

Entropy stable discontinuous Galerkin-Fourier Methods

Yimin Lin

Master Thesis Defense, September 2020

Advisor: Jesse Chan

Department of Computational and Applied Mathematics, Rice University

High order finite element methods for PDEs

- Physical phenomena governed by PDE: aerospace engineering, biological science

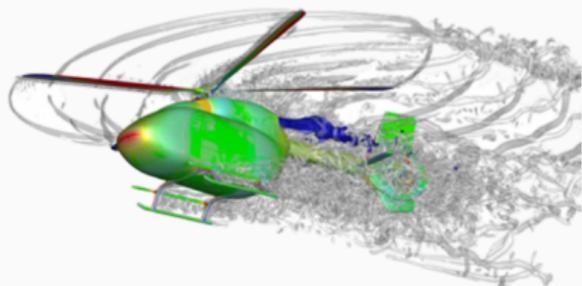


Figure 1: Vortex structures from a helicopter simulation

High order finite element methods for PDEs

- Physical phenomena governed by PDE: aerospace engineering, biological science
- High accuracy computational fluid dynamics on complex geometries

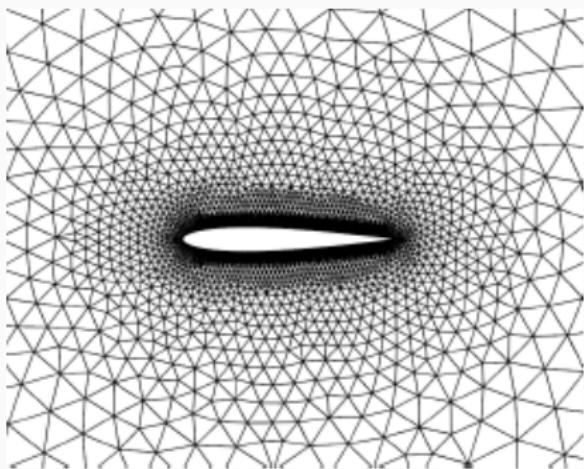


Figure 1: Unstructured mesh for NACA 0012 foil

High order finite element methods for PDEs

- Physical phenomena governed by PDE: aerospace engineering, biological science
- High accuracy computational fluid dynamics on complex geometries
- More accurate per degrees of freedom than low order methods (for smooth solutions)

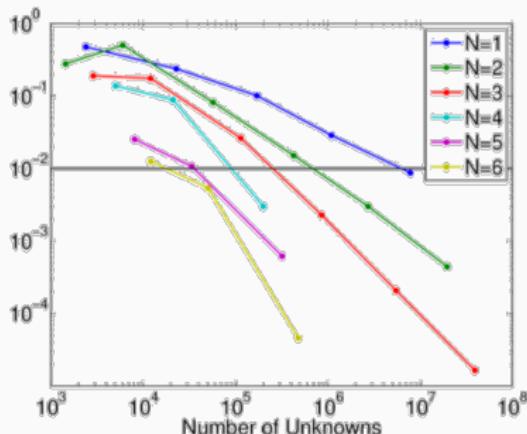


Figure 1: high order methods achieve better accuracy more efficiently

High order discontinuous Galerkin methods for PDEs

- Finite element methods without enforcing strict continuity across interfaces.

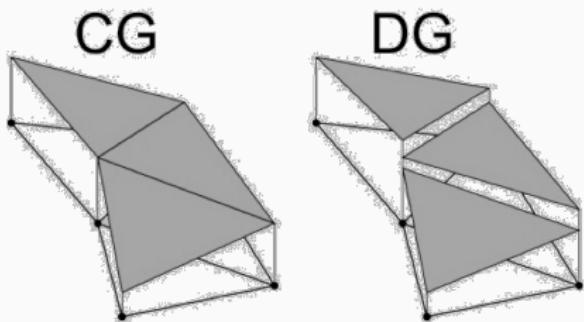


Figure 2: Approximant in finite element vs DG

High order discontinuous Galerkin methods for PDEs

- Finite element methods without enforcing strict continuity across interfaces.
- Shares most of the advantages of FEM: complex geometries, high order accuracy

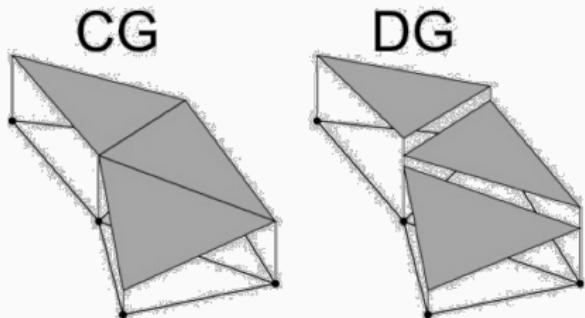


Figure 2: Approximants in finite element vs DG

High order discontinuous Galerkin methods for PDEs

- Finite element methods without enforcing strict continuity across interfaces.
- Shares most of the advantages of FEM: complex geometries, high order accuracy
- Natural parallelizable structure due to local operators

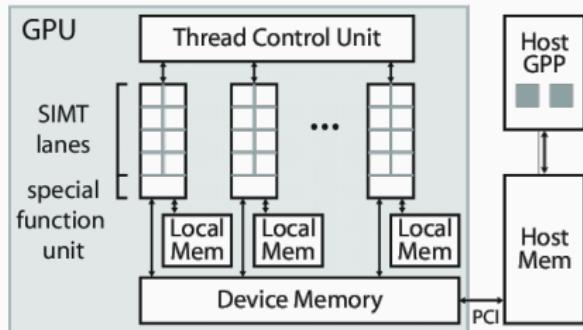


Figure 2: Nvidia GPU internal architecture

High order discontinuous Galerkin methods for PDEs

- Finite element methods without enforcing strict continuity across interfaces.
- Shares most of the advantages of FEM: complex geometries, high order accuracy
- Natural parallelizable structure due to local operators
- Suitable for convection dominated problems (with appropriate numerical fluxes)

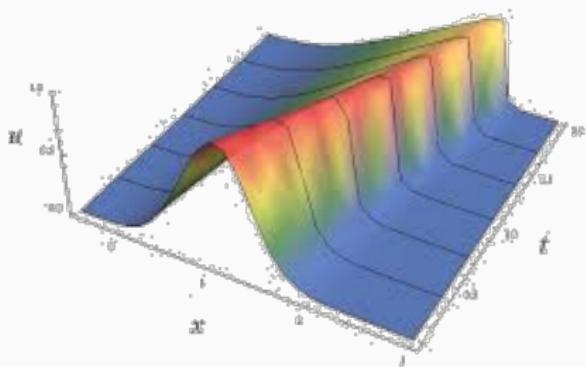


Figure 2: Solution of inviscid Burger's equation

Energy stability for PDEs

- Linear advection equations on the periodic domain $[-1, 1]$

$$\frac{\partial u}{\partial t} = \frac{\partial u}{\partial x}, \quad u(-1, t) = u(1, t)$$

Energy stability for PDEs

- Linear advection equations on the periodic domain $[-1, 1]$

$$\frac{\partial u}{\partial t} = \frac{\partial u}{\partial x}, \quad u(-1, t) = u(1, t)$$

- Then we could derive energy stability

$$\int_{-1}^1 u \frac{\partial u}{\partial t} = \int_{-1}^1 u \frac{\partial u}{\partial x}$$

Test by u

Energy stability for PDEs

- Linear advection equations on the periodic domain $[-1, 1]$

$$\frac{\partial u}{\partial t} = \frac{\partial u}{\partial x}, \quad u(-1, t) = u(1, t)$$

- Then we could derive energy stability

$$\int_{-1}^1 u \frac{\partial u}{\partial t} = \int_{-1}^1 u \frac{\partial u}{\partial x} \quad \text{Test by } u$$

$$\frac{\partial}{\partial t} \|u\|^2 = \frac{\partial}{\partial t} \int_{-1}^1 u^2 = u_R^2 - u_L^2 = 0 \quad \text{Integration by parts}$$

Energy stability for PDEs

- Linear advection equations on the periodic domain $[-1, 1]$

$$\frac{\partial u}{\partial t} = \frac{\partial u}{\partial x}, \quad u(-1, t) = u(1, t)$$

- Then we could derive energy stability

$$\int_{-1}^1 u \frac{\partial u}{\partial t} = \int_{-1}^1 u \frac{\partial u}{\partial x}$$

Test by u

$$\frac{\partial}{\partial t} \underbrace{\|u\|^2}_{\text{energy}} = \frac{\partial}{\partial t} \int_{-1}^1 u^2 = u_R^2 - u_L^2 = 0$$

Integration by parts

Energy stability for PDEs

- Linear advection equations on the periodic domain $[-1, 1]$

$$\frac{\partial u}{\partial t} = \frac{\partial u}{\partial x}, \quad u(-1, t) = u(1, t)$$

- Then we could derive energy stability

$$\int_{-1}^1 u \frac{\partial u}{\partial t} = \int_{-1}^1 u \frac{\partial u}{\partial x}$$

Test by u

$$\frac{\partial}{\partial t} \underbrace{\|u\|^2}_{\text{energy}} = \frac{\partial}{\partial t} \int_{-1}^1 u^2 = u_R^2 - u_L^2 = 0$$

Integration by parts

- Energy stability doesn't work in nonlinear conservation laws

$$\frac{\partial u}{\partial t} + \sum_{i=1}^d \frac{\partial f_i(u)}{\partial x_i} = 0, \quad u, f_i : \mathbb{R}^n \rightarrow \mathbb{R}^n$$

Mathematical entropy

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

Mathematical entropy

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

$$\int_{\Omega} v^T \frac{\partial u}{\partial t} + \sum_{i=1}^d \int_{\Omega} v^T \frac{\partial f_i(u)}{\partial x_i} = 0 \quad \text{Test by } v$$

Mathematical entropy

- With convex entropy η , entropy variable $\mathbf{v} = \frac{\partial \eta(\mathbf{u})}{\partial \mathbf{u}}$ and entropy potential ψ_i . We can derive an [entropy equality](#)

$$\int_{\Omega} \mathbf{v}^T \frac{\partial \mathbf{u}}{\partial t} + \sum_{i=1}^d \int_{\Omega} \mathbf{v}^T \frac{\partial \mathbf{f}_i(\mathbf{u})}{\partial x_i} = 0 \quad \text{Test by } \mathbf{v}$$

$$\int_{\Omega} \frac{\partial \eta(\mathbf{u})}{\partial t} + \sum_{i=1}^d \int_{\partial\Omega} n_i (\mathbf{v}^T \mathbf{f}_i(\mathbf{u}) - \psi_i(\mathbf{u})) = 0 \quad \text{Integration by parts}$$

