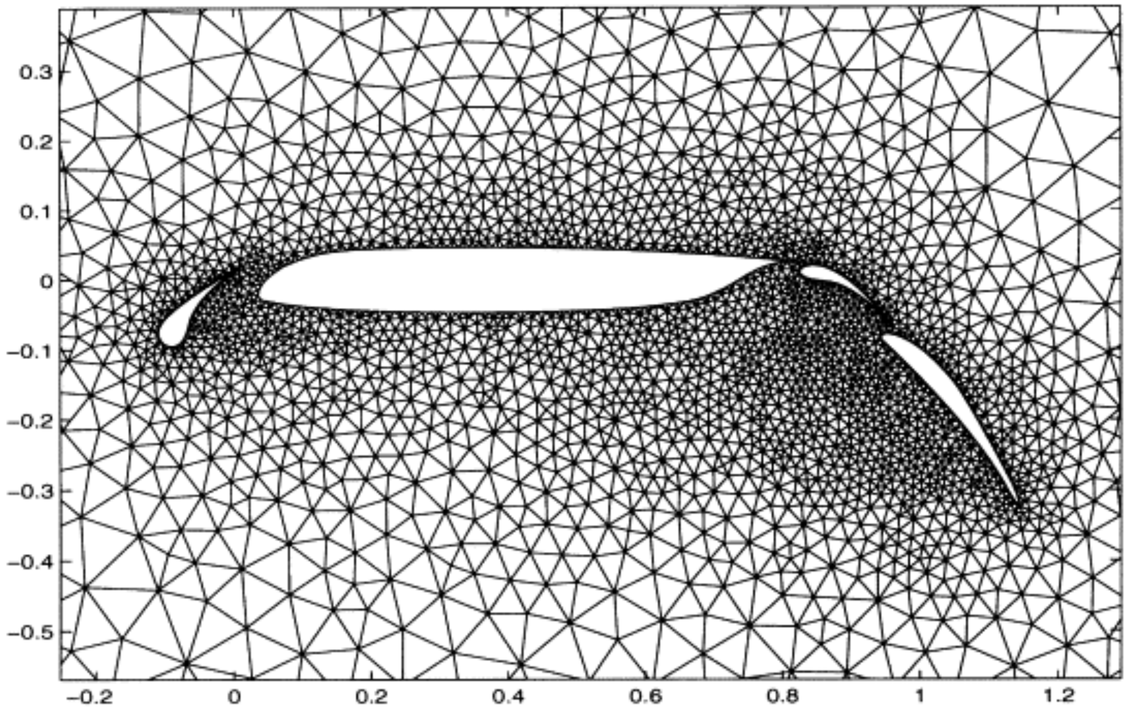
# High order entropy stable discontinuous Galerkin methods

Jesse Chan
Dept. of Computational Applied Mathematics and Operations Research
Rice University

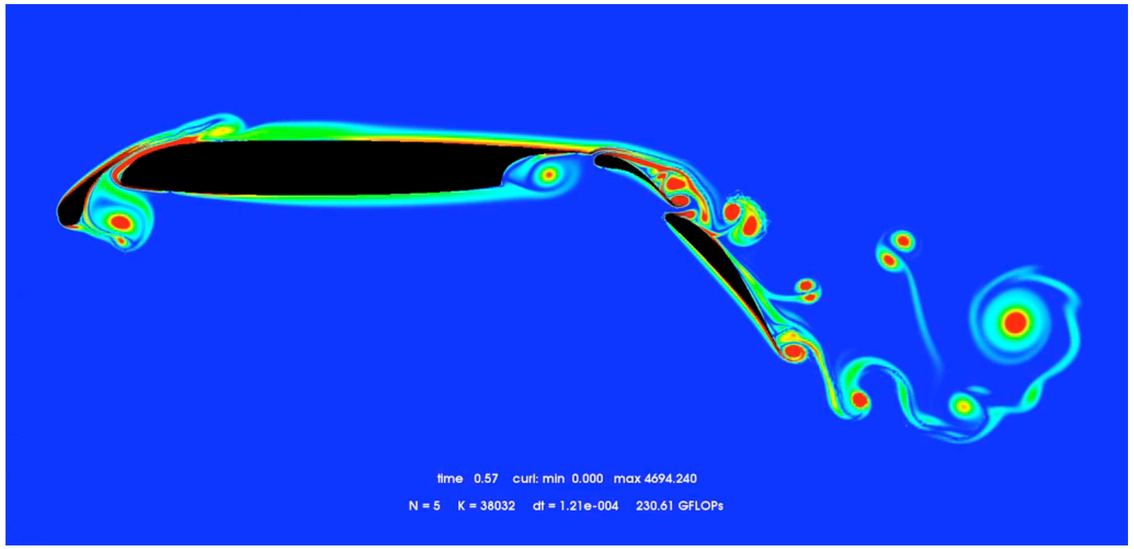
Advanced Modeling & Simulation (AMS) Seminar Series
NASA Ames Research Center
August 13th, 2024

# Goal: accurate unsteady flow simulations

- High order finite element methods, e.g., spectral elements, discontinuous Galerkin (DG), can handle complex geometries.




Mesh from Slawig 2001




- Phenomena in fluid dynamics: acoustic waves, vorticular flows, turbulence, shocks

- Relevant to a range of systems: compressible flows, shallow water, magnetohydrodynamics, …

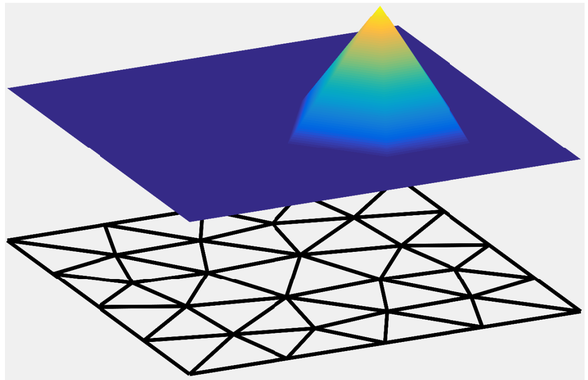
# Discontinuous Galerkin (DG) methods

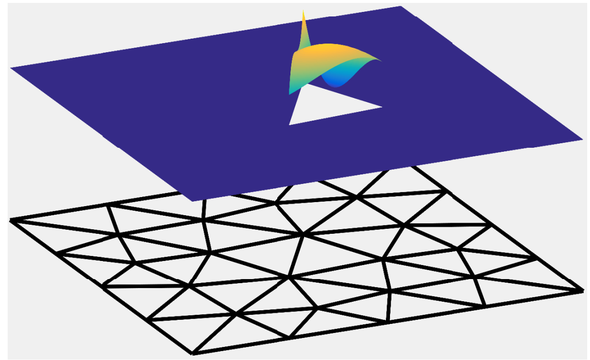
## Easily invertible mass matrices for explicit time-stepping

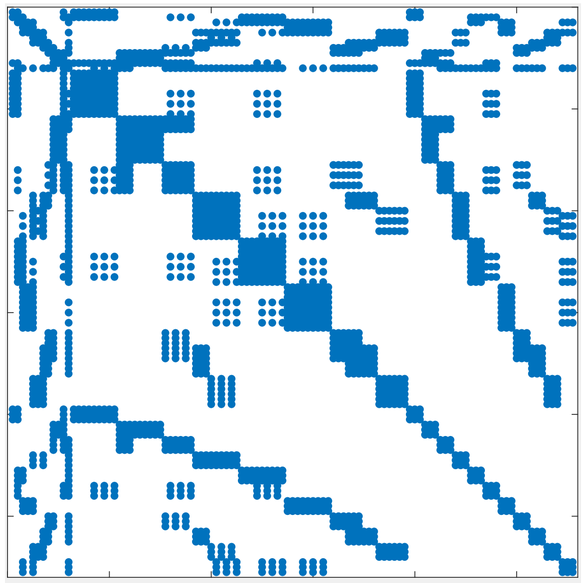
- Finite element discretizations of time-dependent PDEs

$$\mathbf{M}\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} = \mathbf{f}(\mathbf{u}, t)$$

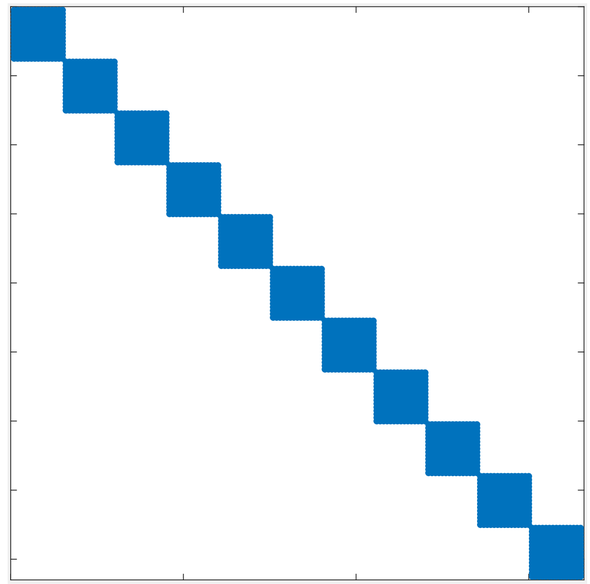
- High order FEM: sparse but globally coupled mass matrix
- High order DG: block diagonal mass matrix (can pre-invert during setup)







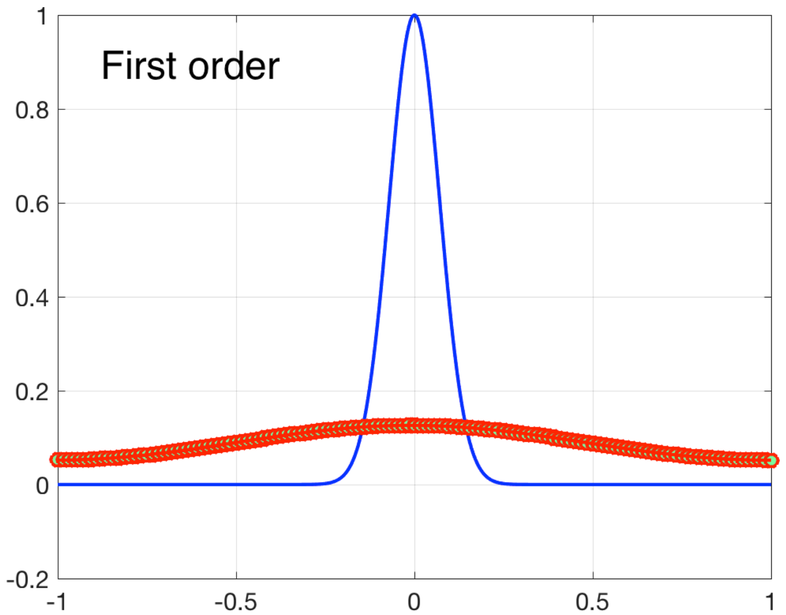
High order FEM

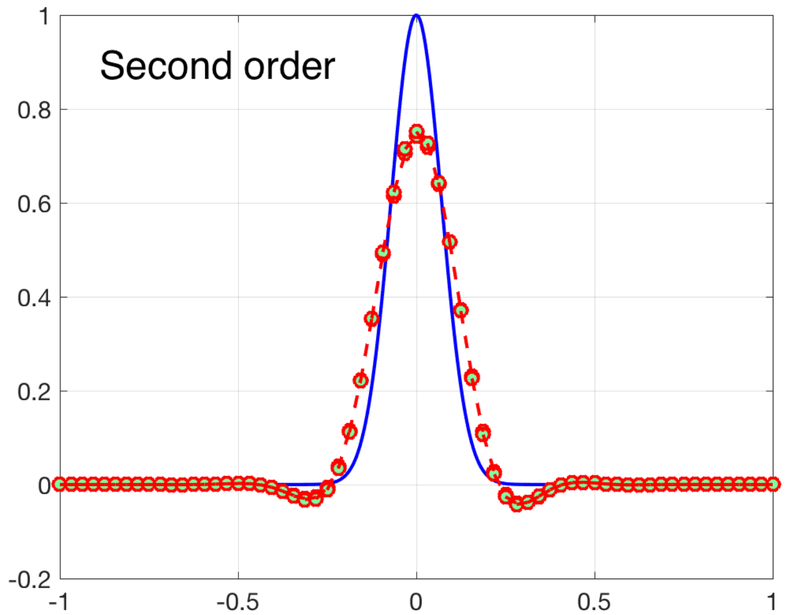


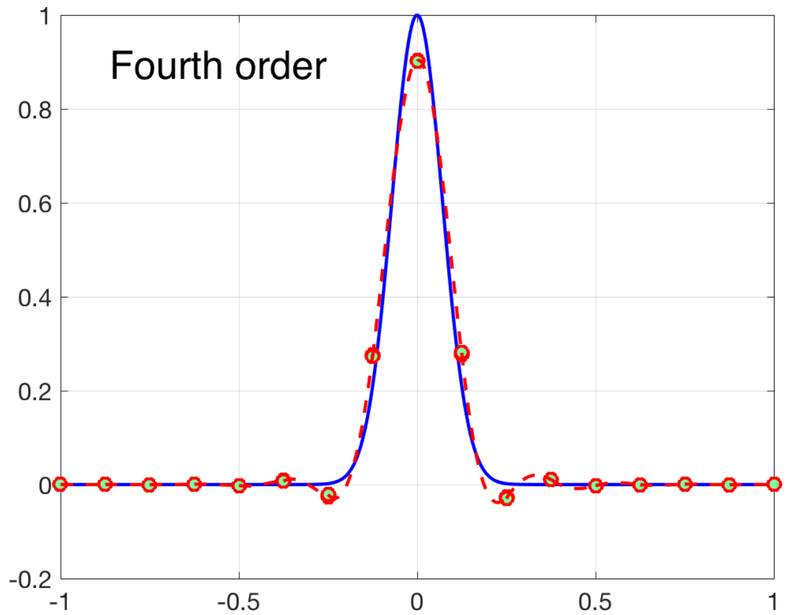
High order DG

# Why high order accuracy?

High order methods have low dispersion and dissipation errors.



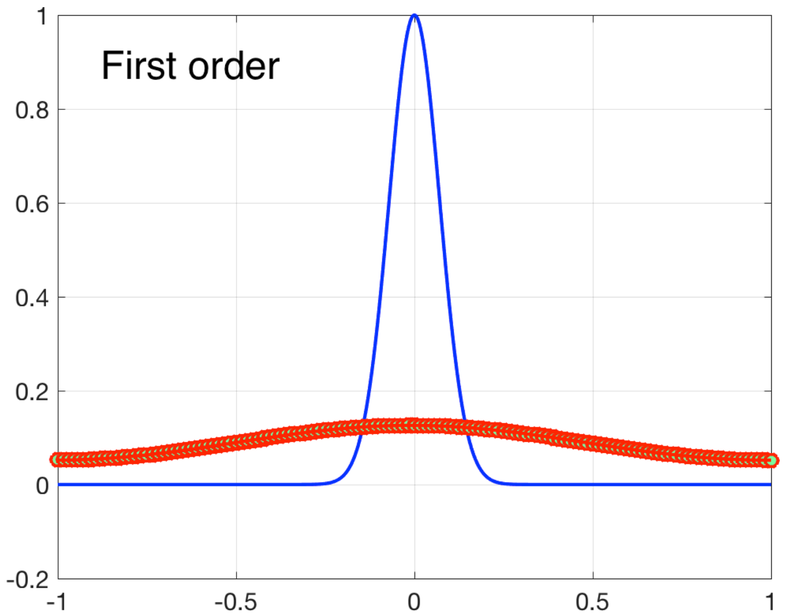



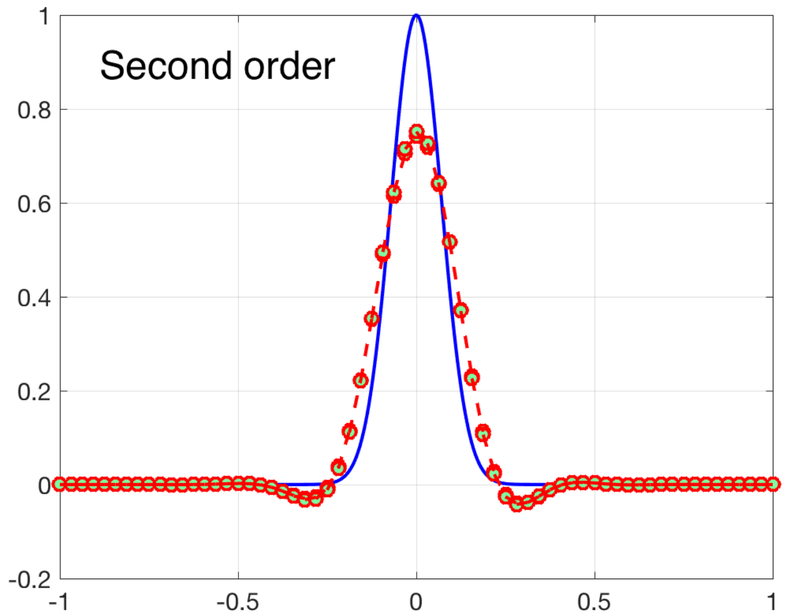



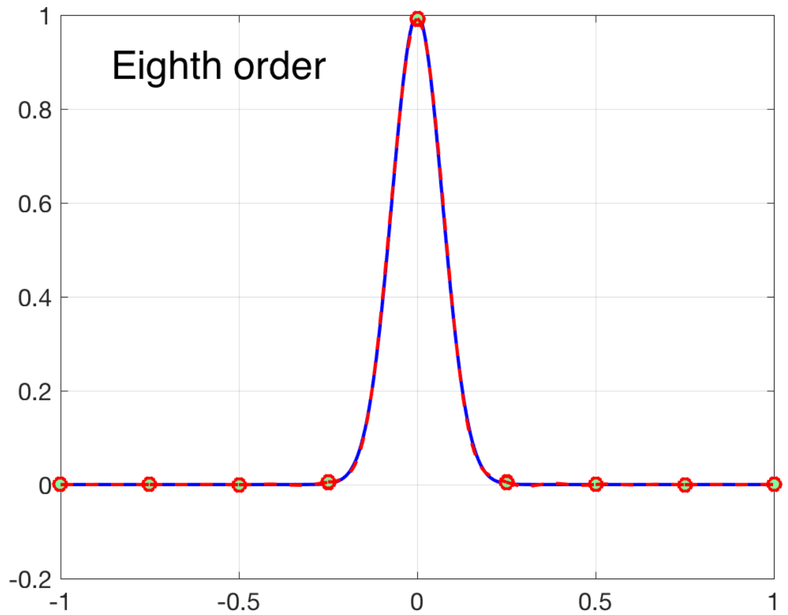

Beck, Gassner (2013). *On the accuracy of high-order discretizations for underresolved turbulence simulations*

# Why high order accuracy?

High order methods have low dispersion and dissipation errors.



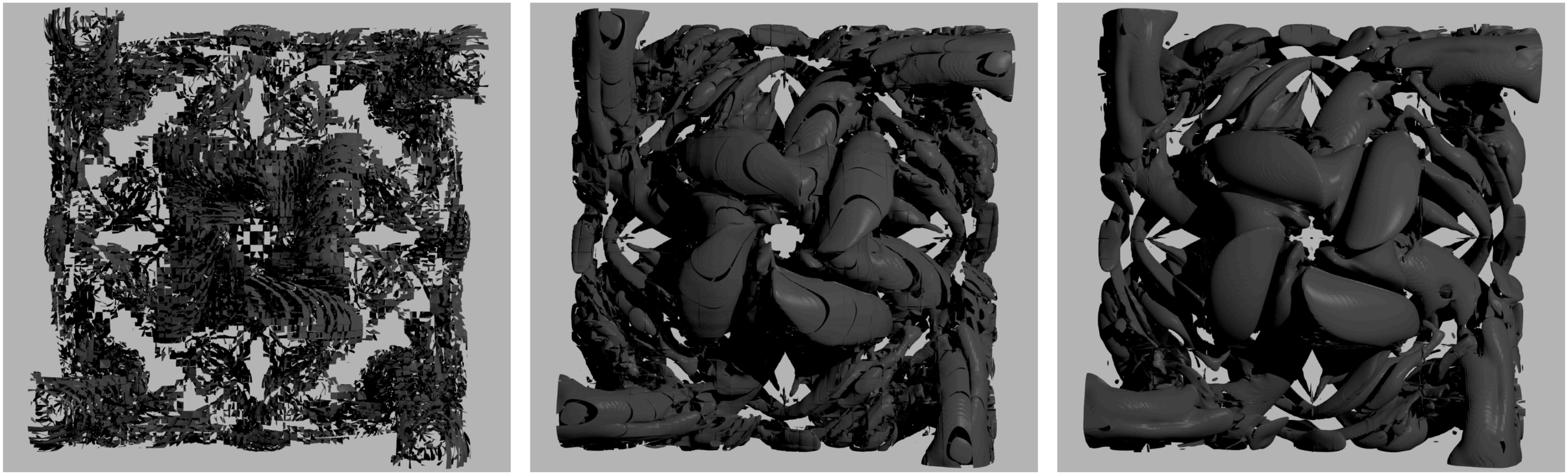




Beck, Gassner (2013). *On the accuracy of high-order discretizations for underresolved turbulence simulations*

# Why high order accuracy?

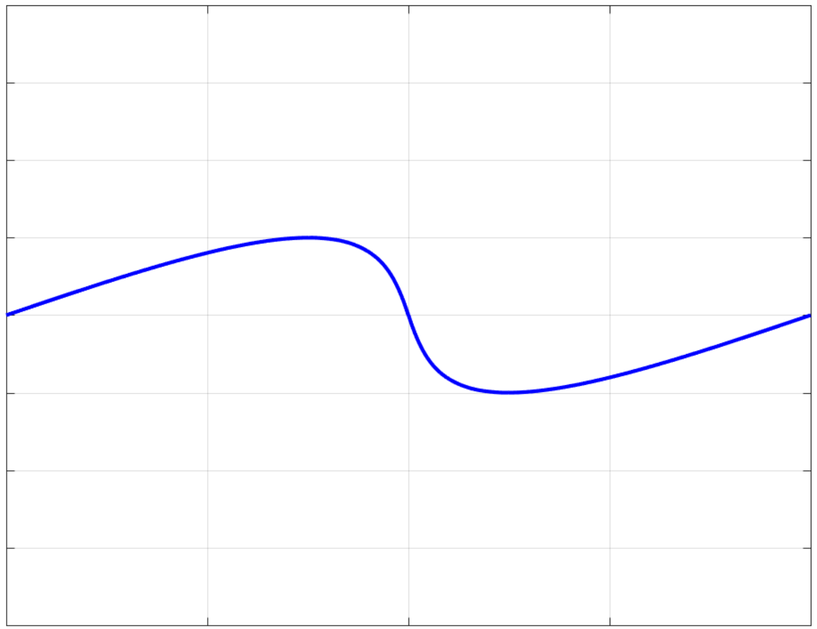
High order methods have low dispersion and dissipation errors.



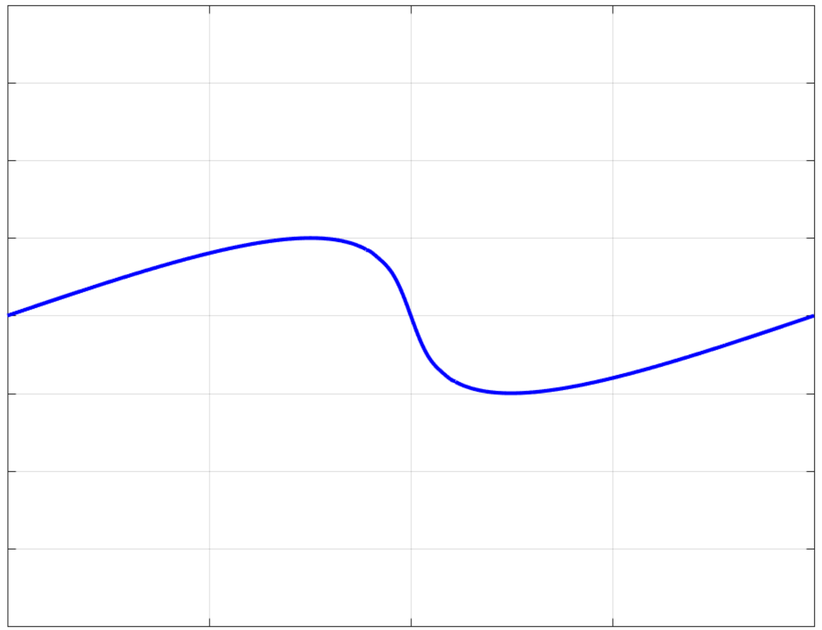
2nd order, 4th order, and 16th order simulations of the Taylor-Green vortex from Beck, Gassner (2013)

Beck, Gassner (2013). *On the accuracy of high-order discretizations for underresolved turbulence simulations*

# Why *not* high order accuracy: stability



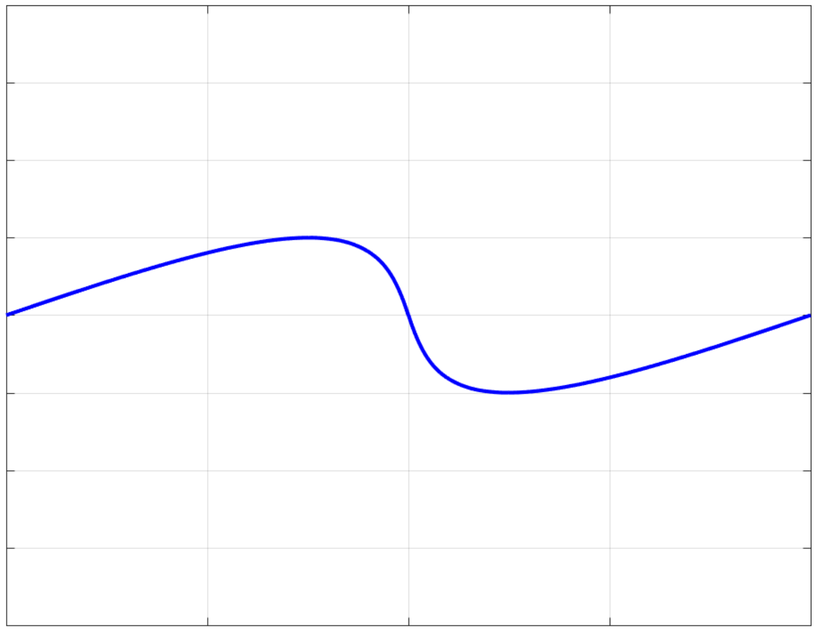
Exact solution of Burger's equation



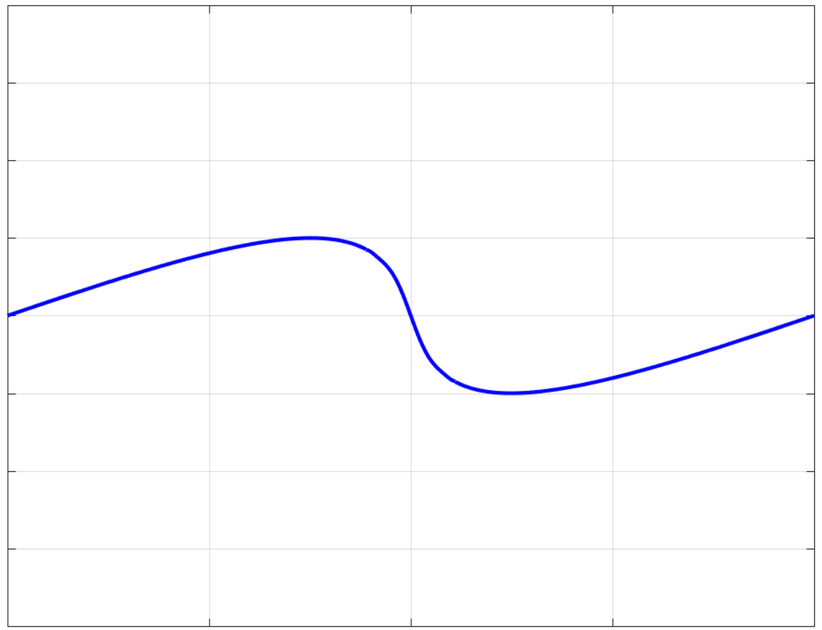
Unstable 8th order DG simulation of Burgers' equation.

- High order methods are typically unstable for nonlinear conservation laws with under-resolved solution features (for example, shocks and turbulence).
- This instability is typically addressed via dissipative *stabilization* techniques.

# Why *not* high order accuracy: stability



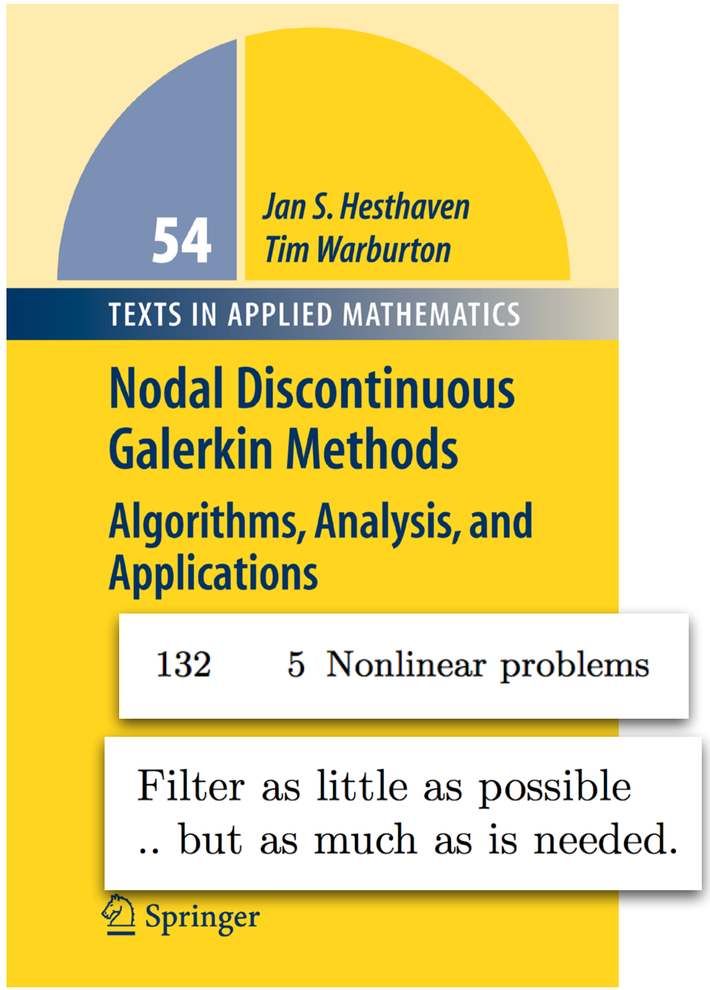
Exact solution of Burger's equation



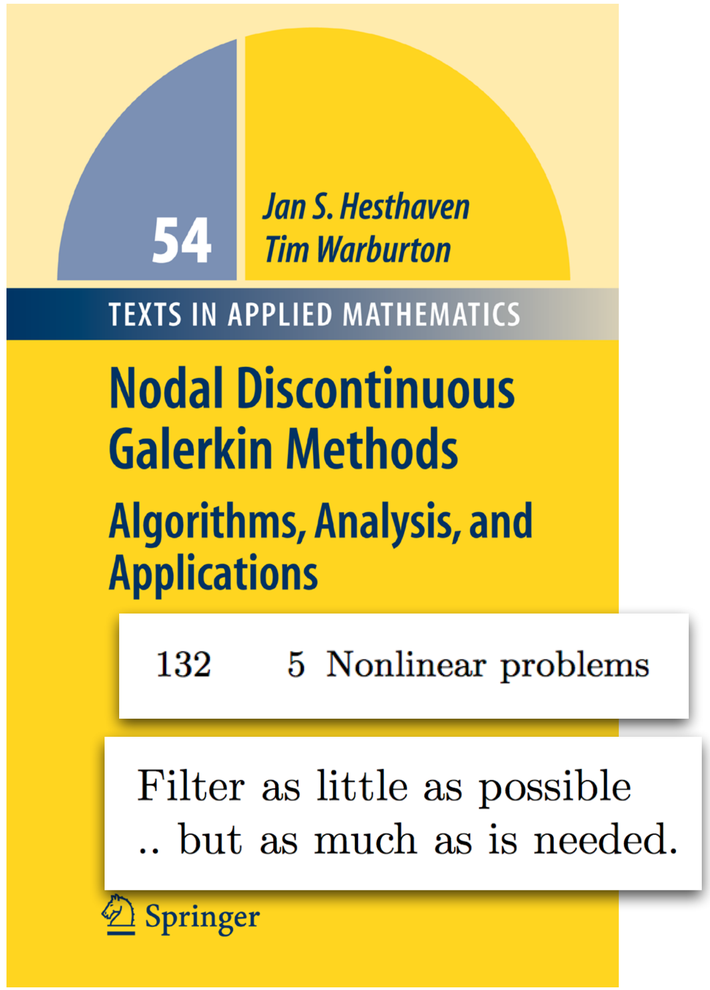
Unstable 8th order DG simulation of Burgers' equation.

- High order methods are typically unstable for nonlinear conservation laws with under-resolved solution features (for example, shocks and turbulence).
- This instability is typically addressed via dissipative *stabilization* techniques.

# The challenge with traditional stabilization



# The challenge with traditional stabilization



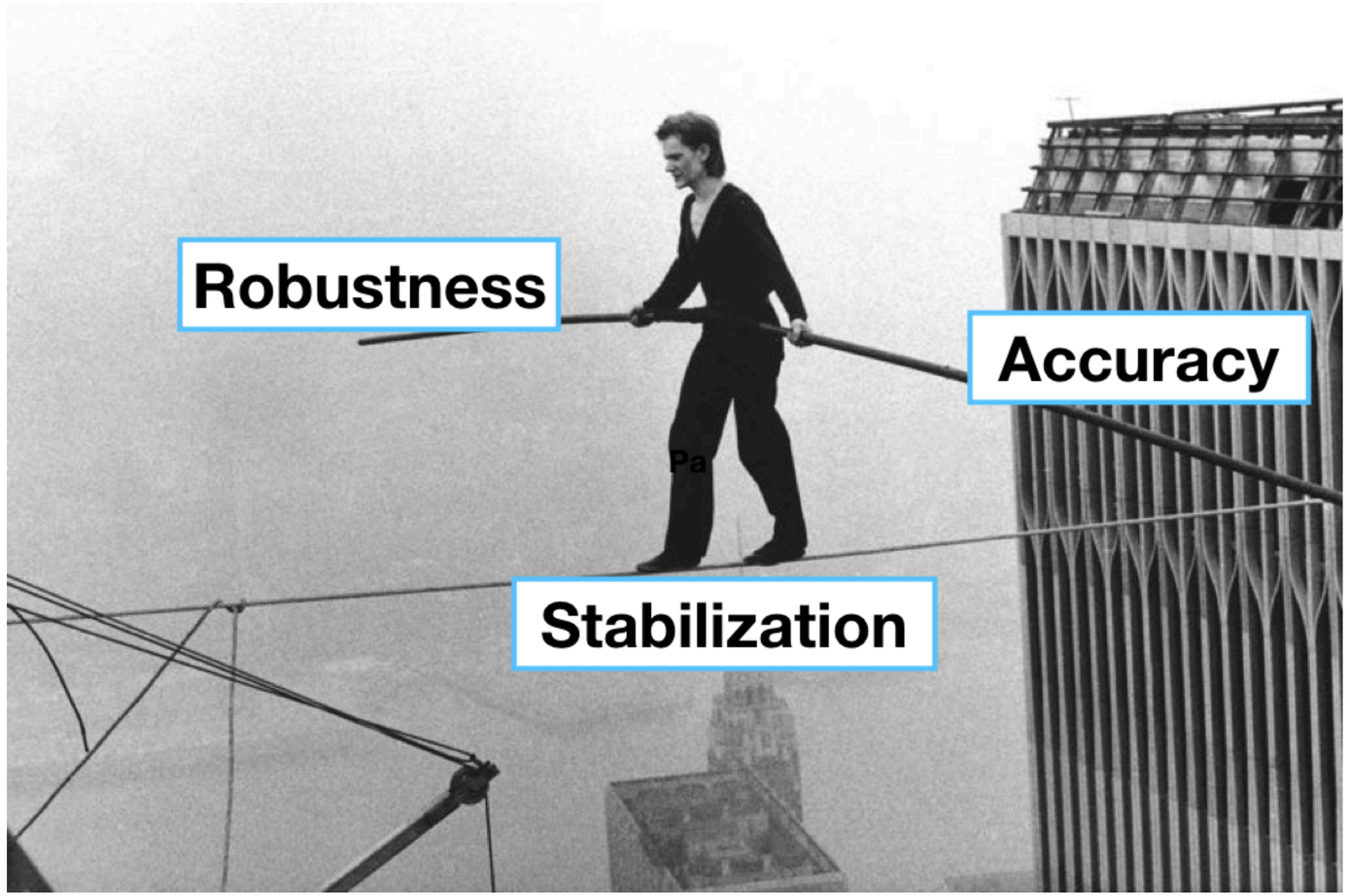




Image from “Man on Wire” (2008)

# Why entropy stability for high order methods?

- The entropy stable schemes I am interested in improve robustness *without explicitly adding numerical dissipation.*
  - Stability of these schemes is independent of both solution resolution and discretization “crimes” (e.g., inexact quadrature, polynomial aliasing errors).
  - Results in formally high order accurate methods which (though not 100% bulletproof) are “good” out of the box.
- Distinct from the entropy stable schemes finite element schemes based on entropy variable symmetrization.

