a mesh-independent constant depending on J. Dividing through by

$$\left\|v\right\|_{L^2(D^k)} = \left\|\Pi_N^k u - P_N u\right\|_{L^2(D^k)}$$

gives the desired result. $\square$

4 Free stream preservation

We seek conditions for which free-stream preservation

$$\frac{\partial \boldsymbol{u}}{\partial t} = \nabla_h \cdot \boldsymbol{u} = 0$$

is satisfied if $\boldsymbol{u}$ constant. Free stream preservation is not always maintained at the discrete level in 3D due to the fact that geometric factors are higher degree polynomials than the corresponding discrete space [5, 6]. For curvilinear meshes, $\widehat{\nabla} \cdot J\boldsymbol{G}^T \neq 0$ due to polynomial aliasing of geometric factors.

This can be remedied by using an interpolation of the curl-conservative form of the geometric factors, which ensures that $\widehat{\nabla} \cdot \left(J\boldsymbol{G}^T\right) = 0$ locally [7, 5]. However, because the geometric factors are computed by applying the curl, the geometric factors are approximated as degree $(N-1)$ polynomials rather than degree N. We take a different approach, based on a strategy described in [8].

Integrating by parts the DG formulation yields, for any constant $\boldsymbol{u}$

$$
\begin{aligned}
\left(\nabla_h \cdot \boldsymbol{u}, v\right)_{\Omega} &= \sum_k \left(\widehat{\nabla} \cdot \left(\boldsymbol{J}\boldsymbol{G}^T \boldsymbol{u}\right), v\right)_{\widehat{D}} + \left\langle \boldsymbol{J}\boldsymbol{G}^T\boldsymbol{u} - \Pi_N \left(\boldsymbol{J}\boldsymbol{G}^T \boldsymbol{u}\right), v\widehat{\boldsymbol{n}}\right\rangle_{\partial \widehat{D}} \\
&= \sum_k \left(-\boldsymbol{J}\boldsymbol{G}^T \boldsymbol{u}, \widehat{\nabla} v\right)_{\widehat{D}} + \left\langle \left(\boldsymbol{J}\boldsymbol{G}\widehat{\boldsymbol{n}}\right)^T \boldsymbol{u}, v\right\rangle_{\partial \widehat{D}} \\
&= \sum_k \left(-\boldsymbol{J}\boldsymbol{G}^T \boldsymbol{u}, \widehat{\nabla} v\right)_{\widehat{D}} + \left\langle \boldsymbol{n} \cdot \boldsymbol{u}, v\right\rangle_{\partial \widehat{D}}.
\end{aligned}
$$

for any $v \in V_h$. Thus, to ensure that this sums to zero, we modify the geometric factors by seeking $\boldsymbol{J}\boldsymbol{G}$ which minimizes the L^2 error for a degree N polynomial approximation to the true geometric factors, subject to a weakly divergence-free constraint

$$
\begin{aligned}
&\min_{\tilde{\boldsymbol{J}\boldsymbol{G}}_i \in P^N} \frac{1}{2} \left\| \tilde{\boldsymbol{J}\boldsymbol{G}}_i - \boldsymbol{J}\boldsymbol{G}_i \right\|^2_{L^2\left(\widehat{D}\right)}, \\
\text{s.t. } &\left\langle \boldsymbol{n}_i, v\right\rangle_{\partial \widehat{D}} - \left(\tilde{\boldsymbol{J}\boldsymbol{G}}_i, \nabla v\right)_{\widehat{D}} = 0, \qquad \forall v \in P^N.
\end{aligned}
$$

We note that the constraint corresponding to $v = 1$ yields that $\left\langle \boldsymbol{n}_i, v\right\rangle_{\partial \widehat{D}} = 0$. Thus, in order to guarantee a solution to this problem, we require that $\left\langle \boldsymbol{n}_i, v\right\rangle_{\partial \widehat{D}} = 0$ for consistency of the right hand side. This is satisfied if the mapping is isoparametric: the cross product formula is exact at quadrature points and the geometric factors are degree $2N-2$ polynomials, which implies that for $v = 1$, $\left\langle \boldsymbol{n}_i, v\right\rangle_{\partial \widehat{D}}$ is exactly integrated by a surface quadrature of degree $2N$.

This strategy was first introduced in the context of SBP-SAT terms in [8]. We approach its implementation slightly differently, and use the fact that the quadratic program can be solved explicitly using the null space method, which is computationally feasible since the null space of the constraint matrix is computed only once on the reference element. This null space corresponds to a discretely divergence-free basis, which we extract using the SVD.

The resulting free-stream preserving geometric factors result in separate approximations of the volume and surface geometric factors. The surface geometric factors are constructed at surface quadrature points, which guarantees that neighboring surface normal terms cancel for watertight meshes. The volume geometric factors are constructed at quadrature points and projected onto a polynomial basis of degree N with weakly divergence-free constraints involving the surface normals.

