Annual Thesis Monitoring Committee (TMC) meeting

Lax-Wendroff Flux Reconstruction : Blending limiters and relation to ADER

September 26, 2022

Arpit Babbar

- Review of Lax-Wendroff Flux Reconstruction with D2 dissipation numerical flux.
- Introduction to the blending limiter of [8] in context of Lax-Wendroff
- Extending Zhang-Shu's limiter [21] to Lax-Wendroff schemes.
- Admissibility preserving MUSCL-Hancock reconstruction on non-cell centred grids used by [8].
- Prove equivalence of ADER and LW-D2 schemes with numerical verification
- Numerical results in 1-D, 2-D demonstrating admissibility preservation and accuracy improvement of limiting procedure.
- Code optimization in `Julia`
- Summary and reading activities of the year

- Review of Lax-Wendroff Flux Reconstruction with D2 dissipation numerical flux.
- Introduction to the blending limiter of [8] in context of Lax-Wendroff
- Extending Zhang-Shu's limiter [21] to Lax-Wendroff schemes.
- Admissibility preserving MUSCL-Hancock reconstruction on non-cell centred grids used by [8].
- Prove equivalence of ADER and LW-D2 schemes with numerical verification
- Numerical results in 1-D, 2-D demonstrating admissibility preservation and accuracy improvement of limiting procedure.
- Code optimization in `Julia`
- Summary and reading activities of the year

- Review of Lax-Wendroff Flux Reconstruction with D2 dissipation numerical flux.
- Introduction to the blending limiter of [8] in context of Lax-Wendroff
- Extending Zhang-Shu's limiter [21] to Lax-Wendroff schemes.
- Admissibility preserving MUSCL-Hancock reconstruction on non-cell centred grids used by [8].
- Prove equivalence of ADER and LW-D2 schemes with numerical verification
- Numerical results in 1-D, 2-D demonstrating admissibility preservation and accuracy improvement of limiting procedure.
- Code optimization in `Julia`
- Summary and reading activities of the year

- Review of Lax-Wendroff Flux Reconstruction with D2 dissipation numerical flux.
- Introduction to the blending limiter of [8] in context of Lax-Wendroff
- Extending Zhang-Shu's limiter [21] to Lax-Wendroff schemes.
- Admissibility preserving MUSCL-Hancock reconstruction on non-cell centred grids used by [8].
- Prove equivalence of ADER and LW-D2 schemes with numerical verification
- Numerical results in 1-D, 2-D demonstrating admissibility preservation and accuracy improvement of limiting procedure.
- Code optimization in `Julia`
- Summary and reading activities of the year

- Review of Lax-Wendroff Flux Reconstruction with D2 dissipation numerical flux.
- Introduction to the blending limiter of [8] in context of Lax-Wendroff
- Extending Zhang-Shu's limiter [21] to Lax-Wendroff schemes.
- Admissibility preserving MUSCL-Hancock reconstruction on non-cell centred grids used by [8].
- Prove equivalence of ADER and LW-D2 schemes with numerical verification
- Numerical results in 1-D, 2-D demonstrating admissibility preservation and accuracy improvement of limiting procedure.
- Code optimization in `Julia`
- Summary and reading activities of the year

- Review of Lax-Wendroff Flux Reconstruction with D2 dissipation numerical flux.
- Introduction to the blending limiter of [8] in context of Lax-Wendroff
- Extending Zhang-Shu's limiter [21] to Lax-Wendroff schemes.
- Admissibility preserving MUSCL-Hancock reconstruction on non-cell centred grids used by [8].
- Prove equivalence of ADER and LW-D2 schemes with numerical verification
- Numerical results in 1-D, 2-D demonstrating admissibility preservation and accuracy improvement of limiting procedure.
- Code optimization in `Julia`
- Summary and reading activities of the year

- Review of Lax-Wendroff Flux Reconstruction with D2 dissipation numerical flux.
- Introduction to the blending limiter of [8] in context of Lax-Wendroff
- Extending Zhang-Shu's limiter [21] to Lax-Wendroff schemes.
- Admissibility preserving MUSCL-Hancock reconstruction on non-cell centred grids used by [8].
- Prove equivalence of ADER and LW-D2 schemes with numerical verification
- Numerical results in 1-D, 2-D demonstrating admissibility preservation and accuracy improvement of limiting procedure.
- Code optimization in `Julia`
- Summary and reading activities of the year

- Review of Lax-Wendroff Flux Reconstruction with D2 dissipation numerical flux.
- Introduction to the blending limiter of [8] in context of Lax-Wendroff
- Extending Zhang-Shu's limiter [21] to Lax-Wendroff schemes.
- Admissibility preserving MUSCL-Hancock reconstruction on non-cell centred grids used by [8].
- Prove equivalence of ADER and LW-D2 schemes with numerical verification
- Numerical results in 1-D, 2-D demonstrating admissibility preservation and accuracy improvement of limiting procedure.
- Code optimization in `Julia`
- Summary and reading activities of the year

$$\boldsymbol{u}_t + \boldsymbol{f}(\boldsymbol{u})_x = 0$$

$$\boldsymbol{u}_t + \boldsymbol{f}(\boldsymbol{u})_x = 0$$

Degree N approximation

$$\Omega = \bigcup_e I_e$$

Flux Reconstruction (FR) by Huynh [9]

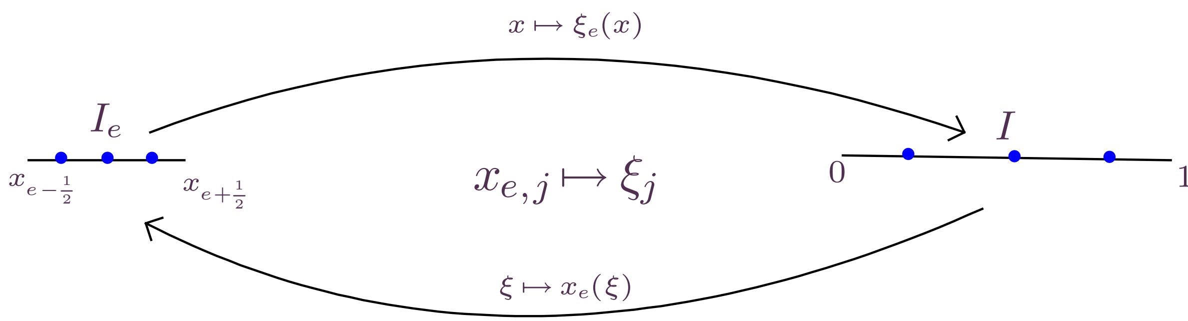
$$\boldsymbol{u}_t + \boldsymbol{f}(\boldsymbol{u})_x = 0$$

Degree N approximation

$$\Omega = \bigcup_e I_e$$

$u_h = u_h(x)$

$r = r(x)$

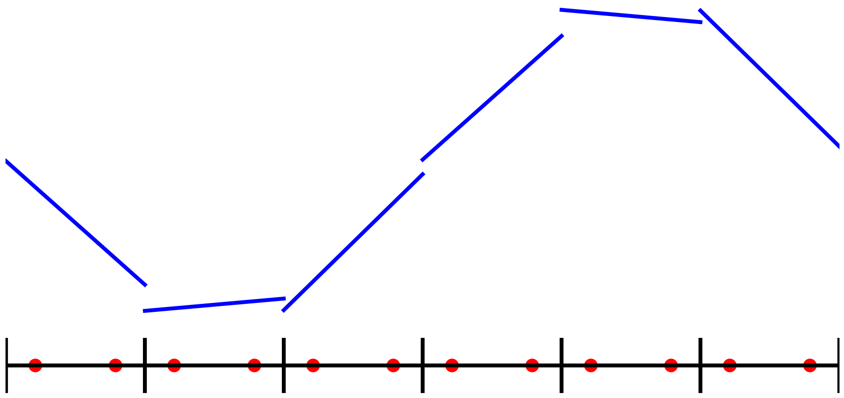


$\hat{r} = \hat{r}(\xi)$

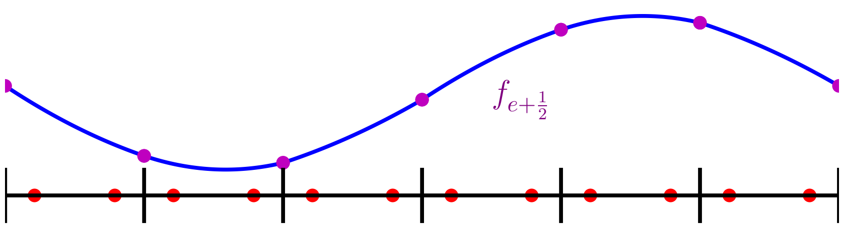
$$\frac{\partial r}{\partial x} = \frac{1}{\Delta x_e} \frac{\partial \hat{r}}{\partial \xi}$$

$$\frac{\mathrm{d}}{\mathrm{d}t} u_{e,i} = -\frac{\partial f_h}{\partial x}(x_{e,i}), \quad 1 \leq i \leq N+1.$$

Degree N approximate solution u_h

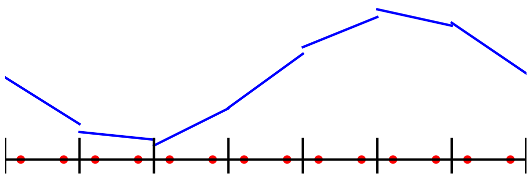


Degree $N+1$ Continuous Flux Approximation f_h

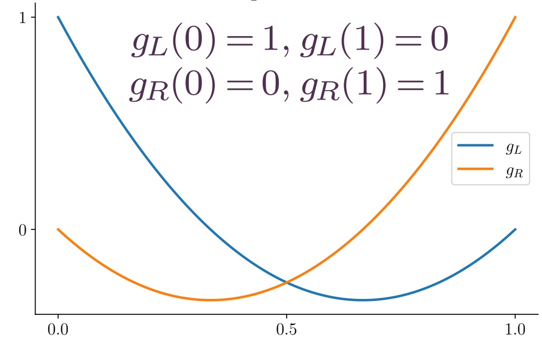


Discontinuous Flux Approximation f_h^δ
Degree N

$$\hat{f}_h^\delta(\xi_i) = f(\hat{u}_h(\xi_i))$$

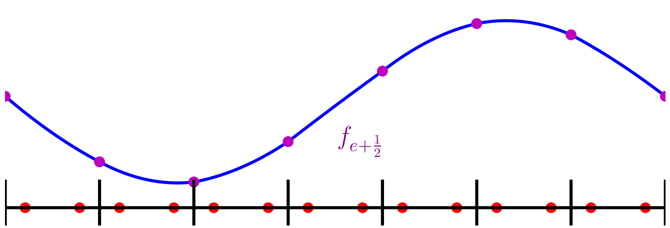


Corrector functions
Degree $N+1$



$$\hat{f}_h = \hat{f}_h^\delta + \left(f_{e-\frac{1}{2}} - \hat{f}_h^\delta(0)\right) g_L + \left(f_{e+\frac{1}{2}} - \hat{f}_h^\delta(1)\right) g_R$$

Continuous Flux Approximation f_h
Degree $N+1$



$$u^{n+1} = u^n - \Delta t\, F_x^n,$$

where $F = f(u) + \dfrac{\Delta t}{2}(f(u))_t + \dfrac{\Delta t^2}{3!} f(u)_{tt} + \cdots + \dfrac{\Delta t^N}{(N+1)!}\dfrac{\partial^N}{\partial t^N} f(u)$

$$u^{n+1} = u^n - \Delta t \, F_x^n,$$
$$\text{where } F = f(u) + \frac{\Delta t}{2}(f(u))_t + \frac{\Delta t^2}{3!} f(u)_{tt} + \cdots + \frac{\Delta t^N}{(N+1)!}\frac{\partial^N}{\partial t^N} f(u)$$

Approximate Lax-Wendroff procedure (Zorio Et Al. [22])

$$\begin{aligned} f(u)_t &\approx \frac{f(u(x,t+\Delta t)) - f(u(x,t-\Delta t))}{2\Delta t} + O(\Delta t^2) \\ &\approx \frac{f(u + \Delta t \, u_t) - f(u - \Delta t \, u_t)}{2\Delta t} + O(\Delta t^2), \end{aligned}$$

and $u_t = -f(u)_x$.

$$u^{n+1} = u^n - \Delta t\, F_x^n,$$
$$\text{where } F = f(u) + \frac{\Delta t}{2}(f(u))_t + \frac{\Delta t^2}{3!} f(u)_{tt} + \cdots + \frac{\Delta t^N}{(N+1)!}\frac{\partial^N}{\partial t^N} f(u)$$

Approximate Lax-Wendroff procedure (Zorio Et Al. [22])

$$\begin{aligned} f(u)_t &\approx \frac{f(u(x, t+\Delta t)) - f(u(x, t-\Delta t))}{2\Delta t} + O(\Delta t^2) \\ &\approx \frac{f(u+\Delta t\, u_t) - f(u-\Delta t\, u_t)}{2\Delta t} + O(\Delta t^2), \end{aligned}$$

and $u_t = -f(u)_x$.

F_h^δ corrected using $F_{e+\frac{1}{2}}$.

$$u^{n+1} = u^n - \Delta t\, F_x^n,$$
$$\text{where } F = f(u) + \frac{\Delta t}{2}(f(u))_t + \frac{\Delta t^2}{3!} f(u)_{tt} + \cdots + \frac{\Delta t^N}{(N+1)!}\frac{\partial^N}{\partial t^N} f(u)$$

Approximate Lax-Wendroff procedure (Zorio Et Al. [22])

$$\begin{aligned} f(u)_t &\approx \frac{f(u(x,t+\Delta t)) - f(u(x,t-\Delta t))}{2\Delta t} + O(\Delta t^2) \\ &\approx \frac{f(u+\Delta t\, u_t) - f(u-\Delta t\, u_t)}{2\Delta t} + O(\Delta t^2), \end{aligned}$$

and $u_t = -f(u)_x$.

F_h^δ corrected using $F_{e+\frac{1}{2}}$.

Past works

$$F_{e+\frac{1}{2}} = \frac{1}{2}\left[F_{e+\frac{1}{2}}^- + F_{e+\frac{1}{2}}^+\right] - \frac{1}{2}\lambda_{e+\frac{1}{2}}\left[u_{e+\frac{1}{2}}^+ - u_{e+\frac{1}{2}}^-\right].$$

$$u^{n+1} = u^n - \Delta t\, F_x^n,$$
$$\text{where } F = f(u) + \frac{\Delta t}{2}(f(u))_t + \frac{\Delta t^2}{3!} f(u)_{tt} + \cdots + \frac{\Delta t^N}{(N+1)!}\frac{\partial^N}{\partial t^N} f(u)$$

Approximate Lax-Wendroff procedure (Zorio Et Al. [22])

$$\begin{aligned} f(u)_t &\approx \frac{f(u(x,t+\Delta t)) - f(u(x,t-\Delta t))}{2\Delta t} + O(\Delta t^2) \\ &\approx \frac{f(u+\Delta t\, u_t) - f(u - \Delta t\, u_t)}{2\Delta t} + O(\Delta t^2), \end{aligned}$$

and $u_t = -f(u)_x$.

F_h^δ corrected using $F_{e+\frac{1}{2}}$.

Past works

$$F_{e+\frac{1}{2}} = \frac{1}{2}[F^-_{e+\frac{1}{2}} + F^+_{e+\frac{1}{2}}] - \frac{1}{2}\lambda_{e+\frac{1}{2}}[u^+_{e+\frac{1}{2}} - u^-_{e+\frac{1}{2}}].$$

Dissipation 2 flux [2]

$$F_{e+\frac{1}{2}} = \frac{1}{2}[F^-_{e+\frac{1}{2}} + F^+_{e+\frac{1}{2}}] - \frac{1}{2}\lambda_{e+\frac{1}{2}}[U^+_{e+\frac{1}{2}} - U^-_{e+\frac{1}{2}}].$$

where

$$U = u + \frac{\Delta t}{2}u_t + \frac{\Delta t^2}{3!}u_{tt} + \cdots + \frac{\Delta t^N}{(N+1)!}\frac{\partial^N u}{\partial t^N} \approx \frac{1}{\Delta t}\int_{t^n}^{t^{n+1}} u_h\, dt.$$

Extension to 2-D

$$u_t + f(u)_x + g(u)_y = 0$$

Extension to 2-D

$$u_t + f(u)_x + g(u)_y = 0$$

$$u^{n+1} = u^n - \Delta t \left[\frac{\partial F}{\partial x}(u^n) + \frac{\partial G}{\partial y}(u^n) \right] + O(\Delta t^n)$$

$$F(u) = \sum_{m:1}^{N+1} \frac{\Delta t^m}{(m+1)!} \partial_t^m f(u), \qquad G(u) = \sum_{m=1}^{N+1} \frac{\Delta t^m}{(m+1)!} \partial_t^m g(u)$$

$$u_t + f(u)_x + g(u)_y = 0$$

$$u^{n+1} = u^n - \Delta t \left[\frac{\partial F}{\partial x}(u^n) + \frac{\partial G}{\partial y}(u^n) \right] + O(\Delta t^n)$$

$$F(u) = \sum_{m:1}^{N+1} \frac{\Delta t^m}{(m+1)!} \partial_t^m f(u), \qquad G(u) = \sum_{m=1}^{N+1} \frac{\Delta t^m}{(m+1)!} \partial_t^m g(u)$$

$$u_h^{\boldsymbol{e}} = \sum_{i,j=1}^{N+1} u_{ij}^{\boldsymbol{e}} \, \ell_i(\xi) \, \ell_j(\eta),$$

$$F_h^\delta(\xi, \eta) = \sum_{i,j=1}^{N+1} F_{ij} \, \ell_i(\xi) \, \ell_j(\eta), \qquad G_h^\delta(\xi, \eta) = \sum_{i,j=1}^{N+1} G_{ij} \, \ell_i(\xi) \, \ell_j(\eta).$$

$$u_t + f(u)_x + g(u)_y = 0$$

$$u^{n+1} = u^n - \Delta t \left[\frac{\partial F}{\partial x}(u^n) + \frac{\partial G}{\partial y}(u^n) \right] + O(\Delta t^n)$$

$$F(u) = \sum_{m:1}^{N+1} \frac{\Delta t^m}{(m+1)!} \partial_t^m f(u), \qquad G(u) = \sum_{m=1}^{N+1} \frac{\Delta t^m}{(m+1)!} \partial_t^m g(u)$$

$$u_h^{\boldsymbol{e}} = \sum_{i,j=1}^{N+1} u_{ij}^{\boldsymbol{e}} \, \ell_i(\xi) \, \ell_j(\eta),$$

$$F_h^\delta(\xi,\eta) = \sum_{i,j=1}^{N+1} F_{ij} \, \ell_i(\xi) \, \ell_j(\eta), \qquad G_h^\delta(\xi,\eta) = \sum_{i,j=1}^{N+1} G_{ij} \, \ell_i(\xi) \, \ell_j(\eta).$$

$$F_h^{\boldsymbol{e}}(\xi,\eta_j) = [F_{e_x-\frac{1}{2},j} - F_h^\delta(0,\eta_j)] \, g_L(\xi) + F_h^\delta(\xi,\eta_j) + [F_{e_x+\frac{1}{2},j} - F_h^\delta(1,\eta_j)] \, g_R(\xi),$$

$$G_h^{\boldsymbol{e}}(\xi_i,\eta) = [G_{i,e_y-\frac{1}{2}} - G_h^\delta(\xi_i,0)] \, g_L(\eta) + G_h^\delta(\xi_i,\eta) + [G_{i,e_y+\frac{1}{2}} - G_h^\delta(\xi_i,1)] \, g_R(\eta).$$

$$u_t + f(u)_x + g(u)_y = 0$$

$$u^{n+1} = u^n - \Delta t \left[\frac{\partial F}{\partial x}(u^n) + \frac{\partial G}{\partial y}(u^n) \right] + O(\Delta t^n)$$

$$F(u) = \sum_{m:1}^{N+1} \frac{\Delta t^m}{(m+1)!} \partial_t^m f(u), \qquad G(u) = \sum_{m=1}^{N+1} \frac{\Delta t^m}{(m+1)!} \partial_t^m g(u)$$

$$u_h^{\boldsymbol{e}} = \sum_{i,j=1}^{N+1} u_{ij}^{\boldsymbol{e}} \ell_i(\xi) \ell_j(\eta),$$

$$F_h^\delta(\xi,\eta) = \sum_{i,j=1}^{N+1} F_{ij} \ell_i(\xi) \ell_j(\eta), \qquad G_h^\delta(\xi,\eta) = \sum_{i,j=1}^{N+1} G_{ij} \ell_i(\xi) \ell_j(\eta).$$

$$F_h^{\boldsymbol{e}}(\xi,\eta_j) = [F_{e_x - \frac{1}{2}, j} - F_h^\delta(0,\eta_j)]\, g_L(\xi) + F_h^\delta(\xi,\eta_j) + [F_{e_x+\frac{1}{2}, j} - F_h^\delta(1,\eta_j)]\, g_R(\xi),$$
$$G_h^{\boldsymbol{e}}(\xi_i,\eta) = [G_{i, e_y - \frac{1}{2}} - G_h^\delta(\xi_i,0)]\, g_L(\eta) + G_h^\delta(\xi_i,\eta) + [G_{i, e_y+\frac{1}{2}} - G_h^\delta(\xi_i,1)]\, g_R(\eta).$$

$$u_{ij}^{n+1} = u_{ij}^n - \Delta t \left[\frac{\partial F_h}{\partial x}(\xi_i,\eta_j) + \frac{\partial G_h}{\partial y}(\xi_i,\eta_j) \right], \qquad 1 \le i, j \le N+1.$$

