Robust and Accurate Viscous Discretization by Hyperbolic Recipe

"What is the best viscous scheme?"

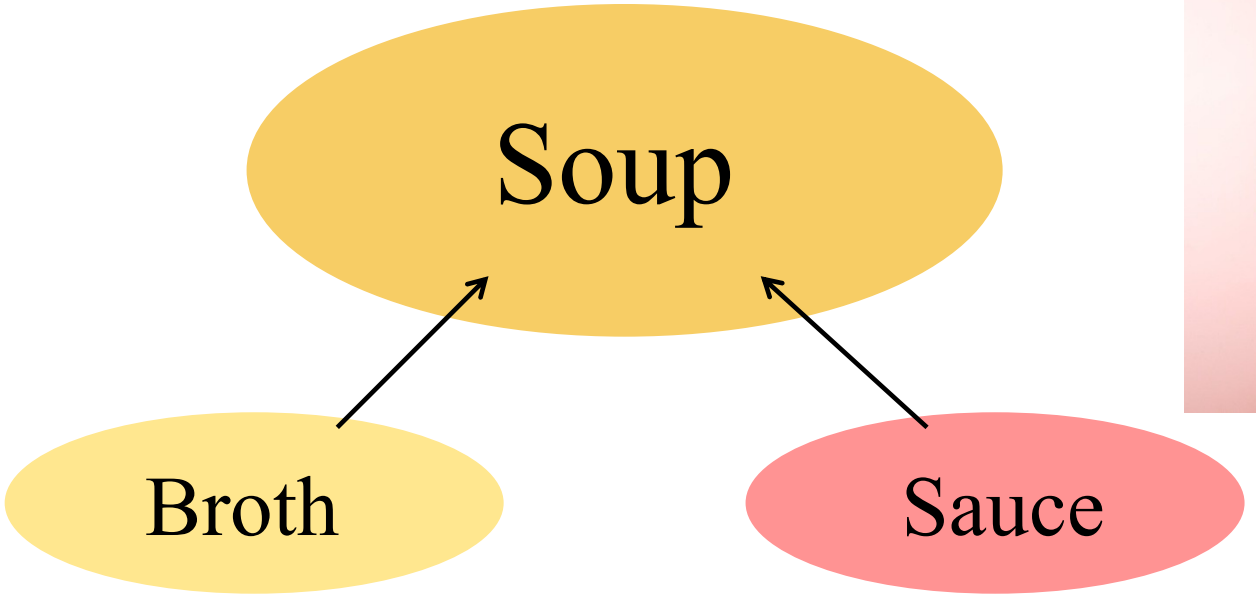
"I give you a recipe. You find the best one."

A story on conventional viscous discretization that lies on the other side of the coin of the hyperbolic viscous system.

Hiroaki Nishikawa
National Institute of Aerospace

Essential Ingredients

Essential Ingredients to Good Ramen Soup: Broth, Sauce



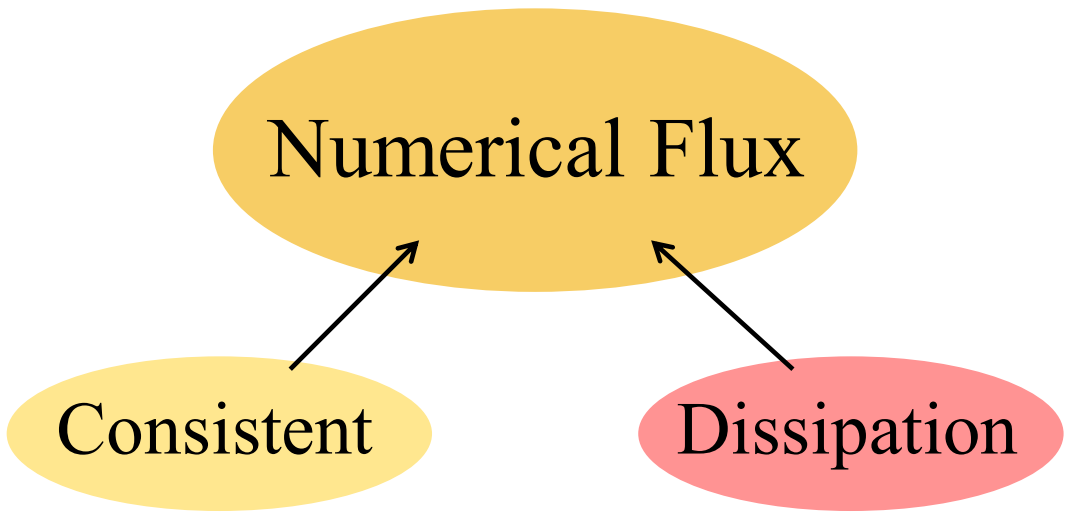
Chicken
Pork
Fish

Soy sauce → Soy-sauce Ramen
Miso → Miso Ramen
Salt → Salt Ramen

1. You cannot forget any of the two.
2. Sauce characterizes the type of ramen.

Essential Ingredients for Advection

Consistent Term and Dissipation Term

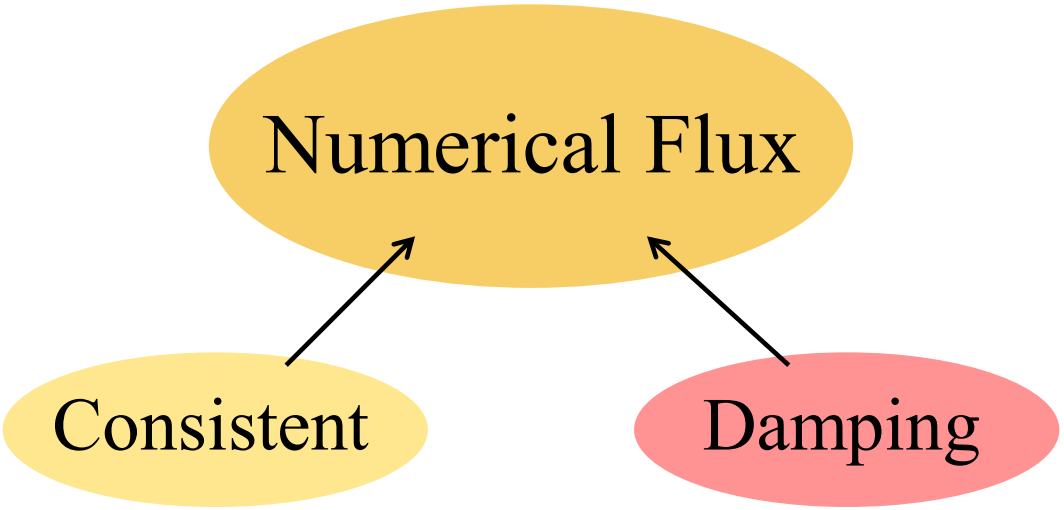


$$\mathbf{F}_{j+1/2} = \underbrace{\frac{1}{2}\left[\mathbf{F}_R + \mathbf{F}_L\right]}_{\textit{Consistent}} - \underbrace{\frac{1}{2}\left|\mathbf{A}\right|\left(\mathbf{U}_R - \mathbf{U}_L\right)}_{\textit{Dissipation}}$$

1. You cannot forget any of the two.
2. *Dissipation* characterizes the type of scheme.

Essential Ingredients for Diffusion

Consistent Term and Damping Term



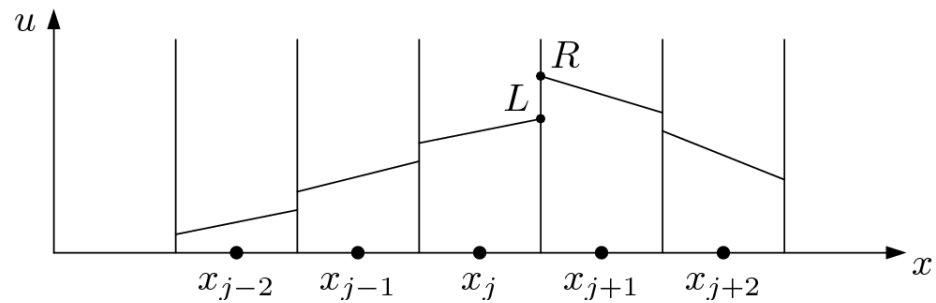
$$\mathbf{F}_{j+1/2} = \underbrace{\frac{1}{2}\left[\mathbf{F}_R + \mathbf{F}_L\right]}_{\textit{Consistent}} - \underbrace{\frac{1}{2}\left|\mathbf{B}\right|\left(\mathbf{U}_R - \mathbf{U}_L\right)}_{\textit{Damping}}$$

1. You cannot forget any of the two.
2. *Damping* characterizes the type of scheme.

A Typical Diffusion Scheme

Diffusion Equation:

$$u_t = (\nu u_x)_x$$



Diffusion Scheme (Nishikawa, AIAA2010, Computers&Fluids2011):

$$\frac{du_j}{dt} = \frac{1}{h}\left[f_{j+1/2} - f_{j-1/2}\right]$$

$$f_{j+1/2} = \frac{1}{2}\left[\nu(u_x)_L + \nu(u_x)_R\right] + \frac{\nu\alpha}{2h}(u_R - u_L)$$

Consistent *Damping*

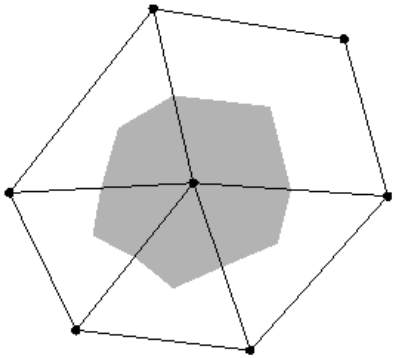
1. Central-difference scheme: α = 2 (Good)
2. 4th-order scheme: α = 8/3 (Recommended) – 6 DG, 3 SV

Damping is essential for accuracy, robustness, consistency.

