Entropy stable high order discontinuous Galerkin methods for nonlinear conservation laws

Jesse Chan

University of Colorado Boulder

Ann and HJ Smead Aerospace Engineering Sciences

November 18, 2020

[1]Department of Computational and Applied Mathematics

Collaborators

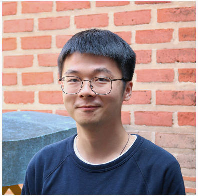
Mark Carpenter (NASA Langley)

DCDR Fernandez (NASA Langley)

Tim Warburton (VT)

Philip Wu (GPU + shallow water)

Christina Taylor (ROMs + implicit)

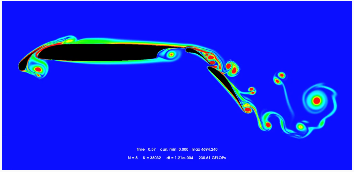


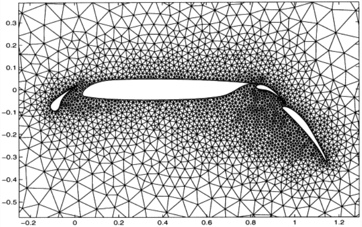
Yimin Lin (GPU + compressible flow)

Mario Bencomo (AMR, adjoints)

High order finite element methods for hyperbolic PDEs

- Aerodynamics applications: acoustics, vorticular flows, turbulence, shocks.
- Goal: high accuracy on unstructured meshes.
- Discontinuous Galerkin (DG) methods: geometric flexibility, high order accuracy.

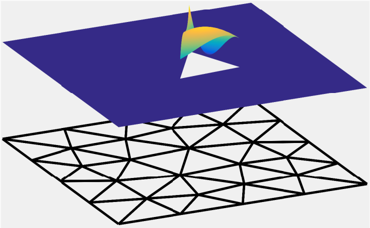




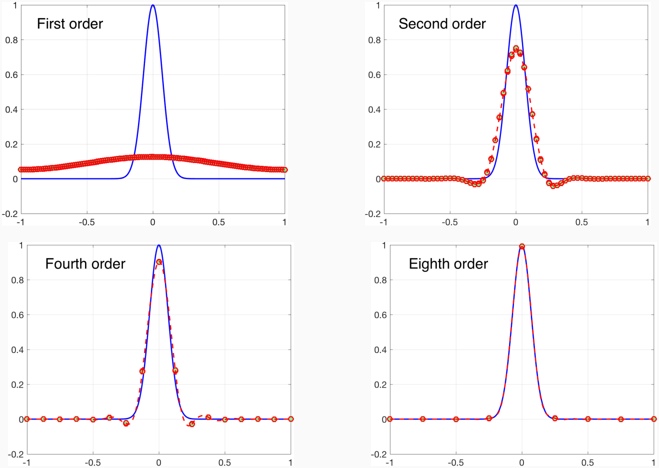
Mesh from Slawig 2001.

High order finite element methods for hyperbolic PDEs

- Aerodynamics applications: acoustics, vorticular flows, turbulence, shocks.
- Goal: high accuracy on unstructured meshes.
- Discontinuous Galerkin (DG) methods: geometric flexibility, high order accuracy.

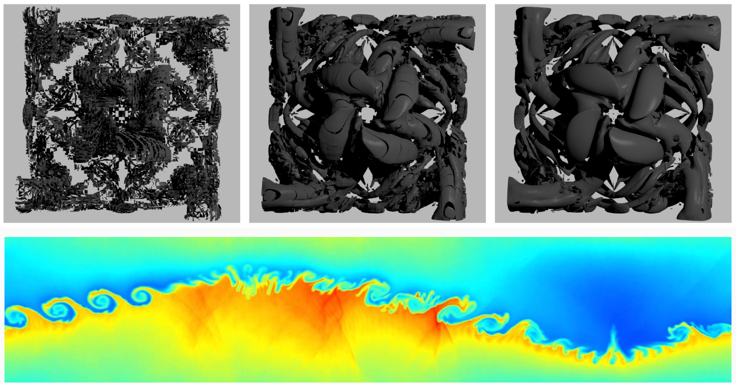


Why high order accuracy?



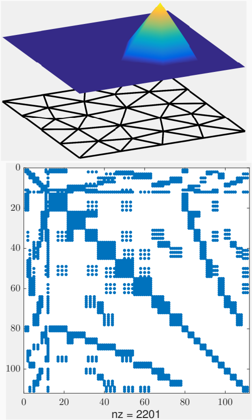
High order accurate resolution of propagating vortices and waves.

Why high order accuracy?

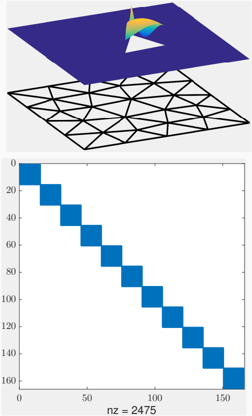


2nd, 4th, and 16th order Taylor-Green (top), 8th order Kelvin-Helmholtz (bottom). Vorticular structures and acoustic waves are both sensitive to numerical dissipation. Results from Beck and Gassner (2013) and Per-Olof Persson's website.

Why discontinuous Galerkin methods?



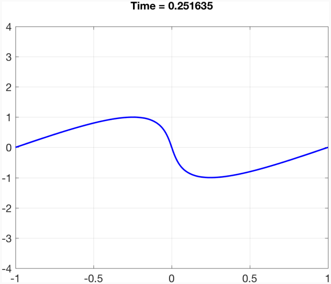
(a) High order FEM



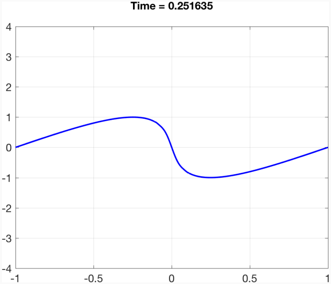
(b) High order DG

The DG mass matrix is easily invertible for explicit time-stepping.

Why *not* high order DG methods?



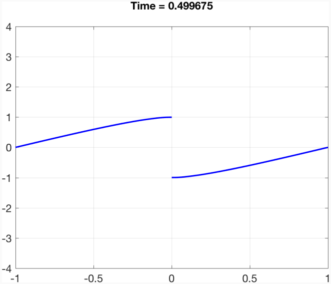
(a) Exact solution



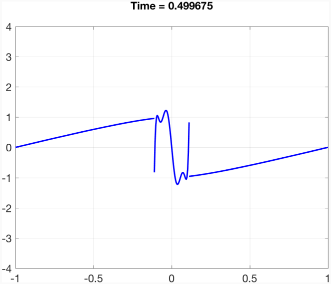
(b) 8th order DG

High order methods blow up for under-resolved solutions of nonlinear conservation laws (e.g., shocks and turbulence).

Why *not* high order DG methods?



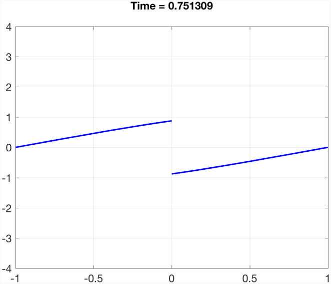
(a) Exact solution



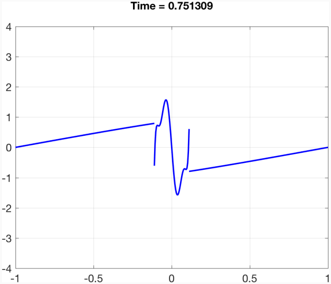
(b) 8th order DG

High order methods blow up for under-resolved solutions of nonlinear conservation laws (e.g., shocks and turbulence).

Why *not* high order DG methods?



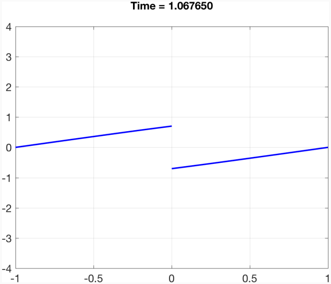
(a) Exact solution



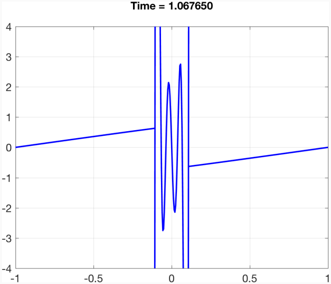
(b) 8th order DG

High order methods blow up for under-resolved solutions of nonlinear conservation laws (e.g., shocks and turbulence).

Why *not* high order DG methods?



(a) Exact solution



(b) 8th order DG

High order methods blow up for under-resolved solutions of nonlinear conservation laws (e.g., shocks and turbulence).

Why entropy stability for high order schemes?

In practice, high order schemes need solution regularization (e.g., artificial viscosity, filtering, slope limiting).

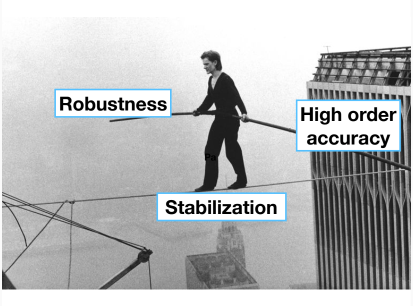
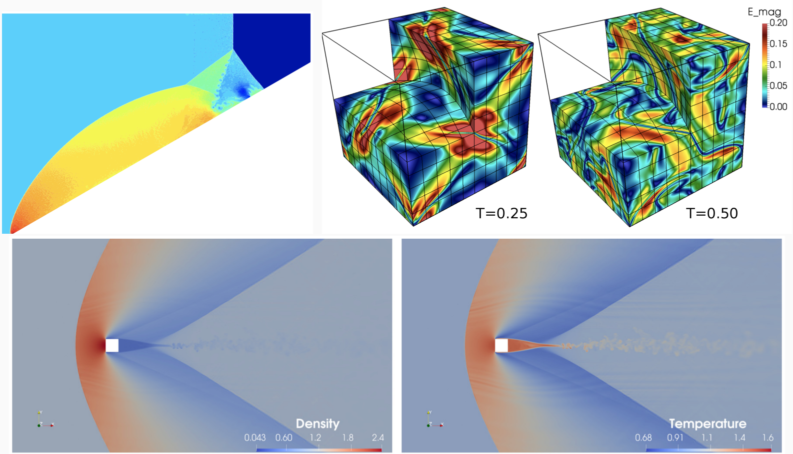


Image adapted from "Man On Wire" (2008)

- Goal: stability independent of solution regularization.
- Entropy stable schemes: improve robustness without reducing accuracy.

Finite volume methods: Tadmor, Chandrashekar, Ray, Svard, Fjordholm, Mishra, LeFloch, Rohde, . . .
High order tensor product elements: Fisher, Carpenter, Gassner, Winters, Kopriva, Persson, . . .
High order general elements: Chen and Shu, Crean, Hicken, Del Rey Fernandez, Zingg, . . .

Examples of high order entropy stable simulations



All simulations run without artificial viscosity, filtering, or slope limiters.

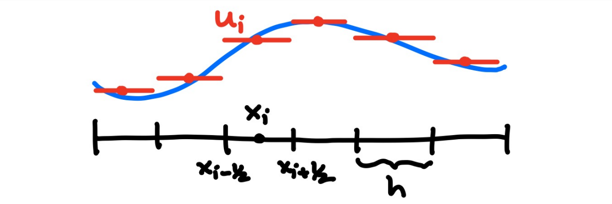
Chen, Shu (2017). *Entropy stable high order DG methods with suitable quadrature rules...*

Bohm et al. (2019). *An entropy stable nodal DG method for the resistive MHD equations. Part I.*

Dalcin et al. (2019). *Conservative and ES solid wall BCs for the compressible NS equations.*

Entropy conservative/stable finite volume methods

Basics of finite volume methods



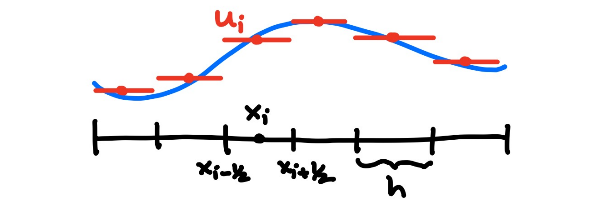
- Solve for $\mathbf{u}_i = \frac{1}{h}\int_{x_{i-1/2}}^{x_{i+1/2}} \boldsymbol{u}(x,t)\,\mathrm{d}x$.

$$\int_{x_{i-1/2}}^{x_{i+1/2}} \frac{\partial \boldsymbol{u}}{\partial t} + \frac{\partial \boldsymbol{f}(\boldsymbol{u})}{\partial x} = \mathbf{0}$$

- Replace $\boldsymbol{f}(\boldsymbol{u}(x_{i\pm 1/2}, t))$ with a *numerical flux*

$$\frac{\mathrm{d}\mathbf{u}_i}{\mathrm{d}t} + \frac{\boldsymbol{f}_S(\mathbf{u}_{i+1}, \mathbf{u}_i) - \boldsymbol{f}_S(\mathbf{u}_i, \mathbf{u}_{i-1})}{h} = \mathbf{0}$$

Basics of finite volume methods



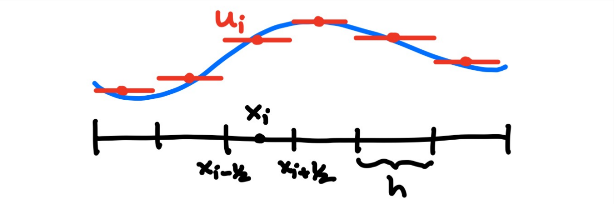
- Solve for $\mathbf{u}_i = \frac{1}{h}\int_{x_{i-1/2}}^{x_{i+1/2}} \boldsymbol{u}(x,t)\,\mathrm{d}x$.

$$\frac{\partial}{\partial t}\int_{x_{i-1/2}}^{x_{i+1/2}} \boldsymbol{u} + \boldsymbol{f}(\boldsymbol{u}(x_{i+1/2},t)) - \boldsymbol{f}(\boldsymbol{u}(x_{i-1/2},t)) = \mathbf{0}$$

- Replace $\boldsymbol{f}(\boldsymbol{u}(x_{i\pm1/2},t))$ with a *numerical flux*

$$\frac{\mathrm{d}\mathbf{u}_i}{\mathrm{d}t} + \frac{\boldsymbol{f}_S(\mathbf{u}_{i+1},\mathbf{u}_i) - \boldsymbol{f}_S(\mathbf{u}_i,\mathbf{u}_{i-1})}{h} = \mathbf{0}$$

Basics of finite volume methods



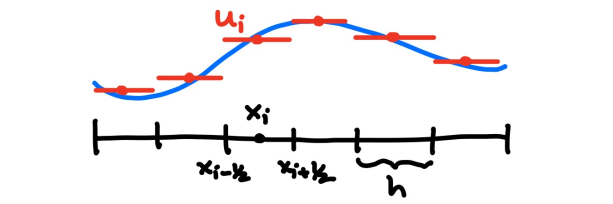
- Solve for $\mathbf{u}_i = \frac{1}{h}\int_{x_{i-1/2}}^{x_{i+1/2}} \boldsymbol{u}(x,t)\,\mathrm{d}x$.

$$\frac{\mathrm{d}\mathbf{u}_i}{\mathrm{dt}} + \frac{\boldsymbol{f}(\boldsymbol{u}(x_{i+1/2},t)) - \boldsymbol{f}(\boldsymbol{u}(x_{i-1/2},t))}{h} = \mathbf{0}$$

- Replace $\boldsymbol{f}(\boldsymbol{u}(x_{i\pm 1/2},t))$ with a *numerical flux*

$$\frac{\mathrm{d}\mathbf{u}_i}{\mathrm{dt}} + \frac{\boldsymbol{f}_S(\mathbf{u}_{i+1},\mathbf{u}_i) - \boldsymbol{f}_S(\mathbf{u}_i,\mathbf{u}_{i-1})}{h} = \mathbf{0}$$

