Entropy stable high order discontinuous Galerkin methods for hyperbolic conservation laws

Chi-Wang Shu

Division of Applied Mathematics
Brown University

Joint work with Tianheng Chen, and with Yong Liu and Mengping Zhang
Outline

- Introduction
- Entropy stable high order DG schemes in one dimension
- Generalization to triangular meshes
- Generalization to convection-diffusion equations
- Numerical experiments
- Concluding remarks
The discontinuous Galerkin (DG) methods are a class of finite element methods for solving hyperbolic conservation laws.

In 1D the conservation law is

\[ u_t + f(u)_x = 0 \]

and in the system case \( u \) is a vector, and the Jacobian \( f'(u) \) is diagonalizable with real eigenvalues.

In 2D the equation is

\[ u_t + f(u)_x + g(u)_y = 0. \]
Several properties of the solutions to hyperbolic conservation laws:

- The solution $u$ may become discontinuous regardless of the smoothness of the initial condition.

- Weak solutions are not unique. The unique, physically relevant entropy solution satisfies additional entropy inequalities

$$U(u)_t + F(u)_x \leq 0$$

in the distribution sense, where $U(u)$ is a convex scalar function of $u$ and the entropy flux $F(u)$ satisfies $F'(u) = U'(u)f'(u)$.
To solve the hyperbolic conservation law:

\[ u_t + f(u)_x = 0, \]  

(1)

we multiply the equation with a test function \( v \), integrate over a cell 

\( I_j = [x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}}] \), and integrate by parts:

\[
\int_{I_j} u_t v \, dx - \int_{I_j} f(u)v_x \, dx + f(u_{j+\frac{1}{2}})v_{j+\frac{1}{2}} - f(u_{j-\frac{1}{2}})v_{j-\frac{1}{2}} = 0
\]
Now assume both the solution $u$ and the test function $v$ come from a finite dimensional approximation space $V_h$, which is usually taken as the space of piecewise polynomials of degree up to $k$:

$$V_h = \{ v : v|_{I_j} \in P^k(I_j), \ j = 1, \cdots, N \}$$

However, the boundary terms $f(u_j + \frac{1}{2}), v_j + \frac{1}{2}$ etc. are not well defined when $u$ and $v$ are in this space, as they are discontinuous at the cell interfaces.
From the conservation and stability (upwinding) considerations, we take

- A single valued numerical flux to replace \( f(u_{j+1/2}) \):

\[
\hat{f}_{j+1/2} = \hat{f}(u_{j+1/2}^- , u_{j+1/2}^+)
\]

where \( \hat{f}(u, u) = f(u) \) (consistency) and \( \hat{f} \) is Lipschitz continuous with respect to both arguments. In the scalar case \( \hat{f} \) is taken as a monotone flux, namely \( \hat{f}(↑, ↓) \) (monotonicity).

- Values from inside \( I_j \) for the test function \( v \)

\[
v_{j+1/2}^-, \quad v_{j-1/2}^+
\]
Hence the DG scheme is: find \( u \in V_h \) such that

\[
\int_{I_j} u_t v dx - \int_{I_j} f(u)v_x dx + \hat{f}_j^{+ \frac{1}{2}} v^{j+\frac{1}{2}} - \hat{f}_j^{- \frac{1}{2}} v^{j-\frac{1}{2}} = 0
\]

(2)

for all \( v \in V_h \).
Time discretization could be by the TVD Runge-Kutta method (Shu and Osher, JCP 1988). For the semi-discrete scheme:

\[ \frac{du}{dt} = L(u) \]

where \( L(u) \) is a discretization of the spatial operator, the third order TVD Runge-Kutta is simply:

\[
\begin{align*}
    u^{(1)} &= u^n + \Delta t L(u^n) \\
    u^{(2)} &= \frac{3}{4} u^n + \frac{1}{4} u^{(1)} + \frac{1}{4} \Delta t L(u^{(1)}) \\
    u^{n+1} &= \frac{1}{3} u^n + \frac{2}{3} u^{(2)} + \frac{2}{3} \Delta t L(u^{(2)})
\end{align*}
\]
Properties and advantages of the DG method:

- Easy handling of complicated geometry and boundary conditions.
- Compact. Communication only with immediate neighbors, regardless of the order of the scheme.
- Explicit. Because of the discontinuous basis, the mass matrix is local to the cell, resulting in explicit time stepping (no systems to solve).
Parallel efficiency. Achieves 99% parallel efficiency for static mesh and over 80% parallel efficiency for dynamic load balancing with adaptive meshes (Biswas, Devine and Flaherty, ANM 94; Remacle, Flaherty and Shephard, SIAM Rev 03; Beck, Bolemann, Flad, Frank, Gassner, Hindenlang and Munz, IJNMF 14; Atak, Beck, Bolemann, Flad, Frank, Hindenlang and Munz, High Performance Computing in Science and Engineering 14, Springer 15). Also friendly to GPU (Klockner et al, JCP10).
• Provable cell entropy inequality for the square entropy and the resulting $L^2$ stability, for arbitrary nonlinear equations in any spatial dimension and any triangulation, for any polynomial degrees, without limiters or assumption on solution regularity (Jiang and Shu, Math. Comp. 94 (scalar case); Hou and Liu, JSC 07 (symmetric systems)).