Common Approach

Integral form for diffusion:

$$\int_\Omega u_t \;=\; \int_\Omega \nabla^2 u = \oint_{\partial\Omega} \nabla u \cdot \mathbf{n}\, dA$$



Compute the interface gradient.

Bad diffusion schemes lack high-frequency damping:

Large errors, poor convergence, lose consistency…….

Good diffusion schemes have a *damping* term (Ask experts!):

Finite-Volume: edge-term.

High-Order Schemes: penalty term or mixed formulation.

Residual-Distribution: ?

Do we really know how to make a diffusion scheme?

Hyperbolic Model for Diffusion

Diffusion Equation (Parabolic)

$$u_t = \nu\, u_{xx}$$

Hyperbolic Model for Diffusion

$$\begin{aligned} u_t &= \nu\, p_x \\ p_t &= (u_x - p)/T_r \end{aligned}$$

$$\lambda = \pm\sqrt{\frac{\nu}{T_r}}$$

Discretization
difficult

Discretization
easy

?

Diffusion Scheme

Advection Scheme

Can we derive a diffusion scheme from the advection scheme? Yes.

From Advection to Diffusion

Diffusion Equation (Parabolic)

$$u_t = \nu\, u_{xx}$$

Hyperbolic Model for Diffusion

$$\begin{aligned} u_t &= \nu\, p_x \\ p_t &= (u_x - p)/T_r \end{aligned}$$

Two models are equivalent if $p = u_x$

Derive a diffusion scheme from an advection scheme.

1. Discretize the hyperbolic system by an **advection** scheme:

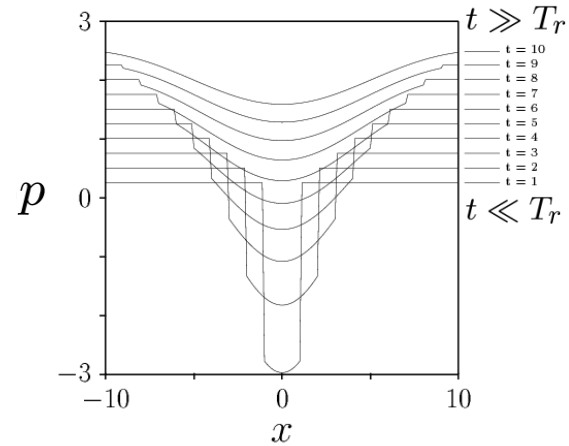
$$\frac{u_j^{n+1} - u_j^n}{\Delta t} = -Res_{j,1}^n, \qquad \frac{p_j^{n+1} - p_j^n}{\Delta t} = -Res_{j,2}^n - \frac{1}{T_r} p_j^n.$$

2. Replace the second equation by a direct approximation of $p = u_x$:

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} = -Res_{j,1}^n, \qquad p_j^n = \text{least-squares gradient, for example}$$

The result is a time-accurate **diffusion** scheme. (no need to store extra variables)

Relaxation Time



The CFL condition:

$$\Delta t \leq \frac{h}{\sqrt{\nu / T_r}}$$

Keep the hyperbolic behavior over every time step:

$$\Delta t_{\mathsf{max}} \equiv \frac{h}{\sqrt{\nu / T_r}} = \alpha T_r$$

Solve for Tr:

$$T_r = \frac{h^2}{\alpha^2 \nu}$$

The CFL condition becomes

$$\Delta t \leq \frac{1}{\alpha} \frac{h^2}{\nu}$$

General Form: $$T_r = \frac{L_r^2}{\alpha^2 \nu}, \qquad L_r = h$$

Recipe for Making Good Diffusion Schemes

$$u_t = \nu\, u_{xx}$$

1. Discretize the hyperbolic system by an advection scheme.

$$\mathbf{U}_t + \mathbf{F}_x = \mathbf{Q}$$

$$\mathbf{U} = \begin{bmatrix} u \\ p \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} -\nu p \\ -u/T_r \end{bmatrix}, \quad \mathbf{Q} = \begin{bmatrix} 0 \\ -p/T_r \end{bmatrix}.$$

2. Ignore the discrete equation for p, and approximate $p = u_x$ directly.

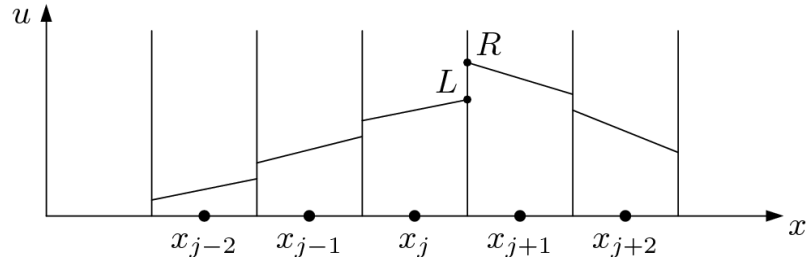
3. The result is a time-accurate diffusion scheme, **a good one!**

Derived diffusion scheme:

1. Automatically equipped with a damping term (edge/penalty term)
2. Implemented in the same way as a corresponding advection scheme.

This recipe is applicable to various discretization methods.

1D Finite-Volume Scheme



1. Discretize the hyperbolic system:

Finite-Volume Method:

$$\frac{d\mathbf{U}_j}{dt} = -\frac{1}{\Delta x}\left[\mathbf{F}_{j+1/2} - \mathbf{F}_{j-1/2}\right] + \frac{1}{\Delta x}\int_{I_j} \mathbf{Q}\, dx$$

$$\mathbf{U} = \begin{bmatrix} u \\ p \end{bmatrix} \qquad \mathbf{F} = \begin{bmatrix} -\nu p \\ -u/T_r \end{bmatrix} \qquad \mathbf{Q} = \begin{bmatrix} 0 \\ -p/T_r \end{bmatrix}.$$

Upwind Flux:

$$|\mathbf{A}| = \sqrt{\nu/T_r} = \alpha\nu/h$$

$$\mathbf{F}_{j+1/2} = \frac{1}{2}\left[\mathbf{F}_R + \mathbf{F}_L\right] - \frac{1}{2}\frac{\nu\alpha}{h}(\mathbf{U}_R - \mathbf{U}_L)$$

2. Discard the second component to get a time-accurate diffusion scheme:

$$\frac{du_j}{dt} = -\frac{1}{\Delta x}\left[f_{j+1/2} - f_{j-1/2}\right]$$

$$f_{j+1/2} = -\frac{\nu}{2}\left[p_R + p_L\right] - \frac{\nu\alpha}{2\Delta x}(u_R - u_L)$$

