Entropy stable schemes for the compressible Navier-Stokes equations: boundary conditions and positivity preserving schemes

Yimin Lin, Jesse Chan, Tim Warburton

USNCCM 16, July 2021

Department of Computational and Applied Mathematics, Rice University
Department of Mathematics, Virginia Tech

Compressible Navier-Stokes equations

- Compressible Navier-Stokes equations

$$\frac{\partial \boldsymbol{U}}{\partial t} + \underbrace{\sum_{i=1}^{3} \frac{\partial \boldsymbol{f}_i(\boldsymbol{U})}{\partial \boldsymbol{x}_i}}_{\text{inviscid flux}} = \underbrace{\sum_{i=1}^{3} \frac{\partial \boldsymbol{g}_i(\boldsymbol{U})}{\partial \boldsymbol{x}_i}}_{\text{viscous flux}}$$

- Entropy variables symmetrizes the viscous fluxes:

$$\sum_{i=1}^{d} \frac{\partial \boldsymbol{g}_i}{\partial x_i} = \sum_{i,j=1}^{d} \frac{\partial}{\partial x_i} \left(K_{ij} \frac{\partial \boldsymbol{v}}{\partial x_j} \right),$$

$$K = \begin{bmatrix} K_{11} & \dots & K_{1d} \\ \vdots & \ddots & \vdots \\ K_{d1} & \dots & K_{dd} \end{bmatrix} = K^T, \qquad K \succeq 0$$

Continuous Entropy Balance

- With convex entropy η, entropy variable $\boldsymbol{v} = \frac{\partial \eta(\boldsymbol{u})}{\partial \boldsymbol{u}}$ and entropy potential ψ_i. We can derive an entropy balance

Continuous Entropy Balance

- With convex entropy η, entropy variable $\boldsymbol{v} = \frac{\partial \eta(\boldsymbol{u})}{\partial \boldsymbol{u}}$ and entropy potential ψ_i. We can derive an entropy balance

$$\int_\Omega \boldsymbol{v}^T \frac{\partial \boldsymbol{u}}{\partial t} + \sum_{i=1}^{d} \int_\Omega \boldsymbol{v}^T \frac{\partial \boldsymbol{f}_i(\boldsymbol{u})}{\partial x_i} = \sum_{i=1}^{d} \int_\Omega \boldsymbol{v}^T \frac{\partial \boldsymbol{g}_i(\boldsymbol{u})}{\partial x_i} \qquad \text{Test by } \boldsymbol{v}$$

Continuous Entropy Balance

- With convex entropy η, entropy variable $\boldsymbol{v} = \frac{\partial \eta(\boldsymbol{u})}{\partial \boldsymbol{u}}$ and entropy potential ψ_i. We can derive an entropy balance

$$\int_\Omega \boldsymbol{v}^T \frac{\partial \boldsymbol{u}}{\partial t} + \sum_{i=1}^{d} \int_\Omega \boldsymbol{v}^T \frac{\partial \boldsymbol{f}_i(\boldsymbol{u})}{\partial x_i} = \sum_{i=1}^{d} \int_\Omega \boldsymbol{v}^T \frac{\partial \boldsymbol{g}_i(\boldsymbol{u})}{\partial x_i} \qquad \text{Test by } \boldsymbol{v}$$

$$\int_\Omega \frac{\partial \eta(\boldsymbol{u})}{\partial t} + \sum_{i=1}^{d} \int_{\partial\Omega} n_i \left(F_i(\boldsymbol{u}) - \frac{1}{c_v T} \kappa \frac{\partial T}{\partial x_i} \right) = - \int_\Omega \sum_{i,j=1}^{d} \left(\frac{\partial \boldsymbol{v}}{\partial x_i} \right)^T \left(K_{ij} \frac{\partial \boldsymbol{v}}{\partial x_j} \right)$$

Integration by parts and chain rule

Continuous Entropy Balance

- With convex entropy η, entropy variable $\boldsymbol{v} = \frac{\partial \eta(\boldsymbol{u})}{\partial \boldsymbol{u}}$ and entropy potential ψ_i. We can derive an entropy balance

$$\int_\Omega \boldsymbol{v}^T \frac{\partial \boldsymbol{u}}{\partial t} + \sum_{i=1}^d \int_\Omega \boldsymbol{v}^T \frac{\partial \boldsymbol{f}_i(\boldsymbol{u})}{\partial x_i} = \sum_{i=1}^d \int_\Omega \boldsymbol{v}^T \frac{\partial \boldsymbol{g}_i(\boldsymbol{u})}{\partial x_i} \qquad \text{Test by } \boldsymbol{v}$$

$$\int_\Omega \frac{\partial \eta(\boldsymbol{u})}{\partial t} + \sum_{i=1}^d \int_{\partial\Omega} n_i \left(F_i(\boldsymbol{u}) - \frac{1}{c_v T} \kappa \frac{\partial T}{\partial x_i} \right) = - \int_\Omega \sum_{i,j=1}^d \left(\frac{\partial \boldsymbol{v}}{\partial x_i} \right)^T \left(K_{ij} \frac{\partial \boldsymbol{v}}{\partial x_j} \right)$$