For $U(u) = \frac{u^2}{2}$:

$$
\frac{d}{dt} \int_{I_j} U(u) dx + \hat{F}_{j+1/2} - \hat{F}_{j-1/2} \leq 0
$$

Summing over $j$:

$$
\frac{d}{dt} \int_a^b u^2 dx \leq 0.
$$

This also holds for fully discrete RKDG methods with third order TVD Runge-Kutta time discretization, for linear equations (Zhang and Shu, SINUM 10).
• Bound preserving limiters, which can preserve strict maximum principle for scalar equations and positivity of relevant physical quantities (e.g. density and pressure for Euler systems for gas dynamics and water height for shallow water equations) while maintaining the original high order accuracy of the DG schemes, have been designed in a series of papers (Zhang and Shu, SINUM 10 (TVD); JCP 10 (scalar); JCP 11 (Euler), JCP 11b (source term); Proc Roy Soc A 11 (survey); Num Math 12 (entropy); Xing, Zhang and Shu, Adv Water Res 10 (shallow water); Zhang, Xia and Shu, JSC 12 (unstructured mesh); Wang et al, JCP 12 (detonations); Qin, Shu and Yang, JCP 16 (relativistic hydrodynamics); Vila, Shu and Maire, JCP 16, JCP 16b (Lagrangian multi-material flows); Yuan, Cheng and Shu, SISC 16 and Ling, Cheng and Shu, JSC to appear (radiative transfer)).
• Easy $h-p$ adaptivity.

• Stable and convergent DG methods are now available for many nonlinear PDEs containing higher derivatives: convection diffusion equations, KdV equations, ...
Two limitations of the Jiang-Shu cell entropy inequality for DG schemes:

1. It applies only to the square entropy $U(u) = \frac{1}{2} u^T u$. This is because the proof starts from taking the test function $v = u$ in the DG formulation. For general $U(u)$, one could not take the test function as $v = U'(u)$ since it is not in the finite element space;

2. It requires the integrals to be evaluated exactly.

The problem with the first limitation is that, for non-symmetric systems (such as Euler equations, which are symmetrizable but not symmetric), $U(u) = \frac{1}{2} u^T u$ is not an entropy, thus there is no cell entropy inequality available.
For symmetrizable systems, there is a one-to-one mapping \( w = U'(u)^T \) such that \( w \) satisfies a symmetric hyperbolic equation.

Of course, one could not use the DG method to solve the PDE for \( w \), as the weak solutions (e.g. shock speeds) for \( u \) and for \( w \) are different.

However, one could use the following DG scheme: find \( w \in V_h \) such that

\[
\int_{I_j} u(w)_t v \, dx - \int_{I_j} f(u(w)) v_x \, dx + \hat{f}_{j+\frac{1}{2}} v^-_{j+\frac{1}{2}} - \hat{f}_{j-\frac{1}{2}} v^+_{j-\frac{1}{2}} = 0 \tag{3}
\]

for all \( v \in V_h \). This approach was first used by Hughes, Franca and Mallet, CMAME 1986 for another scheme.
The advantage is that, now we can take the test function
\[ v = w = U'(u)^T \] as it is now in the finite element space \( V_h \), and hence cell entropy inequality results.

The disadvantage is that, now we have to solve a nonlinear system (because of the \( \int_{I_j} u(w)_t v dx \) term) for time evolution even if we use explicit time-stepping. Therefore, this approach is often used in space-time DG, for which a nonlinear system has to be solved anyway (Bath, LNCSE 1999; Hiltebrand and Mishra, Num Math 2014).

Note that exact evaluation of the integrals is still needed for the proof of the cell entropy inequality.
We introduce a unified framework to overcome the two difficulties simultaneously: We will design special quadrature rules to approximate the integrals in the DG scheme, so that conservation and high order accuracy are not affected (well, accuracy is sometimes decreased in numerical tests for up to one order), and cell entropy inequalities for any given single convex entropy are satisfied. Note: the quadrature rule is tailored towards the specific entropy, hence the scheme is designed to satisfy entropy condition for only one entropy, and schemes thus defined will be different for different entropy chosen by the user.
We start with the so-called “nodal DG” scheme:

- We choose the basis functions of $V_h$ as the Lagrangian polynomials of the $k + 1$ Legendre Gauss-Lobatto points, denoted by $L_i(x)$.