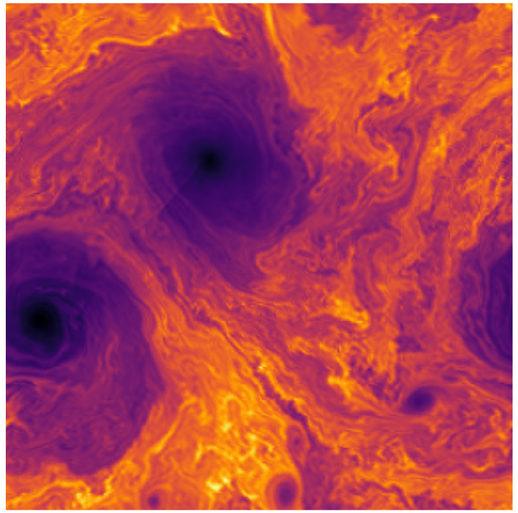
Diosady, Murman (2015). *Higher-order methods for compressible turbulent flows using entropy variables.*

Hughes, Franca, Mallet (1986). *A new FE formulation for CFD: I. Symmetric forms of the compressible Euler and NS equations and the second law of thermodynamics*

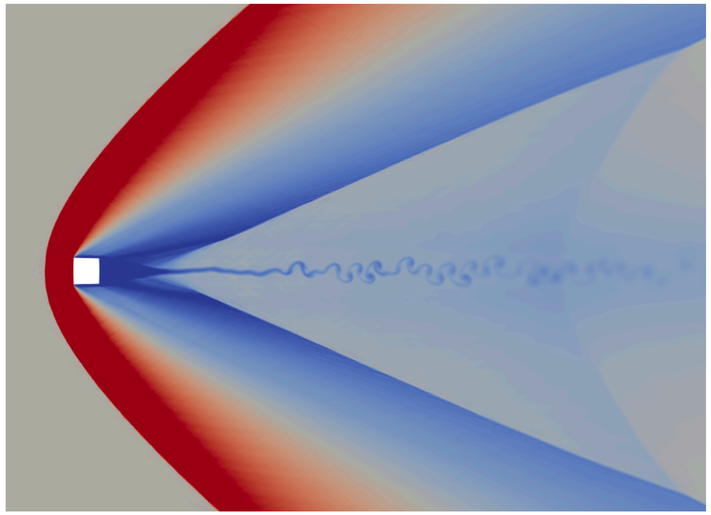
# Examples of entropy stable DG simulations



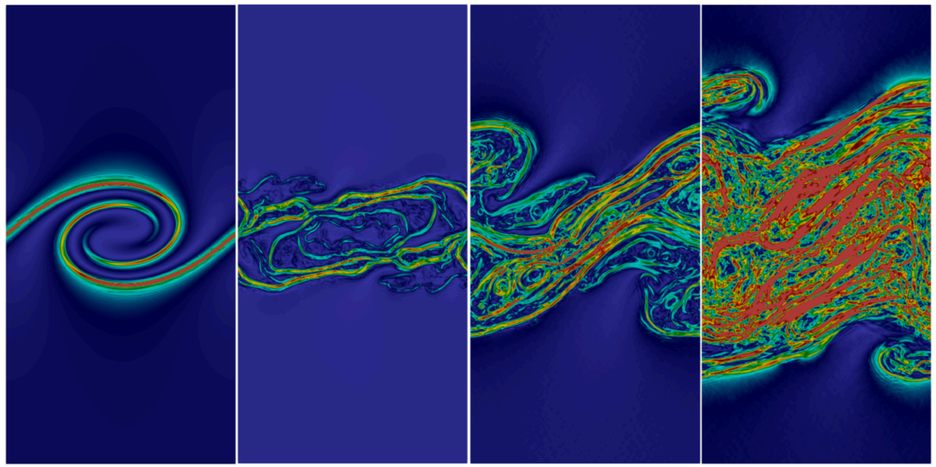

Parsani et al. (2021)



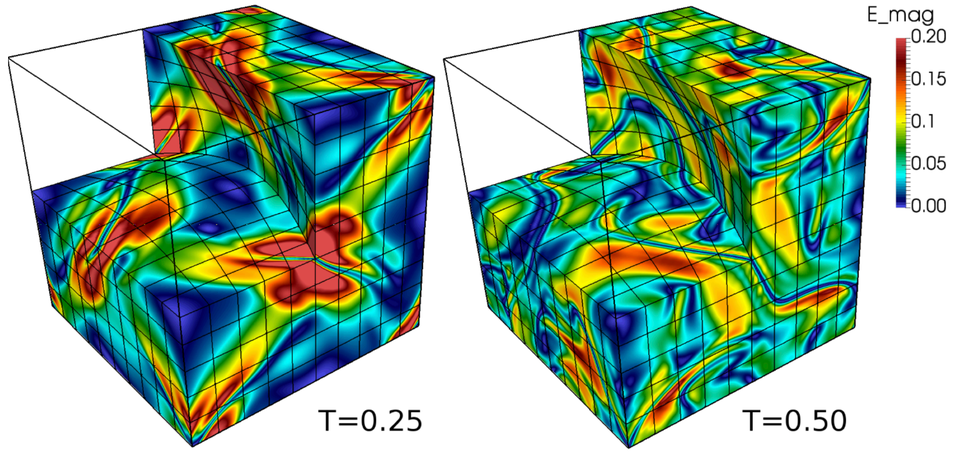
**Chan** et al (2022)



**Chan**, Lin, Warburton (2022)



Rueda-Ramirez, Hindenlang, **Chan**, Gassner (2023)



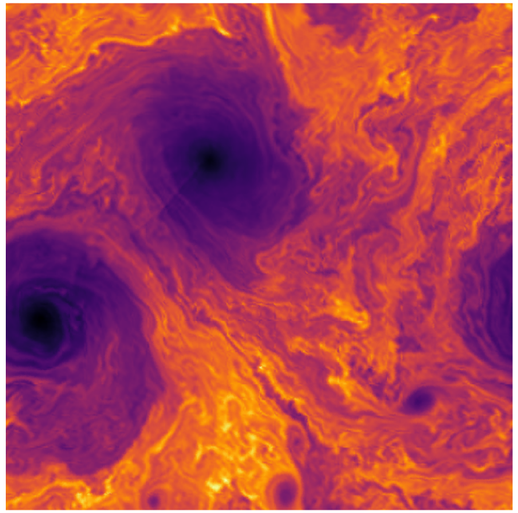

Bohm et al. (2020)

All of these high order simulations run stably *with no additional limiting or artificial viscosity.*

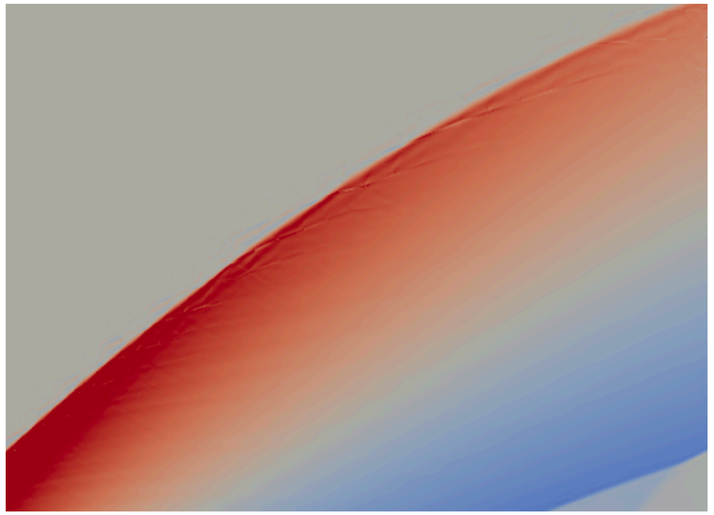
# Examples of entropy stable DG simulations



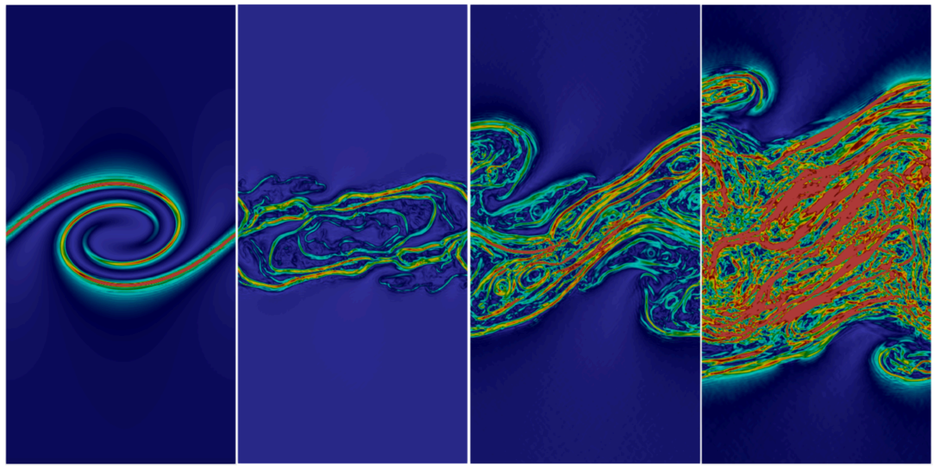

Parsani et al. (2021)



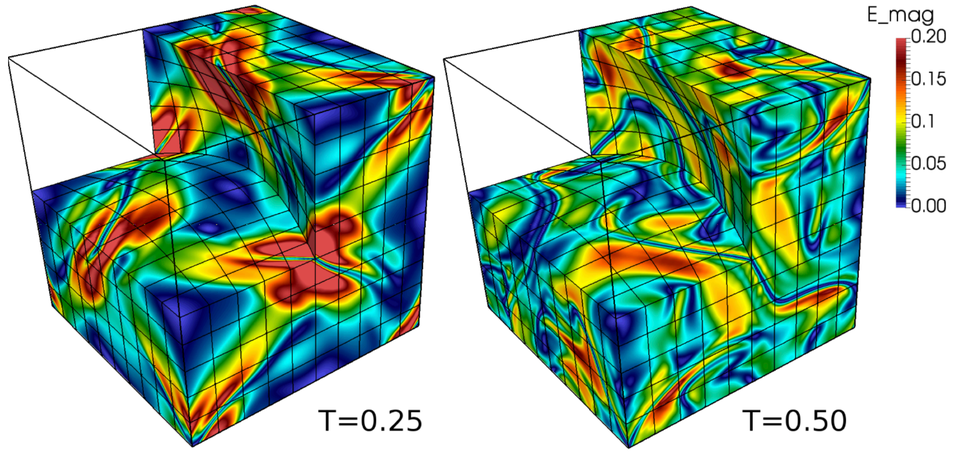
**Chan** et al (2022)



**Chan**, Lin, Warburton (2022)



Rueda-Ramirez, Hindenlang, **Chan**, Gassner (2023)




Bohm et al. (2020)

All of these high order simulations run stably *with no additional limiting or artificial viscosity.*

# An introduction to entropy stable nodal DG methods

# Summary: entropy stable DG methods

**Enforce a high order accurate *semi-discrete* cell entropy inequality**

$$\frac{\partial \boldsymbol{u}}{\partial t} + \frac{\partial \boldsymbol{f}(\boldsymbol{u})}{\partial x} = 0 \quad \longrightarrow \quad \int_a^b \frac{\partial S(\boldsymbol{u})}{\partial t} + \left(\boldsymbol{v}^T \boldsymbol{f}(\boldsymbol{u}) - \psi(\boldsymbol{u})\right)\Big|_a^b \leq 0$$

- A generalization of energy balance for nonlinear conservation laws (Burgers', shallow water, compressible Euler and Navier-Stokes, MHD)
- Proof of the continuous entropy inequality requires the chain rule: does not hold under discretizations (due to e.g., interpolation, inexact quadrature).

# Summary: entropy stable DG methods

## Enforce a high order accurate *semi-discrete* cell entropy inequality

$$\frac{\partial \boldsymbol{u}}{\partial t} + \frac{\partial \boldsymbol{f}(\boldsymbol{u})}{\partial x} = 0 \quad \longrightarrow \quad \int_a^b \frac{\partial S(\boldsymbol{u})}{\partial t} + \left(\boldsymbol{v}^T \boldsymbol{f}(\boldsymbol{u}) - \psi(\boldsymbol{u})\right)\Big|_a^b \leq 0$$

$S(\boldsymbol{u})$ is a convex entropy

- A generalization of energy balance for nonlinear conservation laws (Burgers', shallow water, compressible Euler and Navier-Stokes, MHD)
- Proof of the continuous entropy inequality requires the chain rule: does not hold under discretizations (due to e.g., interpolation, inexact quadrature).

# Summary: entropy stable DG methods

## Enforce a high order accurate *semi-discrete* cell entropy inequality

$$\frac{\partial \boldsymbol{u}}{\partial t} + \frac{\partial \boldsymbol{f}(\boldsymbol{u})}{\partial x} = 0 \quad \longrightarrow \quad \int_a^b \frac{\partial S(\boldsymbol{u})}{\partial t} + \left(\boldsymbol{v}^T \boldsymbol{f}(\boldsymbol{u}) - \psi(\boldsymbol{u})\right)\Big|_a^b \leq 0$$

$S(\boldsymbol{u})$ is a convex entropy

$\boldsymbol{v}(\boldsymbol{u}) = \dfrac{\partial S}{\partial \boldsymbol{u}}$ are entropy variables

- A generalization of energy balance for nonlinear conservation laws (Burgers', shallow water, compressible Euler and Navier-Stokes, MHD)
- Proof of the continuous entropy inequality requires the chain rule: does not hold under discretizations (due to e.g., interpolation, inexact quadrature).

# Summary: entropy stable DG methods

## Enforce a high order accurate *semi-discrete* cell entropy inequality

$$\frac{\partial \boldsymbol{u}}{\partial t} + \frac{\partial \boldsymbol{f}(\boldsymbol{u})}{\partial x} = 0 \quad \longrightarrow \quad \int_a^b \frac{\partial S(\boldsymbol{u})}{\partial t} + \left(\boldsymbol{v}^T \boldsymbol{f}(\boldsymbol{u}) - \psi(\boldsymbol{u})\right)\Big|_a^b \leq 0$$

entropy potential

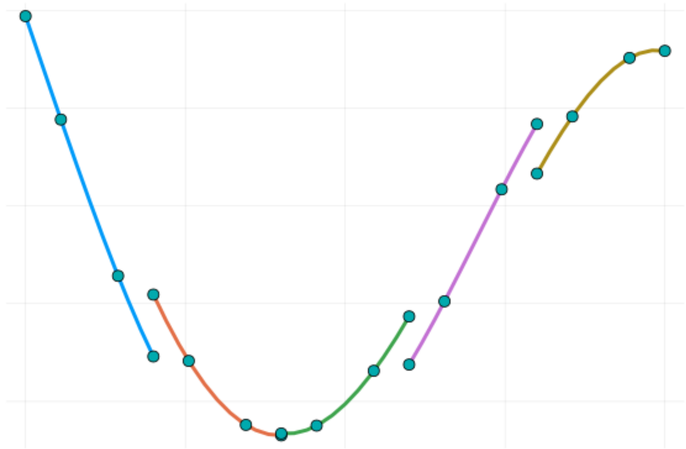
$S(\boldsymbol{u})$ is a convex entropy

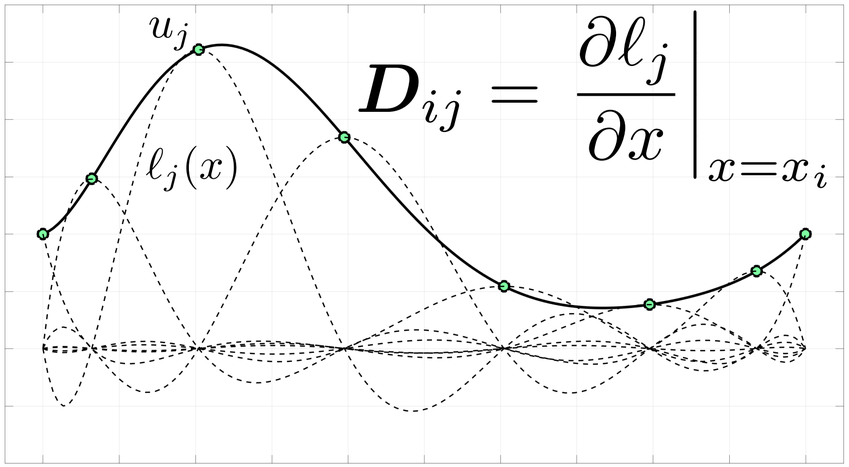
$\boldsymbol{v}(\boldsymbol{u}) = \dfrac{\partial S}{\partial \boldsymbol{u}}$ are entropy variables

- A generalization of energy balance for nonlinear conservation laws (Burgers', shallow water, compressible Euler and Navier-Stokes, MHD)
- Proof of the continuous entropy inequality requires the chain rule: does not hold under discretizations (due to e.g., interpolation, inexact quadrature).

# A brief introduction to *nodal* DG methods

## Also known as “collocation” DG methods



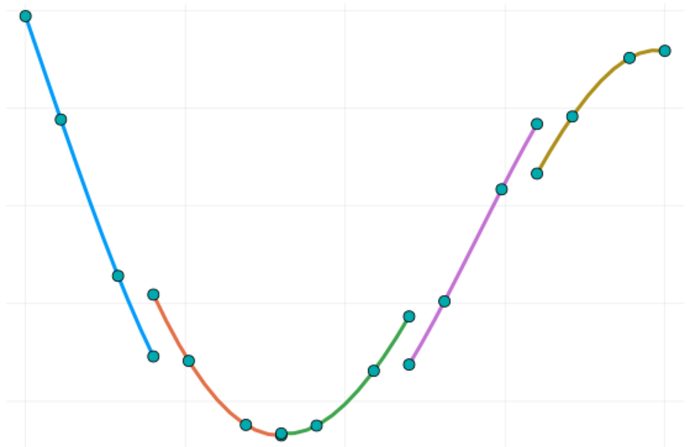


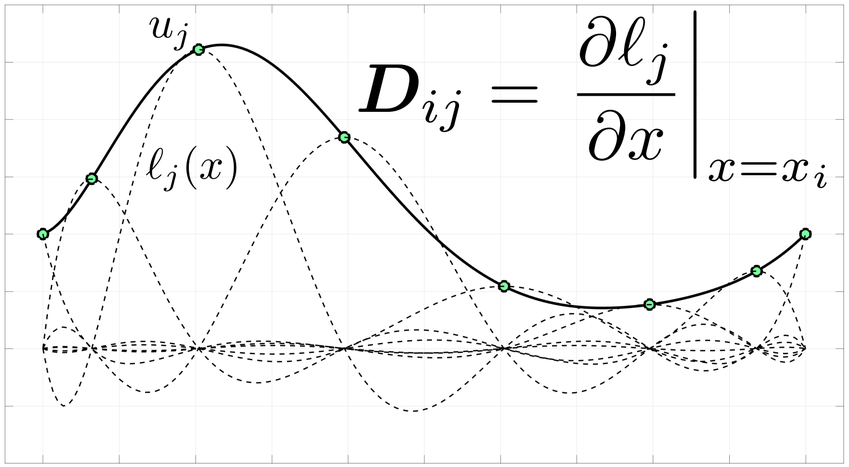

$$\int_{D^k} \left( \frac{\partial \boldsymbol{u}}{\partial t} + \frac{\partial \boldsymbol{f}(\boldsymbol{u})}{\partial x} \right) \ell_i = 0$$

- Variational formulation restricted to a single element $D^k$: multiply by a nodal (Lagrange) basis function, integrate by parts to introduce a *numerical flux*
- Collocation assumption: degree $N$ polynomials, with $(N+1)$-point Gauss-Lobatto quadrature nodes are used for both interpolation and integration.

# A brief introduction to *nodal* DG methods

## Also known as “collocation” DG methods



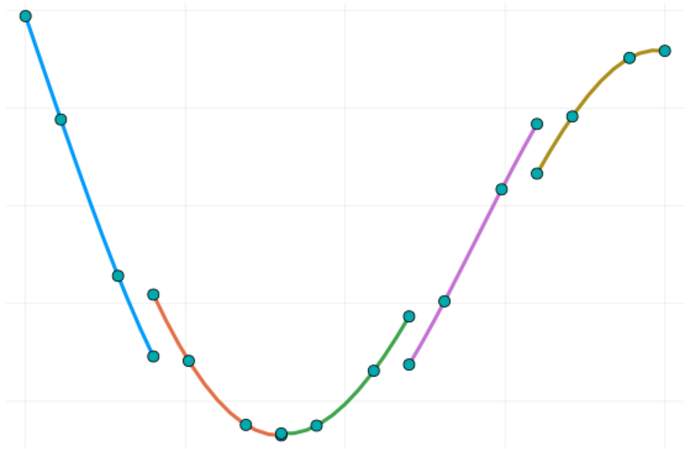


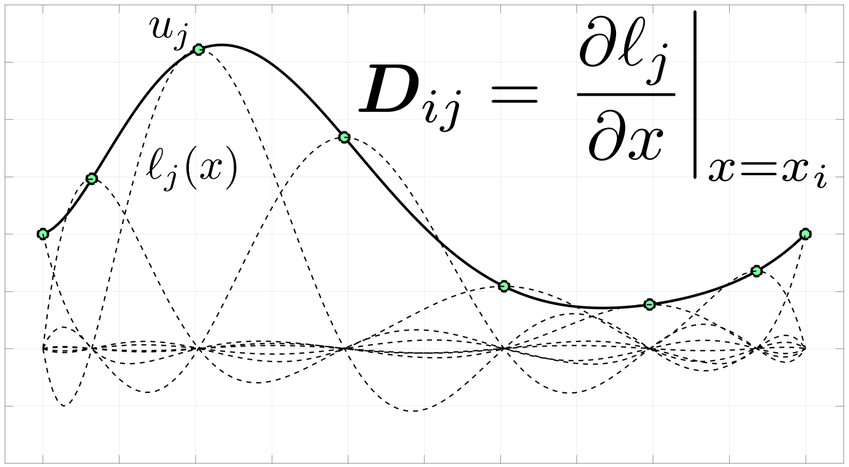

$$\int_{D^k} \left( \frac{\partial \boldsymbol{u}}{\partial t} \ell_i - \boldsymbol{f}(\boldsymbol{u}) \frac{\partial \ell_i}{\partial x} \right) + \int_{\partial D^k} \boldsymbol{f}^*(\boldsymbol{u}^+, \boldsymbol{u}^-) n \ell_i = 0$$

- Variational formulation restricted to a single element $D^k$: multiply by a nodal (Lagrange) basis function, integrate by parts to introduce a *numerical flux*
- Collocation assumption: degree $N$ polynomials, with $(N+1)$-point Gauss-Lobatto quadrature nodes are used for both interpolation and integration.

# A brief introduction to *nodal* DG methods

## Also known as “collocation” DG methods






$$\int_{D^k} \left( \frac{\partial \boldsymbol{u}}{\partial t} + \frac{\partial \boldsymbol{f}(\boldsymbol{u})}{\partial x} \right) \ell_i + \int_{\partial D^k} (\boldsymbol{f}^*(\boldsymbol{u}^+, \boldsymbol{u}^-) - \boldsymbol{f}(\boldsymbol{u}^-)) n \ell_i = 0$$

- Variational formulation restricted to a single element $D^k$: multiply by a nodal (Lagrange) basis function, integrate by parts to introduce a *numerical flux*
- Collocation assumption: degree $N$ polynomials, with $(N+1)$-point Gauss-Lobatto quadrature nodes are used for both interpolation and integration.

# Matrix form of a nodal DG method

$$\int_{D^k} \left( \frac{\partial \boldsymbol{u}}{\partial t} + \frac{\partial \boldsymbol{f}(\boldsymbol{u})}{\partial x} \right) \ell_i + \int_{\partial D^k} (\boldsymbol{f}^*(\boldsymbol{u}^+, \boldsymbol{u}^-) - \boldsymbol{f}(\boldsymbol{u}^-)) n \ell_i = 0$$

# Matrix form of a nodal DG method

$$\int_{D^k} \left( \frac{\partial \boldsymbol{u}}{\partial t} + \frac{\partial \boldsymbol{f}(\boldsymbol{u})}{\partial x} \right) \ell_i + \int_{\partial D^k} (\boldsymbol{f}^*(\boldsymbol{u}^+, \boldsymbol{u}^-) - \boldsymbol{f}(\boldsymbol{u}^-)) n \ell_i = 0$$

$$\mathbf{M} \frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t}$$

Mass matrix **M** is diagonal
under a collocation assumption

# Matrix form of a nodal DG method

$$\int_{D^k} \left( \frac{\partial \boldsymbol{u}}{\partial t} + \frac{\partial \boldsymbol{f}(\boldsymbol{u})}{\partial x} \right) \ell_i + \int_{\partial D^k} (\boldsymbol{f}^*(\boldsymbol{u}^+, \boldsymbol{u}^-) - \boldsymbol{f}(\boldsymbol{u}^-)) n \ell_i = 0$$

$$\mathbf{M} \frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} + \mathbf{Q}\boldsymbol{f}(\mathbf{u})$$

Mass matrix **M** is diagonal under a collocation assumption

$$\mathbf{Q}_{ij} = \int_{D^k} \frac{\partial \ell_j}{\partial x} \ell_i,$$

Exact for Gauss-Lobatto quadrature

# Matrix form of a nodal DG method

$$\int_{D^k} \left( \frac{\partial \boldsymbol{u}}{\partial t} + \frac{\partial \boldsymbol{f}(\boldsymbol{u})}{\partial x} \right) \ell_i + \int_{\partial D^k} (\boldsymbol{f}^*(\boldsymbol{u}^+, \boldsymbol{u}^-) - \boldsymbol{f}(\boldsymbol{u}^-)) n \ell_i = 0$$

$$\mathbf{M} \frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} + \mathbf{Q}\boldsymbol{f}(\mathbf{u}) + \mathbf{E}^T \mathbf{B} \left( \boldsymbol{f}^* \left( \mathbf{u}^+, \mathbf{u}^- \right) - \boldsymbol{f}(\mathbf{u}^-) \right) = \mathbf{0}.$$

Mass matrix **M** is diagonal under a collocation assumption

$$\mathbf{Q}_{ij} = \int_{D^k} \frac{\partial \ell_j}{\partial x} \ell_i,$$

Exact for Gauss-Lobatto quadrature

$$\mathbf{B} = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}$$

Boundary matrix of outward normals

Face interpolation matrix $\mathbf{E} = \begin{bmatrix} \ell_1(-1) & \dots & \ell_{N+1}(-1) \\ \ell_1(1) & \dots & \ell_{N+1}(1) \end{bmatrix} = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & \dots & 0 & 1 \end{bmatrix}.$

# “Flux differencing” nodal DG methods

**Key idea: incorporate numerical fluxes into volume terms.**

- Reformulate standard DG volume terms to expose a central flux.

$$(\mathbf{Q}\boldsymbol{f}(\mathbf{u}))_i = \sum_j \mathbf{Q}_{ij}\boldsymbol{f}(\mathbf{u}_j) = 2\sum_j \mathbf{Q}_{ij}\frac{\boldsymbol{f}(\mathbf{u}_i)+\boldsymbol{f}(\mathbf{u}_j)}{2}$$

$$\int_{D^k} \frac{\partial \boldsymbol{f}(\boldsymbol{u})}{\partial x}\ell_i$$

Gassner, Winters, Kopriva (2016). *Split Form Nodal Discontinuous Galerkin Schemes with Summation-By-Parts Property for the Compressible Euler Equations*

# “Flux differencing” nodal DG methods

**Key idea: incorporate numerical fluxes into volume terms.**

- Reformulate standard DG volume terms to expose a central flux.

$$(\mathbf{Q}\boldsymbol{f}(\mathbf{u}))_i = \sum_j \mathbf{Q}_{ij}\boldsymbol{f}(\mathbf{u}_j) = 2\sum_j \mathbf{Q}_{ij}\frac{\boldsymbol{f}(\mathbf{u}_i)+\boldsymbol{f}(\mathbf{u}_j)}{2}$$

$$\int_{D^k} \frac{\partial \boldsymbol{f}(\boldsymbol{u})}{\partial x}\ell_i \qquad \sum_j \mathbf{Q}_{ij} = 0$$

Gassner, Winters, Kopriva (2016). *Split Form Nodal Discontinuous Galerkin Schemes with Summation-By-Parts Property for the Compressible Euler Equations*

# “Flux differencing” nodal DG methods

**Key idea: incorporate numerical fluxes into volume terms.**

- Reformulate standard DG volume terms to expose a central flux.

$$(\mathbf{Q}\boldsymbol{f}(\mathbf{u}))_i = \sum_j \mathbf{Q}_{ij}\boldsymbol{f}(\mathbf{u}_j) = 2\sum_j \mathbf{Q}_{ij}\frac{\boldsymbol{f}(\mathbf{u}_i)+\boldsymbol{f}(\mathbf{u}_j)}{2}$$

$$\int_{D^k} \frac{\partial \boldsymbol{f}(\boldsymbol{u})}{\partial x}\ell_i \qquad \sum_j \mathbf{Q}_{ij} = 0$$

- What if we swapped the central flux for another symmetric flux?
  - Can recover split forms: e.g., $\frac{1}{2}\left(\frac{\partial \rho u}{\partial x} + \rho\frac{\partial u}{\partial x} + u\frac{\partial \rho}{\partial x}\right) \Longrightarrow \{\{\rho\}\}\{\{u\}\}$

Gassner, Winters, Kopriva (2016). *Split Form Nodal Discontinuous Galerkin Schemes with Summation-By-Parts Property for the Compressible Euler Equations*

# Entropy stable nodal DG methods

## Key idea: incorporate numerical fluxes into volume terms.

- Utilize *entropy conservative* finite volume fluxes (Tadmor 1987):

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

- Replace the central flux with an entropy conservative flux

$$\left(\mathbf{Q}\boldsymbol{f}(\mathbf{u})\right)_i = 2\sum_j \mathbf{Q}_{ij}\frac{\boldsymbol{f}(\mathbf{u}_j)+\boldsymbol{f}(\mathbf{u}_i)}{2} \approx 2\sum_j \mathbf{Q}_{ij}\boldsymbol{f}_{EC}(\mathbf{u}_i,\mathbf{u}_j).$$

- Compact notation using the Hadamard product

$$2\sum_j \mathbf{Q}_{ij}\boldsymbol{f}_{EC}(\mathbf{u}_i,\mathbf{u}_j) = \left(2\left(\mathbf{Q}\circ\mathbf{F}\right)\mathbf{1}\right)_i, \quad \mathbf{F}_{ij} = \boldsymbol{f}_{EC}(\mathbf{u}_i,\mathbf{u}_j).$$

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

# An example of an entropy conservative flux

## 1D compressible Euler equations

- Explicit formulas which are consistent and satisfy Tadmor's entropy conservation condition
- Utilizes a numerically stable procedure to evaluate the logarithmic average
- High arithmetic intensity

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

$$
\text{Average:} \quad \{\!\{u\}\!\} = \frac{1}{2}(u_L + u_R)
$$

$$
\text{Logarithmic average:} \quad \{\!\{u\}\!\}^{\log} = \frac{u_L - u_R}{\log u_L - \log u_R}
$$

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

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

# Entropy conservative/stable nodal DG methods

**Key idea: incorporate numerical fluxes into volume terms.**

- Modify volume terms to incorporate entropy conservative fluxes:

$$\mathbf{M}\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} + \underbrace{\mathbf{Q}\boldsymbol{f}(\mathbf{u})}_{\text{Nodal DG}} + \mathbf{E}^T\mathbf{B}\left(\boldsymbol{f}^*\left(\mathbf{u}^+,\mathbf{u}^-\right) - \boldsymbol{f}(\mathbf{u}^-)\right) = \mathbf{0}$$

# Entropy conservative/stable nodal DG methods

## Key idea: incorporate numerical fluxes into volume terms.

- Modify volume terms to incorporate entropy conservative fluxes:

$$\mathbf{M}\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} + \underbrace{2\left(\mathbf{Q}\circ\mathbf{F}\right)\mathbf{1}}_{\text{Entropy stable nodal DG}} + \mathbf{E}^T\mathbf{B}\left(\boldsymbol{f}^*\left(\mathbf{u}^+,\mathbf{u}^-\right) - \boldsymbol{f}\left(\mathbf{u}^-\right)\right) = \mathbf{0}$$

# Entropy conservative/stable nodal DG methods

**Key idea: incorporate numerical fluxes into volume terms.**

- Modify volume terms to incorporate entropy conservative fluxes:

$$\mathbf{M}\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} + \underbrace{2\left(\mathbf{Q}\circ\mathbf{F}\right)\mathbf{1}}_{\text{Entropy stable nodal DG}} + \mathbf{E}^T\mathbf{B}\left(\boldsymbol{f}^*\left(\mathbf{u}^+,\mathbf{u}^-\right) - \boldsymbol{f}\left(\mathbf{u}^-\right)\right) = \mathbf{0}$$

- Assume that the interface flux is entropy conservative and that the differentiation matrix satisfies the *summation-by-parts* property

$$\mathbf{Q} + \mathbf{Q}^T = \mathbf{E}^T\mathbf{B}\mathbf{E}$$

- Then, a quadrature version of the following cell entropy equality holds:

$$\int_{D^k}\frac{\partial S(\boldsymbol{u})}{\partial t} + \int_{\partial D^k}\left(\boldsymbol{v}^T\boldsymbol{f}^*(\boldsymbol{u}^+,\boldsymbol{u}^-) - \psi\left(\boldsymbol{u}\right)\right)n = 0.$$

# Entropy conservative/stable nodal DG methods

**Key idea: incorporate numerical fluxes into volume terms.**

- Modify volume terms to incorporate entropy conservative fluxes:

$$\mathbf{M}\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} + \underbrace{2\left(\mathbf{Q}\circ\mathbf{F}\right)\mathbf{1}}_{\text{Entropy stable nodal DG}} + \mathbf{E}^T\mathbf{B}\left(\boldsymbol{f}^*\left(\mathbf{u}^+,\mathbf{u}^-\right) - \boldsymbol{f}\left(\mathbf{u}^-\right)\right) = \mathbf{0}$$

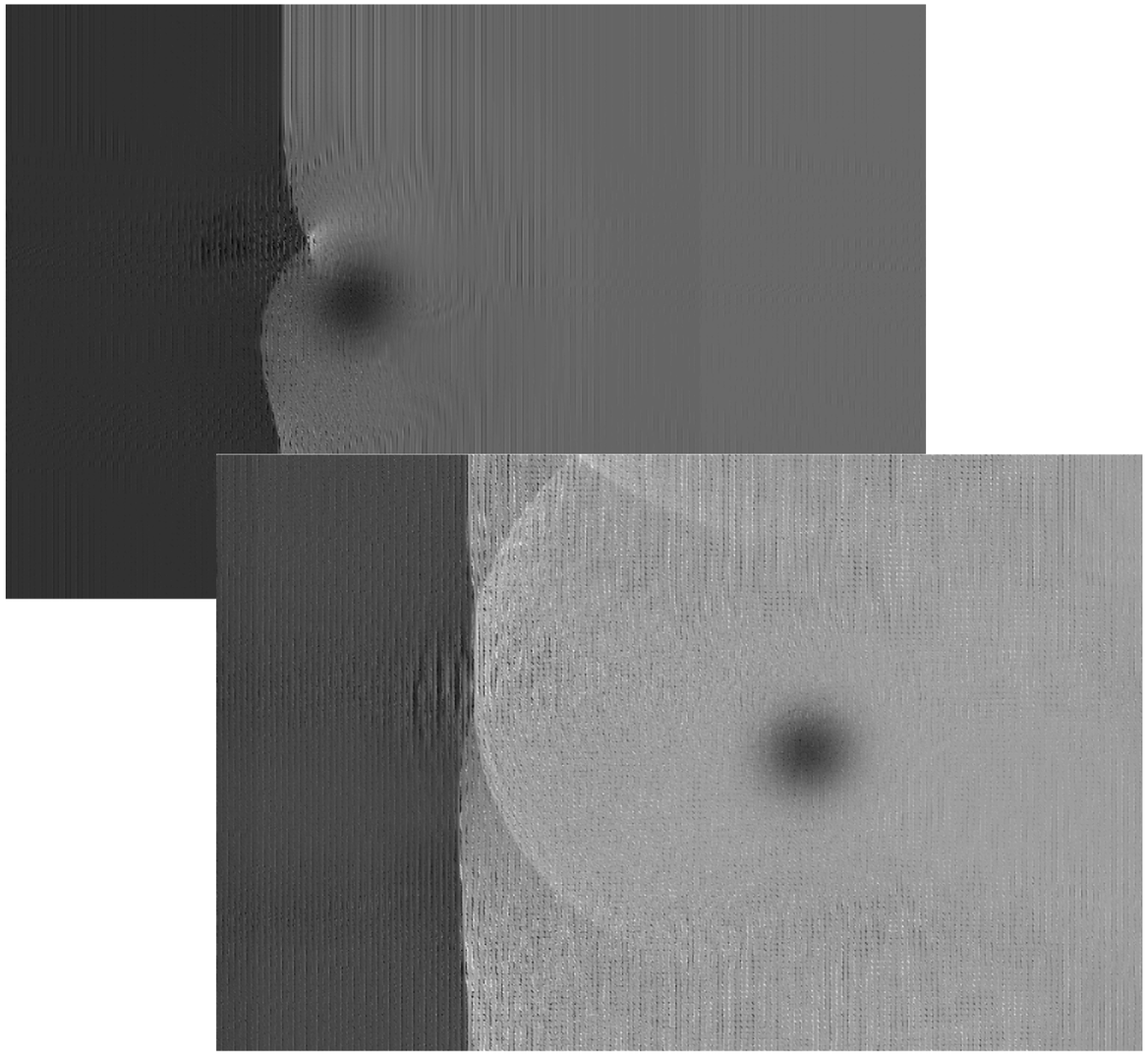
- Assume that the interface flux is entropy dissipative (e.g., a local Lax-Friedrichs flux) and that the differentiation matrix satisfies the *summation-by-parts* property

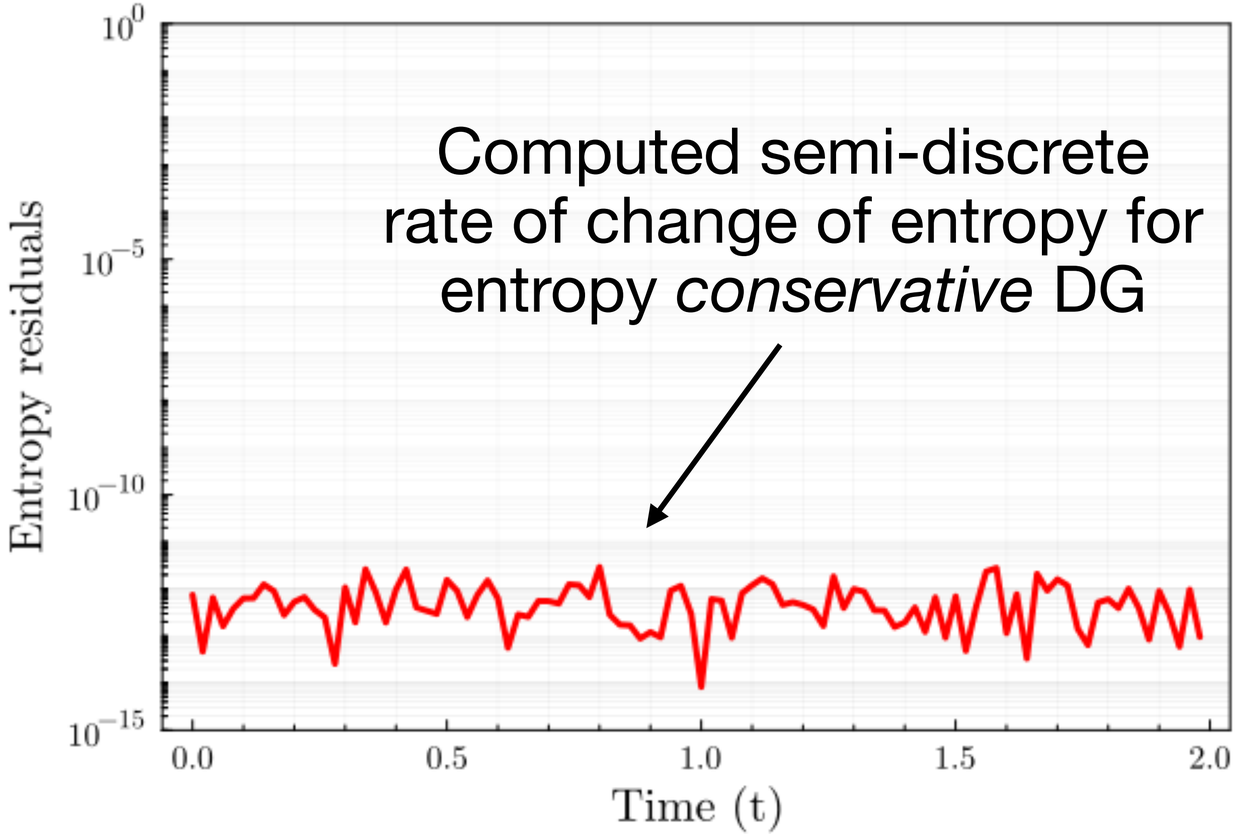
$$\mathbf{Q} + \mathbf{Q}^T = \mathbf{E}^T\mathbf{B}\mathbf{E}$$

- Then, a quadrature version of the following cell entropy inequality holds:

$$\int_{D^k} \frac{\partial S(\boldsymbol{u})}{\partial t} + \int_{\partial D^k} \left(\boldsymbol{v}^T\boldsymbol{f}^*(\boldsymbol{u}^+,\boldsymbol{u}^-) - \psi\left(\boldsymbol{u}\right)\right) n \leq 0.$$