Basics of finite volume methods



- Solve for $\mathbf{u}_i = \frac{1}{h}\int_{x_{i-1/2}}^{x_{i+1/2}} \boldsymbol{u}(x,t)\,\mathrm{d}x$.

$$\frac{\mathrm{d}\mathbf{u}_i}{\mathrm{d}t} + \frac{\boldsymbol{f}(\boldsymbol{u}(x_{i+1/2},t)) - \boldsymbol{f}(\boldsymbol{u}(x_{i-1/2},t))}{h} = \mathbf{0}$$

- Replace $\boldsymbol{f}(\boldsymbol{u}(x_{i\pm 1/2},t))$ with a *numerical flux*

$$\frac{\mathrm{d}\mathbf{u}_i}{\mathrm{d}t} + \frac{\boldsymbol{f}_S(\mathbf{u}_{i+1},\mathbf{u}_i) - \boldsymbol{f}_S(\mathbf{u}_i,\mathbf{u}_{i-1})}{h} = \mathbf{0}$$

Entropy stability for nonlinear problems

- Energy balance for nonlinear conservation laws (Burgers', shallow water, compressible Euler + Navier-Stokes).

$$\frac{\partial \boldsymbol{u}}{\partial t} + \frac{\partial \boldsymbol{f}(\boldsymbol{u})}{\partial x} = 0.$$

- Continuous entropy inequality: convex entropy function $S(\boldsymbol{u})$, "entropy potential" $\psi(\boldsymbol{u})$, entropy variables $\boldsymbol{v}(\boldsymbol{u})$

$$\int_\Omega \boldsymbol{v}^T \left(\frac{\partial \boldsymbol{u}}{\partial t} + \frac{\partial \boldsymbol{f}(\boldsymbol{u})}{\partial x} \right) = 0, \qquad \boxed{\boldsymbol{v}(\boldsymbol{u}) = \frac{\partial S}{\partial \boldsymbol{u}}}$$

$$\Longrightarrow \int_\Omega \frac{\partial S(\boldsymbol{u})}{\partial t} + \left(\boldsymbol{v}^T \boldsymbol{f}(\boldsymbol{u}) - \psi(\boldsymbol{u}) \right) \Big|_{-1}^{1} \leq 0.$$

Entropy conservative finite volume methods

- Finite volume scheme:

$$\frac{\mathrm{d}\mathbf{u}_i}{\mathrm{dt}} + \frac{\boldsymbol{f}_S(\mathbf{u}_{i+1}, \mathbf{u}_i) - \boldsymbol{f}_S(\mathbf{u}_{i+1}, \mathbf{u}_i)}{h} = \mathbf{0}.$$

- Take $\boldsymbol{f}_S$ to be an entropy conservative numerical flux

$$\boldsymbol{f}_S(\boldsymbol{u}, \boldsymbol{u}) = \boldsymbol{f}(\boldsymbol{u}), \qquad \text{(consistency)}$$

$$\boldsymbol{f}_S(\boldsymbol{u}, \boldsymbol{v}) = \boldsymbol{f}_S(\boldsymbol{v}, \boldsymbol{u}), \qquad \text{(symmetry)}$$

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

- Can show numerical scheme conserves entropy

$$\int_\Omega \frac{\partial S(\boldsymbol{u})}{\partial t} \approx \sum_i h \frac{\mathrm{d}S(\mathbf{u}_i)}{\mathrm{dt}} = 0.$$

Tadmor (1987). *The numerical viscosity of entropy stable schemes for systems of conservation laws.*

Entropy stable finite volume methods

- Finite volume scheme with dissipation $\mathbf{d}(\mathbf{u})$:

$$\frac{\mathrm{d}\mathbf{u}_i}{\mathrm{dt}} + \frac{\boldsymbol{f}_S(\mathbf{u}_{i+1}, \mathbf{u}_i) - \boldsymbol{f}_S(\mathbf{u}_{i+1}, \mathbf{u}_i)}{h} = \mathbf{d}(\mathbf{u}).$$

- Take $\boldsymbol{f}_S$ to be an entropy conservative numerical flux

$$\begin{aligned} \boldsymbol{f}_S(\boldsymbol{u}, \boldsymbol{u}) &= \boldsymbol{f}(\boldsymbol{u}), && \text{(consistency)} \\ \boldsymbol{f}_S(\boldsymbol{u}, \boldsymbol{v}) &= \boldsymbol{f}_S(\boldsymbol{v}, \boldsymbol{u}), && \text{(symmetry)} \\ (\boldsymbol{v}_L - \boldsymbol{v}_R)^T \boldsymbol{f}_S(\boldsymbol{u}_L, \boldsymbol{u}_R) &= \psi_L - \psi_R, && \text{(conservation).} \end{aligned}$$

- Can show numerical scheme dissipates entropy

$$\int_\Omega \frac{\partial S(\boldsymbol{u})}{\partial t} \approx \sum_i h \frac{\mathrm{d}S(\mathbf{u}_i)}{\mathrm{dt}} = \mathbf{v}^T \mathbf{d}(\mathbf{u}) \overset{?}{\leq} 0.$$

Tadmor (1987). *The numerical viscosity of entropy stable schemes for systems of conservation laws.*

Example of EC fluxes (compressible Euler equations)

- Define average $\{\{u\}\} = \frac{1}{2}(u_L + u_R)$. In one dimension:

$$f_S^1(\boldsymbol{u}_L, \boldsymbol{u}_R) = \{\{\rho\}\}^{\log} \{\{u\}\}$$
$$f_S^2(\boldsymbol{u}_L, \boldsymbol{u}_R) = \{\{u\}\} f_S^1 + p_{\text{avg}}$$
$$f_S^3(\boldsymbol{u}_L, \boldsymbol{u}_R) = (E_{\text{avg}} + p_{\text{avg}}) \{\{u\}\},$$

$$p_{\text{avg}} = \frac{\{\{\rho\}\}}{2\{\{\beta\}\}}, \qquad E_{\text{avg}} = \frac{\{\{\rho\}\}^{\log}}{2\{\{\beta\}\}^{\log}(\gamma - 1)} + \frac{1}{2}u_L u_R.$$

- Non-standard logarithmic mean, "inverse temperature" β

$$\{\{u\}\}^{\log} = \frac{u_L - u_R}{\log u_L - \log u_R}, \qquad \beta = \frac{\rho}{2p}.$$

Chandreshekar (2013), *Kinetic energy preserving and entropy stable finite volume schemes for the compressible Euler and Navier-Stokes equations.*

Matrix reformulation using Hadamard products

Hadamard product of two matrices $\mathbf{A} \circ \mathbf{B}$

$$\begin{bmatrix} \mathbf{A}_{11} & \dots & \mathbf{A}_{1n} \\ \vdots & \ddots & \vdots \\ \mathbf{A}_{n1} & \dots & \mathbf{A}_{nn} \end{bmatrix} \circ \begin{bmatrix} \mathbf{B}_{11} & \dots & \mathbf{B}_{1n} \\ \vdots & \ddots & \vdots \\ \mathbf{B}_{n1} & \dots & \mathbf{B}_{nn} \end{bmatrix} = \begin{bmatrix} \mathbf{A}_{11}\mathbf{B}_{11} & \dots & \mathbf{A}_{1n}\mathbf{B}_{1n} \\ \vdots & \ddots & \vdots \\ \mathbf{A}_{n1}\mathbf{B}_{n1} & \dots & \mathbf{A}_{nn}\mathbf{B}_{nn} \end{bmatrix}.$$

Rewrite an N-point (periodic) finite volume scheme as

$$\frac{\mathrm{d}}{\mathrm{dt}} \begin{bmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \vdots \\ \mathbf{u}_N \end{bmatrix} + \frac{1}{h} \begin{bmatrix} \boldsymbol{f}_S(\mathbf{u}_1, \mathbf{u}_2) - \boldsymbol{f}_S(\mathbf{u}_N, \mathbf{u}_1) \\ \boldsymbol{f}_S(\mathbf{u}_2, \mathbf{u}_3) - \boldsymbol{f}_S(\mathbf{u}_1, \mathbf{u}_2) \\ \vdots \\ \boldsymbol{f}_S(\mathbf{u}_N, \mathbf{u}_1) - \boldsymbol{f}_S(\mathbf{u}_{N-1}, \mathbf{u}_N) \end{bmatrix} = \mathbf{0}.$$

Matrix reformulation using Hadamard products

Hadamard product of two matrices $\mathbf{A} \circ \mathbf{B}$

$$\begin{bmatrix} \mathbf{A}_{11} & \dots & \mathbf{A}_{1n} \\ \vdots & \ddots & \vdots \\ \mathbf{A}_{n1} & \dots & \mathbf{A}_{nn} \end{bmatrix} \circ \begin{bmatrix} \mathbf{B}_{11} & \dots & \mathbf{B}_{1n} \\ \vdots & \ddots & \vdots \\ \mathbf{B}_{n1} & \dots & \mathbf{B}_{nn} \end{bmatrix} = \begin{bmatrix} \mathbf{A}_{11}\mathbf{B}_{11} & \dots & \mathbf{A}_{1n}\mathbf{B}_{1n} \\ \vdots & \ddots & \vdots \\ \mathbf{A}_{n1}\mathbf{B}_{n1} & \dots & \mathbf{A}_{nn}\mathbf{B}_{nn} \end{bmatrix}.$$

Rewrite an N-point (periodic) finite volume scheme as

$$\frac{\mathrm{d}}{\mathrm{dt}} \begin{bmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \vdots \\ \mathbf{u}_N \end{bmatrix} + \frac{1}{h} \begin{bmatrix} \boldsymbol{f}_S(\mathbf{u}_1, \mathbf{u}_2) - \boldsymbol{f}_S(\mathbf{u}_N, \mathbf{u}_1) \\ \boldsymbol{f}_S(\mathbf{u}_2, \mathbf{u}_3) - \boldsymbol{f}_S(\mathbf{u}_1, \mathbf{u}_2) \\ \vdots \\ \boldsymbol{f}_S(\mathbf{u}_N, \mathbf{u}_1) - \boldsymbol{f}_S(\mathbf{u}_{N-1}, \mathbf{u}_N) \end{bmatrix} = \mathbf{0}.$$

Matrix reformulation using Hadamard products

Hadamard product of two matrices $\mathbf{A} \circ \mathbf{B}$

$$\begin{bmatrix} \mathbf{A}_{11} & \dots & \mathbf{A}_{1n} \\ \vdots & \ddots & \vdots \\ \mathbf{A}_{n1} & \dots & \mathbf{A}_{nn} \end{bmatrix} \circ \begin{bmatrix} \mathbf{B}_{11} & \dots & \mathbf{B}_{1n} \\ \vdots & \ddots & \vdots \\ \mathbf{B}_{n1} & \dots & \mathbf{B}_{nn} \end{bmatrix} = \begin{bmatrix} \mathbf{A}_{11}\mathbf{B}_{11} & \dots & \mathbf{A}_{1n}\mathbf{B}_{1n} \\ \vdots & \ddots & \vdots \\ \mathbf{A}_{n1}\mathbf{B}_{n1} & \dots & \mathbf{A}_{nn}\mathbf{B}_{nn} \end{bmatrix}.$$

Rewrite an N-point (periodic) finite volume scheme as

$$h\frac{\mathrm{d}}{\mathrm{dt}} \begin{bmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \vdots \\ \mathbf{u}_N \end{bmatrix} + \begin{bmatrix} \mathbf{F}_{1,2} - \mathbf{F}_{1,N} \\ \mathbf{F}_{2,3} - \mathbf{F}_{2,1} \\ \vdots \\ \mathbf{F}_{N,1} - \mathbf{F}_{N,N-1} \end{bmatrix} = \mathbf{0}, \qquad \mathbf{F}_{ij} = \boldsymbol{f}_S(\mathbf{u}_i, \mathbf{u}_j).$$

Matrix reformulation using Hadamard products

Hadamard product of two matrices $\mathbf{A} \circ \mathbf{B}$

$$\begin{bmatrix} \mathbf{A}_{11} & \dots & \mathbf{A}_{1n} \\ \vdots & \ddots & \vdots \\ \mathbf{A}_{n1} & \dots & \mathbf{A}_{nn} \end{bmatrix} \circ \begin{bmatrix} \mathbf{B}_{11} & \dots & \mathbf{B}_{1n} \\ \vdots & \ddots & \vdots \\ \mathbf{B}_{n1} & \dots & \mathbf{B}_{nn} \end{bmatrix} = \begin{bmatrix} \mathbf{A}_{11}\mathbf{B}_{11} & \dots & \mathbf{A}_{1n}\mathbf{B}_{1n} \\ \vdots & \ddots & \vdots \\ \mathbf{A}_{n1}\mathbf{B}_{n1} & \dots & \mathbf{A}_{nn}\mathbf{B}_{nn} \end{bmatrix}.$$

Rewrite an N-point (periodic) finite volume scheme as

$$\begin{bmatrix} \mathbf{F}_{1,2} - \mathbf{F}_{1,N} \\ \mathbf{F}_{2,3} - \mathbf{F}_{2,1} \\ \vdots \\ \mathbf{F}_{N,1} - \mathbf{F}_{N,N-1} \end{bmatrix} = \left(\begin{bmatrix} 0 & 1 & & -1 \\ -1 & 0 & 1 & \\ & \ddots & \ddots & 1 \\ 1 & & -1 & 0 \end{bmatrix} \circ \begin{bmatrix} \mathbf{F}_{1,1} & \dots & \mathbf{F}_{1,N} \\ \vdots & \ddots & \vdots \\ \mathbf{F}_{N,1} & \dots & \mathbf{F}_{N,N} \end{bmatrix} \right) \mathbf{1}.$$

Matrix reformulation using Hadamard products

Hadamard product of two matrices $\mathbf{A} \circ \mathbf{B}$

$$\begin{bmatrix} \mathbf{A}_{11} & \dots & \mathbf{A}_{1n} \\ \vdots & \ddots & \vdots \\ \mathbf{A}_{n1} & \dots & \mathbf{A}_{nn} \end{bmatrix} \circ \begin{bmatrix} \mathbf{B}_{11} & \dots & \mathbf{B}_{1n} \\ \vdots & \ddots & \vdots \\ \mathbf{B}_{n1} & \dots & \mathbf{B}_{nn} \end{bmatrix} = \begin{bmatrix} \mathbf{A}_{11}\mathbf{B}_{11} & \dots & \mathbf{A}_{1n}\mathbf{B}_{1n} \\ \vdots & \ddots & \vdots \\ \mathbf{A}_{n1}\mathbf{B}_{n1} & \dots & \mathbf{A}_{nn}\mathbf{B}_{nn} \end{bmatrix}.$$

Rewrite an N-point (periodic) finite volume scheme as

$$\begin{bmatrix} \mathbf{F}_{1,2} - \mathbf{F}_{1,N} \\ \mathbf{F}_{2,3} - \mathbf{F}_{2,1} \\ \vdots \\ \mathbf{F}_{N,1} - \mathbf{F}_{N,N-1} \end{bmatrix} = \left(\underbrace{\begin{bmatrix} 0 & 1 & & -1 \\ -1 & 0 & 1 & \\ & \ddots & \ddots & 1 \\ 1 & & -1 & 0 \end{bmatrix}}_{2\mathbf{Q}} \circ \underbrace{\begin{bmatrix} \mathbf{F}_{1,1} & \dots & \mathbf{F}_{1,N} \\ \vdots & \ddots & \vdots \\ \mathbf{F}_{N,1} & \dots & \mathbf{F}_{N,N} \end{bmatrix}}_{\mathbf{F}} \right) \mathbf{1}.$$

Matrix reformulation using Hadamard products

Hadamard product of two matrices $\mathbf{A} \circ \mathbf{B}$

$$\begin{bmatrix} \mathbf{A}_{11} & \dots & \mathbf{A}_{1n} \\ \vdots & \ddots & \vdots \\ \mathbf{A}_{n1} & \dots & \mathbf{A}_{nn} \end{bmatrix} \circ \begin{bmatrix} \mathbf{B}_{11} & \dots & \mathbf{B}_{1n} \\ \vdots & \ddots & \vdots \\ \mathbf{B}_{n1} & \dots & \mathbf{B}_{nn} \end{bmatrix} = \begin{bmatrix} \mathbf{A}_{11}\mathbf{B}_{11} & \dots & \mathbf{A}_{1n}\mathbf{B}_{1n} \\ \vdots & \ddots & \vdots \\ \mathbf{A}_{n1}\mathbf{B}_{n1} & \dots & \mathbf{A}_{nn}\mathbf{B}_{nn} \end{bmatrix}.$$

Rewrite an N-point (periodic) finite volume scheme as

$$\begin{bmatrix} \mathbf{F}_{1,2} - \mathbf{F}_{1,N} \\ \mathbf{F}_{2,3} - \mathbf{F}_{2,1} \\ \vdots \\ \mathbf{F}_{N,1} - \mathbf{F}_{N,N-1} \end{bmatrix} = 2(\mathbf{Q} \circ \mathbf{F})\mathbf{1}.$$

Interpretation using finite difference matrices

- Let $\mathbf{M} = h\mathbf{I}$. Can reformulate an entropy conservative finite volume method as

$$\mathbf{M}\frac{\mathrm{d}\mathbf{u}}{\mathrm{dt}} + 2\left(\mathbf{Q} \circ \mathbf{F}\right)\mathbf{1} = \mathbf{0}, \qquad \mathbf{Q} = \frac{1}{2}\begin{bmatrix} 0 & 1 & & -1 \\ -1 & 0 & 1 & \\ & \ddots & \ddots & 1 \\ 1 & & -1 & 0 \end{bmatrix}$$

- Generalizable: can show entropy conservation for any matrix which satisfies $\mathbf{Q} = -\mathbf{Q}^T$ and $\mathbf{Q}\mathbf{1} = \mathbf{0}$!
- Note: $\mathbf{M}^{-1}\mathbf{Q}$ is a 2nd order (periodic) differentiation matrix.