Integration by parts and chain rule

$$\int_\Omega \frac{\partial \eta(\boldsymbol{u})}{\partial t} = - \int_\Omega \sum_{i,j=1}^d \left(\frac{\partial \boldsymbol{v}}{\partial x_i} \right)^T \left(K_{ij} \frac{\partial \boldsymbol{v}}{\partial x_j} \right) \leq 0 \qquad \text{Periodic}$$

Discretization of the inviscid term - Nodal ESDG

- Entropy conservative numerical flux

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

$$(\boldsymbol{v}_L - \boldsymbol{v}_R)^T \boldsymbol{f}_S(\boldsymbol{u}_L, \boldsymbol{u}_R) = \psi(\boldsymbol{u}_L) - \psi(\boldsymbol{u}_R)$$

Discretization of the inviscid term - Nodal ESDG

- Entropy conservative numerical flux

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

$$(\boldsymbol{v}_L - \boldsymbol{v}_R)^T \boldsymbol{f}_S(\boldsymbol{u}_L,\boldsymbol{u}_R) = \psi(\boldsymbol{u}_L) - \psi(\boldsymbol{u}_R)$$

- Flux differencing technique

$$\frac{\partial \boldsymbol{f}(\boldsymbol{u}(\boldsymbol{x}))}{\partial x} = 2\, \frac{\partial \boldsymbol{f}_S(\boldsymbol{u}(\boldsymbol{x}),\boldsymbol{u}(\boldsymbol{y}))}{\partial x}\bigg|_{y=x}$$

Discretization of the inviscid term - Nodal ESDG

- Entropy conservative numerical flux

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

$$(\boldsymbol{v}_L - \boldsymbol{v}_R)^T \boldsymbol{f}_S(\boldsymbol{u}_L, \boldsymbol{u}_R) = \psi(\boldsymbol{u}_L) - \psi(\boldsymbol{u}_R)$$

- Flux differencing technique

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

- Collocation on Lobatto quadrature nodes gives summation-by-parts (SBP) operator

$$\mathbf{Q} = \mathbf{M}\mathbf{D}, \qquad \mathbf{Q} + \mathbf{Q}^T = \mathbf{B}, \qquad \mathbf{Q}\mathbf{1} = 0$$

Discretization of the inviscid term - Nodal ESDG

- Entropy conservative numerical flux

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

- Flux differencing technique

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

- Collocation on Lobatto quadrature nodes gives summation-by-parts (SBP) operator

$$\mathsf{Q} = \mathsf{MD}, \qquad \mathsf{Q} + \mathsf{Q}^T = \mathsf{B}, \qquad \mathsf{Q}\mathbf{1} = 0$$

- Discretize the variational form

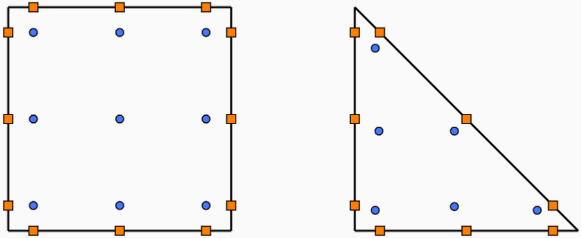
$$\int_{\widehat{D}} \frac{\partial \boldsymbol{f}}{\partial x}\vec{l} \quad \xRightarrow{\text{Discretize}} \quad 2(\mathsf{Q}\circ\mathsf{F}_S)\mathbf{1}, \quad (\mathsf{F}_S)_{ij} = \boldsymbol{f}_S(\boldsymbol{u}_i,\boldsymbol{u}_j)$$

Discretization of the inviscid term - Modal ESDG

- Modal ESDG allows arbitrary choice of approximation basis and quadrature rules

Discretization of the inviscid term - Modal ESDG

- Modal ESDG allows arbitrary choice of approximation basis and quadrature rules
- Extends nodal ESDG via over-integration



Discretization of the inviscid term - Modal ESDG

- "Hyrbridized" SBP operator

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

$$\int_{\widehat{D}} u\frac{\partial v}{\partial x} + \int_{\widehat{D}} v\frac{\partial u}{\partial x} = \int_{\partial\widehat{D}} uv\widehat{\mathsf{n}} \xrightarrow{\text{Discretize}} \mathsf{v}^T\mathsf{Q}_h\mathsf{u} + \mathsf{v}^T\mathsf{Q}_h^T\mathsf{u} = \mathsf{v}_f^T\mathsf{B}\mathsf{u}_f$$

Discretization of the inviscid term - Modal ESDG

- “Hyrbridized” SBP operator

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

$$\int_{\widehat{D}} u\frac{\partial v}{\partial x} + \int_{\widehat{D}} v\frac{\partial u}{\partial x} = \int_{\partial\widehat{D}} uv\widehat{\mathsf{n}} \xRightarrow{\text{Discretize}} \mathsf{v}^T\mathsf{Q}_h\mathsf{u} + \mathsf{v}^T\mathsf{Q}_h^T\mathsf{u} = \mathsf{v}_f^T\mathsf{B}\mathsf{u}_f$$