High order LWFR update

$$\boldsymbol{u}_e^{H,n+1} = \boldsymbol{u}_e^n - \frac{\Delta t}{\Delta x_e} \boldsymbol{R}_e^H.$$

High order LWFR update

$$\boldsymbol{u}_e^{H,n+1} = \boldsymbol{u}_e^n - \frac{\Delta t}{\Delta x_e} \boldsymbol{R}_e^H.$$

Lower order subcell update (FO FVM or MUSCL-Hancock)

$$\boldsymbol{u}_e^{L,n+1} = \boldsymbol{u}_e^n - \frac{\Delta t}{\Delta x_e} \boldsymbol{R}_e^L.$$

Blending limiter (Hennemann Et Al. [8])

High order LWFR update

$$\boldsymbol{u}_e^{H,n+1} = \boldsymbol{u}_e^n - \frac{\Delta t}{\Delta x_e} \boldsymbol{R}_e^H.$$

Lower order subcell update (FO FVM or MUSCL-Hancock)

$$\boldsymbol{u}_e^{L,n+1} = \boldsymbol{u}_e^n - \frac{\Delta t}{\Delta x_e} \boldsymbol{R}_e^L.$$

Blend residual with $\alpha_e \in [0,1]$

$$\boldsymbol{R}_e = (1-\alpha_e)\, \boldsymbol{R}_e^H + \alpha_e\, \boldsymbol{R}_e^L,$$

Limited update

$$\boldsymbol{u}_e^{n+1} = \boldsymbol{u}_e^n - \frac{\Delta t}{\Delta x_e} \boldsymbol{R}_e.$$

Choice of α_e : Smoothness indicator (Hennemann Et Al. [8])

Legendre expansion of degree N polynomial $\epsilon = \epsilon(\xi)$

$$\epsilon = \sum_{j=1}^{N+1} m_j L_j, \qquad m_j = \langle \epsilon, L_j \rangle_{L^2},$$

Energy content (Persson and Peraire [12])

$$\mathbb{E} := \max\left(\frac{m_{N+1}^2}{\beta_1 m_1^2 + \sum_{j=2}^{N+1} m_j^2}, \frac{m_N^2}{\beta_2 m_1^2 + \sum_{j=2}^{N} m_j^2} \right).$$

Legendre expansion of degree N polynomial $\epsilon = \epsilon(\xi)$

$$\epsilon = \sum_{j=1}^{N+1} m_j L_j, \qquad m_j = \langle \epsilon, L_j \rangle_{L^2},$$

Energy content (Persson and Peraire [12])

$$\mathbb{E} := \max\left(\frac{m_{N+1}^2}{\beta_1 m_1^2 + \sum_{j=2}^{N+1} m_j^2}, \frac{m_N^2}{\beta_2 m_1^2 + \sum_{j=2}^{N} m_j^2} \right).$$

In literature, $\beta_1 = \beta_2 = 1$.

Legendre expansion of degree N polynomial $\epsilon = \epsilon(\xi)$

$$\epsilon = \sum_{j=1}^{N+1} m_j L_j, \qquad m_j = \langle \epsilon, L_j \rangle_{L^2},$$

Energy content (Persson and Peraire [12])

$$\mathbb{E} := \max\left(\frac{m_{N+1}^2}{\beta_1 m_1^2 + \sum_{j=2}^{N+1} m_j^2}, \frac{m_N^2}{\beta_2 m_1^2 + \sum_{j=2}^{N} m_j^2} \right).$$

In literature, $\beta_1 = \beta_2 = 1$.

$$\epsilon = \rho\, p$$

Legendre expansion of degree N polynomial $\epsilon = \epsilon(\xi)$

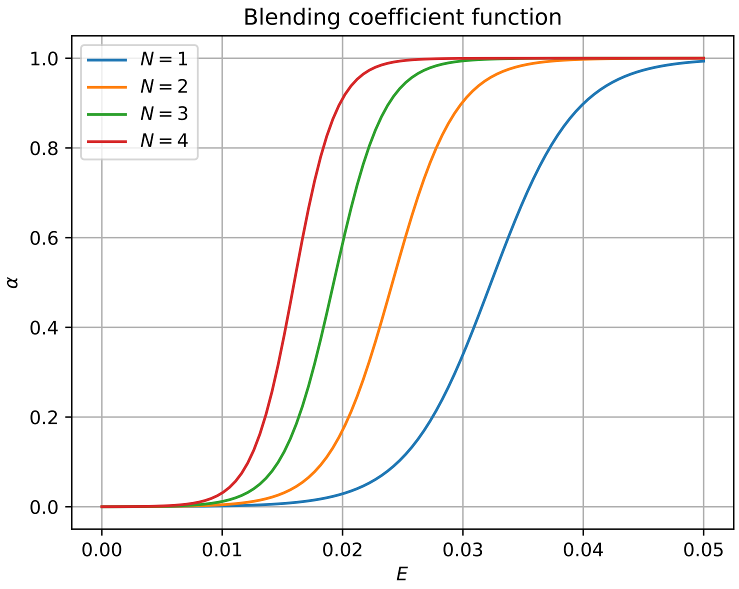
$$\epsilon = \sum_{j=1}^{N+1} m_j L_j, \qquad m_j = \langle \epsilon, L_j \rangle_{L^2},$$

Energy content (Persson and Peraire [12])

$$\mathbb{E} := \max\left(\frac{m_{N+1}^2}{\beta_1 m_1^2 + \sum_{j=2}^{N+1} m_j^2}, \frac{m_N^2}{\beta_2 m_1^2 + \sum_{j=2}^{N} m_j^2} \right).$$

In literature, $\beta_1 = \beta_2 = 1$.

$$\epsilon = \rho\, p$$



$$\alpha(\mathbb{E}) = \frac{1}{1 + \exp\left(-\frac{s}{\mathbb{T}}(\mathbb{E} - \mathbb{T})\right)}$$

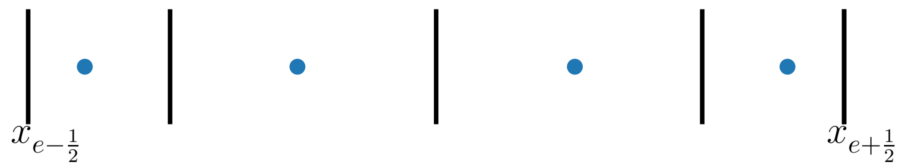
where

$$\mathbb{T}(N) = 0.5 \cdot 10^{-1.8(N+1)^{1/4}}, \qquad \alpha(\mathbb{E} = 0) = 0.0001$$

$$\tilde{\alpha} = \begin{cases} 0, & \text{if } \alpha < \alpha_{\min} \\ \alpha, & \text{if } \alpha_{\min} \leq \alpha \leq 1 - \alpha_{\min} \\ 1, & \text{if } 1 - \alpha_{\min} < \alpha \end{cases}$$

$$\alpha^{\text{final}} = \max_{e \in V_e} \{\alpha, 0.5\, \alpha_e\}$$

Solution points and subcells



Subcell $[x^e_{j-\frac{1}{2}}, x^e_{j+\frac{1}{2}}]$

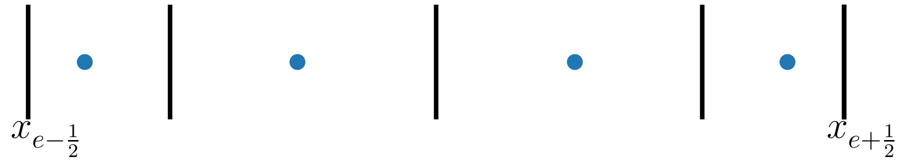
$$x^e_{j+\frac{1}{2}} - x^e_{j-\frac{1}{2}} = \Delta x_e \, w_j, \qquad 1 \le j \le N+1,$$

where $\{w_j\}_{j:1}^{N+1}$ are the Gauss-Legendre quadrature weights.

FVM

$$\overline{u}_e^{n+1} = \overline{u}_e^n - \frac{\Delta t}{\Delta x_e}(F_{e+\frac{1}{2}} - F_{e-\frac{1}{2}}).$$

Solution points and subcells



Subcell $[x^e_{j-\frac{1}{2}}, x^e_{j+\frac{1}{2}}]$

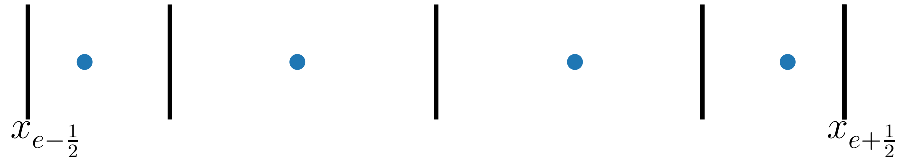
$$x^e_{j+\frac{1}{2}} - x^e_{j-\frac{1}{2}} = \Delta x_e \, w_j \, , \qquad 1 \le j \le N+1,$$

where $\{w_j\}_{j:1}^{N+1}$ are the Gauss-Legendre quadrature weights.

FVM

$$\overline{u}_e^{n+1} = \overline{u}_e^n - \frac{\Delta t}{\Delta x_e} (F_{e+\frac{1}{2}} - F_{e-\frac{1}{2}}).$$

Solution points and subcells

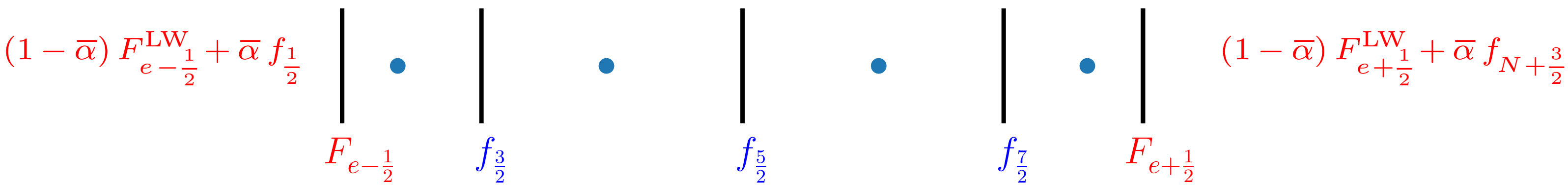


Subcell $[x^e_{j-\frac{1}{2}}, x^e_{j+\frac{1}{2}}]$

$$x^e_{j+\frac{1}{2}} - x^e_{j-\frac{1}{2}} = \Delta x_e \, w_j \,, \qquad 1 \le j \le N+1,$$

where $\{w_j\}_{j:1}^{N+1}$ are the Gauss-Legendre quadrature weights.

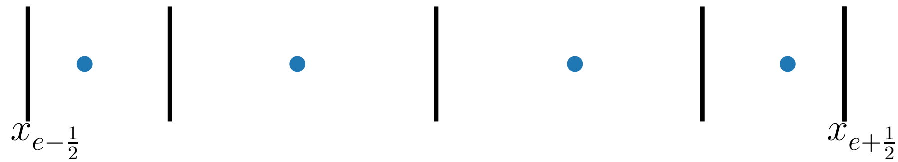
FVM



$$\overline{u}_e^{n+1} = \overline{u}_e^n - \frac{\Delta t}{\Delta x_e}(F_{e+\frac{1}{2}} - F_{e-\frac{1}{2}}).$$

Lower order update

Solution points and subcells

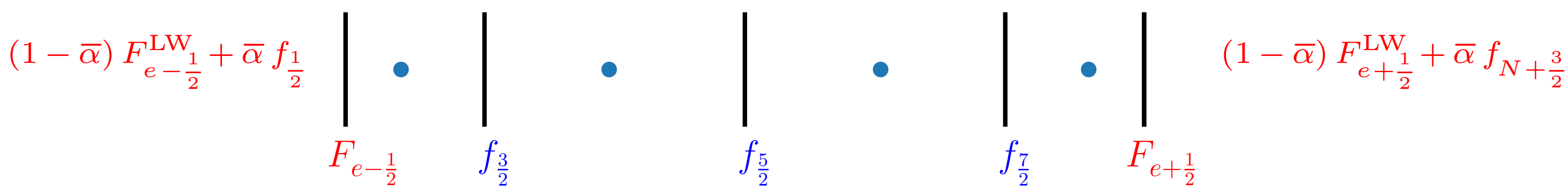


Subcell $[x^e_{j-\frac{1}{2}}, x^e_{j+\frac{1}{2}}]$

$$x^e_{j+\frac{1}{2}} - x^e_{j-\frac{1}{2}} = \Delta x_e \, w_j \,, \qquad 1 \le j \le N+1,$$

where $\{w_j\}_{j:1}^{N+1}$ are the Gauss-Legendre quadrature weights.

FVM



$$\overline{u}^{n+1}_e = \overline{u}^n_e - \frac{\Delta t}{\Delta x_e}(F_{e+\frac{1}{2}} - F_{e-\frac{1}{2}}).$$

Initial candidate

$$\tilde{\boldsymbol{F}}_{e+\frac{1}{2}} = \left(1-\alpha_{e+\frac{1}{2}}\right)\boldsymbol{F}^{\text{LW}}_{e+\frac{1}{2}} + \alpha_{e+\frac{1}{2}}\boldsymbol{f}_{e,N+3/2}, \qquad \alpha_{e+\frac{1}{2}} = \frac{1}{2}(\alpha_e+\alpha_{e+1}).$$

Lower order update of last solution point of cell e

$$\tilde{\boldsymbol{u}}^{n+1} = \boldsymbol{u}^n_{e,N+1} - \frac{\Delta t}{\Delta x_e w_{N+1}}\left(\tilde{\boldsymbol{F}}_{e+\frac{1}{2}} - \boldsymbol{f}_{e,N+1/2}\right).$$

Assume concave p such that

$$\mathcal{U}_{\text{ad}} = \{\boldsymbol{u}\in\Omega\colon p_i(\boldsymbol{u})>0\}.$$

For purely low order

$$\tilde{\boldsymbol{u}}^{n+1}_{\text{low}} = \boldsymbol{u}^n_{e,N+1} - \frac{\Delta t}{\Delta x_e w_{N+1}}(\boldsymbol{f}_{e,N+3/2} - \boldsymbol{f}_{e,N+1/2}) \in \mathcal{U}_{\text{ad}}.$$

Thus, for

$$\theta = \min\left(\left|\frac{\epsilon - p(\tilde{\boldsymbol{u}}^{n+1}_{\text{low}})}{p(\tilde{\boldsymbol{u}}^{n+1}) - p(\tilde{\boldsymbol{u}}^{n+1}_{\text{low}})}\right|, 1\right),$$

we will have

$$p(\theta\,\tilde{\boldsymbol{u}}^{n+1} + (1-\theta)\,\boldsymbol{u}^{n+1}_{\text{low}}) \geq \theta\,p(\tilde{\boldsymbol{u}}^{n+1}) + (1-\theta)\,p(\boldsymbol{u}^{n+1}_{\text{low}}) > \epsilon.$$

Finally choose

$$\boldsymbol{F}_{e+\frac{1}{2}} = \theta\,\tilde{\boldsymbol{F}}_{e+\frac{1}{2}} + (1-\theta)\,\boldsymbol{f}_{e,N+3/2}.$$

Interface numerical flux

Initial candidate

$$\tilde{\boldsymbol{F}}_{e+\frac{1}{2}} = \left(1 - \alpha_{e+\frac{1}{2}}\right) \boldsymbol{F}^{\text{LW}}_{e+\frac{1}{2}} + \alpha_{e+\frac{1}{2}} \boldsymbol{f}_{e,N+3/2}, \qquad \alpha_{e+\frac{1}{2}} = \frac{1}{2}(\alpha_e + \alpha_{e+1}).$$

Lower order update of last solution point of cell e

$$\tilde{\boldsymbol{u}}^{n+1} = \boldsymbol{u}^n_{e,N+1} - \frac{\Delta t}{\Delta x_e w_{N+1}} \left(\tilde{\boldsymbol{F}}_{e+\frac{1}{2}} - \boldsymbol{f}_{e,N+1/2}\right).$$

Assume concave p such that

$$\mathcal{U}_{\text{ad}} = \{\boldsymbol{u} \in \Omega : p_i(\boldsymbol{u}) > 0\}.$$

For purely low order

$$\tilde{\boldsymbol{u}}^{n+1}_{\text{low}} = \boldsymbol{u}^n_{e,N+1} - \frac{\Delta t}{\Delta x_e w_{N+1}} (\boldsymbol{f}_{e,N+3/2} - \boldsymbol{f}_{e,N+1/2}) \in \mathcal{U}_{\text{ad}}.$$

Thus, for

$$\theta = \min\left(\left|\frac{\epsilon - p(\tilde{\boldsymbol{u}}^{n+1}_{\text{low}})}{p(\tilde{\boldsymbol{u}}^{n+1}) - p(\tilde{\boldsymbol{u}}^{n+1}_{\text{low}})}\right|, 1\right),$$

we will have

$$p(\theta\, \tilde{\boldsymbol{u}}^{n+1} + (1-\theta)\, \boldsymbol{u}^{n+1}_{\text{low}}) \geq \theta\, p(\tilde{\boldsymbol{u}}^{n+1}) + (1-\theta)\, p(\boldsymbol{u}^{n+1}_{\text{low}}) > \epsilon.$$

Finally choose

$$\boldsymbol{F}_{e+\frac{1}{2}} = \theta\, \tilde{\boldsymbol{F}}_{e+\frac{1}{2}} + (1-\theta)\, \boldsymbol{f}_{e,N+3/2}.$$

Interface numerical flux

Initial candidate

$$\tilde{\boldsymbol{F}}_{e+\frac{1}{2}} = \left(1-\alpha_{e+\frac{1}{2}}\right)\boldsymbol{F}^{\text{LW}}_{e+\frac{1}{2}} + \alpha_{e+\frac{1}{2}}\boldsymbol{f}_{e,N+3/2}, \qquad \alpha_{e+\frac{1}{2}} = \frac{1}{2}(\alpha_e+\alpha_{e+1}).$$

Lower order update of last solution point of cell e

$$\tilde{\boldsymbol{u}}^{n+1} = \boldsymbol{u}^n_{e,N+1} - \frac{\Delta t}{\Delta x_e w_{N+1}}\left(\tilde{\boldsymbol{F}}_{e+\frac{1}{2}} - \boldsymbol{f}_{e,N+1/2}\right).$$

Assume concave p such that

$$\mathcal{U}_{\text{ad}} = \{\boldsymbol{u}\in\Omega : p_i(\boldsymbol{u}) > 0\}.$$

For purely low order

$$\tilde{\boldsymbol{u}}^{n+1}_{\text{low}} = \boldsymbol{u}^n_{e,N+1} - \frac{\Delta t}{\Delta x_e w_{N+1}}(\boldsymbol{f}_{e,N+3/2} - \boldsymbol{f}_{e,N+1/2}) \in \mathcal{U}_{\text{ad}}.$$

Thus, for

$$\theta = \min\left(\left|\frac{\epsilon - p(\tilde{\boldsymbol{u}}^{n+1}_{\text{low}})}{p(\tilde{\boldsymbol{u}}^{n+1}) - p(\tilde{\boldsymbol{u}}^{n+1}_{\text{low}})}\right|, 1\right),$$

we will have

$$p(\theta\,\tilde{\boldsymbol{u}}^{n+1} + (1-\theta)\,\boldsymbol{u}^{n+1}_{\text{low}}) \geq \theta\, p(\tilde{\boldsymbol{u}}^{n+1}) + (1-\theta)\, p(\boldsymbol{u}^{n+1}_{\text{low}}) > \epsilon.$$

Finally choose

$$\boldsymbol{F}_{e+\frac{1}{2}} = \theta\,\tilde{\boldsymbol{F}}_{e+\frac{1}{2}} + (1-\theta)\,\boldsymbol{f}_{e,N+3/2}.$$

Interface numerical flux

Initial candidate

$$\tilde{\boldsymbol{F}}_{e+\frac{1}{2}} = \left(1 - \alpha_{e+\frac{1}{2}}\right) \boldsymbol{F}^{\text{LW}}_{e+\frac{1}{2}} + \alpha_{e+\frac{1}{2}} \boldsymbol{f}_{e,N+3/2}, \qquad \alpha_{e+\frac{1}{2}} = \frac{1}{2}(\alpha_e + \alpha_{e+1}).$$

Lower order update of last solution point of cell e

$$\tilde{\boldsymbol{u}}^{n+1} = \boldsymbol{u}^n_{e,N+1} - \frac{\Delta t}{\Delta x_e w_{N+1}} \left(\tilde{\boldsymbol{F}}_{e+\frac{1}{2}} - \boldsymbol{f}_{e,N+1/2}\right).$$

Assume concave p such that

$$\mathcal{U}_{\text{ad}} = \{\boldsymbol{u} \in \Omega \colon p_i(\boldsymbol{u}) > 0\}.$$