Consistent part **Damping = O(h^2) quantity**

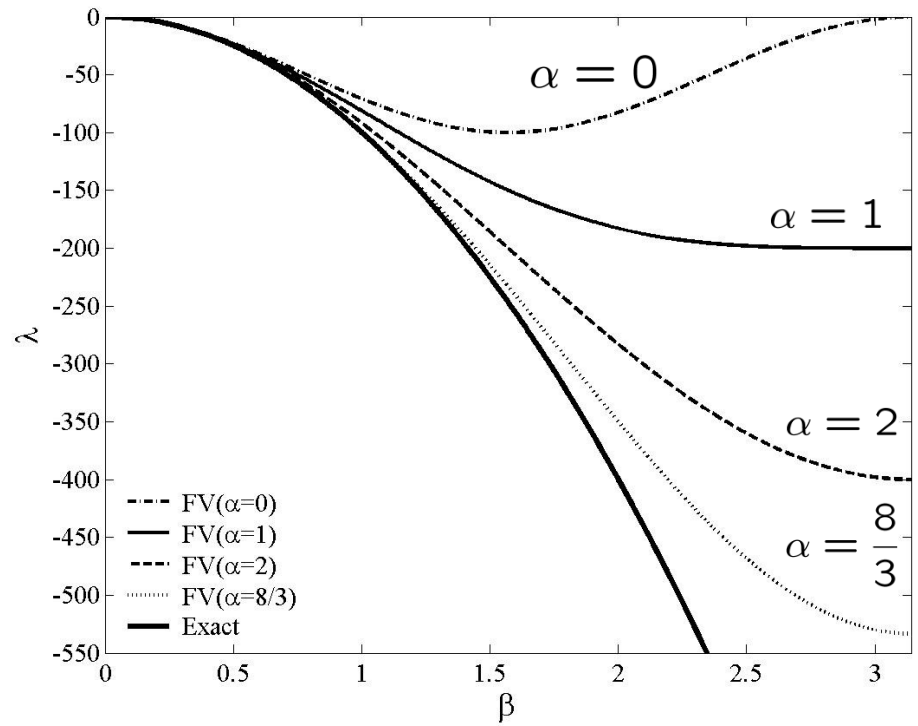
The damping term comes from the dissipation of the advection scheme.

Effect of Damping Term

Fourier transformed: $u_0 \exp(i\beta x/\Delta x)$

$$\frac{du_0}{dt} = -\frac{\nu}{\Delta x^2}\left(\sin^2\beta + 2\alpha \sin^4\frac{\beta}{2}\right) u_0$$

Consistent part **Damping**



Truncation Error:

$$\frac{du_j}{dt} = \nu u_{xx} + \nu u_{xxxx}\left(\frac{1}{3} - \frac{\alpha}{8}\right)\Delta x^2 + O(\Delta x^4)$$

The parameter α **controls damping: 4th-order accurate for** $\alpha = 8/3$ **.**

More 1D Examples

More 1D examples can be found in Computers&Fluid2011 and AIAA2010-5093:

Residual-distribution (equivalent to finite-volume)
Discontinuous Galerkin (2nd-order P2, 4th-order P2)
Spectral-volume schemes (2nd-order P2, 4th-order P2)

Papers (PDF) can be downloaded at http://www.hiroakinishikawa.com

Two Dimensions

$$u_t = \nu\left(u_{xx} + u_{yy}\right)$$

Hyperbolic Model:

$$\mathbf{U}_t + \mathbf{F}_x + \mathbf{G}_y = \mathbf{Q}$$

$$\mathbf{U} = \begin{bmatrix} u \\ p \\ q \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} -\nu p \\ -u/T_r \\ 0 \end{bmatrix}, \quad \mathbf{G} = \begin{bmatrix} -\nu q \\ 0 \\ -u/T_r \end{bmatrix}, \quad \mathbf{Q} = \begin{bmatrix} 0 \\ -p/T_r \\ -q/T_r \end{bmatrix}.$$

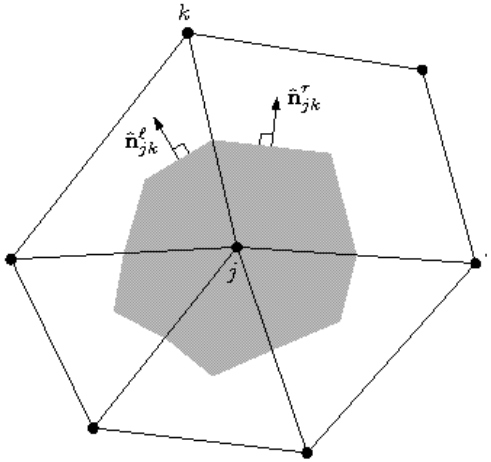
Equivalent to the diffusion equation when $p = u_x, q = u_y$

Absolute Jacobian:

$$|\mathbf{A}_n| = \mathbf{R}_n|\mathbf{\Lambda}_n|\mathbf{R}_n^{-1} = \sqrt{\frac{\nu}{T_r}} \begin{bmatrix} 1 & 0 & 0 \\ 0 & n_x^2 & n_x n_y \\ 0 & n_x n_y & n_y^2 \end{bmatrix}.$$

The first component is all we need.

Node-Centered FV Schemes



1. Discretize hyperbolic system: Edge-based advection scheme:

$$\frac{d\mathbf{U}_j}{dt} = -\frac{1}{V_j}\sum_{k\in\{K_j\}} \mathbf{\Phi}_{jk} A_{jk} + \frac{1}{V_j}\int_{\Omega_j} \mathbf{Q}\, dxdy$$

$$\mathbf{\Phi}_{jk} = \frac{1}{2}\left[\mathbf{H}_{jk}(\mathbf{U}_R) + \mathbf{H}_{jk}(\mathbf{U}_L)\right] - \frac{1}{2}|\mathbf{A}_n|(\mathbf{U}_R - \mathbf{U}_L)$$

2. Discard the second and third components to get a diffusion scheme:

$$\frac{du_j}{dt} = -\frac{1}{V_j}\sum_{k\in\{K_j\}} \phi_{jk} A_{jk}$$

Diffusive flux: $$\phi_{jk} = -\frac{\nu}{2}\left[(p,q)_R + (p,q)_L\right]\cdot\hat{\mathbf{n}}_{jk} - \frac{\nu\alpha}{2L_r}(u_R - u_L)$$

$$= -\frac{\nu}{2}\left[(\nabla u)_j + (\nabla u)_k\right]\cdot\hat{\mathbf{n}}_{jk} - \frac{\nu\alpha}{2L_r}(u_R - u_L)$$

Consistent part **Damping**

Widely-used Avg-LSQ scheme can be reproduced by special choice of alpha.

The Green-Gauss scheme corresponds to using the Green-Gauss gradient for (p,q).

See AIAA2010-5093/CF2011 for details and more examples.

Length Scale and Skewness

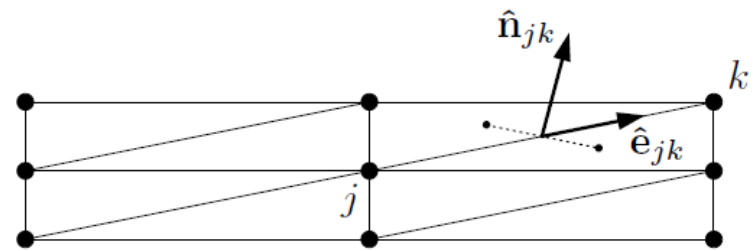
Derived diffusion scheme:

$$\phi_{jk} = -\frac{\nu}{2}\left[(\nabla u)_j + (\nabla u)_k\right]\cdot\hat{\mathbf{n}}_{jk} - \frac{\nu\alpha}{2L_r}(u_R - u_L)$$

Length scale is defined as

$$L_r = \frac{1}{2}|\Delta\mathbf{l}_{jk}\cdot\hat{\mathbf{n}}_{jk}| = \frac{1}{2}\Delta l_{jk}|\hat{\mathbf{e}}_{jk}\cdot\hat{\mathbf{n}}_{jk}|$$

Skewness measure



Damping is amplified for highly-skewed grids.

Widely-used edge-normal Avg-LSQ scheme (Ave-LSQ-EN):

$$\phi_{jk} = -\frac{\nu}{2}\left[(\nabla u)_j + (\nabla u)_k\right]\cdot\hat{\mathbf{n}}_{jk} - \left(\hat{\mathbf{e}}_{jk}\cdot\hat{\mathbf{n}}_{jk}\right)(u_R - u_L)$$

It loses damping for highly-skewed grids (which can be avoided by mixed-grid).

The derived diffusion scheme is very accurate and robust for highly-skewed grids.