Mathematical entropy

- With convex entropy η , entropy variable $\mathbf{v} = \frac{\partial \eta(\mathbf{u})}{\partial \mathbf{u}}$ and entropy potential ψ_i . We can derive an [entropy equality](#)

$$\int_{\Omega} \mathbf{v}^T \frac{\partial \mathbf{u}}{\partial t} + \sum_{i=1}^d \int_{\Omega} \mathbf{v}^T \frac{\partial \mathbf{f}_i(\mathbf{u})}{\partial x_i} = 0 \quad \text{Test by } \mathbf{v}$$

$$\int_{\Omega} \frac{\partial \eta(\mathbf{u})}{\partial t} + \sum_{i=1}^d \int_{\partial \Omega} n_i (\mathbf{v}^T \mathbf{f}_i(\mathbf{u}) - \psi_i(\mathbf{u})) = 0 \quad \text{Integration by parts}$$

$$\int_{\Omega} \frac{\partial \eta(\mathbf{u})}{\partial t} = 0 \quad \text{Periodic domain}$$

Mathematical entropy

- With convex entropy η , entropy variable $\mathbf{v} = \frac{\partial \eta(\mathbf{u})}{\partial \mathbf{u}}$ and entropy potential ψ_i . We can derive an [entropy equality](#)

$$\int_{\Omega} \mathbf{v}^T \frac{\partial \mathbf{u}}{\partial t} + \sum_{i=1}^d \int_{\Omega} \mathbf{v}^T \frac{\partial \mathbf{f}_i(\mathbf{u})}{\partial x_i} = 0 \quad \text{Test by } \mathbf{v}$$

$$\int_{\Omega} \frac{\partial \eta(\mathbf{u})}{\partial t} + \sum_{i=1}^d \int_{\partial \Omega} n_i (\mathbf{v}^T \mathbf{f}_i(\mathbf{u}) - \psi_i(\mathbf{u})) = 0 \quad \text{Integration by parts}$$

$$\int_{\Omega} \frac{\partial \eta(\mathbf{u})}{\partial t} = 0 \quad \text{Periodic domain}$$

- Loss of chain rule at discrete level (discrete effects, inexact quadrature)
⇒ Loss of entropy stability

Typical stabilization procedures

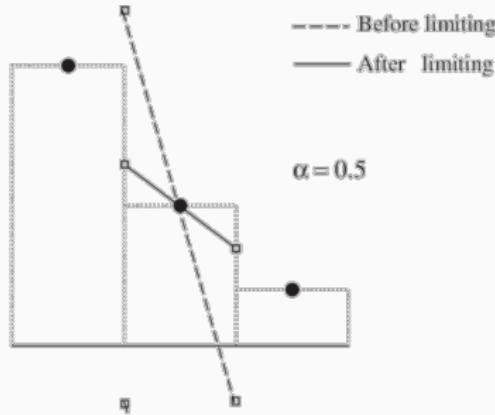


Figure 3: Slope limiting procedure

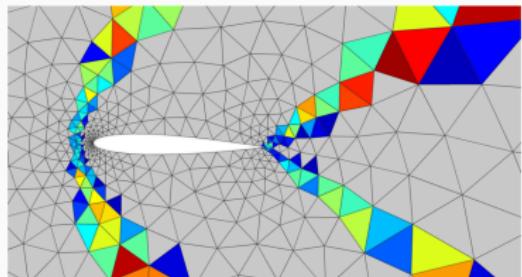


Figure 4: Sub-cell viscosity¹

- Slope limiting, filtering and artificial viscosity: loses high order accuracy, require heuristic tuning

¹Persson and Peraire, "Sub-cell shock capturing for discontinuous Galerkin methods".

Typical stabilization procedures

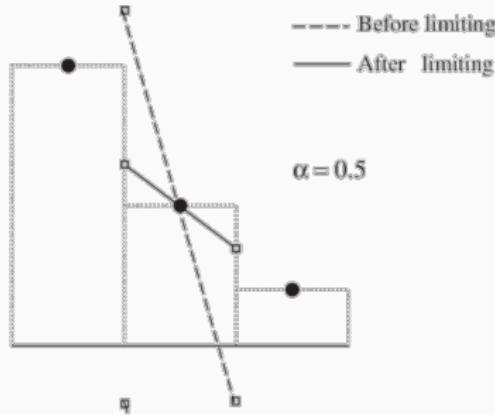


Figure 3: Slope limiting procedure

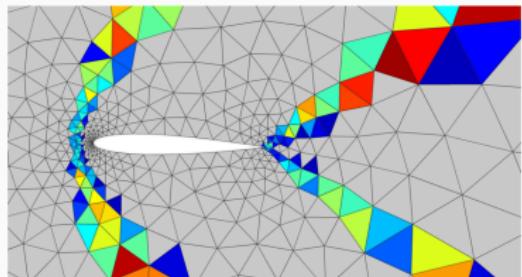


Figure 4: Sub-cell viscosity¹

- Slope limiting, filtering and artificial viscosity: loses high order accuracy, require heuristic tuning
- Limited theoretical justification

¹Persson and Peraire, "Sub-cell shock capturing for discontinuous Galerkin methods".

Robust DG formulation: entropy stable nodal DG

- Collocation nodal DG: matrix vector product

$$\mathbf{M} \frac{d\mathbf{u}}{dt} + \mathbf{Q}\mathbf{f}(\mathbf{u}) = 0$$

- Entropy stable nodal DG: Hadamard product

$$\mathbf{M} \frac{d\mathbf{u}}{dt} + 2(\mathbf{Q} \circ \mathbf{F}_S) \mathbf{1} = 0$$

Robust DG formulation: entropy stable nodal DG

- Collocation nodal DG: matrix vector product

$$\mathbf{M} \frac{d\mathbf{u}}{dt} + \mathbf{Q} \mathbf{f}(\mathbf{u}) = 0$$

- Entropy stable nodal DG: Hadamard product

$$\mathbf{M} \frac{d\mathbf{u}}{dt} + 2(\mathbf{Q} \circ \mathbf{F}_S) \mathbf{1} = 0$$
$$(\mathbf{F}_S)_{ij} = f_S(u_i, u_j)$$

Robust DG formulation: entropy stable nodal DG

- Collocation nodal DG: matrix vector product

$$\mathbf{M} \frac{d\mathbf{u}}{dt} + \mathbf{Q} \mathbf{f}(\mathbf{u}) = 0$$

- Entropy stable nodal DG: Hadamard product

$$\mathbf{M} \frac{d\mathbf{u}}{dt} + 2(\mathbf{Q} \circ \mathbf{F}_S) \mathbf{1} = 0$$
$$(\mathbf{F}_S)_{ij} = f_S(u_i, u_j)$$

- Entropy stable DG is computationally more expensive

Fourier pseudo-spectral method

- Approximate solutions with Fourier basis

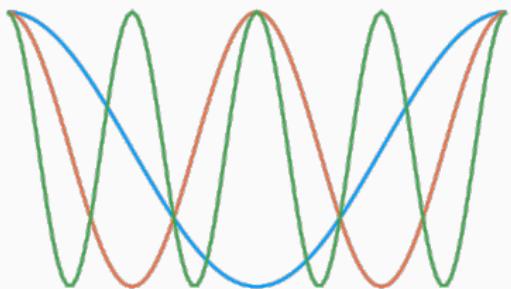


Figure 5: Example of Fourier basis

Fourier pseudo-spectral method

- Approximate solutions with Fourier basis
- Spectral accuracy for smooth solutions

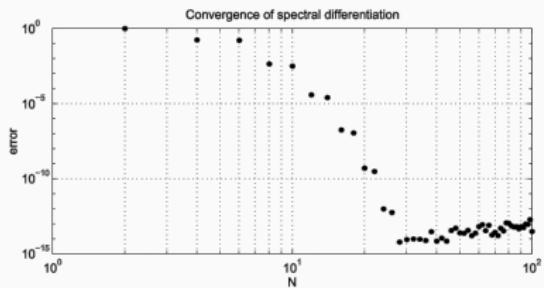


Figure 5: Accuracy of spectral differentiation²

²Trefethen, *Spectral methods in MATLAB*.

Fourier pseudo-spectral method

- Approximate solutions with Fourier basis
- Spectral accuracy for smooth solutions
- Fast Fourier transform

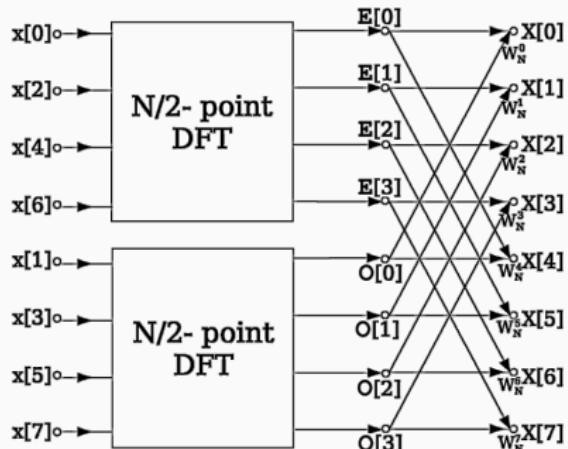


Figure 5: Fast Fourier transform illustration

Fourier pseudo-spectral method

- Approximate solutions with Fourier basis
- Spectral accuracy for smooth solutions
- Fast Fourier transform
- Stabilization via de-aliasing

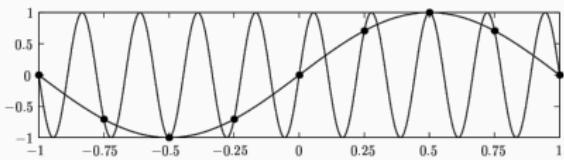


Figure 5: An example of aliasing

Discontinuous Galerkin-Fourier methods

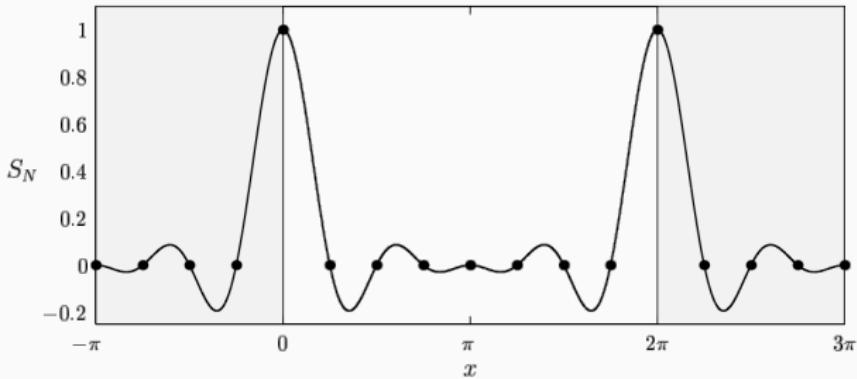


Figure 6: Nodal Fourier basis is periodic²

- Only works on periodic domains

²Trefethen, *Spectral methods in MATLAB*.