- Let $\mathbf{M} = h\mathbf{I}$. Can reformulate an entropy conservative finite volume method as

$$\mathbf{M}\frac{\mathrm{d}\mathbf{u}}{\mathrm{dt}} + 2\left(\mathbf{Q} \circ \mathbf{F}\right)\mathbf{1} = \mathbf{0}, \qquad \mathbf{Q} = \frac{1}{2}\begin{bmatrix} 0 & 1 & & -1 \\ -1 & 0 & 1 & \\ & \ddots & \ddots & 1 \\ 1 & & -1 & 0 \end{bmatrix}$$

- Generalizable: can show entropy conservation for any matrix which satisfies $\mathbf{Q} = -\mathbf{Q}^T$ and $\mathbf{Q1} = \mathbf{0}$!
- Note: $\mathbf{M}^{-1}\mathbf{Q}$ is a 2nd order (periodic) differentiation matrix.

Interpretation using finite difference matrices

- Let $\mathbf{M} = h\mathbf{I}$. Can reformulate an entropy conservative finite volume method as

$$\mathbf{M}\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} + 2\left(\mathbf{Q} \circ \mathbf{F}\right)\mathbf{1} = \mathbf{0}, \qquad \mathbf{Q} = \frac{1}{2}\begin{bmatrix} 0 & 1 & & -1 \\ -1 & 0 & 1 & \\ & \ddots & \ddots & 1 \\ 1 & & -1 & 0 \end{bmatrix}$$

- Generalizable: can show entropy conservation for any matrix which satisfies $\mathbf{Q} = -\mathbf{Q}^T$ and $\mathbf{Q}\mathbf{1} = \mathbf{0}$!
- Note: $\mathbf{M}^{-1}\mathbf{Q}$ is a 2nd order (periodic) differentiation matrix.

Boundary conditions and summation-by-parts (SBP) property

Boundary conditions: choose appropriate "ghost" values $\mathbf{u}_1^+, \mathbf{u}_N^+$

$$\mathbf{M}\frac{\mathrm{d}\mathbf{u}}{\mathrm{dt}} + 2\left(\mathbf{Q} \circ \mathbf{F}\right)\mathbf{1} + \mathbf{B}\begin{bmatrix} \boldsymbol{f}_S(\mathbf{u}_1^+, \mathbf{u}_1) - \boldsymbol{f}(\mathbf{u}_1) \\ \mathbf{0} \\ \boldsymbol{f}_S(\mathbf{u}_N^+, \mathbf{u}_N) - \boldsymbol{f}(\mathbf{u}_N) \end{bmatrix} = \mathbf{0}.$$

Entropy stable if $\mathbf{Q}$ satisfies a summation-by-parts (SBP) property

$$\mathbf{Q} = \frac{1}{2}\begin{bmatrix} -1 & 1 & & 0 \\ -1 & 0 & 1 & \\ & \ddots & \ddots & 1 \\ & & -1 & 1 \end{bmatrix}, \qquad \mathbf{Q}+\mathbf{Q}^T = \mathbf{B} = \begin{bmatrix} -1 & & & \\ & 0 & & \\ & & \ddots & \\ & & & 1 \end{bmatrix}.$$

Main innovation: fully algebraic proof of entropy stability

- Discrete analogue of the entropy identity

$$\boxed{\int_{-1}^{1} \boldsymbol{v}^T \frac{\partial \boldsymbol{f}(\boldsymbol{u})}{\partial x} = \mathbf{v}^T \boldsymbol{f}(\mathbf{u}) - \psi(\mathbf{u})\Big|_{-1}^{1}} \quad \Longleftrightarrow \quad \boxed{\begin{aligned} &\mathbf{v}^T \left(2\mathbf{Q} \circ \mathbf{F}\right) \mathbf{1} \\ &= \mathbf{1}^T \mathbf{B} \left(\mathbf{v}^T \boldsymbol{f}(\mathbf{u}) - \psi(\mathbf{u})\right) \end{aligned}}$$

- Expand $\mathbf{v}^T \left(2\mathbf{Q} \circ \mathbf{F}\right) \mathbf{1}$ using the SBP property $\mathbf{Q} + \mathbf{Q}^T = \mathbf{B}$

$$\mathbf{v}^T \left(2\mathbf{Q} \circ \mathbf{F}\right) \mathbf{1}$$

- Manipulate volume term using properties of $\mathbf{Q}$ and f_S.

$$\mathbf{v}^T \left(\left(\mathbf{Q} - \mathbf{Q}^T\right) \circ \mathbf{F}\right) \mathbf{1} = \sum_{ij} \mathbf{Q}_{ij} \left(\mathbf{v}_i - \mathbf{v}_j\right)^T f_S\left(\mathbf{u}_i, \mathbf{u}_j\right)$$

Tadmor (1987), Carpenter et al. (2014), Gassner, Winters, and Kopriva (2016).

Main innovation: fully algebraic proof of entropy stability

- Discrete analogue of the entropy identity

$$\boxed{\int_{-1}^{1} \boldsymbol{v}^T \frac{\partial \boldsymbol{f}(\boldsymbol{u})}{\partial x} = \mathbf{v}^T \boldsymbol{f}(\mathbf{u}) - \psi(\mathbf{u})\Big|_{-1}^{1}} \quad \Longleftrightarrow \quad \boxed{\begin{aligned} &\mathbf{v}^T (2\mathbf{Q} \circ \mathbf{F}) \mathbf{1} \\ &= \mathbf{1}^T \mathbf{B} \left(\mathbf{v}^T \boldsymbol{f}(\mathbf{u}) - \psi(\mathbf{u}) \right) \end{aligned}}$$

- Expand $\mathbf{v}^T (2\mathbf{Q} \circ \mathbf{F}) \mathbf{1}$ using the SBP property $\mathbf{Q} + \mathbf{Q}^T = \mathbf{B}$

$$\mathbf{v}^T (2\mathbf{Q} \circ \mathbf{F}) \mathbf{1}$$

- Manipulate volume term using properties of $\mathbf{Q}$ and f_S.

$$\mathbf{v}^T \left(\left(\mathbf{Q} - \mathbf{Q}^T \right) \circ \mathbf{F} \right) \mathbf{1} = \sum_{ij} \mathbf{Q}_{ij} \left(\mathbf{v}_i - \mathbf{v}_j \right)^T f_S \left(\mathbf{u}_i, \mathbf{u}_j \right)$$

Tadmor (1987), Carpenter et al. (2014), Gassner, Winters, and Kopriva (2016).

Main innovation: fully algebraic proof of entropy stability

- Discrete analogue of the entropy identity

$$\boxed{\int_{-1}^{1} \boldsymbol{v}^T \frac{\partial \boldsymbol{f}(\boldsymbol{u})}{\partial x} = \mathbf{v}^T \boldsymbol{f}(\mathbf{u}) - \psi(\mathbf{u})\Big|_{-1}^{1}} \quad \Longleftrightarrow \quad \boxed{\begin{aligned} &\mathbf{v}^T \left(2\mathbf{Q} \circ \mathbf{F}\right) \mathbf{1} \\ &= \mathbf{1}^T \mathbf{B} \left(\mathbf{v}^T \boldsymbol{f}(\mathbf{u}) - \psi(\mathbf{u})\right) \end{aligned}}$$

- Expand $\mathbf{v}^T \left(2\mathbf{Q} \circ \mathbf{F}\right) \mathbf{1}$ using the SBP property $\mathbf{Q} + \mathbf{Q}^T = \mathbf{B}$

$$\mathbf{v}^T \left(\left(\mathbf{Q} - \mathbf{Q}^T\right) \circ \mathbf{F}\right) \mathbf{1} + \mathbf{v}^T \left(\mathbf{B} \circ \mathbf{F}\right) \mathbf{1}, \quad \text{(SBP property)}$$

- Manipulate volume term using properties of $\mathbf{Q}$ and f_S.

$$\mathbf{v}^T \left(\left(\mathbf{Q} - \mathbf{Q}^T\right) \circ \mathbf{F}\right) \mathbf{1} = \sum_{ij} \mathbf{Q}_{ij} \left(\mathbf{v}_i - \mathbf{v}_j\right)^T \boldsymbol{f}_S\left(\mathbf{u}_i, \mathbf{u}_j\right)$$

Tadmor (1987), Carpenter et al. (2014), Gassner, Winters, and Kopriva (2016).

Main innovation: fully algebraic proof of entropy stability

- Discrete analogue of the entropy identity

$$\int_{-1}^{1} \boldsymbol{v}^T \frac{\partial \boldsymbol{f}(\boldsymbol{u})}{\partial x} = \mathbf{v}^T \boldsymbol{f}(\mathbf{u}) - \psi(\mathbf{u})\Big|_{-1}^{1} \quad \Longleftrightarrow \quad \begin{aligned} &\mathbf{v}^T (2\mathbf{Q} \circ \mathbf{F}) \mathbf{1} \\ &= \mathbf{1}^T \mathbf{B} \left(\mathbf{v}^T \boldsymbol{f}(\mathbf{u}) - \psi(\mathbf{u})\right) \end{aligned}$$

- Expand $\mathbf{v}^T (2\mathbf{Q} \circ \mathbf{F}) \mathbf{1}$ using the SBP property $\mathbf{Q} + \mathbf{Q}^T = \mathbf{B}$

$$\mathbf{v}^T \left(\left(\mathbf{Q} - \mathbf{Q}^T\right) \circ \mathbf{F}\right) \mathbf{1} + \mathbf{v}^T \mathbf{B} \boldsymbol{f}(\mathbf{u}), \quad \text{(consistency, } \mathbf{B} \text{ diag)}$$

- Manipulate volume term using properties of $\mathbf{Q}$ and f_S.

$$\mathbf{v}^T \left(\left(\mathbf{Q} - \mathbf{Q}^T\right) \circ \mathbf{F}\right) \mathbf{1} = \sum_{ij} \mathbf{Q}_{ij} \left(\mathbf{v}_i - \mathbf{v}_j\right)^T f_S\left(\mathbf{u}_i, \mathbf{u}_j\right)$$

Tadmor (1987), Carpenter et al. (2014), Gassner, Winters, and Kopriva (2016).

Main innovation: fully algebraic proof of entropy stability

- Discrete analogue of the entropy identity

$$\int_{-1}^{1} \boldsymbol{v}^T \frac{\partial \boldsymbol{f}(\boldsymbol{u})}{\partial x} = \mathbf{v}^T \boldsymbol{f}(\mathbf{u}) - \psi(\mathbf{u})\Big|_{-1}^{1} \iff \mathbf{v}^T \left(2\mathbf{Q} \circ \mathbf{F}\right)\mathbf{1} = \mathbf{1}^T\mathbf{B}\left(\mathbf{v}^T \boldsymbol{f}(\mathbf{u}) - \psi(\mathbf{u})\right)$$

- Expand $\mathbf{v}^T \left(2\mathbf{Q} \circ \mathbf{F}\right)\mathbf{1}$ using the SBP property $\mathbf{Q} + \mathbf{Q}^T = \mathbf{B}$

$$\mathbf{v}^T \left(\left(\mathbf{Q} - \mathbf{Q}^T\right) \circ \mathbf{F}\right)\mathbf{1} + \mathbf{v}^T\mathbf{B}\boldsymbol{f}(\mathbf{u}), \quad \text{(consistency, } \mathbf{B} \text{ diag)}$$

- Manipulate volume term using properties of $\mathbf{Q}$ and $\boldsymbol{f}_S$.

$$\mathbf{v}^T \left(\left(\mathbf{Q} - \mathbf{Q}^T\right) \circ \mathbf{F}\right)\mathbf{1} = \sum_{ij} \mathbf{Q}_{ij} \left(\mathbf{v}_i - \mathbf{v}_j\right)^T \boldsymbol{f}_S\left(\mathbf{u}_i, \mathbf{u}_j\right)$$

Tadmor (1987), Carpenter et al. (2014), Gassner, Winters, and Kopriva (2016).

Main innovation: fully algebraic proof of entropy stability

- Discrete analogue of the entropy identity

$$\int_{-1}^{1} \boldsymbol{v}^T \frac{\partial \boldsymbol{f}(\boldsymbol{u})}{\partial x} = \mathbf{v}^T \boldsymbol{f}(\mathbf{u}) - \psi(\mathbf{u})\Big|_{-1}^{1} \iff \mathbf{v}^T (2\mathbf{Q} \circ \mathbf{F})\mathbf{1} = \mathbf{1}^T \mathbf{B}\left(\mathbf{v}^T \boldsymbol{f}(\mathbf{u}) - \psi(\mathbf{u})\right)$$

- Expand $\mathbf{v}^T (2\mathbf{Q} \circ \mathbf{F})\mathbf{1}$ using the SBP property $\mathbf{Q} + \mathbf{Q}^T = \mathbf{B}$

$$\mathbf{v}^T \left(\left(\mathbf{Q} - \mathbf{Q}^T\right) \circ \mathbf{F}\right)\mathbf{1} + \mathbf{v}^T \mathbf{B} \boldsymbol{f}(\mathbf{u}), \quad (\text{consistency, } \mathbf{B} \text{ diag})$$

- Manipulate volume term using properties of $\mathbf{Q}$ and $\boldsymbol{f}_S$.

$$\mathbf{v}^T \left(\left(\mathbf{Q} - \mathbf{Q}^T\right) \circ \mathbf{F}\right)\mathbf{1} = \sum_{ij} \mathbf{Q}_{ij} \left(\psi(\mathbf{u}_i) - \psi(\mathbf{u}_j)\right)$$

Tadmor (1987), Carpenter et al. (2014), Gassner, Winters, and Kopriva (2016).