Residual-Distribution Schemes

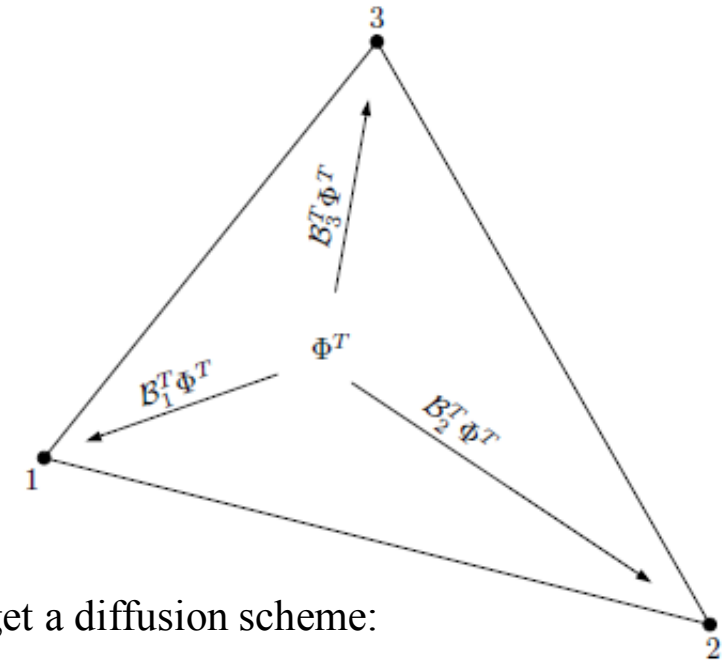
- **Lax-Wendroff Scheme:**

1. Discretize the hyperbolic system:

$$\frac{d\mathbf{U}_j}{dt} = \frac{1}{V_j} \sum_{T \in \{T_j\}} \mathcal{B}_j^T \Phi^T,$$

$$\Phi^T = \int_T (-\mathbf{A}\mathbf{U}_x - \mathbf{B}\mathbf{U}_y + \mathbf{Q})\, dxdy$$

$$\mathcal{B}_i^T = \frac{1}{3}\mathbf{I} + \frac{h}{\sqrt{\nu/T_r}} (\mathbf{A}, \mathbf{B}) \cdot \mathbf{n}_i$$



2. Discard the second and third components to get a diffusion scheme:

$$\frac{du_j}{dt} = \frac{1}{V_j} \sum_{T \in \{T_j\}} \left[\frac{1}{3}\phi^T - \frac{\nu\alpha}{2} \left\{ \nabla u^T - (\overline{p}_T, \overline{q}_T) \right\} \cdot \mathbf{n}_j^T \right]$$

Damping term

This becomes the Galerkin scheme for $\alpha = 2$ (See Nishikawa JCP2007)

- **LDA Scheme (Upwind):** Very accurate scheme. See AIAA2010 or CF2011.

We now have good diffusion schemes for residual-distribution method!

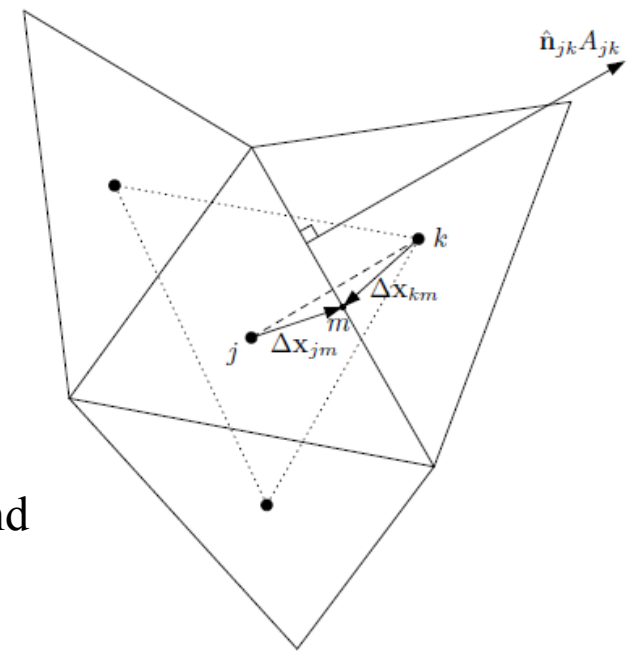
Cell-Centered FV Schemes

Derived diffusion scheme:

$$\frac{du_j}{dt} = -\frac{1}{V_j} \sum_{k \in \{K_j\}} \phi_{jk} A_{jk}$$

$$\phi_{jk} = \underbrace{-\frac{\nu}{2}\left[(p,q)_R + (p,q)_L\right] \cdot \hat{\mathbf{n}}_{jk}}_{\textbf{Consistent part}} - \underbrace{\frac{\nu\alpha}{2L_r}(u_R - u_L)}_{\textbf{Damping}}$$

$$L_r = \frac{1}{2}|\Delta \mathbf{l}_{jk} \cdot \hat{\mathbf{n}}_{jk}| = \frac{1}{2}\Delta l_{jk}|\hat{\mathbf{e}}_{jk} \cdot \hat{\mathbf{n}}_{jk}|$$



Left and right states: reconstruct gradients in cells, and extrapolate at the **face-midpoint**:

$$u_L = u_j + \frac{1}{2}(\nabla u)_j \cdot \Delta\mathbf{x}_{jm}, \qquad u_R = u_k - \frac{1}{2}(\nabla u)_k \cdot \Delta\mathbf{x}_{km}$$

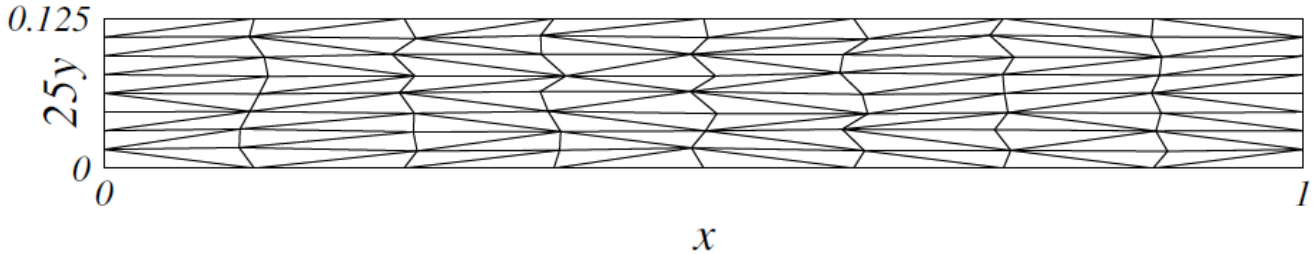
$$(p,q)_L = (\nabla u)_j, \qquad (p,q)_R = (\nabla u)_k,$$

The diffusive flux is evaluated precisely at the quadrature point.

It can be easily applied to high-order methods (See AIAA2010-5093/CF2011 for DG/SV) .

(Widely-used AvgLSQ scheme evaluate the flux at "**edge-midpoint**".)

Test Problem – Highly-Skewed Grids



Problem:

$$u_t = \nu(u_{xx} + u_{yy})$$

$$u(x, y, 0) \;=\; 5\sin(\pi x)\sin(4000\pi y)$$

Compute the solution at t =1.0e-08

Irregular Grids: 25x25, 33x33, …, 129x129, 137x137 (15 grids)

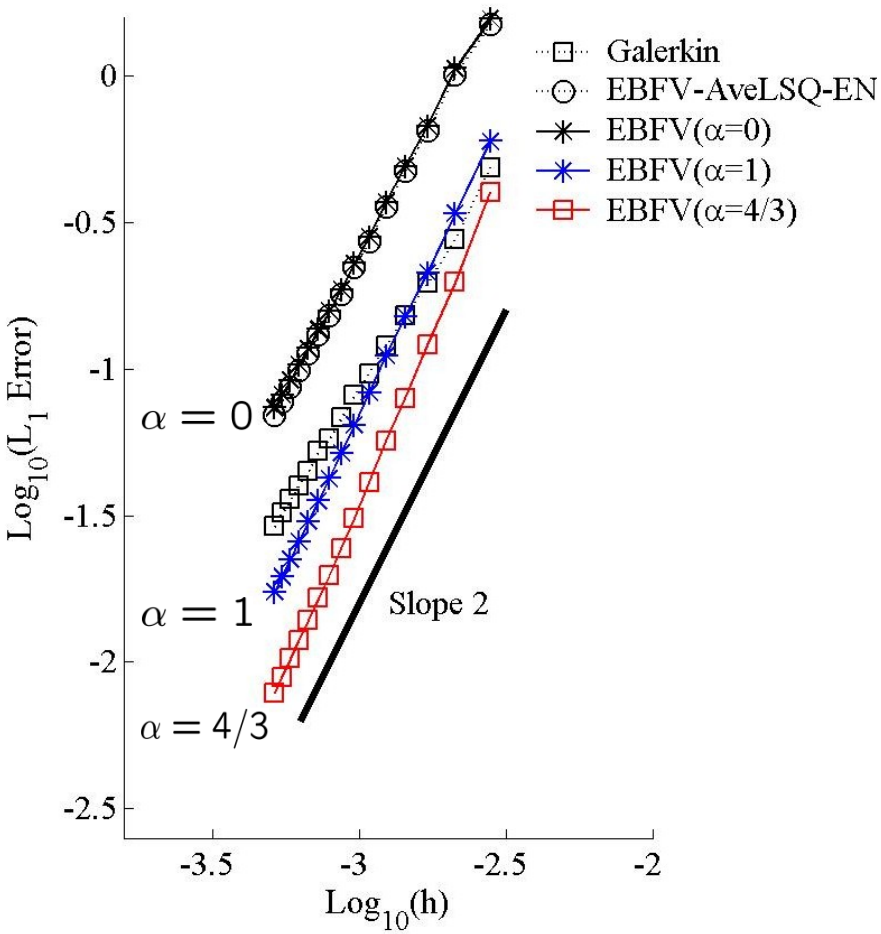
Global time step (the forward-Euler explicit):