For purely low order

$$\tilde{\boldsymbol{u}}^{n+1}_{\text{low}} = \boldsymbol{u}^n_{e,N+1} - \frac{\Delta t}{\Delta x_e w_{N+1}} (\boldsymbol{f}_{e,N+3/2} - \boldsymbol{f}_{e,N+1/2}) \in \mathcal{U}_{\text{ad}}.$$

Thus, for

$$\theta = \min\left(\left|\frac{\epsilon - p(\tilde{\boldsymbol{u}}^{n+1}_{\text{low}})}{p(\tilde{\boldsymbol{u}}^{n+1}) - p(\tilde{\boldsymbol{u}}^{n+1}_{\text{low}})}\right|, 1\right),$$

we will have

$$p(\theta\, \tilde{\boldsymbol{u}}^{n+1} + (1-\theta)\, \boldsymbol{u}^{n+1}_{\text{low}}) \geq \theta\, p(\tilde{\boldsymbol{u}}^{n+1}) + (1-\theta)\, p(\boldsymbol{u}^{n+1}_{\text{low}}) > \epsilon.$$

Finally choose

$$\boldsymbol{F}_{e+\frac{1}{2}} = \theta\, \tilde{\boldsymbol{F}}_{e+\frac{1}{2}} + (1-\theta)\, \boldsymbol{f}_{e,N+3/2}.$$

Initial candidate

$$\tilde{\boldsymbol{F}}_{e+\frac{1}{2}} = \left(1 - \alpha_{e+\frac{1}{2}}\right) \boldsymbol{F}^{\mathrm{LW}}_{e+\frac{1}{2}} + \alpha_{e+\frac{1}{2}} \boldsymbol{f}_{e,N+3/2}, \qquad \alpha_{e+\frac{1}{2}} = \frac{1}{2}(\alpha_e + \alpha_{e+1}).$$

Lower order update of last solution point of cell e

$$\tilde{\boldsymbol{u}}^{n+1} = \boldsymbol{u}^n_{e,N+1} - \frac{\Delta t}{\Delta x_e w_{N+1}} \left(\tilde{\boldsymbol{F}}_{e+\frac{1}{2}} - \boldsymbol{f}_{e,N+1/2} \right).$$

Assume concave p such that

$$\mathcal{U}_{\mathrm{ad}} = \{\boldsymbol{u} \in \Omega \colon p_i(\boldsymbol{u}) > 0\}.$$

For purely low order

$$\tilde{\boldsymbol{u}}^{n+1}_{\mathrm{low}} = \boldsymbol{u}^n_{e,N+1} - \frac{\Delta t}{\Delta x_e w_{N+1}} (\boldsymbol{f}_{e,N+3/2} - \boldsymbol{f}_{e,N+1/2}) \in \mathcal{U}_{\mathrm{ad}}.$$

Thus, for

$$\theta = \min \left(\left| \frac{\epsilon - p(\tilde{\boldsymbol{u}}^{n+1}_{\mathrm{low}})}{p(\tilde{\boldsymbol{u}}^{n+1}) - p(\tilde{\boldsymbol{u}}^{n+1}_{\mathrm{low}})} \right|, 1 \right),$$

we will have

$$p(\theta\, \tilde{\boldsymbol{u}}^{n+1} + (1-\theta)\, \boldsymbol{u}^{n+1}_{\mathrm{low}}) \geq \theta\, p(\tilde{\boldsymbol{u}}^{n+1}) + (1-\theta)\, p(\boldsymbol{u}^{n+1}_{\mathrm{low}}) > \epsilon.$$

Finally choose

$$\boldsymbol{F}_{e+\frac{1}{2}} = \theta\, \tilde{\boldsymbol{F}}_{e+\frac{1}{2}} + (1-\theta)\, \boldsymbol{f}_{e,N+3/2}.$$

Initial candidate

$$\tilde{\boldsymbol{F}}_{e+\frac{1}{2}} = \left(1 - \alpha_{e+\frac{1}{2}}\right) \boldsymbol{F}^{\text{LW}}_{e+\frac{1}{2}} + \alpha_{e+\frac{1}{2}} \boldsymbol{f}_{e,N+3/2}, \qquad \alpha_{e+\frac{1}{2}} = \frac{1}{2}(\alpha_e + \alpha_{e+1}).$$

Lower order update of last solution point of cell e

$$\tilde{\boldsymbol{u}}^{n+1} = \boldsymbol{u}^n_{e,N+1} - \frac{\Delta t}{\Delta x_e w_{N+1}} \left(\tilde{\boldsymbol{F}}_{e+\frac{1}{2}} - \boldsymbol{f}_{e,N+1/2} \right).$$

Assume concave p such that

$$\mathcal{U}_{\text{ad}} = \{\boldsymbol{u} \in \Omega : p_i(\boldsymbol{u}) > 0\}.$$

For purely low order

$$\tilde{\boldsymbol{u}}^{n+1}_{\text{low}} = \boldsymbol{u}^n_{e,N+1} - \frac{\Delta t}{\Delta x_e w_{N+1}} (\boldsymbol{f}_{e,N+3/2} - \boldsymbol{f}_{e,N+1/2}) \in \mathcal{U}_{\text{ad}}.$$

Thus, for

$$\theta = \min\left(\left| \frac{\epsilon - p(\tilde{\boldsymbol{u}}^{n+1}_{\text{low}})}{p(\tilde{\boldsymbol{u}}^{n+1}) - p(\tilde{\boldsymbol{u}}^{n+1}_{\text{low}})} \right|, 1 \right),$$

we will have

$$p(\theta\, \tilde{\boldsymbol{u}}^{n+1} + (1-\theta)\, \boldsymbol{u}^{n+1}_{\text{low}}) \geq \theta\, p(\tilde{\boldsymbol{u}}^{n+1}) + (1-\theta)\, p(\boldsymbol{u}^{n+1}_{\text{low}}) > \epsilon.$$

Finally choose

$$\boldsymbol{F}_{e+\frac{1}{2}} = \theta\, \tilde{\boldsymbol{F}}_{e+\frac{1}{2}} + (1-\theta)\, \boldsymbol{f}_{e,N+3/2}.$$

Extension of Zhang-Shu's limiter to Lax-Wendroff schemes

Low order residual

$$
\begin{aligned}
(\tilde{\boldsymbol{u}}_1^e)^{n+1} &= (\boldsymbol{u}_1^e)^n - \frac{\Delta t}{w_1\,\Delta x_e}\Big[f_{\frac{3}{2}} - F_{e-\frac{1}{2}}\Big],\\
(\tilde{\boldsymbol{u}}_j^e)^{n+1} &= (\boldsymbol{u}_j^e)^n - \frac{\Delta t}{w_j\,\Delta x_e}\Big[f_{j+\frac{1}{2}} - f_{j-\frac{1}{2}}\Big], \qquad 2 \le j \le N,\\
(\tilde{\boldsymbol{u}}_N^e)^{n+1} &= (\boldsymbol{u}_N^e)^n - \frac{\Delta t}{w_{N+1}\,\Delta x_e}\Big[F_{e+\frac{1}{2}} - f_{N-\frac{1}{2}}\Big].
\end{aligned}
$$

By appropriate of $F_{e\pm\frac{1}{2}}$,

$$
(\tilde{\boldsymbol{u}}_j^e)^{n+1} \in \mathcal{U}_{\mathrm{ad}}, \qquad 1 \le j \le N+1.
$$

$$
\Rightarrow \overline{\boldsymbol{u}}_e^{n+1} = \overline{\boldsymbol{u}}_e^n - \frac{\Delta t}{\Delta x_e}\Big(F_{e+\frac{1}{2}} - F_{e-\frac{1}{2}}\Big) = \sum_{j=1}^{N+1} w_j\,\tilde{\boldsymbol{u}}_j^{n+1} \in \mathcal{U}_{\mathrm{ad}}.
$$

Zhang-Shu's limiter applies.

Similar approach used by Rossmanith Et Al. [6].

We can also aposteriorily modify α_e (Gassner and Ramírez [14]).

Low order residual

$$(\tilde{\boldsymbol{u}}_1^e)^{n+1} = (\boldsymbol{u}_1^e)^n - \frac{\Delta t}{w_1 \Delta x_e}\Big[f_{\frac{3}{2}} - F_{e-\frac{1}{2}}\Big],$$
$$(\tilde{\boldsymbol{u}}_j^e)^{n+1} = (\boldsymbol{u}_j^e)^n - \frac{\Delta t}{w_j \Delta x_e}\Big[f_{j+\frac{1}{2}} - f_{j-\frac{1}{2}}\Big], \qquad 2 \le j \le N,$$
$$(\tilde{\boldsymbol{u}}_N^e)^{n+1} = (\boldsymbol{u}_N^e)^n - \frac{\Delta t}{w_{N+1} \Delta x_e}\Big[F_{e+\frac{1}{2}} - f_{N-\frac{1}{2}}\Big].$$

By appropriate of $F_{e\pm\frac{1}{2}}$,

$$(\tilde{\boldsymbol{u}}_j^e)^{n+1} \in \mathcal{U}_{\text{ad}}, \qquad 1 \le j \le N+1.$$

$$\Rightarrow \overline{\boldsymbol{u}}_e^{n+1} = \overline{\boldsymbol{u}}_e^n - \frac{\Delta t}{\Delta x_e}\Big(F_{e+\frac{1}{2}} - F_{e-\frac{1}{2}}\Big) = \sum_{j=1}^{N+1} w_j \, \tilde{\boldsymbol{u}}_j^{n+1} \in \mathcal{U}_{\text{ad}}.$$

Zhang-Shu's limiter applies.

Similar approach used by Rossmanith Et Al. [6].

We can also aposteriorily modify α_e (Gassner and Ramírez [14]).

Low order residual

$$(\tilde{\boldsymbol{u}}_1^e)^{n+1} = (\boldsymbol{u}_1^e)^n - \frac{\Delta t}{w_1 \, \Delta x_e}\Big[f_{\frac{3}{2}} - F_{e-\frac{1}{2}} \Big],$$

$$(\tilde{\boldsymbol{u}}_j^e)^{n+1} = (\boldsymbol{u}_j^e)^n - \frac{\Delta t}{w_j \, \Delta x_e}\Big[f_{j+\frac{1}{2}} - f_{j-\frac{1}{2}} \Big], \qquad 2 \leq j \leq N,$$

$$(\tilde{\boldsymbol{u}}_N^e)^{n+1} = (\boldsymbol{u}_N^e)^n - \frac{\Delta t}{w_{N+1} \, \Delta x_e}\Big[F_{e+\frac{1}{2}} - f_{N-\frac{1}{2}} \Big].$$

By appropriate of $F_{e \pm \frac{1}{2}}$,

$$(\tilde{\boldsymbol{u}}_j^e)^{n+1} \in \mathcal{U}_{\text{ad}}, \qquad 1 \leq j \leq N+1.$$

$$\Rightarrow \overline{\boldsymbol{u}}_e^{n+1} = \overline{\boldsymbol{u}}_e^n - \frac{\Delta t}{\Delta x_e}\Big(F_{e+\frac{1}{2}} - F_{e-\frac{1}{2}} \Big) = \sum_{j=1}^{N+1} w_j \, \tilde{\boldsymbol{u}}_j^{n+1} \in \mathcal{U}_{\text{ad}}.$$

Zhang-Shu's limiter applies.

Similar approach used by Rossmanith Et Al. [6].

We can also aposteriorily modify α_e (Gassner and Ramírez [14]).

Low order residual

$$
\begin{aligned}
(\tilde{\boldsymbol{u}}_1^e)^{n+1} &= (\boldsymbol{u}_1^e)^n - \frac{\Delta t}{w_1\,\Delta x_e}\Big[f_{\frac{3}{2}} - F_{e-\frac{1}{2}}\Big],\\
(\tilde{\boldsymbol{u}}_j^e)^{n+1} &= (\boldsymbol{u}_j^e)^n - \frac{\Delta t}{w_j\,\Delta x_e}\Big[f_{j+\frac{1}{2}} - f_{j-\frac{1}{2}}\Big], \qquad 2 \le j \le N,\\
(\tilde{\boldsymbol{u}}_N^e)^{n+1} &= (\boldsymbol{u}_N^e)^n - \frac{\Delta t}{w_{N+1}\,\Delta x_e}\Big[F_{e+\frac{1}{2}} - f_{N-\frac{1}{2}}\Big].
\end{aligned}
$$

By appropriate of $F_{e\pm\frac{1}{2}}$,

$$
(\tilde{\boldsymbol{u}}_j^e)^{n+1} \in \mathcal{U}_{\text{ad}}, \qquad 1 \le j \le N+1.
$$

$$
\Rightarrow \overline{\boldsymbol{u}}_e^{n+1} = \overline{\boldsymbol{u}}_e^n - \frac{\Delta t}{\Delta x_e}\Big(F_{e+\frac{1}{2}} - F_{e-\frac{1}{2}}\Big) = \sum_{j=1}^{N+1} w_j\,\tilde{\boldsymbol{u}}_j^{n+1} \in \mathcal{U}_{\text{ad}}.
$$

Zhang-Shu's limiter applies.

Similar approach used by Rossmanith Et Al. [6].

We can also aposteriorily modify α_e (Gassner and Ramírez [14]).

Low order residual

$$(\tilde{\boldsymbol{u}}_1^e)^{n+1} = (\boldsymbol{u}_1^e)^n - \frac{\Delta t}{w_1 \Delta x_e}\Big[f_{\frac{3}{2}} - F_{e-\frac{1}{2}} \Big],$$
$$(\tilde{\boldsymbol{u}}_j^e)^{n+1} = (\boldsymbol{u}_j^e)^n - \frac{\Delta t}{w_j \Delta x_e}\Big[f_{j+\frac{1}{2}} - f_{j-\frac{1}{2}} \Big], \qquad 2 \le j \le N,$$
$$(\tilde{\boldsymbol{u}}_N^e)^{n+1} = (\boldsymbol{u}_N^e)^n - \frac{\Delta t}{w_{N+1} \Delta x_e}\Big[F_{e+\frac{1}{2}} - f_{N-\frac{1}{2}} \Big].$$

By appropriate of $F_{e\pm\frac{1}{2}}$,

$$(\tilde{\boldsymbol{u}}_j^e)^{n+1} \in \mathcal{U}_{\text{ad}}, \qquad 1 \le j \le N+1.$$

$$\Rightarrow \overline{\boldsymbol{u}}_e^{n+1} = \overline{\boldsymbol{u}}_e^n - \frac{\Delta t}{\Delta x_e}\Big(F_{e+\frac{1}{2}} - F_{e-\frac{1}{2}} \Big) = \sum_{j=1}^{N+1} w_j \, \tilde{\boldsymbol{u}}_j^{n+1} \in \mathcal{U}_{\text{ad}}.$$

Zhang-Shu's limiter applies.

Similar approach used by Rossmanith Et Al. [6].

We can also aposteriorily modify α_e (Gassner and Ramírez [14]).

Low order residual

$$(\tilde{\boldsymbol{u}}_1^e)^{n+1} = (\boldsymbol{u}_1^e)^n - \frac{\Delta t}{w_1 \, \Delta x_e}\Big[f_{\frac{3}{2}} - F_{e-\frac{1}{2}} \Big],$$
$$(\tilde{\boldsymbol{u}}_j^e)^{n+1} = (\boldsymbol{u}_j^e)^n - \frac{\Delta t}{w_j \, \Delta x_e}\Big[f_{j+\frac{1}{2}} - f_{j-\frac{1}{2}} \Big], \qquad 2 \le j \le N,$$
$$(\tilde{\boldsymbol{u}}_N^e)^{n+1} = (\boldsymbol{u}_N^e)^n - \frac{\Delta t}{w_{N+1} \, \Delta x_e}\Big[F_{e+\frac{1}{2}} - f_{N-\frac{1}{2}} \Big].$$

By appropriate of $F_{e\pm\frac{1}{2}}$,

$$(\tilde{\boldsymbol{u}}_j^e)^{n+1} \in \mathcal{U}_{\text{ad}}, \qquad 1 \le j \le N+1.$$

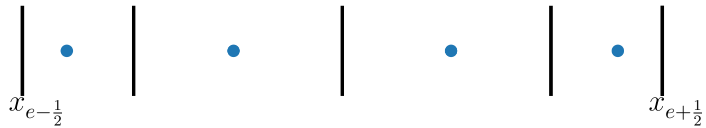
$$\Rightarrow \overline{\boldsymbol{u}}_e^{n+1} = \overline{\boldsymbol{u}}_e^n - \frac{\Delta t}{\Delta x_e}\Big(F_{e+\frac{1}{2}} - F_{e-\frac{1}{2}} \Big) = \sum_{j=1}^{N+1} w_j \, \tilde{\boldsymbol{u}}_j^{n+1} \in \mathcal{U}_{\text{ad}}.$$

Zhang-Shu's limiter applies.

Similar approach used by Rossmanith Et Al. [6].

We can also aposteriorily modify α_e (Gassner and Ramírez [14]).

Solution points and subcells



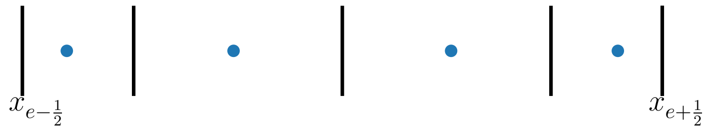
Low order residual : MUSCL-Hancock

Solution points and subcells

Integrate conservation law $I_j^e \times [t^n, t^{n+1}]$

$$\Delta x_e \, w_j \, (\boldsymbol{u}_j^{n+1} - \boldsymbol{u}_j^n) + \int_{t^n}^{t^{n+1}} (\boldsymbol{f}_{j+\frac{1}{2}} - \boldsymbol{f}_{j-\frac{1}{2}}) \, dt = 0.$$

Solution points and subcells



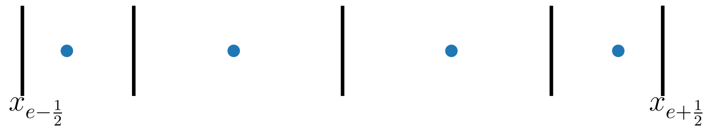
Integrate conservation law $I_j^e \times [t^n, t^{n+1}]$

$$\Delta x_e \, w_j \, (\boldsymbol{u}_j^{n+1} - \boldsymbol{u}_j^n) + \int_{t^n}^{t^{n+1}} (\boldsymbol{f}_{j+\frac{1}{2}} - \boldsymbol{f}_{j-\frac{1}{2}}) \, dt = 0.$$

Using the mid-point rule

$$\boldsymbol{u}_j^{n+1} = \boldsymbol{u}_j^n - \frac{\Delta t}{\Delta x_e w_j} \left(\boldsymbol{f}_{j+1/2}^{n+1/2} - \boldsymbol{f}_{j-1/2}^{n+1/2} \right)$$

Solution points and subcells



Integrate conservation law $I_j^e \times [t^n, t^{n+1}]$

$$\Delta x_e \, w_j \, (\boldsymbol{u}_j^{n+1} - \boldsymbol{u}_j^n) + \int_{t^n}^{t^{n+1}} (\boldsymbol{f}_{j+\frac{1}{2}} - \boldsymbol{f}_{j-\frac{1}{2}}) \, dt = 0.$$

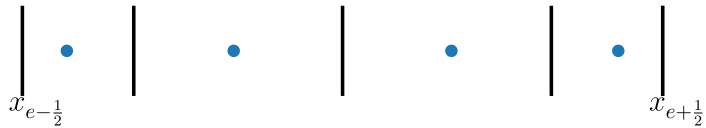
Using the mid-point rule

$$\boldsymbol{u}_j^{n+1} = \boldsymbol{u}_j^n - \frac{\Delta t}{\Delta x_e w_j} \left(\boldsymbol{f}_{j+1/2}^{n+1/2} - \boldsymbol{f}_{j-1/2}^{n+1/2} \right)$$

$$\boldsymbol{f}_{j+1/2}^{n+1/2} = \boldsymbol{f}\left(\boldsymbol{u}_{j+1/2-}^{n+1/2}, \boldsymbol{u}_{j+1/2+}^{n+1/2} \right)$$

Low order residual : MUSCL-Hancock

Solution points and subcells



Integrate conservation law $I_j^e \times [t^n, t^{n+1}]$

$$\Delta x_e \, w_j \, (\boldsymbol{u}_j^{n+1} - \boldsymbol{u}_j^n) + \int_{t^n}^{t^{n+1}} (\boldsymbol{f}_{j+\frac{1}{2}} - \boldsymbol{f}_{j-\frac{1}{2}}) \, dt = 0.$$