Note: this step does not alter the DG scheme, which is determined by the weak formulation and the space $V_h$, but not by the specific choice of basis functions of $V_h$. However, it does provide the convenience that the degrees of freedom to be evolved, namely the coefficients of the numerical solution when written as a linear combination of the basis functions, are the point values of the numerical solution at the Gauss-Lobatto points. This is sometimes called nodal DG methods.
• We replace both integrals \( \int_{I_j} u_t v dx \) (corresponding to the mass matrix) and \( \int_{I_j} f(u) v_x dx \) (corresponding to the stiffness matrix) by the Gauss-Lobatto quadrature rule.

Note: this step does change both of the integrals for nonlinear problems. The Gauss-Lobatto quadrature rule is only exact for polynomials of degree up to \( 2k - 1 \), hence the evaluation of the mass matrix \( \int_{I_j} L_i(x) L_\ell(x) dx \) is not exact. This is common to the spectral element methods. For linear PDEs, \( f(u) = au \) where \( a \) is a constant, the evaluation of the second integral \( \int_{I_j} f(u) v_x dx \) for the stiffness matrix is exact.

Unfortunately, the “nodal DG” scheme as defined above does not satisfy any entropy stability.
Notice that the Gauss-Lobatto quadrature rule for \( \int_{I_j} f(u) v_x dx \) can be written in a matrix-vector form as

\[
\sum_{l=0}^{k} D_{il} f(u_l)
\]

with a “derivative matrix” \( D_{il} \), where \( u_l \) denotes the point value of the numerical solution inside cell \( I_j \) at the \( l \)-th Gauss-Lobatto quadrature point. It is easy to verify that this can be rewritten as

\[
2 \sum_{l=0}^{k} D_{il} f_S(u_i, u_l)
\]

if we define

\[
f_S(u_i, u_l) = \frac{1}{2}(f(u_i) + f(u_l)). \tag{4}
\]
Now, our entropy condition modified DG scheme is just the “nodal DG” scheme above with \( f_S(u_i, u_l) \) in (4) replaced by any entropy conservative flux:

**Definition:** a consistent, symmetric two-point numerical flux \( f_S(u_L, u_R) \) is entropy conservative for a given entropy function \( U \) if

\[
(w_R - w_L)^T f_S(u_L, u_R) - (\psi_R - \psi_L) = 0
\]

where \( w = U'(u)^T \) is the entropy variable and \( \psi \) is the potential flux given by \( \psi_m(w) = (f_m(u(w)))^T w - F_m(u(w)) \).

The flux is called entropy stable if we drop the “symmetric” requirement and ask for \( \leq 0 \) in (5). The interface numerical fluxes in the DG scheme is required to be entropy stable.
**Theorem:** The modified nodal DG scheme defined above satisfies cell entropy inequalities. It is also at least $k$-th order accurate measured by truncation errors.

*Note:* The one-dimensional result is heavily based on previous work in the literature: Gassner, SISC 2013; Carpenter, Fisher, Nielsen and Frankel, SISC 2014.
For discussions of entropy conservative and entropy stable fluxes, see Tadmor, Math Comp 1987; ACTA Num 2003. For the scalar case, entropy conservative flux is uniquely determined once the entropy $\mathcal{U}(u)$ is given, and all monotone fluxes are entropy stable. For systems, entropy conservative flux is not unique. We use the one in Chandrashekar, CiCP 2013 in our numerical tests. Many existing numerical fluxes, such as the Godunov flux and the HLL (or LF) flux with suitable choice of maximum and minimum wave speeds, are entropy stable.

Entropy stable DG schemes provide some stability mechanism, but they are not non-oscillatory in the presence of strong shocks, and they may not be bound-preserving.
We can prove the bound-preserving limiter in Zhang and Shu, JCP 2010 does not increase entropy.

We can also prove that the minmod type TVD or TVB limiters do not increase entropy in the scalar case.
The design of entropy stable DG schemes can be generalized to two-dimensional triangular meshes. It requires a numerical quadrature on the triangle which satisfies the following requirements:

- It is symmetric so that adjacent elements can be glued together;
- The quadrature weights should be positive;
- It is exact for polynomials up to degree $2k - 1$;
- The quadrature points include $k + 1$ Legendre-Gauss points on each edge.
We remark that the bound-preserving limiter on arbitrary triangulations (Zhang, Xia and Shu, JSC 2012) requires the same kind of quadrature rules. We could use either the tensor product based rules (which has a large number of degrees of freedom), or the rules in Zhang, Cui and Liu, JCM 2009.
Figure 1: Distributions of quadrature points on $T$ with $k = 1$ (left, 6 points) and $k = 2$ (right, 10 points).
Figure 2: Distributions of quadrature points on $T$ with $k = 3$ (left, 18 points) and $k = 4$ (right, 22 points).
We have figured out a suitable extension of a modification of these “nodal DG schemes” so that they become entropy stable and remain high order accurate and conservative.
Generalization to convection-diffusion equations