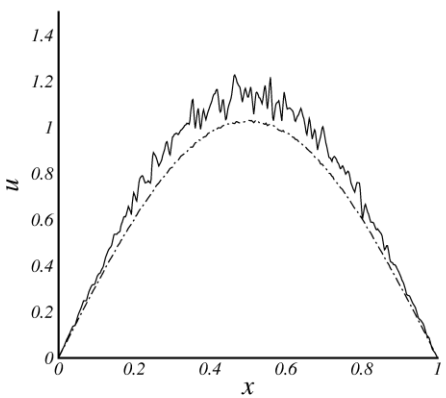
$$\Delta t = 0.003\, h^2$$

Total time steps = 77, 137, 214, 308, 419, 547, 692, 854, 1033, 1229, 1443, 1673, 1921, 2185, 2467.

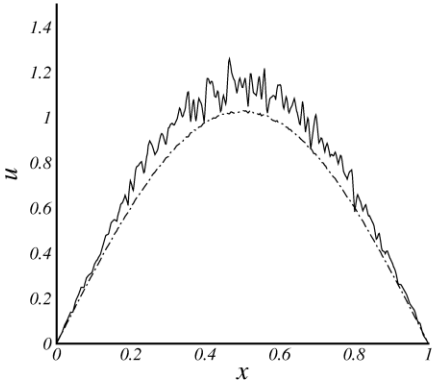
Compute the error at the data point at the final time.

Results for EBFV Schemes

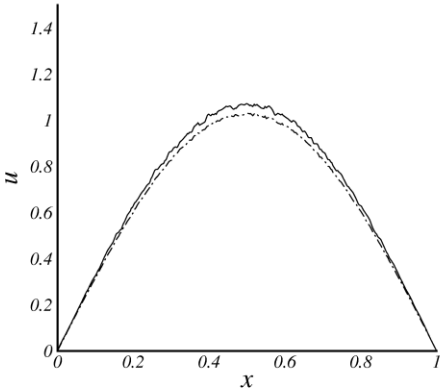




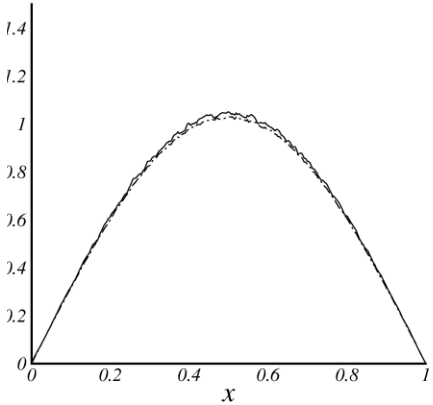
Avg-LSQ-EN



EBS(α=0)



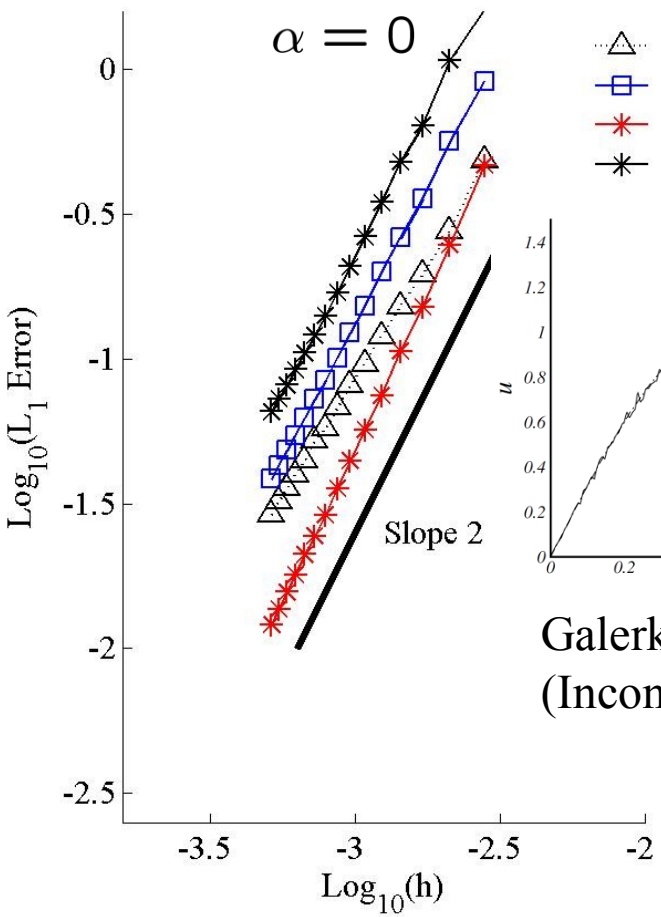
EBS(α=1)

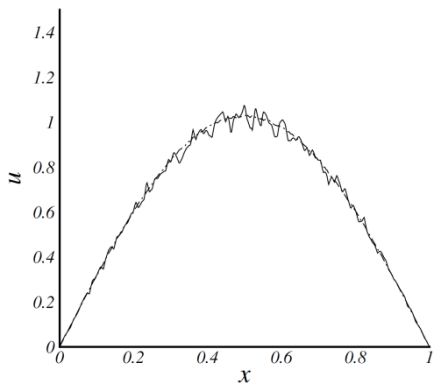


EBS(α=4/3)

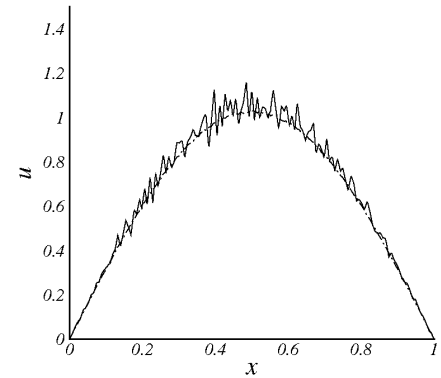
Lack of damping → Large error, Oscillations

Results for RD Schemes

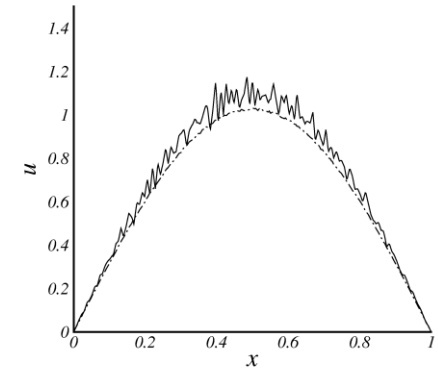




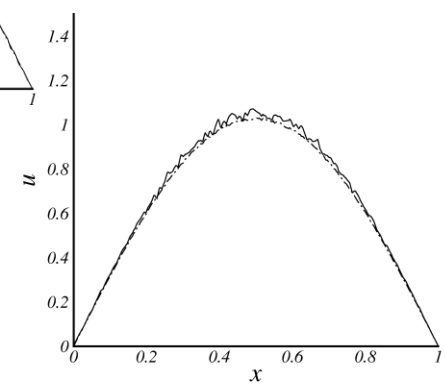
Galerkin Positive (Inconsistent)



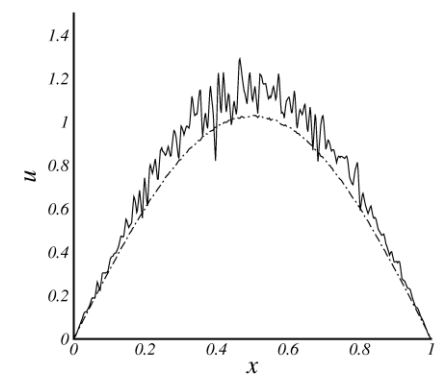
Galerkin



LW (α=1)