Discontinuous Galerkin-Fourier methods

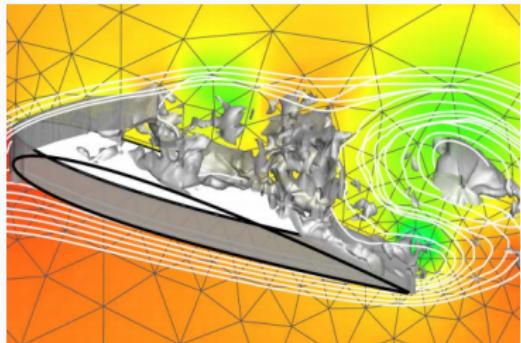


Figure 6: Flow around airfoil

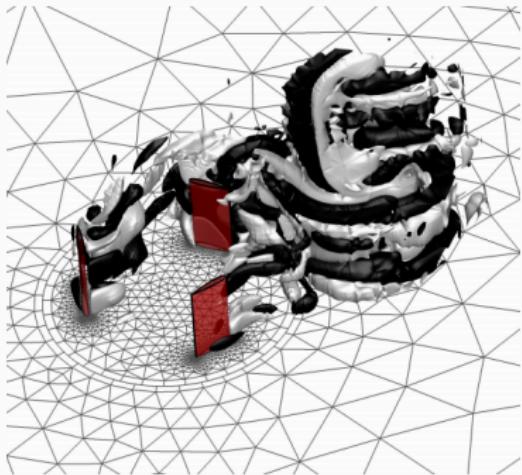


Figure 7: Flow around wind turbine²

- Only works on periodic domains
- Applications: simulation of quasi-2D flows

²Ferrer and Willden, "A high order discontinuous Galerkin-Fourier incompressible 3D Navier-Stokes solver with rotating sliding meshes".

Geometric assumptions

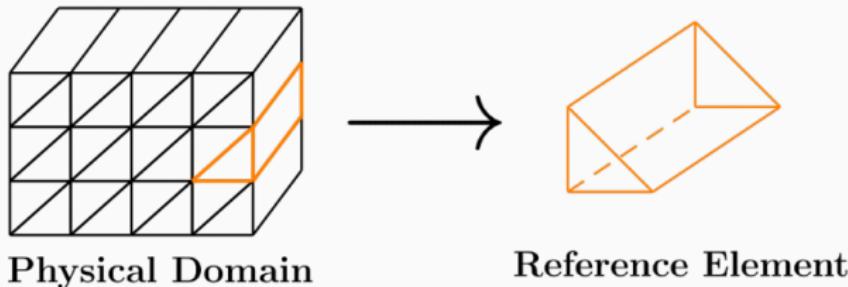


Figure 8: Example physical domain and reference element

- The geometry is homogeneous and periodic in one direction
- Reference element in 3D: wedge

Approximation space

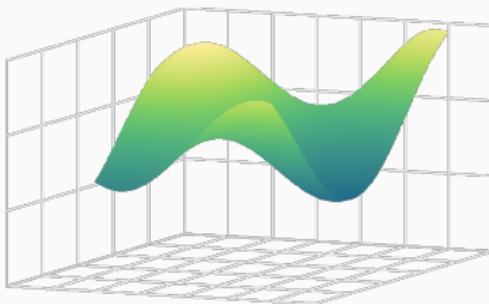


Figure 9: Polynomial basis

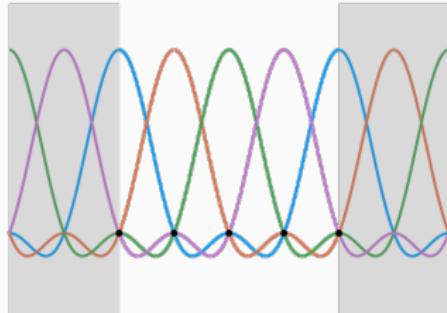


Figure 10: Nodal Fourier basis

- Triangles: polynomial of degree N^P
 z -direction: nodal Fourier basis

$$S(\hat{z}) = \frac{\sin\left(N^F \frac{\hat{z}}{2}\right)}{N^F \tan\left(\frac{\hat{z}}{2}\right)}, \quad S_i(\hat{z}) = S(\hat{z} - \hat{z}_i)$$

Approximation space

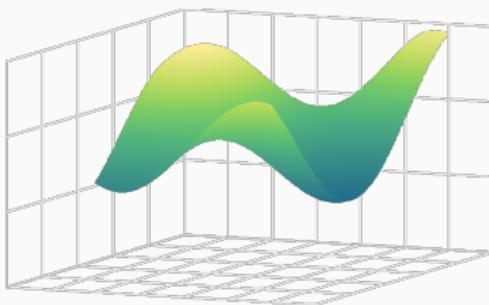


Figure 9: Polynomial basis

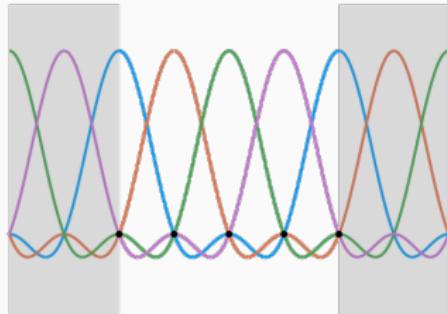


Figure 10: Nodal Fourier basis

- Approximation space: tensor product of polynomial and sinc basis

$$\begin{aligned} V_h(\widehat{D}) &= P^{N^P}(\widehat{D}^P) \otimes F^{N^F}(\widehat{D}^F) \\ &= \{\varphi_i(\widehat{x}, \widehat{z}) = p_{k_1, \dots, k_{d-1}}(\widehat{x}) S_i(\widehat{z}), \quad \widehat{x} \in \widehat{D}^P, \widehat{z} \in \widehat{D}^F\} \end{aligned}$$

Quadrature rules

- Quadrature on triangles $(\hat{\mathbf{x}}_i, w_i)$, $(\hat{\mathbf{x}}_i^f, w_i^f)$: degree $2N^P$ volume and surface quadratures

$$\left(\frac{\partial u}{\partial x_n}, v \right)_{\widehat{D}^P} = \langle u, v \rangle_{\partial \widehat{D}^P} - \left(u, \frac{\partial v}{\partial x_n} \right)_{\widehat{D}^P}$$

Quadrature rules

- Quadrature on triangles $(\hat{\mathbf{x}}_i, w_i)$, $(\hat{\mathbf{x}}_i^f, w_i^f)$: degree $2N^P$ volume and surface quadratures
- Quadrature rule on the spanwise direction, $h = \frac{2\pi}{N_p^F}$

$$\int_0^{2\pi} f(z) dz \approx h \sum_{i=1}^{N_p^F} f(\hat{z}_i), \quad \hat{z}_i = hi, \quad i = 1, \dots, N_p^F$$

Quadrature rules

- Quadrature on triangles $(\hat{\mathbf{x}}_i, w_i)$, $(\hat{\mathbf{x}}_i^f, w_i^f)$: degree $2N^P$ volume and surface quadratures
- Quadrature rule on the spanwise direction, $h = \frac{2\pi}{N_p^F}$
- Combine through tensor product

$$\{(\hat{\mathbf{x}}_i, \hat{z}_j), h w_i\}_{\substack{i=1 \dots N_q^P \\ j=1 \dots N^F}} \quad \{(\hat{\mathbf{x}}_i^f, \hat{z}_j), h w_i^f\}_{\substack{i=1 \dots N_{f,q}^P \\ j=1 \dots N^F}}$$

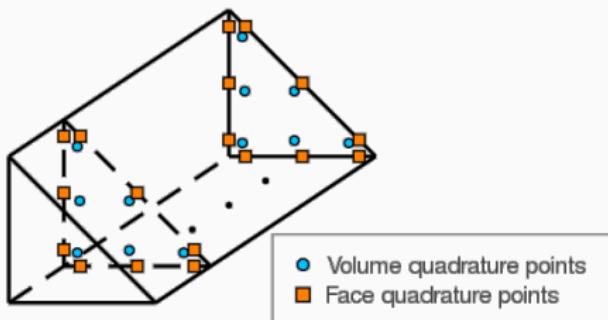
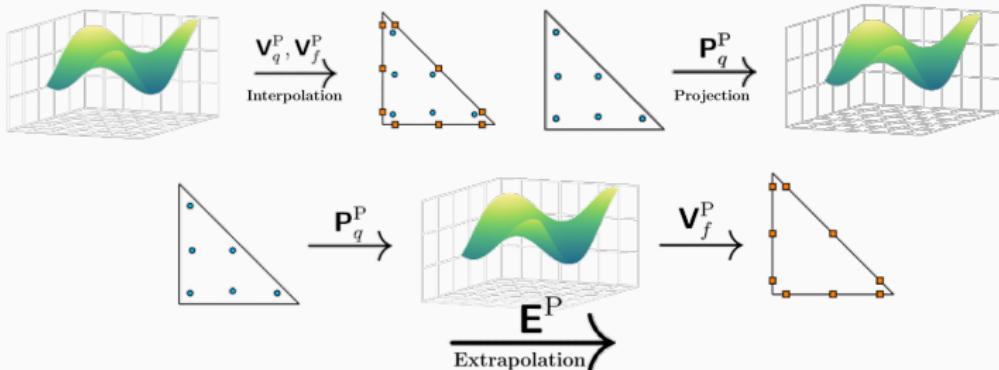


Figure 11: Reference element with quadrature nodes

Quadrature based operators on the reference triangle



- Mass matrix, boundary integration, differentiation

$$M^P \approx \int_{\widehat{D}^P} p_i(\widehat{x}) p_j(\widehat{x}) d\widehat{x}, \quad B_n^P \approx \int_{\partial \widehat{D}^P} u v \widehat{n}_n, \quad D_{q,n}^P \approx \frac{\partial}{\partial x} (\Pi_{N^P})$$

- Chan's hybridized SBP operators, $n = 1, 2$

$$Q_{h,n}^P = \begin{bmatrix} Q_{q,n}^P - \frac{1}{2} (E^P)^T B_n^P E^P & \frac{1}{2} (E^P)^T B_n^P \\ -\frac{1}{2} B_n^P E^P & \frac{1}{2} B_n^P \end{bmatrix}$$

Quadrature based operators on the reference wedge

- By tensor product structure of approximation space, extend modal operators \mathbf{A} :

$$\mathbf{A} = \mathbf{I}_{N^F} \otimes \mathbf{A}^P$$

Quadrature based operators on the reference wedge

- By tensor product structure of approximation space, extend modal operators \mathbf{A} :

$$\mathbf{A} = \mathbf{I}_{N^F} \otimes \mathbf{A}^P$$

- By the quadrature rule on the z direction, extend quadrature-based operators \mathbf{T}

$$\int_{\widehat{D}} \varphi(\widehat{\mathbf{x}}, \widehat{z}) = \int_{\widehat{D}^P} p(\widehat{\mathbf{x}}) \int_{\widehat{D}^F} S(\widehat{z}) \implies \mathbf{T} = h \mathbf{I}_{N^F} \otimes \mathbf{T}^P$$