# Example of entropy *conservative* DG

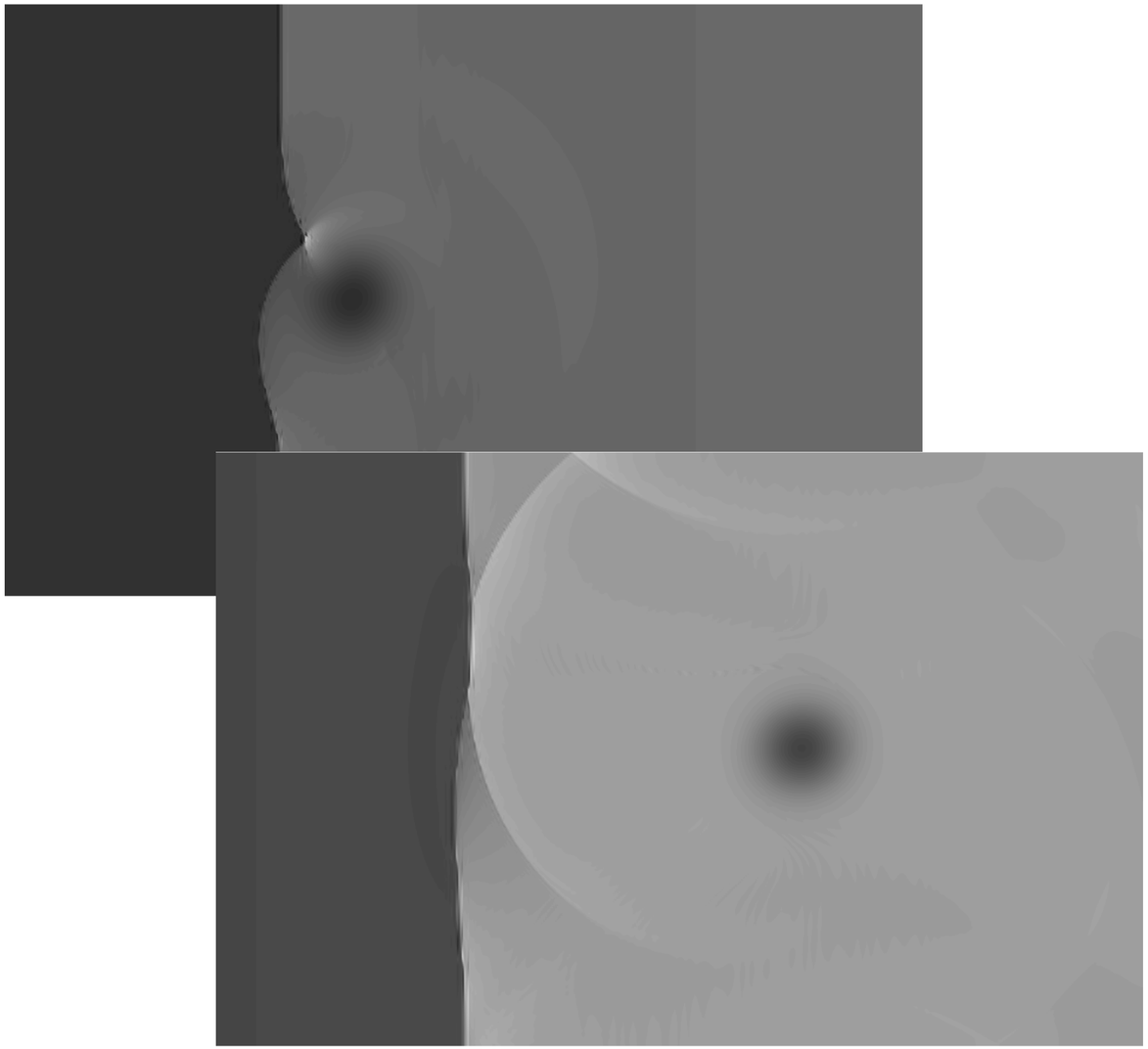


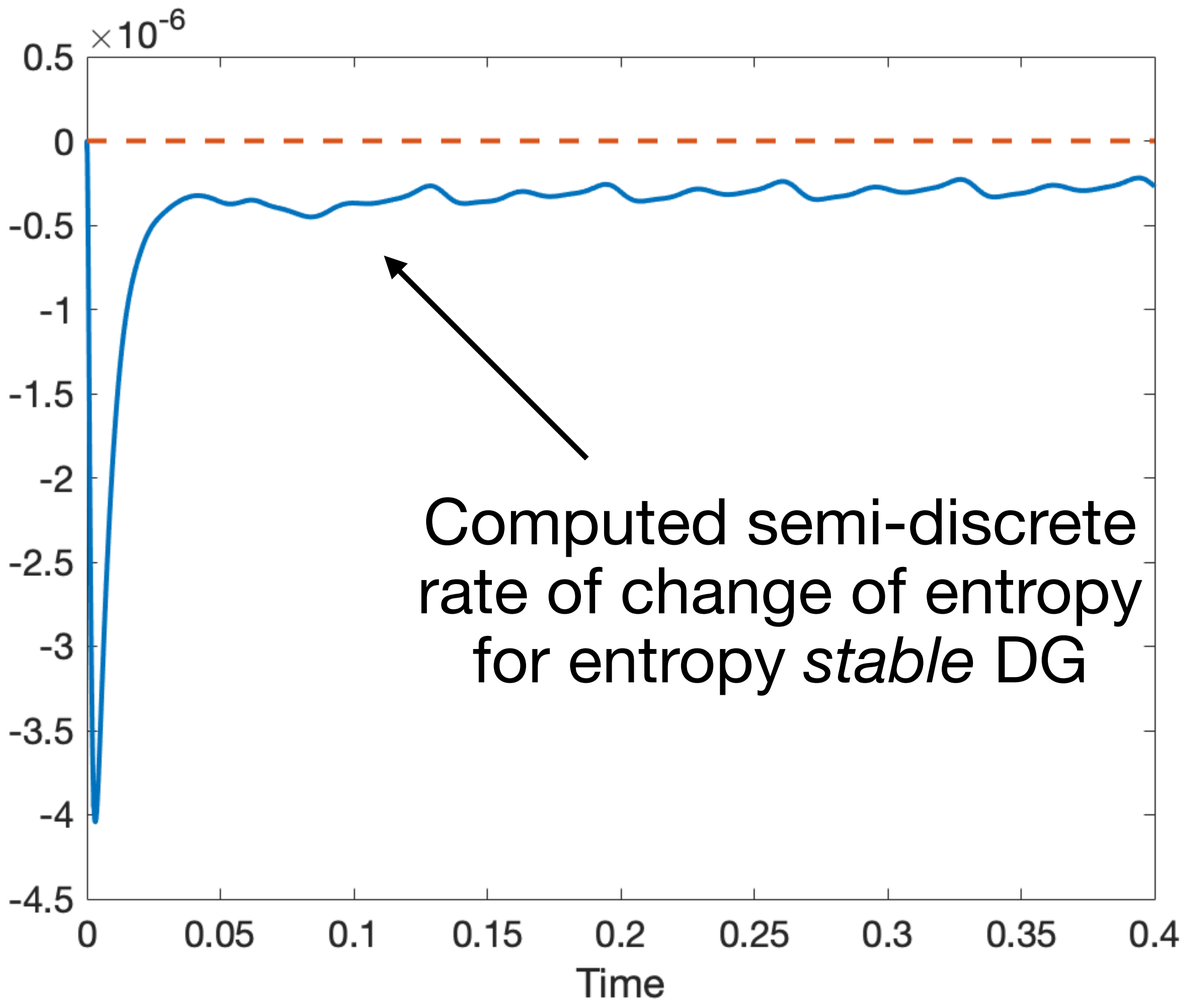



Solutions have spurious oscillations, but run stably *with no upwinding, additional limiting, or artificial viscosity.*

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

# Example of entropy *stable* DG






Spurious oscillations damped by interface dissipation; still no *additional limiting, or artificial viscosity.*

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

# Outline of this talk

**Entropy stable nodal DG methods and positivity preservation**

**Entropy stable "modal" DG formulations**

Cut-cell DG, entropy stable reduced order modeling

**Current work: entropy stable DG methods via knapsack limiting**

# Outline of this talk

**Entropy stable nodal DG methods and positivity preservation**

More restrictions on discretization and physics, but stronger theory

**Entropy stable "modal" DG formulations**

Cut-cell DG, entropy stable reduced order modeling

**Current work: entropy stable DG methods via knapsack limiting**

# Outline of this talk

**Entropy stable nodal DG methods and positivity preservation**

More restrictions on discretization and physics, but stronger theory

Flexibility with respect to the discretization

**Entropy stable "modal" DG formulations**

Cut-cell DG, entropy stable reduced order modeling

**Current work: entropy stable DG methods via knapsack limiting**

# Outline of this talk

**Entropy stable nodal DG methods and positivity preservation**

More restrictions on discretization and physics, but stronger theory

Flexibility with respect to the discretization

**Entropy stable "modal" DG formulations**

Cut-cell DG, entropy stable reduced order modeling

Flexibility with respect to the physics

**Current work: entropy stable DG methods via knapsack limiting**

# Positivity-preserving entropy stable nodal DG methods for compressible Navier-Stokes

# Our contribution: positivity preserving entropy stable nodal DG

## Application: compressible Navier-Stokes equations

- Entropy conservative fluxes involve logarithms of density, pressure; requires *positivity.*
- New positivity-preserving techniques:
  - guarantees positivity under a CFL condition
  - retains *subcell resolution*
- Old idea: blend the robust low order method with a high order (entropy stable) method.
  - Entropy stability allows us to enforce *weaker bounds* to preserve accuracy.



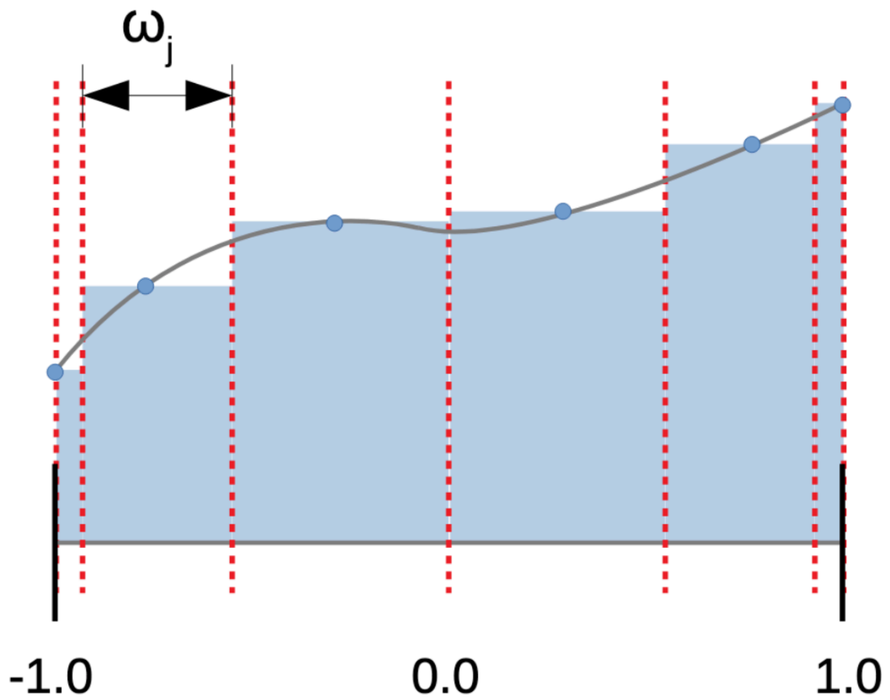
Entropy stable schemes do not guarantee positivity of density or pressure!

Lin, Chan, and Tomas (2023). *A positivity preserving strategy for entropy stable DG discretizations of the compressible Euler and Navier-Stokes equations*

# Our contribution: positivity preserving entropy stable nodal DG

## Application: compressible Navier-Stokes equations

- Entropy conservative fluxes involve logarithms of density, pressure; requires *positivity.*
- New positivity-preserving techniques:
  - guarantees positivity under a CFL condition
  - retains *subcell resolution*
- Old idea: blend the robust low order method with a high order (entropy stable) method.
  - Entropy stability allows us to enforce *weaker bounds* to preserve accuracy.




Interpretation of Gauss-Lobatto nodal DG as a subcell finite volume scheme (figure from Hennemann et al 2021)

Lin, Chan, and Tomas (2023). *A positivity preserving strategy for entropy stable DG discretizations of the compressible Euler and Navier-Stokes equations*

# Enforcing positivity: a first order subcell scheme

## Begin with forward Euler, extend to higher order using SSP-RK

First step: construct a robust invariant domain preserving first order scheme by adding sub-cell local Lax-Friedrichs type dissipation.

$$\mathbf{M}_{ii}\frac{\mathbf{u}_i^{k+1}-\mathbf{u}_i}{\Delta t}=\sum_{j\in N(i)}\mathbf{Q}_{ij}^T\boldsymbol{f}(\mathbf{u}_j)+\underbrace{\mathbf{d}_{ij}\left(\mathbf{u}_j-\mathbf{u}_i\right)}_{\text{algebraic dissipation}}+\begin{array}{c}\text{LxF interface}\\ \text{fluxes}\end{array}$$

$$\mathbf{d}_{ij}\geq\lambda_{\max}\left(\mathbf{u}_i,\mathbf{u}_j,\mathbf{Q}_{ij}\right)$$

# Enforcing positivity: a first order subcell scheme

## Begin with forward Euler, extend to higher order using SSP-RK

First step: construct a robust invariant domain preserving first order scheme by adding sub-cell local Lax-Friedrichs type dissipation.

$$\mathbf{M}_{ii}\frac{\mathbf{u}_i^{k+1}-\mathbf{u}_i}{\Delta t}=\sum_{j\in N(i)}\mathbf{Q}_{ij}^T\boldsymbol{f}(\mathbf{u}_j)+\underbrace{\mathbf{d}_{ij}\left(\mathbf{u}_j-\mathbf{u}_i\right)}_{\text{algebraic dissipation}}+\begin{array}{c}\text{LxF interface}\\ \text{fluxes}\end{array}$$

$$\mathbf{d}_{ij}\geq\lambda_{\max}\left(\mathbf{u}_i,\mathbf{u}_j,\mathbf{Q}_{ij}\right)$$

Note: use *sparsified* low order SBP matrices
to avoid excessive algebraic dissipation

# Reformulation as a convex combination

- Using SBP properties of $\mathbf{Q}$, reformulate $\mathbf{u}_i^{k+1}$ as a convex combination:

$$\frac{\mathbf{M}_{ii}}{\Delta t}\mathbf{u}_i^{k+1} = \left(\frac{\mathbf{M}_{ii}}{\Delta t} - \sum_{j\neq i} 2\mathbf{d}_{ij}\right)\mathbf{u}_i + \sum_{j\neq i} 2\mathbf{d}_{ij}\bar{\mathbf{u}}_{ij}\,. \qquad \text{(for interior nodes)}$$

Guermond, Popov, and Tomas (2019). “Invariant domain preserving discretization-independent schemes and convex limiting for hyperbolic systems.”

# Reformulation as a convex combination

- Using SBP properties of $\mathbf{Q}$, reformulate $\mathbf{u}_i^{k+1}$ as a convex combination:

$$\frac{\mathbf{M}_{ii}}{\Delta t}\mathbf{u}_i^{k+1} = \left(\frac{\mathbf{M}_{ii}}{\Delta t} - \sum_{j\neq i} 2\mathbf{d}_{ij}\right)\mathbf{u}_i + \sum_{j\neq i} 2\mathbf{d}_{ij}\bar{\mathbf{u}}_{ij}. \qquad \text{(for interior nodes)}$$

$$\bar{\mathbf{u}}_{ij} = \frac{1}{2}\left(\mathbf{u}_i + \mathbf{u}_j\right) - \frac{\mathbf{Q}_{ij}}{\mathbf{d}_{ij}}\left(\mathbf{f}_j - \mathbf{f}_i\right)$$

Guermond, Popov, and Tomas (2019). “Invariant domain preserving discretization-independent schemes and convex limiting for hyperbolic systems.”

# Reformulation as a convex combination

- Using SBP properties of $\mathbf{Q}$, reformulate $\mathbf{u}_i^{k+1}$ as a convex combination:

$$\frac{\mathbf{M}_{ii}}{\Delta t}\mathbf{u}_i^{k+1} = \left(\frac{\mathbf{M}_{ii}}{\Delta t} - \sum_{j\neq i} 2\mathbf{d}_{ij}\right)\mathbf{u}_i + \sum_{j\neq i} 2\mathbf{d}_{ij}\bar{\mathbf{u}}_{ij}. \qquad \text{(for interior nodes)}$$

- Solution update is positive (in fact, invariant domain preserving) if intermediate "bar states" $\bar{\mathbf{u}}_{ij}$ are positive and a CFL condition holds

$$\bar{\mathbf{u}}_{ij} = \frac{1}{2}\left(\mathbf{u}_i + \mathbf{u}_j\right) - \frac{\mathbf{Q}_{ij}}{\mathbf{d}_{ij}}\left(\mathbf{f}_j - \mathbf{f}_i\right)$$

$$\mathbf{d}_{ij} \geq \lambda_{\max}\left(\mathbf{u}_i, \mathbf{u}_j, \mathbf{Q}_{ij}\right)$$

Guermond, Popov, and Tomas (2019). "Invariant domain preserving discretization-independent schemes and convex limiting for hyperbolic systems."

# Reformulation as a convex combination

- Using SBP properties of $\mathbf{Q}$, reformulate $\mathbf{u}_i^{k+1}$ as a convex combination:

$$\frac{\mathbf{M}_{ii}}{\Delta t}\mathbf{u}_i^{k+1} = \left(\frac{\mathbf{M}_{ii}}{\Delta t} - \sum_{j\neq i} 2\mathbf{d}_{ij}\right)\mathbf{u}_i + \sum_{j\neq i} 2\mathbf{d}_{ij}\bar{\mathbf{u}}_{ij}. \qquad \text{(for interior nodes)}$$

- Solution update is positive (in fact, invariant domain preserving) if intermediate "bar states" $\bar{\mathbf{u}}_{ij}$ are positive and a CFL condition holds

$$\bar{\mathbf{u}}_{ij} = \frac{1}{2}\left(\mathbf{u}_i + \mathbf{u}_j\right) - \frac{\mathbf{Q}_{ij}}{\mathbf{d}_{ij}}\left(\mathbf{f}_j - \mathbf{f}_i\right)$$

$$\mathbf{d}_{ij} \geq \lambda_{\max}\left(\mathbf{u}_i, \mathbf{u}_j, \mathbf{Q}_{ij}\right)$$

$$\Delta t \leq \min_i \frac{\mathbf{M}_{ii}}{2\sum_{i\neq j}\mathbf{d}_{ij}}.$$

Guermond, Popov, and Tomas (2019). "Invariant domain preserving discretization-independent schemes and convex limiting for hyperbolic systems."

# Our contribution: extension to Navier-Stokes

- Repeat the process but incorporate viscous terms $\boldsymbol{\sigma}_j$ : yields new bar states

$$\mathbf{M}\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} = \sum_j \mathbf{Q}_{ij}^T \left(\mathbf{f}_j - \boldsymbol{\sigma}_j\right) + \mathbf{d}_{ij}\left(\mathbf{u}_j - \mathbf{u}_i\right) + \begin{matrix}\text{interface}\\ \text{fluxes}\end{matrix}$$

$$\rightarrow \quad \bar{\mathbf{u}}_{ij} = \frac{1}{2}\left(\mathbf{u}_i + \mathbf{u}_j\right) - \frac{\mathbf{Q}_{ij}}{\mathbf{d}_{ij}}\left(\left(\mathbf{f}_j - \boldsymbol{\sigma}_j\right) - \left(\mathbf{f}_i - \boldsymbol{\sigma}_i\right)\right)$$

- New bar states are positive under Zhang's (2017) modified wavespeed estimate, which incorporates the viscous stresses and heat flux $\boldsymbol{\tau}, \boldsymbol{q}$

$$\mathbf{d}_{ij} = \max\left(\beta(\mathbf{u}_i), \beta(\mathbf{u}_j), \lambda_{\max}(\mathbf{u}_i, \mathbf{u}_j, \mathbf{Q}_{ij})\right)$$

$$\beta(\boldsymbol{u}) > |\boldsymbol{v}\cdot\boldsymbol{n}| + \frac{1}{2\rho^2 e}\left(\sqrt{\rho^2(\boldsymbol{q}\cdot\boldsymbol{n})^2 + 2\rho^2 e\,\|\boldsymbol{\tau}\cdot\boldsymbol{n} - p\boldsymbol{n}\|}\right) + \rho\,|\boldsymbol{q}\cdot\boldsymbol{n}|$$

Chan, Lin, Warburton (2022). *Entropy stable modal DG schemes and wall boundary conditions for the compressible Navier-Stokes equations.*
Zhang (2017). *On positivity-preserving high order discontinuous Galerkin schemes for the compressible Navier-Stokes equations.*

# Putting it all together

- Element-wise blending of high and low order solution for $\ell \in [0,1]$

$$\mathbf{u}^{k+1} = (1-\ell)\mathbf{u}^{k+1,\text{low}} + \ell\mathbf{u}^{k+1,\text{high}}$$

- Impose minimal nodal bounds based on the low order solution

$$\rho \geq \alpha\rho^{\text{low}}, \quad p \geq \alpha p^{\text{low}} \qquad \alpha \in [0,1]$$

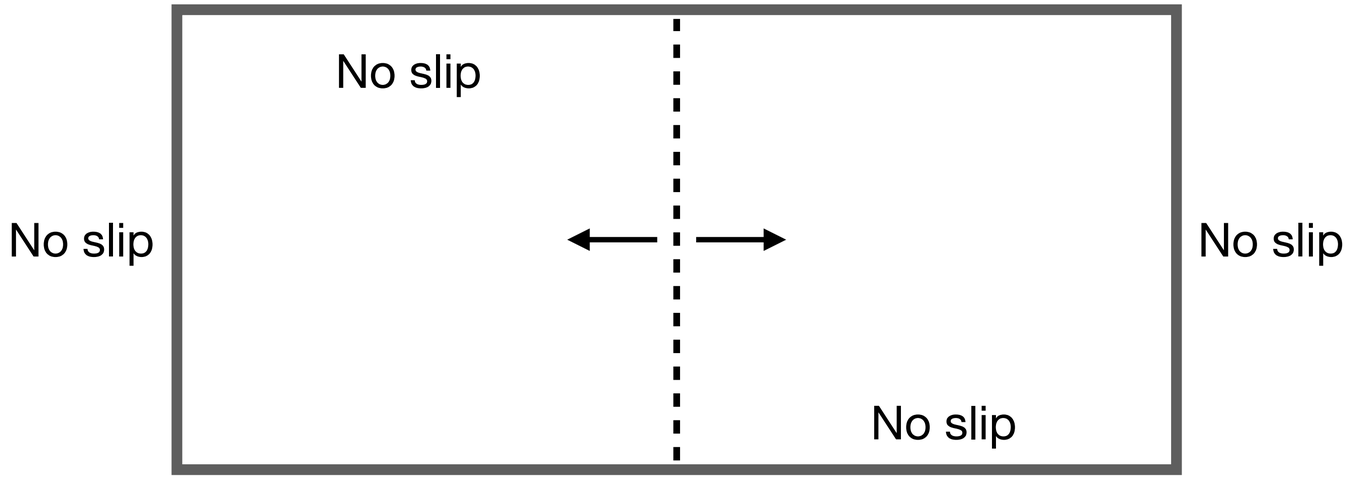
Relaxation factor

- Element-wise blending preserves:
  - Local conservation (if high and low order schemes use the same interface flux)
  - Entropy stability for inviscid systems
  - High order accuracy (in numerical experiments)

# Our contribution: positivity preserving entropy stable nodal DG

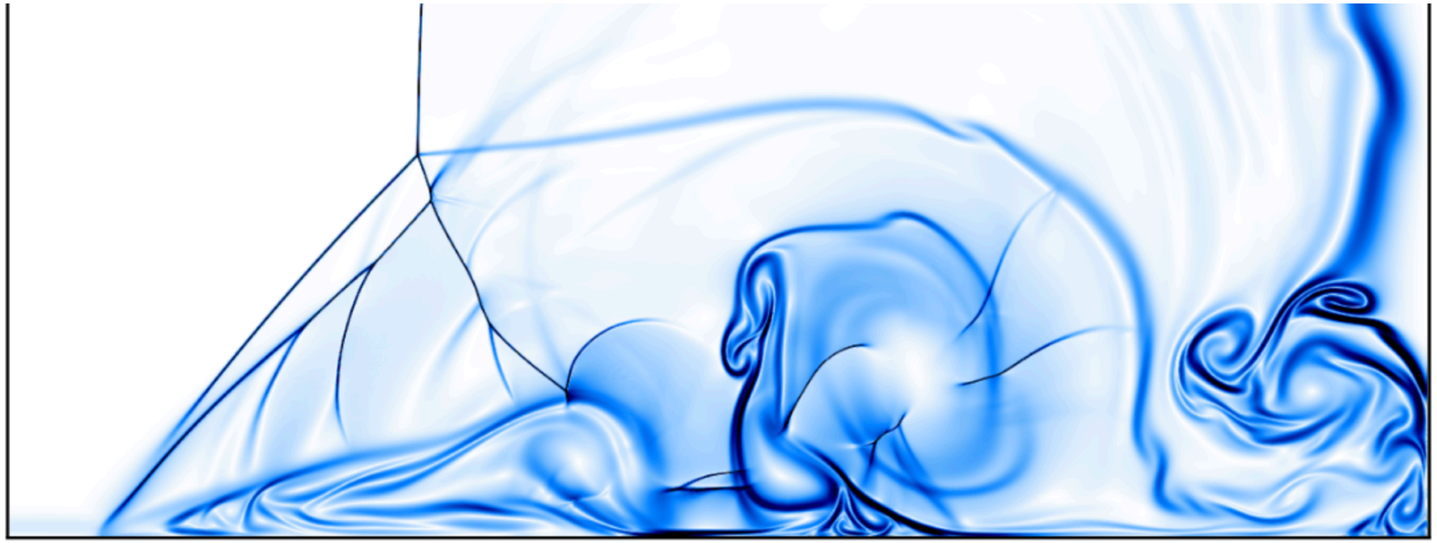
## Application: compressible Navier-Stokes equations

Daru-Tenaud shock tube: simplified problem with shock-boundary layer interaction




**Guermond et al. 2022**

Fine-grid reference solution
2nd order invariant-domain P1-FEM
Graded mesh, 512M dofs



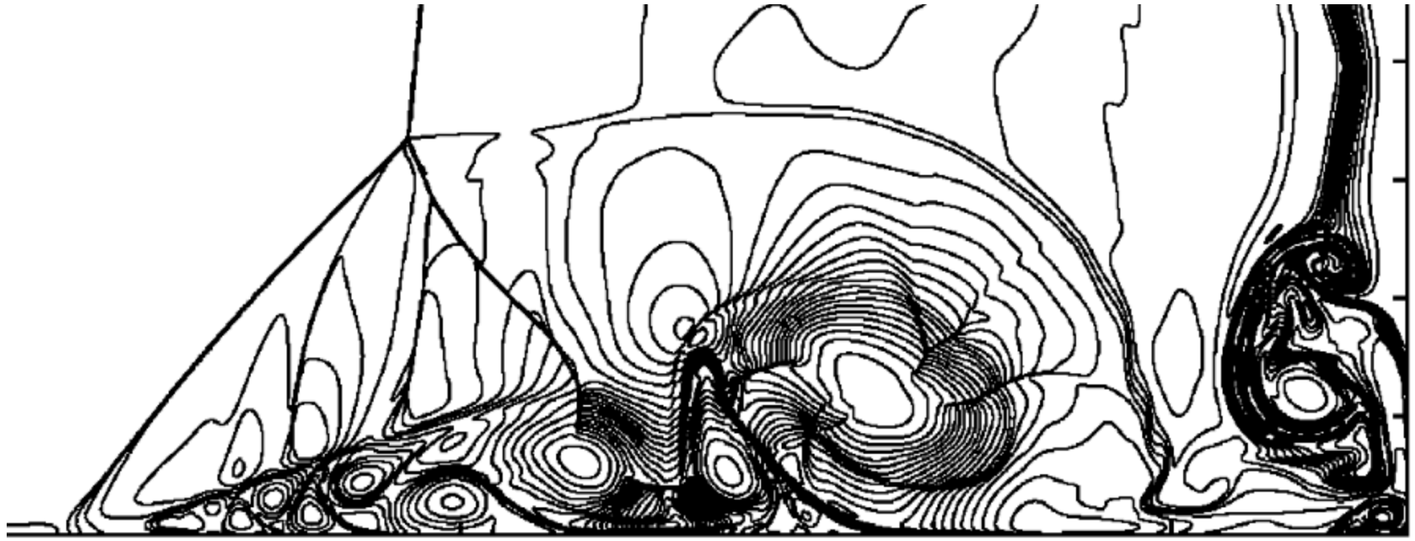
Daru, Tenaud (2009). *Numerical simulation of the viscous shock tube problem by using a high resolution monotonicity-preserving scheme*.
Guermond et al (2022). *On the implementation of a robust and efficient finite element-based parallel solver for the compressible Navier-Stokes equations*.

# Our contribution: positivity preserving entropy stable nodal DG

## Application: compressible Navier-Stokes equations

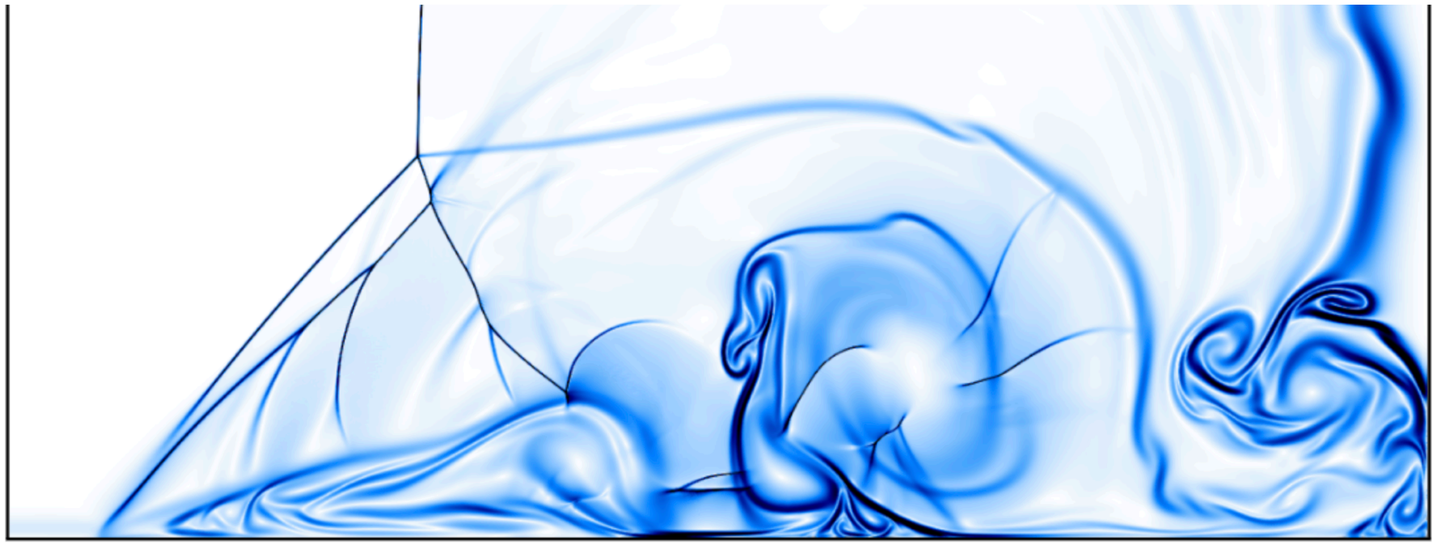
### Sjogreen, Yee 2003

Wavelet-based adaptive filter + RK4
4000 x 2000 grid points



### Guermond et al. 2022

Fine-grid reference solution
2nd order invariant-domain P1-FEM
Graded mesh, 512M dofs



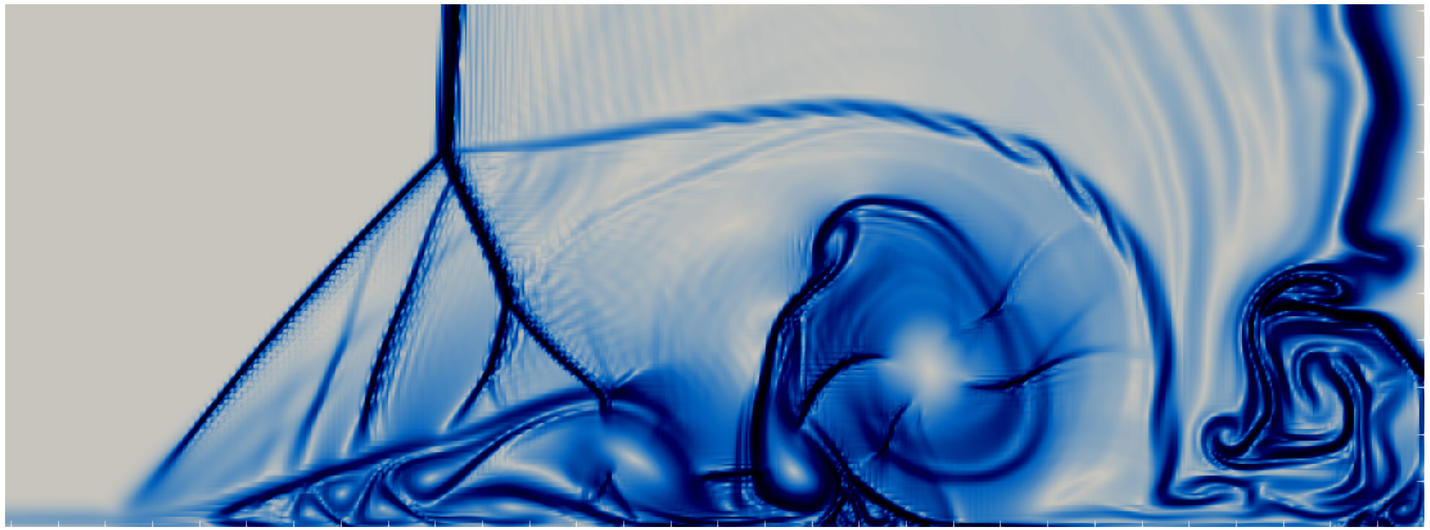
Sjogreen, Yee (2003). *Grid convergence of high order methods for multiscale complex unsteady viscous compressible flows*.

# Our contribution: positivity preserving entropy stable nodal DG

## Application: compressible Navier-Stokes equations

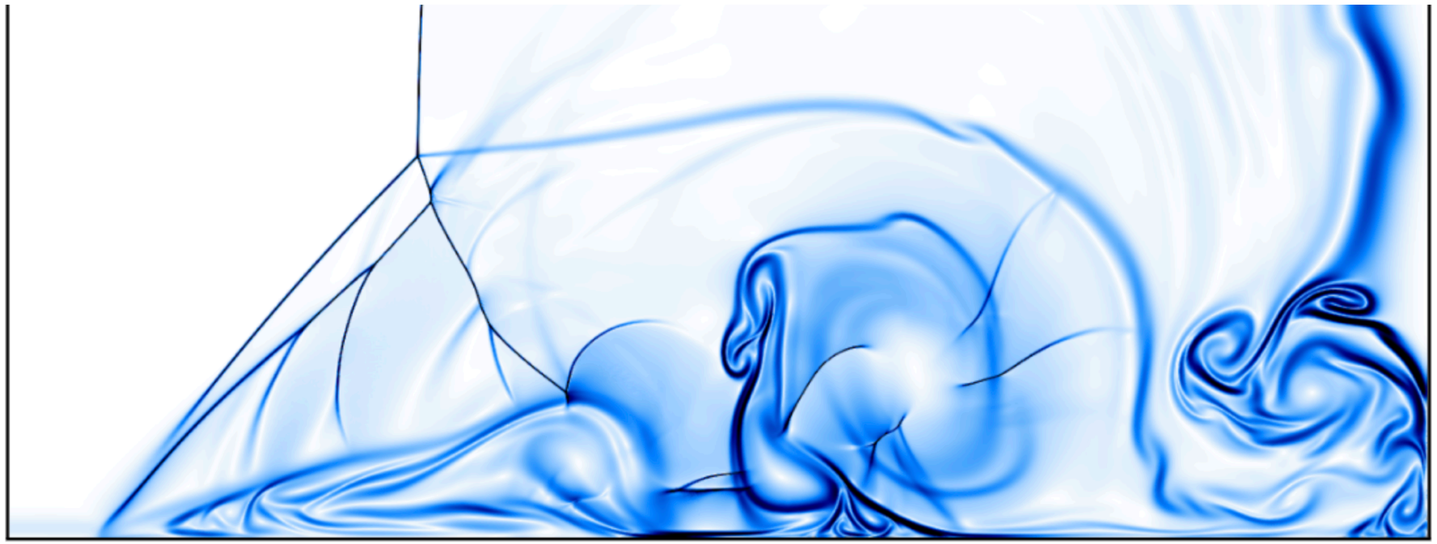
### Lin, Chan, Tomas 2023

4th order entropy stable DG
200 x 100 uniform mesh
Modal indicator shock capturing



### Guermond et al. 2022

Fine-grid reference solution
2nd order invariant-domain P1-FEM
Graded mesh, 512M dofs



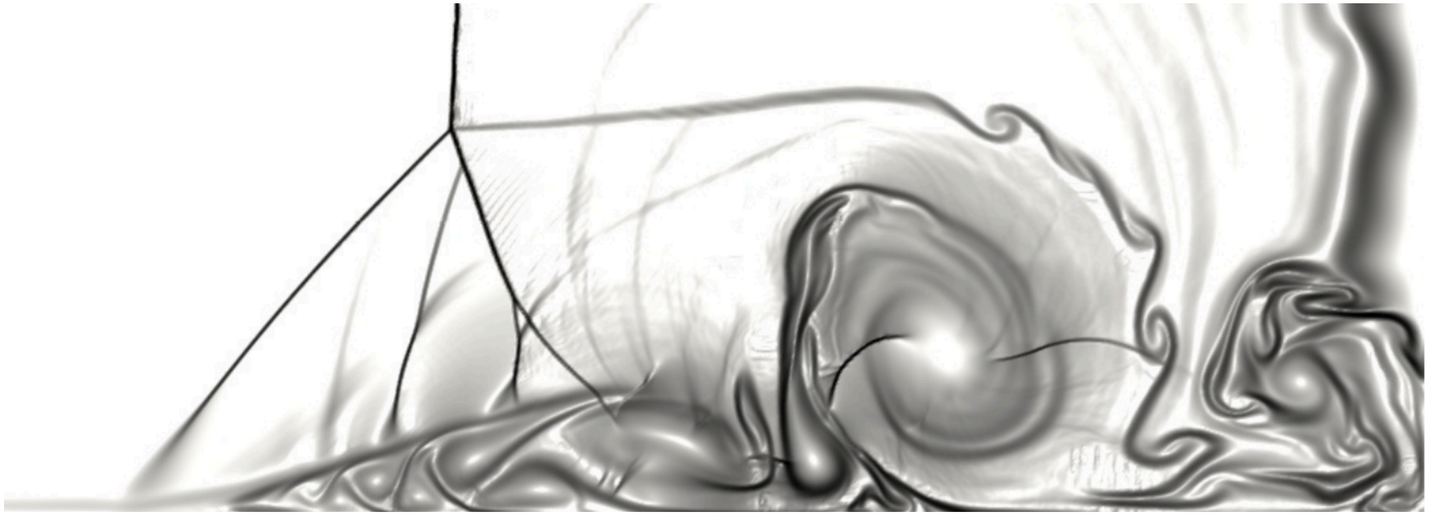
Hennemann et al (2021). *A provably entropy stable subcell shock capturing approach for high order split form DG for the compressible Euler equations*.