Using the mid-point rule

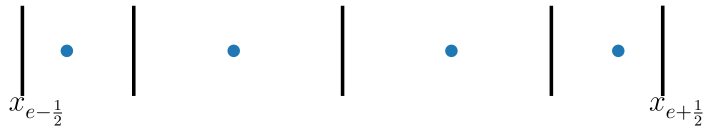
$$\boldsymbol{u}_j^{n+1} = \boldsymbol{u}_j^n - \frac{\Delta t}{\Delta x_e w_j} \left(\boldsymbol{f}_{j+1/2}^{n+1/2} - \boldsymbol{f}_{j-1/2}^{n+1/2} \right)$$

$$\boldsymbol{f}_{j+1/2}^{n+1/2} = \boldsymbol{f}\left(\boldsymbol{u}_{j+1/2-}^{n+1/2}, \boldsymbol{u}_{j+1/2+}^{n+1/2} \right)$$

$$\boldsymbol{u}_{j-\frac{1}{2}+}^{n+1/2} = \boldsymbol{u}_{j-\frac{1}{2}+}^n - \frac{\Delta t}{2} \frac{f(\boldsymbol{u}_{j+\frac{1}{2}+}) - f(\boldsymbol{u}_{j-\frac{1}{2}-})}{x_{j+\frac{1}{2}} - x_{j-\frac{1}{2}}}, \quad \boldsymbol{u}_{j+\frac{1}{2}-}^{n+1/2} = \boldsymbol{u}_{j+\frac{1}{2}-}^n - \frac{\Delta t}{2} \frac{f(\boldsymbol{u}_{j+\frac{1}{2}-}) - f(\boldsymbol{u}_{j-\frac{1}{2}+})}{x_{j+\frac{1}{2}} - x_{j-\frac{1}{2}}}.$$

Low order residual : MUSCL-Hancock

Solution points and subcells



Integrate conservation law $I_j^e \times [t^n, t^{n+1}]$

$$\Delta x_e \, w_j \, (\boldsymbol{u}_j^{n+1} - \boldsymbol{u}_j^n) + \int_{t^n}^{t^{n+1}} (\boldsymbol{f}_{j+\frac{1}{2}} - \boldsymbol{f}_{j-\frac{1}{2}}) \, dt = 0.$$

Using the mid-point rule

$$\boldsymbol{u}_j^{n+1} = \boldsymbol{u}_j^n - \frac{\Delta t}{\Delta x_e w_j} \left(\boldsymbol{f}_{j+1/2}^{n+1/2} - \boldsymbol{f}_{j-1/2}^{n+1/2} \right)$$

$$\boldsymbol{f}_{j+1/2}^{n+1/2} = \boldsymbol{f}\left(\boldsymbol{u}_{j+1/2-}^{n+1/2}, \boldsymbol{u}_{j+1/2+}^{n+1/2} \right)$$

$$\boldsymbol{u}_{j-\frac{1}{2}+}^{n+1/2} = \boldsymbol{u}_{j-\frac{1}{2}+}^{n} - \frac{\Delta t}{2} \frac{f(\boldsymbol{u}_{j+\frac{1}{2}+}) - f(\boldsymbol{u}_{j-\frac{1}{2}-})}{x_{j+\frac{1}{2}} - x_{j-\frac{1}{2}}}, \quad \boldsymbol{u}_{j+\frac{1}{2}-}^{n+1/2} = \boldsymbol{u}_{j+\frac{1}{2}-}^{n} - \frac{\Delta t}{2} \frac{f(\boldsymbol{u}_{j+\frac{1}{2}-}) - f(\boldsymbol{u}_{j-\frac{1}{2}+})}{x_{j+\frac{1}{2}} - x_{j-\frac{1}{2}}}.$$

$$\boldsymbol{u}_{j-\frac{1}{2}+} = \boldsymbol{u}_j(x_{j-\frac{1}{2}}), \quad \boldsymbol{u}_{j+\frac{1}{2}-} = \boldsymbol{u}_j(x_{j+\frac{1}{2}})$$
$$\boldsymbol{u}_j(x) = \boldsymbol{u}_j^n + \boldsymbol{\sigma}_j \, (x - x_j)$$

Low order residual : MUSCL-Hancock

Solution points and subcells

Integrate conservation law $I_j^e \times [t^n, t^{n+1}]$

$$\Delta x_e \, w_j \, (\boldsymbol{u}_j^{n+1} - \boldsymbol{u}_j^n) + \int_{t^n}^{t^{n+1}} (\boldsymbol{f}_{j+\frac{1}{2}} - \boldsymbol{f}_{j-\frac{1}{2}}) \, dt = 0.$$

Using the mid-point rule

$$\boldsymbol{u}_j^{n+1} = \boldsymbol{u}_j^n - \frac{\Delta t}{\Delta x_e w_j} \left(\boldsymbol{f}_{j+1/2}^{n+1/2} - \boldsymbol{f}_{j-1/2}^{n+1/2} \right)$$

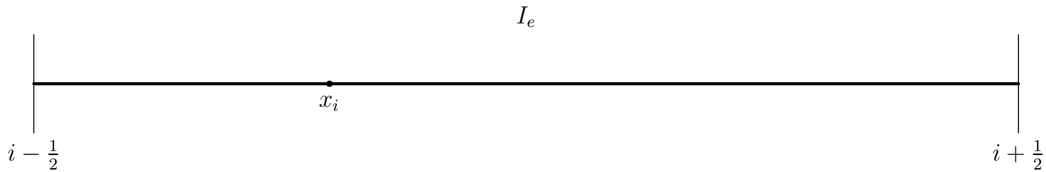
$$\boldsymbol{f}_{j+1/2}^{n+1/2} = \boldsymbol{f}\left(\boldsymbol{u}_{j+1/2-}^{n+1/2}, \boldsymbol{u}_{j+1/2+}^{n+1/2} \right)$$

$$\boldsymbol{u}_{j-\frac{1}{2}+}^{n+1/2} = \boldsymbol{u}_{j-\frac{1}{2}+}^{n} - \frac{\Delta t}{2} \frac{f(\boldsymbol{u}_{j+\frac{1}{2}+}) - f(\boldsymbol{u}_{j-\frac{1}{2}-})}{x_{j+\frac{1}{2}} - x_{j-\frac{1}{2}}}, \quad \boldsymbol{u}_{j+\frac{1}{2}-}^{n+1/2} = \boldsymbol{u}_{j+\frac{1}{2}-}^{n} - \frac{\Delta t}{2} \frac{f(\boldsymbol{u}_{j+\frac{1}{2}-}) - f(\boldsymbol{u}_{j-\frac{1}{2}+})}{x_{j+\frac{1}{2}} - x_{j-\frac{1}{2}}}.$$

$$\boldsymbol{u}_{j-\frac{1}{2}+} = \boldsymbol{u}_j(x_{j-\frac{1}{2}}), \quad \boldsymbol{u}_{j+\frac{1}{2}-} = \boldsymbol{u}_j(x_{j+\frac{1}{2}})$$
$$\boldsymbol{u}_j(x) = \boldsymbol{u}_j^n + \boldsymbol{\sigma}_j \, (x - x_j)$$

$$\boldsymbol{\sigma}_j = \operatorname{minmod}\left(\beta_e \frac{\boldsymbol{u}_{j+1} - \boldsymbol{u}_j}{x_{j+1} - x_j}, D_{\text{cent}}(\boldsymbol{u})_j, \beta_e \frac{\boldsymbol{u}_j^n - \boldsymbol{u}_{j-1}^n}{x_j - x_{j-1}} \right)$$
$$\beta_e = 2 - \alpha_e$$

Admissibility of low order method



Theorem. (*Extension of Berthon* [3]) *Consider the conservation law*

$$\boldsymbol{u}_t + \boldsymbol{f}(\boldsymbol{u})_x = 0$$

which preserves a convex set Ω*. Let* $\{\boldsymbol{u}_i^n\}_{i\in\mathbb{Z}}$ *be the approximate solution at time level* n *and assume that* $\boldsymbol{u}_i^n \in \Omega$ *for all* $i \in \mathbb{Z}$*. Consider conservative reconstructions*

$$\boldsymbol{u}_i^{n,+} = \boldsymbol{u}_i + (x_{i+1/2} - x_i)\boldsymbol{\sigma}_i, \quad \boldsymbol{u}_i^{n,-} = \boldsymbol{u}_i + (x_{i-1/2} - x_i)\boldsymbol{\sigma}_i.$$

Define $\boldsymbol{u}_i^{*,\pm}$ *satisfying*

$$\mu^- \boldsymbol{u}_i^{n,-} + \boldsymbol{u}_i^{*,\pm} + \mu^+ \boldsymbol{u}_i^{n,+} = 2\boldsymbol{u}_i^{n,\pm},$$

where

$$\mu^- = \frac{x_{i+1/2} - x_i}{x_{i+1/2} - x_{i-1/2}}, \quad \mu^+ = \frac{x_i - x_{i-1/2}}{x_{i+1/2} - x_{i-1/2}}.$$

Assume that the slope $\boldsymbol{\sigma}_i$ *is chosen so that*

$$\boldsymbol{u}_i^{*,\pm} \in \mathcal{U}_{\mathrm{ad}}.$$

Then, under appropriate time step restrictions, the updated solution $\boldsymbol{u}_i^{n+1}$*, defined by the MUSCL-Hancock procedure is in* $\mathcal{U}_{\mathrm{ad}}$*.*

Berthon defined $\boldsymbol{u}_i^{*,\pm}$

$$\frac{1}{2}\boldsymbol{u}_i^{n,-} + \boldsymbol{u}_i^{*,\pm} + \frac{1}{2}\boldsymbol{u}_i^{n,+} = 2\boldsymbol{u}_i^{n,\pm}.$$

Berthon defined $\boldsymbol{u}_i^{*,\pm}$

$$\frac{1}{2}\boldsymbol{u}_i^{n,-} + \boldsymbol{u}_i^{*,\pm} + \frac{1}{2}\boldsymbol{u}_i^{n,+} = 2\boldsymbol{u}_i^{n,\pm}.$$

Our generalization

$$\mu^- \, \boldsymbol{u}_i^{n,-} + \boldsymbol{u}_i^{*,\pm} + \mu^+ \, \boldsymbol{u}_i^{n,+} = 2\boldsymbol{u}_i^{n,\pm},$$

where

$$\mu^- = \frac{x_{i+1/2} - x_i}{x_{i+1/2} - x_{i-1/2}}, \qquad \mu^+ = \frac{x_i - x_{i-1/2}}{x_{i+1/2} - x_{i-1/2}}.$$

Berthon defined $\boldsymbol{u}_i^{*,\pm}$

$$\frac{1}{2}\boldsymbol{u}_i^{n,-} + \boldsymbol{u}_i^{*,\pm} + \frac{1}{2}\boldsymbol{u}_i^{n,+} = 2\boldsymbol{u}_i^{n,\pm}.$$

Our generalization

$$\mu^- \boldsymbol{u}_i^{n,-} + \boldsymbol{u}_i^{*,\pm} + \mu^+ \boldsymbol{u}_i^{n,+} = 2\boldsymbol{u}_i^{n,\pm},$$

where

$$\mu^- = \frac{x_{i+1/2} - x_i}{x_{i+1/2} - x_{i-1/2}}, \quad \mu^+ = \frac{x_i - x_{i-1/2}}{x_{i+1/2} - x_{i-1/2}}.$$

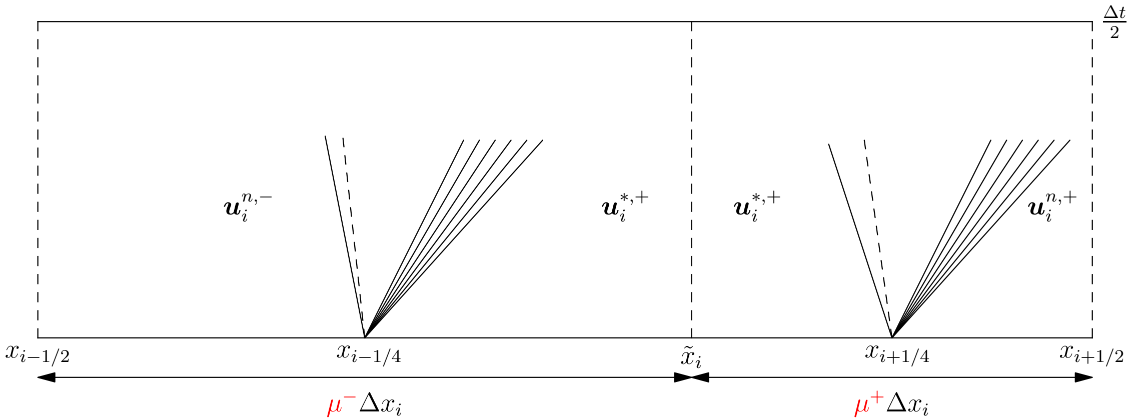
For **conservative reconstruction**

$$\boldsymbol{u}_i^{*,\pm} = \boldsymbol{u}_i^n + 2(x_{i\pm1/2} - x_i)\boldsymbol{\sigma}_i,$$

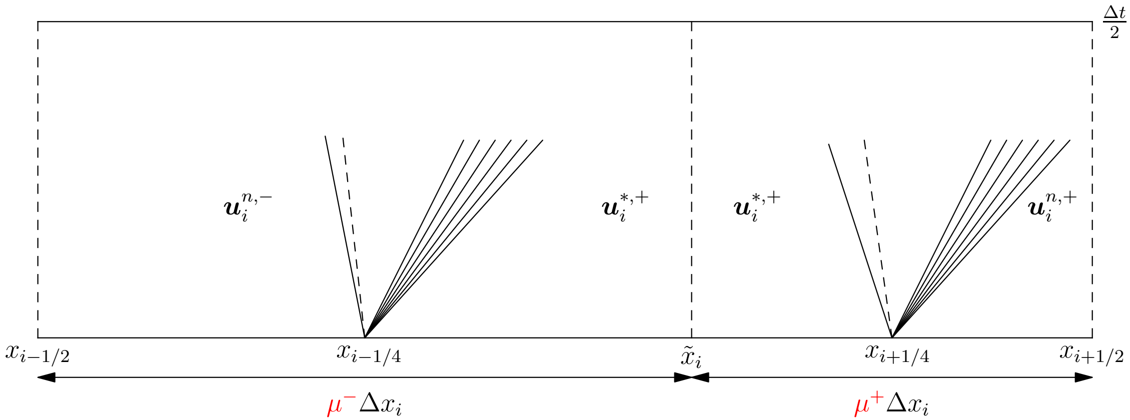
noting that

$$\boldsymbol{u}_i^{n,+} = \boldsymbol{u}_i + (x_{i+1/2} - x_i)\boldsymbol{\sigma}_i, \quad \boldsymbol{u}_i^{n,-} = \boldsymbol{u}_i + (x_{i-1/2} - x_i)\boldsymbol{\sigma}_i.$$

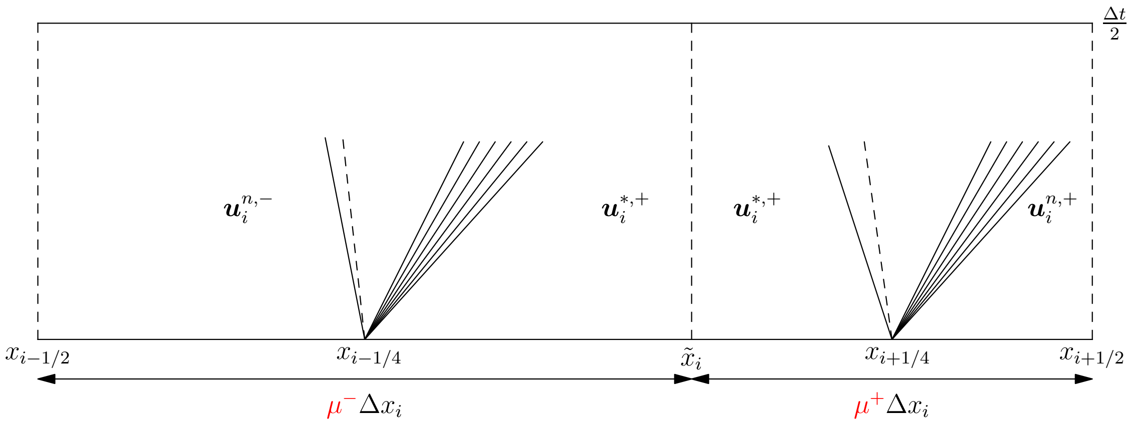
Idea of proof



Idea of proof

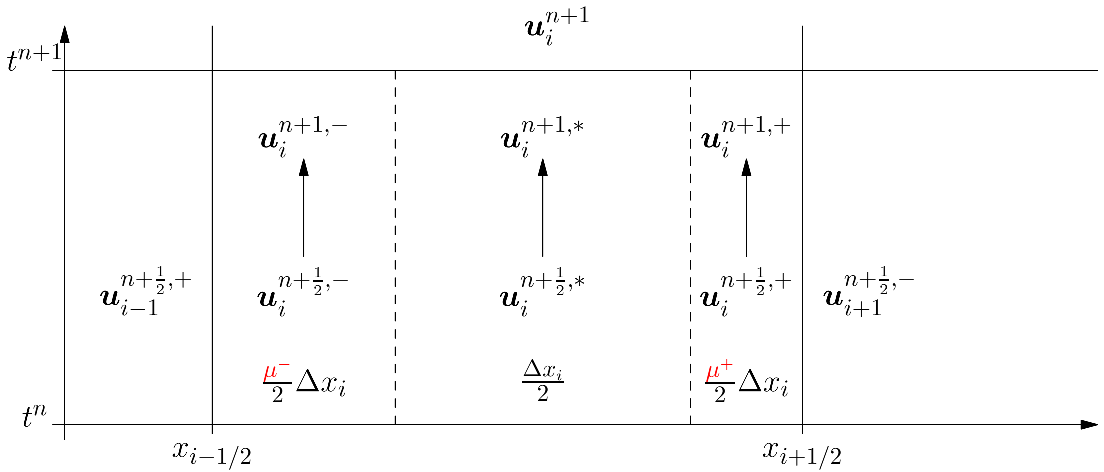


$$\frac{1}{\Delta x_i}\int_{x_{i-1/2}}^{x_{i+1/2}} \boldsymbol{u}^h(x, \Delta t/2)\, dx = \boldsymbol{u}_i^{n+\frac{1}{2},+}$$



$$\frac{1}{\Delta x_i}\int_{x_{i-1/2}}^{x_{i+1/2}} \boldsymbol{u}^h(x,\Delta t/2)\,dx = \boldsymbol{u}_i^{n+\frac{1}{2},+}$$

$$\frac{\mu^-}{2}\boldsymbol{u}_i^{n+\frac{1}{2},-} + \frac{1}{2}\boldsymbol{u}_i^{n+\frac{1}{2},*} + \frac{\mu^+}{2}\boldsymbol{u}_i^{n+\frac{1}{2},+} = \boldsymbol{u}_i^n.$$



Given candidate slope $\boldsymbol{\sigma}_i$,

$$\boldsymbol{u}_i^{*,\pm} := \boldsymbol{u}_i^n + 2\left(x_{i\pm 1/2} - x_i\right)\boldsymbol{\sigma}_i.$$

Find $\theta \in [0, 1]$

$$\boldsymbol{u}_i^n + 2\left(x_{i\pm 1/2} - x_i\right)\theta\,\boldsymbol{\sigma}_i \in \mathcal{U}_{\mathrm{ad}}$$

by Zhang-Shu type procedure.