Main innovation: fully algebraic proof of entropy stability

- Discrete analogue of the entropy identity

$$\int_{-1}^{1} \boldsymbol{v}^T \frac{\partial \boldsymbol{f}(\boldsymbol{u})}{\partial x} = \mathbf{v}^T \boldsymbol{f}(\mathbf{u}) - \psi(\mathbf{u})\Big|_{-1}^{1} \iff \mathbf{v}^T (2\mathbf{Q} \circ \mathbf{F})\,\mathbf{1} = \mathbf{1}^T \mathbf{B} \left(\mathbf{v}^T \boldsymbol{f}(\mathbf{u}) - \psi(\mathbf{u})\right)$$

- Expand $\mathbf{v}^T (2\mathbf{Q} \circ \mathbf{F})\,\mathbf{1}$ using the SBP property $\mathbf{Q} + \mathbf{Q}^T = \mathbf{B}$

$$\mathbf{v}^T \left(\left(\mathbf{Q} - \mathbf{Q}^T\right) \circ \mathbf{F}\right)\mathbf{1} + \mathbf{v}^T \mathbf{B} \boldsymbol{f}(\mathbf{u}), \quad \text{(consistency, } \mathbf{B} \text{ diag)}$$

- Manipulate volume term using properties of $\mathbf{Q}$ and $\boldsymbol{f}_S$.

$$\mathbf{v}^T \left(\left(\mathbf{Q} - \mathbf{Q}^T\right) \circ \mathbf{F}\right)\mathbf{1} = \psi(\mathbf{u})^T \mathbf{Q}\mathbf{1} - \mathbf{1}^T \mathbf{Q}\psi(\mathbf{u})$$

Tadmor (1987), Carpenter et al. (2014), Gassner, Winters, and Kopriva (2016).

Main innovation: fully algebraic proof of entropy stability

- Discrete analogue of the entropy identity

$$\boxed{\int_{-1}^{1} \boldsymbol{v}^T \frac{\partial \boldsymbol{f}(\boldsymbol{u})}{\partial x} = \mathbf{v}^T \boldsymbol{f}(\mathbf{u}) - \psi(\mathbf{u})\Big|_{-1}^{1}} \iff \boxed{\begin{aligned} &\mathbf{v}^T \left(2\mathbf{Q} \circ \mathbf{F}\right)\mathbf{1} \\ &= \mathbf{1}^T \mathbf{B}\left(\mathbf{v}^T \boldsymbol{f}(\mathbf{u}) - \psi(\mathbf{u})\right)\end{aligned}}$$

- Expand $\mathbf{v}^T \left(2\mathbf{Q} \circ \mathbf{F}\right)\mathbf{1}$ using the SBP property $\mathbf{Q} + \mathbf{Q}^T = \mathbf{B}$

$$\mathbf{v}^T \left(\left(\mathbf{Q} - \mathbf{Q}^T\right) \circ \mathbf{F}\right)\mathbf{1} + \mathbf{v}^T \mathbf{B} \boldsymbol{f}(\mathbf{u}), \quad \text{(consistency, } \mathbf{B} \text{ diag)}$$

- Manipulate volume term using properties of $\mathbf{Q}$ and $\boldsymbol{f}_S$.

$$\mathbf{v}^T \left(\left(\mathbf{Q} - \mathbf{Q}^T\right) \circ \mathbf{F}\right)\mathbf{1} = \underbrace{\psi(\mathbf{u})^T \mathbf{Q}\mathbf{1}}_{=\mathbf{0}} - \mathbf{1}^T \mathbf{Q} \psi(\mathbf{u})$$

Tadmor (1987), Carpenter et al. (2014), Gassner, Winters, and Kopriva (2016).

Main innovation: fully algebraic proof of entropy stability

- Discrete analogue of the entropy identity

$$\int_{-1}^{1} \boldsymbol{v}^T \frac{\partial \boldsymbol{f}(\boldsymbol{u})}{\partial x} = \mathbf{v}^T \boldsymbol{f}(\mathbf{u}) - \psi(\mathbf{u})\Big|_{-1}^{1} \iff \mathbf{v}^T \left(2\mathbf{Q} \circ \mathbf{F}\right)\mathbf{1} = \mathbf{1}^T \mathbf{B}\left(\mathbf{v}^T \boldsymbol{f}(\mathbf{u}) - \psi(\mathbf{u})\right)$$

- Expand $\mathbf{v}^T \left(2\mathbf{Q} \circ \mathbf{F}\right)\mathbf{1}$ using the SBP property $\mathbf{Q} + \mathbf{Q}^T = \mathbf{B}$

$$\mathbf{v}^T \left(\left(\mathbf{Q} - \mathbf{Q}^T\right) \circ \mathbf{F}\right)\mathbf{1} + \mathbf{v}^T \mathbf{B} \boldsymbol{f}(\mathbf{u}), \quad \text{(consistency, } \mathbf{B} \text{ diag)}$$

- Manipulate volume term using properties of $\mathbf{Q}$ and $\boldsymbol{f}_S$.

$$\mathbf{v}^T \left(\left(\mathbf{Q} - \mathbf{Q}^T\right) \circ \mathbf{F}\right)\mathbf{1} = \underbrace{\psi(\mathbf{u})^T \mathbf{Q}\mathbf{1}}_{=0} - \underbrace{\mathbf{1}^T \mathbf{Q}\psi(\mathbf{u})}_{=-\mathbf{1}^T \mathbf{B}\psi(\mathbf{u})}$$

Tadmor (1987), Carpenter et al. (2014), Gassner, Winters, and Kopriva (2016).

Main innovation: fully algebraic proof of entropy stability

- Discrete analogue of the entropy identity

$$\int_{-1}^{1} \boldsymbol{v}^T \frac{\partial \boldsymbol{f}(\boldsymbol{u})}{\partial x} = \mathbf{v}^T \boldsymbol{f}(\mathbf{u}) - \psi(\mathbf{u})\Big|_{-1}^{1} \iff \mathbf{v}^T \left(2\mathbf{Q} \circ \mathbf{F}\right) \mathbf{1} = \mathbf{1}^T \mathbf{B} \left(\mathbf{v}^T \boldsymbol{f}(\mathbf{u}) - \psi(\mathbf{u})\right)$$

- Expand $\mathbf{v}^T \left(2\mathbf{Q} \circ \mathbf{F}\right) \mathbf{1}$ using the SBP property $\mathbf{Q} + \mathbf{Q}^T = \mathbf{B}$

$$\mathbf{v}^T \left(\left(\mathbf{Q} - \mathbf{Q}^T\right) \circ \mathbf{F}\right) \mathbf{1} + \mathbf{v}^T \mathbf{B} \boldsymbol{f}(\mathbf{u}), \quad \text{(consistency, } \mathbf{B} \text{ diag)}$$

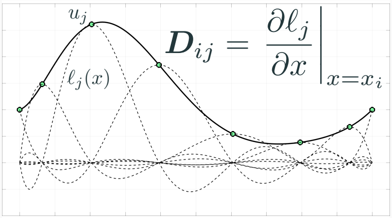
- Manipulate volume term using properties of $\mathbf{Q}$ and $\boldsymbol{f}_S$.

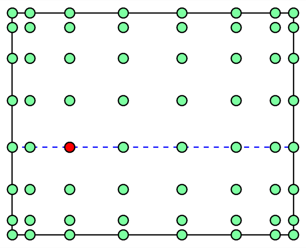
$$\mathbf{v}^T \left(\left(\mathbf{Q} - \mathbf{Q}^T\right) \circ \mathbf{F}\right) \mathbf{1} = -\mathbf{1}^T \mathbf{B} \psi(\mathbf{u})$$

Tadmor (1987), Carpenter et al. (2014), Gassner, Winters, and Kopriva (2016).

Entropy stable high order summation by parts (SBP) schemes

High order nodal differentiation matrices





- Nodal differentiation matrix $\mathbf{D}$ has zero row sums

$$\sum_j \mathbf{D}_{ij} = 0 \quad \Longrightarrow \quad \mathbf{D1} = \mathbf{0}.$$

- Lobatto quadrature nodes recover summation-by-parts property! Let $\mathbf{M}$ = lumped diagonal mass matrix:

$$\mathbf{Q} = \mathbf{MD}, \qquad \boxed{\mathbf{Q} + \mathbf{Q}^T = \mathbf{B}}.$$

Entropy stable nodal DG: a brief summary

- If $\mathbf{Q}$ satisfies $\mathbf{Q1} = \mathbf{0}$ and the summation-by-parts (SBP) property, then the DG formulation is entropy *conservative*

$$\boxed{\mathbf{M}\frac{\mathrm{d}\mathbf{u}}{\mathrm{dt}} + 2\left(\mathbf{Q} \circ \mathbf{F}\right)\mathbf{1} + \mathbf{B}\Big(\underbrace{\boldsymbol{f}_S\left(\mathbf{u}^+, \mathbf{u}\right)}_{\text{interface flux}} - \boldsymbol{f}(\mathbf{u})\Big) = \mathbf{0}}$$

- Generalizes to arbitrarily high polynomial degree N.
- Adding interface dissipation (e.g., Lax-Friedrichs) yields an entropy stable DG scheme.

$$\boldsymbol{f}_S\left(\mathbf{u}^+, \mathbf{u}\right) \rightarrow \boldsymbol{f}_S\left(\mathbf{u}^+, \mathbf{u}\right) - \frac{\lambda}{2}[\![\mathbf{u}]\!]\boldsymbol{n}, \qquad \lambda > 0.$$

Fisher and Carpenter (2014), Gassner, Winters, and Kopriva (2016), Chen and Shu (2017), etc.

Entropy stable nodal DG: a brief summary

- If $\mathbf{Q}$ satisfies $\mathbf{Q1} = \mathbf{0}$ and the summation-by-parts (SBP) property, then the DG formulation is entropy *conservative*

$$\boxed{\mathbf{M}\frac{\mathrm{d}\mathbf{u}}{\mathrm{dt}} + 2\left(\mathbf{Q} \circ \mathbf{F}\right)\mathbf{1} + \mathbf{B}\Big(\underbrace{\boldsymbol{f}_S\left(\mathbf{u}^+, \mathbf{u}\right)}_{\text{interface flux}} - \boldsymbol{f}(\mathbf{u})\Big) = \mathbf{0}}$$

- Generalizes to arbitrarily high polynomial degree N.
- Adding interface dissipation (e.g., Lax-Friedrichs) yields an entropy stable DG scheme.

$$\boldsymbol{f}_S\left(\mathbf{u}^+, \mathbf{u}\right) \rightarrow \boldsymbol{f}_S\left(\mathbf{u}^+, \mathbf{u}\right) - \frac{\lambda}{2}[\![\mathbf{u}]\!]\boldsymbol{n}, \qquad \lambda > 0.$$

Fisher and Carpenter (2014), Gassner, Winters, and Kopriva (2016), Chen and Shu (2017), etc.

Entropy stable nodal DG: a brief summary

- If $\mathbf{Q}$ satisfies $\mathbf{Q1} = \mathbf{0}$ and the summation-by-parts (SBP) property, then the DG formulation is entropy *conservative*

$$\boxed{\mathbf{M}\frac{\mathrm{d}\mathbf{u}}{\mathrm{dt}} + 2\left(\mathbf{Q} \circ \mathbf{F}\right)\mathbf{1} + \mathbf{B}\Big(\underbrace{\boldsymbol{f}_S\left(\mathbf{u}^+, \mathbf{u}\right)}_{\text{interface flux}} - \boldsymbol{f}(\mathbf{u})\Big) = \mathbf{0}}$$

- Generalizes to arbitrarily high polynomial degree N.
- Adding interface dissipation (e.g., Lax-Friedrichs) yields an entropy stable DG scheme.

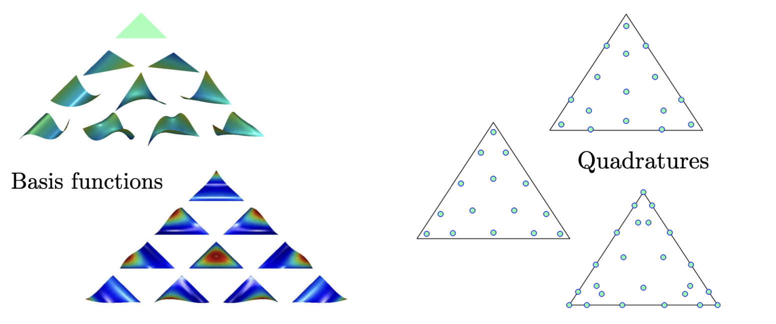
$$\boldsymbol{f}_S\left(\mathbf{u}^+, \mathbf{u}\right) \rightarrow \boldsymbol{f}_S\left(\mathbf{u}^+, \mathbf{u}\right) - \frac{\lambda}{2}[\![\mathbf{u}]\!]\boldsymbol{n}, \qquad \lambda > 0.$$

Fisher and Carpenter (2014), Gassner, Winters, and Kopriva (2016), Chen and Shu (2017), etc.

Entropy stable modal discontinuous Galerkin formulations

Why "modal" formulations?

Nodal formulations: tied to a specific set of nodes.
"Modal" formulations: arbitrary basis functions and quadrature.

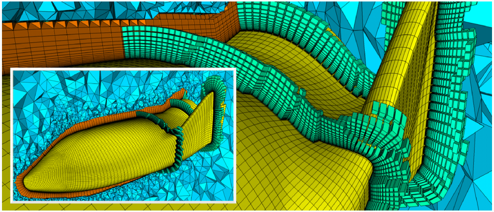


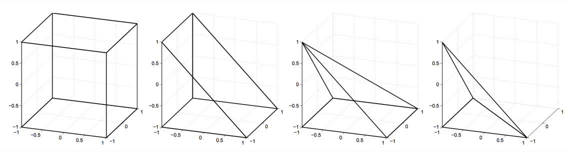
Enables use of standard tools in finite elements.

Figures from http://www2.compute.dtu.dk/ apek/DGFEMCourse2009/Lecture05.pdf.

Why “modal” formulations?

Nodal formulations: tied to a specific set of nodes.
“Modal” formulations: arbitrary basis functions and quadrature.



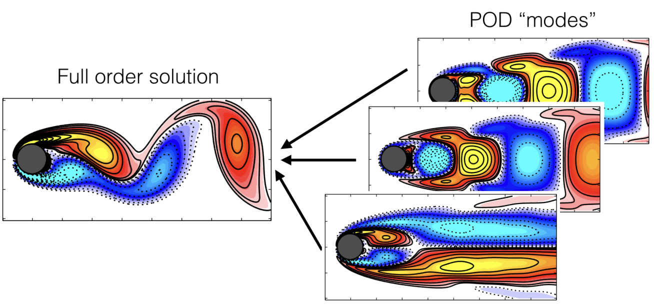


Applicable for any type of reference element.

https://www.pointwise.com/news/2014/Hybrid-Hex-Tet-Meshing-Latest-Pointwise-Release.html.

Why “modal” formulations?

Nodal formulations: tied to a specific set of nodes.
“Modal” formulations: arbitrary basis functions and quadrature.

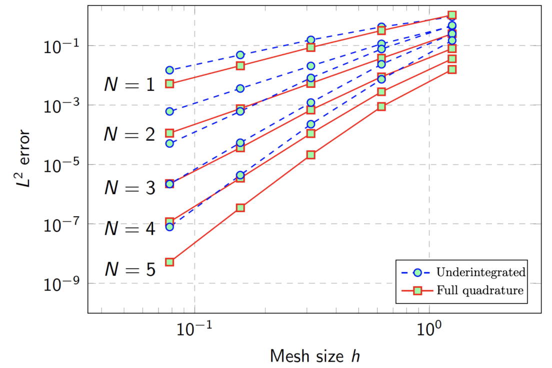


Projection-based reduced order models: learn basis functions from data.