- Entropy projection ensures consistency with respect to the polynomial basis and the entropy conservative flux

$$\mathsf{v}_h = \Pi_N \boldsymbol{v}, \qquad \widetilde{\mathsf{u}} = \boldsymbol{u}\left(\Pi_N \boldsymbol{v}\right)$$

$$\left(\Pi_N \boldsymbol{v}\left(\mathsf{u}_L\right) - \Pi_N \boldsymbol{v}\left(\mathsf{u}_R\right)\right)^T \boldsymbol{f}_S\left(\widetilde{\mathsf{u}}_L, \widetilde{\mathsf{u}}_R\right) = \psi\left(\widetilde{\boldsymbol{u}}_L\right) - \psi\left(\widetilde{\boldsymbol{u}}_R\right)$$

Discretization of the inviscid term - Modal ESDG

- "Hyrbridized" SBP operator

$$\mathsf{Q}_h = \frac{1}{2}\begin{bmatrix} \mathsf{Q}-\mathsf{Q}^T & \mathsf{E}^T\mathsf{B} \\ -\mathsf{BE} & \mathsf{B} \end{bmatrix}, \qquad \mathsf{Q}_h\mathbf{1} = 0, \qquad \mathsf{Q}_h + \mathsf{Q}_h^T = \begin{bmatrix} 0 & 0 \\ 0 & \mathsf{B} \end{bmatrix}$$

$$\int_{\widehat{D}} u\frac{\partial v}{\partial x} + \int_{\widehat{D}} v\frac{\partial u}{\partial x} = \int_{\partial\widehat{D}} uv\widehat{\mathsf{n}} \xRightarrow{\text{Discretize}} \mathsf{v}^T\mathsf{Q}_h\mathsf{u} + \mathsf{v}^T\mathsf{Q}_h^T\mathsf{u} = \mathsf{v}_f^T\mathsf{B}\mathsf{u}_f$$

- Entropy projection ensures consistency with respect to the polynomial basis and the entropy conservative flux

$$\mathsf{v}_h = \Pi_N\boldsymbol{v}, \qquad \widetilde{\mathsf{u}} = \boldsymbol{u}\left(\Pi_N\boldsymbol{v}\right)$$

$$\left(\Pi_N\boldsymbol{v}\left(\mathsf{u}_L\right) - \Pi_N\boldsymbol{v}\left(\mathsf{u}_R\right)\right)^T\boldsymbol{f}_S\left(\widetilde{\mathsf{u}}_L, \widetilde{\mathsf{u}}_R\right) = \psi\left(\widetilde{\boldsymbol{u}}_L\right) - \psi\left(\widetilde{\boldsymbol{u}}_R\right)$$

- Discretize the variational form

$$\int_{\widehat{D}} \frac{\partial \boldsymbol{f}}{\partial x}\vec{\boldsymbol{\phi}} \quad \xRightarrow{\text{Discretize}} \quad 2\left(\mathsf{Q}_h \circ \mathsf{F}_S\right)\mathbf{1}, \quad \left(\mathsf{F}_S\right)_{ij} = \boldsymbol{f}_S\left(\widetilde{\boldsymbol{u}}_i, \widetilde{\boldsymbol{u}}_j\right)$$

Discretization of the viscous term

- Usual approach: discretize the viscous term by writing it as a first order system

$$\begin{cases} \boldsymbol{\Theta} = \frac{\partial \boldsymbol{v}}{\partial x} \\ \boldsymbol{G}_{\text{visc}} = \frac{\partial \boldsymbol{K}\boldsymbol{\Theta}}{\partial x} \end{cases} \implies \begin{cases} (\boldsymbol{\Theta}, \varphi)_\Omega = \left(\frac{\partial \boldsymbol{v}}{\partial x}, \varphi\right)_\Omega + \langle [\![\boldsymbol{v}]\!] n_i, \varphi \rangle_{\partial\Omega} \\ (\boldsymbol{G}_{\text{visc}}, \psi)_\Omega = -\left(\boldsymbol{K}\boldsymbol{\Theta}, \frac{\partial \psi}{\partial x}\right)_\Omega + \langle \{\{\boldsymbol{K}(\boldsymbol{v})\,\boldsymbol{\Theta}\}\}\, n_i, \psi \rangle_{\partial\Omega} \end{cases}$$

Discretization of the viscous term

- Usual approach: discretize the viscous term by writing it as a first order system

$$\begin{cases} \boldsymbol{\Theta} = \frac{\partial \boldsymbol{v}}{\partial x} \\ \boldsymbol{G}_{\text{visc}} = \frac{\partial K \boldsymbol{\Theta}}{\partial x} \end{cases} \implies \begin{cases} (\boldsymbol{\Theta}, \varphi)_\Omega = \left(\frac{\partial \boldsymbol{v}}{\partial x}, \varphi\right)_\Omega + \langle [\![\boldsymbol{v}]\!] n_i, \varphi \rangle_{\partial\Omega} \\ (\boldsymbol{G}_{\text{visc}}, \psi)_\Omega = -\left(K\boldsymbol{\Theta}, \frac{\partial \psi}{\partial x}\right)_\Omega + \langle \{\{ K(\boldsymbol{v})\, \boldsymbol{\Theta} \}\}\, n_i, \psi \rangle_{\partial\Omega} \end{cases}$$