# Our contribution: positivity preserving entropy stable nodal DG

## Application: compressible Navier-Stokes equations

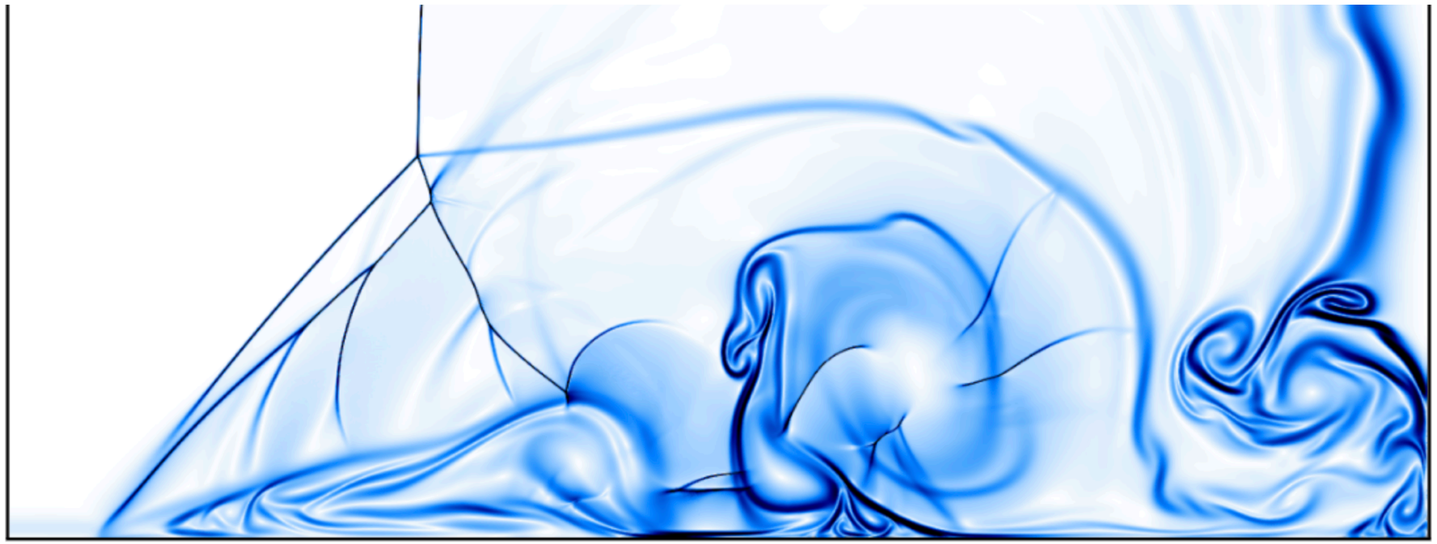
### Dzanic, Witherden 2022

5th order DG
800 x 400 uniform mesh (~8 million dofs)
Adaptive filtering (positivity, min entropy)



### Guermond et al. 2022

Fine-grid reference solution
2nd order invariant-domain P1-FEM
Graded mesh, 512M dofs



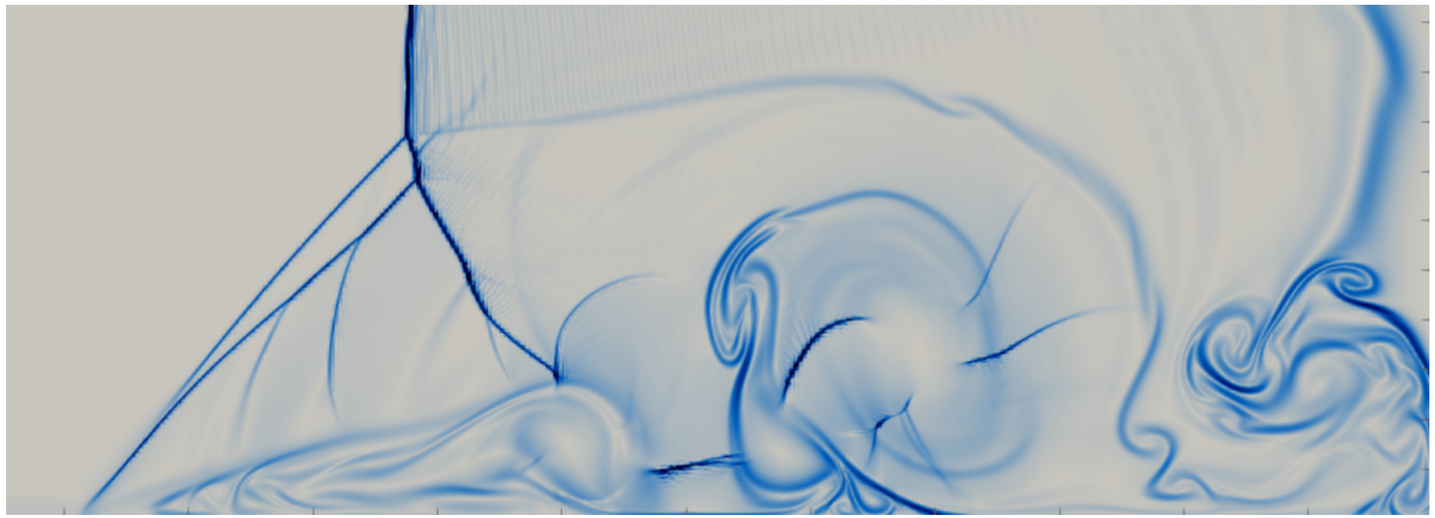
Dzanic, Witherden (2022). *Positivity-preserving entropy-based adaptive filtering for discontinuous spectral element methods*.

# Our contribution: positivity preserving entropy stable nodal DG

## Application: compressible Navier-Stokes equations

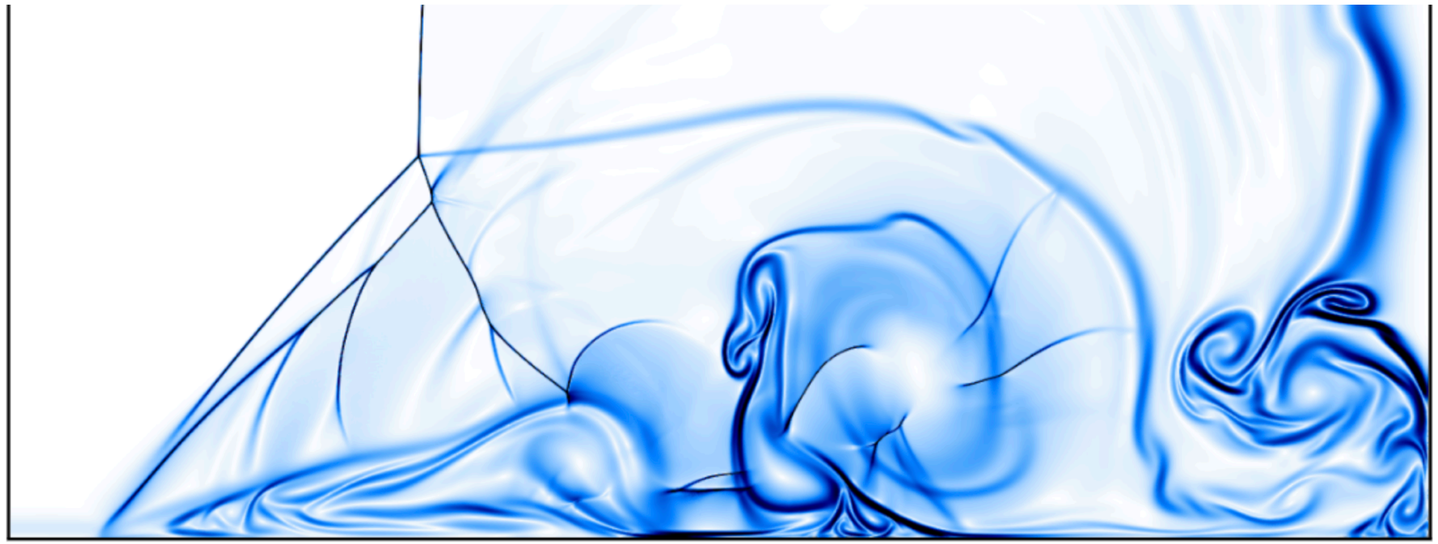
### Lin, Chan, Tomas 2023

4th order entropy stable DG
600 x 300 uniform mesh (~3 million dofs)
ESDG + subcell positivity limiting



### Guermond et al. 2022

Fine-grid reference solution
2nd order invariant-domain P1-FEM
Graded mesh, 512M dofs



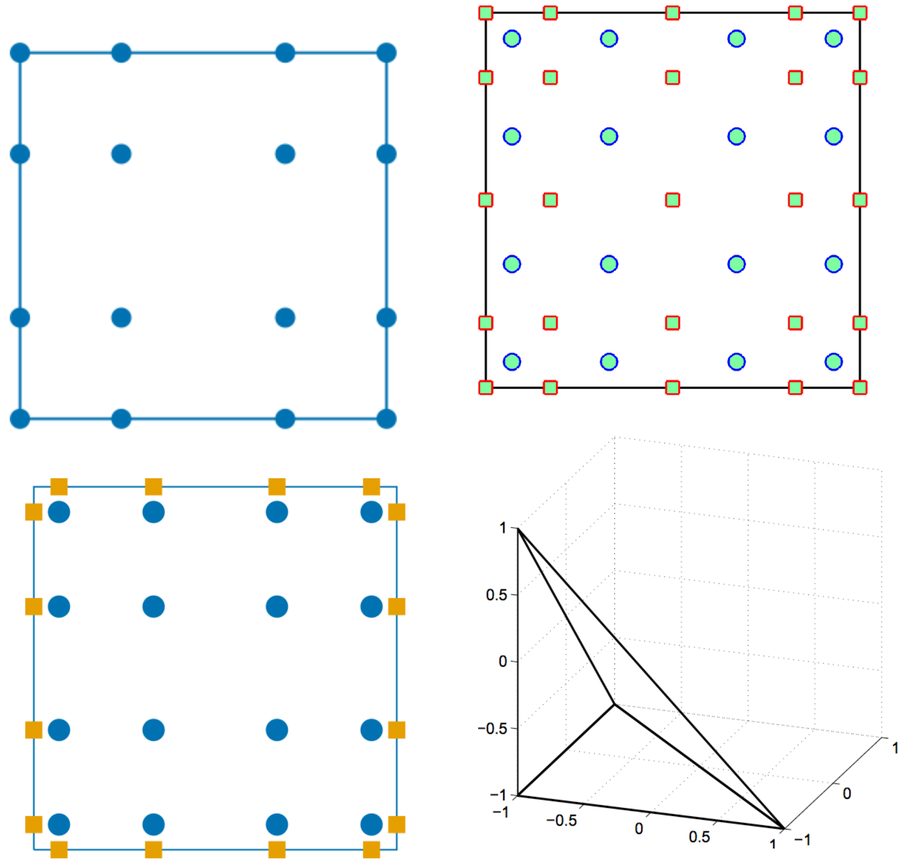
Lin, Chan, Tomas (2023). *A positivity preserving strategy for entropy stable discontinuous Galerkin discretizations of the compressible Euler and Navier-Stokes equations.*

# Beyond nodal DG: more general entropy stable formulations

# More general entropy stable “modal” formulations

## Beyond nodal DG schemes

- “Nodal” formulations: collocation schemes with simple algebraic structure.
- “Modal” formulations: unified framework
  - Recovers nodal collocation schemes (Lobatto, Gauss, staggered)
  - Generalizes to arbitrary basis, quadrature elements (simplices, pyramids, etc)




“Modal” DG formulations provide a unified treatment of many different schemes (DGSEM, staggered grid methods, spectral difference) on arbitrary elements.


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

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

# Modal entropy stable formulations are flexible

## Examples: cut cell meshes and reduced order models (ROMs)

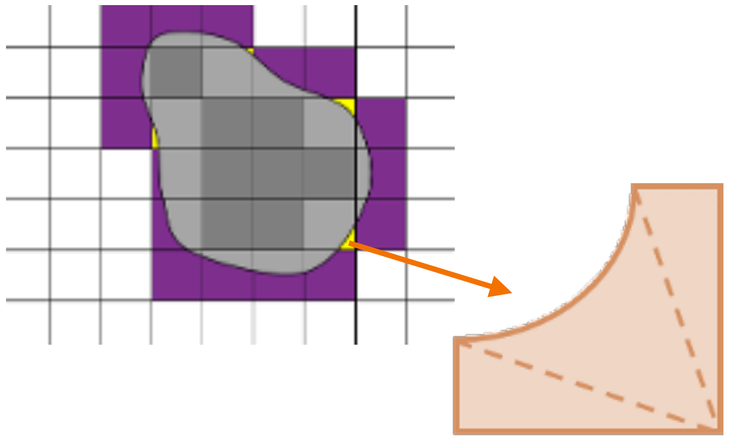
Why focus on cut cells/ROMs? Non-standard basis functions and quadrature.

# Modal entropy stable formulations are flexible

## Examples: cut cell meshes and reduced order models (ROMs)

Why focus on cut cells/ROMs? Non-standard basis functions and quadrature.

Cut-cell meshes often use *physical-frame* basis functions, quadrature.

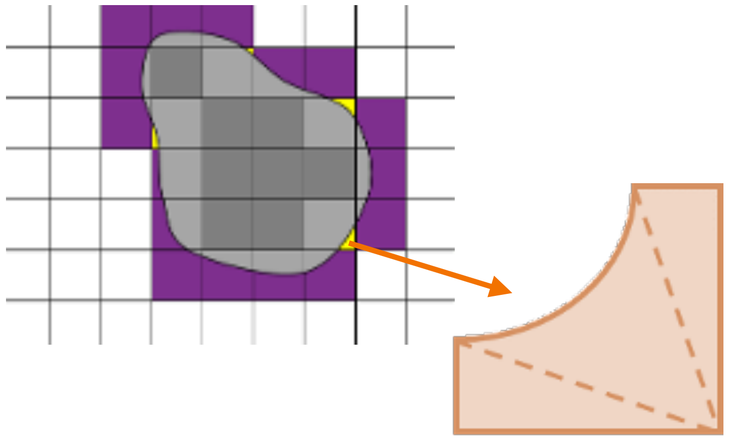


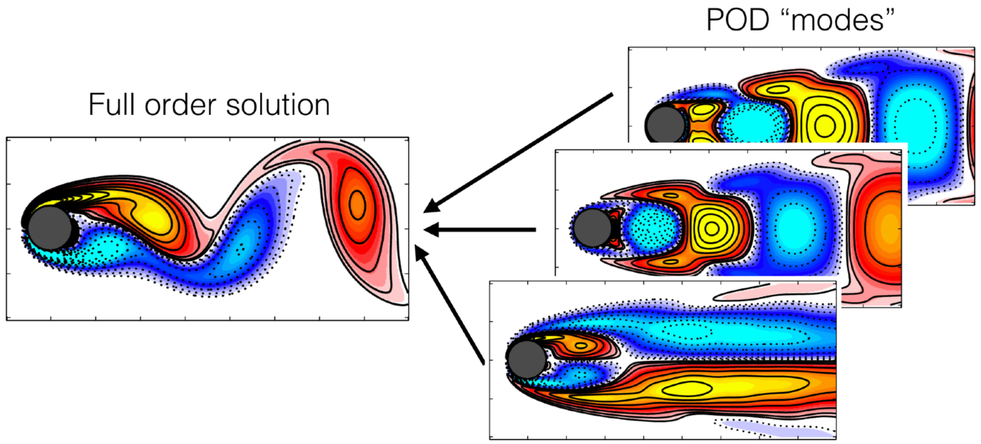
# Modal entropy stable formulations are flexible

## Examples: cut cell meshes and reduced order models (ROMs)

Why focus on cut cells/ROMs? Non-standard basis functions and quadrature.

Cut-cell meshes often use *physical-frame* basis functions, quadrature.






Reduced order modeling: data-driven basis functions, “quadrature” via hyper-reduction

Figure adapted from Brunton, Proctor, Kutz (2016). *Discovering governing equations from data by sparse identification of nonlinear dynamical systems*

# A short summary of modal entropy stable DG

## Entropy projection, “hybridized” summation-by-parts (SBP)

Step 1: use the “entropy projection”
(generalization of solution interpolation)

$$\tilde{\boldsymbol{u}} = \boldsymbol{u}\left(\Pi_N \boldsymbol{v}(\boldsymbol{u}_h)\right)$$

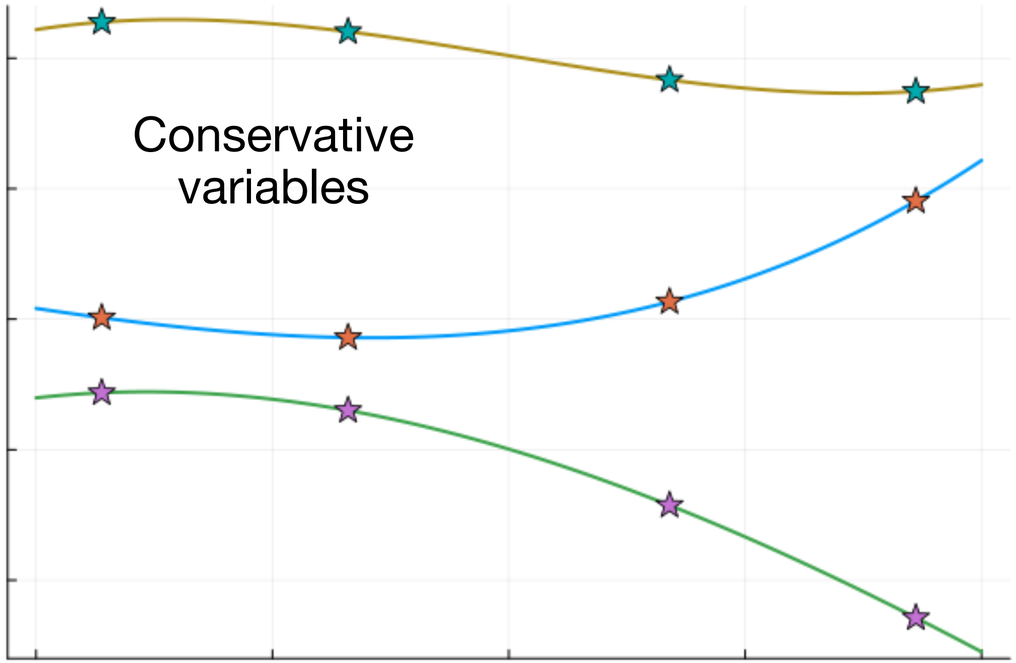
Chan (2018). *Skew-symmetric entropy stable modal discontinuous Galerkin formulations.*

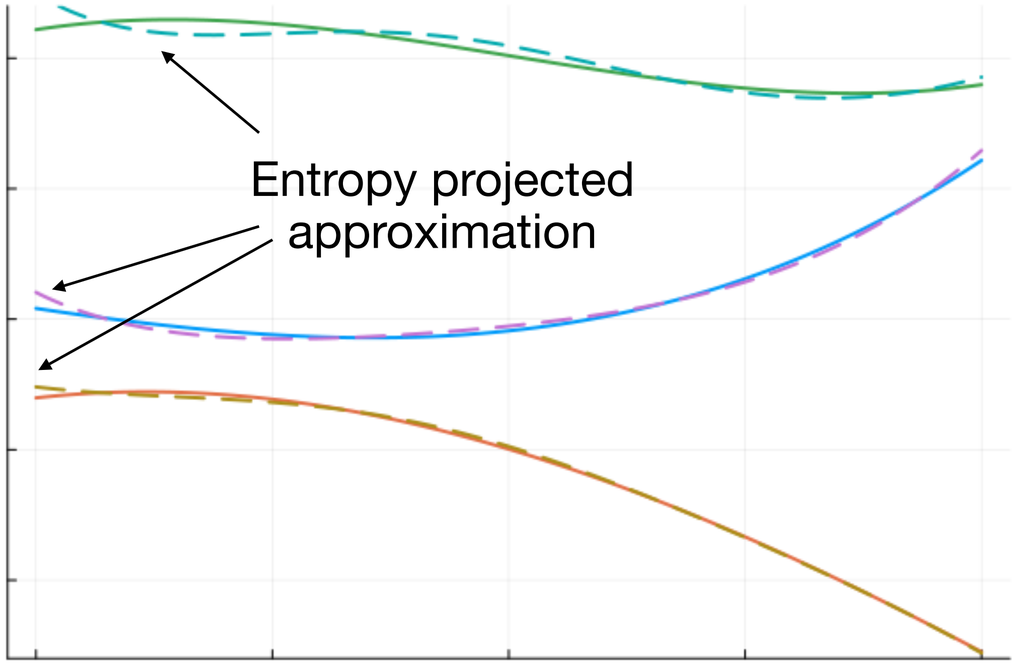
# A short summary of modal entropy stable DG

## Entropy projection, “hybridized” summation-by-parts (SBP)

Step 1: use the “entropy projection” (generalization of solution interpolation)

$$\tilde{\boldsymbol{u}} = \boldsymbol{u}\left(\Pi_N \boldsymbol{v}(\boldsymbol{u}_h)\right)$$




Chan (2018). *Skew-symmetric entropy stable modal discontinuous Galerkin formulations.*

# A short summary of modal entropy stable DG

## Entropy projection, “hybridized” summation-by-parts (SBP)

Step 1: use the “entropy projection” (generalization of solution interpolation)

$$\tilde{\boldsymbol{u}} = \boldsymbol{u}\left(\Pi_N \boldsymbol{v}(\boldsymbol{u}_h)\right)$$

Step 2: use “hybridized” SBP operators, which result in weaker conditions for entropy stability compared to traditional SBP operators.

Traditional SBP operators

$$\mathbf{Q}\mathbf{1} = \mathbf{0}$$

$$\mathbf{Q} + \mathbf{Q}^T = \mathbf{E}^T \mathbf{B} \mathbf{E}$$

Hybridized SBP operators

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

$$\mathbf{Q}_h \mathbf{1} = \mathbf{0}, \quad \mathbf{Q}_h + \mathbf{Q}_h^T = \begin{bmatrix} \mathbf{0} & \\ & \mathbf{B} \end{bmatrix}$$

Chan (2018). *Skew-symmetric entropy stable modal discontinuous Galerkin formulations.*

# A short summary of modal entropy stable DG

## Entropy projection, “hybridized” summation-by-parts (SBP)

Step 1: use the “entropy projection” (generalization of solution interpolation)

$$\tilde{\boldsymbol{u}} = \boldsymbol{u}\left(\Pi_N \boldsymbol{v}(\boldsymbol{u}_h)\right)$$

Step 2: use “hybridized” SBP operators, which result in weaker conditions for entropy stability compared to traditional SBP operators.

Traditional SBP operators

$$\mathbf{Q}\mathbf{1} = \mathbf{0}$$

$$\mathbf{Q} + \mathbf{Q}^T = \mathbf{E}^T\mathbf{B}\mathbf{E}$$

Hybridized SBP operators

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

$$\mathbf{Q}_h\mathbf{1} = \mathbf{0}, \quad \mathbf{Q}_h + \mathbf{Q}_h^T = \begin{bmatrix} \mathbf{0} & \\ & \mathbf{B} \end{bmatrix}$$

Conditions for entropy stability

Chan (2018). *Skew-symmetric entropy stable modal discontinuous Galerkin formulations.*

# A short summary of modal entropy stable DG

## Entropy projection, “hybridized” summation-by-parts (SBP)

Step 1: use the “entropy projection” (generalization of solution interpolation)

$$\tilde{\boldsymbol{u}} = \boldsymbol{u}\left(\Pi_N \boldsymbol{v}(\boldsymbol{u}_h)\right)$$

Step 2: use “hybridized” SBP operators, which result in weaker conditions for entropy stability compared to traditional SBP operators.

Traditional SBP operators

$$\mathbf{Q}\mathbf{1} = \mathbf{0}$$

$$\mathbf{Q} + \mathbf{Q}^T = \mathbf{E}^T\mathbf{B}\mathbf{E}$$

Related to quadrature accuracy

Hybridized SBP operators

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

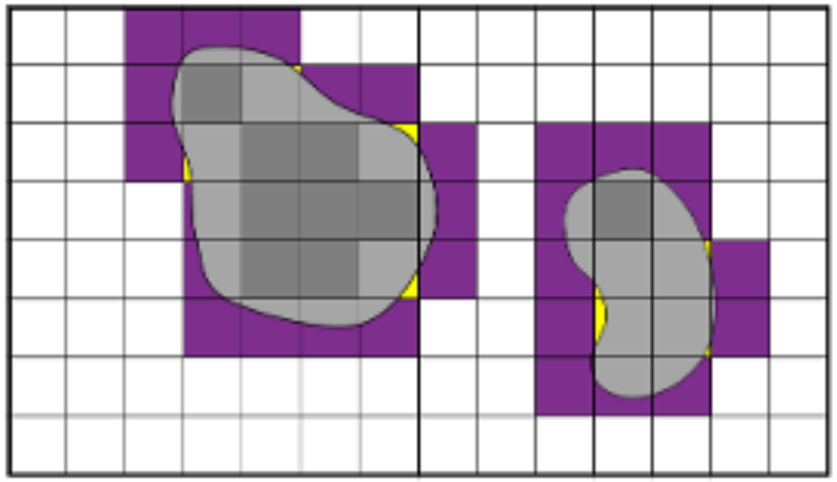
$$\mathbf{Q}_h\mathbf{1} = \mathbf{0}, \quad \mathbf{Q}_h + \mathbf{Q}_h^T = \begin{bmatrix} \mathbf{0} & \\ & \mathbf{B} \end{bmatrix}$$

Conditions for entropy stability

Chan (2018). *Skew-symmetric entropy stable modal discontinuous Galerkin formulations.*

# Application: entropy stable cut-cell DG

- Challenge: constructing traditional summation-by-parts (SBP) operators on arbitrary cut cells.
- Using hybridized SBP operators reduces this to constructing polynomially exact quadratures for physical-frame basis functions on cut cells.



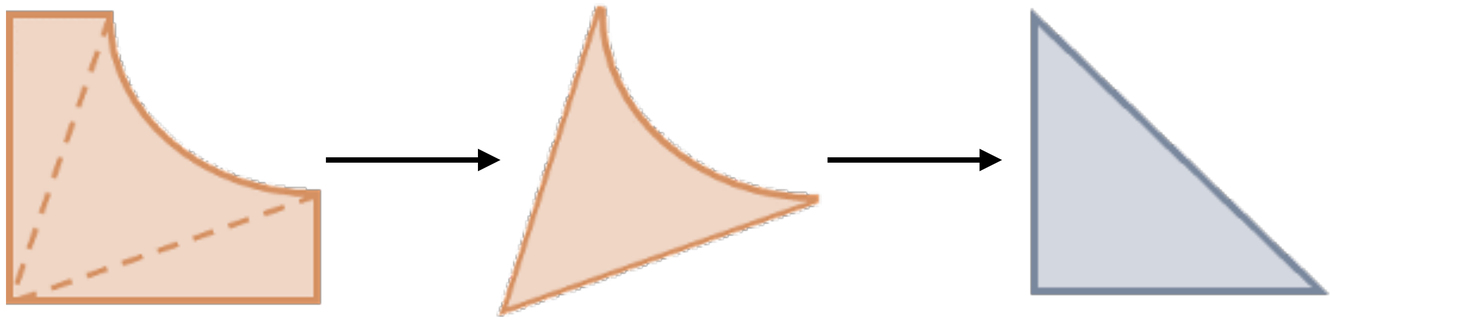
To guarantee entropy stability ($\mathbf{Q}_h\mathbf{1} = \mathbf{0}$), cut cell volume/surface quadratures only need to satisfy the fundamental theorem of calculus for basis $\phi_j$.

$$\int_{D^k} \frac{\partial \phi_j}{\partial x_i} = \int_{\partial D^k} \phi_j n_i$$

# Constructing polynomially exact quadrature on cut cells

## Assume cut cells yield an isoparametric curvilinear subtriangulation

We can prove entropy stability under exact integration of physical-frame *polynomials.*



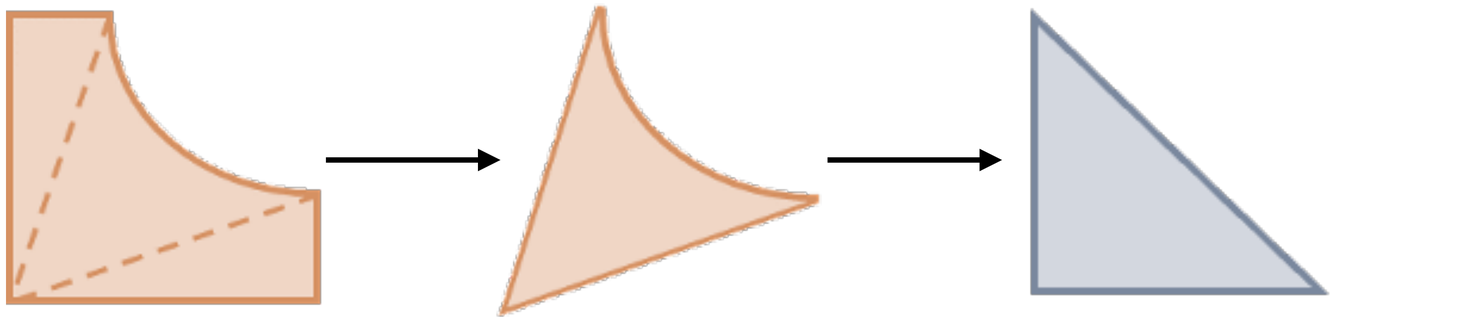
$$\int_{D^k} p(\boldsymbol{x})\,\mathrm{d}x \qquad \int_{\widehat{D}} p\left(\boldsymbol{x}\left(\widehat{\boldsymbol{x}}\right)\right)\left|\frac{\partial \boldsymbol{x}}{\partial \widehat{\boldsymbol{x}}}\right|\mathrm{d}\widehat{x}$$

$$p(\boldsymbol{x}) \in P^N\left(D^k\right)$$

# Constructing polynomially exact quadrature on cut cells

## Assume cut cells yield an isoparametric curvilinear subtriangulation

We can prove entropy stability under exact integration of physical-frame *polynomials.*



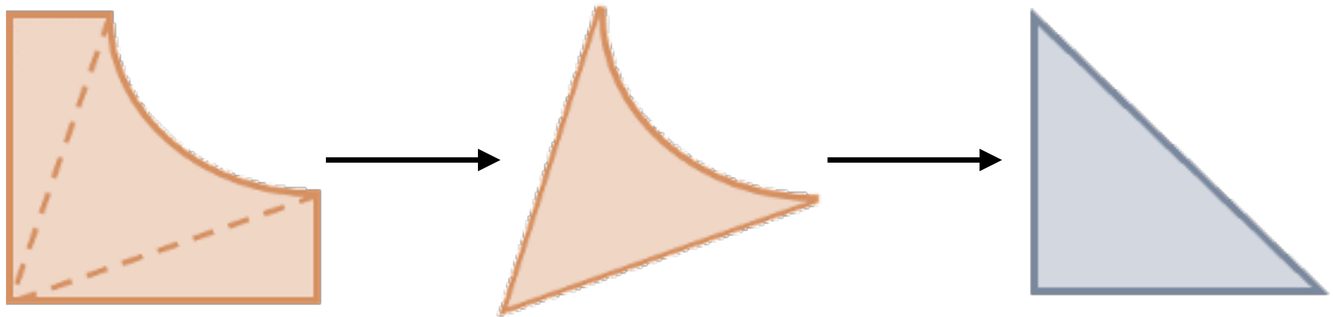
$$\int_{D^k} p(\boldsymbol{x})\,\mathrm{d}x \qquad \text{Degree } N \qquad \int_{\widehat{D}} p\left(\boldsymbol{x}\left(\widehat{\boldsymbol{x}}\right)\right)\left|\frac{\partial \boldsymbol{x}}{\partial \widehat{\boldsymbol{x}}}\right|\mathrm{d}\widehat{x}$$

$$p(\boldsymbol{x}) \in P^N\left(D^k\right) \qquad \text{geometric mapping}$$

# Constructing polynomially exact quadrature on cut cells

## Assume cut cells yield an isoparametric curvilinear subtriangulation

We can prove entropy stability under exact integration of physical-frame *polynomials.*



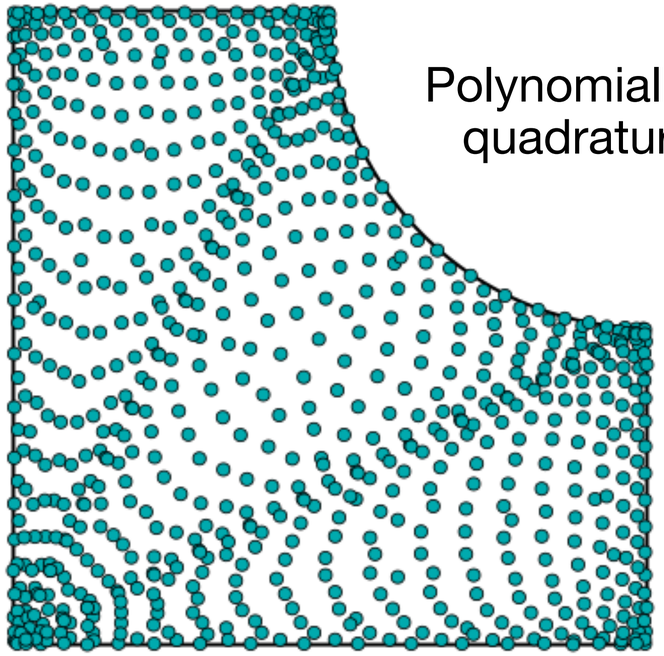
$$\int_{D^k} p(\boldsymbol{x})\,\mathrm{d}x \qquad \text{Degree } N \qquad \int_{\widehat{D}} p\left(\boldsymbol{x}\left(\widehat{\boldsymbol{x}}\right)\right)\left|\frac{\partial \boldsymbol{x}}{\partial \widehat{\boldsymbol{x}}}\right|\mathrm{d}\widehat{x}$$

$$p(\boldsymbol{x}) \in P^N\left(D^k\right) \qquad \text{geometric mapping} \qquad p(\boldsymbol{x}(\widehat{\boldsymbol{x}})) \in P^{N^2}\left(\widehat{D}\right)$$

Can exactly integrate, but needs *very* high degree reference element quadrature.

# Reduced quadrature on cut cells

"Caratheodory pruning": reduces a quadrature rule to $n$ points (where $n$ is the number of moment conditions desired) while preserving positivity and exactness (sufficient conditions for entropy conservation or stability).



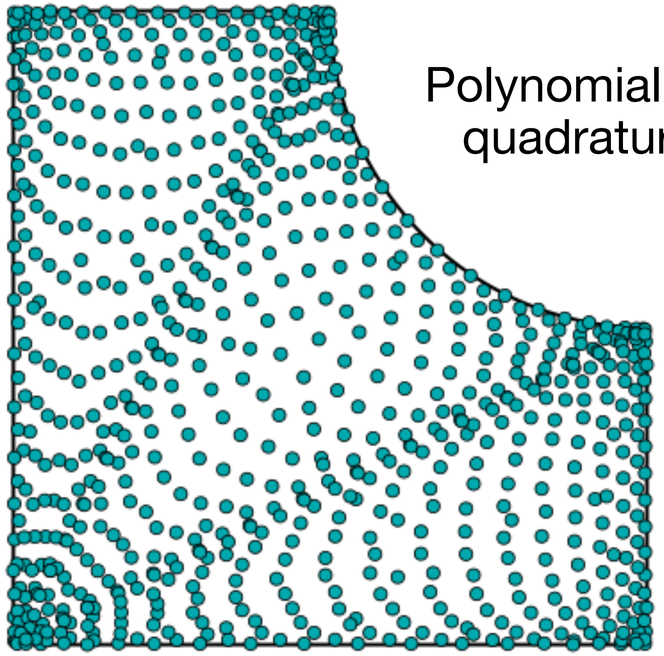
Polynomially exact quadrature rule

Wilson (1969). *A general algorithm for nonnegative quadrature formulas*.

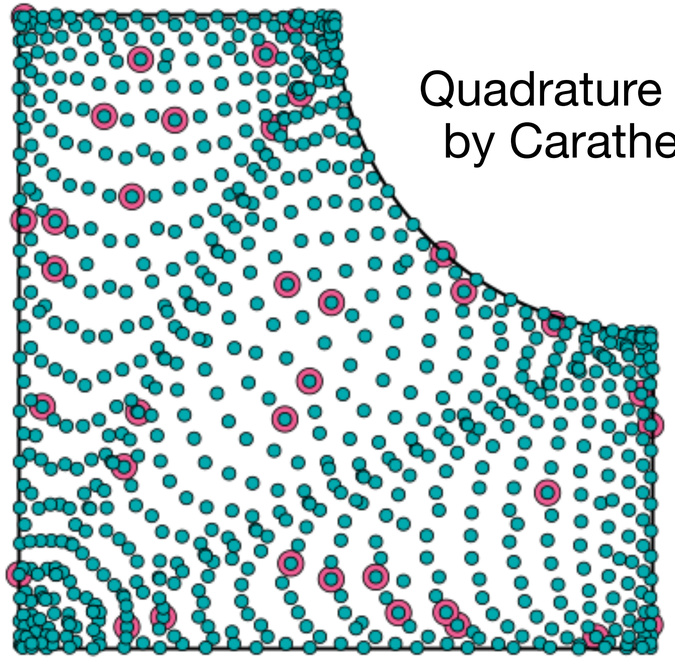
Van Den Bos, Sanderse, Bierbooms, Van Bussel (2020). *Generating nested quadrature rules with positive weights based on arbitrary sample sets*.

# Reduced quadrature on cut cells

"Caratheodory pruning": reduces a quadrature rule to $n$ points (where $n$ is the number of moment conditions desired) while preserving positivity and exactness (sufficient conditions for entropy conservation or stability).



Polynomially exact quadrature rule



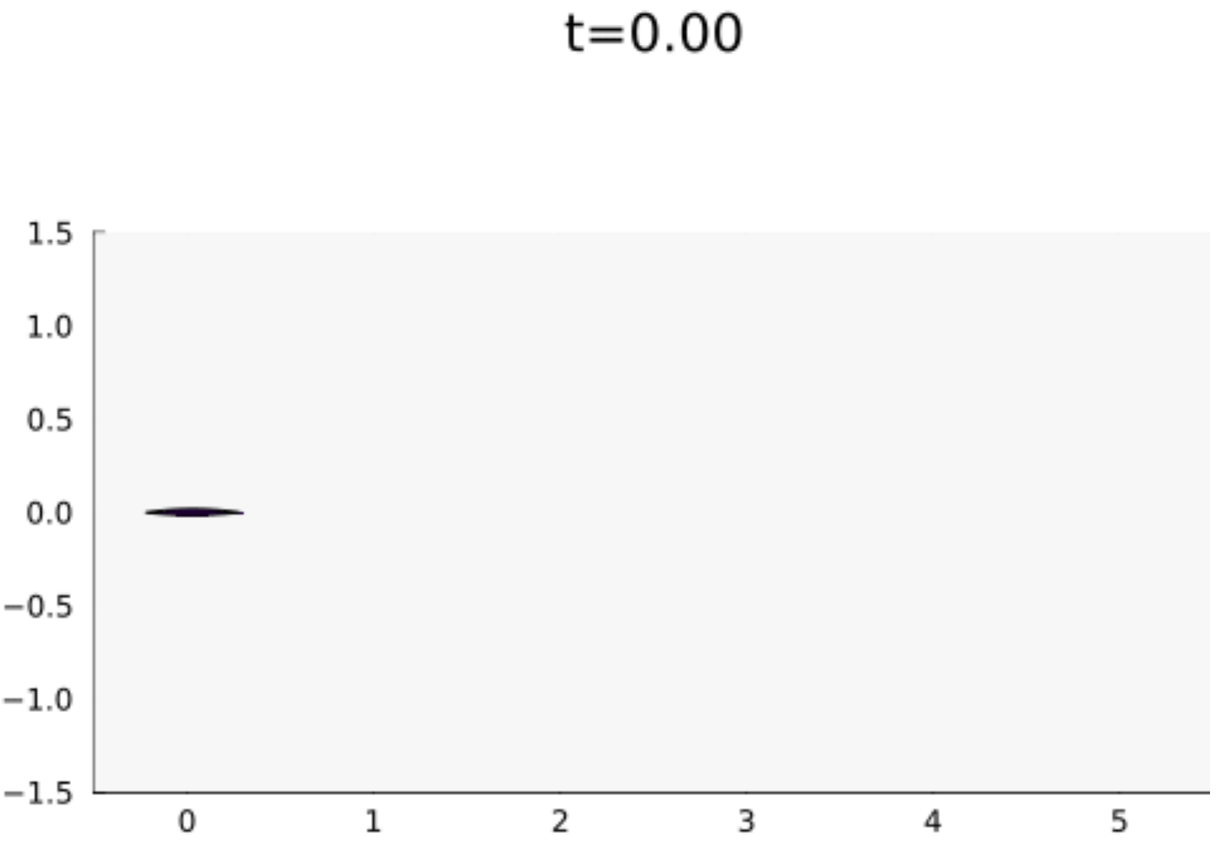
Quadrature nodes selected by Caratheodory pruning

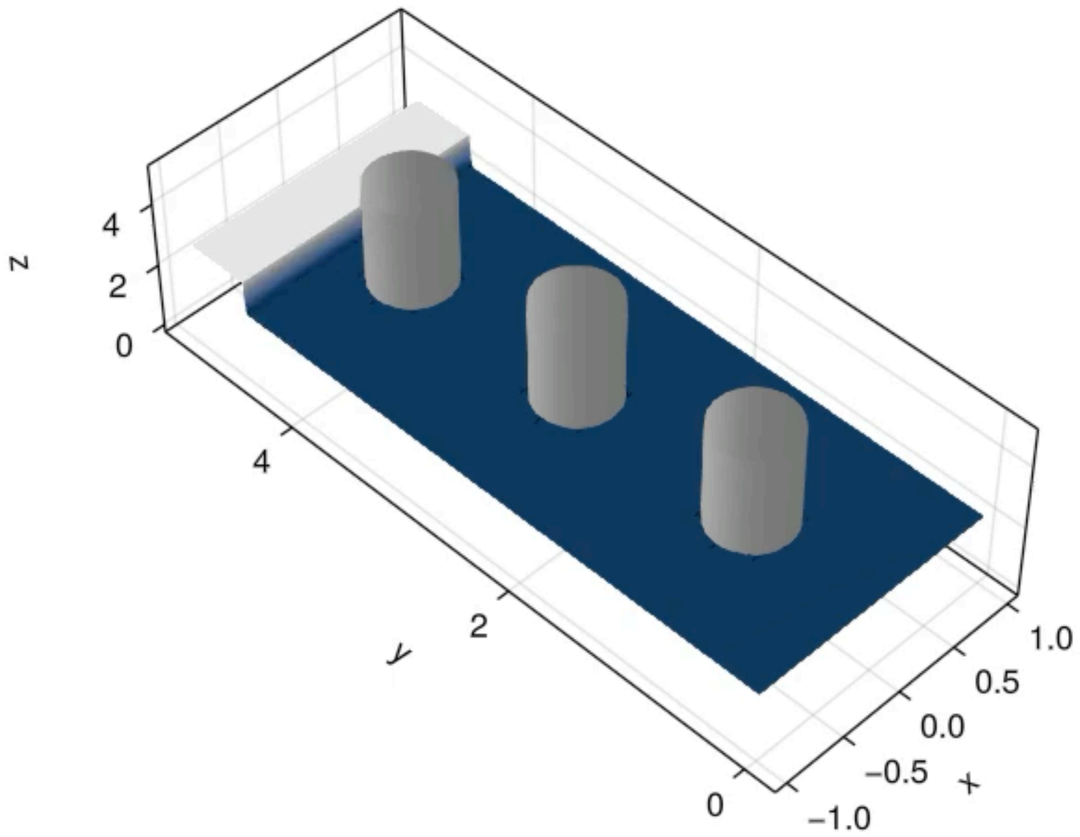

Wilson (1969). *A general algorithm for nonnegative quadrature formulas*.