Quadrature based operators on the reference wedge

- By tensor product structure of approximation space, extend modal operators \mathbf{A} :

$$\mathbf{A} = \mathbf{I}_{N^F} \otimes \mathbf{A}^P$$

- By the quadrature rule on the z direction, extend quadrature-based operators \mathbf{T}

$$\int_{\widehat{D}} \varphi(\widehat{\mathbf{x}}, \widehat{z}) = \int_{\widehat{D}^P} p(\widehat{\mathbf{x}}) \int_{\widehat{D}^F} S(\widehat{z}) \implies \mathbf{T} = h \mathbf{I}_{N^F} \otimes \mathbf{T}^P$$

- Hybridized operators on x, y directions retain SBP property

$$\mathbf{Q}_{h,n} + (\mathbf{Q}_{h,n})^T = h \mathbf{I}_{N^F} \otimes \begin{bmatrix} 0 & \\ & \mathbf{B}_n^P \end{bmatrix}, \quad n = 1, 2$$

$$\approx \int_{\widehat{D}} u \frac{\partial v}{\partial x_n} + \int_{\widehat{D}} v \frac{\partial u}{\partial x_n} = \int_{\partial \widehat{D}} u v \widehat{\mathbf{n}}_n$$

Quadrature based operators on the reference wedge

- Spectral differentiation matrix D^F on z direction

$$D^F = \begin{bmatrix} 0 & & & -\frac{1}{2} \cot \frac{h}{2} \\ -\frac{1}{2} \cot \frac{h}{2} & \ddots & \ddots & \frac{1}{2} \cot h \\ \frac{1}{2} \cot h & \ddots & \ddots & -\frac{1}{2} \cot \frac{3h}{2} \\ \vdots & \ddots & \ddots & \vdots \\ \frac{1}{2} \cot \frac{h}{2} & & & 0 \end{bmatrix}, \quad D^F + (D^F)^T = 0, \quad D^F \mathbf{1} = 0$$

Quadrature based operators on the reference wedge

- Spectral differentiation matrix D^F on z direction

$$D^F = \begin{bmatrix} 0 & & & -\frac{1}{2} \cot \frac{h}{2} \\ -\frac{1}{2} \cot \frac{h}{2} & \ddots & \ddots & \frac{1}{2} \cot h \\ \frac{1}{2} \cot h & \ddots & \ddots & -\frac{1}{2} \cot \frac{3h}{2} \\ \vdots & \ddots & \ddots & \vdots \\ \frac{1}{2} \cot \frac{h}{2} & & & 0 \end{bmatrix}, \quad D^F + (D^F)^T = 0, \quad D^F \mathbf{1} = 0$$

Quadrature based operators on the reference wedge

- Spectral differentiation matrix D^F on z direction

$$D^F = \begin{bmatrix} 0 & & & -\frac{1}{2} \cot \frac{h}{2} \\ -\frac{1}{2} \cot \frac{h}{2} & \ddots & \ddots & \frac{1}{2} \cot h \\ \frac{1}{2} \cot h & \ddots & \ddots & -\frac{1}{2} \cot \frac{3h}{2} \\ \vdots & \ddots & \ddots & \vdots \\ \frac{1}{2} \cot \frac{h}{2} & & & 0 \end{bmatrix}, \quad D^F + (D^F)^T = 0, \quad D^F \mathbf{1} = 0$$

- Hybridized operator on z direction

$$Q_{h,3} = h D^F \otimes \begin{bmatrix} w^P & \\ & 0 \end{bmatrix}$$

mimics integration by parts

$$Q_{h,3} + (Q_{h,3})^T = 0 \cong \int_{\widehat{D}} u \frac{\partial v}{\partial x_3} + \int_{\widehat{D}} v \frac{\partial u}{\partial x_3} = \int_{\partial \widehat{D}} uv \widehat{n}_3 = 0$$

Quadrature based operators on the reference wedge

- Spectral differentiation matrix D^F on z direction

$$D^F = \begin{bmatrix} 0 & & & -\frac{1}{2} \cot \frac{h}{2} \\ -\frac{1}{2} \cot \frac{h}{2} & \ddots & \ddots & \frac{1}{2} \cot h \\ \frac{1}{2} \cot h & \ddots & \ddots & -\frac{1}{2} \cot \frac{3h}{2} \\ \vdots & \ddots & \ddots & \vdots \\ \frac{1}{2} \cot \frac{h}{2} & & & 0 \end{bmatrix}, \quad D^F + (D^F)^T = 0, \quad D^F \mathbf{1} = 0$$

- Hybridized operator on z direction

$$Q_{h,3} = h D^F \otimes \begin{bmatrix} w^P \\ 0 \end{bmatrix}$$

mimics integration by parts

$$Q_{h,3} + (Q_{h,3})^T = 0 \cong \int_{\widehat{D}} u \frac{\partial v}{\partial x_3} + \int_{\widehat{D}} v \frac{\partial u}{\partial x_3} = \int_{\partial \widehat{D}} uv \widehat{n}_3 = 0$$

Entropy conservative fluxes and flux differencing

- Tadmor's entropy conservative numerical flux:

$$f_S(u, u) = f(u) \quad (\text{consistency})$$

$$f_S(u_L, u_R) = f_S(u_R, u_L) \quad (\text{symmetry})$$

$$(v(u_L) - v(u_R))^T f_S(u_L, u_R) = \psi(v(u_L)) - \psi(v(u_R)) \quad (\text{entropy conservation})$$

Entropy conservative fluxes and flux differencing

- Tadmor's entropy conservative numerical flux:

$$f_S(u, u) = f(u) \quad (\text{consistency})$$

$$f_S(u_L, u_R) = f_S(u_R, u_L) \quad (\text{symmetry})$$

$$(v(u_L) - v(u_R))^T f_S(u_L, u_R) = \psi(v(u_L)) - \psi(v(u_R)) \quad (\text{entropy conservation})$$

- Flux differencing technique

$$\frac{\partial f(u(x))}{\partial x_i} = 2 \frac{\partial f_S(u(x), u(y))}{\partial x_i} \Big|_{x=y} \approx 2(D \circ F)\mathbf{1}$$

Entropy projection

- Recall the first step in proof of entropy inequality

$$\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = 0 \implies \int_{\Omega} v^T \frac{\partial u}{\partial t} + \int_{\Omega} v^T \frac{\partial f(u)}{\partial x} = 0$$

Entropy projection

- Recall the first step in proof of entropy inequality

$$\frac{\partial \mathbf{u}}{\partial t} + \frac{\partial f(\mathbf{u})}{\partial \mathbf{x}} = 0 \implies \int_{\Omega} \mathbf{v}^T \frac{\partial \mathbf{u}}{\partial t} + \int_{\Omega} \mathbf{v}^T \frac{\partial f(\mathbf{u})}{\partial \mathbf{x}} = 0$$

- Entropy projected variables, approximation of $\Pi_N \mathbf{v}$

$$\mathbf{v}_h = \mathbf{P}_q \mathbf{v}(\mathbf{u}_q)$$

Entropy projection

- Recall the first step in proof of entropy inequality

$$\frac{\partial \mathbf{u}}{\partial t} + \frac{\partial \mathbf{f}(\mathbf{u})}{\partial x} = 0 \implies \int_{\Omega} \mathbf{v}^T \frac{\partial \mathbf{u}}{\partial t} + \int_{\Omega} \mathbf{v}^T \frac{\partial \mathbf{f}(\mathbf{u})}{\partial x} = 0$$

- Entropy projected variables, approximation of $\Pi_N \mathbf{v}$

$$\mathbf{v}_h = \mathbf{P}_q \mathbf{v}(\mathbf{u}_q)$$

- Entropy conservation of the numerical flux

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

Entropy projection

- Recall the first step in proof of entropy inequality

$$\frac{\partial \mathbf{u}}{\partial t} + \frac{\partial \mathbf{f}(\mathbf{u})}{\partial x} = 0 \implies \int_{\Omega} \mathbf{v}^T \frac{\partial \mathbf{u}}{\partial t} + \int_{\Omega} \mathbf{v}^T \frac{\partial \mathbf{f}(\mathbf{u})}{\partial x} = 0$$

- Entropy projected variables, approximation of $\Pi_N \mathbf{v}$

$$\mathbf{v}_h = \mathbf{P}_q \mathbf{v}(\mathbf{u}_q)$$

- Entropy conservation of the numerical flux

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

- Entropy projected conservative variables, approximation of $\mathbf{u}(\Pi_N \mathbf{v})$

$$\tilde{\mathbf{u}} = \mathbf{u}(\mathbf{V}_h \mathbf{v}_h)$$

ESDG-Fourier formulation on the reference element

- DG variational formulation

$$\frac{\partial \mathbf{u}}{\partial t} + \sum_{i=1}^3 \frac{\partial f_i(\mathbf{u})}{\partial x_i} = 0$$

ESDG-Fourier formulation on the reference element

- DG variational formulation

$$\frac{\partial \mathbf{u}}{\partial t} + \sum_{i=1}^3 \frac{\partial \mathbf{f}_i(\mathbf{u})}{\partial x_i} = 0$$

$$\left(\frac{\partial \mathbf{u}_N}{\partial t}, \varphi_m \right)_{\widehat{D}} + \sum_{i=1}^3 \left(\frac{\partial \mathbf{f}_i(\mathbf{u}_N)}{\partial x_i}, \varphi_m \right)_{\widehat{D}} + \sum_{i=1}^3 \langle \widehat{\mathbf{n}}_i \cdot (\mathbf{f}_i^* - \mathbf{f}_i(\mathbf{u}_N)), \varphi_m \rangle_{\partial \widehat{D}} = 0,$$

$$m = 1, \dots, N_p \quad \text{(Galerkin approximation)}$$

ESDG-Fourier formulation on the reference element

- DG variational formulation

$$\frac{\partial \mathbf{u}}{\partial t} + \sum_{i=1}^3 \frac{\partial \mathbf{f}_i(\mathbf{u})}{\partial x_i} = 0$$

$$\left(\frac{\partial \mathbf{u}_N}{\partial t}, \varphi_m \right)_{\widehat{D}} + \sum_{i=1}^3 \left(\frac{\partial \mathbf{f}_i(\mathbf{u}_N)}{\partial x_i}, \varphi_m \right)_{\widehat{D}} + \sum_{i=1}^2 \langle \widehat{\mathbf{n}}_i \cdot (\mathbf{f}_i^* - \mathbf{f}_i(\mathbf{u}_N)), \varphi_m \rangle_{\partial \widehat{D}} = 0,$$

$$m = 1, \dots, N_p \quad \text{(Galerkin approximation)}$$

ESDG-Fourier formulation on the reference element

- DG variational formulation

$$\frac{\partial \mathbf{u}}{\partial t} + \sum_{i=1}^3 \frac{\partial \mathbf{f}_i(\mathbf{u})}{\partial x_i} = 0$$

$$\left(\frac{\partial \mathbf{u}_N}{\partial t}, \varphi_m \right)_{\widehat{D}} + \sum_{i=1}^3 \left(\frac{\partial \mathbf{f}_i(\mathbf{u}_N)}{\partial x_i}, \varphi_m \right)_{\widehat{D}} + \sum_{i=1}^2 \langle \widehat{\mathbf{n}}_i \cdot (\mathbf{f}_i^* - \mathbf{f}_i(\mathbf{u}_N)), \varphi_m \rangle_{\partial \widehat{D}} = 0,$$

$$m = 1, \dots, N_p \quad \text{(Galerkin approximation)}$$

- ESDG-Fourier formulation

$$\mathbf{M} \frac{\partial \mathbf{u}_h}{\partial t} = -\mathbf{V}_h^T \left(\sum_{i=1}^3 2(\mathbf{Q}_{h,i} \circ \mathbf{F}_i) \mathbf{1} \right) - \mathbf{V}_f^T \sum_{i=1}^2 \mathbf{B}_i (\mathbf{f}_i^* - \mathbf{f}_i(\widetilde{\mathbf{u}}_f))$$

$$(\mathbf{F}_n)_{ij} = \mathbf{f}_{n,S}(\widetilde{\mathbf{u}}_i, \widetilde{\mathbf{u}}_j), \quad 1 \leq i, j \leq N_h$$