- Previous enforcement of wall boundary conditions (Dalcin et al.) involve transformations between primitive and entropy variables.
    - Step 1: Enforce boundary conditions on primitive variables $\boldsymbol{w}$, define $\boldsymbol{v}^+, K^+$ accordingly.
    - Step 2: Rotate $\boldsymbol{\Theta}$ to the gradient of primitive variables $\boldsymbol{\Pi}$
    - Step 3: Enforce Neumann boundary conditions on $\boldsymbol{\Pi}$, get $\boldsymbol{\Pi}^+$
    - Step 4: Rotate $\boldsymbol{\Pi}^+$ back to $\boldsymbol{\Theta}^+$

Viscous term discretization

- We write the system differently:

$$\begin{cases} \boldsymbol{\Theta} = \frac{\partial \mathbf{v}}{\partial x} \\ \boldsymbol{\sigma} = K\boldsymbol{\Theta} = g \\ G_{\text{visc}} = \frac{\partial \sigma}{\partial x} \end{cases} \implies \begin{cases} (\boldsymbol{\Theta}, \varphi)_\Omega = \left(\frac{\partial \mathbf{v}}{\partial x}, \varphi\right)_\Omega + \langle [\![\mathbf{v}]\!] n_i, \varphi \rangle_{\partial\Omega} \\ (\boldsymbol{\sigma}, \eta)_\Omega = (K\boldsymbol{\Theta}, \eta)_\Omega \\ (G_{\text{visc}}, \psi)_\Omega = -\left(\boldsymbol{\sigma}, \frac{\partial \psi}{\partial x}\right)_\Omega + \langle \{\{\boldsymbol{\sigma}\}\}\, n_i, \psi \rangle_{\partial\Omega} \end{cases}$$

Viscous term discretization

- We write the system differently:

$$\begin{cases} \boldsymbol{\Theta} = \frac{\partial \mathbf{v}}{\partial x} \\ \boldsymbol{\sigma} = K\boldsymbol{\Theta} = g \\ G_{\text{visc}} = \frac{\partial \sigma}{\partial x} \end{cases} \implies \begin{cases} (\boldsymbol{\Theta}, \varphi)_\Omega = \left(\frac{\partial \mathbf{v}}{\partial x}, \varphi\right)_\Omega + \langle [\![\mathbf{v}]\!] n_i, \varphi \rangle_{\partial\Omega} \\ (\boldsymbol{\sigma}, \eta)_\Omega = (K\boldsymbol{\Theta}, \eta)_\Omega \\ (G_{\text{visc}}, \psi)_\Omega = -\left(\boldsymbol{\sigma}, \frac{\partial \psi}{\partial x}\right)_\Omega + \langle \{\{\boldsymbol{\sigma}\}\}\, n_i, \psi \rangle_{\partial\Omega} \end{cases}$$

- Viscous term dissipates entropy

$$\sum_k (G_{\text{visc}}, \mathbf{v})_{D^k} = \sum_k \sum_{i,j=1}^{d} -\left(K_{ij}\boldsymbol{\Theta}_j, \boldsymbol{\Theta}_i\right)_{D^k} \leq 0$$

Simpler enforcement of wall boundary conditions

- Adiabatic no-slip wall boundary condition

$$\begin{cases} u_n & = 0 \\ u_t & = u_{\text{wall}} \\ \kappa \frac{\partial T}{\partial n} \frac{1}{T} & = g(t) \end{cases} \implies \begin{cases} \{\{v_{1+i}\}\} & = -u_{i,\text{wall}} v_4 \\ \{\{\sigma_{4,i}\}\} & = u_{1,\text{wall}} \sigma_{2,i} + u_{2,\text{wall}} \sigma_{3,i} + \frac{g(t) n_i}{c_v v_4} \end{cases}$$

Simpler enforcement of wall boundary conditions

- Adiabatic no-slip wall boundary condition

$$\begin{cases} u_n & = 0 \\ u_t & = u_{\text{wall}} \\ \kappa \frac{\partial T}{\partial n} \frac{1}{T} & = g(t) \end{cases} \implies \begin{cases} \{\{\mathbf{v}_{1+i}\}\} & = -u_{i,\text{wall}} \mathbf{v}_4 \\ \{\{\sigma_{4,i}\}\} & = u_{1,\text{wall}} \sigma_{2,i} + u_{2,\text{wall}} \sigma_{3,i} + \frac{g(t) n_i}{c_v \mathbf{v}_4} \end{cases}$$