5 Limiting

The evaluation of $\boldsymbol{u}_v = \boldsymbol{u}\left(\Pi_N v\right)$ can increase entropy pointwise, such that $S(\boldsymbol{u}_v) \geq S(\boldsymbol{u})$. This can manifest as spikes in $\boldsymbol{u}_v$. We wish to mollify the effect of such spikes.

The first approach is to limit the conservative variable $\boldsymbol{u}$

$$
\tilde{\boldsymbol{u}} = \bar{\boldsymbol{u}} + \Theta(\boldsymbol{u} - \bar{\boldsymbol{u}})
$$

where Θ is some diagonal matrix with entries in $[0, 1]$. We want to ensure that $\rho, E - \frac{\rho u^2}{2} > 0$.

The entropy for the compressible Euler equations is

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

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

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

where the internal energy $\rho e = E - u^2/2$.

The inverse mapping is given by

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

where ρe and s in terms of the entropy variables are

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

The mapping is invertible so long as $\rho, E - \frac{\rho u^2}{2} > 0$, which can be ensured using standard limiters. However, we have to ensure also that $\rho(\Pi_N \boldsymbol{v}) > 0$ (similarly for internal energy). This boils down to ensuring that $\Pi_N v_3(x) < 0$, which guarantees that $(\rho e) > 0$ as well.

It can be helpful to ensure a stronger condition, that $\Pi_N v_3(x) \leq \max_x v_3(x)$. This guarantees a bound constraint on the conservative variables evaluated using the projected entropy variables.

Another approach is to change the time-step size based on the difference between $\boldsymbol{u}$ and $\boldsymbol{u}_v$. This would be similar to local time-stepping (high order interpolation of the numerical fluxes and multiple evaluations of)

- Adaptively choose time-step size?
- Expensive option: bisection or Newton algorithm for finding theta.

6 Implementation

The semi-discrete evolution equation is as follows

$$\frac{\partial \boldsymbol{u}}{\partial t} = \begin{bmatrix} \boldsymbol{P}_q & \boldsymbol{L}_q \end{bmatrix} \left(\begin{bmatrix} \boldsymbol{D}_q & \boldsymbol{V}_q \boldsymbol{L}_q \\ \boldsymbol{V}_f \boldsymbol{P}_q & \boldsymbol{I} \end{bmatrix} \circ \begin{bmatrix} \boldsymbol{f}_S(\boldsymbol{u}, \boldsymbol{u}_f) & \boldsymbol{f}_S(\boldsymbol{u}, \boldsymbol{u}_f) \\ \boldsymbol{f}_S(\boldsymbol{u}, \boldsymbol{u}_f) & \boldsymbol{f}_S(\boldsymbol{u}_f^+, \boldsymbol{u}_f) \end{bmatrix} \boldsymbol{1} \right).$$

where $\boldsymbol{u}, \boldsymbol{u}_f$ are evaluations at volume and surface quadrature points.

The method is outlined as follows: we store the conservation variables $\boldsymbol{u}$ at quadrature points, and compute projected entropy variables $\boldsymbol{v}_u = \Pi_N(\boldsymbol{v}(\boldsymbol{u}))$

1. Apply volume flux differencing ($\boldsymbol{D}_q$ and $\boldsymbol{V}_q\boldsymbol{L}_q$) and projection ($\boldsymbol{V}_q\boldsymbol{P}_q$) (volume kernel)
2. Apply surface flux differencing ($\boldsymbol{V}_f\boldsymbol{P}_q$) and lifting ($\boldsymbol{V}_q\boldsymbol{L}_q$) (volume kernel)
3. Apply WADG (scale by $1/J$ and apply $\boldsymbol{V}_q\boldsymbol{P}_q$ and update $\boldsymbol{u}$ at volume quad points. Compute projected entropy variables $\boldsymbol{v}_u = \Pi_N(\boldsymbol{v}(\boldsymbol{u}))$, interpolate to volume quad points, and compute conservative variable volume values (update)
4. Interpolate entropy variables at surface quadrature points and write out conservative variable surface values (face).

References

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

[2] 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.

[3] Jesse Chan, Russell J Hewett, and T Warburton. Weight-adjusted discontinuous Galerkin methods: wave propagation in heterogeneous media. *arXiv preprint arXiv:1608.01944*, 2016.

[4] Jesse Chan, Russell J Hewett, and T Warburton. Weight-adjusted discontinuous Galerkin methods: curvilinear meshes. *arXiv preprint arXiv:1608.03836*, 2016.

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

[6] Amaury Johnen, J.F. Remacle, and Christophe Geuzaine. Geometrical validity of curvilinear finite elements. *Journal of Computational Physics*, 233:359–372, 2013.

[7] Miguel R Visbal and Datta V Gaitonde. On the use of higher-order finite-difference schemes on curvilinear and deforming meshes. *Journal of Computational Physics*, 181(1):155–185, 2002.

[8] 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.