Proof of entropy conservation

Theorem (Conservation of entropy)

Let $f_{i,S}$ be an entropy conservative flux. Assuming continuity in time, if $\eta(\mathbf{u}_h)$ is convex, solutions \mathbf{u}_h satisfy a semi-discrete conservation of entropy

$$\mathbf{1}^T \mathbf{W} \frac{d\eta(\mathbf{u}_q)}{dt} = \sum_{i=1}^{d-1} \mathbf{1}^T \mathbf{W}_f (\text{diag}(\widehat{\mathbf{n}}_i) (\psi_i(\widetilde{\mathbf{u}}_f) - \widetilde{\mathbf{v}}_f^T \mathbf{f}_i^*))$$

which are the quadrature approximations to the following

$$\int_{\widehat{D}} \frac{\partial \eta(\mathbf{u}_N)}{\partial t} dx = \sum_{i=1}^d \int_{\partial \widehat{D}} (\psi_i(\Pi_N \mathbf{v}) - (\Pi_N \mathbf{v})^T \mathbf{f}_i^*) \mathbf{n}_i^k,$$

Proof of entropy conservation

Theorem (Conservation of entropy)

Let $f_{i,S}$ be an entropy conservative flux. Assuming continuity in time, if $\eta(\mathbf{u}_h)$ is convex, solutions \mathbf{u}_h satisfy a semi-discrete conservation of entropy

$$\mathbf{1}^T \mathbf{W} \frac{d\eta(\mathbf{u}_q)}{dt} = \sum_{i=1}^{d-1} \mathbf{1}^T \mathbf{W}_f (\text{diag}(\widehat{\mathbf{n}}_i) (\psi_i(\widetilde{\mathbf{u}}_f) - \widetilde{\mathbf{v}}_f^T \mathbf{f}_i^*))$$

which are the quadrature approximations to the following

$$\int_{\widehat{D}} \frac{\partial \eta(\mathbf{u}_N)}{\partial t} dx = \sum_{i=1}^d \int_{\partial \widehat{D}} (\psi_i(\Pi_N \mathbf{v}) - (\Pi_N \mathbf{v})^T \mathbf{f}_i^*) \mathbf{n}_i^k,$$

- Test with \mathbf{v}_h . Volume contributions

$$-\mathbf{v}_h^T \mathbf{V}_h^T \left(\sum_{n=1}^d 2(\mathbf{Q}_{h,n} \circ \mathbf{F}_n) \mathbf{1} \right) = - \sum_{n=1}^d \sum_{i,j} (\mathbf{Q}_{h,n})_{ij} (\widetilde{\mathbf{v}}_i - \widetilde{\mathbf{v}}_j)^T \mathbf{f}_{n,S} (\widetilde{\mathbf{u}}_i, \widetilde{\mathbf{u}}_j)$$

Proof of entropy conservation

Theorem (Conservation of entropy)

Let $f_{i,S}$ be an entropy conservative flux. Assuming continuity in time, if $\eta(\mathbf{u}_h)$ is convex, solutions \mathbf{u}_h satisfy a semi-discrete conservation of entropy

$$\mathbf{1}^T \mathbf{W} \frac{d\eta(\mathbf{u}_q)}{dt} = \sum_{i=1}^{d-1} \mathbf{1}^T \mathbf{W}_f (\text{diag}(\widehat{\mathbf{n}}_i) (\psi_i(\widetilde{\mathbf{u}}_f) - \widetilde{\mathbf{v}}_f^T \mathbf{f}_i^*))$$

which are the quadrature approximations to the following

$$\int_{\widehat{D}} \frac{\partial \eta(\mathbf{u}_N)}{\partial t} dx = \sum_{i=1}^d \int_{\partial \widehat{D}} (\psi_i(\Pi_N \mathbf{v}) - (\Pi_N \mathbf{v})^T \mathbf{f}_i^*) \mathbf{n}_i^k,$$

- Test with \mathbf{v}_h . Volume contributions

$$-\mathbf{v}_h^T \mathbf{V}_h^T \left(\sum_{n=1}^d 2(\mathbf{Q}_{h,n} \circ \mathbf{F}_n) \mathbf{1} \right) = - \sum_{n=1}^d \sum_{i,j} (\mathbf{Q}_{h,n})_{ij} (\psi_n(\widetilde{\mathbf{v}}_i) - \psi_n(\widetilde{\mathbf{v}}_j))$$

Proof of entropy conservation

Theorem (Conservation of entropy)

Let $f_{i,S}$ be an entropy conservative flux. Assuming continuity in time, if $\eta(\mathbf{u}_h)$ is convex, solutions \mathbf{u}_h satisfy a semi-discrete conservation of entropy

$$\mathbf{1}^T \mathbf{W} \frac{d\eta(\mathbf{u}_q)}{dt} = \sum_{i=1}^{d-1} \mathbf{1}^T \mathbf{W}_f (\text{diag}(\widehat{\mathbf{n}}_i) (\psi_i(\widetilde{\mathbf{u}}_f) - \widetilde{\mathbf{v}}_f^T \mathbf{f}_i^*))$$

which are the quadrature approximations to the following

$$\int_{\widehat{D}} \frac{\partial \eta(\mathbf{u}_N)}{\partial t} dx = \sum_{i=1}^d \int_{\partial \widehat{D}} (\psi_i(\Pi_N \mathbf{v}) - (\Pi_N \mathbf{v})^T \mathbf{f}_i^*) \mathbf{n}_i^k,$$

- Test with \mathbf{v}_h . Volume contributions

$$-\mathbf{v}_h^T \mathbf{V}_h^T \left(\sum_{n=1}^d 2(\mathbf{Q}_{h,n} \circ \mathbf{F}_n) \mathbf{1} \right) = - \sum_{n=1}^d \psi_n(\widetilde{\mathbf{v}})^T (\mathbf{Q}_{h,n} - \mathbf{Q}_{h,n}^T) \mathbf{1}$$

Proof of entropy conservation

Theorem (Conservation of entropy)

Let $f_{i,S}$ be an entropy conservative flux. Assuming continuity in time, if $\eta(\mathbf{u}_h)$ is convex, solutions \mathbf{u}_h satisfy a semi-discrete conservation of entropy

$$\mathbf{1}^T \mathbf{W} \frac{d\eta(\mathbf{u}_q)}{dt} = \sum_{i=1}^{d-1} \mathbf{1}^T \mathbf{W}_f (\text{diag}(\widehat{\mathbf{n}}_i) (\psi_i(\widetilde{\mathbf{u}}_f) - \widetilde{\mathbf{v}}_f^T \mathbf{f}_i^*))$$

which are the quadrature approximations to the following

$$\int_{\widehat{D}} \frac{\partial \eta(\mathbf{u}_N)}{\partial t} dx = \sum_{i=1}^d \int_{\partial \widehat{D}} (\psi_i(\Pi_N \mathbf{v}) - (\Pi_N \mathbf{v})^T \mathbf{f}_i^*) \mathbf{n}_i^k,$$

- Test with \mathbf{v}_h . Volume contributions

$$-\mathbf{v}_h^T \mathbf{V}_h^T \left(\sum_{n=1}^d 2(\mathbf{Q}_{h,n} \circ \mathbf{F}_n) \mathbf{1} \right) = - \sum_{n=1}^{d-1} \psi_n(\widetilde{\mathbf{v}})^T (2\mathbf{Q}_{h,n} - \mathbf{B}_{h,n}^T) \mathbf{1} - \psi_d(\widetilde{\mathbf{v}})^T 2\mathbf{Q}_{h,d} \mathbf{1}$$

Proof of entropy conservation

Theorem (Conservation of entropy)

Let $f_{i,S}$ be an entropy conservative flux. Assuming continuity in time, if $\eta(\mathbf{u}_h)$ is convex, solutions \mathbf{u}_h satisfy a semi-discrete conservation of entropy

$$\mathbf{1}^T \mathbf{W} \frac{d\eta(\mathbf{u}_q)}{dt} = \sum_{i=1}^{d-1} \mathbf{1}^T \mathbf{W}_f (\text{diag}(\widehat{\mathbf{n}}_i) (\psi_i(\widetilde{\mathbf{u}}_f) - \widetilde{\mathbf{v}}_f^T \mathbf{f}_i^*))$$

which are the quadrature approximations to the following

$$\int_{\widehat{D}} \frac{\partial \eta(\mathbf{u}_N)}{\partial t} dx = \sum_{i=1}^d \int_{\partial \widehat{D}} (\psi_i(\Pi_N \mathbf{v}) - (\Pi_N \mathbf{v})^T \mathbf{f}_i^*) \mathbf{n}_i^k,$$

- Test with \mathbf{v}_h . Volume contributions

$$-\mathbf{v}_h^T \mathbf{V}_h^T \left(\sum_{n=1}^d 2(\mathbf{Q}_{h,n} \circ \mathbf{F}_n) \mathbf{1} \right) = \sum_{n=1}^{d-1} \psi_n(\widetilde{\mathbf{v}})^T \mathbf{B}_{h,n}^T \mathbf{1}$$

ESDG-Fourier formulation on the mapped elements

- Affine mapping from the reference element $\Phi^k(\hat{D}) = D^k$

$$\Phi^k(r, s) = \Phi^{P,k}(r)\Phi^F(s), \quad J^k = \left| \frac{\partial \Phi^k(r, s)}{\partial(r, s)} \right| = \left| \frac{\partial \Phi^{P,k}(r)}{\partial r} \right| \left| \frac{\partial \Phi^F(s)}{\partial s} \right| = J^{P,k}J^F$$

ESDG-Fourier formulation on the mapped elements

- Affine mapping from the reference element $\Phi^k(\widehat{D}) = D^k$

$$\Phi^k(r, s) = \Phi^{P,k}(r)\Phi^F(s), \quad J^k = \left| \frac{\partial \Phi^k(r, s)}{\partial(r, s)} \right| = \left| \frac{\partial \Phi^{P,k}(r)}{\partial r} \right| \left| \frac{\partial \Phi^F(s)}{\partial s} \right| = J^{P,k}J^F$$