Current solution

$$x \in I_e: \qquad u_h^n(\xi) = \sum_{j=1}^{N+1} u_j^n \ell_j(\xi)$$

Current solution

$$x \in I_e: \qquad u_h^n(\xi) = \sum_{j=1}^{N+1} u_j^n \ell_j(\xi)$$

Cell local space-time solution and flux: $\tau = (t - t_n)/\Delta t$

$$\tilde{u}_h(\xi,\tau) = \sum_{r,s=1}^{N+1} \tilde{u}_{rs}\ell_r(\xi)\ell_s(\tau), \quad \tilde{f}_h(\xi,\tau) = \sum_{r,s=1}^{N+1} f(\tilde{u}_{rs})\ell_r(\xi)\ell_s(\tau).$$

Current solution

$$x \in I_e: \qquad u_h^n(\xi) = \sum_{j=1}^{N+1} u_j^n \ell_j(\xi)$$

Cell local space-time solution and flux: $\tau = (t - t_n)/\Delta t$

$$\tilde{u}_h(\xi, \tau) = \sum_{r,s=1}^{N+1} \tilde{u}_{rs}\ell_r(\xi)\ell_s(\tau), \quad \tilde{f}_h(\xi, \tau) = \sum_{r,s=1}^{N+1} f(\tilde{u}_{rs})\ell_r(\xi)\ell_s(\tau).$$

Find $\tilde{u}_h$ by cell local Galerkin method

$$\int_{t_n}^{t_{n+1}} \int_{I_e} (\partial_t \tilde{u}_h + \partial_x \tilde{f}_h)\, \ell_r(\xi)\ell_s(\tau)\, dx\, dt, \qquad 1 \le r, s \le N+1.$$

Current solution

$$x \in I_e: \qquad u_h^n(\xi) = \sum_{j=1}^{N+1} u_j^n \ell_j(\xi)$$

Cell local space-time solution and flux: $\tau = (t - t_n) / \Delta t$

$$\tilde{u}_h(\xi, \tau) = \sum_{r,s=1}^{N+1} \tilde{u}_{rs} \ell_r(\xi) \ell_s(\tau), \quad \tilde{f}_h(\xi, \tau) = \sum_{r,s=1}^{N+1} f(\tilde{u}_{rs}) \ell_r(\xi) \ell_s(\tau).$$

Find $\tilde{u}_h$ by cell local Galerkin method

$$\int_{t_n}^{t_{n+1}} \int_{I_e} (\partial_t \tilde{u}_h + \partial_x \tilde{f}_h) \, \ell_r(\xi) \ell_s(\tau) \, dx \, dt, \qquad 1 \le r, s \le N+1.$$

Integrate by parts in time

$$\begin{gathered} -\int_{t_n}^{t_{n+1}} \int_{I_e} \tilde{u}_h \ell_r(\xi) \partial_t \ell_s(\tau) \, dx \, dt + \int_{I_e} \tilde{u}_h(\xi, 1) \ell_r(\xi) \ell_s(1) \, dx - \int_{I_e} u_h^n(\xi) \ell_r(\xi) \ell_s(0) \, d\xi \\ + \int_{t_n}^{t_{n+1}} \int_{I_e} \partial_x \tilde{f}_h \, \ell_r(\xi) \ell_s(\tau) dx \, dt = 0. \end{gathered}$$

$$\int_{t_n}^{t_{n+1}} \int_{I_e} (\partial_t u_h + \partial_x f(u_h)) \ell_i(\xi) \, dx \, dt = 0$$

$$\int_{t_n}^{t_{n+1}} \int_{I_e} (\partial_t u_h + \partial_x f(u_h)) \ell_i(\xi)\, dx\, dt = 0$$

Integrate by parts in space

$$\begin{aligned}\int_{I_e} u_h^{n+1} \ell_i(\xi)\, dx \;&=\; \int_{I_e} u_h^n\, \ell_i(\xi)\, dx + \int_{t^n}^{t^{n+1}} \int_{I_e} \tilde{f}_h \partial_x \ell_i(\xi)\, dx\, dt \\ &+ \int_{t^n}^{t^{n+1}} f(\tilde{u}^-_{e-\frac{1}{2}}(t), \tilde{u}^+_{e-\frac{1}{2}}(t)) \ell_i(0) dt - \int_{t^n}^{t^{n+1}} f(\tilde{u}^-_{e+\frac{1}{2}}(t), \tilde{u}^+_{e+\frac{1}{2}}(t)) \ell_i(1) dt.\end{aligned}$$

$$\int_{t_n}^{t_{n+1}} \int_{I_e} (\partial_t u_h + \partial_x f(u_h)) \ell_i(\xi) \, dx \, dt = 0$$

Integrate by parts in space

$$\begin{aligned} \int_{I_e} u_h^{n+1} \ell_i(\xi) \, dx &= \int_{I_e} u_h^n \, \ell_i(\xi) \, dx + \int_{t^n}^{t^{n+1}} \int_{I_e} \tilde{f}_h \partial_x \ell_i(\xi) \, dx \, dt \\ &\quad + \int_{t^n}^{t^{n+1}} f(\tilde{u}^-_{e-\frac{1}{2}}(t), \tilde{u}^+_{e-\frac{1}{2}}(t)) \ell_i(0) dt - \int_{t^n}^{t^{n+1}} f(\tilde{u}^-_{e+\frac{1}{2}}(t), \tilde{u}^+_{e+\frac{1}{2}}(t)) \ell_i(1) dt. \end{aligned}$$

FR form

$$\begin{aligned} & u_i^n - \int_{t^n}^{t^{n+1}} \partial_x \tilde{f}_h(\xi_i, \tau) \, dt \\ \Rightarrow \quad u_i^{n+1} \;=\; & - g_L'(\xi_i) \int_{t^n}^{t^{n+1}} f(\tilde{u}^-_{e-\frac{1}{2}}(t), \tilde{u}^+_{e-\frac{1}{2}}(t)) - \tilde{f}_h(0, \tau) dt \\ & - g_R'(\xi_i) \int_{t^n}^{t^{n+1}} f(\tilde{u}^-_{e+\frac{1}{2}}(t), \tilde{u}^+_{e+\frac{1}{2}}(t)) - \tilde{f}_h(1, \tau) dt \end{aligned}$$

For

$$u_t + u_x = 0,$$

ADER update

$$u_i^{n+1} = \begin{aligned} & u_i^n - \partial_x \int_{t^n}^{t^{n+1}} \tilde{u}_h(\xi_i, \tau)\, dt \\ & - g_L'(\xi_i)\left[f\left(\int_{t^n}^{t^{n+1}} \tilde{u}^-_{e-\frac{1}{2}}(t)\, dt, \int_{t^n}^{t^{n+1}} \tilde{u}^+_{e-\frac{1}{2}}(t)\, dt \right) - \int_{t^n}^{t^{n+1}} \tilde{u}_h(0,\tau) dt \right] \\ & - g_R'(\xi_i)\left[f\left(\int_{t^n}^{t^{n+1}} \tilde{u}^-_{e+\frac{1}{2}}(t)\, dt, \int_{t^n}^{t^{n+1}} \tilde{u}^+_{e+\frac{1}{2}}(t)\, dt \right) - \int_{t^n}^{t^{n+1}} \tilde{u}_h(1,\tau)\, dt \right] \end{aligned} \tag{1}$$

For

$$u_t + u_x = 0,$$

ADER update

$$u_i^{n+1} = \begin{aligned}[t] & u_i^n - \partial_x \int_{t^n}^{t^{n+1}} \tilde{u}_h(\xi_i, \tau)\, dt \\ & - g_L'(\xi_i)\left[f\left(\int_{t^n}^{t^{n+1}} \tilde{u}^-_{e-\frac{1}{2}}(t)\, dt, \int_{t^n}^{t^{n+1}} \tilde{u}^+_{e-\frac{1}{2}}(t)\, dt \right) - \int_{t^n}^{t^{n+1}} \tilde{u}_h(0,\tau) dt \right] \\ & - g_R'(\xi_i)\left[f\left(\int_{t^n}^{t^{n+1}} \tilde{u}^-_{e+\frac{1}{2}}(t)\, dt, \int_{t^n}^{t^{n+1}} \tilde{u}^+_{e+\frac{1}{2}}(t)\, dt \right) - \int_{t^n}^{t^{n+1}} \tilde{u}_h(1,\tau)\, dt \right] \end{aligned} \tag{3}$$

LWFR-D2 update

$$u_i^{n+1} = u_i^n - \Delta t \left[\partial_x U_h(\xi_i) - g_L'(\xi_i) \left[f(U^-_{e-\frac{1}{2}}, U^+_{e-\frac{1}{2}}) - U_h(0) \right] - g_R'(\xi_i) \left[f(U^-_{e+\frac{1}{2}}, U^+_{e+\frac{1}{2}}) - U_h(1) \right] \right], \tag{4}$$

where

$$\begin{aligned} U_h^n &= u + \frac{\Delta t}{2} u_t + \frac{\Delta t^2}{3!} u_{tt} + \cdots + \frac{\Delta t^N}{(N+1)!} \frac{\partial^N u}{\partial t^N} \\ &= u - \frac{\Delta t}{2} u_x + \frac{\Delta t^2}{3!} u_{xx} + \cdots + (-1)^N \frac{\Delta t^N}{(N+1)!} \frac{\partial^N u}{\partial x^N}. \end{aligned}$$

Theorem 1. *For the linear advection equation*

$$u_t + u_x = 0,$$

the Lax-Wendroff Flux Reconstruction scheme with D2 dissipation and ADER-FR scheme are equivalent.

Theorem 2. *For the linear advection equation*

$$u_t + u_x = 0,$$

the Lax-Wendroff Flux Reconstruction scheme with D2 dissipation and ADER-FR scheme are equivalent.

Proof. Let $u_e^n = u_e^n(x)$ denote the solution polynomial at time level n in element e.

Theorem 3. *For the linear advection equation*

$$u_t + u_x = 0,$$

the Lax-Wendroff Flux Reconstruction scheme with D2 dissipation and ADER-FR scheme are equivalent.

Proof. Let $u_e^n = u_e^n(x)$ denote the solution polynomial at time level n in element e.

Then, $\tilde{u}_h(x,t) := u_e^n(x - (t - t^n))$ is a weak solution of the equation

$$\begin{aligned}
\tilde{u}_t + \tilde{u}_x &= 0, \qquad x \in [x_{e-1/2}, x_{e+1/2}], t \in (t_n, t_{n+1}],\\
\tilde{u}(x, t^n) &= u_e^n(x), \qquad t = t_n.
\end{aligned}$$

Theorem 4. *For the linear advection equation*

$$u_t + u_x = 0,$$

the Lax-Wendroff Flux Reconstruction scheme with D2 dissipation and ADER-FR scheme are equivalent.

Proof. Let $u_e^n = u_e^n(x)$ denote the solution polynomial at time level n in element e.

Then, $\tilde{u}_h(x,t) := u_e^n(x - (t - t^n))$ is a weak solution of the equation

$$\begin{aligned} \tilde{u}_t + \tilde{u}_x &= 0, & x \in [x_{e-1/2}, x_{e+1/2}], t \in (t_n, t_{n+1}], \\ \tilde{u}(x, t^n) &= u_e^n(x), & t = t_n. \end{aligned}$$

Since the predictor equation has a unique solution of degree N [10, 5], the specified $\tilde{u}_h$ must be the predictor solution.

Theorem 5. *For the linear advection equation*

$$u_t + u_x = 0,$$

the Lax-Wendroff Flux Reconstruction scheme with D2 dissipation and ADER-FR scheme are equivalent.

Proof. Let $u_e^n = u_e^n(x)$ denote the solution polynomial at time level n in element e.

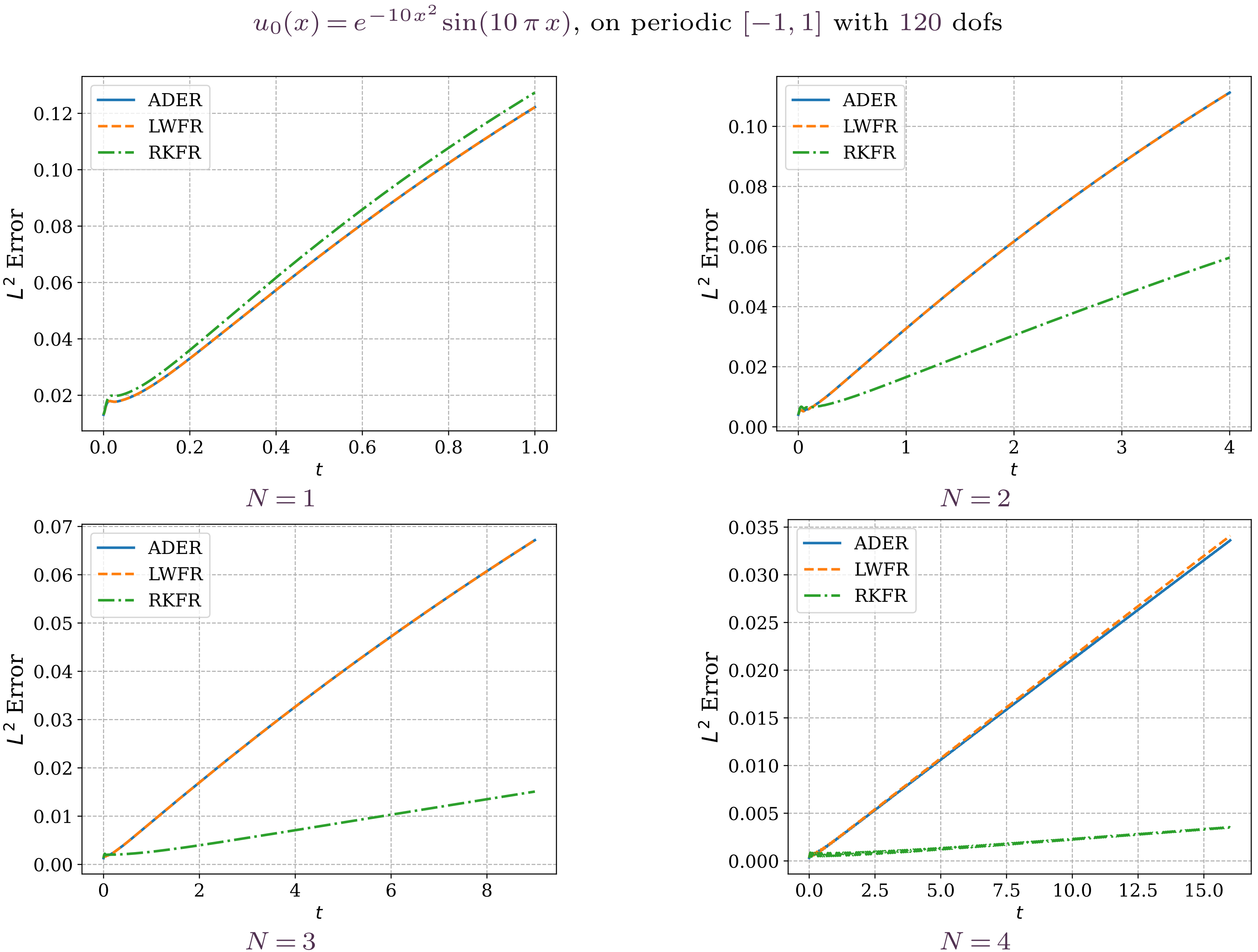
Then, $\tilde{u}_h(x,t) := u_e^n(x - (t - t^n))$ is a weak solution of the equation

$$\begin{aligned} \tilde{u}_t + \tilde{u}_x &= 0, \qquad x \in [x_{e-1/2}, x_{e+1/2}], t \in (t_n, t_{n+1}], \\ \tilde{u}(x, t^n) &= u_e^n(x), \qquad t = t_n. \end{aligned}$$

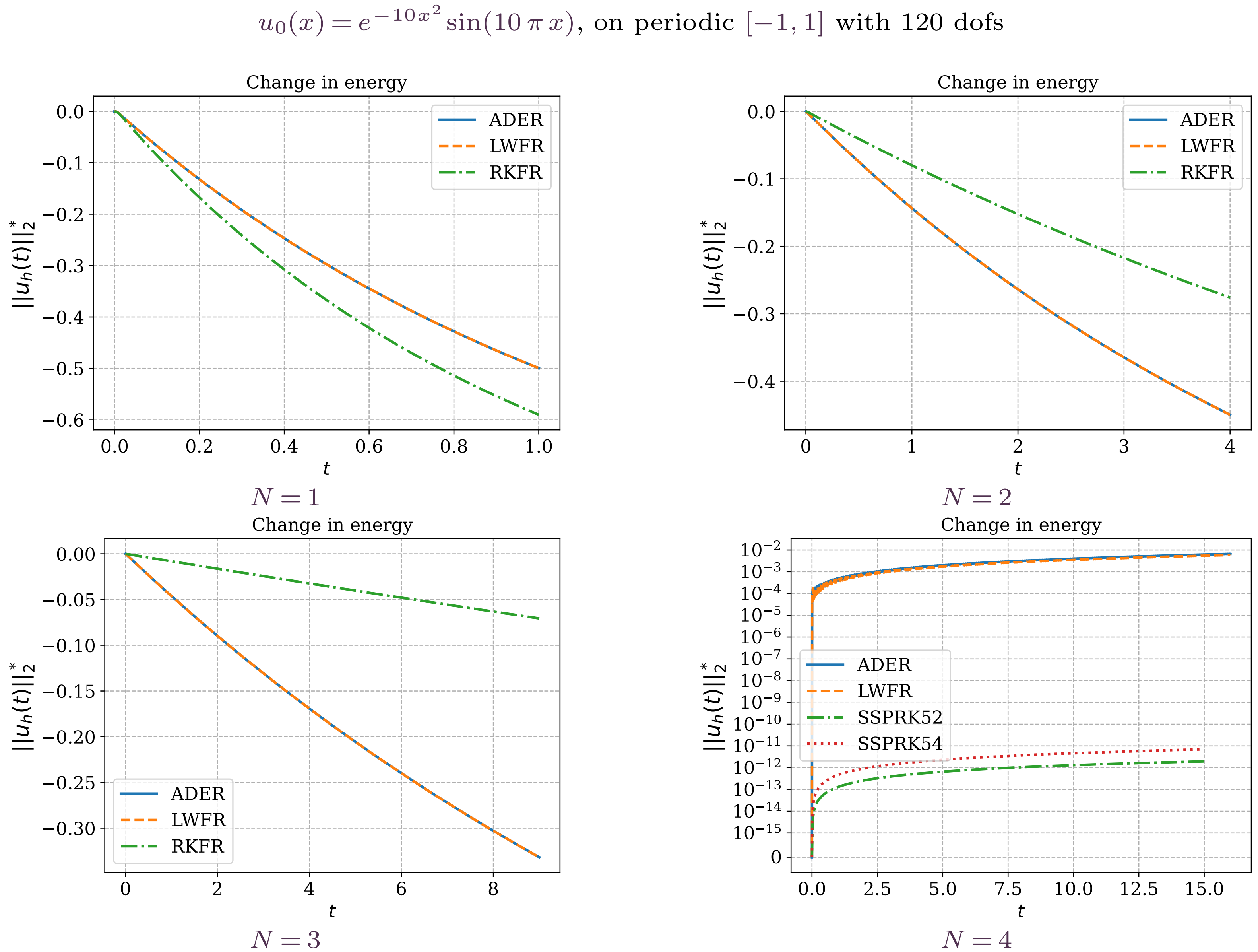
Since the predictor equation has a unique solution of degree N [10, 5], the specified $\tilde{u}_h$ must be the predictor solution.

$$\begin{aligned} \tilde{u}_h(x,t) &= \tilde{u}_h(x,t^n) + (t-t^n)\frac{\partial}{\partial t}\tilde{u}_h(x,t^n) + \ldots + \frac{(t-t^n)^N}{N!}\frac{\partial^N}{\partial t^N}\tilde{u}_h(x,t^n) \\ &= u^n(x) - (t-t^n)\frac{\partial}{\partial x}u^n(x) + \ldots + (-1)^N\frac{(t-t^n)^N}{N!}\frac{\partial^N}{\partial x^N}u^n(x) \\ \Rightarrow \quad \frac{1}{\Delta t}\int_{t^n}^{t^{n+1}} \tilde{u}_h(x,t)\,dt &= u^n(x) - \frac{\Delta t}{2}\frac{\partial}{\partial x}u^n(x) + \ldots + (-1)^N\frac{\Delta t^N}{(N+1)!}\frac{\partial^N}{\partial x^N}u^n dt, \\ \Rightarrow \quad \frac{1}{\Delta t}\int_{t^n}^{t^{n+1}} \tilde{u}_h(x,t)\,dt &= U_h^n(x) \end{aligned}$$

Thus, the ADER update (3) and the LWFR-D2 update (4) are the same. $\square$



$u_0(x) = e^{-10x^2}\sin(10\,\pi\,x)$, on periodic $[-1,1]$ with 120 dofs
ADER
LWFR
RKFR
L^2 Error
t
$N = 1$
$N = 2$
$N = 3$
$N = 4$



$u_0(x) = e^{-10x^2}\sin(10\,\pi\,x)$, on periodic $[-1,1]$ with 120 dofs
Change in energy
ADER
LWFR
RKFR
$||u_h(t)||_2^*$
t
$N=1$
Change in energy
ADER
LWFR
RKFR
$||u_h(t)||_2^*$
t
$N=2$
Change in energy
ADER
LWFR
RKFR
$||u_h(t)||_2^*$
t
$N=3$
Change in energy
ADER
LWFR
SSPRK52
SSPRK54
$||u_h(t)||_2^*$
t
$N=4$

Theorem 6. *If the predictor solution of ADER satisfies*

$$\|U^n\|_{C^{N+1}} \leq C$$

where C *is a constant independent of* $n, \Delta x, \Delta t$*, then the ADER and LW solution will satisfy*

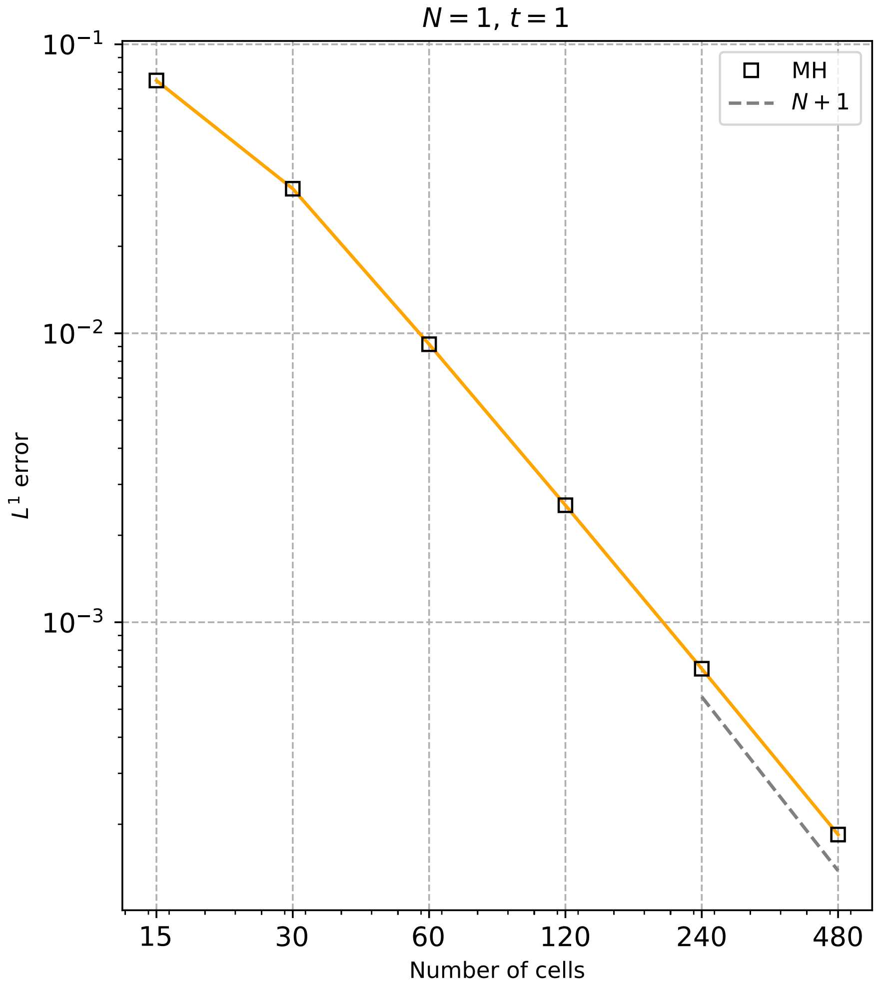
$$\|u_{\text{ADER}}^{n+1} - u_{\text{LW}}^{n+1}\|_\infty = O(\Delta t^{N+1}).$$