Van Den Bos, Sanderse, Bierbooms, Van Bussel (2020). *Generating nested quadrature rules with positive weights based on arbitrary sample sets*.

# Examples of high order entropy stable cut-cell DG

## State redistribution (Berger and Giuliani) for the small-cell problem.

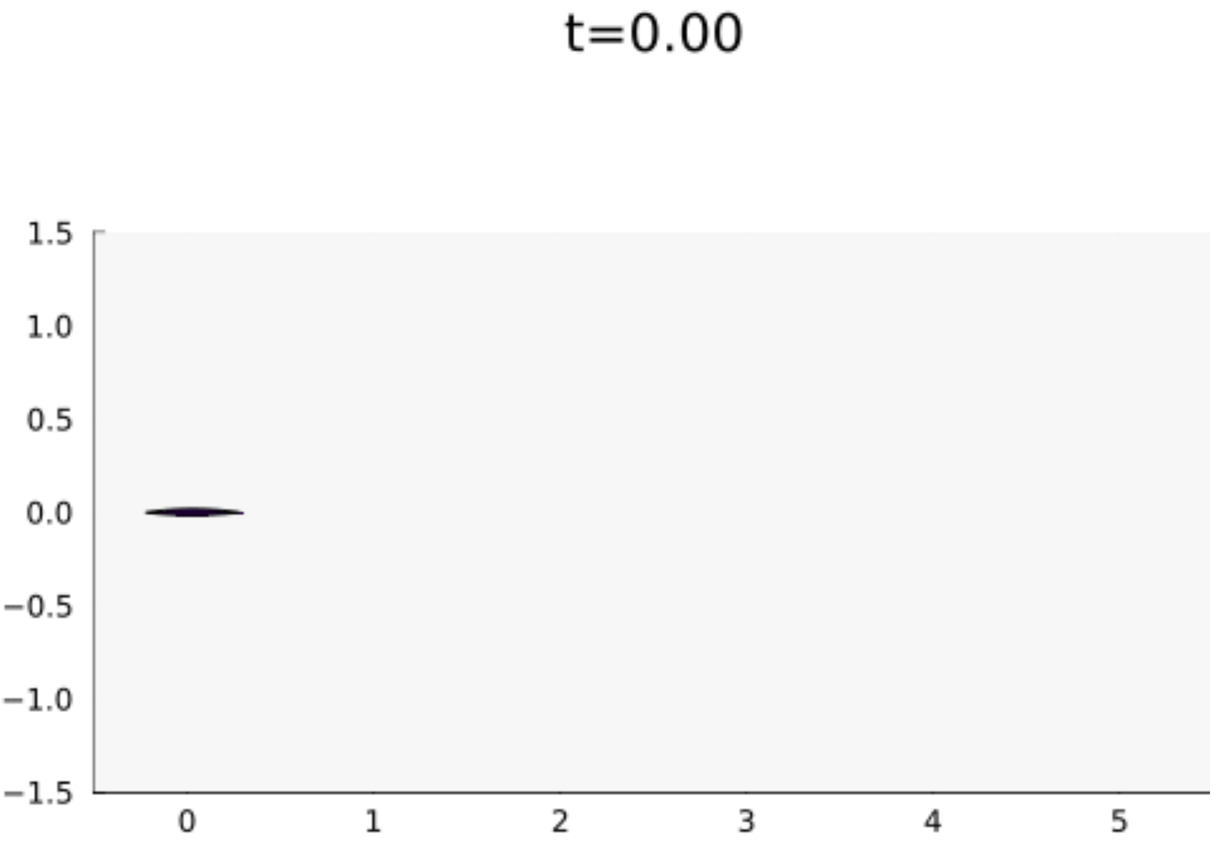


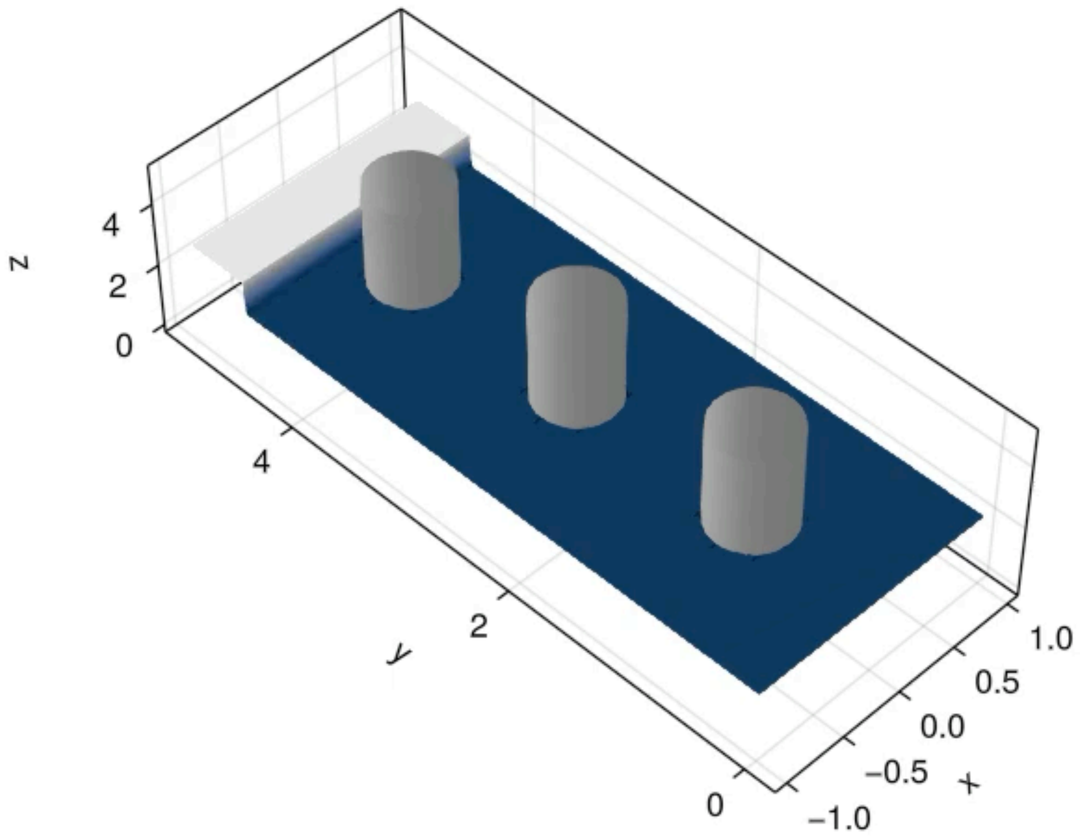




Berger and Giuliani (2021). A state redistribution algorithm for finite volume schemes on cut cell meshes.
Giuliani (2022). *A two-dimensional stabilized discontinuous Galerkin method on curvilinear embedded boundary grids.*

# Examples of high order entropy stable cut-cell DG

## State redistribution (Berger and Giuliani) for the small-cell problem.

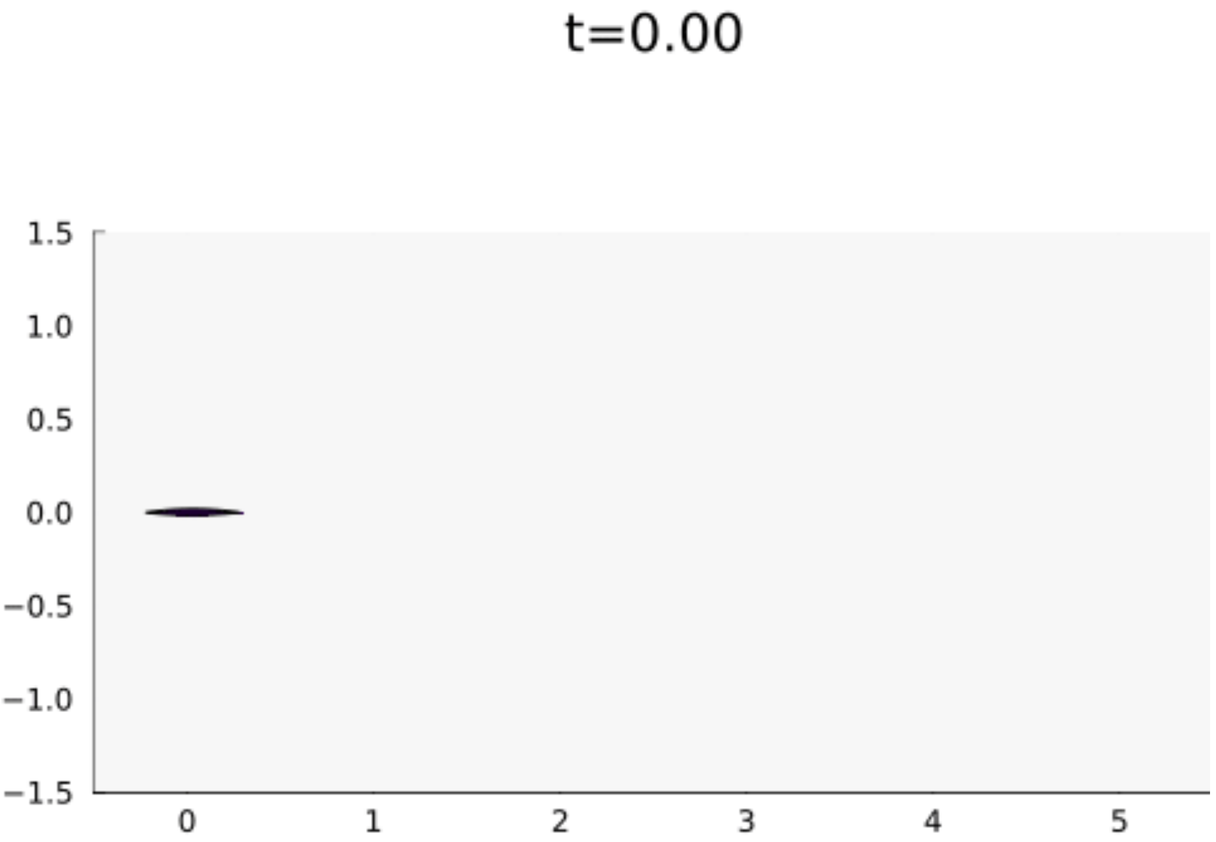


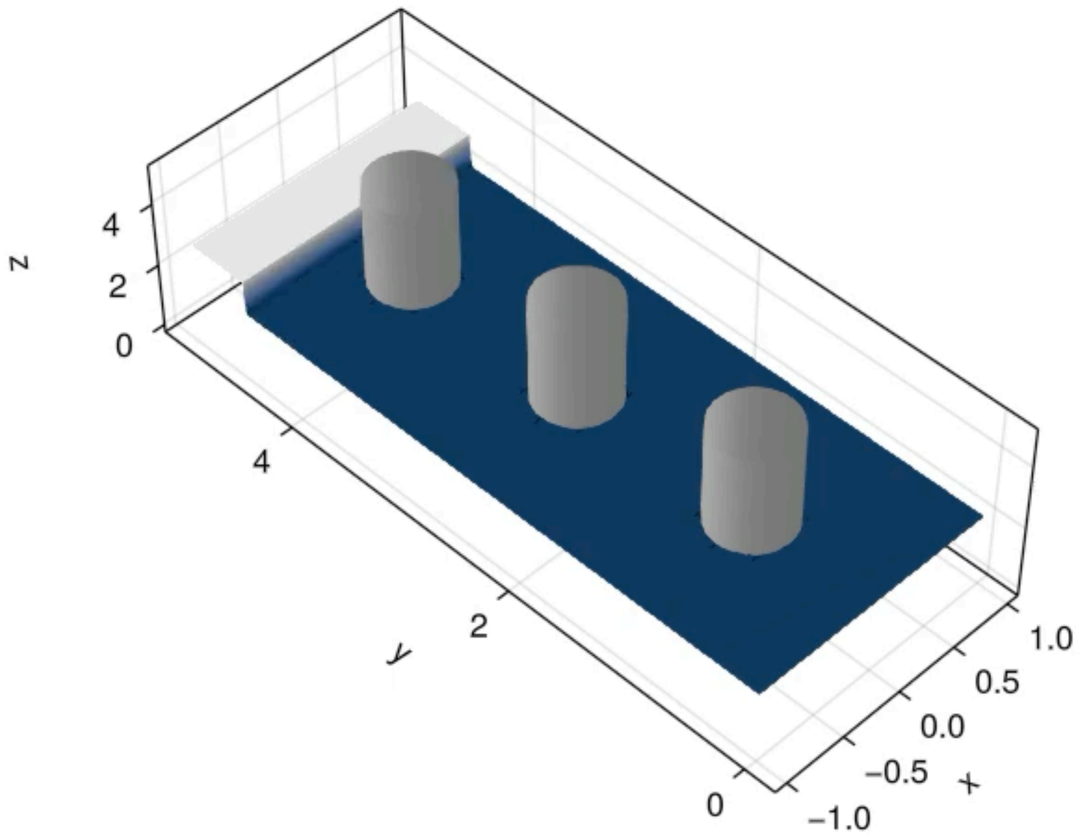




Berger and Giuliani (2021). A state redistribution algorithm for finite volume schemes on cut cell meshes.
Giuliani (2022). *A two-dimensional stabilized discontinuous Galerkin method on curvilinear embedded boundary grids.*

# Examples of high order entropy stable cut-cell DG

## State redistribution (Berger and Giuliani) for the small-cell problem.




Berger and Giuliani (2021). A state redistribution algorithm for finite volume schemes on cut cell meshes.
Giuliani (2022). *A two-dimensional stabilized discontinuous Galerkin method on curvilinear embedded boundary grids.*

# Application: entropy stable reduced order modeling

- Goal: reduce cost of many-query scenarios such as UQ.
- Main steps:
  - An offline training phase to collect snapshots of solutions from a full order model (FOM).
  - Cheap online phase with reduced order model (ROM).

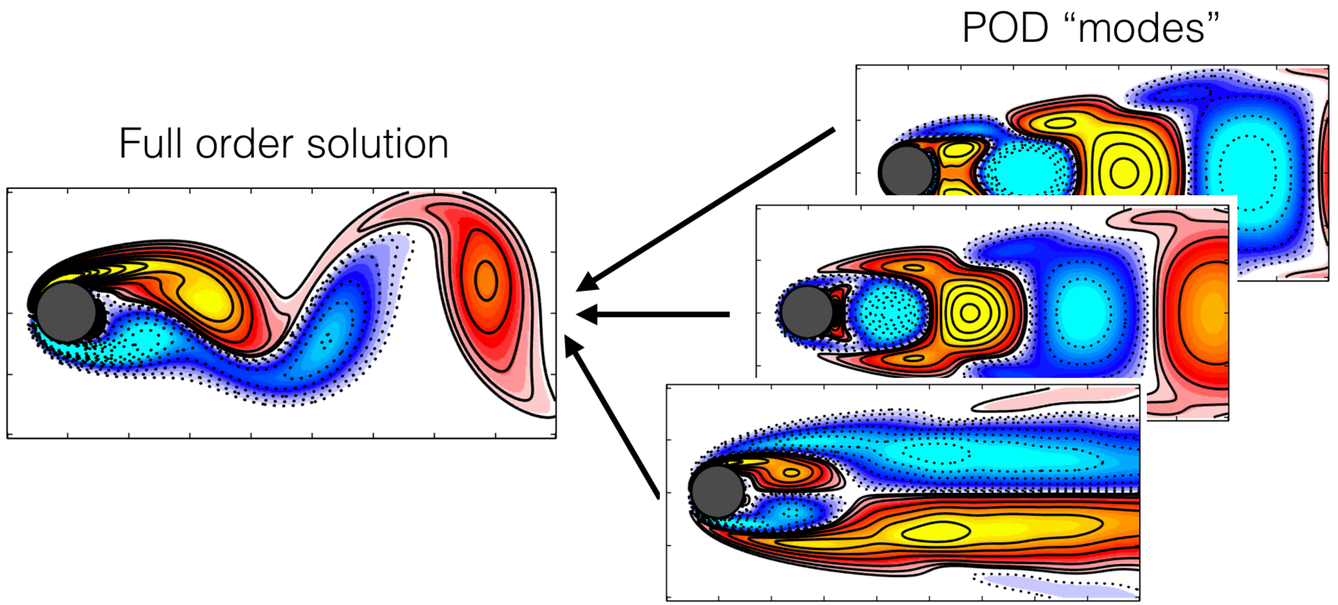
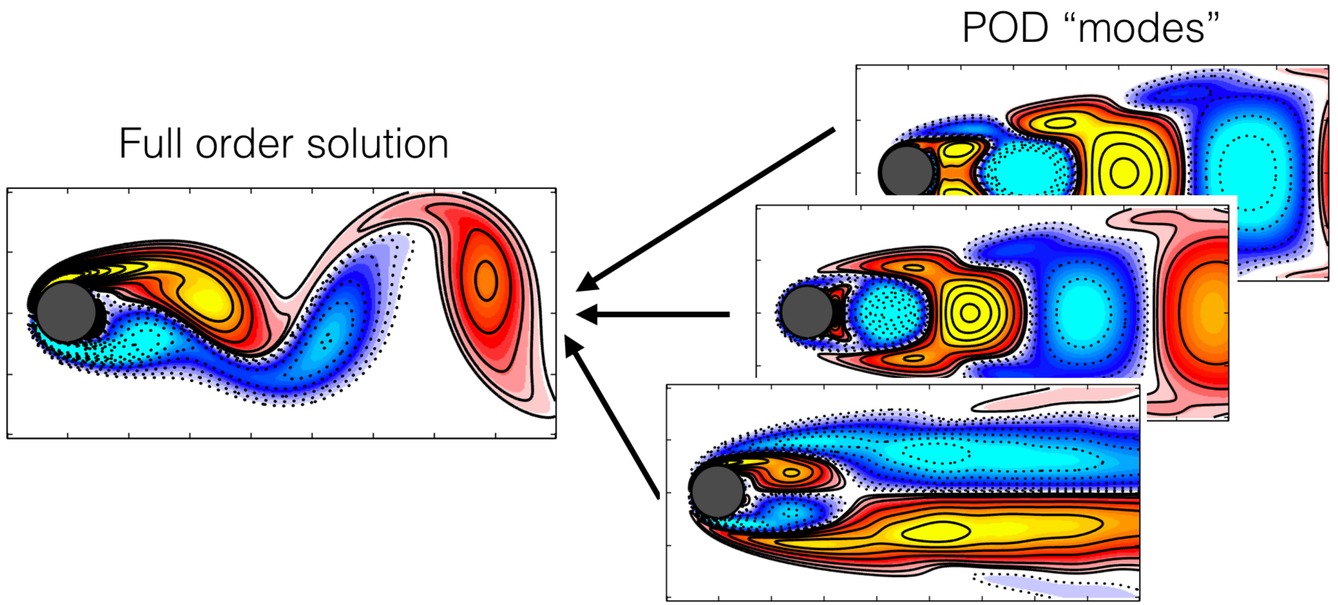



Figure adapted from Brunton, Proctor, Kutz (2016). “Discovering governing equations from data by sparse identification of nonlinear dynamical systems”

# Application: entropy stable reduced order modeling

- Goal: reduce cost of many-query scenarios such as UQ.
- Main steps:
  - An offline training phase to collect snapshots of solutions from a full order model (FOM).
  - Cheap online phase with reduced order model (ROM).

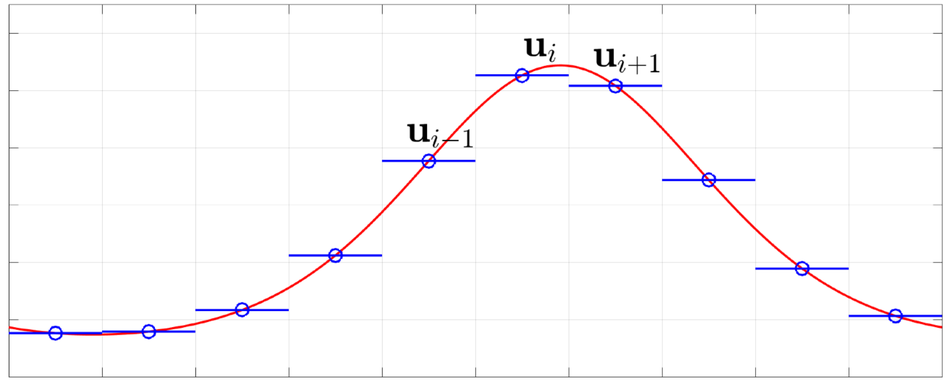



Challenge: ROMs do not inherit stability of FOM for nonlinear hyperbolic PDEs!

Figure adapted from Brunton, Proctor, Kutz (2016). “Discovering governing equations from data by sparse identification of nonlinear dynamical systems”

# Full order method (FOM)

## On a periodic domain




- Entropy conservative finite volumes

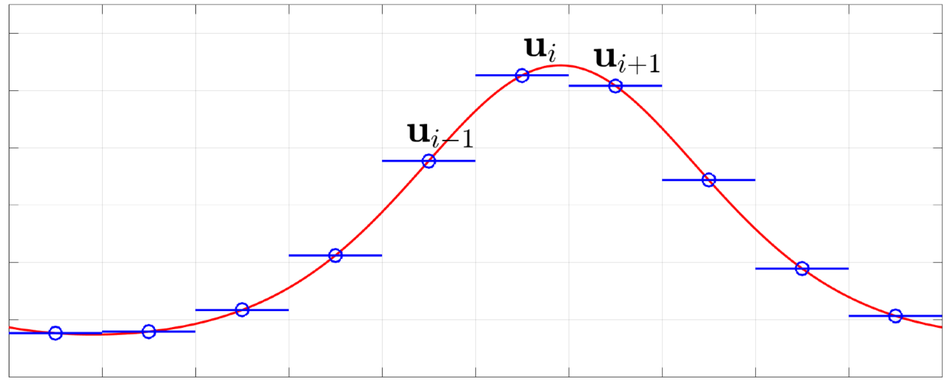
$$\Delta x \frac{\mathrm{d}\mathbf{u}_i}{\mathrm{d}t} + \boldsymbol{f}_{EC}(\mathbf{u}_{i+1}, \mathbf{u}_i) - \boldsymbol{f}_{EC}(\mathbf{u}_i, \mathbf{u}_{i-1}) = \text{viscous terms}$$

- Matrix form of the finite volume method:

$$\mathbf{M}\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} + 2\left(\mathbf{Q} \circ \mathbf{F}\right)\mathbf{1} + \epsilon\mathbf{K}\mathbf{u} = 0$$

# Full order method (FOM)

## On a periodic domain




- Entropy conservative finite volumes

$$\Delta x \frac{\mathrm{d}\mathbf{u}_i}{\mathrm{d}t} + \boldsymbol{f}_{EC}(\mathbf{u}_{i+1}, \mathbf{u}_i) - \boldsymbol{f}_{EC}(\mathbf{u}_i, \mathbf{u}_{i-1}) = \text{viscous terms}$$

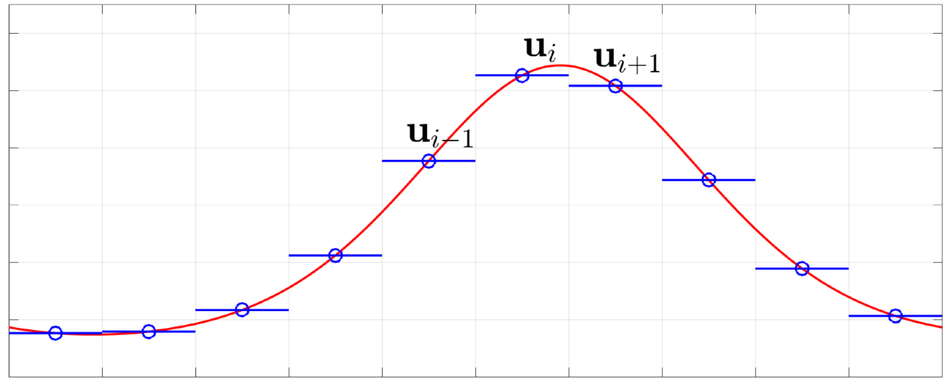
- Matrix form of the finite volume method:

$$\mathbf{M}\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} + 2\left(\mathbf{Q} \circ \mathbf{F}\right)\mathbf{1} + \epsilon\mathbf{K}\mathbf{u} = 0$$

$$\mathbf{M} = \Delta x \mathbf{I}$$

# Full order method (FOM)

## On a periodic domain




- Entropy conservative finite volumes

$$\Delta x \frac{\mathrm{d}\mathbf{u}_i}{\mathrm{d}t} + \boldsymbol{f}_{EC}(\mathbf{u}_{i+1}, \mathbf{u}_i) - \boldsymbol{f}_{EC}(\mathbf{u}_i, \mathbf{u}_{i-1}) = \text{viscous terms}$$

- Matrix form of the finite volume method:

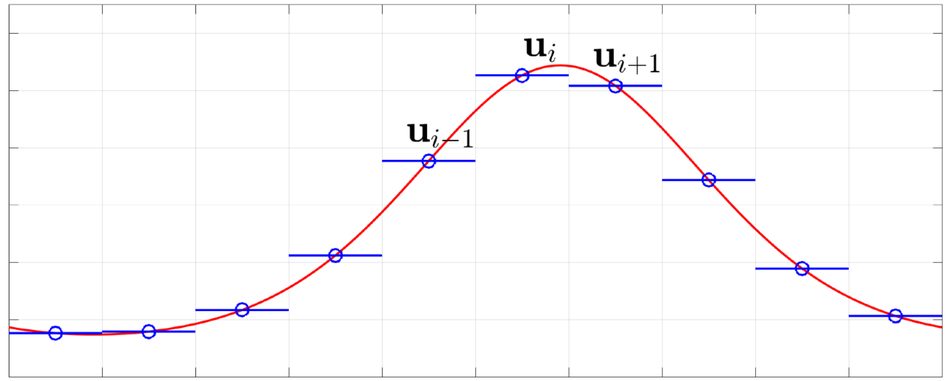
$$\mathbf{M}\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} + 2\left(\mathbf{Q} \circ \mathbf{F}\right)\mathbf{1} + \epsilon\mathbf{K}\mathbf{u} = 0$$

$$\mathbf{M} = \Delta x \mathbf{I}$$

$$\mathbf{Q} = \frac{1}{2}\begin{bmatrix} 0 & 1 & & -1 \\ -1 & 0 & 1 & \\ & \ddots & \ddots & 1 \\ 1 & & -1 & 0 \end{bmatrix}$$

# Full order method (FOM)

## On a periodic domain




- Entropy conservative finite volumes

$$\Delta x \frac{\mathrm{d}\mathbf{u}_i}{\mathrm{d}t} + \boldsymbol{f}_{EC}(\mathbf{u}_{i+1}, \mathbf{u}_i) - \boldsymbol{f}_{EC}(\mathbf{u}_i, \mathbf{u}_{i-1}) = \text{viscous terms}$$

- Matrix form of the finite volume method:

$$\mathbf{M}\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} + 2\left(\mathbf{Q} \circ \mathbf{F}\right)\mathbf{1} + \epsilon\mathbf{K}\mathbf{u} = 0$$

$$\mathbf{M} = \Delta x \mathbf{I}$$

$$\mathbf{K} = \frac{1}{\Delta x^2}\begin{bmatrix} 2 & -1 & & -1 \\ -1 & 2 & -1 & \\ & \ddots & \ddots & \ddots \\ -1 & & -1 & 2 \end{bmatrix}$$

# Naive entropy stable POD-Galerkin

- **Offline**: compute a proper orthogonal decomposition (POD) basis

$$\mathbf{V} = \left[\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_N\right]$$

Note: linear approximation spaces are inefficient for convection-dominated PDEs

- **Online**: Galerkin projection of the matrix system. Let $\mathbf{u} = \mathbf{V}\mathbf{u}_N$

$$\mathbf{V}^T \left( \mathbf{M}\mathbf{V}\frac{\mathrm{d}\mathbf{u}_N}{\mathrm{d}t} + 2\left(\mathbf{Q} \circ \mathbf{F}\right)\mathbf{1} + \epsilon \mathbf{K}\mathbf{V}\mathbf{u}_N \right) = \mathbf{0}.$$

- **Online**: Utilize the entropy projection for an entropy stable ROM

$$\tilde{\mathbf{u}} = \boldsymbol{u}\left(\mathbf{P}\boldsymbol{v}\left(\mathbf{V}\mathbf{u}_N\right)\right), \qquad \mathbf{P} = \mathbf{V}\mathbf{V}^{\dagger}, \qquad \mathbf{F}_{ij} = \boldsymbol{f}_{EC}\left(\tilde{\mathbf{u}}_i, \tilde{\mathbf{u}}_j\right)$$

We construct POD basis from snapshots of *both* conservative and entropy variables to ensure accuracy of the entropy projection.

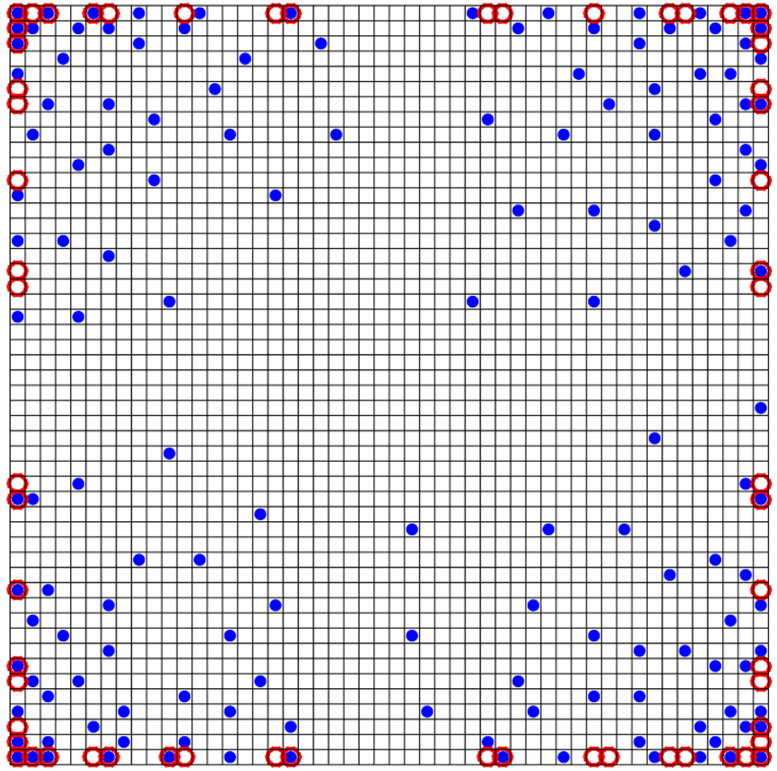
# Hyper-reduction for nonlinear terms

- Challenge: cost of ROM nonlinear terms scales with number of grid cells! The cost is still comparable to that of the full order model (FOM).

$$\tilde{\mathbf{u}} = \boldsymbol{u}\left(\mathbf{P}\boldsymbol{v}\left(\mathbf{V}\mathbf{u}_N\right)\right), \qquad 2\left(\mathbf{Q} \circ \mathbf{F}\right)\mathbf{1}$$

- Quadrature-like hyper-reduction: approximate nonlinear terms with

$$\mathbf{V}^T\boldsymbol{g}(\mathbf{V}\mathbf{u}_N) \approx \mathbf{V}(\mathcal{I},:)^T\mathbf{W}\boldsymbol{g}(\mathbf{V}(\mathcal{I},:)\mathbf{u}_N)$$



- Examples: gappy POD, ECSW, empirical cubature…

Bui and Willcox, Chantarantabut and Sorensen, Farhat et al, Patera and Yano, Hernandez et al., …

# Hyper-reduction for nonlinear terms

- Challenge: cost of ROM nonlinear terms scales with number of grid cells! The cost is still comparable to that of the full order model (FOM).

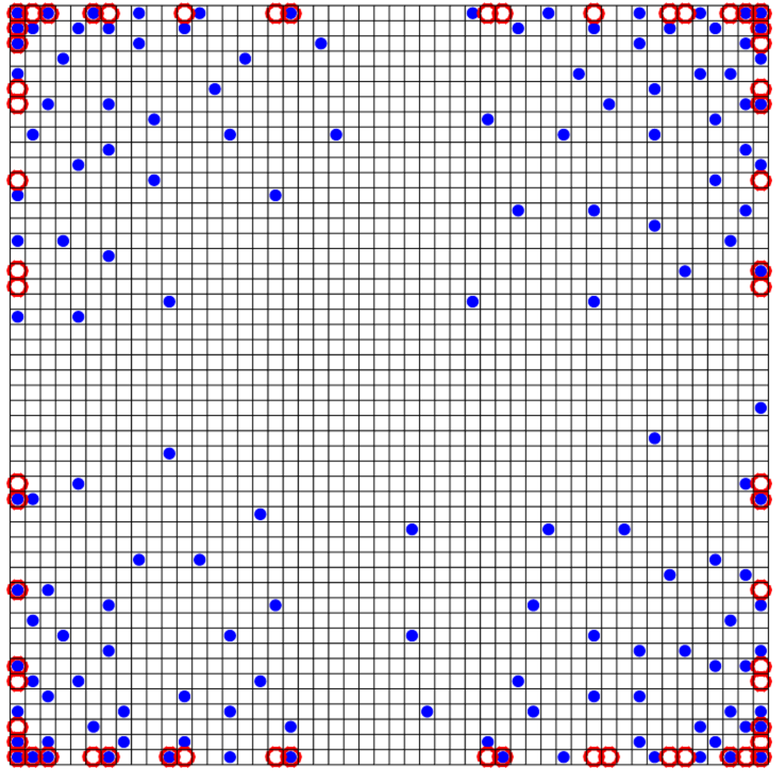
$$\tilde{\mathbf{u}} = \boldsymbol{u}\left(\mathbf{P}\boldsymbol{v}\left(\mathbf{V}\mathbf{u}_N\right)\right), \qquad 2\left(\mathbf{Q} \circ \mathbf{F}\right)\mathbf{1}$$

- Quadrature-like hyper-reduction: approximate nonlinear terms with

$$\mathbf{V}^T\boldsymbol{g}(\mathbf{V}\mathbf{u}_N) \approx \mathbf{V}(\mathcal{I},:)^T\mathbf{W}\boldsymbol{g}(\mathbf{V}(\mathcal{I},:)\mathbf{u}_N)$$

Sampled rows

- Examples: gappy POD, ECSW, empirical cubature…



Bui and Willcox, Chantarantabut and Sorensen, Farhat et al, Patera and Yano, Hernandez et al., …

# Hyper-reduction for nonlinear terms

- Challenge: cost of ROM nonlinear terms scales with number of grid cells! The cost is still comparable to that of the full order model (FOM).

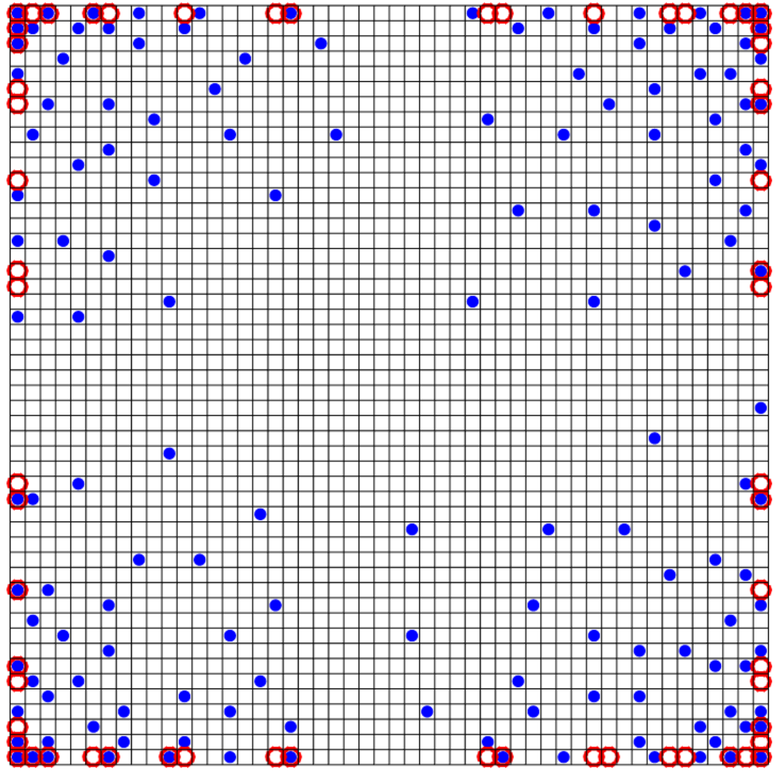
$$\tilde{\mathbf{u}} = \boldsymbol{u}\left(\mathbf{P}\boldsymbol{v}\left(\mathbf{V}\mathbf{u}_N\right)\right), \qquad 2\left(\mathbf{Q} \circ \mathbf{F}\right)\mathbf{1}$$

- Quadrature-like hyper-reduction: approximate nonlinear terms with

$$\mathbf{V}^T\boldsymbol{g}(\mathbf{V}\mathbf{u}_N) \approx \mathbf{V}(\mathcal{I},:)^T\mathbf{W}\boldsymbol{g}(\mathbf{V}(\mathcal{I},:)\mathbf{u}_N)$$

Diagonal weight matrix

- Examples: gappy POD, ECSW, empirical cubature…



Bui and Willcox, Chantarantabut and Sorensen, Farhat et al, Patera and Yano, Hernandez et al., …

# Entropy stability-preserving hyper-reduction

- How to hyper-reduce $(\mathbf{Q} \circ \mathbf{F})\,\mathbf{1}$ while retaining entropy stability?
- On periodic domains, skew-symmetry and conservation is sufficient:

$$\mathbf{Q} = -\mathbf{Q}^T, \qquad \mathbf{Q}\mathbf{1} = \mathbf{0}$$

# Entropy stability-preserving hyper-reduction

- How to hyper-reduce $(\mathbf{Q} \circ \mathbf{F})\,\mathbf{1}$ while retaining entropy stability?
- On periodic domains, skew-symmetry and conservation is sufficient:

$$\mathbf{Q} = -\mathbf{Q}^T, \qquad \mathbf{Q}\mathbf{1} = \mathbf{0}$$

- Solution: a “compress and project” hyper-reduction
  - Step 1: define a “test” basis $\mathbf{V}_t$ such that $\mathcal{R}(\mathbf{V}_t) = \operatorname{span}\left\{\mathbf{1}, \mathbf{V}, \mathbf{Q}^T\mathbf{V}\right\}$
  - Step 2: define the hyper-reduced matrix $\mathbf{Q}_s = \mathbf{P}_t^T \left(\mathbf{V}_t^T \mathbf{Q} \mathbf{V}_t\right) \mathbf{P}_t$
- Here, $\mathbf{P}_t = \left(\mathbf{V}_t^T \mathbf{W} \mathbf{V}_t\right)^{-1} \mathbf{V}_t^T \mathbf{W}$ is the $\mathbf{W}$-orthogonal projection.