The design of entropy stable DG schemes can be generalized to convection-diffusion equations:

\[
\frac{\partial u}{\partial t} + \sum_{j=1}^{2} \frac{\partial}{\partial x_j} (f_j(u) - \sum_{l=1}^{2} \hat{C}_{jl}(w) \frac{\partial w}{\partial x_l}) = 0
\]

where \( w = U'(u)^T \) is the entropy variable and

\[
\begin{bmatrix}
\hat{C}_{11}(w) & \hat{C}_{12}(w) \\
\hat{C}_{21}(w) & \hat{C}_{22}(w)
\end{bmatrix}
\]

is symmetric semi-positive-definite to ensure entropy dissipation.
The convective part is handled in the same way as before. For the diffusive part, we discretize it by a local DG (LDG) method (Cockburn and Shu, SINUM 1998). Entropy stability can be proved.
Example 1. We solve the one-dimensional linear advection equation

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

with periodic boundary condition and initial data \( u(x, 0) = \sin^4(x) \). The entropy function is the exponential function \( U = e^u \), and the entropy conservative flux is given by

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

We observe optimal convergence for all values of \( k \) for this example.
Table 1: Accuracy test of the one-dimensional linear advection equation associated with initial data \( u(x, 0) = \sin^4(x) \) and exponential entropy function at \( t = 2\pi \).

<table>
<thead>
<tr>
<th>( k )</th>
<th>( N )</th>
<th>( L^1 ) error</th>
<th>order</th>
<th>( L^2 ) error</th>
<th>order</th>
<th>( L^\infty ) error</th>
<th>order</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>20</td>
<td>7.030e-2</td>
<td>-</td>
<td>3.347e-2</td>
<td>-</td>
<td>2.688e-2</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3.097e-3</td>
<td>-</td>
<td>1.514e-3</td>
<td>-</td>
<td>1.890e-3</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2.608e-2</td>
<td>-</td>
<td>1.178e-2</td>
<td>-</td>
<td>8.580e-3</td>
<td>-</td>
</tr>
<tr>
<td>40</td>
<td>5.363e-3</td>
<td>3.712</td>
<td></td>
<td>2.669e-3</td>
<td>3.649</td>
<td>2.340e-3</td>
<td>3.522</td>
</tr>
<tr>
<td>80</td>
<td>4.575e-4</td>
<td>3.551</td>
<td></td>
<td>2.205e-4</td>
<td>3.598</td>
<td>1.846e-4</td>
<td>3.664</td>
</tr>
<tr>
<td>160</td>
<td>4.414e-5</td>
<td>3.374</td>
<td></td>
<td>2.230e-5</td>
<td>3.305</td>
<td>2.582e-5</td>
<td>2.838</td>
</tr>
<tr>
<td>320</td>
<td>4.745e-6</td>
<td>3.218</td>
<td></td>
<td>2.590e-6</td>
<td>3.103</td>
<td>3.626e-6</td>
<td>2.832</td>
</tr>
<tr>
<td>640</td>
<td>5.485e-7</td>
<td>3.113</td>
<td></td>
<td>3.181e-7</td>
<td>3.028</td>
<td>4.794e-7</td>
<td>2.919</td>
</tr>
<tr>
<td>3</td>
<td>20</td>
<td>1.675e-4</td>
<td>4.208</td>
<td></td>
<td>8.672e-5</td>
<td>4.126</td>
<td>1.359e-4</td>
</tr>
<tr>
<td>40</td>
<td>1.053e-5</td>
<td>3.993</td>
<td></td>
<td>5.372e-6</td>
<td>4.013</td>
<td>8.928e-6</td>
<td>3.928</td>
</tr>
<tr>
<td>80</td>
<td>6.571e-7</td>
<td>4.002</td>
<td></td>
<td>3.354e-7</td>
<td>4.001</td>
<td>5.664e-7</td>
<td>3.978</td>
</tr>
<tr>
<td>160</td>
<td>4.107e-8</td>
<td>4.000</td>
<td></td>
<td>2.096e-8</td>
<td>4.000</td>
<td>3.553e-8</td>
<td>3.995</td>
</tr>
<tr>
<td>320</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>10</td>
<td>2.623e-5</td>
<td>4.988</td>
<td></td>
<td>1.179e-5</td>
<td>4.997</td>
<td>9.860e-6</td>
</tr>
<tr>
<td>20</td>
<td>8.170e-7</td>
<td>5.004</td>
<td></td>
<td>3.683e-7</td>
<td>5.000</td>
<td>3.084e-7</td>
<td>4.999</td>
</tr>
<tr>
<td>40</td>
<td>2.553e-8</td>
<td>5.000</td>
<td></td>
<td>1.151e-8</td>
<td>5.000</td>
<td>9.454e-9</td>
<td>5.028</td>
</tr>
</tbody>
</table>
Example 2. Next we consider the one-dimensional Burgers equation

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

with periodic boundary condition and initial data $u(x, 0) = 0.5 + \sin x$. We choose square entropy function $U = u^2/2$ and present the errors at $t = 0.5$ when the solution is still smooth.