- Reflective wall boundary condition

$$\begin{cases} u_n & = 0 \\ \sum t_i \sigma_{ij} n_j & = 0 \\ \kappa \frac{\partial T}{\partial n} & = 0 \end{cases} \implies \begin{cases} \{\{\mathbf{v}\}\} = \mathbf{v} - v_n \mathbf{n} \\ \begin{bmatrix} \{\{\sigma_{2,1}\}\} & \{\{\sigma_{2,2}\}\} \\ \{\{\sigma_{3,1}\}\} & \{\{\sigma_{3,2}\}\} \end{bmatrix} \mathbf{n} = \mathbf{n}\mathbf{n}^T \begin{bmatrix} \sigma_{2,1} & \sigma_{2,2} \\ \sigma_{3,1} & \sigma_{3,2} \end{bmatrix} \mathbf{n} \\ \{\{v_4\}\} = v_4, \quad \{\{\sigma_{4,i}\}\} = 0 \end{cases}$$

Simpler enforcement of wall boundary conditions

- Adiabatic no-slip wall boundary condition

$$\begin{cases} u_n & = 0 \\ u_t & = u_{\text{wall}} \\ \kappa \frac{\partial T}{\partial n} \frac{1}{T} & = g(t) \end{cases} \implies \begin{cases} \{\{\mathbf{v}_{1+i}\}\} & = -u_{i,\text{wall}} \mathbf{v}_4 \\ \{\{\sigma_{4,i}\}\} & = u_{1,\text{wall}} \sigma_{2,i} + u_{2,\text{wall}} \sigma_{3,i} + \frac{g(t) n_i}{c_v \mathbf{v}_4} \end{cases}$$

- Reflective wall boundary condition

$$\begin{cases} u_n & = 0 \\ \sum t_i \sigma_{ij} n_j & = 0 \\ \kappa \frac{\partial T}{\partial n} & = 0 \end{cases} \implies \begin{cases} \{\{\mathbf{v}\}\} = \mathbf{v} - \mathbf{v}_n \boldsymbol{n} \\ \begin{bmatrix} \{\{\sigma_{2,1}\}\} & \{\{\sigma_{2,2}\}\} \\ \{\{\sigma_{3,1}\}\} & \{\{\sigma_{3,2}\}\} \end{bmatrix} \boldsymbol{n} = \boldsymbol{n}\boldsymbol{n}^T \begin{bmatrix} \sigma_{2,1} & \sigma_{2,2} \\ \sigma_{3,1} & \sigma_{3,2} \end{bmatrix} \boldsymbol{n} \\ \{\{v_4\}\} = v_4, \quad \{\{\sigma_{4,i}\}\} = 0 \end{cases}$$

- Simplifies the proofs of entropy stability under these wall boundary conditions.

Numerical results: Viscous shocktube

$h \backslash N$	$N = 2$	Rate	$N = 3$	Rate	$N = 4$	Rate
$1/4$	0.0318		0.00566		0.0052	
$1/8$	0.00306	3.376	0.000462	3.616	0.000185	4.814
$1/16$	0.00033	3.213	5.22e-05	3.146	5.18e-06	5.157
$1/32$	4.5e-05	2.874	3.13e-06	4.060	1.81e-07	4.842
$1/64$	5.81e-06	2.952	2.1e-07	3.898	6.24e-09	4.856

(a) L^1 errors

$h \backslash N$	$N = 2$	Rate	$N = 3$	Rate	$N = 4$	Rate
$1/4$	0.0639		0.00916		0.011	
$1/8$	0.00631	3.339	0.00103	3.156	0.000449	4.618
$1/16$	0.000853	2.887	0.000162	2.665	1.38e-05	5.020
$1/32$	0.000132	2.694	9.71e-06	4.062	6.19e-07	4.483
$1/64$	1.73e-05	2.931	6.87e-07	3.821	2.16e-08	4.839

(b) L^2 errors

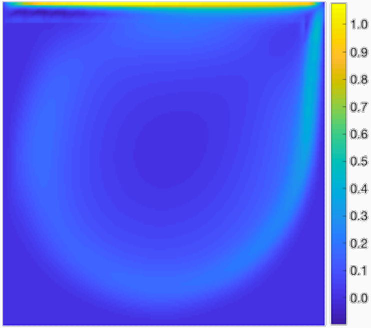
Table 1: L^1 and L^2 errors for the viscous shock tube problem.