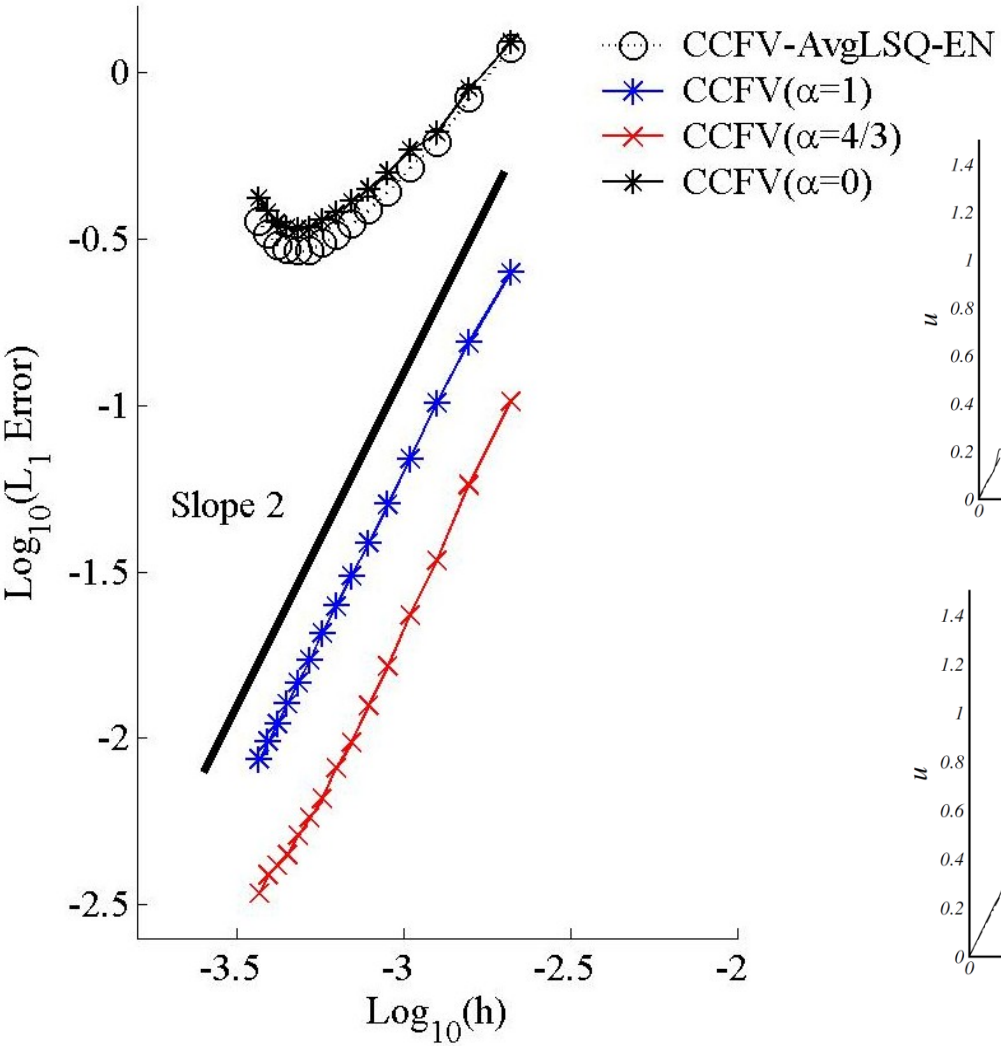
LDA (α=1)

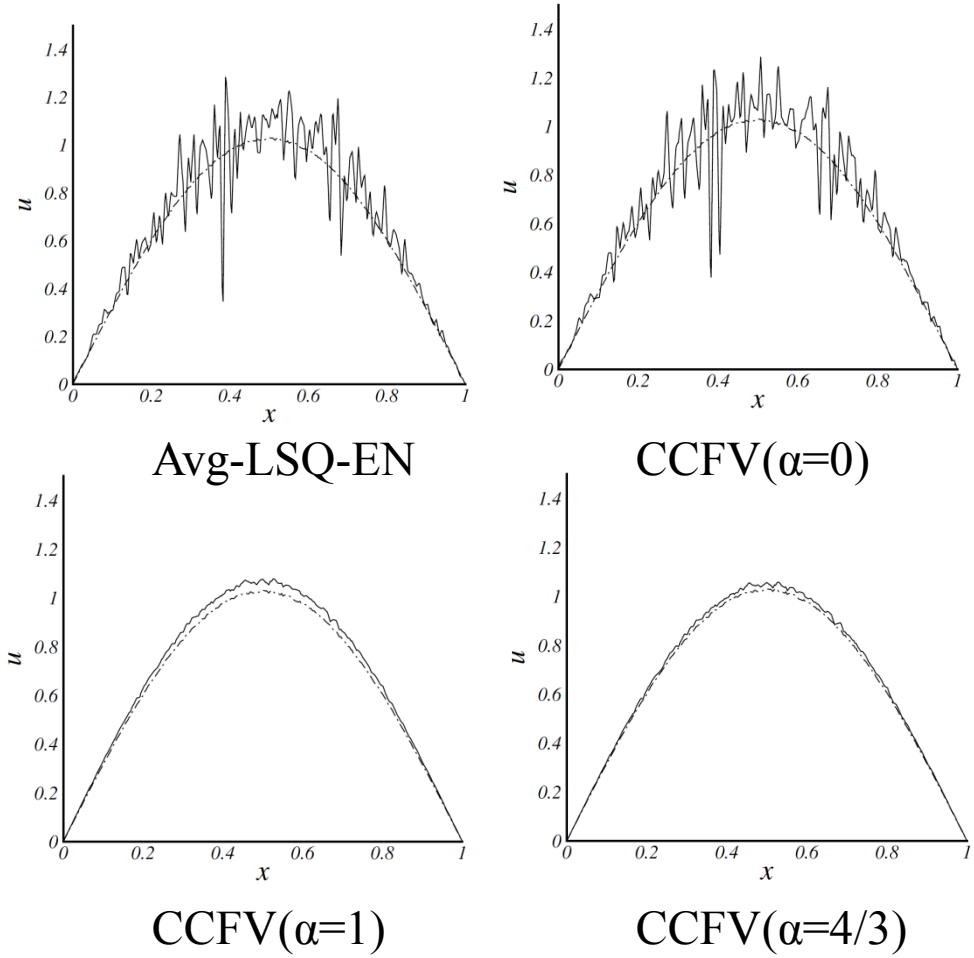


LDA (α=0)

Lack of damping → Large error, Oscillations

Results for CCFV Schemes





Lack of damping → Large error, Oscillations

More 2D Examples

More 2D examples can be found in Computers&Fluid2011 and AIAA2010-5093:

Compact Finite-volume
Element-Based Finite-Volume
Discontinuous Galerkin
Spectral-volume schemes

Papers (PDF) can be downloaded at http://www.hiroakinishikawa.com

Hyperbolic Recipe for Diffusion

Diffusion Equation (Parabolic)

$$u_t = \nu\, u_{xx}$$

Hyperbolic Model for Diffusion

$$\begin{aligned} u_t &= \nu\, p_x \\ p_t &= (u_x - p)/T_r \end{aligned}$$

? ? ?

? ? ?

Discretization

easy

$$\lambda = \pm\sqrt{\frac{\nu}{T_r}}$$

$$\frac{1}{2}\left[\nu(u_x)_R + \nu(u_x)_L\right] + \frac{\nu\alpha}{2h}(u_R - u_L)$$

Time accurate diffusion scheme

$$\frac{1}{2}\left[\mathbf{F}_R + \mathbf{F}_L\right] - \frac{1}{2}\left|\mathbf{A}\right|\left(\mathbf{U}_R - \mathbf{U}_L\right)$$

Not time-accurate for diffusion

Damping term comes directly from dissipation term.

"How can we extend this to the viscous term in Navier-Stokes?"

There are two ways to do it.

Two Ways to Viscous Discretization

***Gradient Formula**:*

Identify the gradient formula in the diffusion scheme, and directly evaluate the viscous flux.

***Hyperbolic Recipe**:*

Discretize a hyperbolic viscous system and ignore the extra equations.

Good gradient formula must be available.

Hyperbolic system must be available for the viscous term.

Viscous Discretization

Damping mechanism must be incorporated.