It is evident that the convergence rate is slightly below optimal, especially for the $L^\infty$ error. However, when $k = 3$ we still have optimal convergence.
Table 2: Accuracy test of the one-dimensional Burgers equation associated with initial data $u(x, 0) = 0.5 + \sin x$ and square entropy function at $t = 0.5$.

<table>
<thead>
<tr>
<th>$k$</th>
<th>$N$</th>
<th>$L^1$ error</th>
<th>order</th>
<th>$L^2$ error</th>
<th>order</th>
<th>$L^\infty$ error</th>
<th>order</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>40</td>
<td>1.320e-3</td>
<td>-</td>
<td>1.178e-3</td>
<td>-</td>
<td>3.269e-3</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>80</td>
<td>2.071e-4</td>
<td>2.672</td>
<td>2.284e-4</td>
<td>2.366</td>
<td>7.923e-4</td>
<td>2.045</td>
</tr>
<tr>
<td></td>
<td>160</td>
<td>3.162e-5</td>
<td>2.711</td>
<td>4.316e-5</td>
<td>2.404</td>
<td>2.078e-4</td>
<td>1.931</td>
</tr>
<tr>
<td></td>
<td>320</td>
<td>4.724e-6</td>
<td>2.743</td>
<td>7.979e-6</td>
<td>2.435</td>
<td>5.100e-5</td>
<td>2.026</td>
</tr>
<tr>
<td></td>
<td>640</td>
<td>6.911e-7</td>
<td>2.773</td>
<td>1.450e-6</td>
<td>2.460</td>
<td>1.290e-5</td>
<td>1.983</td>
</tr>
<tr>
<td></td>
<td>1280</td>
<td>9.930e-8</td>
<td>2.799</td>
<td>2.606e-7</td>
<td>2.477</td>
<td>3.209e-6</td>
<td>2.008</td>
</tr>
<tr>
<td>3</td>
<td>40</td>
<td>4.344e-5</td>
<td>-</td>
<td>4.566e-5</td>
<td>-</td>
<td>1.658e-4</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>80</td>
<td>3.348e-6</td>
<td>3.698</td>
<td>3.703e-6</td>
<td>3.624</td>
<td>1.610e-5</td>
<td>3.364</td>
</tr>
<tr>
<td></td>
<td>160</td>
<td>2.344e-7</td>
<td>3.836</td>
<td>2.771e-7</td>
<td>3.740</td>
<td>1.306e-6</td>
<td>3.624</td>
</tr>
<tr>
<td>4</td>
<td>20</td>
<td>6.782e-5</td>
<td>-</td>
<td>6.319e-5</td>
<td>-</td>
<td>1.525e-4</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>40</td>
<td>2.630e-6</td>
<td>4.688</td>
<td>2.849e-6</td>
<td>4.471</td>
<td>1.126e-5</td>
<td>3.760</td>
</tr>
<tr>
<td></td>
<td>80</td>
<td>1.067e-7</td>
<td>4.624</td>
<td>1.374e-7</td>
<td>4.375</td>
<td>7.149e-7</td>
<td>3.977</td>
</tr>
<tr>
<td></td>
<td>320</td>
<td>1.576e-10</td>
<td>4.737</td>
<td>2.858e-10</td>
<td>4.481</td>
<td>2.620e-9</td>
<td>4.050</td>
</tr>
</tbody>
</table>
Example 3. We solve the two-dimensional linear advection equation

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

with periodic boundary condition and initial data

\[u(\mathbf{x}, 0) = \sin(2\pi x_1) \sin(2\pi x_2),\]

and the square entropy function \(U = u^2/2\). We test the DG scheme on a hierarchy of unstructured triangular meshes. Errors and orders of convergence at \(t = 0.2\) are shown in Table 3. Once again we obtain optimal convergence.
Table 3: Accuracy test of the two-dimensional linear advection equation associated with initial data $u(x, 0) = \sin(2\pi x_1) \sin(2\pi x_2)$ and square entropy function at $t = 0.2$.

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

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

with periodic boundary condition and initial data

\[ u(\mathbf{x}, 0) = 0.5 \sin(2\pi(x_1 + x_2)), \]

and square entropy function

\[ U = \frac{u^2}{2}. \]

The entropy stable nodal DG scheme is evolved up to \( t = 0.05 \) when the solution is still smooth. Errors and orders of convergence are displayed in Table 4. The results are similar to its one-dimensional counterpart. Convergence rate is slightly below optimal.
Table 4: Accuracy test of the two-dimensional Burgers equation associated with initial data \( u(x, 0) = 0.5 \sin(2\pi(x_1 + x_2)) \) and square entropy function at \( t = 0.05 \).