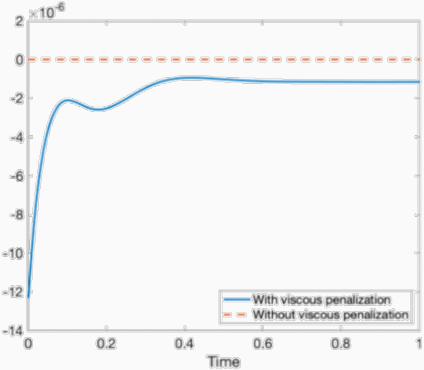
Numerical results: Lid driven cavity

- Evolution of viscous entropy residual $r(t)$

$$r(t) = \sum_{k} \left[\left(\mathbf{G}_{\text{visc}}, \mathbf{v}\right)_{D^k} + \sum_{i,j=1}^{d} \left(\mathbf{K}_{ij}\boldsymbol{\Theta}_j, \boldsymbol{\Theta}_i\right) \right] = \left\langle \frac{g(t)}{c_v}, 1 \right\rangle_{\partial\Omega}$$

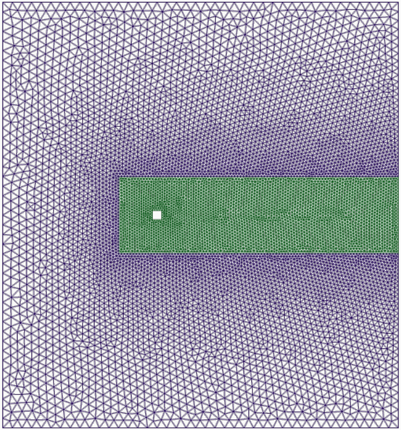
- Lid driven cavity with $Re = 1000$, adiabatic wall boundary condition $g(t) = 0$



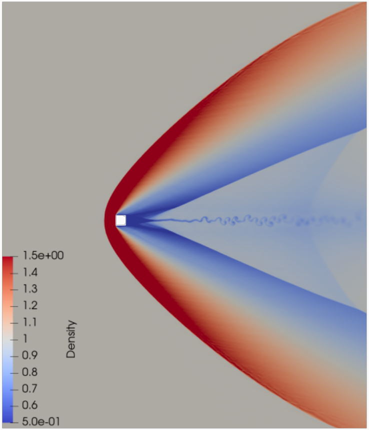


Numerical results: Flow over cylinder

- Supersonic flow over cylinder with $Re = 10^4$, $Ma = 1.5$



(a) Mesh



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

Current work: Positivity Limiting for nodal ESDG

- The entropy is well-defined only if densities and pressures are well-defined.

Current work: Positivity Limiting for nodal ESDG

- Strong shock forms - Negative densities

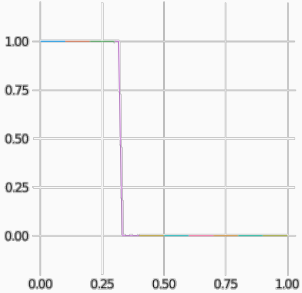


Figure 1: Exact solution

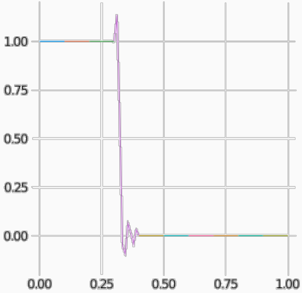


Figure 2: Solution in polynomial basis

- Oscillation by Gibbs phenomenon leads to negative density

Limiting strategy

- Step 1. Compute high order target scheme (nodal ESDG)

Limiting strategy

- Step 1. Compute high order target scheme (nodal ESDG)
- Step 2. Compute low order positivity-preserving scheme

Limiting strategy

- Step 1. Compute high order target scheme (nodal ESDG)
- Step 2. Compute low order positivity-preserving scheme
- Step 3. Blend two schemes together through convex limiting

Limiting strategy

- Step 1. Compute high order target scheme (nodal ESDG)
- Step 2. Compute low order positivity-preserving scheme
- Step 3. Blend two schemes together through convex limiting
  - Low order positivity-preserving and ESDG in algebraic flux form:

$$\frac{\boldsymbol{m}_i}{\tau}(\boldsymbol{u}_i^{L,n+1} - \boldsymbol{u}_i^n) + \sum \mathsf{F}_{ij}^{L,n} = 0$$
$$\frac{\boldsymbol{m}_i}{\tau}(\boldsymbol{u}_i^{H,n+1} - \boldsymbol{u}_i^n) + \sum \mathsf{F}_{ij}^{H,n} = 0$$

Limiting strategy

- Step 1. Compute high order target scheme (nodal ESDG)
- Step 2. Compute low order positivity-preserving scheme
- Step 3. Blend two schemes together through convex limiting
  - Low order positivity-preserving and ESDG in algebraic flux form:

$$\frac{\boldsymbol{m}_i}{\tau}(\boldsymbol{u}_i^{L,n+1} - \boldsymbol{u}_i^n) + \sum \mathsf{F}_{ij}^{L,n} = 0$$
$$\frac{\boldsymbol{m}_i}{\tau}(\boldsymbol{u}_i^{H,n+1} - \boldsymbol{u}_i^n) + \sum \mathsf{F}_{ij}^{H,n} = 0$$

  - Choose suitable parameter $l_{ij} \in [0,1]$ to satisfy positivity

$$\boldsymbol{m}_i\boldsymbol{u}_i^{n+1} = \boldsymbol{m}_i\boldsymbol{u}_i^{L,n+1} + \sum \tau l_{ij}(\mathsf{F}_{ij}^{L,n} - \mathsf{F}_{ij}^{H,n})$$