Figure adapted from Brunton, Proctor, Kutz (2016), *Discovering governing equations from data ...*.

Why "modal" formulations?

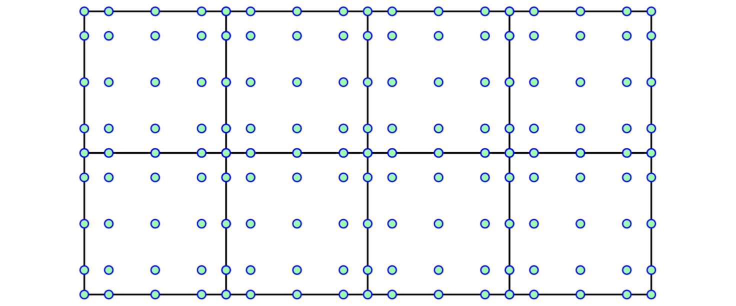
Nodal formulations: tied to a specific set of nodes.
"Modal" formulations: arbitrary basis functions and quadrature.



Can avoid *underintegration errors* for nonlinear terms + curved elements.

Why “modal” formulations?

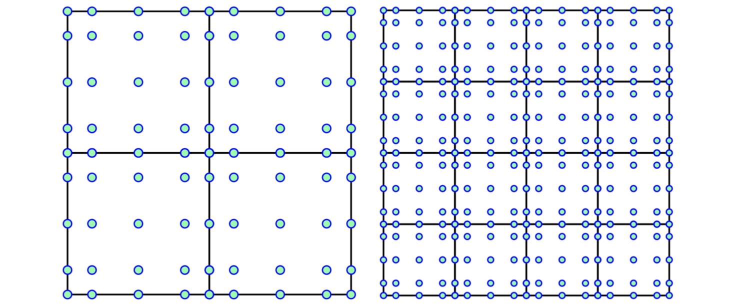
Nodal formulations: tied to a specific set of nodes.
“Modal” formulations: arbitrary basis functions and quadrature.



Nodal formulations are great for conforming high order meshes. . .

Why "modal" formulations?

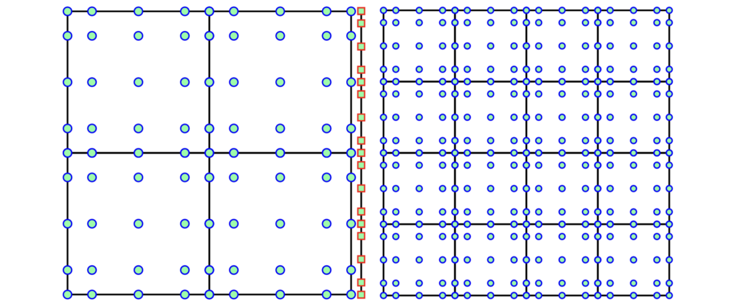
Nodal formulations: tied to a specific set of nodes.
"Modal" formulations: arbitrary basis functions and quadrature.



. . . but modal formulations make non-conforming meshes simpler.

Why “modal” formulations?

Nodal formulations: tied to a specific set of nodes.
“Modal” formulations: arbitrary basis functions and quadrature.



. . . but modal formulations make non-conforming meshes simpler.

Challenge 1 for modal formulations: entropy projection

- Test functions must be polynomial. Entropy variables are not.

- If $\boldsymbol{u}_N$ is polynomial, testing with L^2 projection of entropy variables $\Pi_N \boldsymbol{v}(\boldsymbol{u}_N)$ recovers rate of change of entropy

$$\int_{D^k} \Pi_N \boldsymbol{v}(\boldsymbol{u}_N)^T \frac{\partial \boldsymbol{u}_N}{\partial t} = \int_{D^k} \underbrace{\boldsymbol{v}(\boldsymbol{u}_N)^T}_{\frac{\partial S(\boldsymbol{u})}{\partial \boldsymbol{u}}} \frac{\partial \boldsymbol{u}_N}{\partial t} = \int_{D^k} \frac{\partial S(\boldsymbol{u}_N)}{\partial t}$$

- For consistency, must also evaluate fluxes using projected entropy variables $\widetilde{\boldsymbol{u}} = \boldsymbol{u}(\Pi_N \boldsymbol{v}(\boldsymbol{u}_N))$.

$$(\mathbf{v}_i - \mathbf{v}_j)^T \boldsymbol{f}_S(\mathbf{u}_i, \mathbf{u}_j) \neq \psi(\mathbf{u}_i) - \psi(\mathbf{u}_j) \quad \text{if } \mathbf{v}_i \neq \mathbf{v}(\mathbf{u}_i).$$

Challenge 1 for modal formulations: entropy projection

- Test functions must be polynomial. Entropy variables are not.

- If $\boldsymbol{u}_N$ is polynomial, testing with L^2 projection of entropy variables $\Pi_N \boldsymbol{v}\left(\boldsymbol{u}_N\right)$ recovers rate of change of entropy

$$\int_{D^k} \Pi_N \boldsymbol{v}\left(\boldsymbol{u}_N\right)^T \frac{\partial \boldsymbol{u}_N}{\partial t} = \int_{D^k} \underbrace{\boldsymbol{v}\left(\boldsymbol{u}_N\right)^T}_{\frac{\partial S(\boldsymbol{u})}{\partial \boldsymbol{u}}} \frac{\partial \boldsymbol{u}_N}{\partial t} = \int_{D^k} \frac{\partial S(\boldsymbol{u}_N)}{\partial t}$$

- For consistency, must also evaluate fluxes using projected entropy variables $\widetilde{\boldsymbol{u}} = \boldsymbol{u}\left(\Pi_N \boldsymbol{v}\left(\boldsymbol{u}_N\right)\right)$.

$$\left(\mathbf{v}_i - \mathbf{v}_j\right)^T \boldsymbol{f}_S\left(\mathbf{u}_i, \mathbf{u}_j\right) \neq \psi(\mathbf{u}_i) - \psi(\mathbf{u}_j) \quad \text{if } \mathbf{v}_i \neq \mathbf{v}\left(\mathbf{u}_i\right).$$

Challenge 1 for modal formulations: entropy projection

- Test functions must be polynomial. Entropy variables are not.

- If $\boldsymbol{u}_N$ is polynomial, testing with L^2 projection of entropy variables $\Pi_N \boldsymbol{v}(\boldsymbol{u}_N)$ recovers rate of change of entropy

$$\int_{D^k} \Pi_N \boldsymbol{v}(\boldsymbol{u}_N)^T \frac{\partial \boldsymbol{u}_N}{\partial t} = \int_{D^k} \underbrace{\boldsymbol{v}(\boldsymbol{u}_N)^T}_{\frac{\partial S(\boldsymbol{u})}{\partial \boldsymbol{u}}} \frac{\partial \boldsymbol{u}_N}{\partial t} = \int_{D^k} \frac{\partial S(\boldsymbol{u}_N)}{\partial t}$$

- For consistency, must also evaluate fluxes using projected entropy variables $\widetilde{\boldsymbol{u}} = \boldsymbol{u}(\Pi_N \boldsymbol{v}(\boldsymbol{u}_N))$.

$$(\mathbf{v}_i - \mathbf{v}_j)^T \boldsymbol{f}_S(\mathbf{u}_i, \mathbf{u}_j) \neq \psi(\mathbf{u}_i) - \psi(\mathbf{u}_j) \quad \text{if } \mathbf{v}_i \neq \mathbf{v}(\mathbf{u}_i).$$

Illustration of entropy projection: $N = 3$, 32 elements

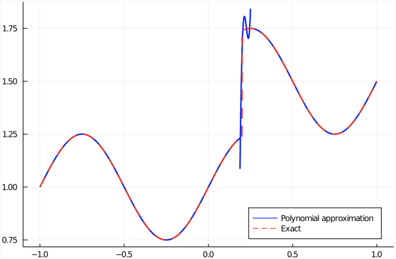
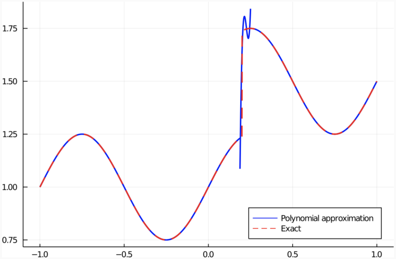


Illustration of entropy projection: $N = 3$, 32 elements



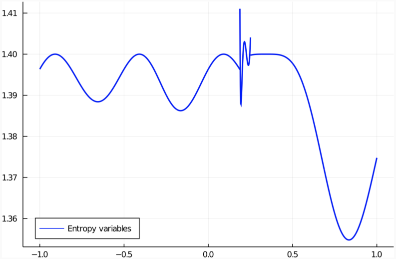


Illustration of entropy projection: $N = 3$, 32 elements

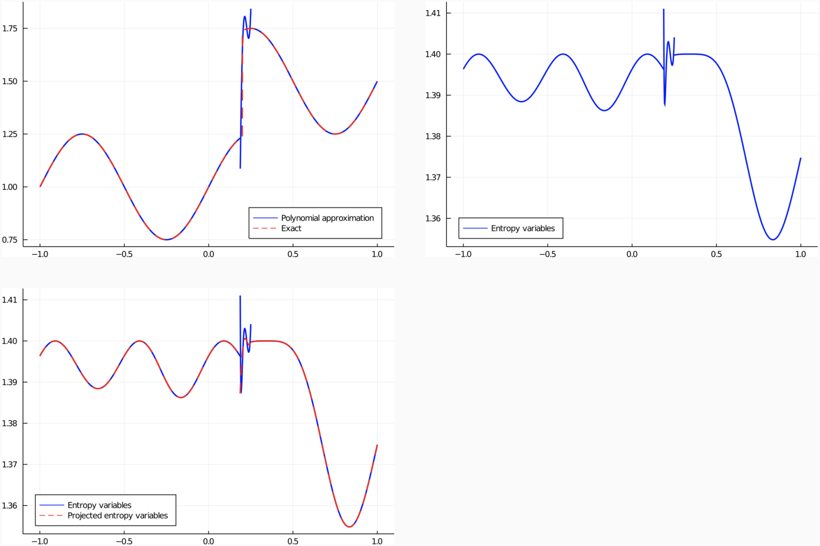
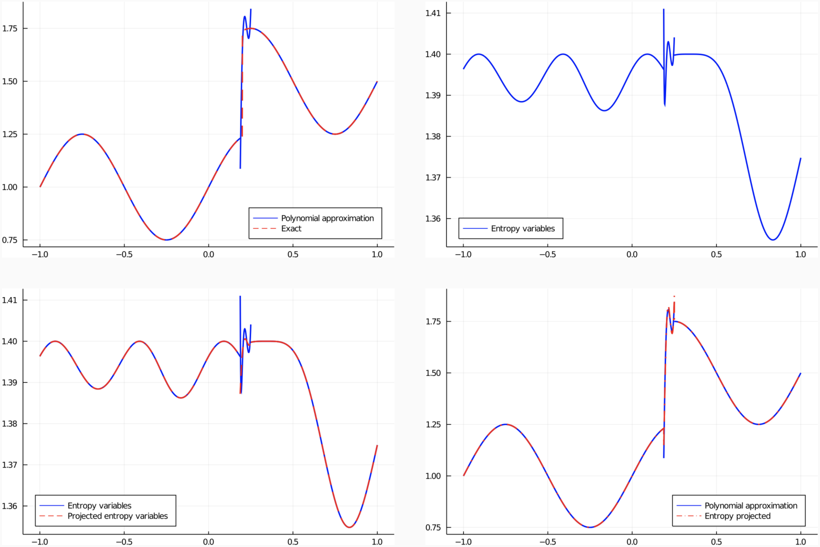
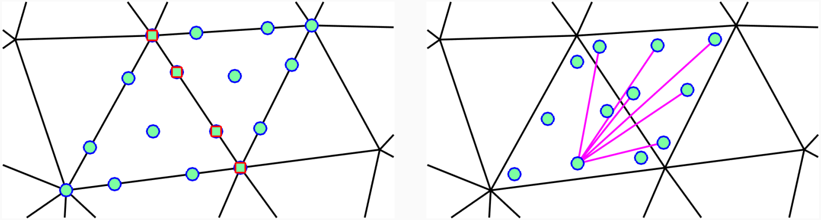


Illustration of entropy projection: $N = 3$, 32 elements



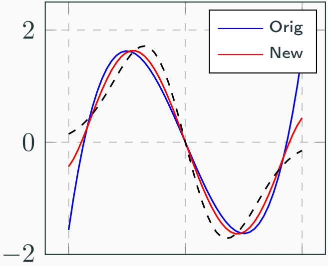
Challenge 2 for modal formulations: interface coupling



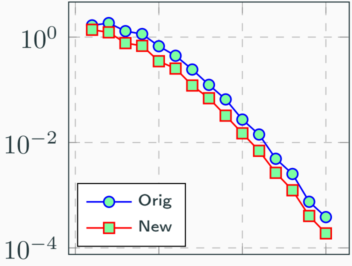
Entropy stable interface coupling with/without boundary nodes

- Interface fluxes must be designed to cancel other boundary terms in the discrete entropy balance.
- Entropy stable interface fluxes previously involved all-to-all coupling between nodes on different elements.

Efficient interface fluxes via "hybridization"



(a) Approximated derivatives

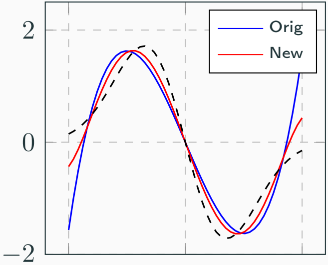


(b) L^2 error, degree $N = 1, \ldots, 15$

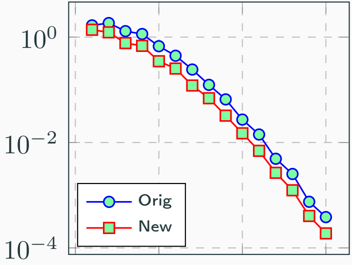
- Avoid coupling by adding correction terms akin to "$\mathbf{E}\boldsymbol{f}(\mathbf{u}) - \boldsymbol{f}(\mathbf{E}\mathbf{u})$", where $\mathbf{E}$ is a face extrapolation matrix.
- Interpret as a Hadamard product + *hybridized* SBP operator.

$$\mathbf{Q}_h = \frac{1}{2}\begin{bmatrix} \mathbf{Q} - \mathbf{Q}^T & \mathbf{E}^T\mathbf{B} \\ -\mathbf{B}\mathbf{E} & \mathbf{B} \end{bmatrix}, \qquad \frac{\partial}{\partial x} \approx \mathbf{M}^{-1}\begin{bmatrix} \mathbf{V}_q \\ \mathbf{V}_f \end{bmatrix}^T \mathbf{Q}_h$$

Efficient interface fluxes via "hybridization"



(a) Approximated derivatives



(b) L^2 error, degree $N = 1, \ldots, 15$

- Avoid coupling by adding correction terms akin to "$\mathbf{E}\boldsymbol{f}(\mathbf{u}) - \boldsymbol{f}(\mathbf{E}\mathbf{u})$", where $\mathbf{E}$ is a face extrapolation matrix.
- Interpret as a Hadamard product + *hybridized* SBP operator.

$$\mathbf{Q}_h = \frac{1}{2}\begin{bmatrix} \mathbf{Q} - \mathbf{Q}^T & \mathbf{E}^T\mathbf{B} \\ -\mathbf{B}\mathbf{E} & \mathbf{B} \end{bmatrix}, \qquad \frac{\partial}{\partial x} \approx \mathbf{M}^{-1}\begin{bmatrix} \mathbf{V}_q \\ \mathbf{V}_f \end{bmatrix}^T \mathbf{Q}_h$$