# Entropy stable hyper-reduced ROM

- For 1D periodic domains:

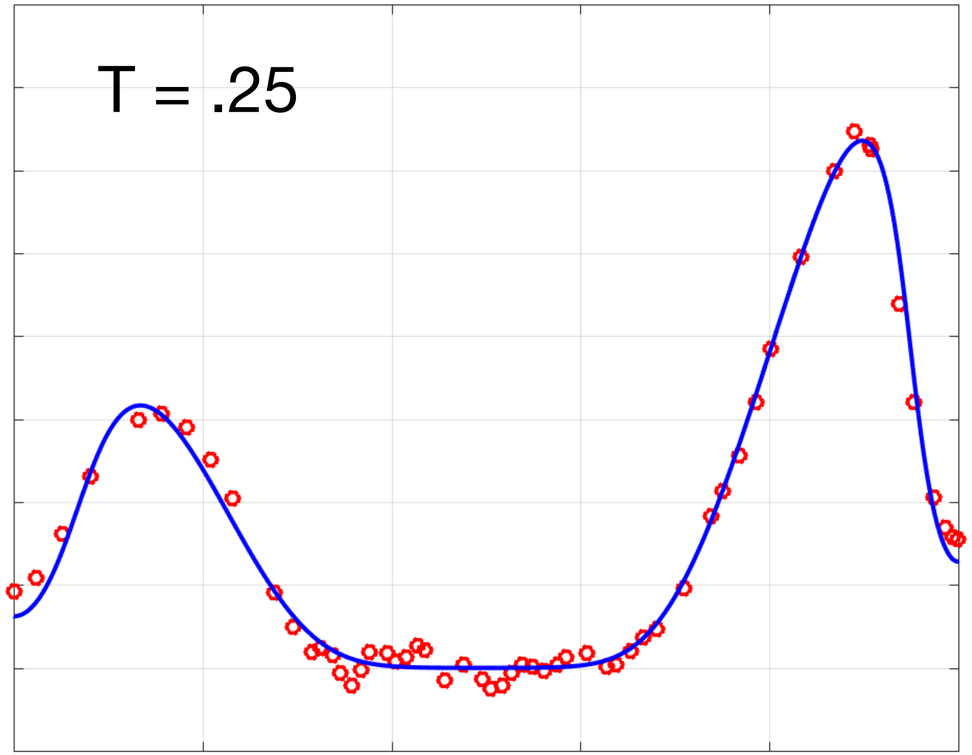
$$\mathbf{V}(\mathcal{I},:)^T \left( \mathbf{W}\mathbf{V}(\mathcal{I},:)\frac{\mathrm{d}\mathbf{u}_N}{\mathrm{d}t} + 2\left(\mathbf{Q}_s \circ \mathbf{F}\right)\mathbf{1} \right) + \epsilon \mathbf{V}^T\mathbf{K}\mathbf{V}\mathbf{u}_N = \mathbf{0}$$

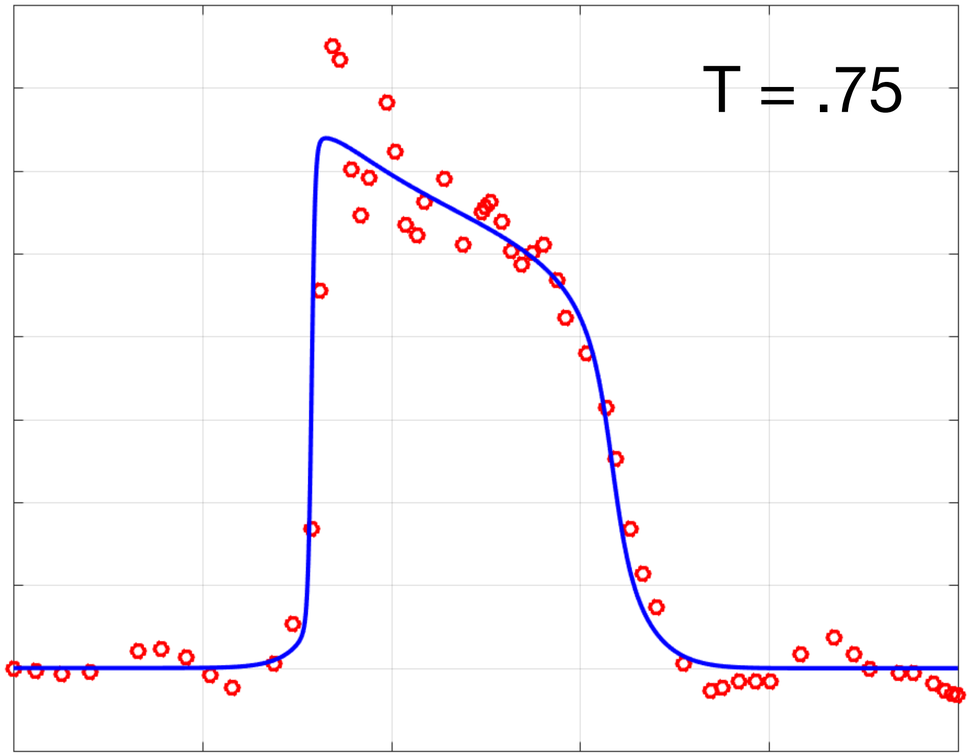
$$\mathbf{F}_{ij} = \boldsymbol{f}_{EC}\left(\tilde{\mathbf{u}}_i, \tilde{\mathbf{u}}_j\right), \qquad \tilde{\mathbf{u}} = \boldsymbol{u}\left(\mathbf{V}(\mathcal{I},:)\mathbf{P}\boldsymbol{v}\left(\mathbf{V}(\mathcal{I},:)\mathbf{u}_N\right)\right)$$

- Can extend to non-periodic domains using hybridized SBP operators.
- In higher dimensions, entropy stability requires a moment-preserving hyper-reduction strategy on the domain boundary (e.g., Caratheodory pruning).

# Entropy stable hyper-reduced ROM

## 1D compressible Euler equations, reflective BCs, T=.75



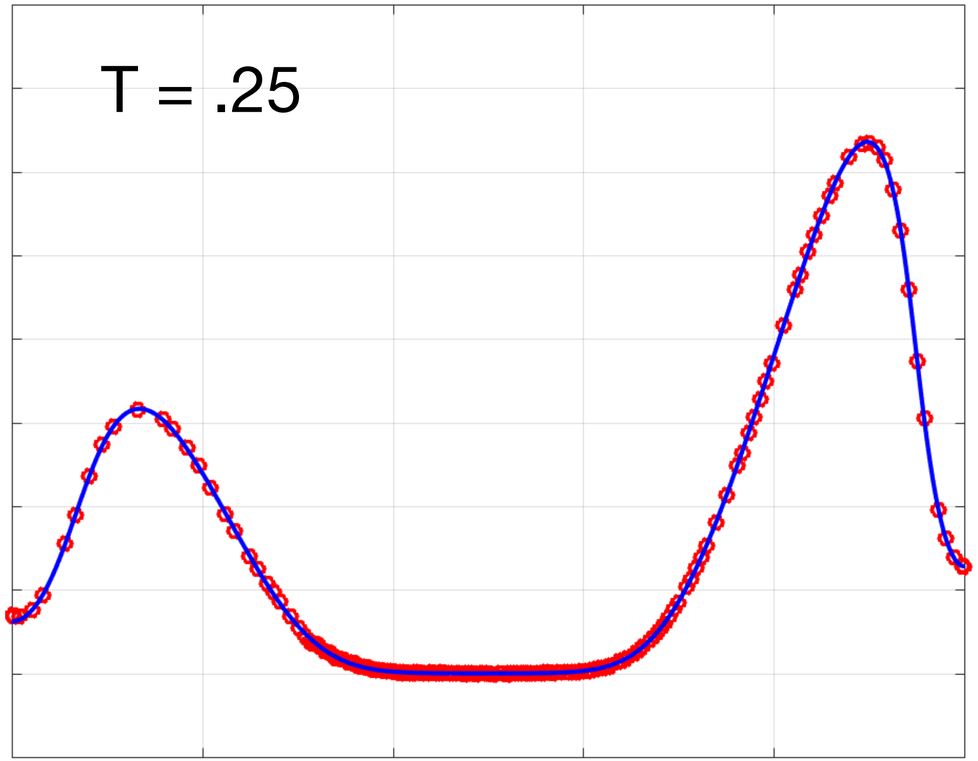



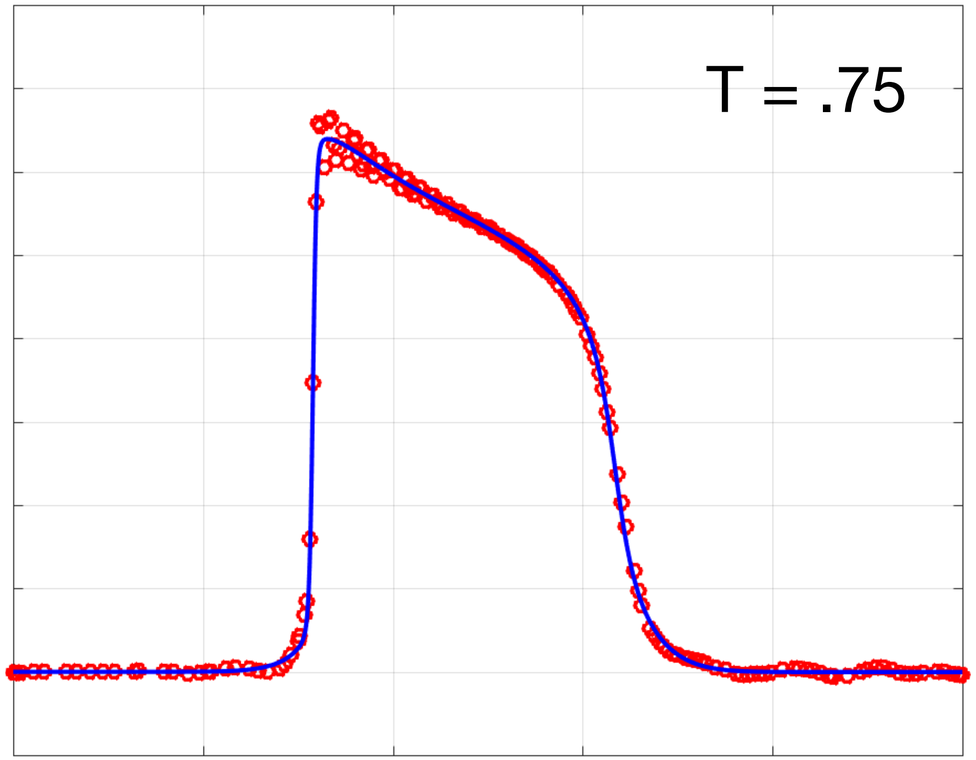

FOM with 2500 grid cells, viscosity $\epsilon = 2 \times 10^{-4}$, ROM with 25 modes.

Roughly twice as many hyper-reduced points as POD modes.

# Entropy stable hyper-reduced ROM

## 1D compressible Euler equations, reflective BCs, T=.75



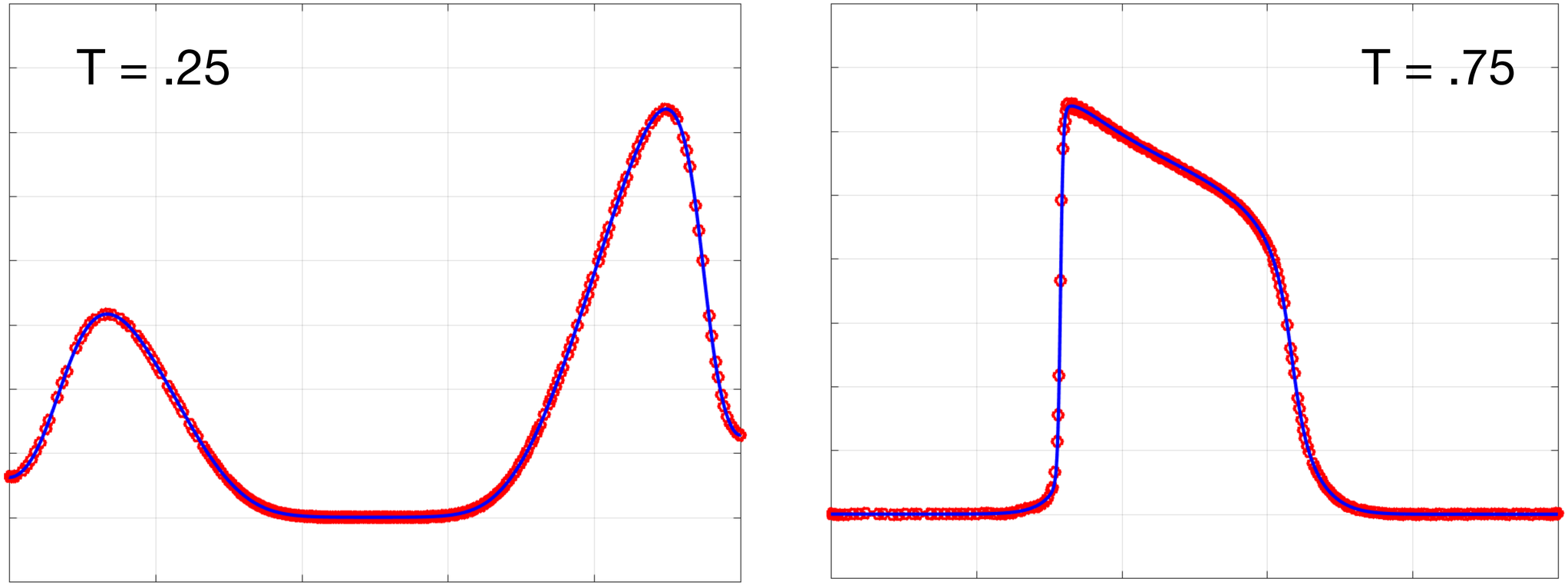




FOM with 2500 grid cells, viscosity $\epsilon = 2 \times 10^{-4}$, ROM with 75 modes.

Roughly twice as many hyper-reduced points as POD modes.

# Entropy stable hyper-reduced ROM

## 1D compressible Euler equations, reflective BCs, T=.75

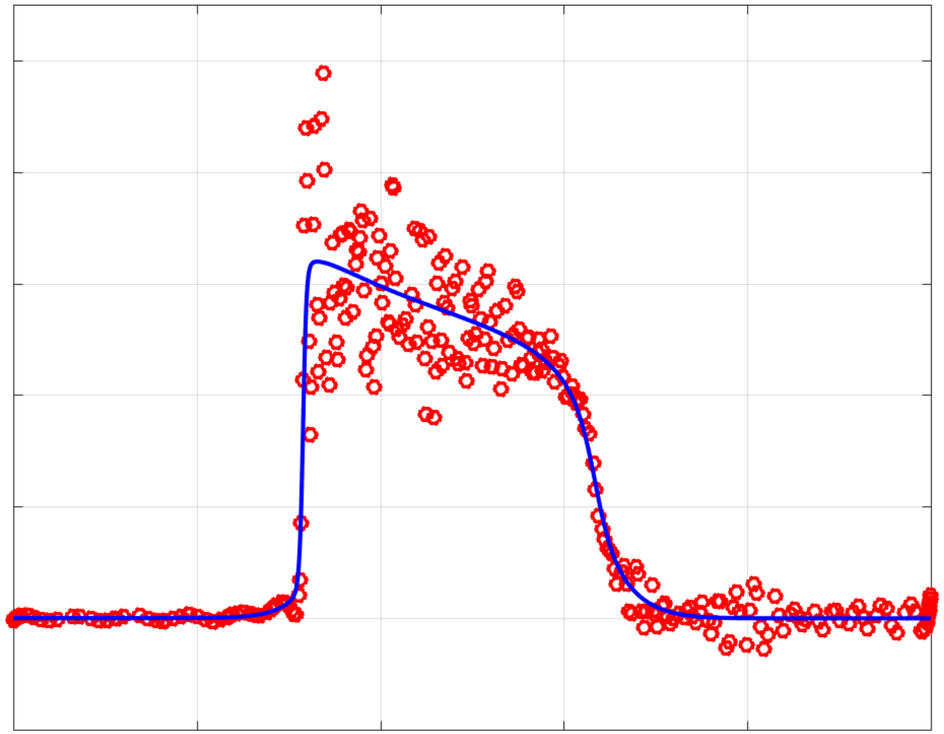



FOM with 2500 grid cells, viscosity $\epsilon = 2 \times 10^{-4}$, ROM with 125 modes.

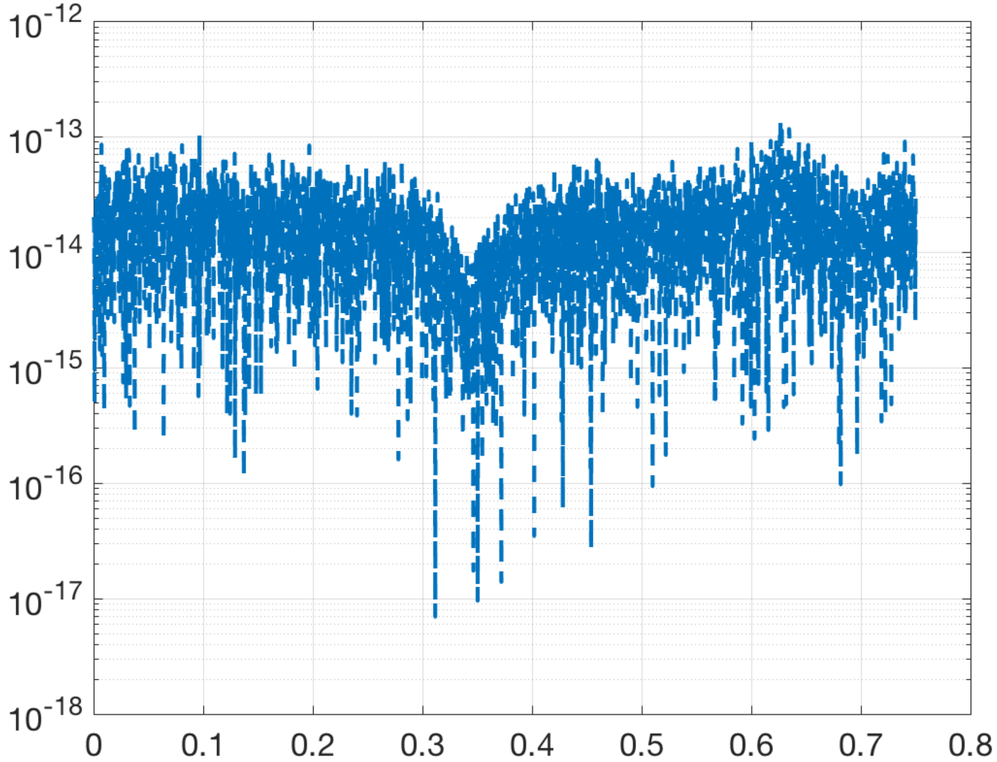
Roughly twice as many hyper-reduced points as POD modes.

# Entropy stable hyper-reduced ROM

## 1D compressible Euler equations, reflective BCs, T=.75



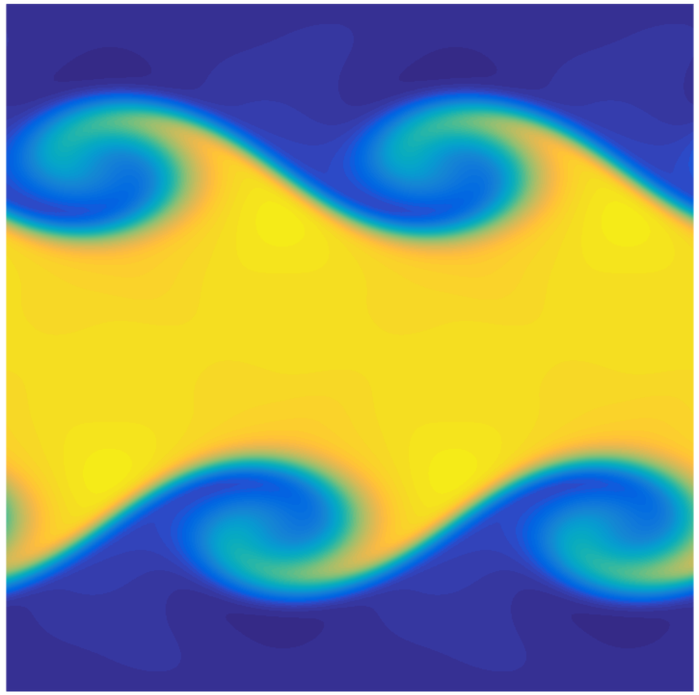
Entropy stable ROM remains stable even when run with zero viscosity



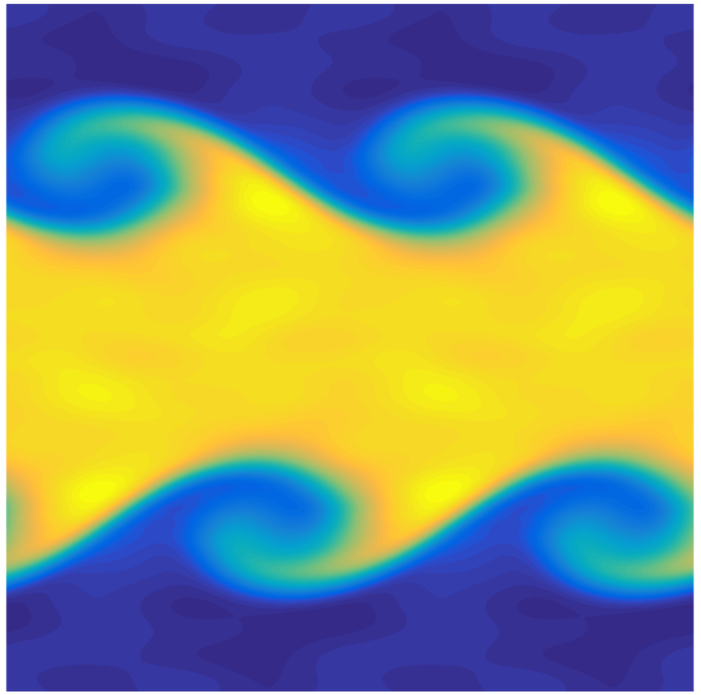

We can verify that the rate of change of the average entropy over time is zero.

# 2D ROM experiments

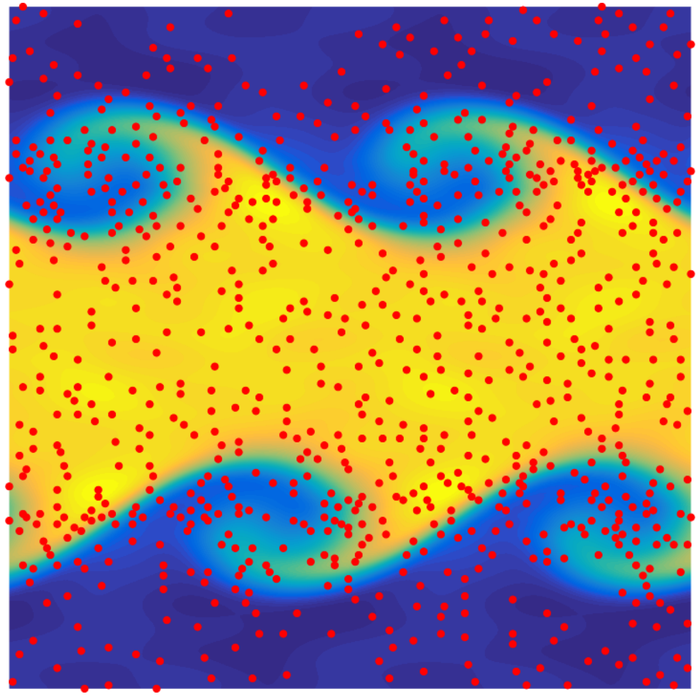
## Smoothed Kelvin-Helmholtz instability, T = 3



FOM: $200^2$ points,
viscosity $\epsilon = 10^{-3}$



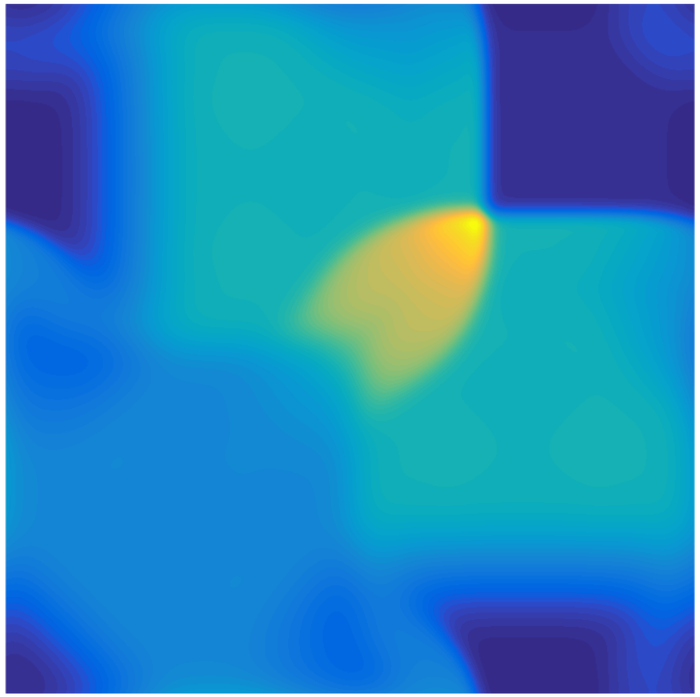
HR-ROM: 75 modes,
1.02% relative error



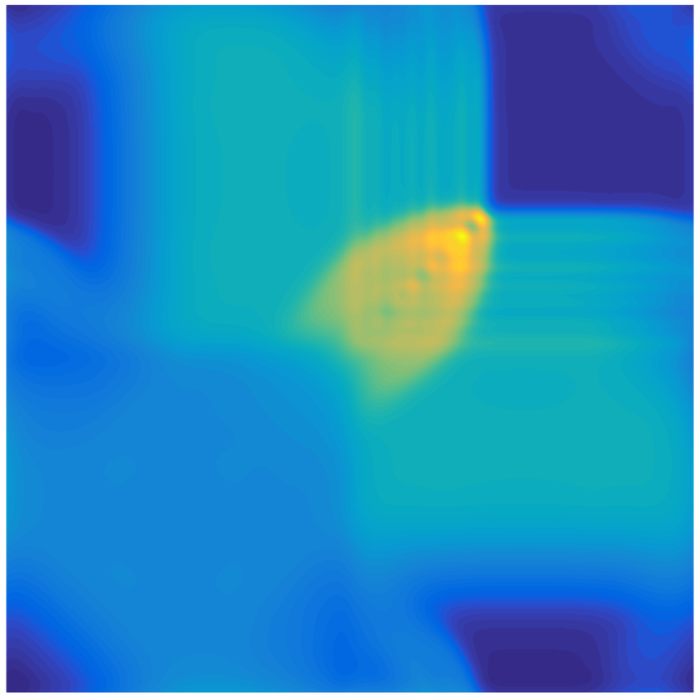
HR-ROM: 884 hyper-reduced points

# 2D ROM experiments

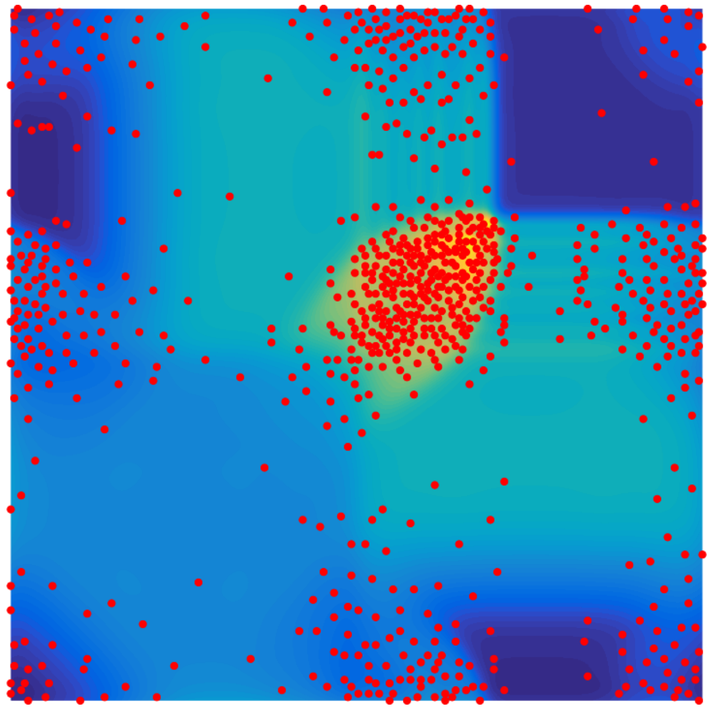
## Riemann problem (periodic), T = .25



FOM: $200^2$ points,
viscosity $\epsilon = 5 \times 10^{-3}$



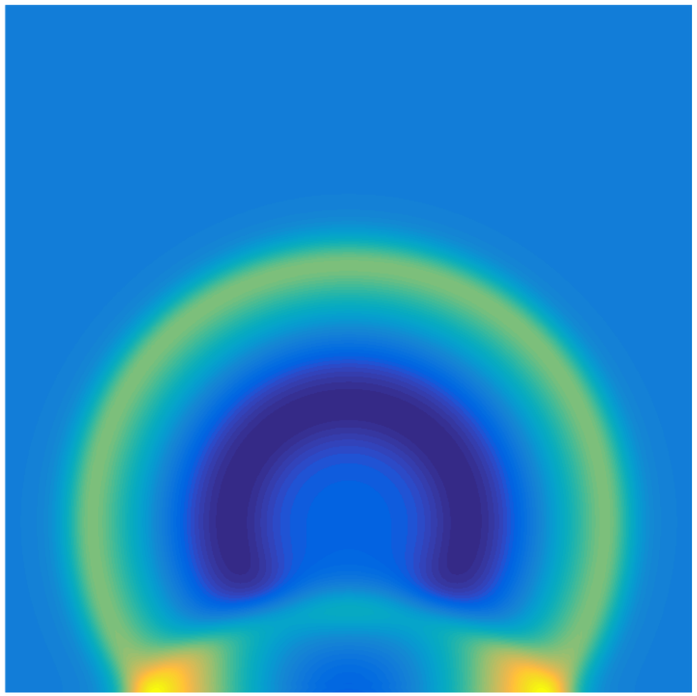
HR-ROM: 50 modes,
3.28% relative error



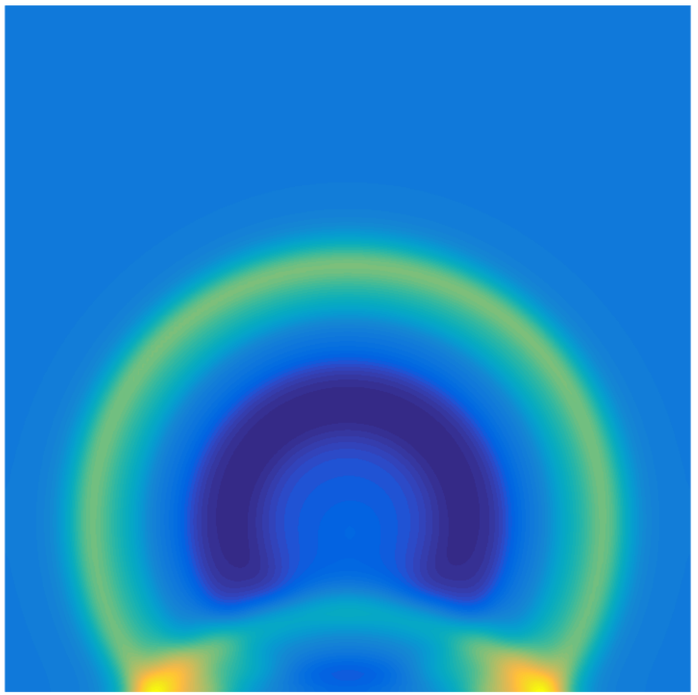
HR-ROM: 812 hyper-reduced points

# 2D ROM experiments

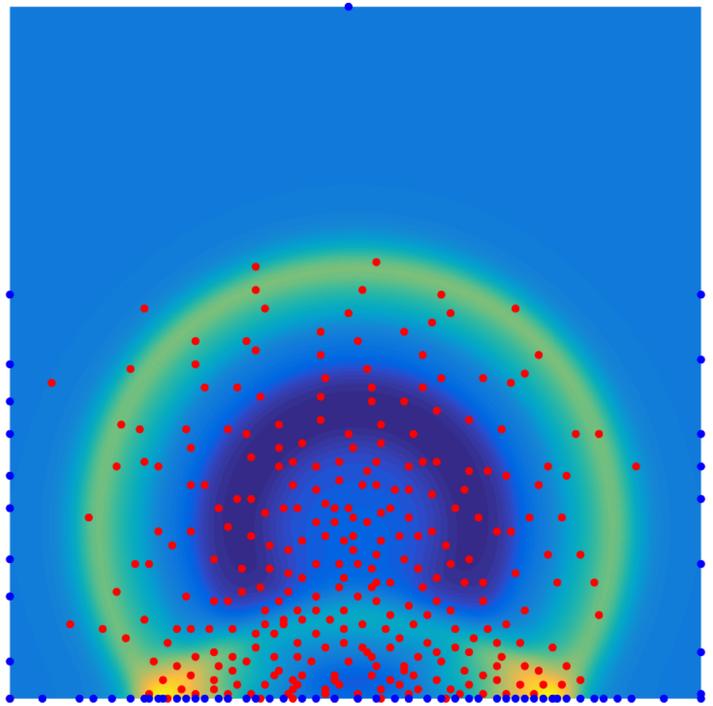
## Gaussian pulse, T = .25



FOM: $100^2$ points,
viscosity $\epsilon = 10^{-3}$



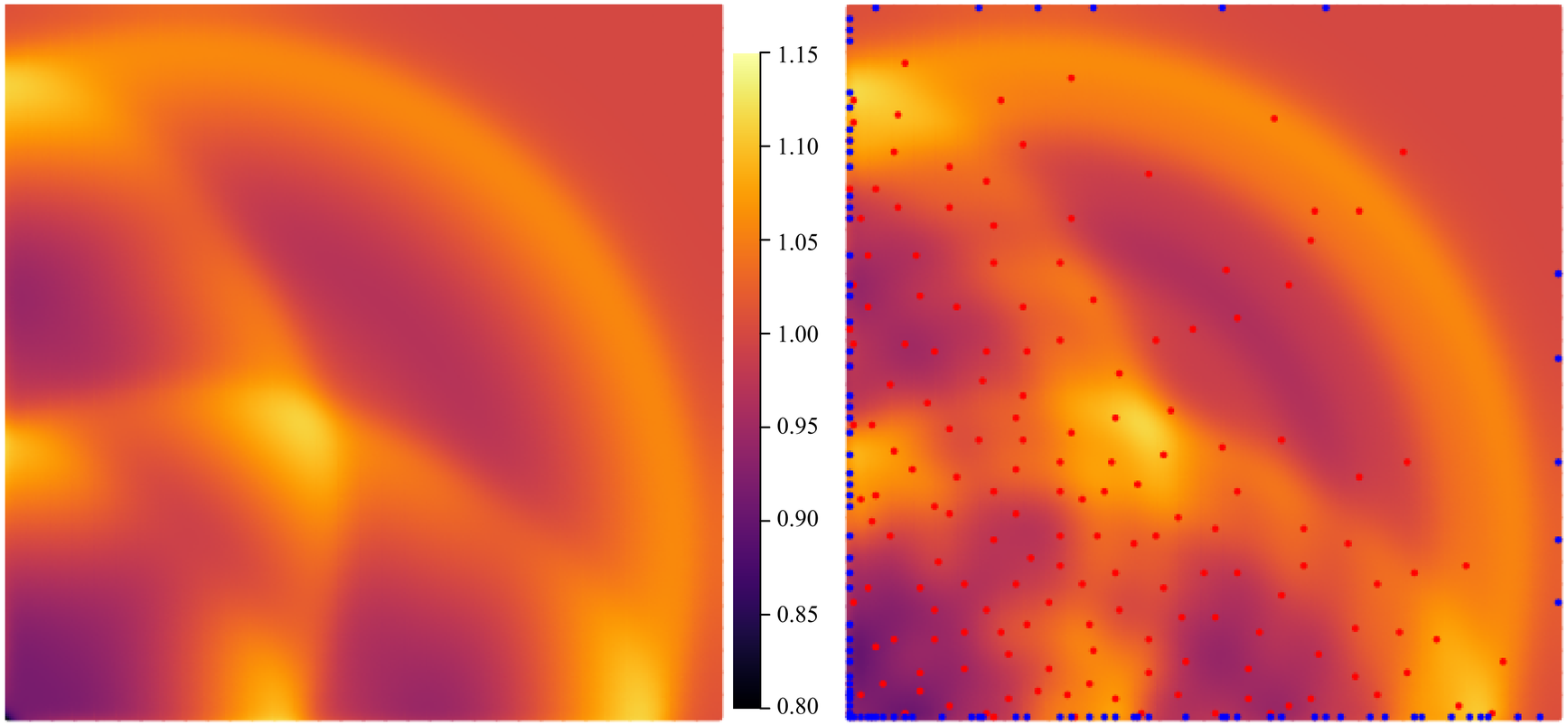
HR-ROM: 25 modes,
0.57% relative error



HR-ROM: 306 hyper-reduced volume points,
82 surface points

# 2D ROM experiments

## Gaussian pulse, T = .25




High order DG FOM, hyper-reduced ROM.

Current work: extension to high order FOMs, domain decomposition ROMs

**Qu**, Narayan, **Chan** (in preparation). “Entropy stable reduced order modeling of nonlinear conservation laws using DG methods”

# Entropy stable DG methods via knapsack limiting

# Summary of knapsack limiting

- Entropy stable DG methods satisfy a cell entropy inequality.
- Knapsack limiting: new way to enforce a cell entropy inequality on nodal DG
  - Enforces an entropy inequality using spatially localized blending; minimizes numerical dissipation via an efficient knapsack optimization problem.
  - Trade a little more dissipation for flexibility - enforces entropy stability for less expensive entropy conservative flux alternatives such as kinetic energy preserving (KEP), asymptotically entropy conservative (AEC) fluxes.
  - More flexible with respect to the equations, EOS, etc.

Lin, Chan (2024). *High order entropy stable discontinuous Galerkin spectral element methods through subcell limiting*.
De Michele, Coppola (2023). *Asymptotically entropy-conservative and kinetic-energy preserving numerical fluxes for compressible Euler equations.*
Tamaki, Kuya, Kawai (2022). *Comprehensive analysis of EC property of non-dissipative schemes for compressible flows: KEEP scheme redefined.*

# Subcell *blending* for nodal DG

- Assume: conservative high and low order collocation discretization

$$\mathbf{M}\frac{\mathrm{d}\mathbf{u}_H}{\mathrm{d}t} - \mathbf{r}_H(\mathbf{u}_H) + \mathbf{E}^T\mathbf{B}\boldsymbol{f}^*\left(\mathbf{u}^+, \mathbf{u}^-\right) = \mathbf{0}$$

$$\mathbf{M}\frac{\mathrm{d}\mathbf{u}_L}{\mathrm{d}t} - \mathbf{r}_L(\mathbf{u}_L) + \mathbf{E}^T\mathbf{B}\boldsymbol{f}^*\left(\mathbf{u}^+, \mathbf{u}^-\right) = \mathbf{0}.$$

$$\mathbf{1}^T\mathbf{r}_H(\mathbf{u}_H) = \mathbf{1}^T\mathbf{r}_L(\mathbf{u}_L) = 0$$

- We have subcell resolution; can we blend in a subcell manner too?

# “Finite volume” algebraic reformulation of nodal DG

$$\mathbf{M}_{ii}\frac{\mathrm{d}\mathbf{u}_i}{\mathrm{d}t} + \mathbf{r}_i(\mathbf{u}) + \begin{bmatrix} -\mathbf{f}_L^* \\ \mathbf{0} \\ \mathbf{f}_R^* \end{bmatrix} = \mathbf{0}, \qquad \underbrace{\mathbf{1}^T\mathbf{r}(\mathbf{u}) = 0}_{\text{conservation}}$$

Rewrite volume term in “finite volume” flux form to take advantage of FV-type blending.