Numerical Results

$$\boldsymbol{u}_t + \boldsymbol{u}_x = \boldsymbol{0}$$

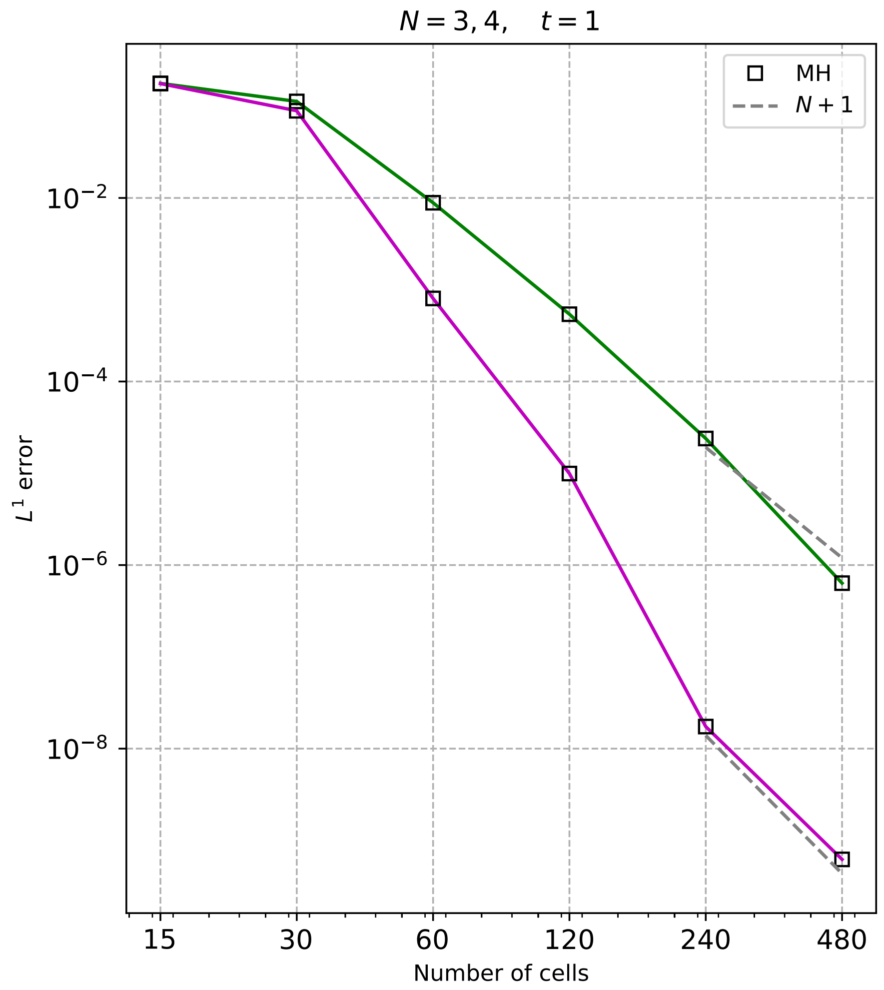
Convergence of pure MUSCL-Hancock method

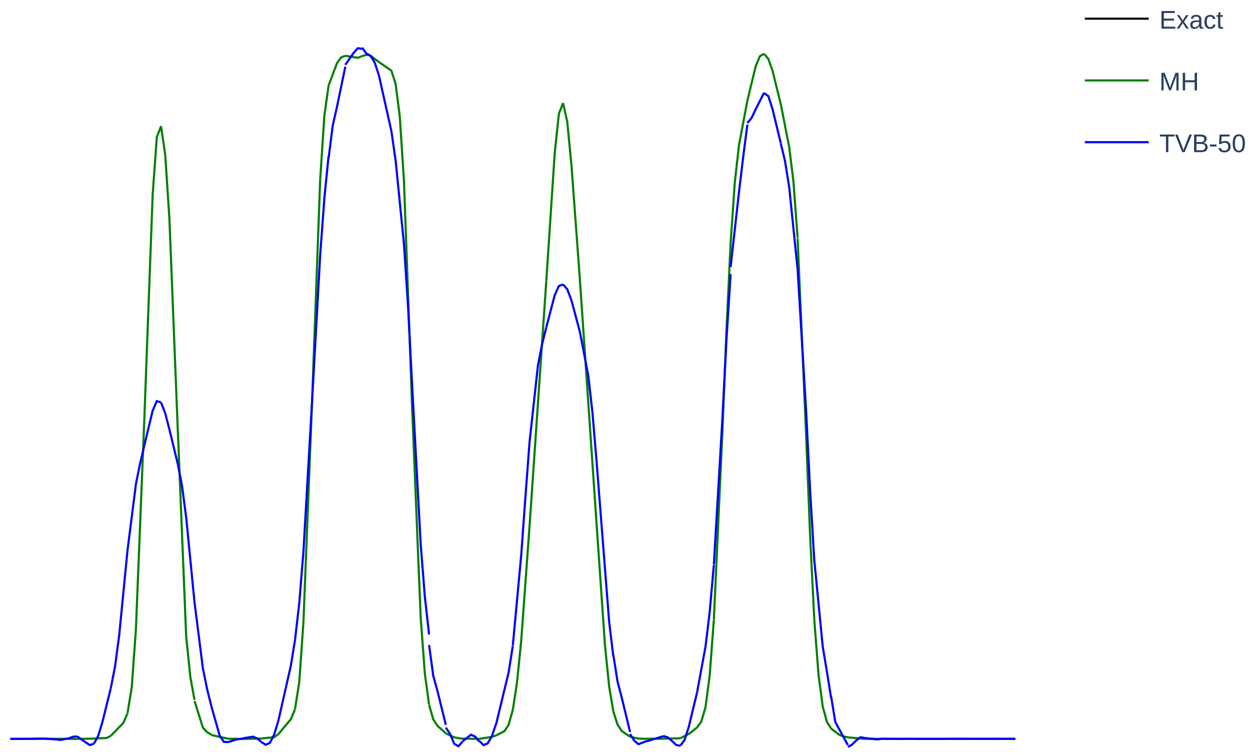
$$
\begin{aligned}
u_t + u_x &= 0, \qquad\qquad x \in [-1,1], \quad t > 0, \\
u_0(x) &= \sin(2\,\pi\,x).
\end{aligned}
$$



Optimal convergence with limiter

$$
\begin{aligned}
u_t + u_x &= 0, & x \in [-2,2], \quad t > 0,\\
u_0(x) &= e^{-10x^2}\sin(10\,\pi\,x), & x \in [-2,2].
\end{aligned}
$$

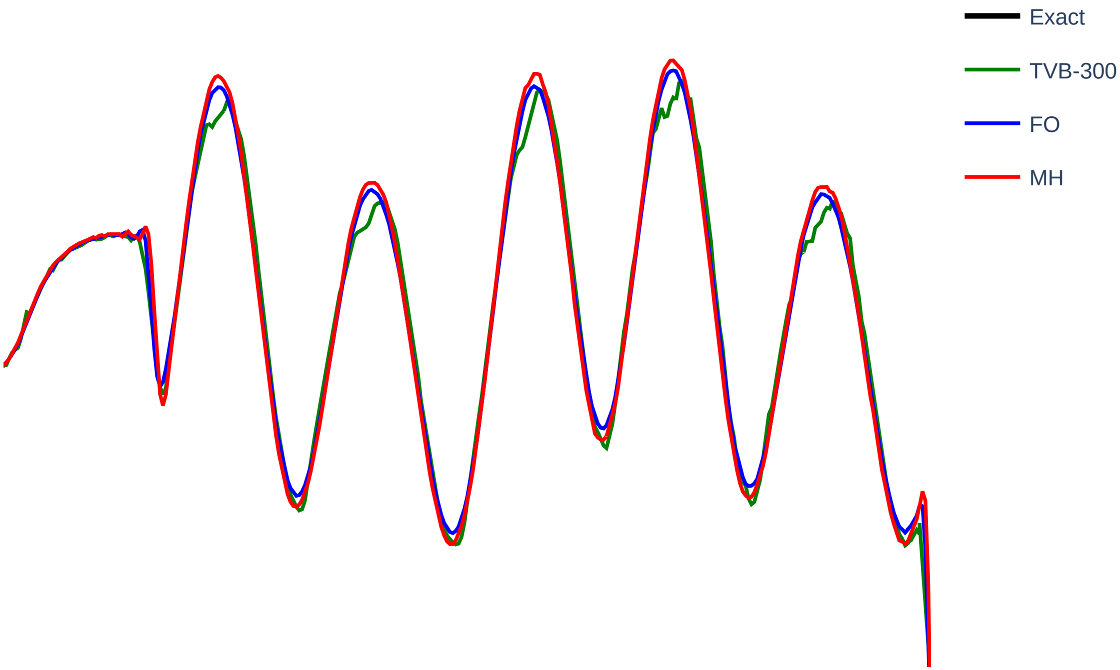




Indicator parameters - $\beta_1 = 0.1$, $\beta_2 = 1.0$

1-D Euler's equations

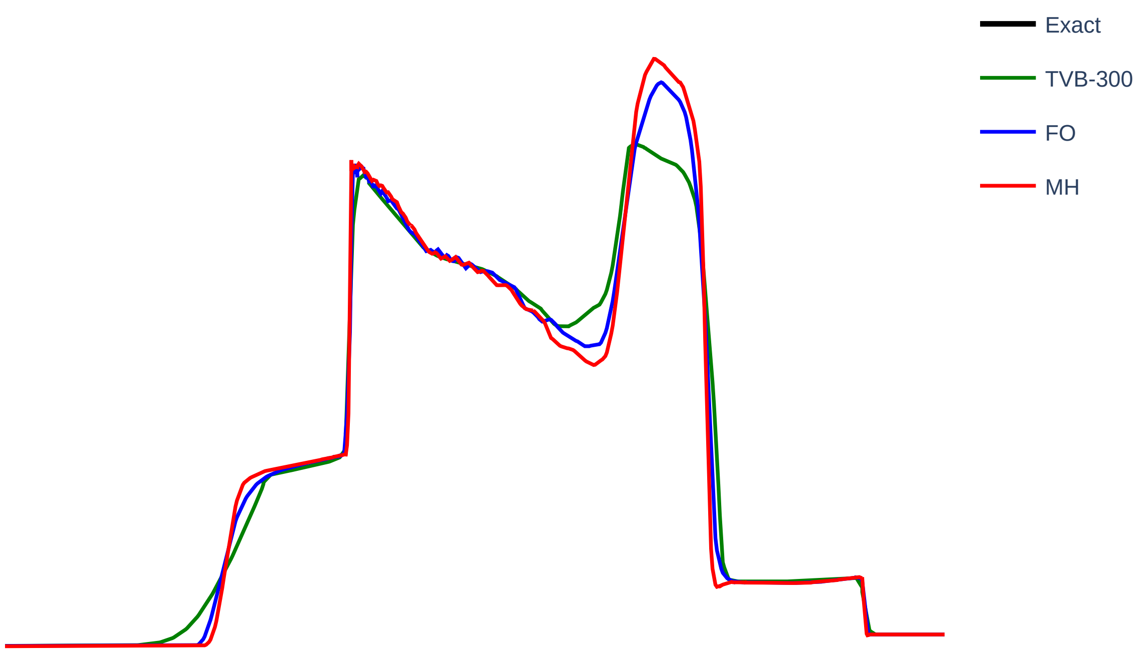
$$\begin{aligned} \rho &\longrightarrow \text{density} \\ v &\longrightarrow \text{velocity} \\ p &\longrightarrow \text{pressure} \end{aligned}$$



Transmissive boundary conditions on $[-5, 5]$ with

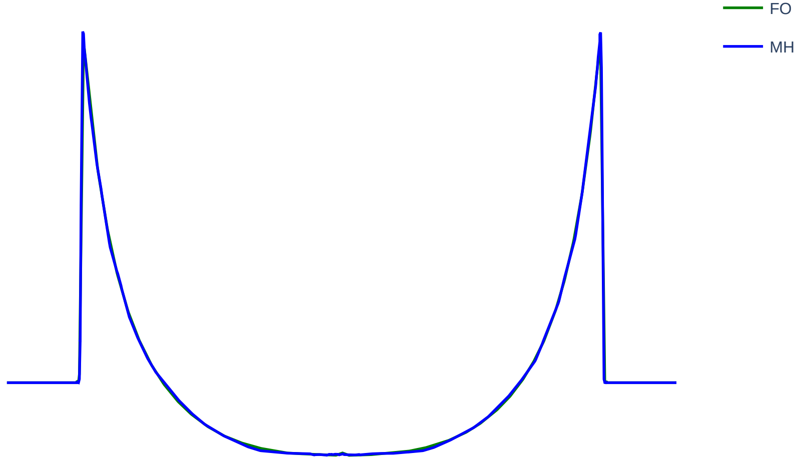
$$(\rho, v, p) = \begin{cases} (3.857143, 2.629369, 10.333333), & \text{if } x < -4, \\ (1 + 0.2\sin(5x), 0, 1), & \text{if } x \geq -4. \end{cases}$$

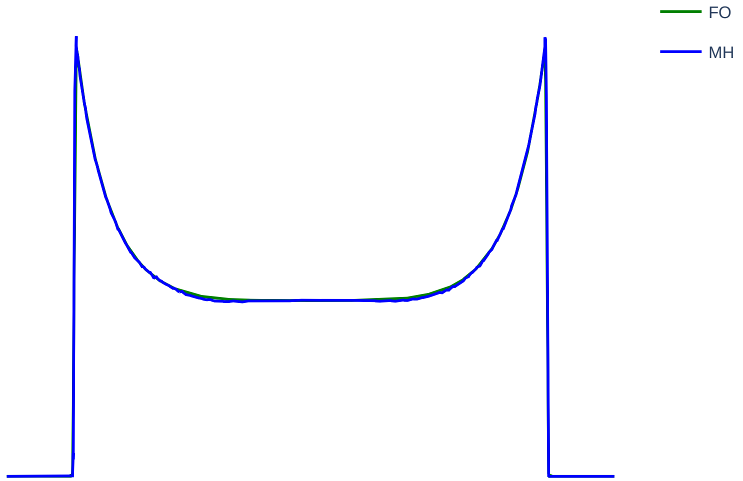
Blast wave [19]



Solid wall boundary conditions on $[0,1]$ with intial condition

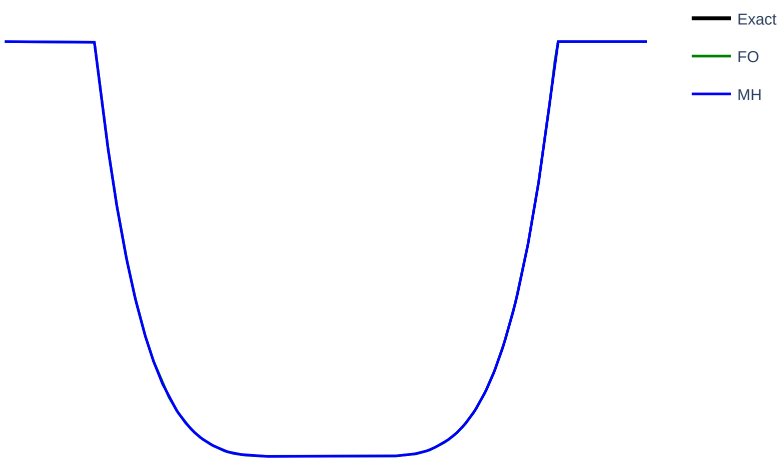
$$(\rho, v, p) = \begin{cases} (1, 0, 1000), & \text{if } x < 0.1, \\ (1, 0, 0.01), & 0.1 < x < 0.9, \\ (1, 0, 100), & x > 0.9. \end{cases}$$

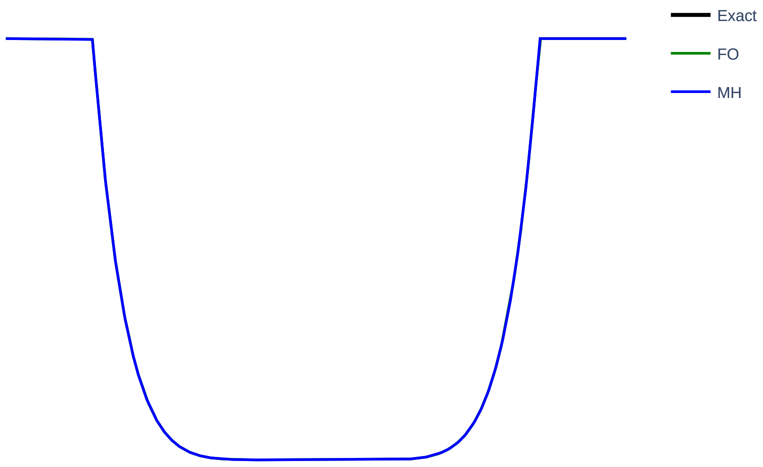




We impose transmissive boundary conditions with initial data

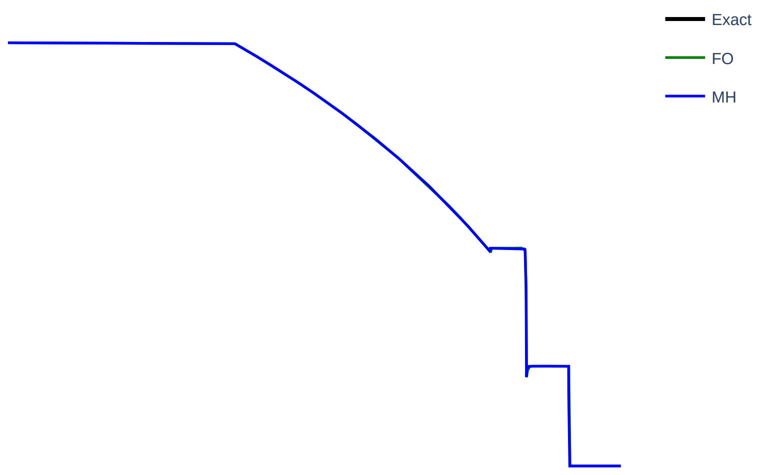
$$\rho(x) = 1, \quad v(x) = 0, \quad p(x) = \begin{cases} (\gamma - 1)\dfrac{3.2 \times 10^6}{\Delta x}, & |x| \leq \dfrac{\Delta x}{2}, \\ (\gamma - 1)10^{-12}, & \text{otherwise.} \end{cases}$$

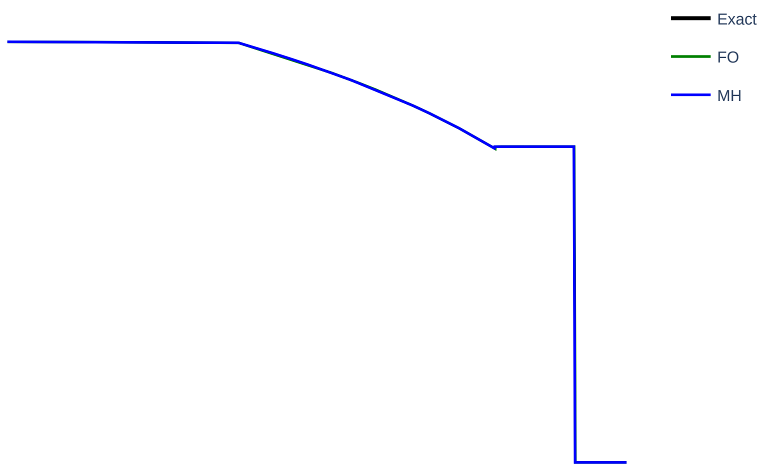




We impose transmissive boundary conditions with intial conditions

$$(\rho, v, p) = \begin{cases} (7, -1, 0.2), & \text{if } x < 0, \\ (7, 1, 0.2), & \text{if } x > 0. \end{cases}$$





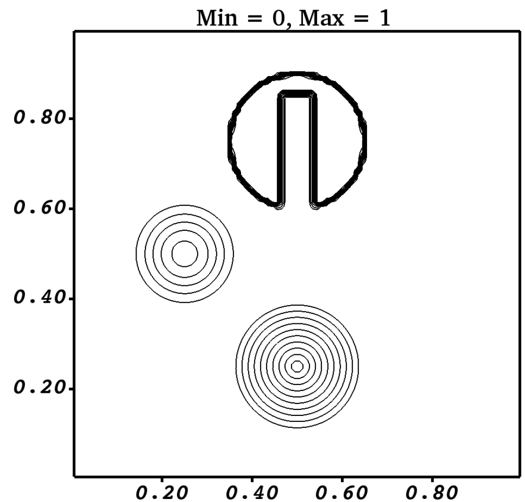
We impose transmissive boundary conditions with intial conditions

$$(\rho, v, p) = \begin{cases} (2, 0, 10^9), & \text{if } x < 0, \\ (0.001, 0, 1), & \text{if } x > 0. \end{cases}$$