<table>
<thead>
<tr>
<th>k</th>
<th>h</th>
<th>( L^1 ) error</th>
<th>order</th>
<th>( L^2 ) error</th>
<th>order</th>
<th>( L^\infty ) error</th>
<th>order</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>1/16</td>
<td>1.354e-3</td>
<td>-</td>
<td>3.275e-3</td>
<td>-</td>
<td>5.954e-2</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>1/32</td>
<td>2.394e-4</td>
<td>2.500</td>
<td>7.046e-4</td>
<td>2.217</td>
<td>1.646e-2</td>
<td>1.855</td>
</tr>
<tr>
<td></td>
<td>1/64</td>
<td>3.900e-5</td>
<td>2.618</td>
<td>1.406e-4</td>
<td>2.325</td>
<td>4.894e-3</td>
<td>1.750</td>
</tr>
<tr>
<td></td>
<td>1/128</td>
<td>5.773e-6</td>
<td>2.756</td>
<td>2.456e-5</td>
<td>2.518</td>
<td>1.269e-3</td>
<td>1.948</td>
</tr>
<tr>
<td></td>
<td>1/256</td>
<td>8.431e-7</td>
<td>2.776</td>
<td>4.109e-6</td>
<td>2.579</td>
<td>2.413e-4</td>
<td>2.394</td>
</tr>
<tr>
<td>3</td>
<td>1/16</td>
<td>1.890e-4</td>
<td>-</td>
<td>6.252e-4</td>
<td>-</td>
<td>1.968e-2</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>1/32</td>
<td>2.482e-5</td>
<td>2.929</td>
<td>1.058e-4</td>
<td>2.563</td>
<td>4.859e-3</td>
<td>2.018</td>
</tr>
<tr>
<td></td>
<td>1/64</td>
<td>2.327e-6</td>
<td>3.415</td>
<td>1.106e-5</td>
<td>3.258</td>
<td>7.311e-4</td>
<td>2.733</td>
</tr>
<tr>
<td></td>
<td>1/128</td>
<td>2.065e-7</td>
<td>3.494</td>
<td>1.158e-6</td>
<td>3.255</td>
<td>1.195e-4</td>
<td>2.613</td>
</tr>
<tr>
<td>4</td>
<td>1/16</td>
<td>3.740e-5</td>
<td>-</td>
<td>1.454e-4</td>
<td>-</td>
<td>6.039e-3</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>1/32</td>
<td>2.787e-6</td>
<td>3.746</td>
<td>1.427e-5</td>
<td>3.349</td>
<td>1.068e-3</td>
<td>2.500</td>
</tr>
<tr>
<td></td>
<td>1/64</td>
<td>1.348e-7</td>
<td>4.370</td>
<td>7.651e-7</td>
<td>4.221</td>
<td>8.839e-5</td>
<td>3.595</td>
</tr>
<tr>
<td></td>
<td>1/256</td>
<td>2.293e-10</td>
<td>4.602</td>
<td>1.696e-9</td>
<td>4.456</td>
<td>3.059e-7</td>
<td>4.387</td>
</tr>
</tbody>
</table>
Example 5. The last smooth test case is the isentropic vortex advection problem for the two-dimensional Euler equations. The computational domains is $[0, 10]^2$ and the initial condition is given by

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

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

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

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

The parameters are $\varepsilon = \frac{5}{2\pi}, \alpha = 0.5$ and $(y_1, y_2) = (5, 5)$. Table 5 summarizes errors and orders of convergence of the density at $t = 1$. Here we get nearly optimal convergence.
Table 5: Accuracy test of isentropic vortex problem for two-dimensional Euler equations at $t = 1$. Results of the density are tabulated.