Limiting strategy

- Step 1. Compute high order target scheme (nodal ESDG)
- Step 2. Compute low order positivity-preserving scheme
- Step 3. Blend two schemes together through convex limiting
    - Low order positivity-preserving and ESDG in algebraic flux form:

$$\frac{\boldsymbol{m}_i}{\tau}(\boldsymbol{u}_i^{L,n+1} - \boldsymbol{u}_i^n) + \sum \mathsf{F}_{ij}^{L,n} = 0$$
$$\frac{\boldsymbol{m}_i}{\tau}(\boldsymbol{u}_i^{H,n+1} - \boldsymbol{u}_i^n) + \sum \mathsf{F}_{ij}^{H,n} = 0$$

    - Choose suitable parameter $l_{ij} \in [0, 1]$ to satisfy positivity

$$\boldsymbol{m}_i\boldsymbol{u}_i^{n+1} = \boldsymbol{m}_i\boldsymbol{u}_i^{L,n+1} + \sum \tau l_{ij}(\mathsf{F}_{ij}^{L,n} - \mathsf{F}_{ij}^{H,n})$$

    - $l_{ij} = 1 \implies$ recovers ESDG.
      $l_{ij} = 0 \implies$ recovers low order positivity-preserving scheme.

Limiting strategy

- Step 1. Compute high order target scheme (nodal ESDG)
- Step 2. Compute low order positivity-preserving scheme
- Step 3. Blend two schemes together through convex limiting
  - Low order positivity-preserving and ESDG in algebraic flux form:

$$\frac{\boldsymbol{m}_i}{\tau}(\boldsymbol{u}_i^{L,n+1} - \boldsymbol{u}_i^n) + \sum \mathsf{F}_{ij}^{L,n} = 0$$
$$\frac{\boldsymbol{m}_i}{\tau}(\boldsymbol{u}_i^{H,n+1} - \boldsymbol{u}_i^n) + \sum \mathsf{F}_{ij}^{H,n} = 0$$

  - Choose suitable parameter $l_{ij} \in [0,1]$ to satisfy positivity

$$\boldsymbol{m}_i \boldsymbol{u}_i^{n+1} = \boldsymbol{m}_i \boldsymbol{u}_i^{L,n+1} + \sum \tau l_{ij} (\mathsf{F}_{ij}^{L,n} - \mathsf{F}_{ij}^{H,n})$$

  - $l_{ij} = 1 \implies$ recovers ESDG.
    $l_{ij} = 0 \implies$ recovers low order positivity-preserving scheme.
  - Find largest possible l_{ij} that satisify positivity.

Positivity preserving discretization

- Low order positivity preserving method could be written as

$$\underbrace{\boldsymbol{m}_i \frac{\partial \boldsymbol{u}}{\partial t} + \sum \mathsf{Q}_{ij} \left(\boldsymbol{f}(\boldsymbol{u}_j) - \boldsymbol{\sigma}_j \right)}_{\text{low order nodal DG on LGL nodes}} - \underbrace{\sum d_{ij} (\boldsymbol{u}_j - \boldsymbol{u}_i)}_{\text{graph viscosity}} = 0$$

Positivity preserving discretization

- Low order positivity preserving method could be written as

$$\underbrace{\boldsymbol{m}_i \frac{\partial \boldsymbol{u}}{\partial t} + \sum \boldsymbol{Q}_{ij}\left(\boldsymbol{f}(\boldsymbol{u}_j) - \boldsymbol{\sigma}_j\right)}_{\text{low order nodal DG on LGL nodes}} - \underbrace{\sum d_{ij}(\boldsymbol{u}_j - \boldsymbol{u}_i)}_{\text{graph viscosity}} = 0$$

- Weighted differentiation matrix $\boldsymbol{Q}$ is a sparse low order (SBP) operator:

$$\begin{bmatrix} -\frac{1}{2} & \frac{1}{2} & 0 & 0 & \dots & 0 \\ -\frac{1}{2} & 0 & \frac{1}{2} & 0 & \dots & 0 \\ 0 & -\frac{1}{2} & 0 & \frac{1}{2} & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix}$$

Positivity preserving discretization

- Low order positivity preserving method could be written as

$$\underbrace{\boldsymbol{m}_i \frac{\partial \boldsymbol{u}}{\partial t} + \sum \boldsymbol{Q}_{ij} \left(\boldsymbol{f}(\boldsymbol{u}_j) - \boldsymbol{\sigma}_j \right)}_{\text{low order nodal DG on LGL nodes}} - \underbrace{\sum d_{ij}(\boldsymbol{u}_j - \boldsymbol{u}_i)}_{\text{graph viscosity}} = 0$$

- Weighted differentiation matrix $\boldsymbol{Q}$ is a sparse low order (SBP) operator:

$$\begin{bmatrix} -\frac{1}{2} & \frac{1}{2} & 0 & 0 & \dots & 0 \\ -\frac{1}{2} & 0 & \frac{1}{2} & 0 & \dots & 0 \\ 0 & -\frac{1}{2} & 0 & \frac{1}{2} & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix}$$