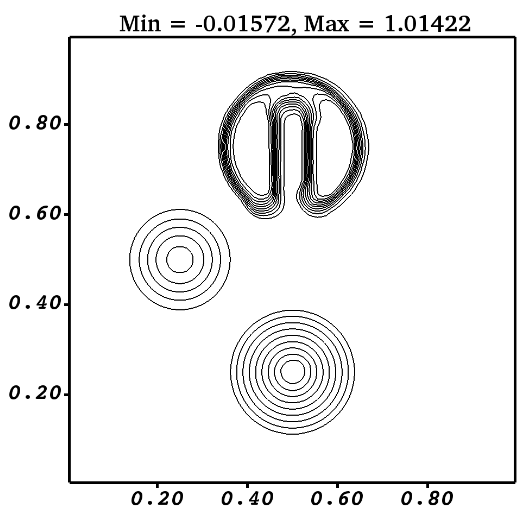
2-D Results

2-D Composite signal [11]

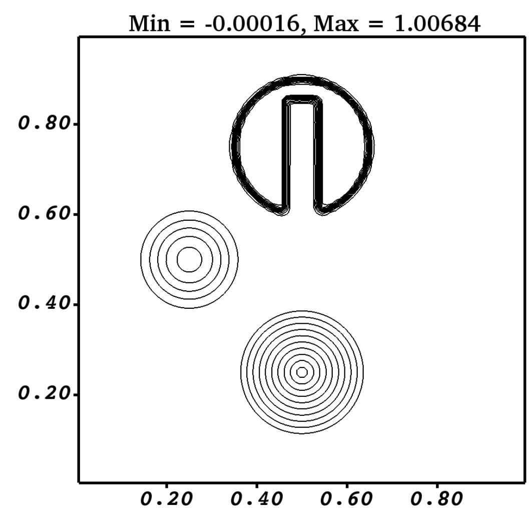
$t = 2\,\pi, \quad \mathrm{NC} = 100 \times 100, \quad N = 4$



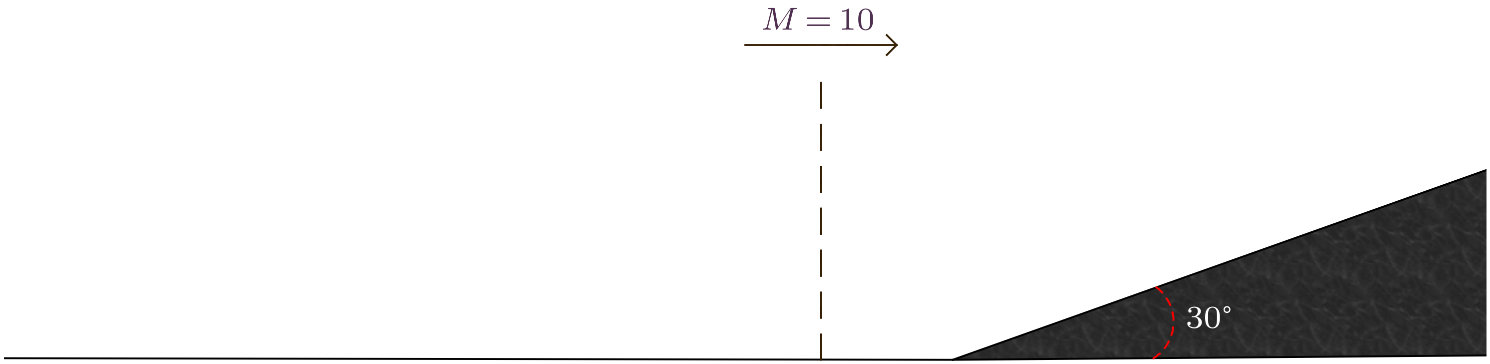
Initial State



$\mathrm{TVB} - 100$

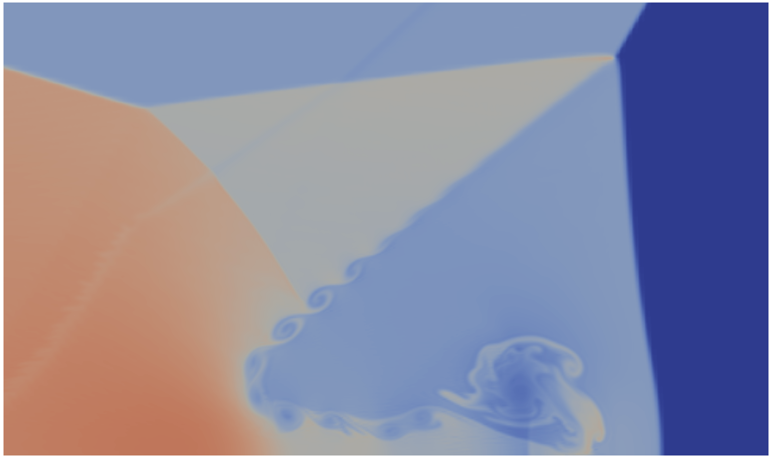


MUSCL-Hancock

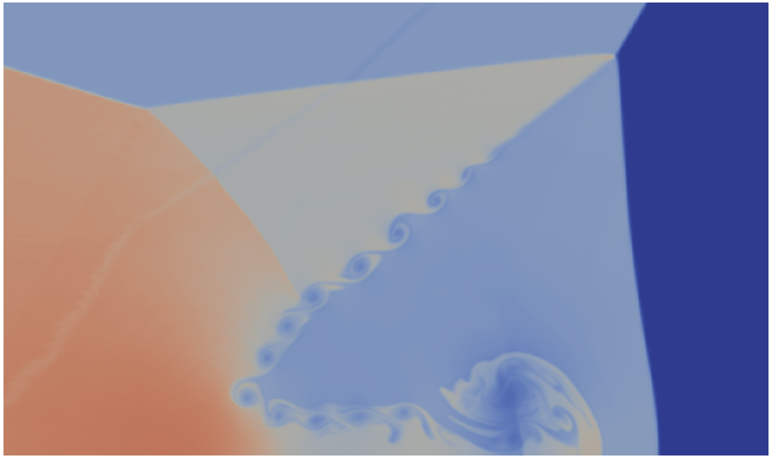


Animation link

$t = 0.2$, NC $= 568 \times 142$, Rusanov, $N = 4$

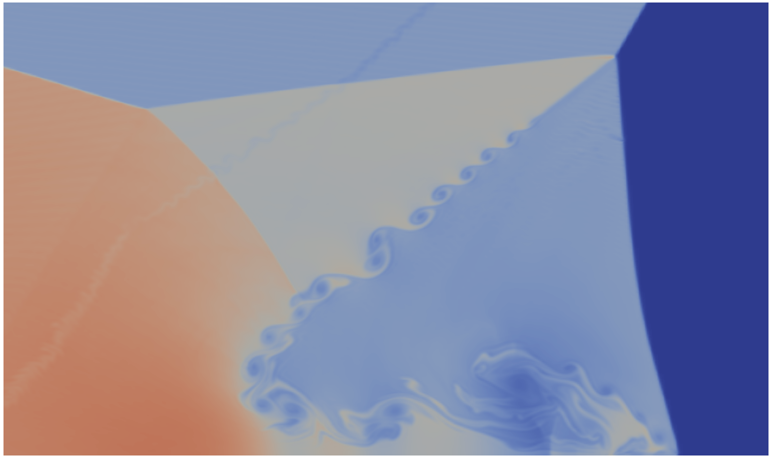


`Trixi.jl`

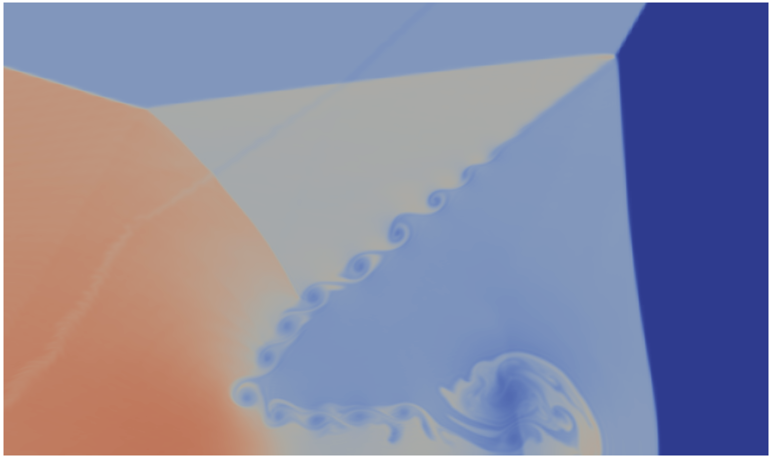


LWFR

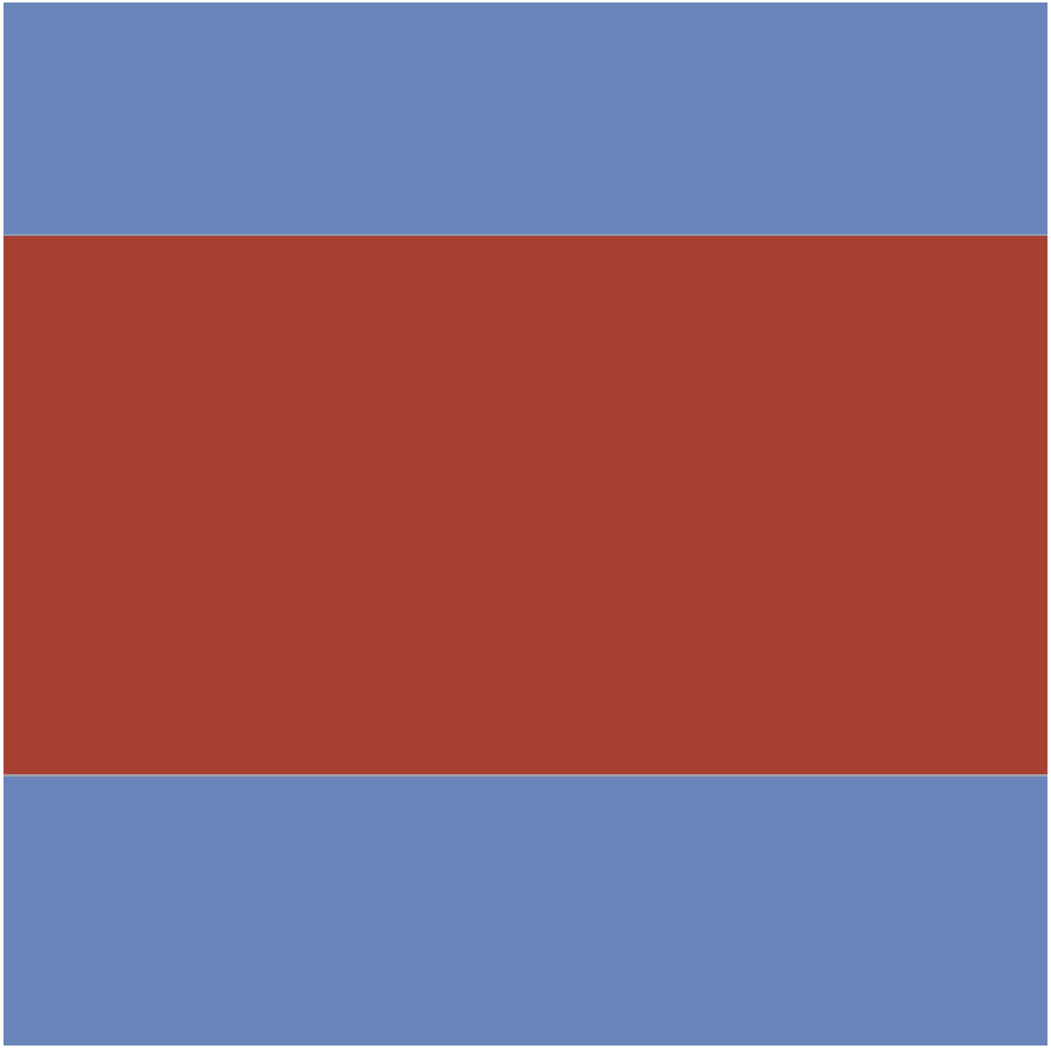
$t = 0.2$, NC$=568 \times 142$, LWFR, $N = 4$



HLLC, $\beta_1 = 0.1$, $\beta_2 = 1.0$



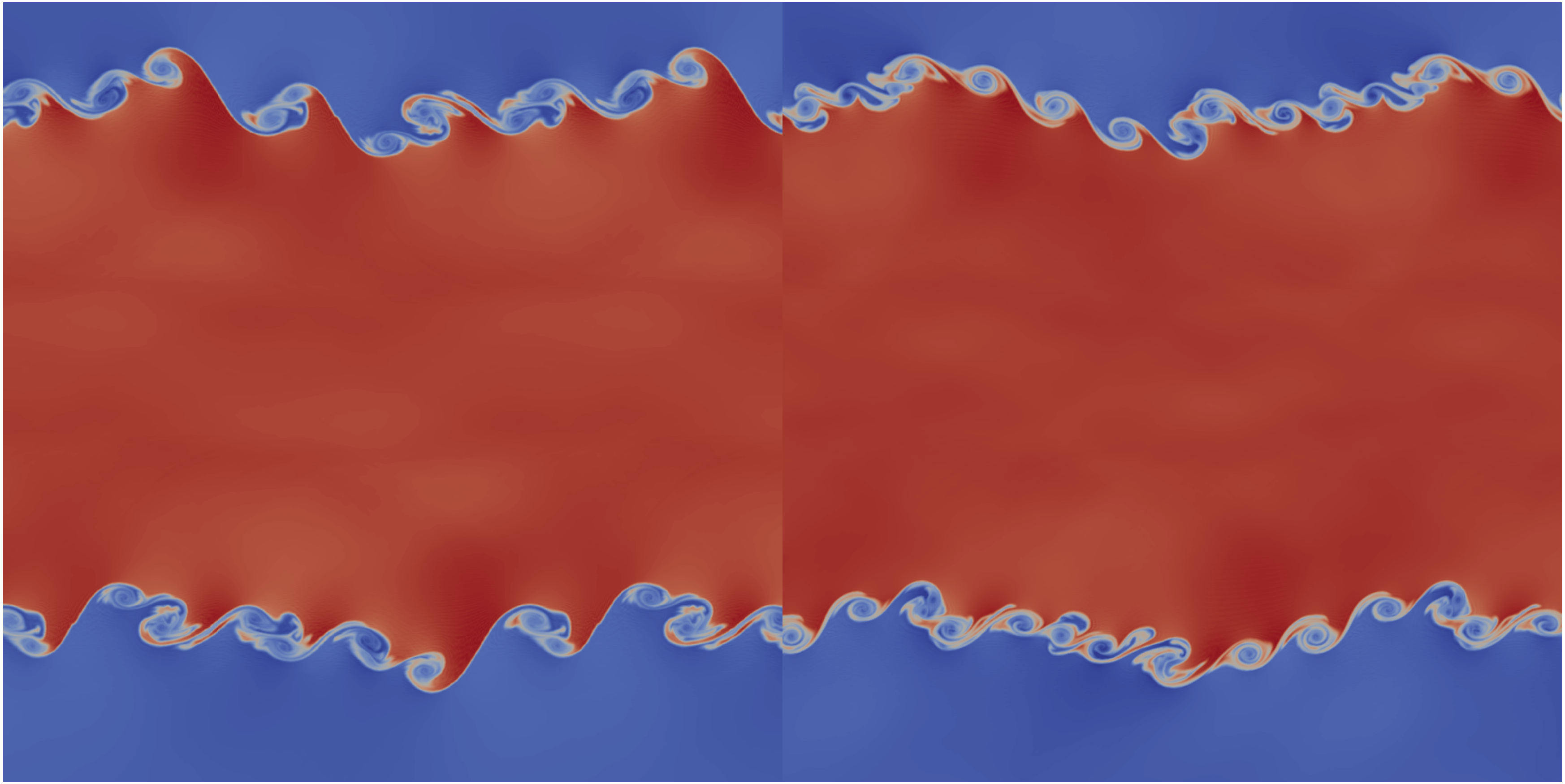
Rusanov. $\beta_1 = \beta_2 = 1.0$



Animation link

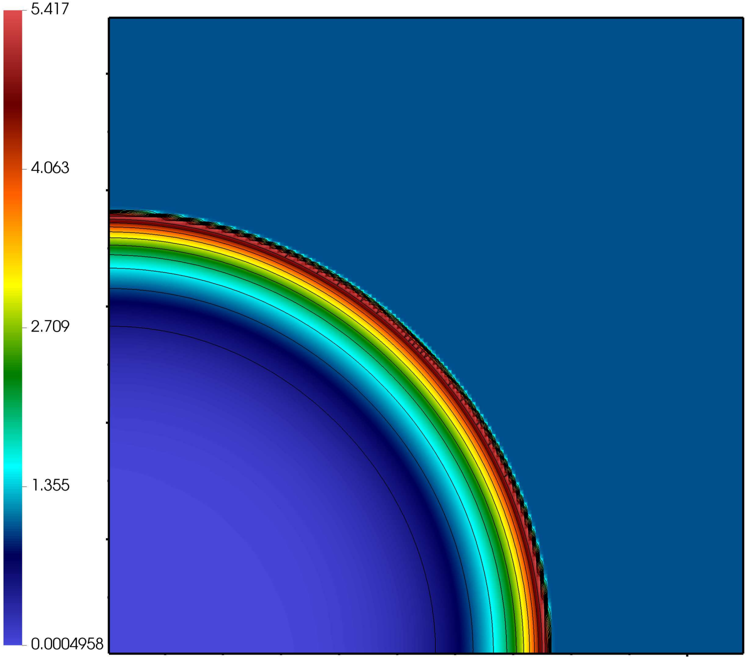
Kelvin-Helmholtz Instability [18, 15]

Density profile $t = 0.4,\ \mathrm{NC} = 256^2,\ N = 4$

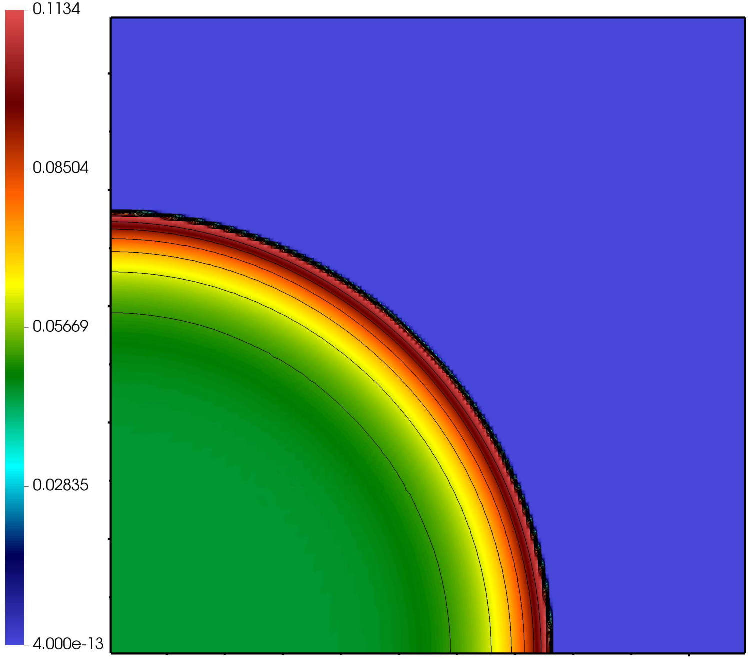


`Trixi.jl` LWFR

$$t = 0.001, \quad \mathrm{NC} = 160^2, \quad N = 4$$



Density



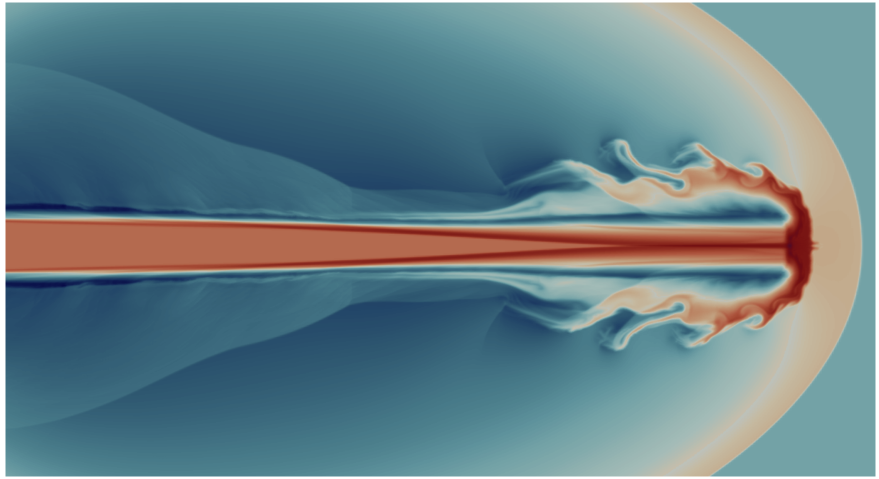
Pressure

The initial condition is given by

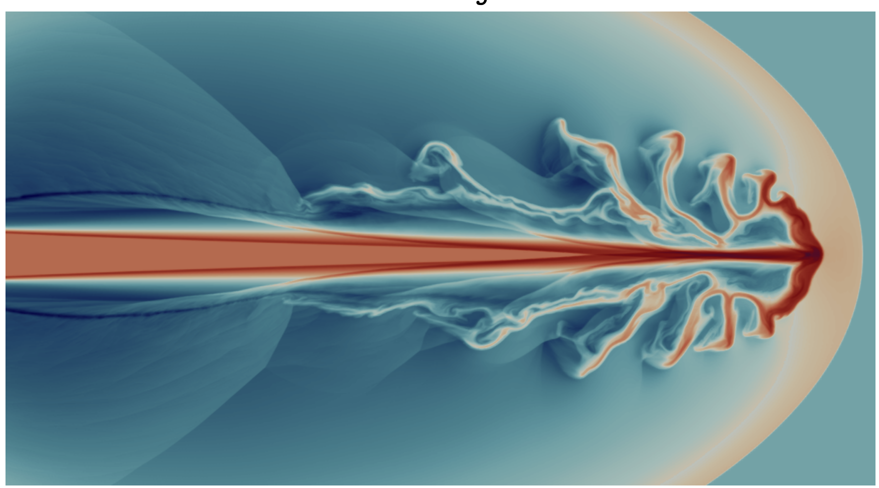
$$\rho = 1.0, \quad v_1 = v_2 = 0.0, \quad E(x, y) = \begin{cases} \frac{0.244816}{\Delta x \, \Delta y} & x < \Delta x, \, y < \Delta y, \\ 10^{-12} & \text{otherwise.} \end{cases}$$