# “Finite volume” algebraic reformulation of nodal DG

$$\mathbf{M}_{ii}\frac{\mathrm{d}\mathbf{u}_i}{\mathrm{d}t} + \mathbf{r}_i(\mathbf{u}) + \begin{bmatrix} -\mathbf{f}_L^* \\ \mathbf{0} \\ \mathbf{f}_R^* \end{bmatrix} = \mathbf{0}, \qquad \underbrace{\mathbf{1}^T\mathbf{r}(\mathbf{u}) = 0}_{\text{conservation}}$$

$$\bar{\mathbf{f}}_{i+1} - \bar{\mathbf{f}}_i$$

Rewrite volume term in “finite volume” flux form to take advantage of FV-type blending.

# “Finite volume” algebraic reformulation of nodal DG

$$\mathbf{M}_{ii}\frac{\mathrm{d}\mathbf{u}_i}{\mathrm{d}t} + \mathbf{r}_i(\mathbf{u}) + \begin{bmatrix} -\mathbf{f}_L^* \\ \mathbf{0} \\ \mathbf{f}_R^* \end{bmatrix} = \mathbf{0}, \qquad \underbrace{\mathbf{1}^T\mathbf{r}(\mathbf{u}) = 0}_{\text{conservation}}$$

$$\bar{\mathbf{f}}_{i+1} - \bar{\mathbf{f}}_i$$

$$\bar{\mathbf{f}}_1 = 0$$

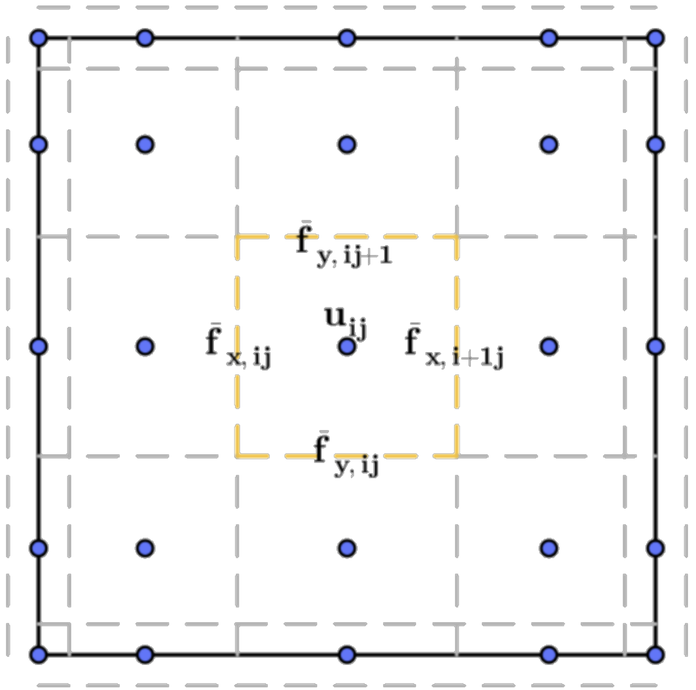
$$\bar{\mathbf{f}}_{1+i} = \sum_{j=1}^{i} \mathbf{r}_j, \qquad i = 1, \ldots, N+1$$

$$\bar{\mathbf{f}}_{N+2} = 0.$$

Rewrite volume term in “finite volume” flux form to take advantage of FV-type blending.

# Subcell blending for nodal DG methods

$$\mathbf{M}_{ii}\frac{\mathrm{d}\mathbf{u}_i}{\mathrm{d}t} + \underbrace{\left[\theta_{i+1}\bar{\mathbf{f}}^{\mathrm{H}}_{i+1} + (1-\theta_{i+1})\,\bar{\mathbf{f}}^{\mathrm{L}}_{i+1}\right]}_{\bar{\mathbf{f}}_{i+1}} - \underbrace{\left[\theta_i\bar{\mathbf{f}}^{\mathrm{H}}_{i} + (1-\theta_{i})\,\bar{\mathbf{f}}^{\mathrm{L}}_{i}\right]}_{\bar{\mathbf{f}}_{i}} = \text{interface fluxes.}$$




- Blend high and low order methods to enforce *nodal* bounds (e.g., positivity)
- Unfortunately, subcell blending does not preserve a cell entropy inequality!

# Knapsack limiting

- Assume low order method is entropy stable: solve an optimization problem to maximize limiting factors while enforcing a cell entropy inequality and

$$\sum_i \mathbf{v}_i^T \left(\bar{\mathbf{f}}_{i+1}(\boldsymbol{\theta}) - \bar{\mathbf{f}}_i(\boldsymbol{\theta})\right) \leq \left(\psi(\mathbf{u}_{N+1}) - \mathbf{v}_{N+1}^T \mathbf{f}_R^*\right) - \left(\psi(\mathbf{u}_1) - \mathbf{v}_1^T \mathbf{f}_L^*\right)$$

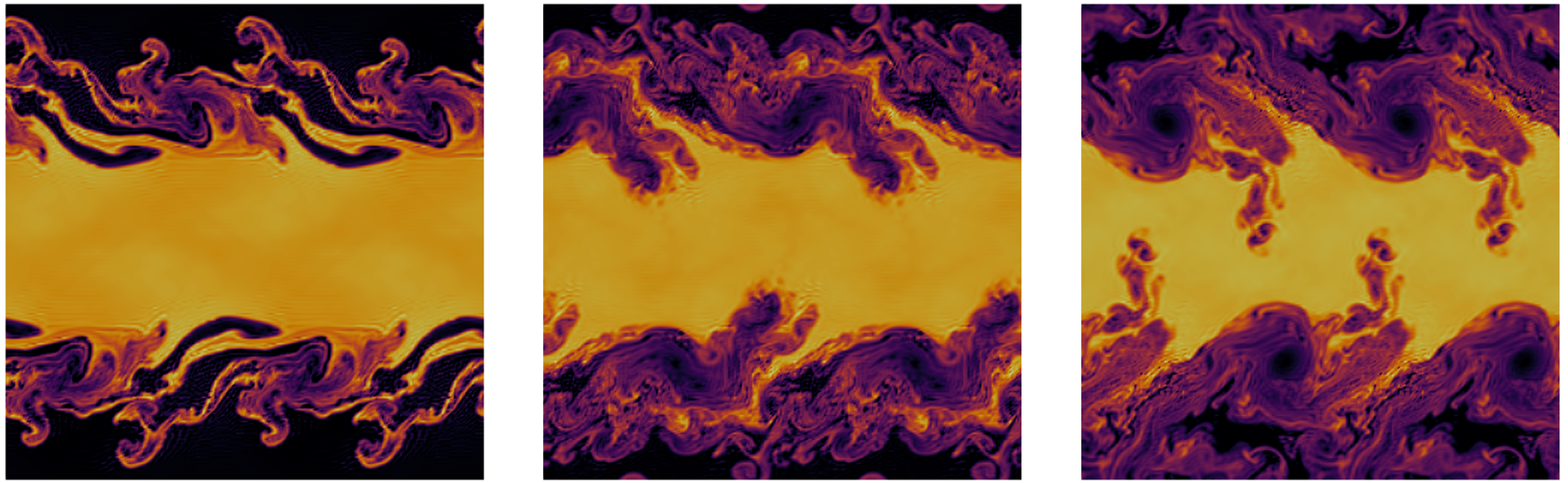
- Inequality is linear w.r.t. limiting parameters; yields a knapsack problem with highly efficient solution algorithms!

$$\max \sum_{i=1}^n \theta_i \quad \text{or} \quad \max \sum_{i=1}^n \theta_i^2$$

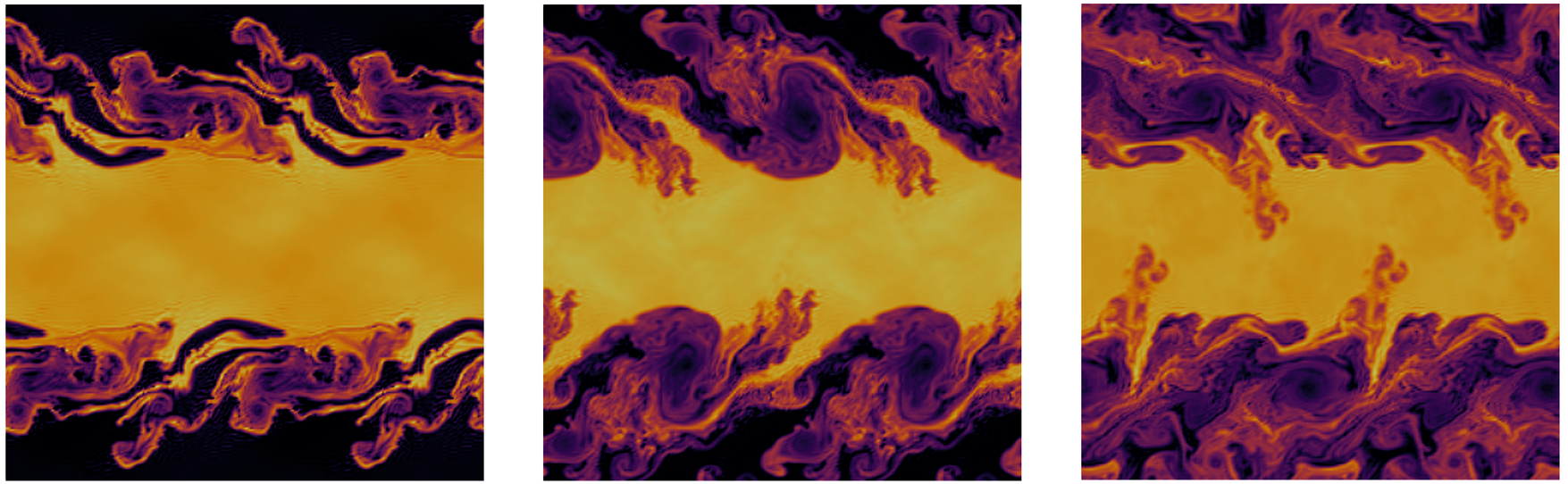
$$\text{s.t.} \quad \sum_{i=1}^n a_i \theta_i \leq b, \qquad 0 \leq \theta_i \leq \theta_i^{\max}.$$

# Long-time Kelvin-Helmholtz instability

ESDG

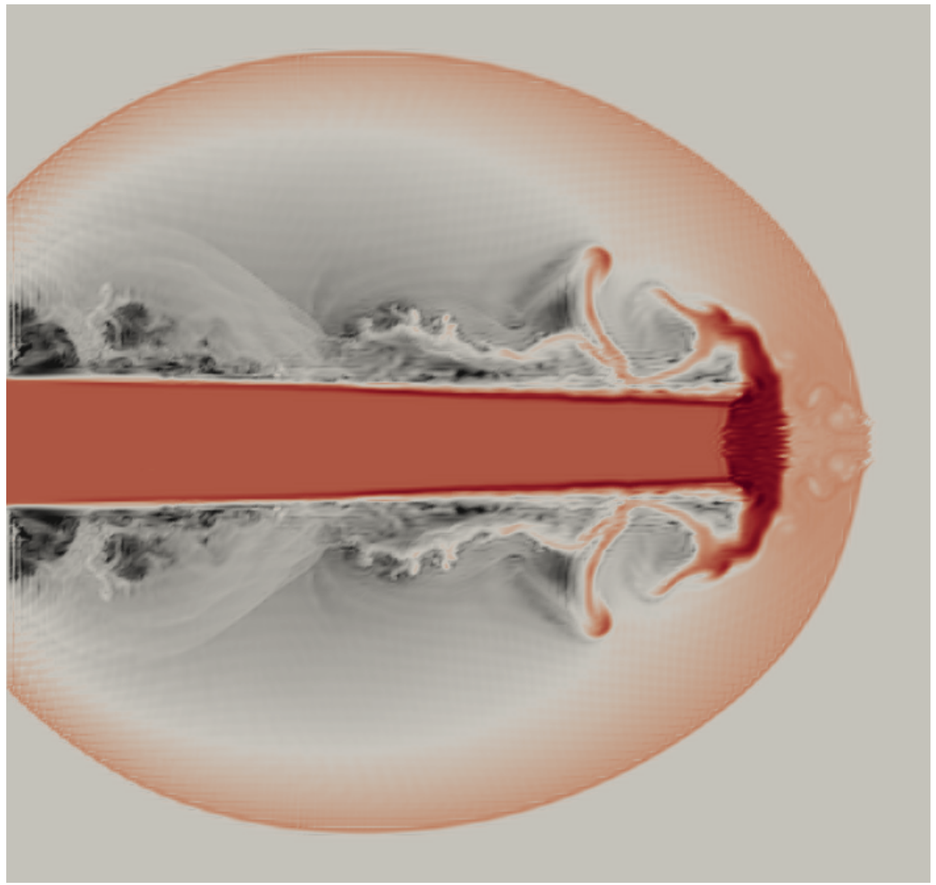


DGSEM

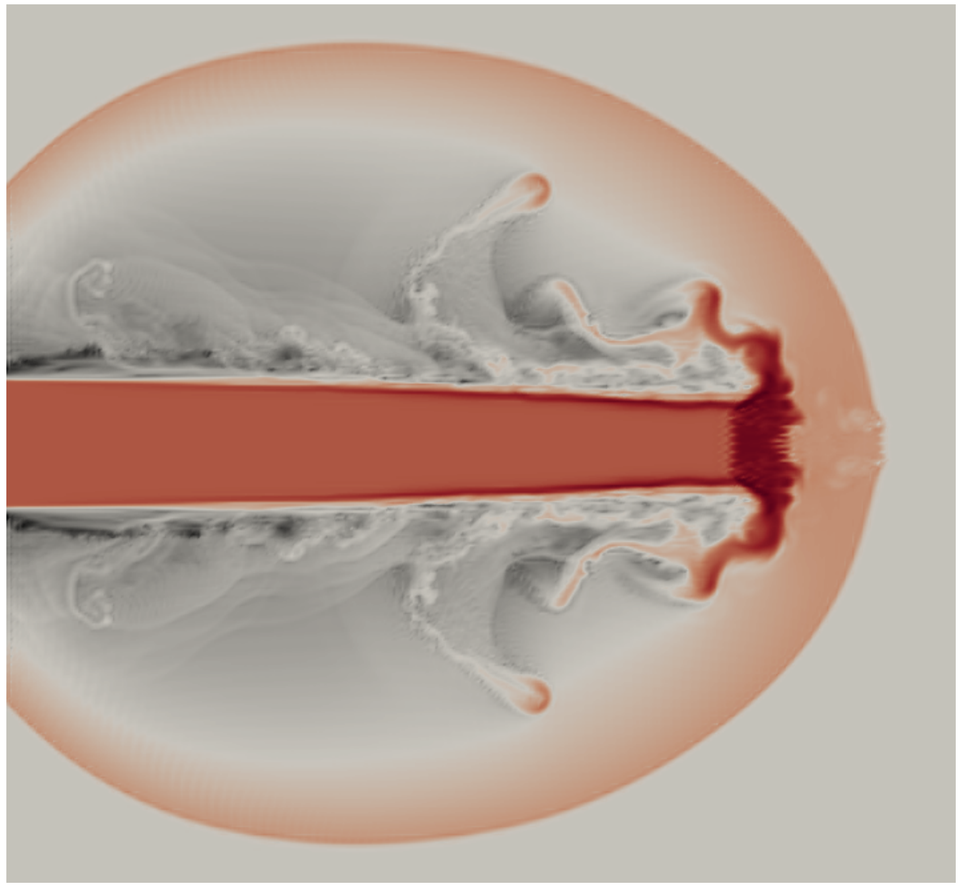


Lin, Chan (2024). *High order entropy stable discontinuous Galerkin spectral element methods through subcell limiting*.

# Mach 2000 astrophysical jet



ESDG with knapsack limiting



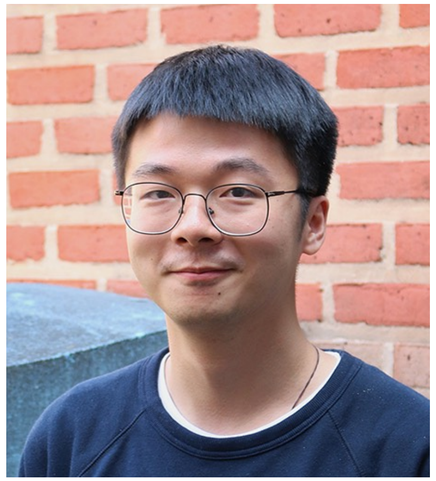
DGSEM with knapsack limiting

Degree $N = 3$, $150 \times 150$ elements with relaxed positivity bounds and cell entropy inequality.

Lin, Chan (2024). *High order entropy stable discontinuous Galerkin spectral element methods through subcell limiting*.

# Conclusion and acknowledgments

## Entropy stable methods enable robust high order accurate simulations



Dr. Yimin Lin
Positivity-preservation, knapsack limiting



Dr. Christina G. Taylor
Entropy stable cut-cell DG methods



Ray Qu
Entropy stable reduced order modeling



Brian Christner
Knapsack limiting

# Thank you!

## Questions?

This work has been supported by:

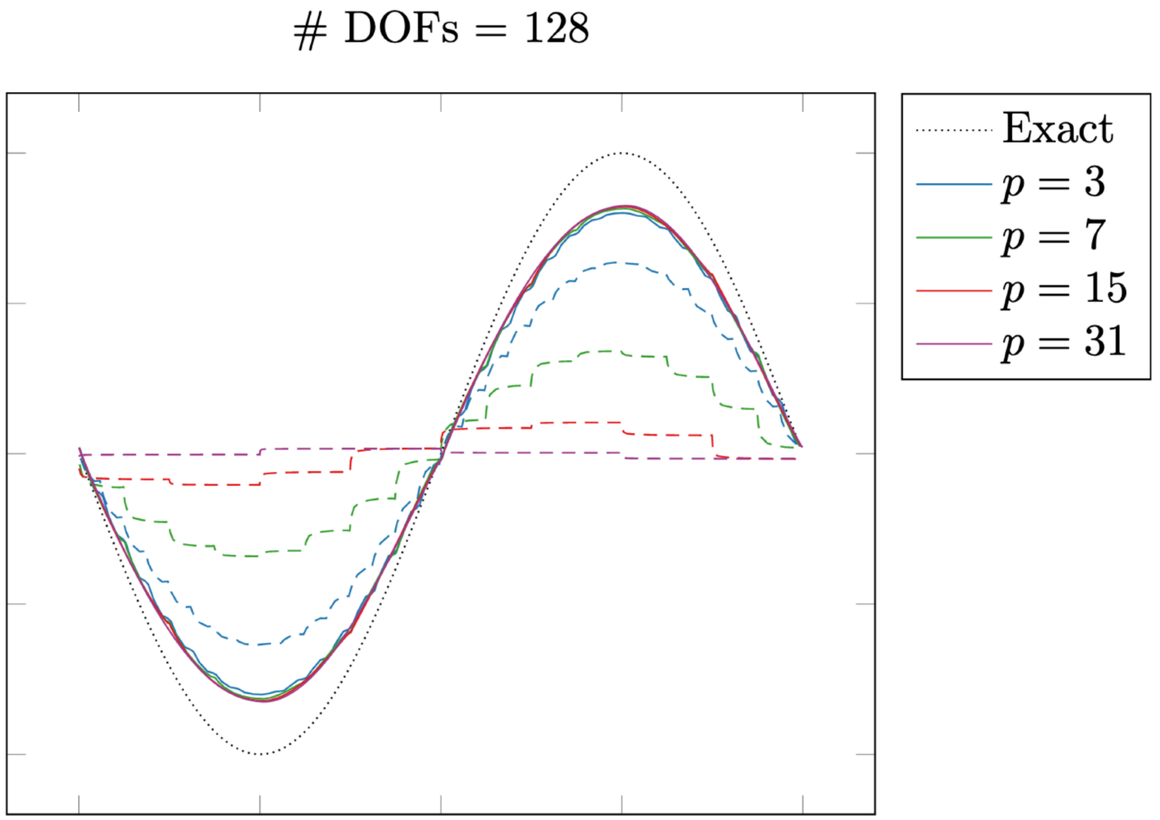
- NSF DMS-2231482 (2023-2028).
- NSF DMS-1943186 (2020-2025).

# Backup slides

# Low order positivity-preserving scheme: sparsified matrices

$$\mathbf{Q} = \frac{1}{2}\begin{bmatrix} -1 & 1 & & & \\ -1 & 0 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 0 & 1 \\ & & & -1 & 1 \end{bmatrix}$$

Sparsified subcell matrix satisfying the SBP property




Effect of sparsification vs no sparsification on the low order solution.

Pazner (2021). "Sparse invariant domain preserving discontinuous Galerkin methods with subcell convex limiting."

# Positivity preservation: 1D convergence rates

| | $N = 2$ | | $N = 5$ | |
|---|---|---|---|---|
| $h$ | $L^1$ error | Rate | $L^1$ error | Rate |
| 0.02 | $8.681 \times 10^{-2}$ | | $5.956 \times 10^{-2}$ | . |
| 0.01 | $3.658 \times 10^{-2}$ | 1.25 | $1.436 \times 10^{-2}$ | 2.05 |
| 0.005 | $1.329 \times 10^{-2}$ | 1.46 | $3.630 \times 10^{-3}$ | 1.98 |
| 0.0025 | $6.015 \times 10^{-3}$ | 1.14 | $1.129 \times 10^{-3}$ | 1.69 |
| 0.00125 | $2.910 \times 10^{-3}$ | 1.05 | $5.889 \times 10^{-4}$ | 0.94 |

(a) Leblanc shock tube, relaxation factor $\alpha = 0.5$

| | $N = 2$ | | $N = 3$ | |
|---|---|---|---|---|
| $h$ | $L^1$ error | Rate | $L^1$ error | Rate |
| 0.025 | $2.305 \times 10^{-2}$ | | $2.071 \times 10^{-2}$ | |
| 0.0125 | $9.858 \times 10^{-2}$ | 1.23 | $6.749 \times 10^{-3}$ | 1.62 |
| 0.00625 | $3.382 \times 10^{-3}$ | 1.54 | $1.278 \times 10^{-3}$ | 2.40 |
| 0.003125 | $5.765 \times 10^{-4}$ | 2.55 | $1.163 \times 10^{-4}$ | 3.45 |
| 0.0015625 | $8.836 \times 10^{-5}$ | 2.71 | $1.269 \times 10^{-5}$ | 3.20 |

(b) 1D viscous shock, Re = 1000, relaxation factor $\alpha = 0.5$

Viscous shock is run at Mach 20 to generate positivity violations.

# Positivity preservation: 2D convergence rates

| | $N = 2$ | | $N = 3$ | | $N = 4$ | |
|---|---|---|---|---|---|---|
| $h$ | $L^2$ error | Rate | $L^2$ error | Rate | $L^2$ error | Rate |
| 2.5 | $1.148 \times 10^{0}$ | | $5.958 \times 10^{-1}$ | 1.28 | $4.073 \times 10^{-1}$ | |
| 1.25 | $4.865 \times 10^{-1}$ | 1.24 | $1.905 \times 10^{-1}$ | 1.64 | $8.987 \times 10^{-2}$ | 2.18 |
| 0.625 | $1.223 \times 10^{-1}$ | 1.99 | $2.308 \times 10^{-2}$ | 3.05 | $1.511 \times 10^{-2}$ | 2.57 |
| 0.3125 | $1.706 \times 10^{-2}$ | 2.84 | $2.393 \times 10^{-3}$ | 3.27 | $1.915 \times 10^{-4}$ | 6.30 |

(a) Quadrilateral meshes, relaxation factor $\alpha = 0.5$

| | $N = 2$ | | $N = 3$ | | $N = 4$ | |
|---|---|---|---|---|---|---|
| $h$ | $L^2$ error | Rate | $L^2$ error | Rate | $L^2$ error | Rate |
| 2.5 | $7.887 \times 10^{-1}$ | | $5.034 \times 10^{-1}$ | | $4.059 \times 10^{-1}$ | |
| 1.25 | $3.834 \times 10^{-1}$ | 1.04 | $1.881 \times 10^{-1}$ | 1.42 | $9.890 \times 10^{-2}$ | 2.04 |
| 0.625 | $8.993 \times 10^{-2}$ | 2.09 | $2.944 \times 10^{-2}$ | 2.68 | $1.578 \times 10^{-2}$ | 2.65 |
| 0.3125 | $1.298 \times 10^{-2}$ | 2.79 | $2.606 \times 10^{-3}$ | 3.50 | $4.258 \times 10^{-4}$ | 5.21 |

(b) Triangular meshes, relaxation factor $\alpha = 0.5$

Challenging vortex parameters: $\rho_{\min} = 2.145 \times 10^{-3}$!

# Minimum entropy principle vs cell entropy inequality

Enforcing a pointwise minimum entropy principle

| | $N=2$ | | $N=3$ | | $N=4$ | |
|---|---|---|---|---|---|---|
| K | $L^2$ error | Rate | $L^2$ error | Rate | $L^2$ error | Rate |
| 5 | $7.498 \times 10^{-1}$ | | $4.499 \times 10^{-1}$ | | $3.135 \times 10^{-1}$ | |
| 10 | $3.343 \times 10^{-1}$ | 1.17 | $2.109 \times 10^{-1}$ | 1.09 | $1.486 \times 10^{-1}$ | 1.08 |
| 20 | $1.894 \times 10^{-1}$ | 0.82 | $1.092 \times 10^{-1}$ | 0.95 | $7.509 \times 10^{-2}$ | 0.98 |
| 40 | $9.718 \times 10^{-2}$ | 0.96 | $5.956 \times 10^{-2}$ | 0.87 | $4.160 \times 10^{-2}$ | 0.85 |
| 80 | $5.116 \times 10^{-2}$ | 0.93 | $3.186 \times 10^{-2}$ | 0.90 | $2.157 \times 10^{-2}$ | 0.95 |

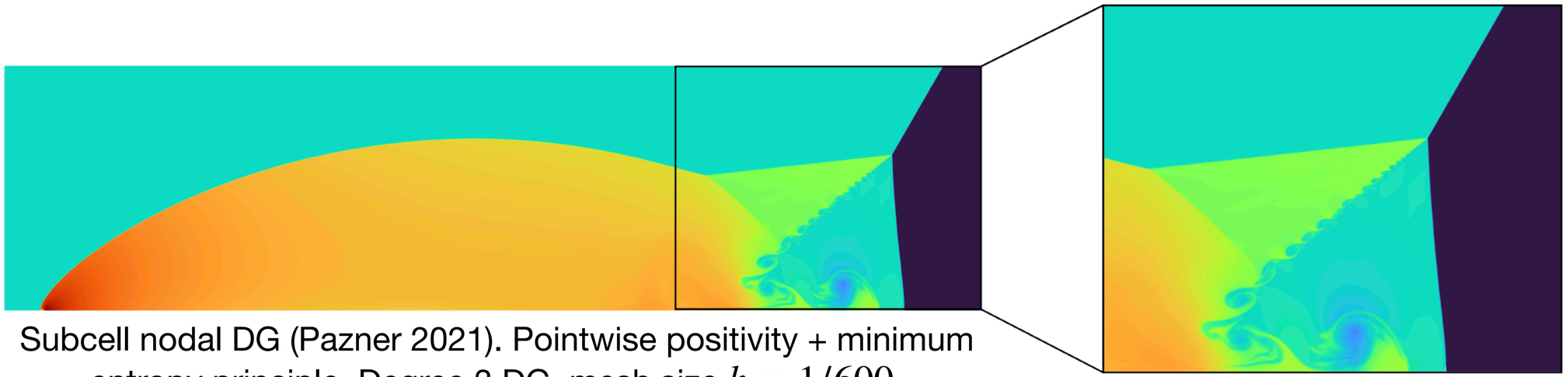
| | $N=2$ | | $N=3$ | | $N=4$ | |
|---|---|---|---|---|---|---|
| K | $L^2$ error | Rate | $L^2$ error | Rate | $L^2$ error | Rate |
| 5 | $6.935 \times 10^{-1}$ | | $2.498 \times 10^{-1}$ | | $1.587 \times 10^{-1}$ | |
| 10 | $1.785 \times 10^{-1}$ | 1.96 | $7.083 \times 10^{-2}$ | 1.82 | $2.000 \times 10^{-2}$ | 2.99 |
| 20 | $4.126 \times 10^{-2}$ | 1.11 | $8.898 \times 10^{-3}$ | 2.99 | $9.557 \times 10^{-4}$ | 4.39 |
| 40 | $6.714 \times 10^{-3}$ | 2.62 | $8.163 \times 10^{-4}$ | 3.45 | $3.142 \times 10^{-5}$ | 4.93 |
| 80 | $1.210 \times 10^{-3}$ | 2.74 | $4.208 \times 10^{-5}$ | 4.28 | $1.530 \times 10^{-6}$ | 4.36 |

Enforcing a cell entropy inequality

# Double Mach reflection



Subcell positivity-preserving entropy stable nodal DG. Degree 3 DG, mesh size $h = 1/250$.



Subcell nodal DG (Pazner 2021). Pointwise positivity + minimum entropy principle. Degree 3 DG, mesh size $h = 1/600$.

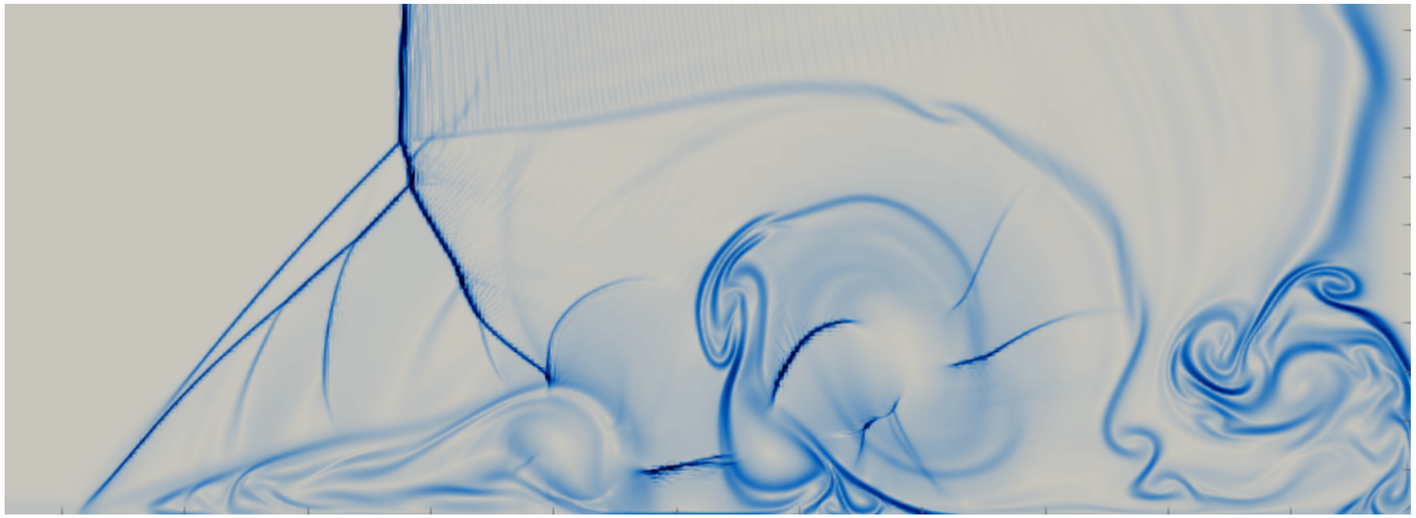
Pazner (2021). "Sparse invariant domain preserving discontinuous Galerkin methods with subcell convex limiting."

# Our contribution: positivity preserving entropy stable nodal DG

## Application: compressible Navier-Stokes equations

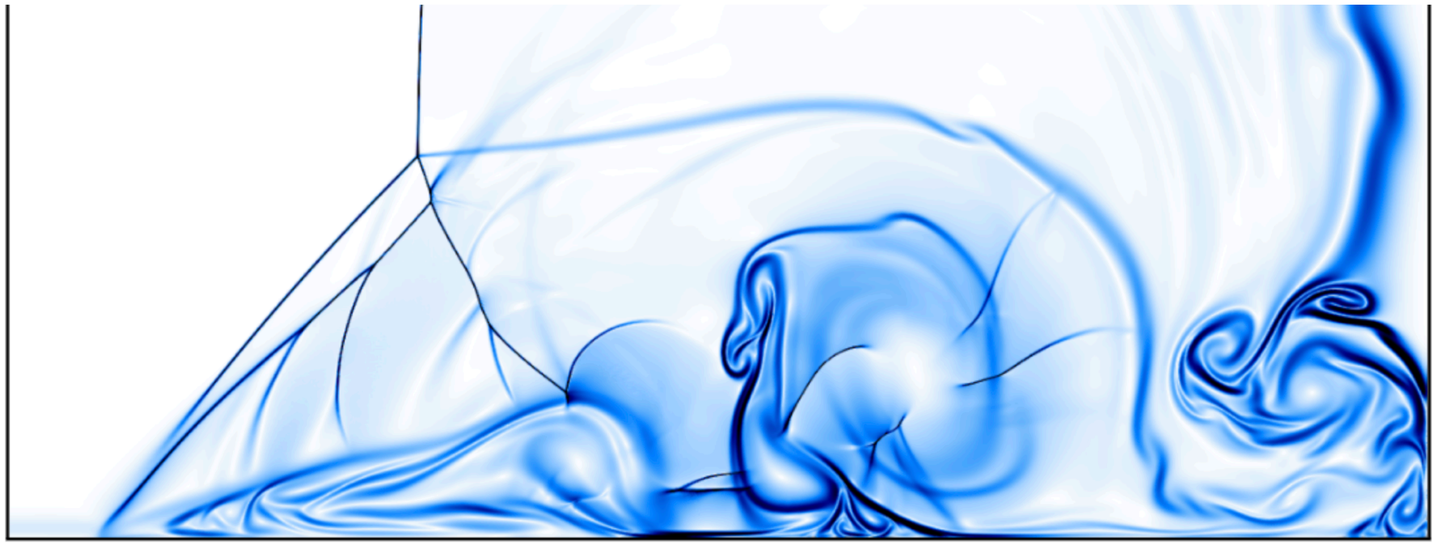
### Lin, Chan, Tomas 2023

4th order entropy stable DG
300 x 100 uniform mesh
Subcell positivity limiting.



### Guermond et al. 2022

Fine-grid reference solution
2nd order invariant-domain P1-FEM
Graded mesh, 512M dofs



**Lin, Chan**, and Tomas (2023). "A positivity preserving strategy for entropy stable DG discretizations of the compressible Euler and Navier-Stokes equations."

# Our contribution: positivity preserving entropy stable nodal DG

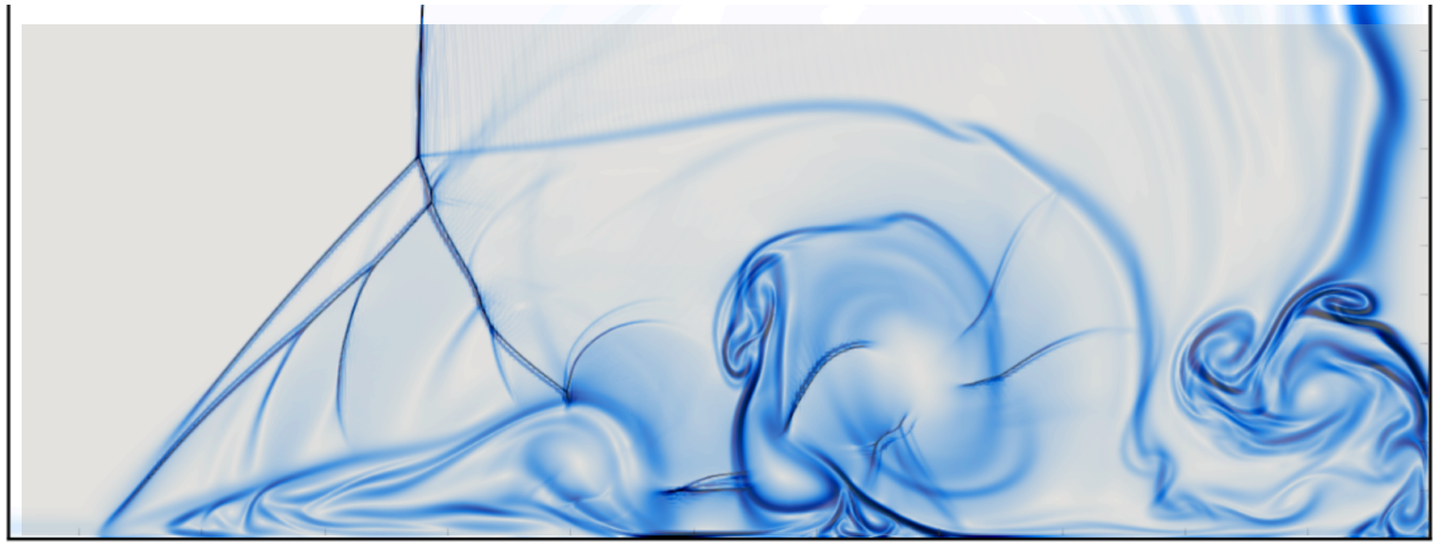
## Application: compressible Navier-Stokes equations

### Lin, Chan, Tomas 2023

4th order entropy stable DG
300 x 100 uniform mesh
Subcell positivity limiting.

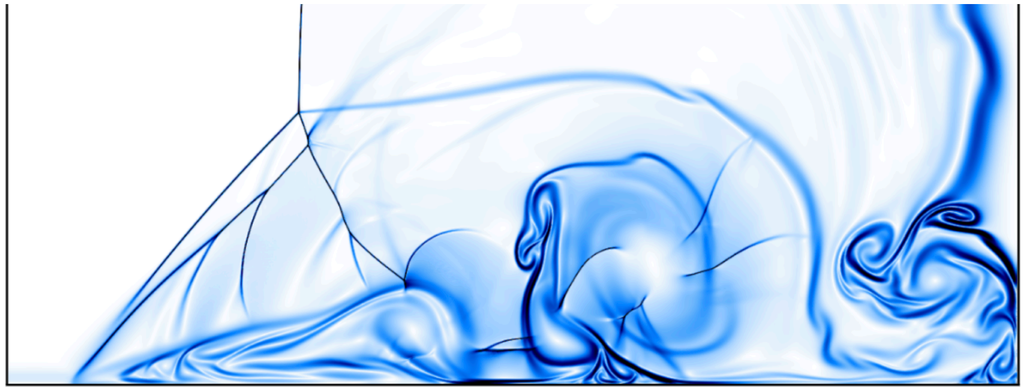
### Guermond et al. 2022

Fine-grid reference solution
2nd order invariant-domain P1-FEM
Graded mesh, 512M dofs

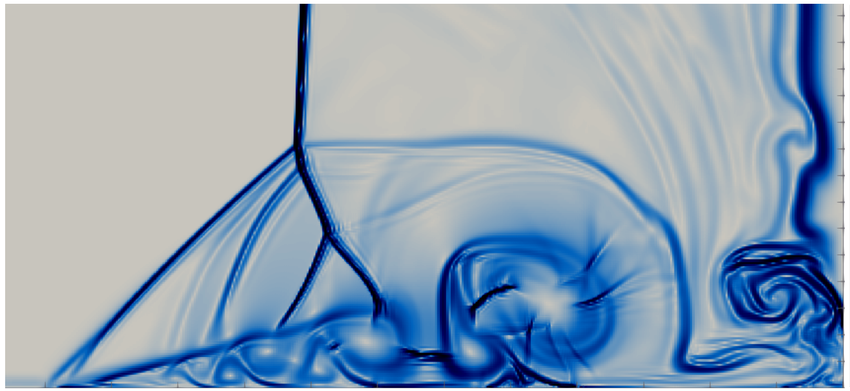


**Lin, Chan**, and Tomas (2023). "A positivity preserving strategy for entropy stable DG discretizations of the compressible Euler and Navier-Stokes equations."