- Geometric terms $G^P = \frac{\partial(\Phi^{P,k})^{-1}(x)}{\partial x}$, reference and physical normals \hat{n}_i, n_i^k are related through identity

$$\sum J^{P,k} G_{ij}(\hat{n}_i \circ \hat{J}_f) = n_i^k \circ J_f^k$$

ESDG-Fourier formulation on the mapped elements

- Affine mapping from the reference element $\Phi^k(\hat{D}) = D^k$

$$\Phi^k(r, s) = \Phi^{P,k}(r)\Phi^F(s), \quad J^k = \left| \frac{\partial \Phi^k(r, s)}{\partial(r, s)} \right| = \left| \frac{\partial \Phi^{P,k}(r)}{\partial r} \right| \left| \frac{\partial \Phi^F(s)}{\partial s} \right| = J^{P,k}J^F$$

- Geometric terms $G^P = \frac{\partial(\Phi^{P,k})^{-1}(x)}{\partial x}$, reference and physical normals \hat{n}_i, n_i^k are related through identity

$$\sum J^{P,k} G_{ij}(\hat{n}_i \circ \hat{J}_f) = n_i^k \circ J_f^k$$

- SBP property invariant under affine transformation

$$\begin{aligned} Q_{h,i}^k &= J^k G_{ij}^P Q_{h,i}, \quad i = 1, 2, \quad Q_{h,3}^k = J^{P,k} Q_{h,3} \\ B_i^k &= J^k G_{ij}^P B_i \end{aligned}$$

ESDG-Fourier formulation on the mapped elements

- ESDG-Fourier formulation on mapped elements

$$\mathbf{M}^k \frac{\partial \mathbf{u}_h}{\partial t} = -(\mathbf{V}_h^k)^T \left(\sum_{i=1}^d 2(\mathbf{Q}_{h,i}^k \circ \mathbf{F}_i^k) \mathbf{1} \right) - (\mathbf{V}_f^k)^T \sum_{i=1}^{d-1} \mathbf{B}_i^k (\mathbf{f}_i^* - \mathbf{f}_i(\tilde{\mathbf{u}}_f^k))$$
$$(\mathbf{F}_n^k)_{ij} = \mathbf{f}_{n,S}(\tilde{\mathbf{u}}_i^k, \tilde{\mathbf{u}}_j^k), \quad 1 \leq i, j \leq N_h, \quad \mathbf{f}_n^* = \mathbf{f}_{n,S}(\tilde{\mathbf{u}}_f, \tilde{\mathbf{u}}_f^+)$$

ESDG-Fourier formulation on the mapped elements

- ESDG-Fourier formulation on mapped elements

$$\mathbf{M}^k \frac{\partial \mathbf{u}_h}{\partial t} = -(\mathbf{V}_h^k)^T \left(\sum_{i=1}^d 2(\mathbf{Q}_{h,i}^k \circ \mathbf{F}_i^k) \mathbf{1} \right) - (\mathbf{V}_f^k)^T \sum_{i=1}^{d-1} \mathbf{B}_i^k (\mathbf{f}_i^* - \mathbf{f}_i(\tilde{\mathbf{u}}_f^k))$$
$$(\mathbf{F}_n^k)_{ij} = \mathbf{f}_{n,S}(\tilde{\mathbf{u}}_i^k, \tilde{\mathbf{u}}_j^k), \quad 1 \leq i, j \leq N_h, \quad \mathbf{f}_n^* = \mathbf{f}_{n,S}(\tilde{\mathbf{u}}_f, \tilde{\mathbf{u}}_f^+)$$

- Achieve entropy stability through entropy dissipative penalization

$$\mathbf{f}_n^* = \mathbf{f}_{n,S}(\tilde{\mathbf{u}}_f, \tilde{\mathbf{u}}_f^+) - \frac{\lambda}{2} \llbracket \tilde{\mathbf{u}} \rrbracket$$

- Implement in Julia with *CUDA.jl* package



Figure 12: Julia programming languages

GPU programming

- Implement in Julia with *CUDA.jl* package
- Run on Google Cloud with Nvidia Tesla V100



Figure 12: Nvidia Tesla V100 GPU

GPU programming

- Implement in Julia with *CUDA.jl* package
- Run on Google Cloud with Nvidia Tesla V100
- GPU programming + explicit timestepping

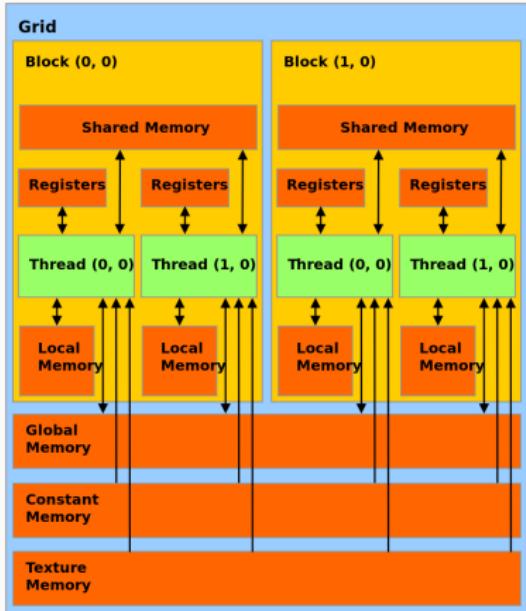


Figure 12: GPU memory hierarchy

GPU implementation - Two kernel split

- Skew-symmetric formulation

$$\begin{aligned}\frac{\partial \mathbf{u}_h^k}{\partial t} = & -\frac{1}{J^k} \mathbf{M}^{-1} \mathbf{V}_h^T \left(\sum_{n=1}^2 ((\mathbf{Q}_{h,n}^k - (\mathbf{Q}_{h,n}^k)^T) \circ \mathbf{F}_n^k) \mathbf{1} + 2(\mathbf{Q}_{h,3}^k \circ \mathbf{F}_3^k) \right) \\ & - \frac{1}{J^k} \mathbf{M}^{-1} \mathbf{V}_f^T \sum_{m=1}^{d-1} \mathbf{B}_m^k \mathbf{f}_m^*, \quad (\mathbf{F}_n^k)_{ij} = \mathbf{f}_{n,S}(\tilde{\mathbf{u}}_i^k, \tilde{\mathbf{u}}_j^k)\end{aligned}$$

GPU implementation - Two kernel split

- Skew-symmetric formulation

$$\underbrace{\frac{\partial \mathbf{u}_h^k}{\partial t}}_{\text{update kernel}} = -\frac{1}{J^k} \mathbf{M}^{-1} \mathbf{V}_h^T \left(\underbrace{\sum_{n=1}^2 ((\mathbf{Q}_{h,n}^k - (\mathbf{Q}_{h,n}^k)^T) \circ \mathbf{F}_n^k) \mathbf{1}}_{\text{xy flux-differencing kernel}} + \underbrace{2(\mathbf{Q}_{h,3}^k \circ \mathbf{F}_3^k)}_{\text{z flux-differencing kernel}} \right)$$
$$-\underbrace{\frac{1}{J^k} \mathbf{M}^{-1} \mathbf{V}_f^T \sum_{m=1}^{d-1} \mathbf{B}_m^k \mathbf{f}_m^*}_{\text{surface kernel}}, \quad (\mathbf{F}_n^k)_{ij} = \mathbf{f}_{n,S} \underbrace{(\tilde{\mathbf{u}}_i^k, \tilde{\mathbf{u}}_j^k)}_{\text{entropy projection kernel}}$$

GPU implementation - Two kernel split

- Skew-symmetric formulation

$$\underbrace{\frac{\partial \mathbf{u}_h^k}{\partial t}}_{\text{update kernel}} = -\frac{1}{J^k} \mathbf{M}^{-1} \mathbf{V}_h^T \left(\underbrace{\sum_{n=1}^2 ((\mathbf{Q}_{h,n}^k - (\mathbf{Q}_{h,n}^k)^T) \circ \mathbf{F}_n^k) \mathbf{1}}_{\text{xy flux-differencing kernel}} + \underbrace{2(\mathbf{Q}_{h,3}^k \circ \mathbf{F}_3^k)}_{\text{z flux-differencing kernel}} \right)$$
$$-\underbrace{\frac{1}{J^k} \mathbf{M}^{-1} \mathbf{V}_f^T \sum_{m=1}^{d-1} \mathbf{B}_m^k \mathbf{f}_m^*}_{\text{surface kernel}}, \quad (\mathbf{F}_n^k)_{ij} = \mathbf{f}_{n,S} \underbrace{(\tilde{\mathbf{u}}_i^k, \tilde{\mathbf{u}}_j^k)}_{\text{entropy projection kernel}}$$

GPU implementation - Two kernel split

- Skew-symmetric formulation

$$\underbrace{\frac{\partial \mathbf{u}_h^k}{\partial t}}_{\text{update kernel}} = -\frac{1}{J^k} \mathbf{M}^{-1} \mathbf{V}_h^T \left(\underbrace{\sum_{n=1}^2 ((\mathbf{Q}_{h,n}^k - (\mathbf{Q}_{h,n}^k)^T) \circ \mathbf{F}_n^k) \mathbf{1}}_{\text{xy flux-differencing kernel}} + \underbrace{2(\mathbf{Q}_{h,3}^k \circ \mathbf{F}_3^k)}_{\text{z flux-differencing kernel}} \right)$$
$$-\underbrace{\frac{1}{J^k} \mathbf{M}^{-1} \mathbf{V}_f^T \sum_{m=1}^{d-1} \mathbf{B}_m^k \mathbf{f}_m^*}_{\text{surface kernel}}, \quad (\mathbf{F}_n^k)_{ij} = \mathbf{f}_{n,S} \underbrace{(\tilde{\mathbf{u}}_i^k, \tilde{\mathbf{u}}_j^k)}_{\text{entropy projection kernel}}$$

GPU implementation - Two kernel split

- Skew-symmetric formulation

$$\underbrace{\frac{\partial \mathbf{u}_h^k}{\partial t}}_{\text{update kernel}} = -\frac{1}{J^k} \mathbf{M}^{-1} \mathbf{V}_h^T \left(\underbrace{\sum_{n=1}^2 ((\mathbf{Q}_{h,n}^k - (\mathbf{Q}_{h,n}^k)^T) \circ \mathbf{F}_n^k) \mathbf{1}}_{\text{xy flux-differencing kernel}} + \underbrace{2(\mathbf{Q}_{h,3}^k \circ \mathbf{F}_3^k)}_{\text{z flux-differencing kernel}} \right)$$
$$-\underbrace{\frac{1}{J^k} \mathbf{M}^{-1} \mathbf{V}_f^T \sum_{m=1}^{d-1} \mathbf{B}_m^k \mathbf{f}_m^*}_{\text{surface kernel}}, \quad (\mathbf{F}_n^k)_{ij} = \mathbf{f}_{n,S} \underbrace{(\tilde{\mathbf{u}}_i^k, \tilde{\mathbf{u}}_j^k)}_{\text{entropy projection kernel}}$$