<table>
<thead>
<tr>
<th>$k$</th>
<th>$h$</th>
<th>$L^1$ error</th>
<th>order</th>
<th>$L^2$ error</th>
<th>order</th>
<th>$L^\infty$ error</th>
<th>order</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>10/8</td>
<td>2.299e-1</td>
<td></td>
<td>6.053e-2</td>
<td></td>
<td>8.735e-2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>10/16</td>
<td>4.204e-2</td>
<td>2.451</td>
<td>1.223e-2</td>
<td>2.307</td>
<td>2.957e-2</td>
<td>1.563</td>
</tr>
<tr>
<td></td>
<td>10/32</td>
<td>6.598e-3</td>
<td>2.671</td>
<td>1.918e-3</td>
<td>2.673</td>
<td>5.162e-3</td>
<td>2.518</td>
</tr>
<tr>
<td></td>
<td>10/64</td>
<td>9.330e-4</td>
<td>2.822</td>
<td>2.688e-4</td>
<td>2.835</td>
<td>1.064e-3</td>
<td>2.279</td>
</tr>
<tr>
<td></td>
<td>10/128</td>
<td>1.273e-4</td>
<td>2.873</td>
<td>3.609e-5</td>
<td>2.897</td>
<td>1.717e-4</td>
<td>2.631</td>
</tr>
<tr>
<td></td>
<td>10/256</td>
<td>1.652e-5</td>
<td>2.947</td>
<td>4.779e-6</td>
<td>2.917</td>
<td>2.280e-5</td>
<td>2.913</td>
</tr>
<tr>
<td>3</td>
<td>10/8</td>
<td>4.332e-2</td>
<td></td>
<td>1.144e-2</td>
<td></td>
<td>2.555e-2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>10/16</td>
<td>3.976e-3</td>
<td>3.446</td>
<td>1.156e-3</td>
<td>3.307</td>
<td>4.271e-3</td>
<td>2.581</td>
</tr>
<tr>
<td></td>
<td>10/32</td>
<td>3.632e-4</td>
<td>3.453</td>
<td>1.030e-4</td>
<td>3.487</td>
<td>2.652e-4</td>
<td>4.009</td>
</tr>
<tr>
<td></td>
<td>10/64</td>
<td>3.041e-5</td>
<td>3.578</td>
<td>8.538e-6</td>
<td>3.593</td>
<td>4.557e-5</td>
<td>2.541</td>
</tr>
<tr>
<td></td>
<td>10/128</td>
<td>2.536e-6</td>
<td>3.584</td>
<td>7.148e-7</td>
<td>3.578</td>
<td>3.793e-6</td>
<td>3.587</td>
</tr>
<tr>
<td>4</td>
<td>10/8</td>
<td>7.754e-3</td>
<td></td>
<td>2.136e-3</td>
<td></td>
<td>8.836e-3</td>
<td></td>
</tr>
<tr>
<td></td>
<td>10/16</td>
<td>3.941e-4</td>
<td>4.298</td>
<td>1.308e-4</td>
<td>4.030</td>
<td>5.582e-4</td>
<td>3.985</td>
</tr>
<tr>
<td></td>
<td>10/32</td>
<td>1.546e-5</td>
<td>4.672</td>
<td>4.858e-6</td>
<td>4.750</td>
<td>2.549e-5</td>
<td>4.452</td>
</tr>
<tr>
<td></td>
<td>10/64</td>
<td>5.620e-7</td>
<td>4.782</td>
<td>1.806e-7</td>
<td>4.749</td>
<td>1.680e-6</td>
<td>3.923</td>
</tr>
</tbody>
</table>
For the following discontinuous tests we take $k = 2$.

**Example 6.** We consider the following Riemann problem of the Buckley-Leverett equation

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

The exact entropy solution contains two shock waves connected by a flat rarefaction wave that is close to 0. For such a nonconvex flux function, the choice of entropy function will affect the performance of numerical scheme substantially.
We first test the scheme with square entropy function $U = u^2/2$. The computational domain is $[-0.5, 0.5]$ and the end time $t = 1$. We also apply the bound-preserving limiter with $\Omega = [-3, 3]$. The numerical solution on 80 cells is plotted in the left panel of Figure 3. Then we try an ad hoc entropy function $U = \int \arctan(20u) \, du$. The entropy variable $v = \arctan(20u)$, which emphasizes the states near $u = 0$. In fact it can be viewed as a mollified version of the Kruzhkov’s entropy function $U = |u|$. The numerical solution with the same setting is depicted in the right panel of Figure 3. The result is quite satisfactory thanks to the carefully chosen entropy function.
Figure 3: Numerical solution of the Riemann problem of Buckley-Leverett equation at $t = 1$ with the square entropy function and an ad hoc entropy function. Computational domain $[-0.5, 0.5]$ is decomposed into $N = 80$ cells. Bound-preserving limiter is used. The solid line represents the exact entropy solution and the triangle symbols are cell averages.
Example 7. Sod’s shock tube. It is a classical Riemann problem of one-dimensional Euler equations. The computational domain is $[-0.5, 0.5]$ and the initial condition is

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

The classical DG scheme tends to blow up due to emergence of negative density or negative pressure unless we apply positivity-preserving limiter or TVD/TVB limiter. The entropy stable nodal DG scheme, on the other hand, can be evolved without any limiter. Figure 4 illustrates the profiles of density, velocity and pressure at $t = 0.13$ with 130 cells. Entropy stability contributes to a more robust scheme for this test problem.
Figure 4: Numerical solution of Sod’s shock tube problem at $t = 0.13$ with 130 cells. We do not apply any limiter. The solid line represents the exact entropy solution and the triangle symbols are cell averages. Left: density; Right: velocity.
Figure 5: (Continued): Numerical solution of Sod’s shock tube problem at $t = 0.13$ with 130 cells. We do not apply any limiter. The solid line represents the exact entropy solution and the triangle symbols are cell averages. Pressure.
Example 8. Sine-shock interaction (Shu-Osher problem). The solution has complicated structure in that it contains both strong and weak shock waves and highly oscillatory smooth waves. The computational domain is \([-5, 5]\) and the initial condition is