Viscous Term in NS System

Viscous Part of the Navier-Stokes Equations

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}^{vis}}{\partial x} = 0 \qquad \mathbf{U} = \begin{bmatrix} \rho \\ \rho u \\ \rho E \end{bmatrix}, \quad \mathbf{F}^{vis} = \begin{bmatrix} 0 \\ -\tau \\ -\tau u + q \end{bmatrix}$$

Viscous stress and heat flux:

$$\tau = \mu_v \frac{\partial u}{\partial x}, \quad q = -\frac{\mu_h}{\gamma(\gamma - 1)} \frac{\partial T}{\partial x}. \quad \left(\mu_v = \frac{4}{3}\mu, \quad \mu_h = \frac{\gamma \mu}{Pr}.\right)$$

Typically, we seek the gradients at the interface: $\frac{\partial u}{\partial x} \quad \frac{\partial T}{\partial x}$

1. Extension by Gradient Formula

The diffusive flux,

$$f_{j+1/2} = \frac{1}{2}\left[\nu(u_x)_L + \nu(u_x)_R\right] + \frac{\nu\alpha}{2h}(u_R - u_L)$$

can be written as:

$$f_{j+1/2} = \nu\, u_x|_{j+1/2}$$

$$\boxed{u_x|_{j+1/2} = \frac{1}{2}\left[(u_x)_L + (u_x)_R\right] + \frac{\alpha}{2h}(u_R - u_L)}$$

Interface Gradient Formula for $\frac{\partial u}{\partial x}$ $\frac{\partial T}{\partial x}$

Compute the physical viscous flux using this formula.

Gradient-based viscous flux

2. Extension by Hyperbolic Recipe

Hyperbolic Viscous System [Nishikawa, AIAA2011-3043]:

$$\mathbf{P}^{-1}\frac{\partial \mathbf{V}}{\partial t} + \frac{\partial \mathbf{F}^v}{\partial x} = \mathbf{S}$$

$$\mathbf{P}^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & T_v/\mu_v & 0 \\ 0 & 0 & 0 & 0 & T_h/\mu_h \end{bmatrix}, \mathbf{V} = \begin{bmatrix} \rho \\ \rho u \\ \rho E \\ \tau \\ q \end{bmatrix}, \mathbf{F}^v = \begin{bmatrix} 0 \\ -\tau \\ -\tau u + q \\ -u \\ \dfrac{a^2}{\gamma(\gamma-1)} \end{bmatrix}, \mathbf{S} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \tau/\mu_v \\ q/\mu_h \end{bmatrix}$$

Relaxation times: $T_v = \dfrac{h^2}{\alpha^2 \nu_v}, \quad T_h = \dfrac{h^2}{\alpha^2 \nu_h},$ [AIAA2010,CF20111]:

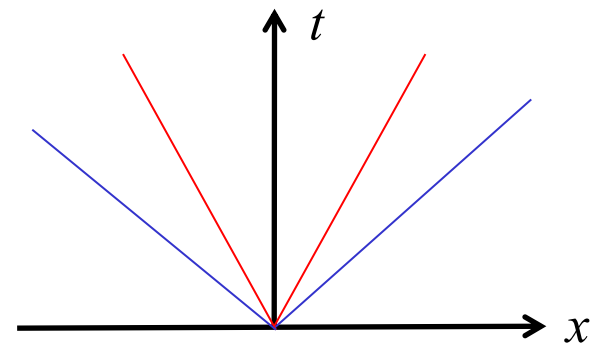
Relaxation times are O(h^2), not O(1).

Hyperbolic Viscous System

Eigenvalues of the viscous system:

$$\lambda = \pm a_v, \quad \pm a_h, \quad 0$$

Viscous and heating waves



Viscous and Heating Wave Speeds:

$$a_v = \sqrt{\frac{\nu_v}{T_v}} = \alpha \frac{\nu_v}{h}, \qquad a_h = \sqrt{\frac{\nu_h}{T_h}} = \alpha \frac{\nu_h}{h}.$$

The corresponding eigenvectors are linearly independent (AIAA2011-3044).

Construct an upwind flux, ignore extras to get viscous flux.

Many choices are available: FVS, Riemann solvers, Multi-D upwinding, etc.

Derived Viscous Flux

Derived from the upwind flux proposed in Nishikawa AIAA2011-3043

$$\mathbf{F}^{vis}_{j+1/2} = \underbrace{\frac{1}{2}\left[\mathbf{F}^{vis}(\mathbf{U}_L, \nabla\mathbf{U}_L) + \mathbf{F}^{vis}(\mathbf{U}_R, \nabla\mathbf{U}_R)\right]}_{\textit{Consistent}} - \underbrace{\frac{1}{2}\Delta\mathbf{F}^{vis}}_{\textit{Damping} = O(h^2)}$$

The damping term is given by

$$\Delta\mathbf{F}^{vis} = \begin{bmatrix} 0 \\ \rho a_v \Delta u \\ \rho a_v u \Delta u + \dfrac{\rho a_h \Delta T}{\gamma(\gamma-1)} + \dfrac{\tau\Delta\tau}{\rho(a_v + a_h)} \end{bmatrix}$$

Damping term comes directly from the dissipation term of the upwind flux.

Looks similar to inviscid flux? Yes.

Full Navier-Stokes Flux

$$\mathbf{F}_{j+1/2} = \frac{1}{2}[\mathbf{F}(\mathbf{U}_L, \nabla\mathbf{U}_L) + \mathbf{F}(\mathbf{U}_R, \nabla\mathbf{U}_R)] - \frac{1}{2}(\Delta\mathbf{F}^{inv} + \Delta\mathbf{F}^{vis})$$

where

$\Delta\mathbf{F}^{inv}$: *Dissipation* term of the inviscid flux.

$\Delta\mathbf{F}^{vis}$: *Damping* term of the viscous flux.

- *Loop over the faces,*
- *Get the left/right states,* $(\mathbf{U}_L, \nabla\mathbf{U}_L), (\mathbf{U}_R, \nabla\mathbf{U}_R)$
- *Compute the numerical flux at quadrature pts.*

Explicit Time Step

The CFL condition for Hyperbolic Navier-Stokes Scheme:

$$\Delta t = \mathsf{CFL}\frac{h}{\max(|u| + a + a_h)_j}$$

For the derived viscous flux:

$$a_h = \sqrt{\frac{\nu_h}{T_h}} = \alpha\frac{\nu_h}{h}.$$

$$\Delta t = \mathsf{CFL}\frac{h}{\max(|u| + a + \alpha\nu_h/h)_j}$$

The CFL condition is inherited from upwind scheme.

1D Numerical Results

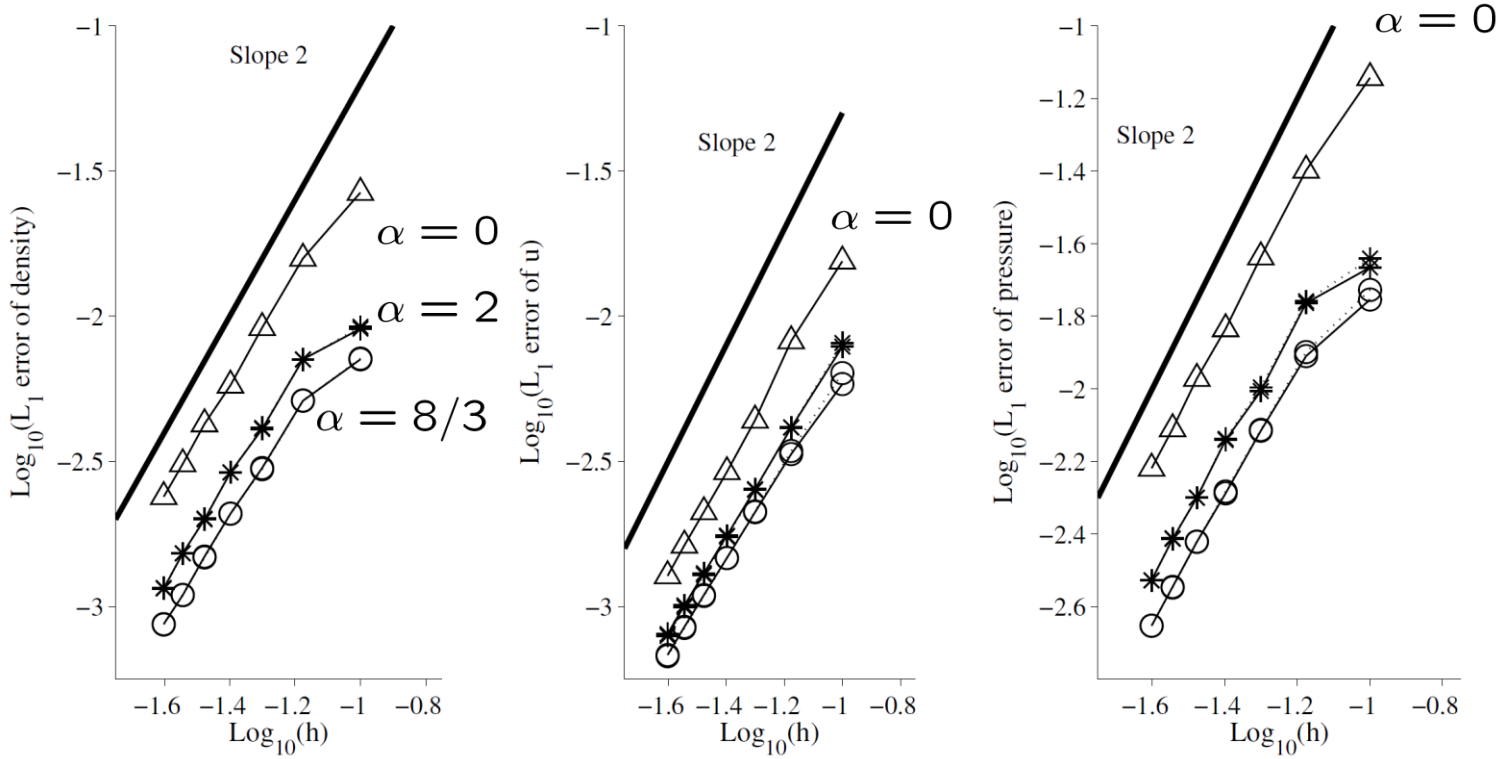
1D viscous shock structure problem on uniform grids: 20-80.