Entropy stable schemes using hybridized SBP operators

- Replace SBP operator with hybridized SBP operator

$$\mathbf{M}\frac{\mathrm{d}\mathbf{u}}{\mathrm{dt}} + 2\left(\mathbf{Q}\circ\mathbf{F}\right)\mathbf{1} + \mathbf{B}\left(\mathbf{f}^{*} - \boldsymbol{f}(\mathbf{u})\right) = 0.$$

- $\mathbf{F}$ is the matrix of flux evaluations using solution values at *both* volume and face nodes + entropy projection:

$$\mathbf{F}_{ij} = \boldsymbol{f}_S\left(\widetilde{\mathbf{u}}_i, \widetilde{\mathbf{u}}_j\right), \qquad \widetilde{\mathbf{u}} = \text{ evaluate } \boldsymbol{u}\left(\Pi_N \boldsymbol{v}(\mathbf{u})\right).$$

- Entropy stability if $\mathbf{Q}_h\mathbf{1} = \mathbf{0}$ + a weak SBP condition related to quadrature accuracy.

$$\mathbf{Q}+\mathbf{Q}^T = \mathbf{E}^T\mathbf{B}\mathbf{E} \quad \Longrightarrow \quad \mathbf{Q}^T\mathbf{1} = \mathbf{E}^T\mathbf{B}\mathbf{1} \quad \text{(weaker conditions)}$$

Chan (2019). *Skew-symmetric entropy stable modal discontinuous Galerkin formulations*.
Chan (2018). *On discretely entropy conservative and entropy stable discontinuous Galerkin methods.*

Entropy stable schemes using hybridized SBP operators

- Replace SBP operator with hybridized SBP operator

$$\mathbf{M}\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} + 2\begin{bmatrix}\mathbf{V}_q\\ \mathbf{V}_f\end{bmatrix}^T (\mathbf{Q}_h \circ \mathbf{F})\,\mathbf{1} + \mathbf{V}_f^T\mathbf{B}\left(\mathbf{f}^* - \boldsymbol{f}(\mathbf{u})\right) = 0.$$

- $\mathbf{F}$ is the matrix of flux evaluations using solution values at *both* volume and face nodes + entropy projection:

$$\mathbf{F}_{ij} = \boldsymbol{f}_S\left(\widetilde{\mathbf{u}}_i, \widetilde{\mathbf{u}}_j\right), \qquad \widetilde{\mathbf{u}} = \text{ evaluate } \boldsymbol{u}\left(\Pi_N \boldsymbol{v}(\mathbf{u})\right).$$

- Entropy stability if $\mathbf{Q}_h\mathbf{1} = \mathbf{0}$ + a weak SBP condition related to quadrature accuracy.

$$\mathbf{Q}+\mathbf{Q}^T = \mathbf{E}^T\mathbf{B}\mathbf{E} \quad \Longrightarrow \quad \mathbf{Q}^T\mathbf{1} = \mathbf{E}^T\mathbf{B}\mathbf{1} \quad \text{(weaker conditions)}$$

Chan (2019). *Skew-symmetric entropy stable modal discontinuous Galerkin formulations.*

Chan (2018). *On discretely entropy conservative and entropy stable discontinuous Galerkin methods.*

Entropy stable schemes using hybridized SBP operators

- Replace SBP operator with hybridized SBP operator

$$\mathbf{M}\frac{\mathrm{d}\mathbf{u}}{\mathrm{dt}} + 2\begin{bmatrix}\mathbf{V}_q\\ \mathbf{V}_f\end{bmatrix}^T(\mathbf{Q}_h \circ \mathbf{F})\mathbf{1} + \mathbf{V}_f^T\mathbf{B}\left(\mathbf{f}^* - \boldsymbol{f}(\mathbf{u})\right) = 0.$$

- $\mathbf{F}$ is the matrix of flux evaluations using solution values at *both* volume and face nodes + entropy projection:

$$\mathbf{F}_{ij} = \boldsymbol{f}_S\left(\widetilde{\mathbf{u}}_i, \widetilde{\mathbf{u}}_j\right), \qquad \widetilde{\mathbf{u}} = \text{ evaluate } \boldsymbol{u}\left(\Pi_N \boldsymbol{v}(\mathbf{u})\right).$$

- Entropy stability if $\mathbf{Q}_h\mathbf{1} = \mathbf{0}$ + a weak SBP condition related to quadrature accuracy.

$$\mathbf{Q}+\mathbf{Q}^T = \mathbf{E}^T\mathbf{B}\mathbf{E} \quad \Longrightarrow \quad \mathbf{Q}^T\mathbf{1} = \mathbf{E}^T\mathbf{B}\mathbf{1} \quad \text{(weaker conditions)}$$

Chan (2019). *Skew-symmetric entropy stable modal discontinuous Galerkin formulations.*

Chan (2018). *On discretely entropy conservative and entropy stable discontinuous Galerkin methods.*

Entropy stable schemes using hybridized SBP operators

- Replace SBP operator with hybridized SBP operator

$$\mathbf{M}\frac{\mathrm{d}\mathbf{u}}{\mathrm{dt}} + 2\begin{bmatrix}\mathbf{V}_q\\ \mathbf{V}_f\end{bmatrix}^T (\mathbf{Q}_h \circ \mathbf{F})\,\mathbf{1} + \mathbf{V}_f^T\mathbf{B}\left(\mathbf{f}^* - \boldsymbol{f}(\mathbf{u})\right) = 0.$$

- $\mathbf{F}$ is the matrix of flux evaluations using solution values at *both* volume and face nodes + entropy projection:

$$\mathbf{F}_{ij} = \boldsymbol{f}_S\left(\widetilde{\mathbf{u}}_i, \widetilde{\mathbf{u}}_j\right), \qquad \widetilde{\mathbf{u}} = \text{ evaluate } \boldsymbol{u}\left(\Pi_N \boldsymbol{v}(\mathbf{u})\right).$$

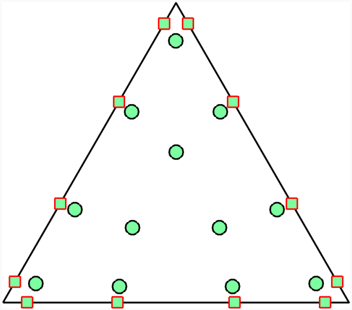
- Entropy stability if $\mathbf{Q}_h\mathbf{1} = \mathbf{0}$ + a weak SBP condition related to quadrature accuracy.

$$\mathbf{Q}+\mathbf{Q}^T = \mathbf{E}^T\mathbf{B}\mathbf{E} \quad \Longrightarrow \quad \mathbf{Q}^T\mathbf{1} = \mathbf{E}^T\mathbf{B}\mathbf{1} \quad \text{(weaker conditions)}$$

Chan (2019). *Skew-symmetric entropy stable modal discontinuous Galerkin formulations.*
Chan (2018). *On discretely entropy conservative and entropy stable discontinuous Galerkin methods.*

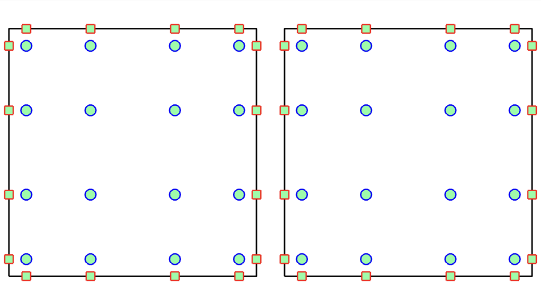
Example: triangular and tetrahedral meshes

- Degree N polynomial approximation + degree $\geq 2N$ volume/face quadratures.
- Uniform 32×32 mesh: degree $N = 3$, CFL $.125$, Lax-Friedrichs flux penalization.



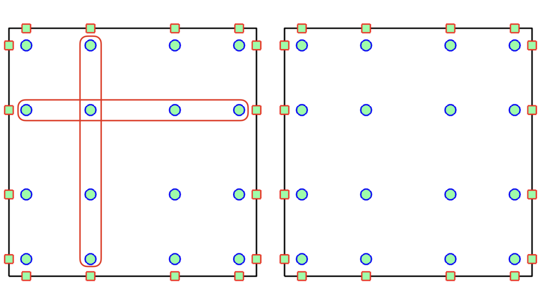
Results computed on larger periodic domain ("natural" boundary conditions unstable).

Example: entropy stable Gauss collocation on quad/hex meshes (with MH Carpenter + DCDR Fernandez)



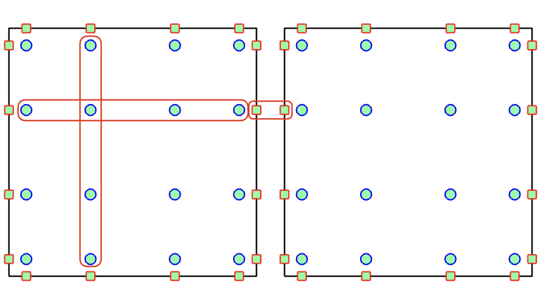
- Hex or quad elements: tensor product polynomial basis
- Tensor product $(N+1)$-point Gauss quadrature for integrals.
- Simplifies to a collocation scheme, Kronecker product reduces flux evaluations from $O(N^6)$ to $O(N^4)$ in 3D.

Example: entropy stable Gauss collocation on quad/hex meshes (with MH Carpenter + DCDR Fernandez)



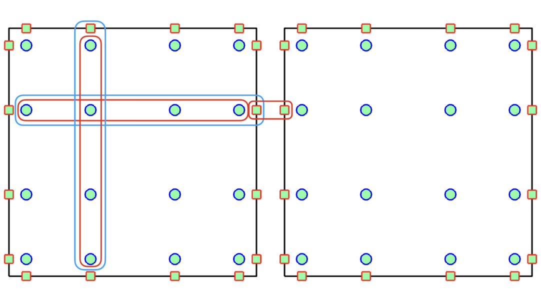
- Hex or quad elements: tensor product polynomial basis
- Tensor product $(N+1)$-point Gauss quadrature for integrals.
- Simplifies to a collocation scheme, Kronecker product reduces flux evaluations from $O(N^6)$ to $O(N^4)$ in 3D.

Example: entropy stable Gauss collocation on quad/hex meshes (with MH Carpenter + DCDR Fernandez)

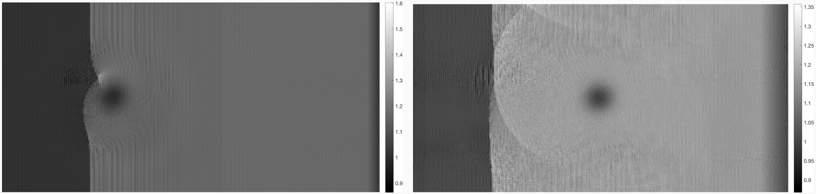


- Hex or quad elements: tensor product polynomial basis
- Tensor product $(N+1)$-point Gauss quadrature for integrals.
- Simplifies to a collocation scheme, Kronecker product reduces flux evaluations from $O(N^6)$ to $O(N^4)$ in 3D.

Example: entropy stable Gauss collocation on quad/hex meshes (with MH Carpenter + DCDR Fernandez)



- Hex or quad elements: tensor product polynomial basis
- Tensor product $(N+1)$-point Gauss quadrature for integrals.
- Simplifies to a collocation scheme, Kronecker product reduces flux evaluations from $O(N^6)$ to $O(N^4)$ in 3D.



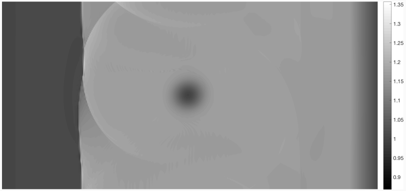
(a) Entropy conservative flux, $T = .3$ **(b)** Entropy conservative flux, $T = .7$

Figure 1: Shock vortex interaction problem using high order entropy stable Gauss collocation schemes with $N = 4, h = 1/100$.

Winters, Derigs, Gassner, and Walch (2017). *A uniquely defined entropy stable matrix dissipation operator for high Mach number ideal MHD and compressible Euler simulations.*

Shock vortex interaction

(a) With entropy dissipation, $T = .3$

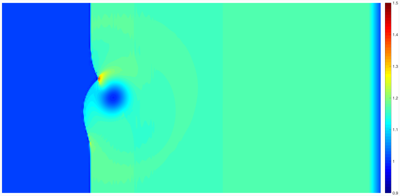


(b) With entropy dissipation, $T = .7$

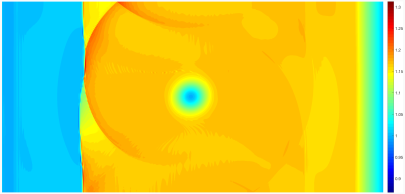
Figure 1: Shock vortex interaction problem using high order entropy stable Gauss collocation schemes with $N = 4, h = 1/100$.

Winters, Derigs, Gassner, and Walch (2017). *A uniquely defined entropy stable matrix dissipation operator for high Mach number ideal MHD and compressible Euler simulations.*

Shock vortex interaction



(a) With entropy dissipation, $T = .3$

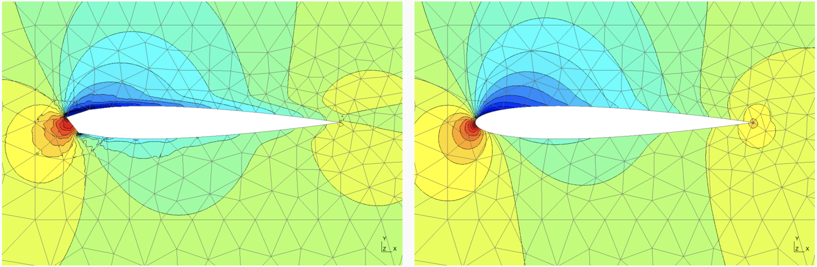


(b) With entropy dissipation, $T = .7$

Figure 1: Shock vortex interaction problem using high order entropy stable Gauss collocation schemes with $N = 4, h = 1/100$.

Winters, Derigs, Gassner, and Walch (2017). *A uniquely defined entropy stable matrix dissipation operator for high Mach number ideal MHD and compressible Euler simulations.*

Curved meshes are required for high order accuracy

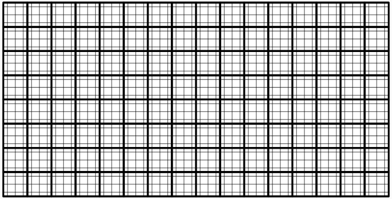


(a) Straight-sided mesh **(b)** Curved mesh

High order numerical simulations using straight-sided and curved geometry representations.

Remacle, Lambrechts, Geuzaine, Toulorge (2014). *Optimizing the geometrical accuracy of 2D curvilinear meshes.*

Gauss quadrature improves errors on curved meshes



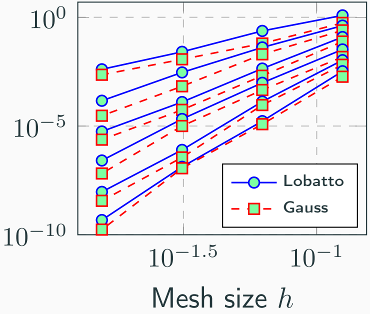
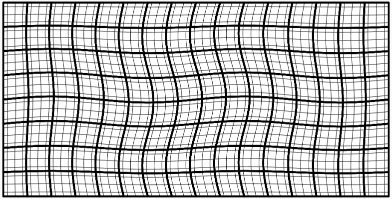


Figure 2: L^2 errors for 2D isentropic vortex at time $T = 5$ for degree $N = 2, \ldots, 7$ Lobatto and Gauss collocation schemes.