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

Once again the classical DG scheme suffers from negative pressure or negative density, while the entropy stable nodal DG scheme works without any limiter. The plots of density, velocity and pressure at \(t = 1.8\) with 150 cells are displayed in Figure 6. The scheme performs well despite some minor oscillations.
Figure 6: Numerical solution of sine-shock interaction test problem at $t = 1.8$ with 150 cells. We do not apply any limiter. The solid line represents the reference solution and the triangle symbols are cell averages. Left: density; Right: velocity.
Figure 7: (Continued): Numerical solution of sine-shock interaction test problem at $t = 1.8$ with 150 cells. We do not apply any limiter. The solid line represents the reference solution and the triangle symbols are cell averages. Pressure.
Example 9. We solve the Riemann problem of the two-dimensional Burgers equation

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

subject to the initial condition

\[ u(x, 0) = \begin{cases} 0.25 & \text{if } x_1 < 0.5 \text{ and } x_2 < 0.5 \\ -0.1 & \text{if } x_1 < 0.5 \text{ and } x_2 \geq 0.5 \\ 0.4 & \text{if } x_1 \geq 0.5 \text{ and } x_2 < 0.5 \\ -0.5 & \text{if } x_1 \geq 0.5 \text{ and } x_2 \geq 0.5 \end{cases} \]
We choose the square entropy function $U = u^2/2$ and run the entropy stable nodal DG scheme up to $t = 0.5$ on a triangular mesh with $h = 1/128$. The bound-preserving limiter with $\Omega = [-0.5, 0.4]$ is also imposed. The numerical result is shown in the left panel of Figure 8, the absolute value error is also plotted in the right panel where we use logarithmic scale and values less than $10^{-16}$ are transformed to $10^{-16}$. The scheme successfully captures the correct profile. Error is very small unless near shock waves.
Figure 8: Numerical solution (left) and error (right) of a Riemann problem of two-dimensional Burgers equation at $t = 0.5$ on a mesh with $h = 1/128$. Entropy function is $U = u^2/2$ and bound-preserving limiter is used.
**Example 10.** Double Mach reflection. It involves a Mach 10 shock which makes a 60-degree angle with a reflecting wall. We illustrate the computational domain and the unstructured mesh with $h = 1/20$ in Figure 9. Initially the shock is positioned at $x_1 = 0$. Inflow/outflow boundary conditions are prescribed for the left and right boundaries, and at the top boundary the flow values are set to describe the exact motion of shock.
Figure 9: Illustration of the computational domain and the unstructured mesh with $h = 1/20$. 

Division of Applied Mathematics, Brown University
The entropy stable nodal DG scheme is implemented with positivity-preserving limiter and the local Lax-Friedrichs flux. The plots of density and pressure at $t = 0.2$ with mesh size $h = 1/240$ are given in Figure 10.

Figure 10: Profiles of density (left) and pressure (right) at $t = 0.2$ on a mesh with $h = 1/240$. 40 equally spaced contour levels are used for both plots.
Example 11. A shock wave diffracting at a sharp corner is another popular test problem for two-dimensional Euler equations. We study a Mach 10 shock diffracting at a 120 degree. The computational domain and the triangular mesh with $h = 1/4$ are demonstrated in Figure 11. The shock is initially located at $x_1 = 3.4$ and $6 \leq x_2 \leq 11$, moving into undisturbed air with a density of 1.4 and a pressure of 1. Boundary conditions are inflow at the left/top boundary (in accordance with the exact shock motion), and outflow at the right/bottom boundary.
Figure 11: Illustration of the computational domain and the unstructured mesh with $h = 1/4$. 
We still use positivity-preserving limiter and the local Lax-Friedrichs interface flux. The contour plots of density and pressure at $t = 0.9$ with mesh size $h = 1/40$ are depicted in Figure 12.

Figure 12: Profiles of density (left) and pressure (right) at $t = 0.9$ on a mesh with $h = 1/20$. 40 equally spaced contour levels are used for both plots.
Concluding remarks

- We have constructed a (formally) high order entropy stable DG scheme for systems of conservation laws.
- The method does not require exact integration and can be designed to be stable with respect to an arbitrary entropy function.
Our scheme has good flexibility in that it is compatible with

1. bound-preserving limiter and (one-dimensional scalar) TVD/TVB limiter.
2. unstructured triangular meshes, and potentially simplex meshes and polygonal meshes.
3. convection-diffusion equations through an LDG type approach.
4. reflecting wall boundary conditions of Euler equations.
• This result has also been extended to ideal compressible MHD, using the symmetrizable version of the equations as introduced by Godunov. The additional non-conservative terms in this version causes additional difficulties.

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


The End

THANK YOU!