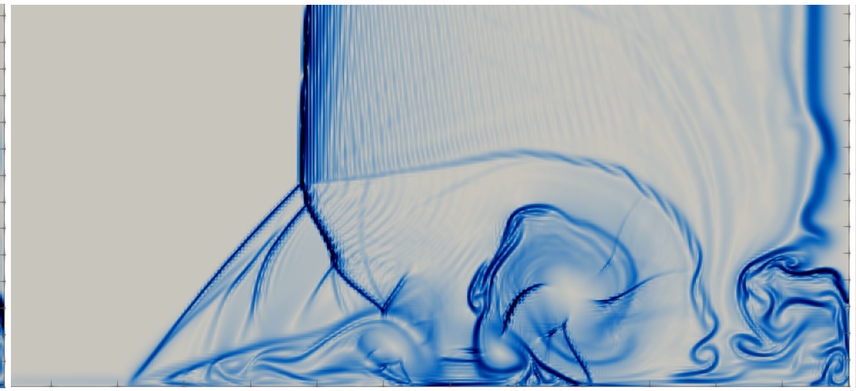
# Coarsening the mesh and increasing order



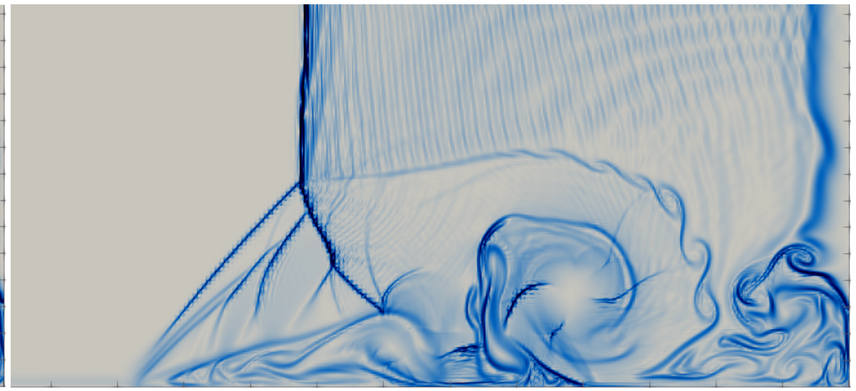
Guermond et al (2022) reference solution



Degree 1, 600 x 300 mesh



Degree 2, 400 x 200 mesh



Degree 3, 300 x 150 mesh

**Lin, Chan**, and Tomas (2023). "A positivity preserving strategy for entropy stable DG discretizations of the compressible Euler and Navier-Stokes equations."

# Entropy projection

## Two observations

- Since test functions must be polynomial, recovering the rate of change of entropy requires testing with the $L^2$ projection of the entropy variables $v(u)$.

$$\int_{D^k} \frac{\partial \boldsymbol{u}_h}{\partial t} \Pi_N \boldsymbol{v}(\boldsymbol{u}_h) = \int_{D^k} \frac{\partial \boldsymbol{u}_h}{\partial t} \boldsymbol{v}(\boldsymbol{u}_h) = \int_{D^k} \frac{\partial S(\boldsymbol{u}_h)}{\partial t}$$

- Tadmor's entropy conservation condition requires ensuring that the entropy variables and conservative variables are images of each other, which we enforce by using the entropy projection whenever the solution is interpolated.

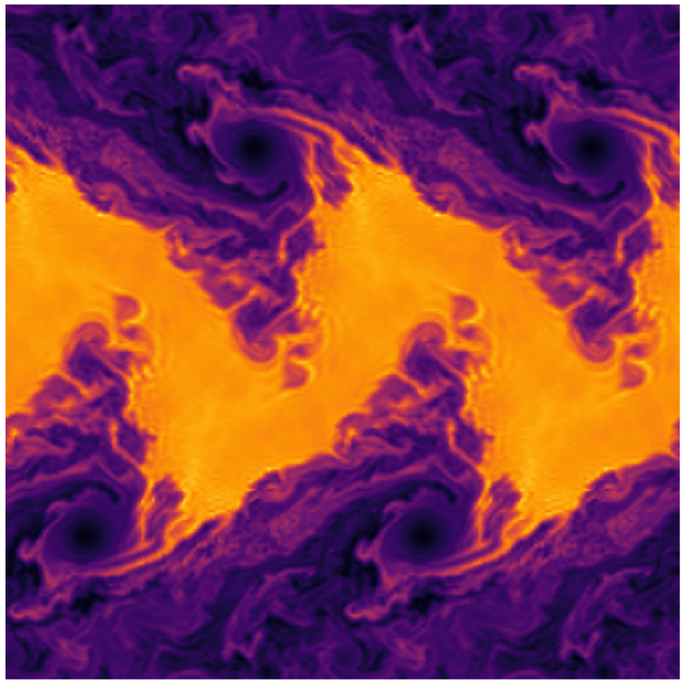
$$\boldsymbol{u}_h \longrightarrow \boldsymbol{u}\left(\Pi_N \boldsymbol{v}(\boldsymbol{u}_h)\right)$$

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

# Modal entropy stable formulations

## Application: new entropy stable methods for under-resolved flows

Observation: for under-resolved flows, nodal (collocation) DG loses positivity, while schemes which utilize the entropy projection remain long-time stable.



Kelvin-Helmholtz

| Solver \ Degree | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| Collocation | 15 | 4.81 | 3.77 | 4.43 | 3.74 | 3.37 | 3.64 |
| Entropy projection | 15 | 15 | 15 | 15 | 15 | 15 | 15 |

$N_{\text{cells}} = 16$

| Solver \ Degree | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| Collocation | 15 | 4.12 | 3.65 | 4.27 | 3.54 | 3.66 | 3.56 |
| Entropy projection | 15 | 15 | 15 | 15 | 15 | 15 | 15 |

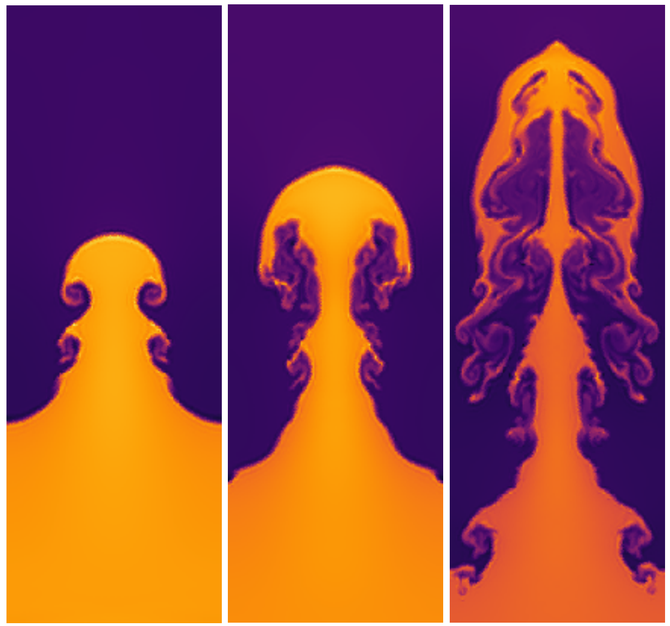
$N_{\text{cells}} = 32$

**Chan**, et al (2022). "On the entropy projection and the robustness of high order entropy stable discontinuous Galerkin schemes for under-resolved flows."

# Modal entropy stable formulations

## Application: new entropy stable methods for under-resolved flows

Observation: for under-resolved flows, nodal (collocation) DG loses positivity, while schemes which utilize the entropy projection remain long-time stable.



Rayleigh-Taylor

| Solver \ Degree | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| Collocation | 3.67 | 3.4 | 3.33 | 3.26 | 3.11 | 3.03 | 3.04 |
| Entropy projection | 15 | 15 | 15 | 15 | 15 | 15 | 15 |

$N_{\text{cells}} = 16$

| Solver \ Degree | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| Collocation | 4.00 | 3.14 | 3.44 | 3.16 | 3.03 | 2.97 | 2.98 |
| Entropy projection | 15 | 15 | 15 | 15 | 15 | 15 | 15 |

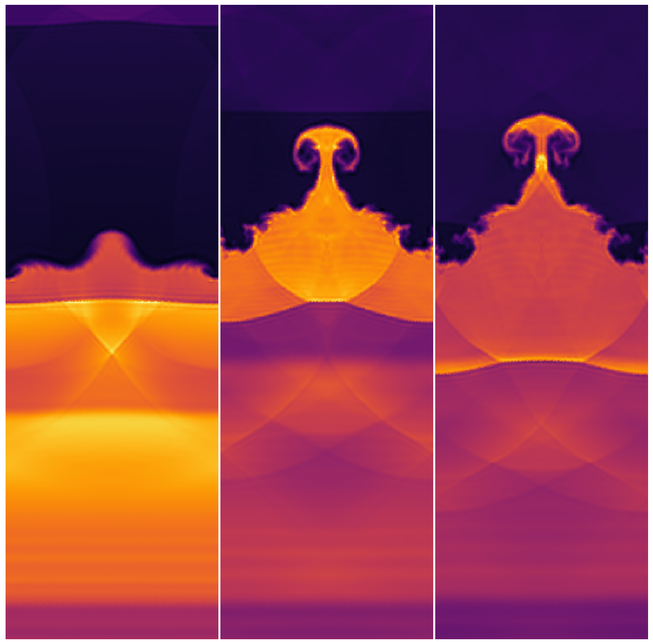
$N_{\text{cells}} = 32$

**Chan**, et al (2022). "On the entropy projection and the robustness of high order entropy stable discontinuous Galerkin schemes for under-resolved flows."

# Modal entropy stable formulations

## Application: new entropy stable methods for under-resolved flows

Observation: for under-resolved flows, nodal (collocation) DG loses positivity, while schemes which utilize the entropy projection remain long-time stable.



Richtmeyer-Meshkov

| Solver \ Degree | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| Collocation | 30 | 30 | 27.96 | 24.94 | 8.851 | 8.853 | 8.85 |
| Entropy projection | 30 | 30 | 30 | 30 | 30 | 30 | 30 |

$N_{\text{cells}} = 16$

| Solver \ Degree | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| Collocation | 30 | 25.52 | 23.34 | 8.759 | 7.808 | 7.014 | 7.01 |
| Entropy projection | 30 | 30 | 30 | 30 | 30 | 30 | 30 |

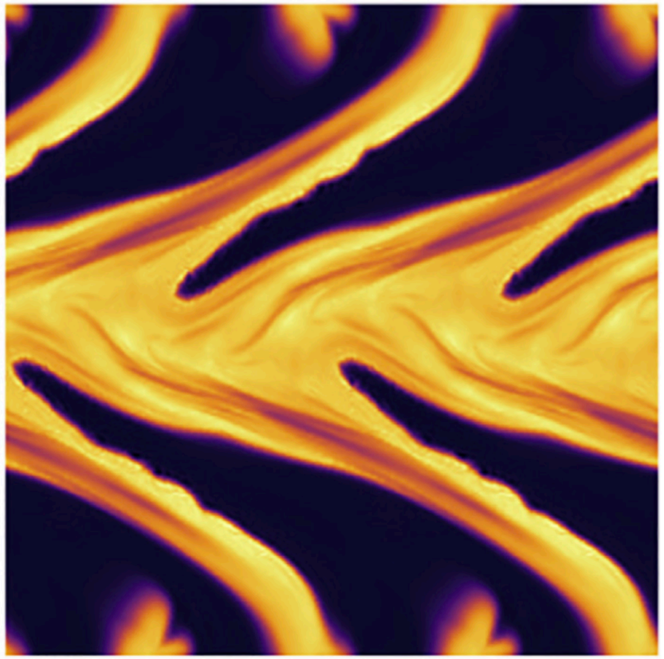
$N_{\text{cells}} = 32$

**Chan**, et al (2022). "On the entropy projection and the robustness of high order entropy stable discontinuous Galerkin schemes for under-resolved flows."

# Modal entropy stable formulations

## Application: new entropy stable methods for under-resolved flows

Observation: for under-resolved flows, nodal (collocation) DG loses positivity, while schemes which utilize the entropy projection remain long-time stable.



Magnetized Kelvin-Helmholtz

| Solver \ Degree | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| Collocation | 15 | 15 | 11.50 | 10.99 | 10.32 | 10.23 | 10.27 |
| Entropy projection | 15 | 15 | 15 | 15 | 15 | 15 | 15 |

$N_{\text{cells}} = 16$

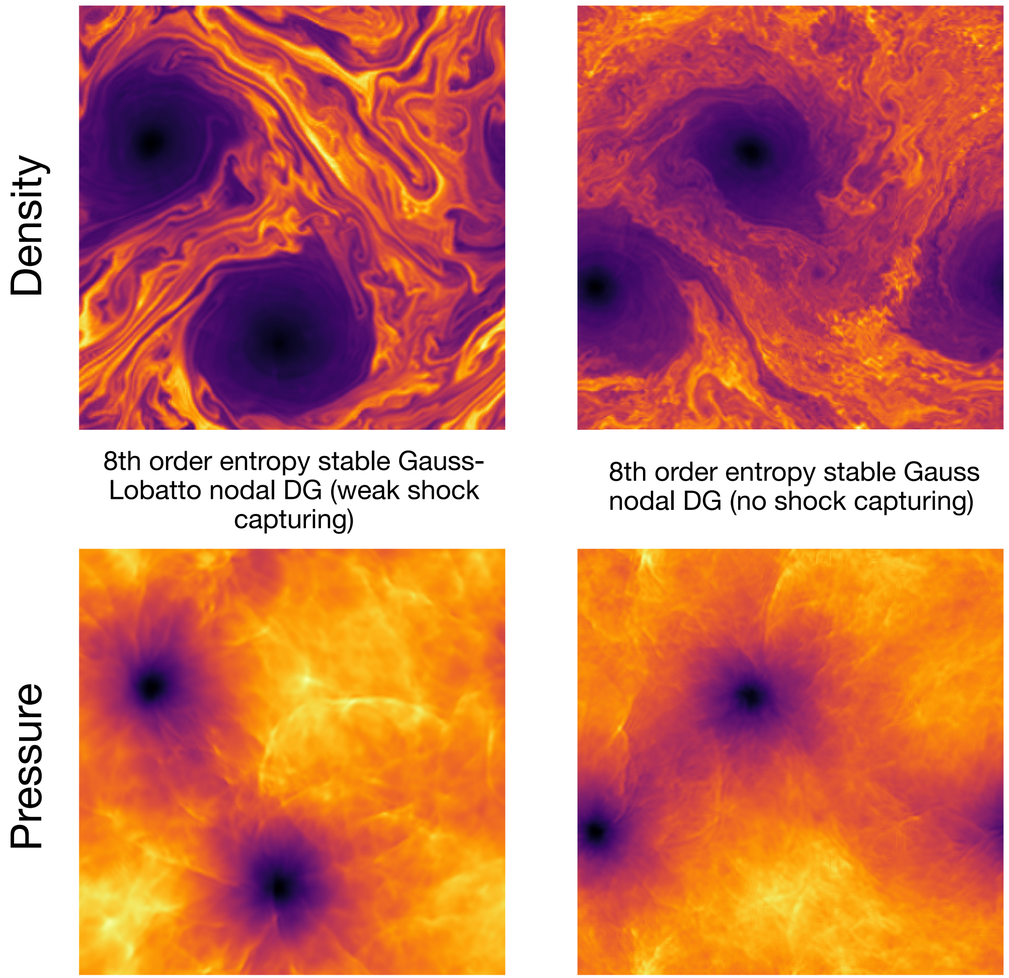
| Solver \ Degree | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| Collocation | 15 | 11.64 | 11.05 | 11.11 | 11.48 | 10.17 | 10.92 |
| Entropy projection | 15 | 15 | 15 | 15 | 15 | 15 | 15 |

$N_{\text{cells}} = 32$

**Chan**, et al (2022). "On the entropy projection and the robustness of high order entropy stable discontinuous Galerkin schemes for under-resolved flows."

# Modal entropy stable formulations

## Application: new entropy stable methods for under-resolved flows

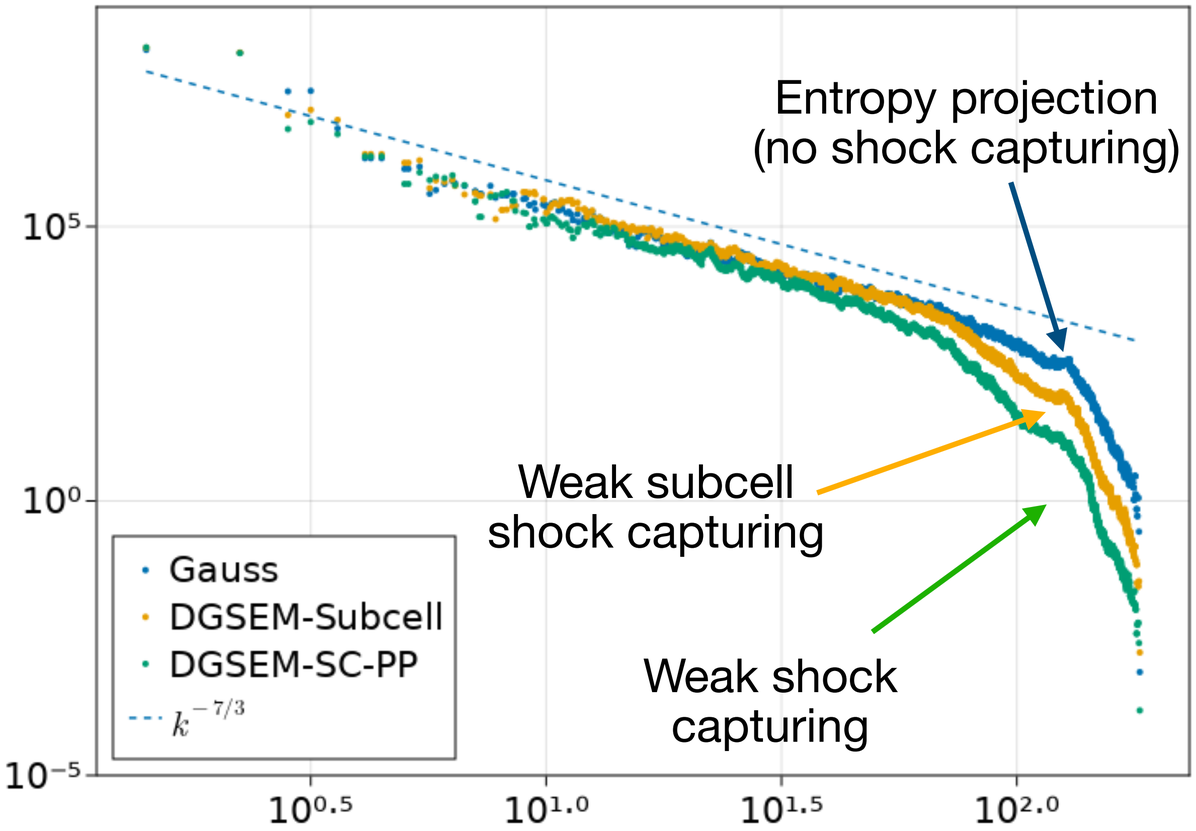



- New entropy stable collocation-like DG methods based on Gauss quadrature.
  - More accurate than nodal entropy stable DG methods based on Gauss-Lobatto quadrature.
- The use of the entropy projection appears to improve robustness for under-resolved flows

**Chan**, Del Rey Fernandez, Carpenter (2019). “Efficient entropy stable Gauss collocation methods”.
**Chan**, et al (2022). "On the entropy projection and the robustness of high order entropy stable discontinuous Galerkin schemes for under-resolved flows."

# Modal entropy stable formulations

## Application: new entropy stable methods for under-resolved flows




$N = 7$ mesh of $32^2$ elements

- New entropy stable collocation-like DG methods based on Gauss quadrature.
  - More accurate than nodal entropy stable DG methods based on Gauss-Lobatto quadrature.
- The use of the entropy projection appears to improve robustness for under-resolved flows


**Chan**, Del Rey Fernandez, Carpenter (2019). “Efficient entropy stable Gauss collocation methods”.

**Chan**, et al (2022). "On the entropy projection and the robustness of high order entropy stable discontinuous Galerkin schemes for under-resolved flows."

# Hybridized summation-by-parts operators

- *Hybridized* SBP operators involve *both* volume/face nodes.

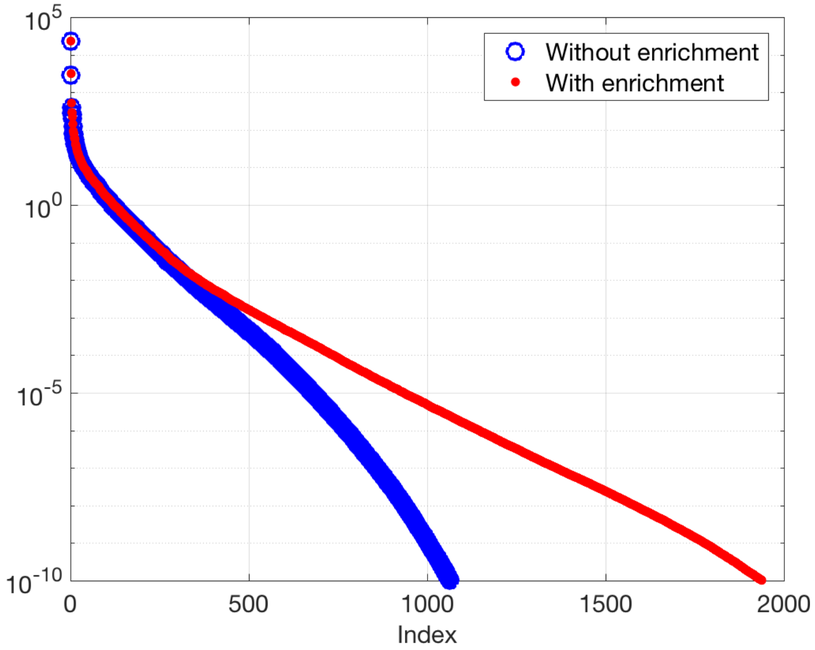
$$\mathbf{Q}_h = \frac{1}{2} \begin{bmatrix} \mathbf{Q} - \mathbf{Q}^T & \mathbf{E}^T \mathbf{B} \\ -\mathbf{B}\mathbf{E} & \mathbf{B} \end{bmatrix}$$

- Approximating derivatives using hybridized SBP operators:

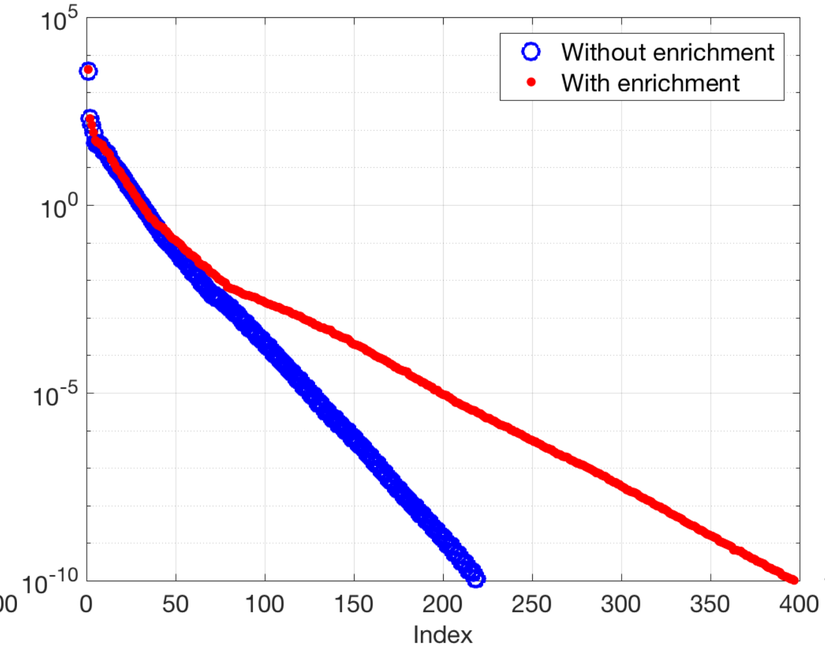
$$\frac{\partial g}{\partial x} \approx \mathbf{M}^{-1} \begin{bmatrix} \mathbf{V}_q \\ \mathbf{V}_f \end{bmatrix}^T \mathbf{Q}_h \begin{bmatrix} g(\mathbf{x}_q) \\ g(\mathbf{x}_f) \end{bmatrix}$$

- Can be interpreted as adding correction terms of the form $\mathbf{E}\boldsymbol{f}(\mathbf{u}) - \boldsymbol{f}(\mathbf{E}\mathbf{u})$

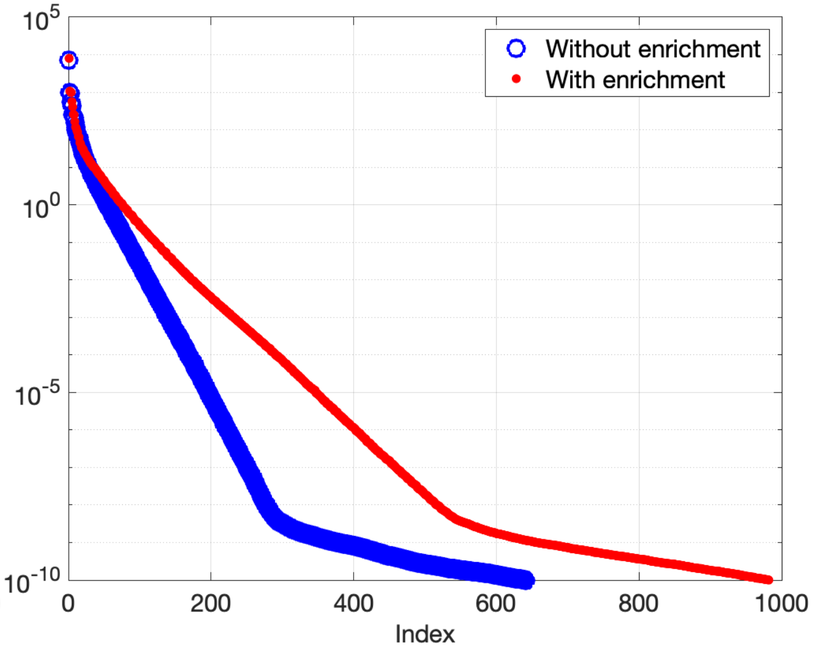
# Singular values with entropy variable enrichment




Kelvin-Helmholtz instability



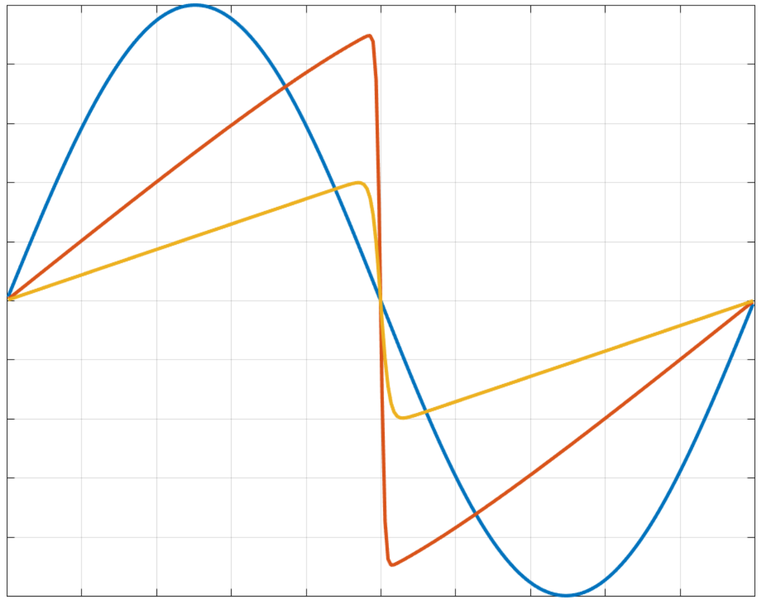

Gaussian pulse



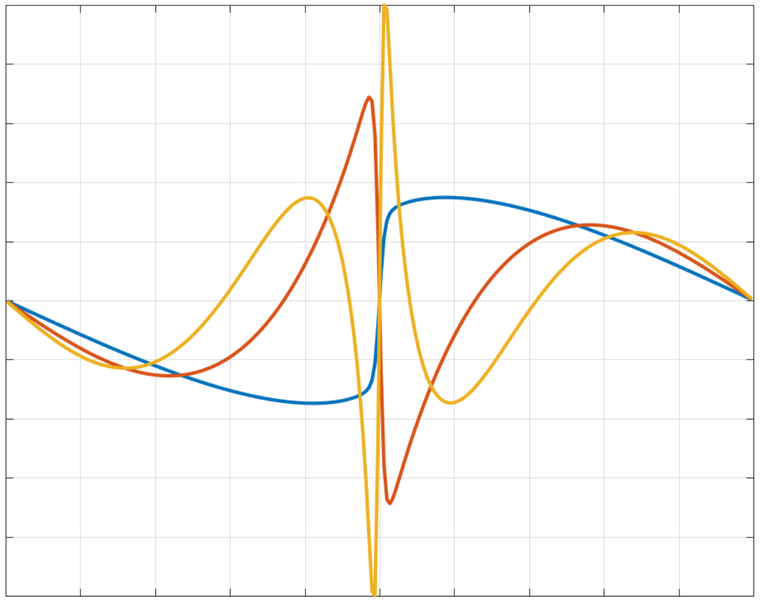

Riemann problem

The solution snapshot singular values decay more slowly when using entropy variable enrichment for strongly transport-dominated or shock solutions.

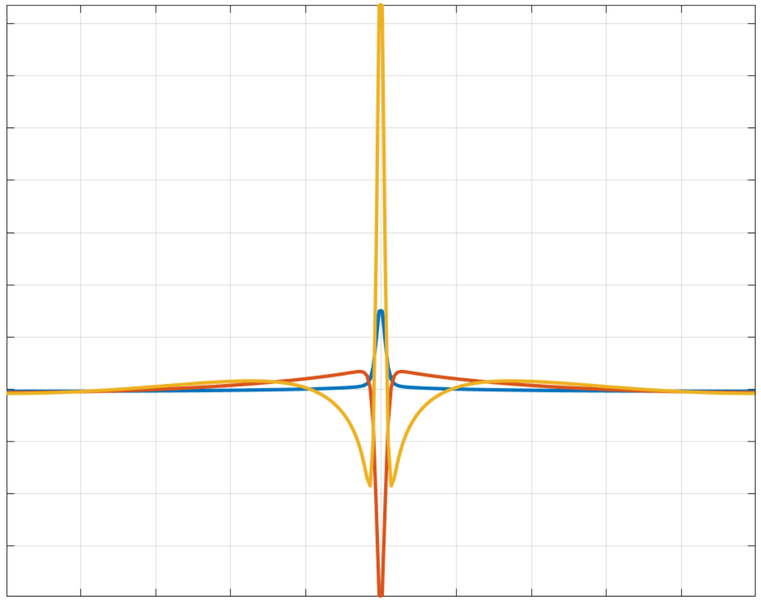
# Motivation for the “test basis”



Burgers’ equation snapshots



POD modes



Mode derivatives QV

- For shock solutions, the POD basis can poorly sample the derivative matrix $\mathbf{V}^T\mathbf{QV} \approx \mathbf{0}$
- Solution: augment the sampling basis with $\mathbf{QV}$

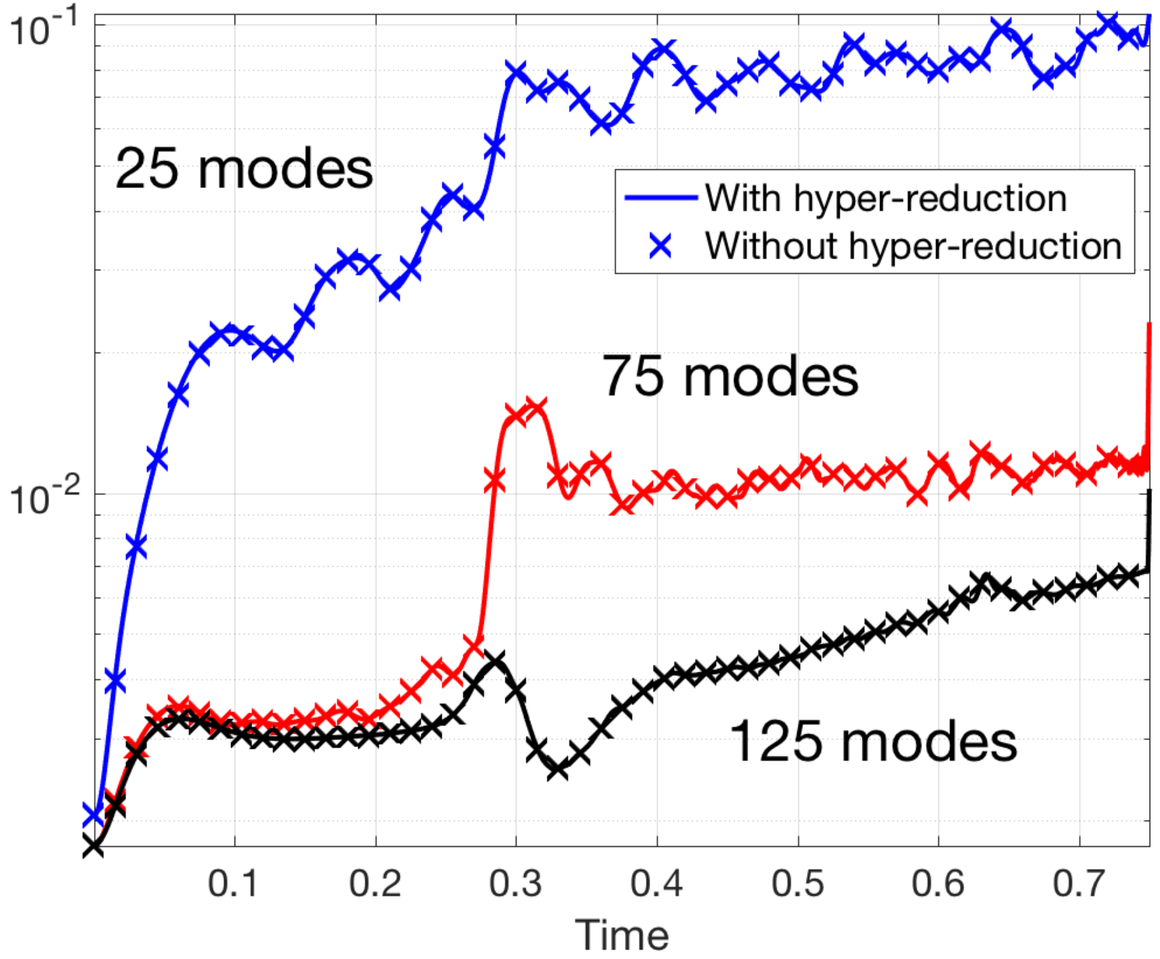
# Effectiveness of hyper-reduction

- Hyper-reduction targets accuracy for the mass matrix

$$\mathbf{V}^T\mathbf{V} \approx \mathbf{V}(\mathcal{I},:)^T\mathbf{W}\mathbf{V}(\mathcal{I},:)$$

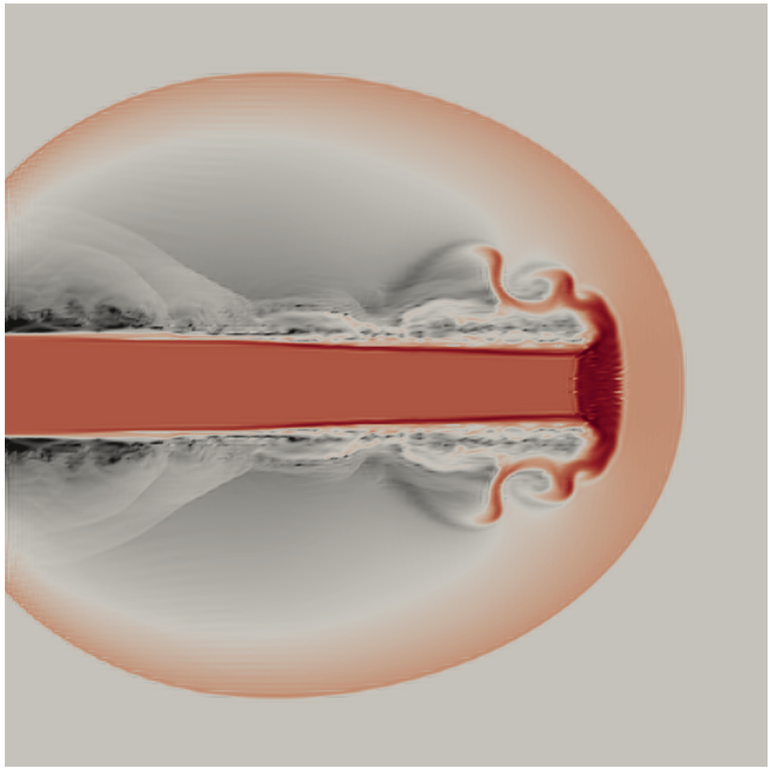
- Add “stabilizing points” in case the test mass matrix is near-singular under hyper-reduction

$$\mathbf{V}_t(\mathcal{I},:)^T\mathbf{W}\mathbf{V}_t(\mathcal{I},:)$$

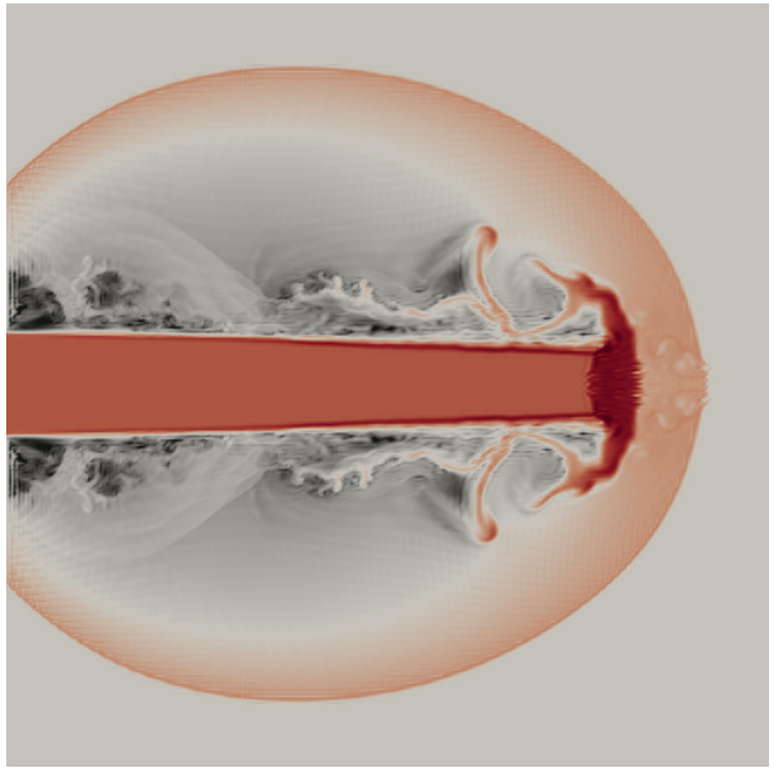



$L^2$ error over time

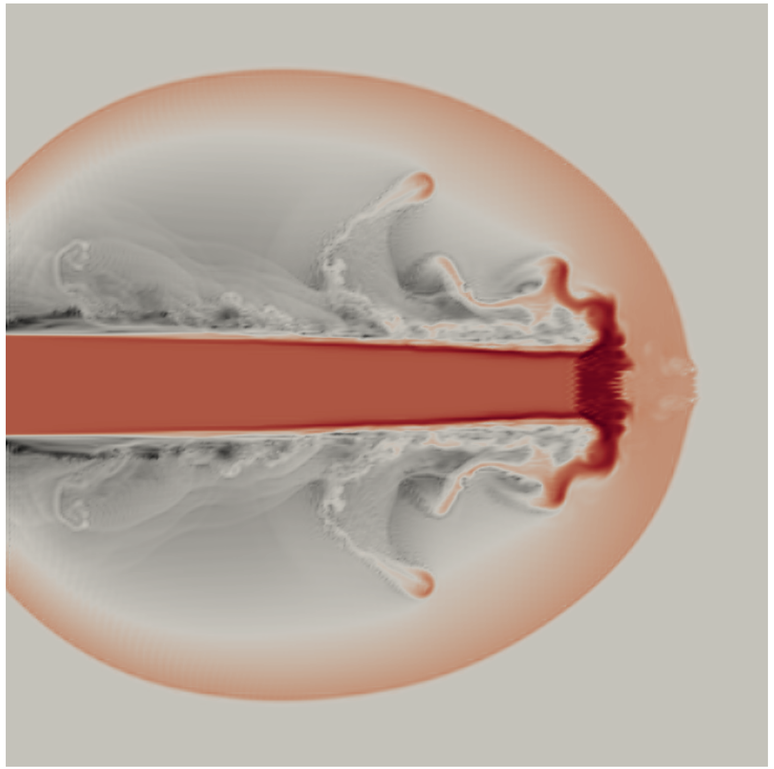
# Knapsack limiting: Mach 2000 astrophysical jet



ESDG with element-wise limiting



ESDG with knapsack limiting



DGSEM with knapsack limiting

Degree $N = 3$, $150 \times 150$ elements with relaxed positivity bounds and cell entropy inequality.

**Lin**, **Chan** (2024). *High order entropy stable discontinuous Galerkin spectral element methods through subcell limiting*.