Gauss quadrature improves errors on curved meshes



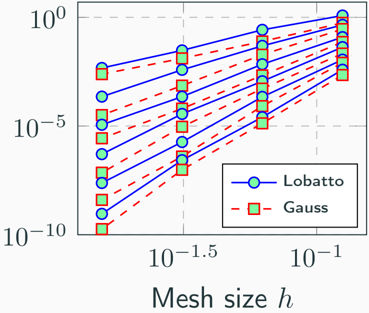
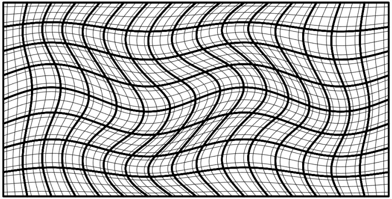


Figure 2: L^2 errors for 2D isentropic vortex at time $T = 5$ for degree $N = 2, \ldots, 7$ Lobatto and Gauss collocation schemes.

Gauss quadrature improves errors on curved meshes



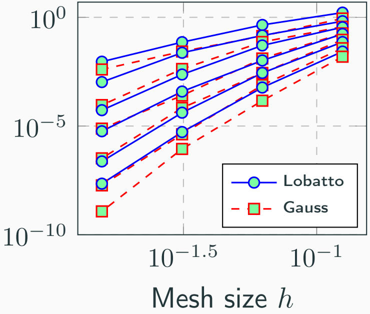
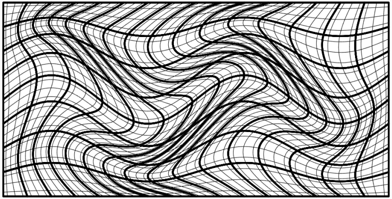


Figure 2: L^2 errors for 2D isentropic vortex at time $T = 5$ for degree $N = 2, \ldots, 7$ Lobatto and Gauss collocation schemes.

Gauss quadrature improves errors on curved meshes



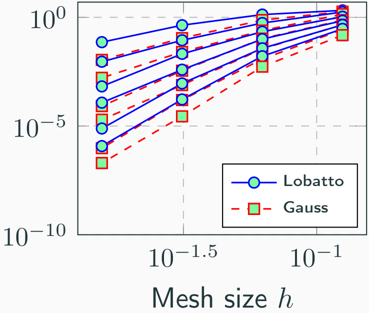


Figure 2: L^2 errors for 2D isentropic vortex at time $T = 5$ for degree $N = 2, \ldots, 7$ Lobatto and Gauss collocation schemes.

Applications of modal DG formulations

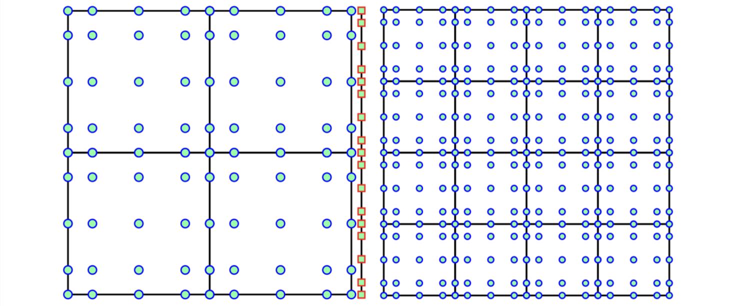
- Tri and tet meshes (Chan 2018 + Chan, Wilcox 2019)
- Collocation methods on quad/hex meshes (Chan, Fernandez, Carpenter 2019)
- Hybrid meshes (Chan 2019)
- Shallow water (Wu, Kubatko, Chan 2019 + Wu, Chan 2020)
- Reduced order modeling (Chan 2020)
- Non-conforming meshes (Chan, Bencomo, Fernandez 2020)
- Jacobian matrices, time-implicit solvers (Chan, Taylor 2020)
- Viscous compressible flow (Chan, Lin, Warburton 2020)

Some recent work

Some recent work

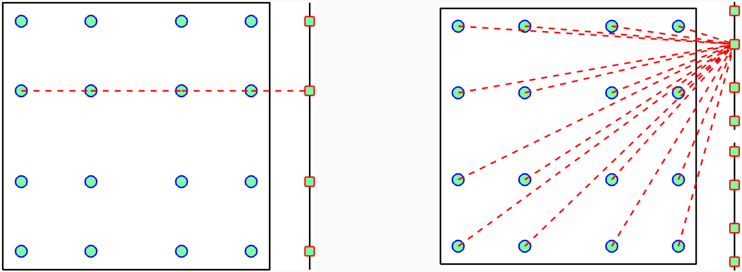
Non-conforming meshes (with M. Bencomo, D. Del Rey Fernandez)

Non-conforming meshes



- Volume and surface nodes coupled thru $f_S(\mathbf{u}_i, \mathbf{u}_j)$ and stencil of interpolation operator $\mathbf{E}$.
- Fix: use a mortar for non-conforming couplings.

Non-conforming meshes

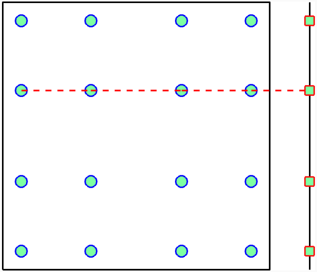


(a) Conforming surface nodes

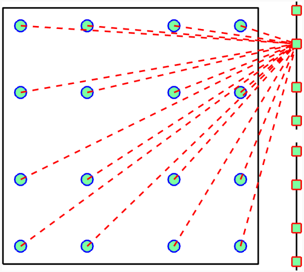
(b) Non-conforming nodes

- Volume and surface nodes coupled thru $\boldsymbol{f}_S(\mathbf{u}_i, \mathbf{u}_j)$ and stencil of interpolation operator $\mathbf{E}$.
- Fix: use a mortar for non-conforming couplings.

Non-conforming meshes



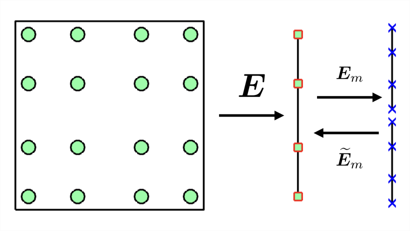
(a) Conforming surface nodes



(b) Non-conforming nodes

- Volume and surface nodes coupled thru $\boldsymbol{f}_S(\mathbf{u}_i, \mathbf{u}_j)$ and stencil of interpolation operator **E**.
- Fix: use a mortar for non-conforming couplings.

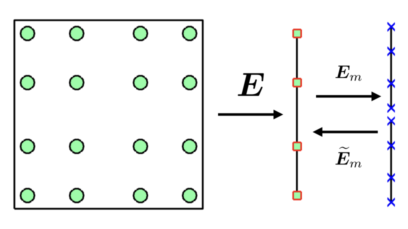
A mortar-based hybridized SBP operator



- Define transfer operators $\mathbf{E}_m, \widetilde{\mathbf{E}}_m$ between conforming and non-conforming (mortar) nodes.
- Modify the hybridized SBP volume term:

$$\sum_{i=1}^{d} \begin{bmatrix} \mathbf{I} \\ \mathbf{E} \end{bmatrix}^T \left(\begin{bmatrix} \mathbf{Q}_i - \mathbf{Q}_i^T & \mathbf{E}^T \mathbf{B}_i \\ -\mathbf{B}_i \mathbf{E} & \mathbf{B}_i \end{bmatrix} \circ \mathbf{F}_i \right) \mathbf{1}$$

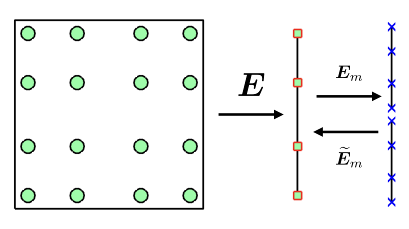
A mortar-based hybridized SBP operator



- Define transfer operators $\mathbf{E}_m, \widetilde{\mathbf{E}}_m$ between conforming and non-conforming (mortar) nodes.
- Modify the hybridized SBP volume term:

$$\sum_{i=1}^{d} \begin{bmatrix} \mathbf{I} \\ \mathbf{E} \end{bmatrix}^T \left(\begin{bmatrix} \mathbf{Q}_i - \mathbf{Q}_i^T & \mathbf{E}^T \mathbf{B}_i \\ -\mathbf{B}_i \mathbf{E} & \mathbf{B}_i \end{bmatrix} \circ \mathbf{F}_i \right) \mathbf{1}$$

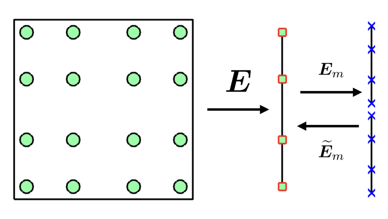
A mortar-based hybridized SBP operator



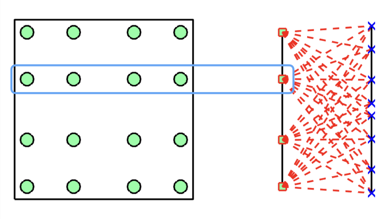
- Define transfer operators $\mathbf{E}_m, \widetilde{\mathbf{E}}_m$ between conforming and non-conforming (mortar) nodes.
- Modify the hybridized SBP volume term:

$$\sum_{i=1}^{d} \begin{bmatrix} \mathbf{I} \\ \mathbf{E} \\ \mathbf{E}_m \mathbf{E} \end{bmatrix}^T \left(\begin{bmatrix} \mathbf{Q}_i - \mathbf{Q}_i^T & \mathbf{E}^T \mathbf{B}_i & \\ -\mathbf{B}_i \mathbf{E} & & \mathbf{B}_i \widetilde{\mathbf{E}}_m \\ & -\widetilde{\mathbf{B}}_i \mathbf{E}_m & \widetilde{\mathbf{B}}_i \end{bmatrix} \circ \mathbf{F}_i \right) \mathbf{1}$$

An efficient mortar reformulation



(a) Mortar operators



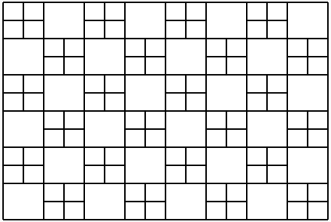
(b) Volume/surface/mortar coupling

$$\mathbf{M}\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} + \sum_{i=1}^{d} \begin{bmatrix} \mathbf{I} \\ \mathbf{E} \end{bmatrix}^T \left(2\mathbf{Q}_h^i \circ \mathbf{F}_i\right)\mathbf{1} + \mathbf{E}^T\mathbf{B}_i\widetilde{\mathbf{f}}_i^* = 0$$

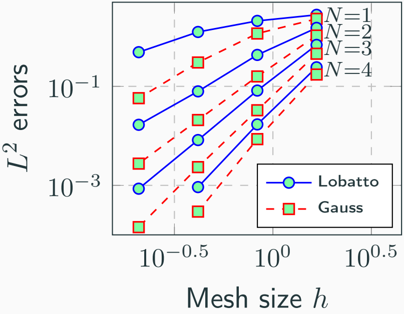
$$\widetilde{\mathbf{f}}_i^* = \widetilde{\mathbf{E}}_m\left(\mathbf{f}_i^* - \boldsymbol{f}_i(\mathbf{u})\right) + \left(\widetilde{\mathbf{E}}_m \circ \mathbf{F}_{i,sm}\right)\mathbf{1} - \widetilde{\mathbf{E}}_m\left(\mathbf{E}_m \circ \mathbf{F}_{i,ms}\right)\mathbf{1}$$

Reformulate as an entropy stable correction to the numerical flux.

Numerical results: non-conforming meshes



(a) Non-conforming mesh



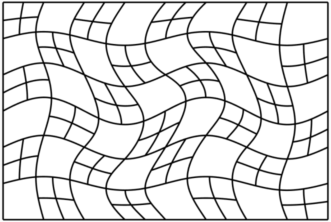
(b) L^2 errors

Convergence rate is lower if under-integrated: Lobatto rates are $O(h^N)$ while Gauss rates are $O(h^{N+1})$.

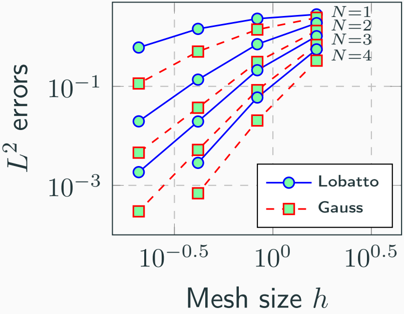
Chan (2019). *Skew-symmetric entropy stable modal discontinuous Galerkin formulations.*

Chan, Bencomo, Del Rey Fernandez (2020). *Mortar-based entropy stable discontinuous Galerkin methods on non-conforming quadrilateral and hexahedral meshes.*

Numerical results: non-conforming meshes



(a) Non-conforming mesh



(b) L^2 errors

Convergence rate is lower if under-integrated: Lobatto rates are $O(h^N)$ while Gauss rates are $O(h^{N+1})$.

Chan (2019). *Skew-symmetric entropy stable modal discontinuous Galerkin formulations.*

Chan, Bencomo, Del Rey Fernandez (2020). *Mortar-based entropy stable discontinuous Galerkin methods on non-conforming quadrilateral and hexahedral meshes.*

Some recent work

Efficient computation of Jacobian matrices (with C. Taylor)

Current methods for computing Jacobian matrices

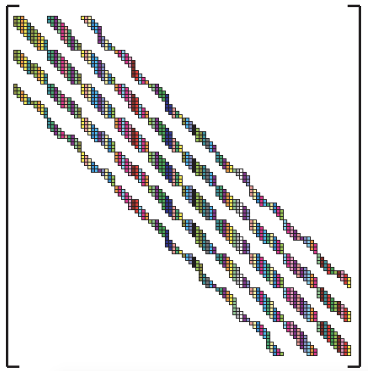


Figure from Gebremedhin, Manne, Pothen (2005), *What color is your Jacobian? Graph coloring for computing derivatives.*

- Compute entries using automatic differentiation (AD)
- Graph coloring reduces AD costs, but only for sparse matrices
- In general, cost of AD scales with input and output dimensions.

Jacobian matrices for flux differencing

Hadamard product structure yields simple Jacobians.

Theorem

Assume $\mathbf{Q} = \pm\mathbf{Q}^T$. *Consider a scalar "collocation" discretization*

$$\mathbf{r}(\mathbf{u}) = (\mathbf{Q} \circ \mathbf{F})\,\mathbf{1}, \qquad \mathbf{F}_{ij} = f_S(\mathbf{u}_i, \mathbf{u}_j).$$

The Jacobian matrix is then

$$\frac{\mathrm{d}\mathbf{r}}{\mathrm{d}\mathbf{u}} = (\mathbf{Q} \circ \partial\mathbf{F}_R) \pm \operatorname{diag}\left(\mathbf{1}^T\left(\mathbf{Q} \circ \partial\mathbf{F}_R\right)\right),$$

$$(\partial\mathbf{F}_R)_{ij} = \left.\frac{\partial f_S(u_L, u_R)}{\partial u_R}\right|_{\mathbf{u}_i,\mathbf{u}_j}.$$

Observations about flux differencing Jacobian formulas

Separates "template" matrix Q and flux contributions.

$$\frac{\mathrm{d}\mathbf{r}}{\mathrm{d}\mathbf{u}} = (\mathbf{Q} \circ \partial \mathbf{F}_R) \pm \operatorname{diag}\left(\mathbf{1}^T \left(\mathbf{Q} \circ \partial \mathbf{F}_R\right)\right),$$

$$(\partial \mathbf{F}_R)_{ij} = \left.\frac{\partial f_S(u_L, u_R)}{\partial u_R}\right|_{\mathbf{u}_i, \mathbf{u}_j}.$$

$O(1)$ inputs/outputs $\rightarrow$ AD is efficient. In Julia:

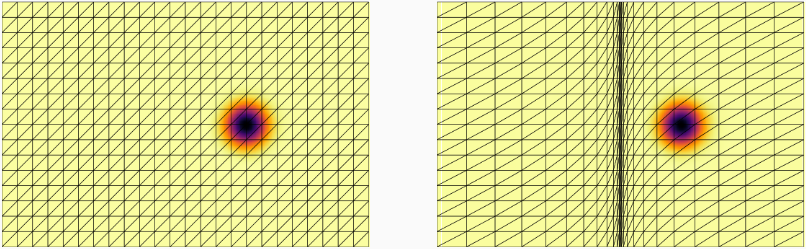
```
using ForwardDiff
f(uL,uR) = (1/6)*(uL^2 + uL*uR + uR^2)
dF(uL,uR) = ForwardDiff.derivative(uR->f(uL,uR),uR)
```

Computational timings

Jacobian timings for $f_S(u_L, u_R) = \frac{1}{6}\left(u_L^2 + u_L u_R + u_R^2\right)$ and dense differentiation matrices $\mathbf{Q} \in \mathbb{R}^{N\times N}$.