- Evaluate flux matrix \mathbf{F} on-the-fly

GPU implementation - Two kernel split

- Skew-symmetric formulation

$$\underbrace{\frac{\partial \mathbf{u}_h^k}{\partial t}}_{\text{update kernel}} = -\frac{1}{J^k} \mathbf{M}^{-1} \mathbf{V}_h^T \left(\underbrace{\sum_{n=1}^2 ((\mathbf{Q}_{h,n}^k - (\mathbf{Q}_{h,n}^k)^T) \circ \mathbf{F}_n^k) \mathbf{1}}_{\text{xy flux-differencing kernel}} + \underbrace{2(\mathbf{Q}_{h,3}^k \circ \mathbf{F}_3^k)}_{\text{z flux-differencing kernel}} \right)$$
$$-\underbrace{\frac{1}{J^k} \mathbf{M}^{-1} \mathbf{V}_f^T \sum_{m=1}^{d-1} \mathbf{B}_m^k \mathbf{f}_m^*}_{\text{surface kernel}}, \quad (\mathbf{F}_n^k)_{ij} = \mathbf{f}_{n,S} \underbrace{(\tilde{\mathbf{u}}_i^k, \tilde{\mathbf{u}}_j^k)}_{\text{entropy projection kernel}}$$

- Evaluate flux matrix \mathbf{F} on-the-fly
- Two kernel split: data locality

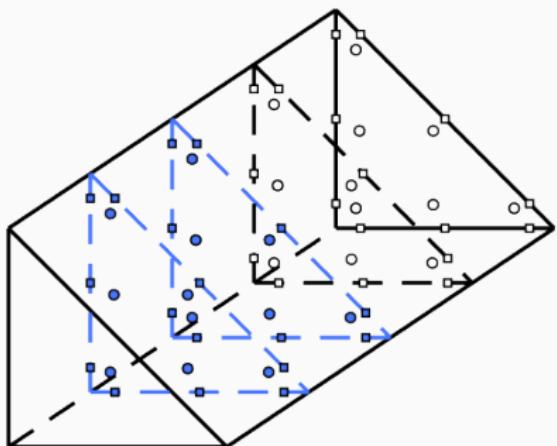
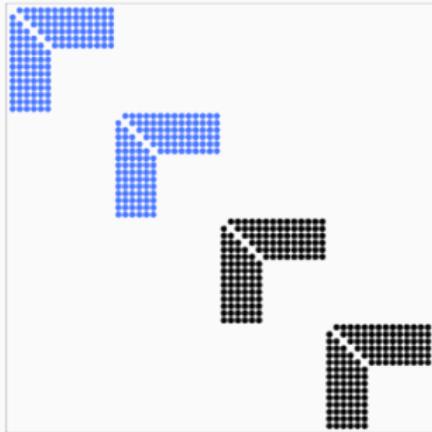
GPU implementation - Triangle kernel

$$\sum_{n=1}^2 ((\mathbf{Q}_{h,n}^k - (\mathbf{Q}_{h,n}^k)^T) \circ \mathbf{F}_n^k) \mathbf{1}$$

GPU implementation - Triangle kernel

$$\sum_{n=1}^2 ((\mathbf{Q}_{h,n}^k - (\mathbf{Q}_{h,n}^k)^T) \circ \mathbf{F}_n^k) \mathbf{1}$$

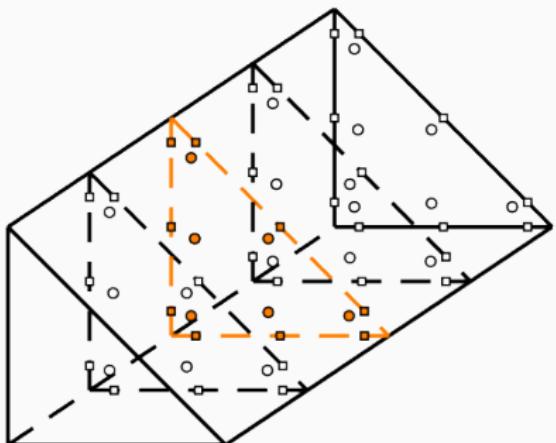
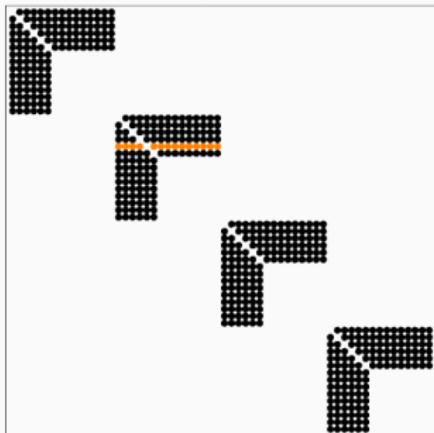
Sparisity pattern of \mathbf{Q}



GPU implementation - Triangle kernel

At row i : $\sum_{n=1}^2 (((\mathbf{Q}_{h,n}^k - (\mathbf{Q}_{h,n}^k)^T) \circ \mathbf{F}_n^k) \mathbf{1})_i$

Sparisity pattern of \mathbf{Q}



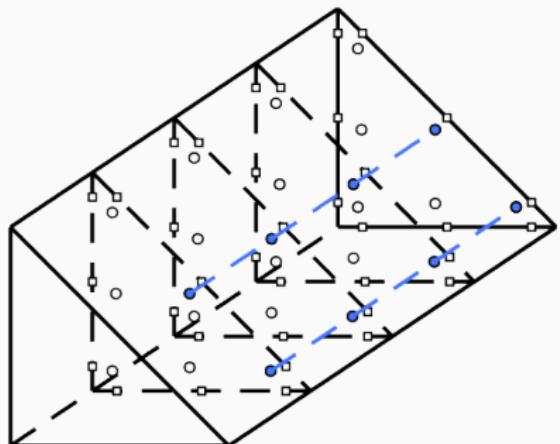
GPU implementation - Fourier mode kernel

$$(\mathbf{Q}_{h,3}^k \circ \mathbf{F}_3^k) \mathbf{1}$$

GPU implementation - Fourier mode kernel

$$(\mathbf{Q}_{h,3}^k \circ \mathbf{F}_3^k) \mathbf{1}$$

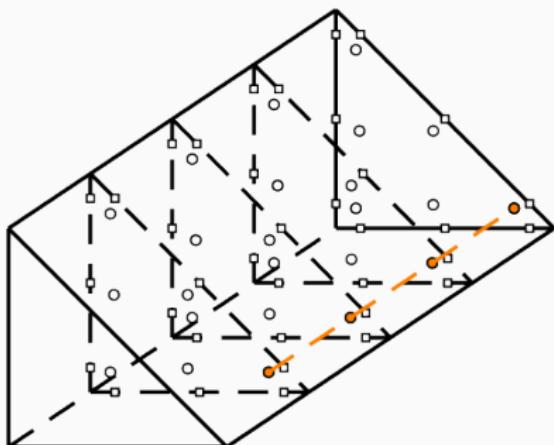
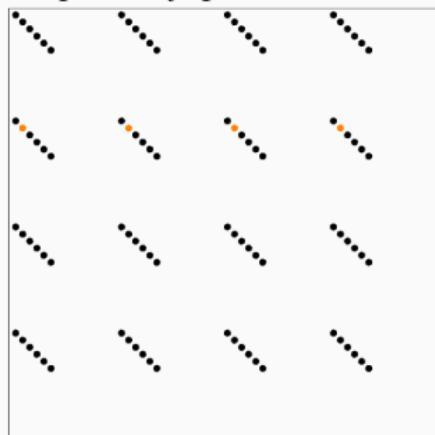
Sparisity pattern of \mathbf{Q}



GPU implementation - Fourier mode kernel

At row i : $((\mathbf{Q}_{h,n}^k \circ \mathbf{F}_n^k) \mathbf{1})_i$

Sparisity pattern of \mathbf{Q}



Compressible Euler equations

- Compressible Euler equations in conservation form

$$\frac{\partial \mathbf{U}}{\partial t} + \sum_{i=1}^3 \frac{\partial f_i(\mathbf{U})}{\partial x_i} = 0$$
$$\mathbf{U} = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ E \end{bmatrix}, \quad f_i(\mathbf{U}) = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho uw \\ u(E + p) \end{bmatrix}, \begin{bmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ \rho vw \\ v(E + p) \end{bmatrix}, \begin{bmatrix} \rho w \\ \rho uw \\ \rho vw \\ \rho w^2 + p \\ w(E + p) \end{bmatrix}$$

- Chandrashekhar's entropy conservative flux³, log mean $\{\!\{u\}\!\}^{\log} = \frac{u_L - u_R}{\log u_L - \log u_R}$

$$f_{i,S}(\mathbf{U}_L, \mathbf{U}_R) = \begin{bmatrix} \{\!\{\rho\}\!\}^{\log} \{\!\{u\}\!\} \\ \{\!\{\rho\}\!\}^{\log} \{\!\{u\}\!\}^2 + p_{avg} \\ \{\!\{\rho\}\!\}^{\log} \{\!\{u\}\!\} \{\!\{v\}\!\} \\ \{\!\{\rho\}\!\}^{\log} \{\!\{u\}\!\} \{\!\{w\}\!\} \\ (E_{avg} + p_{avg}) \{\!\{u\}\!\} \end{bmatrix}, \begin{bmatrix} \{\!\{\rho\}\!\}^{\log} \{\!\{v\}\!\} \\ \{\!\{\rho\}\!\}^{\log} \{\!\{u\}\!\} \{\!\{v\}\!\} \\ \{\!\{\rho\}\!\}^{\log} \{\!\{v\}\!\}^2 + p_{avg} \\ \{\!\{\rho\}\!\}^{\log} \{\!\{v\}\!\} \{\!\{w\}\!\} \\ (E_{avg} + p_{avg}) \{\!\{v\}\!\} \end{bmatrix}, \begin{bmatrix} \{\!\{\rho\}\!\}^{\log} \{\!\{w\}\!\} \\ \{\!\{\rho\}\!\}^{\log} \{\!\{u\}\!\} \{\!\{w\}\!\} \\ \{\!\{\rho\}\!\}^{\log} \{\!\{v\}\!\} \{\!\{w\}\!\} \\ \{\!\{\rho\}\!\}^{\log} \{\!\{w\}\!\}^2 + p_{avg} \\ (E_{avg} + p_{avg}) \{\!\{w\}\!\} \end{bmatrix}$$

³Chandrashekhar, "Kinetic energy preserving and entropy stable finite volume schemes for compressible Euler and Navier-Stokes equations". 25