- Define the graph viscosity coefficients:

$$d_{ij} = \max \left\{ \beta(\boldsymbol{u}_i, \boldsymbol{u}_j, \boldsymbol{n}_{ij}) \|\boldsymbol{Q}_{ij}\|, \beta(\boldsymbol{u}_j, \boldsymbol{u}_i, \boldsymbol{n}_{ji}) \|\boldsymbol{Q}_{ji}\| \right\}, \boldsymbol{n}_{ij} = \boldsymbol{Q}_{ij} / \|\boldsymbol{Q}_{ij}\|$$

Positivity preserving discretization

- Low order positivity preserving method could be written as

$$\underbrace{\boldsymbol{m}_i \frac{\partial \boldsymbol{u}}{\partial t} + \sum \boldsymbol{Q}_{ij} \left(\boldsymbol{f}(\boldsymbol{u}_j) - \boldsymbol{\sigma}_j \right)}_{\text{low order nodal DG on LGL nodes}} - \underbrace{\sum d_{ij}(\boldsymbol{u}_j - \boldsymbol{u}_i)}_{\text{graph viscosity}} = 0$$

- Weighted differentiation matrix $\boldsymbol{Q}$ is a sparse low order (SBP) operator:

$$\begin{bmatrix} -\frac{1}{2} & \frac{1}{2} & 0 & 0 & \dots & 0 \\ -\frac{1}{2} & 0 & \frac{1}{2} & 0 & \dots & 0 \\ 0 & -\frac{1}{2} & 0 & \frac{1}{2} & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix}$$

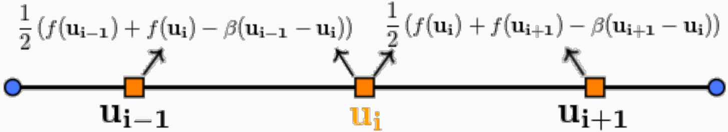
- Define the graph viscosity coefficients:

$$d_{ij} = \max \left\{ \beta(\boldsymbol{u}_i, \boldsymbol{u}_j, \boldsymbol{n}_{ij}) \|\boldsymbol{Q}_{ij}\|, \beta(\boldsymbol{u}_j, \boldsymbol{u}_i, \boldsymbol{n}_{ji}) \|\boldsymbol{Q}_{ji}\| \right\}, \boldsymbol{n}_{ij} = \boldsymbol{Q}_{ij} / \|\boldsymbol{Q}_{ij}\|$$

- Compressible Navier-Stokes - Zhang's positivity preserving flux
  Compressible Euler - Maximum wavespeed (Lax-Friedrichs flux)

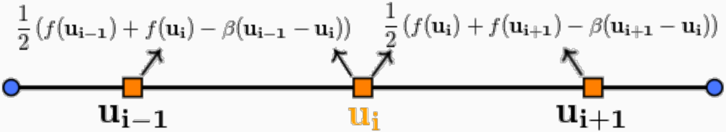
Positivity preserving discretization

- Interpretation: subcell Lax-Friedriches type dissipation

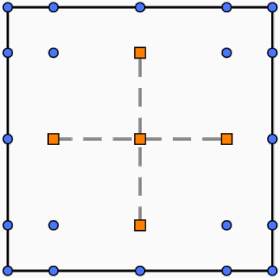


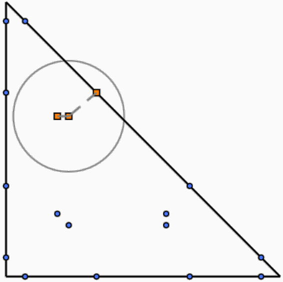
Positivity preserving discretization

- Interpretation: subcell Lax-Friedriches type dissipation



- Extension to 2D (tensor product and simplex elements)





Numerical results: LeBlanc shocktube

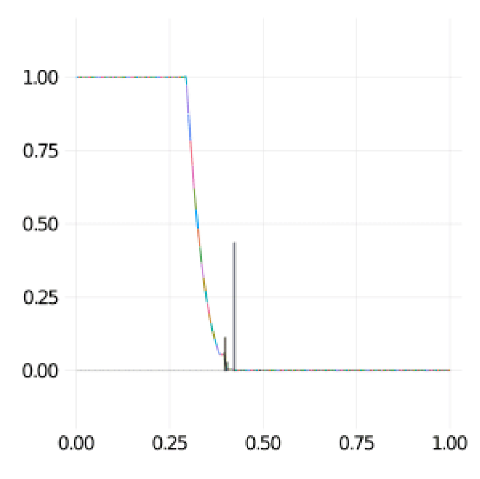


Figure 3: LeBlanc shocktube, N = 3, K = 200

Summary and future works

- We present an entropy stable approach to discretize the viscous term and explicit formulas for entropy stable imposition of no-slip and reflective boundary conditions.
- We briefly preview a positive limiting strategy for nodal ESDG.
- Future work: curvilinear and moving meshes, positivity limiting for modal ESDG.

Thank you!