	N = 10	N = 25	N = 50
Direct automatic differentiation	5.666	60.388	373.633
`FiniteDiff.jl`	1.429	17.324	125.894
Jacobian formula (analytic deriv.)	.209	1.005	3.249
Jacobian formula (AD flux deriv.)	.210	1.030	3.259
Evaluation of $\mathbf{f}(\mathbf{u})$ (reference)	.120	.623	2.403

Implicit midpoint method for compressible Euler



(a) Uniform, L^2 error $.0901$ **(b)** Anisotropic, L^2 error $.0935$

Figure 3: Solutions for a degree $N = 3$ modal DG method with $dt = .1$ on uniform and "squeezed" meshes.

Some recent work

Compressible Navier-Stokes (with Y. Lin, T. Warburton)

Compressible Navier-Stokes: discretization of viscous terms

Compressible Navier-Stokes equations: inviscid fluxes $\boldsymbol{f}_i(\boldsymbol{u})$ and viscous fluxes $\boldsymbol{g}_i(\boldsymbol{u})$

$$\frac{\partial \boldsymbol{u}}{\partial t} + \sum_{i=1}^{d} \frac{\partial \boldsymbol{f}_i(\boldsymbol{u})}{\partial x_i} = \sum_{i=1}^{d} \frac{\partial \boldsymbol{g}_i(\boldsymbol{u})}{\partial x_i}.$$

Symmetrize viscous terms by transforming to entropy variables $\boldsymbol{v}(\boldsymbol{u})$

$$\sum_{i=1}^{d} \frac{\partial \boldsymbol{g}_i(\boldsymbol{u})}{\partial x_i} = \sum_{i,j=1}^{d} \frac{\partial}{\partial x_i} \left(\boldsymbol{K}_{ij}(\boldsymbol{v}) \frac{\partial \boldsymbol{v}}{\partial x_j} \right), \qquad \boldsymbol{K}_{ij} \succeq \boldsymbol{0}.$$

Hughes, Franca, Mallet (1986). *A new finite element formulation for CFD: I. Symmetric forms of the compressible Euler and Navier-Stokes equations and the second law of thermodynamics.*

DG formulation and boundary conditions

Write viscous terms as a first order system

$$\boldsymbol{\Theta}_i = \frac{\partial \boldsymbol{v}}{\partial x_i}$$

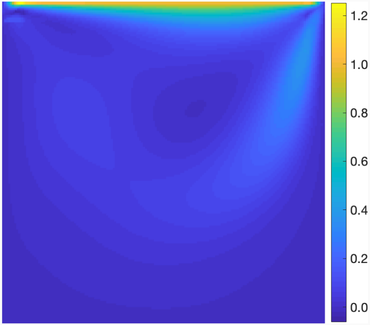
$$\boldsymbol{\sigma}_i = \sum_{j=1}^{d} \boldsymbol{K}_{ij}(\boldsymbol{v})\boldsymbol{\Theta}_j$$

$$\boldsymbol{g}_{\text{visc}} = \sum_{i=1}^{d} \frac{\partial \boldsymbol{\sigma}_i}{\partial x_i}$$

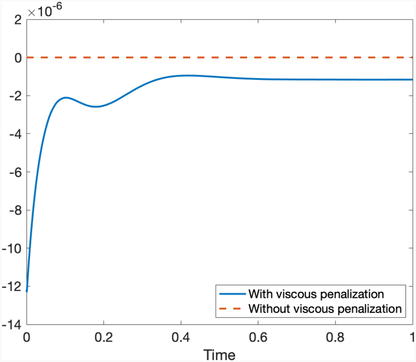
Entropy dissipative if discretized with standard DG techniques and we

- impose BCs on $\boldsymbol{u}$, entropy variables $\boldsymbol{v}$, and $\boldsymbol{\sigma}$.
- get exactly entropy conservative BCs for no-slip adiabatic and symmetry walls, entropy *mimetic* for no-slip isothermal.

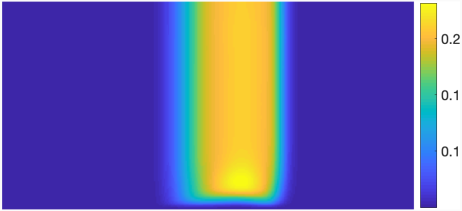
Verification of entropy conservation/dissipation



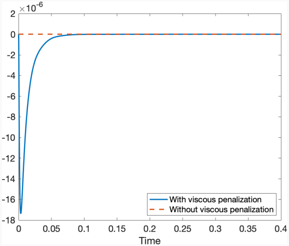
(a) Adiabatic lid-driven cavity, $\mathrm{Ma} = .1$, $\mathrm{Re} = 1000$



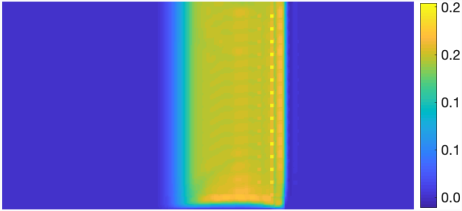
(b) Viscous entropy dissipation



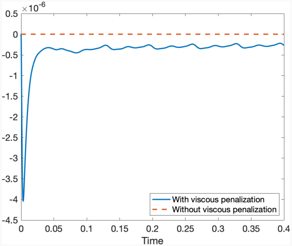
(c) Wall/sym. BCs, $\mathrm{Ma} = 1.5$, $\mathrm{Re} = 100$



(d) Entropy dissipation

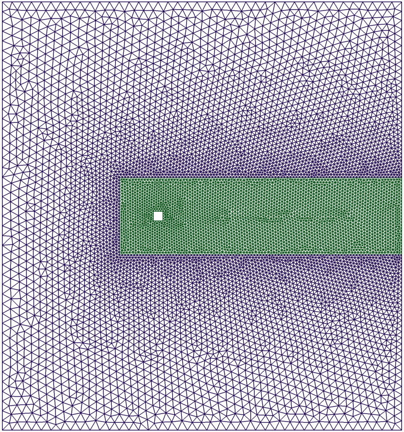


(e) Wall/sym. BCs, $\mathrm{Ma} = 1.5$, $\mathrm{Re} = 1000$

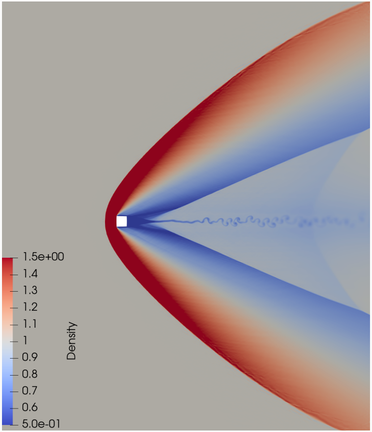


(f) Entropy dissipation

Flow over a square cylinder



(a) Mesh



(b) Zoom of ρ at $T_{\text{final}} = 100$

Figure 4: Mesh and density ρ at $T_{\text{final}} = 100$ for $\mathrm{Re} = 10^4$, $\mathrm{Ma} = 1.5$, and a degree $N = 3$ approximation.

Summary and future work

This work is supported by the NSF under awards DMS-1719818, DMS-1712639, and DMS-CAREER-1943186.

Thank you! Questions?

Chan, Lin, Warburton (2020). *Entropy stable modal discontinuous Galerkin schemes and wall boundary conditions for the compressible Navier-Stokes equation*.

Chan, Taylor (2020). *Efficient computation of Jacobian matrices for ES SBP schemes*.

Chan, Bencomo, Del Rey Fernandez (2020). *Mortar-based entropy-stable discontinuous Galerkin methods on non-conforming quadrilateral and hexahedral meshes*.

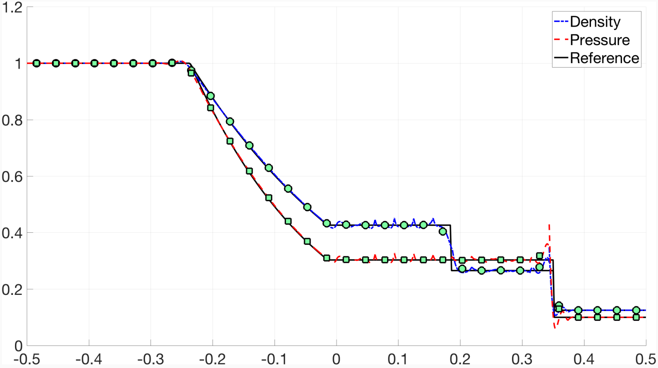
Chan, Del Rey Fernandez, Carpenter (2018). *Efficient entropy stable Gauss collocation methods*.

Chan (2018). *On discretely entropy conservative and entropy stable discontinuous Galerkin methods*.

Additional slides

1D Sod shock tube

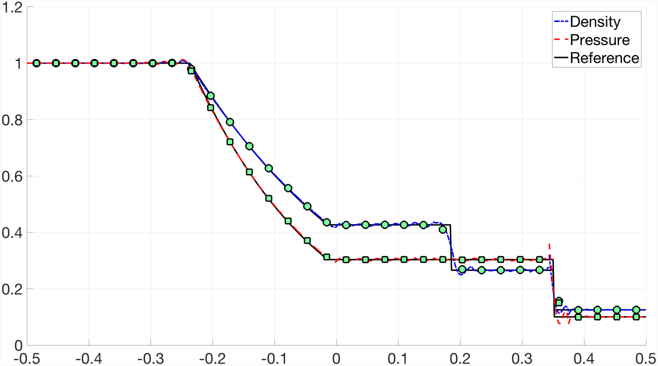
- Circles are cell averages, CFL of .125, LSRK-45 time-stepping.
- Comparison between $(N+1)$-point Lobatto and $(N+2)$-point Gauss.



$N = 4, K = 32$, $(N+1)$ point Lobatto quadrature.

1D Sod shock tube

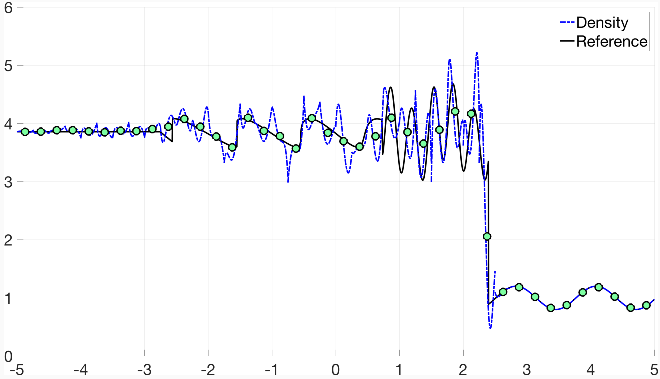
- Circles are cell averages, CFL of .125, LSRK-45 time-stepping.
- Comparison between $(N+1)$-point Lobatto and $(N+2)$-point Gauss.



$N = 4, K = 32$, $(N+2)$ point Gauss quadrature.

1D sine-shock interaction

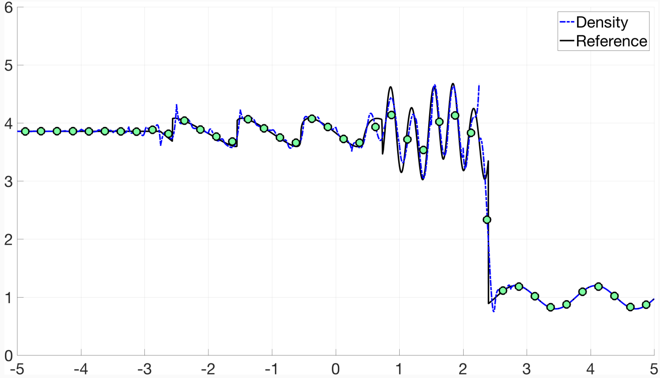
- $(N+2)$-point Gauss, smaller CFL (.05 vs .125) for stability.



$N = 4, K = 40, CFL = .05$, $(N+1)$ point Lobatto quadrature.

1D sine-shock interaction

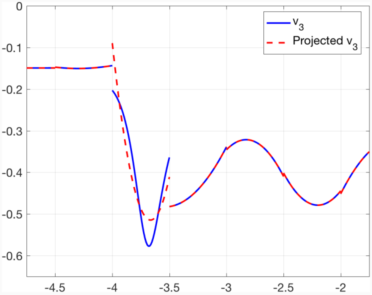
- $(N+2)$-point Gauss, smaller CFL (.05 vs .125) for stability.



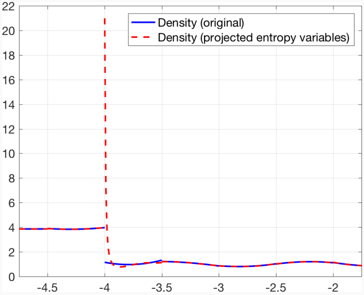
$N = 4, K = 40, CFL = .05$, $(N+2)$ point Gauss quadrature.

Loss of control with the entropy projection

- For $(N+1)$-point Lobatto, $\widetilde{\boldsymbol{u}} = \boldsymbol{u}$ at nodal points.
- For $(N+2)$-point Gauss, discrepancy between $\boldsymbol{v}(\boldsymbol{u})$ and projection on the boundary of elements.
- Still need positivity of thermodynamic quantities for stability!



(a) $v_3(x), (\Pi_N v_3)(x)$



(b) $\rho(x), \rho((\Pi_N \boldsymbol{v})(x))$

Taylor-Green vortex

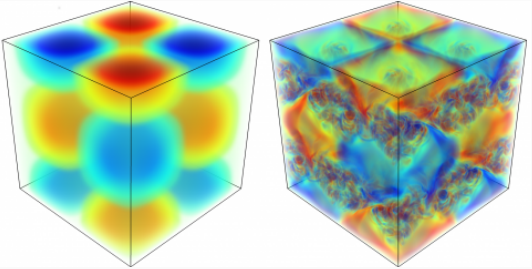
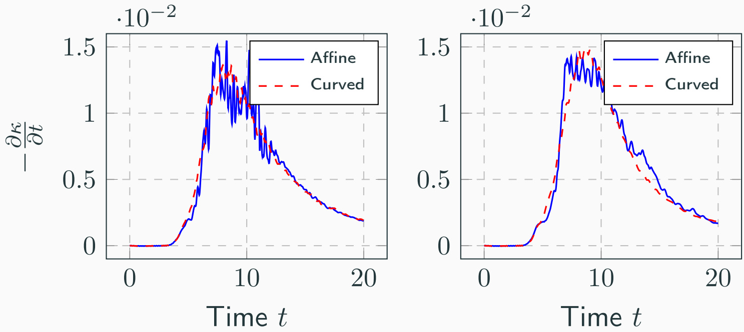


Figure 5: Isocontours of z-vorticity for Taylor-Green at $t = 0, 10$ seconds.

- Simple turbulence-like behavior (generation of small scales).
- Inviscid Taylor-Green: tests robustness w.r.t. under-resolved solutions.

https://how4.cenaero.be/content/bs1-dns-taylor-green-vortex-re1600.

3D inviscid Taylor-Green vortex



(a) GLL collocation **(b)** Gauss collocation

Figure 6: Kinetic energy dissipation rate for entropy stable GLL and Gauss collocation schemes with $N = 7$ and $h = \pi/8$.