Numerical experiments: vortex propagation, convecting waves

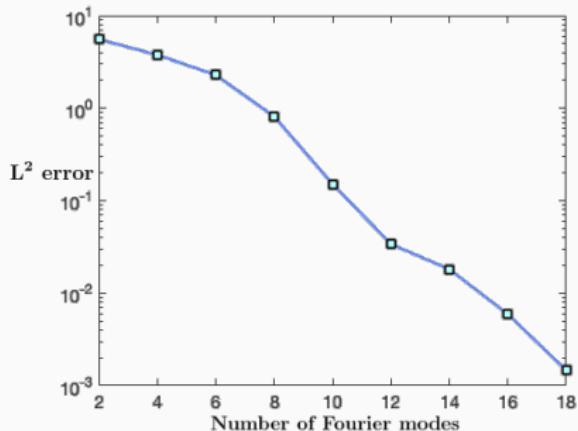


Figure 13: Convergence of vortex propagation in z direction

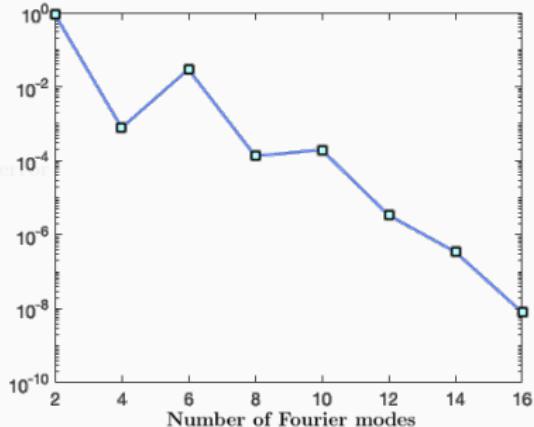


Figure 14: Convergence of convecting wave in x , y and z direction

- Increasing the number of Fourier modes shows spectral convergence

Numerical experiments: doubly periodic shear layers

- Doubly periodic shear layers with varying layer width

$$u(x, t) = \begin{cases} \tanh(\epsilon(1 + z^2)(y + 0.25)) & \text{if } y < 0 \\ \tanh(\epsilon(1 + z^2)(0.25 - y)) & \text{otherwise} \end{cases}$$

$$v(x, t) = \delta \cos(2\pi x), \quad w(x, t) = \delta \cos(2\pi x),$$

$$\rho(x, t) = 1, \quad p(x, t) = \frac{1}{\gamma Ma^2}$$

- Advance in time using adaptive 5-th order Dormand–Prince Runge–Kutta method
- Solve on a refined mesh ($K = 128 \times 128 \times 1$) with 8 Fourier modes, degree 4 polynomials.

Numerical results: doubly periodic shear layers

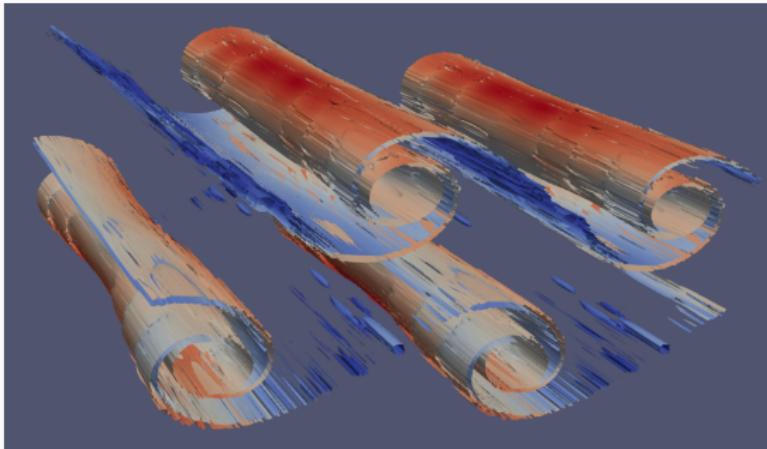


Figure 15: Isosurface of q -criterion $Q = 0.05$, colored by squared norm of velocity

- Q -criterion, Ω vorticity tensor, S rate-of-strain tensor.

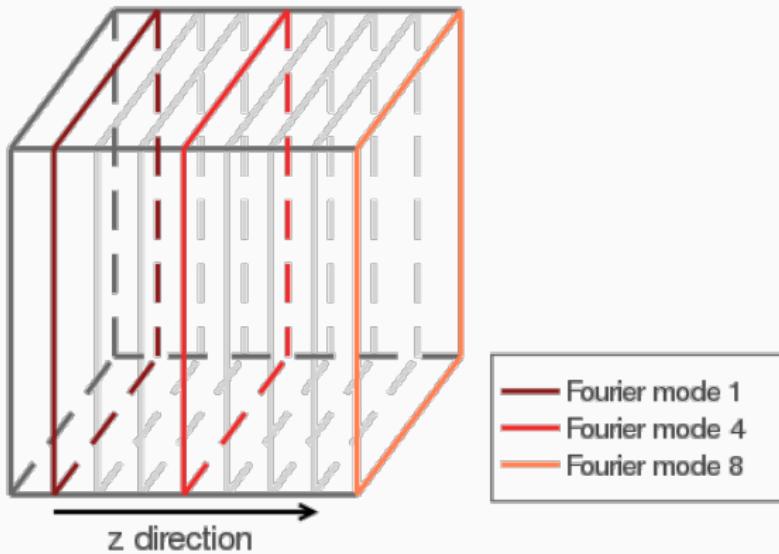
$$Q = \frac{1}{2} \|\boldsymbol{\Omega}\|^2 - \|\mathbf{S}\|^2 > 0$$

- Squared norm of velocity

$$u^2 + v^2 + w^2$$

Numerical results: doubly periodic shear layers

- 3D plotting



- z-component vorticity

$$\frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}$$

Numerical results: doubly periodic shear layers

- z-vorticity with Mach numbers $\text{Ma} = 0.3$ and $\text{Ma} = 0.7$: shocks formed.
Noise around shocks by differentiation

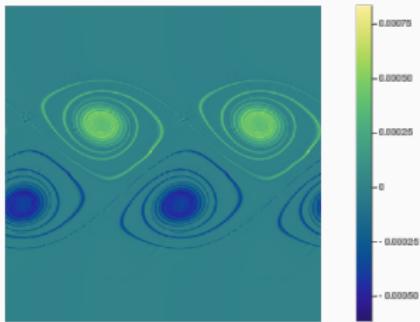


Figure 16: $\text{Ma} = 0.3$, mode 4

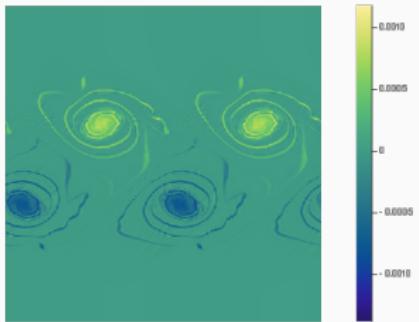


Figure 17: $\text{Ma} = 0.3$, mode 8

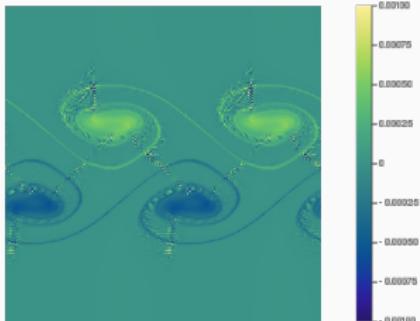


Figure 18: $\text{Ma} = 0.7$, mode 4

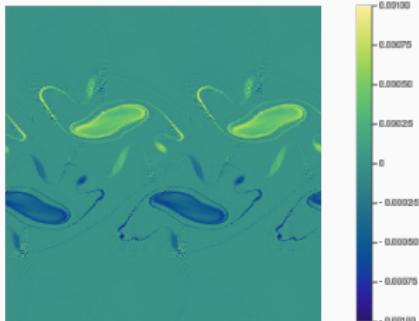


Figure 19: $\text{Ma} = 0.7$, mode 8

Numerical results: doubly periodic shear layers

- Density with Mach numbers $\text{Ma} = 0.3$ and $\text{Ma} = 0.7$: shocks formed

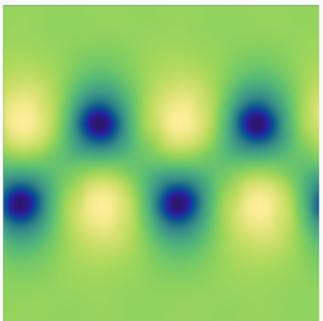


Figure 20: $\text{Ma} = 0.3$, mode 4

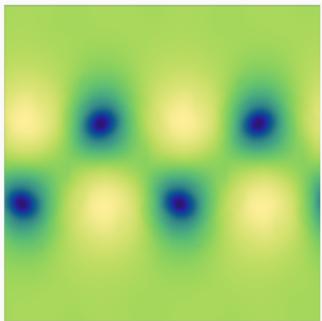


Figure 21: $\text{Ma} = 0.3$, mode 8

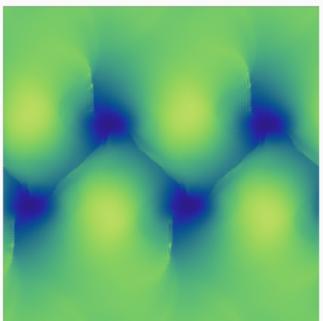


Figure 22: $\text{Ma} = 0.7$, mode 4

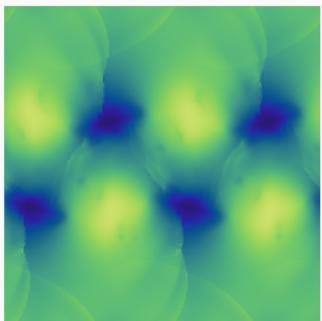


Figure 23: $\text{Ma} = 0.7$, mode 8

Numerical results: underresolved doubly periodic shear layers

- Results with Mach $\text{Ma} = 0.7$ and mesh size $K = 20 \times 20 \times 1$. 8 Fourier modes, degree 7 polynomials.

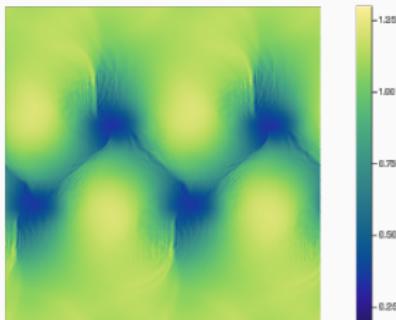


Figure 24: Mode 4

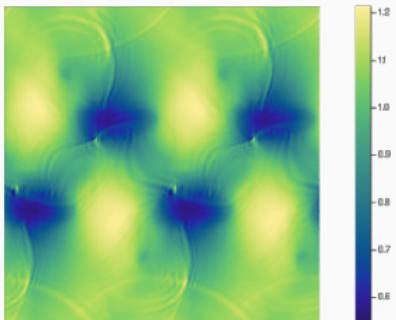


Figure 25: Mode 8

- Remain robust in the underresolved case

Summary and future works

- Entropy stable high order discontinuous Galerkin-Fourier methods:
semi-discrete entropy stability
- GPU implementation using a two-kernel splitting
- Future work: curvilinear and moving meshes, positivity limiter

Thanks everyone for joining!

Thanks Professor Heinkenschloss and Riviere for being on my committee.

Any questions?