M_∞=3.5, Pr=$\frac{3}{4}$, γ=1.4, Re_∞=25, T_∞=400[K].

Steady state is reached at residual reduction by 6 orders.

——— Gradient-based viscous flux $*: \alpha = 2$ $\circ: \alpha = 8/3$ $\triangle: \alpha = 0$

····· Derived viscous flux $*: \alpha = 2$ $\circ: \alpha = 8/3$



No-damping scheme gives significantly larger error.

2D Numerical Results

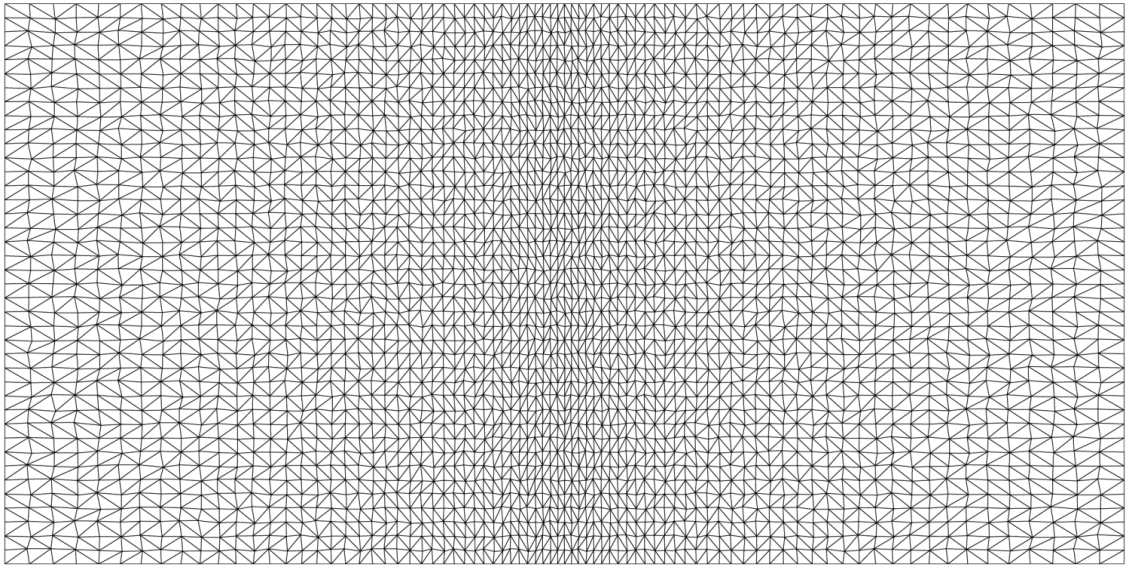
Node-centered FV scheme:

Inviscid flux: Roe flux

Viscous flux: 1. Gradient-based viscous flux

2. Derived viscous flux

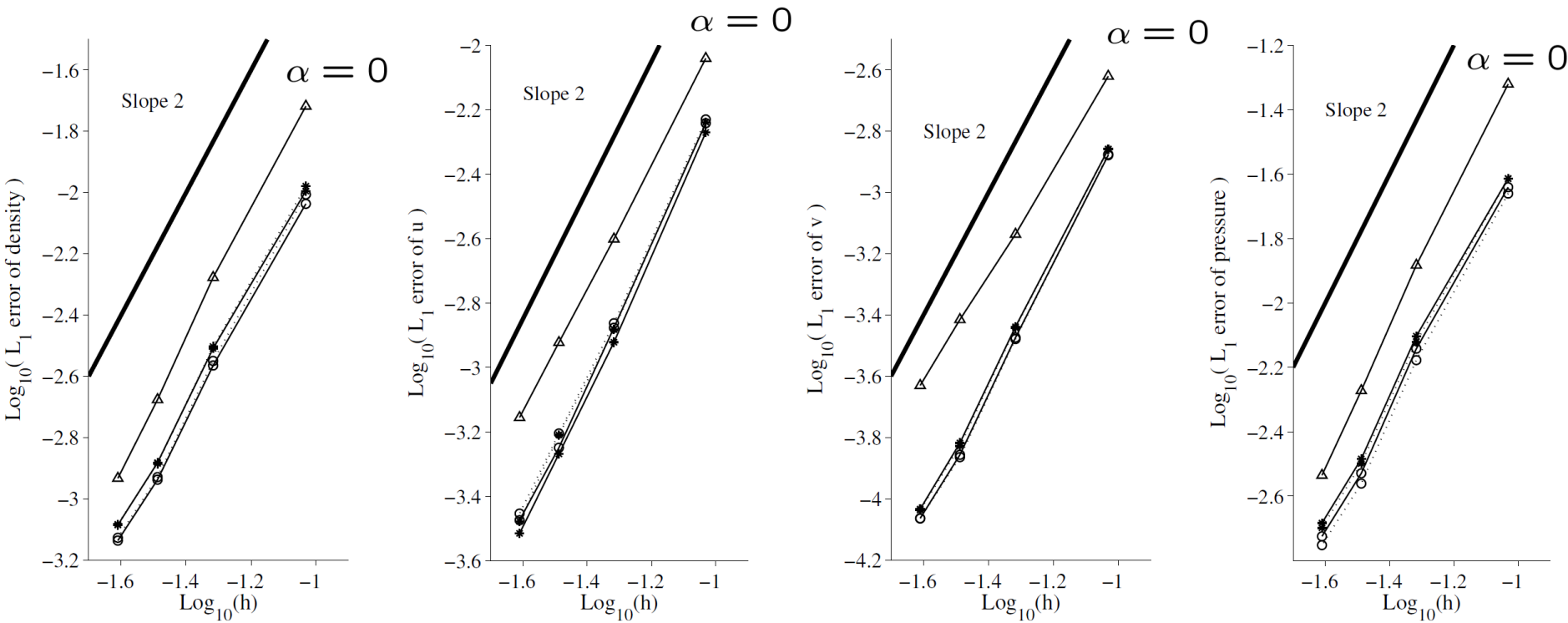
Irregular triangular grids: 231, 861, 1891, 3321 nodes.



Error Convergence (2D)

Gradient-based viscous flux $*: \alpha = 1$ $\circ: \alpha = 4/3$ $\triangle: \alpha = 0$

Derived viscous flux $*: \alpha = 1$ $\circ: \alpha = 4/3$



No-damping scheme gives significantly larger error.

Conclusions

Two Essential Ingredients – ***Consistent and damping terms.***

Hyperbolic Recipe – ***Derive viscous scheme from inviscid scheme.***

Importance of damping term demonstrated:

Lack of damping leads to inaccurate solutions.
Maybe, we should worry about damping rather than positivity? (See AIAA2010-5093)

Hyperbolic Recipe is simple and useful:

Damping term is ***automatically*** introduced in the hyperbolic recipe.
Various robust and accurate viscous discretization can be easily constructed.

Two ways from diffusion to Navier-Stokes:

Gradient Formula – Gradient-based Viscous Flux
Upwind Flux – Derived Viscous Flux

Now you can make a good tasty viscous scheme at home!

Future Work

Highly-skewed grids for NS computations – *Damping term is critical (TBD).*

Optimal value of α – *Accuracy, smoothing, multi-dimensional analysis*

Greater Variety of Viscous Fluxes –*FVS, Riemann solvers, multi-D upwinding…..*

Other Discretization Methods – *FV, FE, CESE, RD, DG, SV, SD…..*

Higher-Order PDEs – *Construct a hyperbolic system, discretize it, and extract a scheme.*

Grab your favorite inviscid scheme, and derive a viscous scheme.

Now everybody can be an expert in viscous discretizaion!

Download papers, slides, and codes at http://www.hiroakinishikawa.com