Animation link

Density profile, $\mathrm{NC} = 448 \times 224$, $t = 0.07$, $N = 4$

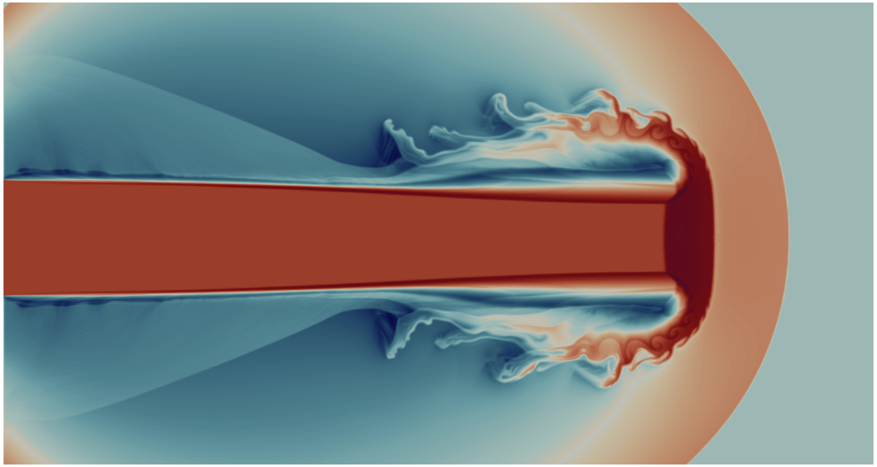


`Trixi.jl`

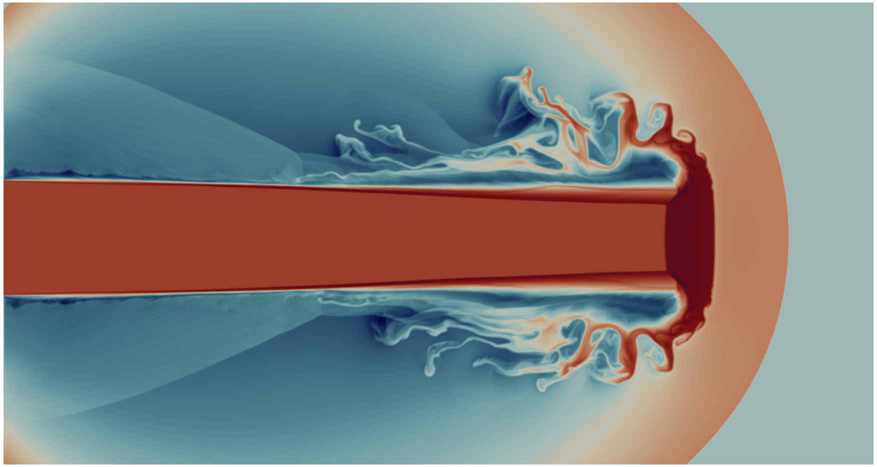


LWFR

Density profile, $\mathrm{NC} = 640 \times 320$, $t = 0.001$, $N = 4$



Trixi.jl



LWFR

- **Modularity** - new conservation law can be added without modifying base code. User need only supply physical flux, numerical flux and wave speed estimates.
- **Portability** - Dependencies are handled by Julia's package manager
- **Parallelization** - Shared-memory via multithreading
- **Efficiency** - noticeably faster than some C++ implementations
- **Visualization** - Postprocessing to `vtr` format

- **Modularity** - new conservation law can be added without modifying base code. User need only supply physical flux, numerical flux and wave speed estimates.
- **Portability** - Dependencies are handled by Julia's package manager
- **Parallelization** - Shared-memory via multithreading
- **Efficiency** - noticeably faster than some C++ implementations
- **Visualization** - Postprocessing to `vtr` format

- **Modularity** - new conservation law can be added without modifying base code. User need only supply physical flux, numerical flux and wave speed estimates.
- **Portability** - Dependencies are handled by Julia's package manager
- **Parallelization** - Shared-memory via multithreading
- **Efficiency** - noticeably faster than some C++ implementations
- **Visualization** - Postprocessing to `vtr` format

- **Modularity** - new conservation law can be added without modifying base code. User need only supply physical flux, numerical flux and wave speed estimates.
- **Portability** - Dependencies are handled by Julia's package manager
- **Parallelization** - Shared-memory via multithreading
- **Efficiency** - noticeably faster than some C++ implementations
- **Visualization** - Postprocessing to `vtr` format

- **Modularity** - new conservation law can be added without modifying base code. User need only supply physical flux, numerical flux and wave speed estimates.
- **Portability** - Dependencies are handled by Julia's package manager
- **Parallelization** - Shared-memory via multithreading
- **Efficiency** - noticeably faster than some C++ implementations
- **Visualization** - Postprocessing to `vtr` format

```
container = Dict( "u" => u, ... )
...
u = container["u"]
for cell in eachelement(grid)
    ! heavy computation with u
end
```

Bad version

```
container = (; u, ...)
...
u = container.u
for cell in eachelement(grid)
    ! heavy computation with u
end
```

Good version

```
container = Dict( "u" => u, ... )
...
u = container["u"]
for cell in eachelement(grid)
    ! heavy computation with u
end
```

Bad version

```
container = (; u, ...)
...
u = container.u
for cell in eachelement(grid)
    ! heavy computation with u
end
```

Good version

Tool to find type instabilities - `ProfileView.jl`

```
container = Dict( "u" => u, ... )
...
u = container["u"]
for cell in eachelement(grid)
    ! heavy computation with u
end
```

Bad version

```
container = (; u, ...)
...
u = container.u
for cell in eachelement(grid)
    ! heavy computation with u
end
```

Good version

Tool to find type instabilities - `ProfileView.jl`

Tool to measure allocations - `BenchmarkTools.jl`, `TimerOutputs.jl`

```
container = Dict( "u" => u, ... )
...
u = container["u"]
for cell in eachelement(grid)
    ! heavy computation with u
end
```

Bad version

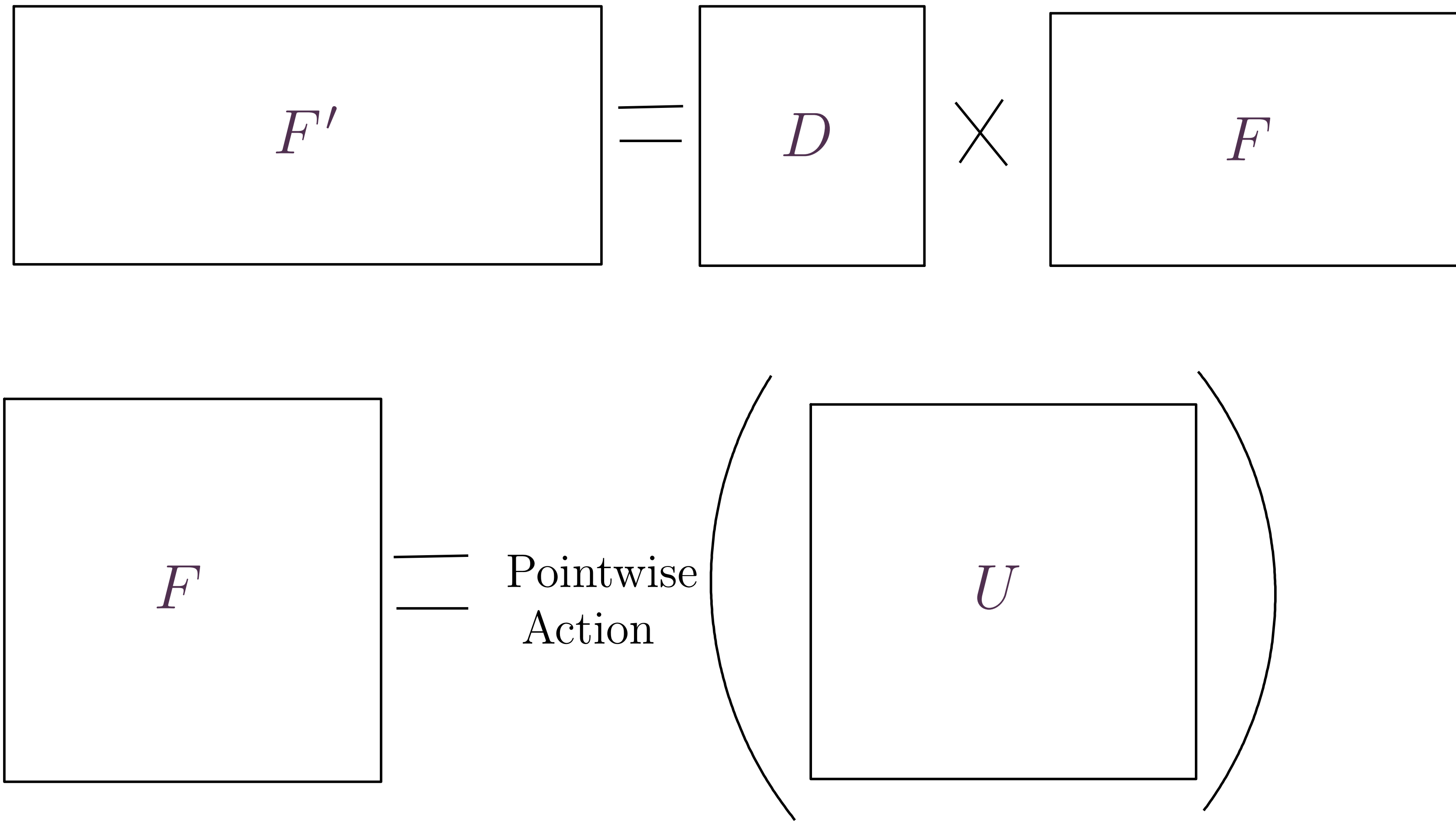
```
container = (; u, ...)
...
u = container.u
for cell in eachelement(grid)
    ! heavy computation with u
end
```

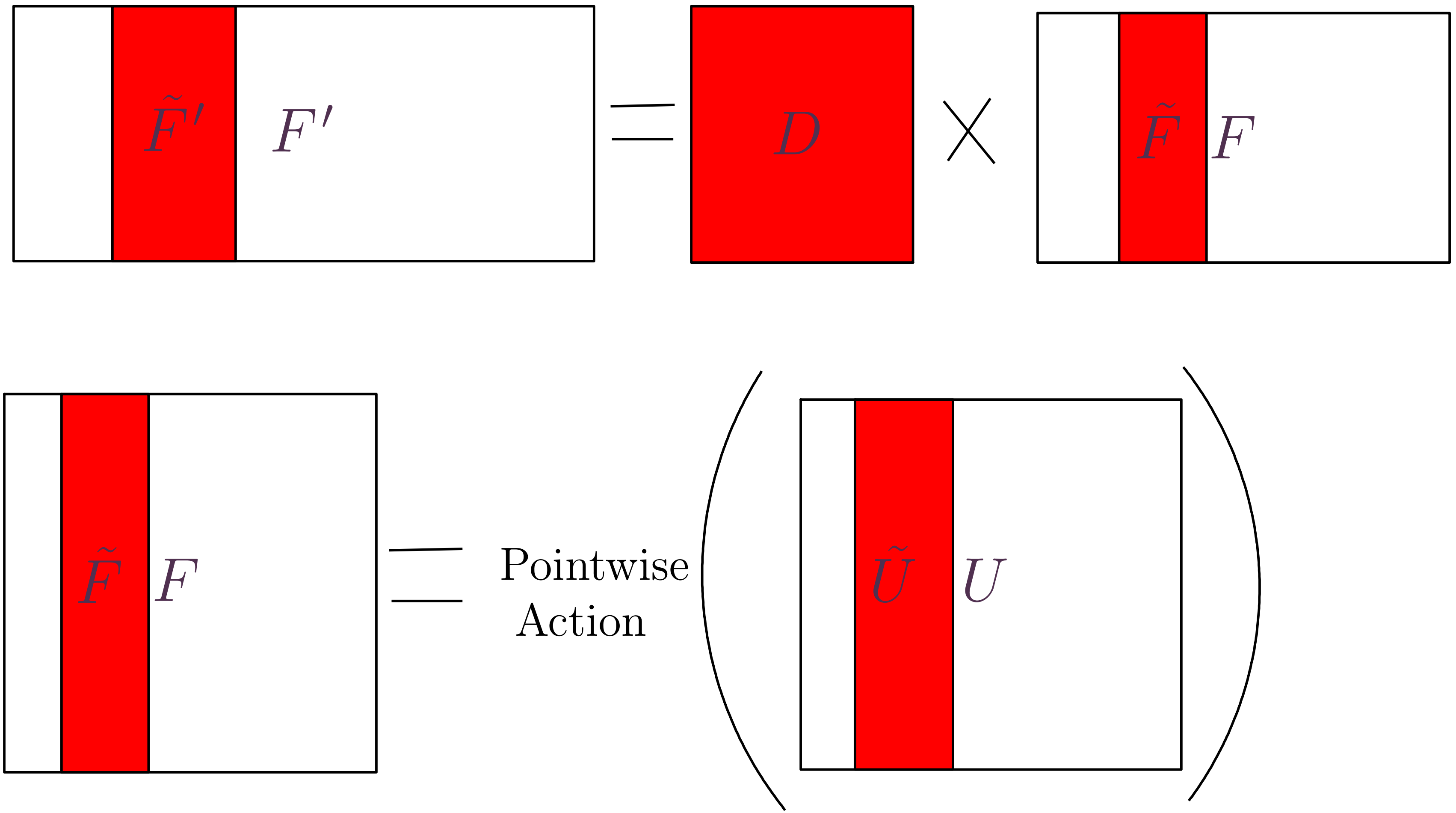
Good version

Tool to find type instabilities - `ProfileView.jl`

Tool to measure allocations - `BenchmarkTools.jl`, `TimerOutputs.jl`

Fixing the problem - JuliaLang - forum, Zulip, Slack.


F'
$=$
D
$\times$
F
F
$=$
Pointwise Action
U


$\tilde{F}'$
F'
$=$
D
$\times$
$\tilde{F}$
F
$\tilde{F}$
F
$=$
Pointwise Action
$\tilde{U}$
U

```
for cell in eachelement(grid) ! Cell loop
  for i in eachnode(basis)    ! DoF loop
    f[:,i,cell] = flux(u[:,i,cell])
  end
  BLAS.mul(D, f, res)
end
```

Bad version

```
for cell in eachelement(grid) ! Cell loop
  for i in eachnode(basis)    ! DoF loop
    u_node = get_node_vars(eq, u, i, cell)
    f_node = flux(u_node)
    for ix in eachnode(basis)
      ! res[:,ix,i,cell] += D[ix,i] * f_node
      multiply_add_to_node_vars(eq, D[ix,i],
                                f_node, res,
                                iix, cell)
    end
  end
end
```

Good version

- Sub-cell based blending limiter based on [8] with MUSCL-Hancock reconstruction constructed for Lax-Wendroff schemes
- MUSCL-Hancock blending more accurate than first order blending
- MUSCL-Hancock for general grids proved to be admissibility preserving
- Admissibility preserving Lax-Wendroff scheme constructed
- Efficient 2-D implementation in `Julia`
- LW-D2 and ADER equivalent for linear, and *close* for non-linear equations
- Mild instability for ADER and LW schemes for $N = 4$

Future Plans

- Extend the scheme to unstructured grids.

- Sub-cell based blending limiter based on [8] with MUSCL-Hancock reconstruction constructed for Lax-Wendroff schemes
- MUSCL-Hancock blending more accurate than first order blending
- MUSCL-Hancock for general grids proved to be admissibility preserving
- Admissibility preserving Lax-Wendroff scheme constructed
- Efficient 2-D implementation in `Julia`
- LW-D2 and ADER equivalent for linear, and *close* for non-linear equations
- Mild instability for ADER and LW schemes for $N = 4$

Future Plans

- Extend the scheme to unstructured grids.

- Sub-cell based blending limiter based on [8] with MUSCL-Hancock reconstruction constructed for Lax-Wendroff schemes
- MUSCL-Hancock blending more accurate than first order blending
- MUSCL-Hancock for general grids proved to be admissibility preserving
- Admissibility preserving Lax-Wendroff scheme constructed
- Efficient 2-D implementation in `Julia`
- LW-D2 and ADER equivalent for linear, and *close* for non-linear equations
- Mild instability for ADER and LW schemes for $N = 4$

Future Plans

- Extend the scheme to unstructured grids.

- Sub-cell based blending limiter based on [8] with MUSCL-Hancock reconstruction constructed for Lax-Wendroff schemes
- MUSCL-Hancock blending more accurate than first order blending
- MUSCL-Hancock for general grids proved to be admissibility preserving
- Admissibility preserving Lax-Wendroff scheme constructed
- Efficient 2-D implementation in `Julia`
- LW-D2 and ADER equivalent for linear, and *close* for non-linear equations
- Mild instability for ADER and LW schemes for $N = 4$

Future Plans

- Extend the scheme to unstructured grids.

- Sub-cell based blending limiter based on [8] with MUSCL-Hancock reconstruction constructed for Lax-Wendroff schemes
- MUSCL-Hancock blending more accurate than first order blending
- MUSCL-Hancock for general grids proved to be admissibility preserving
- Admissibility preserving Lax-Wendroff scheme constructed
- Efficient 2-D implementation in `Julia`
- LW-D2 and ADER equivalent for linear, and *close* for non-linear equations
- Mild instability for ADER and LW schemes for $N = 4$

Future Plans

- Extend the scheme to unstructured grids.

- Sub-cell based blending limiter based on [8] with MUSCL-Hancock reconstruction constructed for Lax-Wendroff schemes
- MUSCL-Hancock blending more accurate than first order blending
- MUSCL-Hancock for general grids proved to be admissibility preserving
- Admissibility preserving Lax-Wendroff scheme constructed
- Efficient 2-D implementation in `Julia`
- LW-D2 and ADER equivalent for linear, and *close* for non-linear equations
- Mild instability for ADER and LW schemes for $N = 4$

Future Plans

- Extend the scheme to unstructured grids.

- Sub-cell based blending limiter based on [8] with MUSCL-Hancock reconstruction constructed for Lax-Wendroff schemes
- MUSCL-Hancock blending more accurate than first order blending
- MUSCL-Hancock for general grids proved to be admissibility preserving
- Admissibility preserving Lax-Wendroff scheme constructed
- Efficient 2-D implementation in `Julia`
- LW-D2 and ADER equivalent for linear, and *close* for non-linear equations
- Mild instability for ADER and LW schemes for $N = 4$

Future Plans

- Extend the scheme to unstructured grids.

- Sub-cell based blending limiter based on [8] with MUSCL-Hancock reconstruction constructed for Lax-Wendroff schemes
- MUSCL-Hancock blending more accurate than first order blending
- MUSCL-Hancock for general grids proved to be admissibility preserving
- Admissibility preserving Lax-Wendroff scheme constructed
- Efficient 2-D implementation in `Julia`
- LW-D2 and ADER equivalent for linear, and *close* for non-linear equations
- Mild instability for ADER and LW schemes for $N = 4$

Future Plans

- Extend the scheme to unstructured grids.

- Sub-cell based blending limiter based on [8] with MUSCL-Hancock reconstruction constructed for Lax-Wendroff schemes
- MUSCL-Hancock blending more accurate than first order blending
- MUSCL-Hancock for general grids proved to be admissibility preserving
- Admissibility preserving Lax-Wendroff scheme constructed
- Efficient 2-D implementation in `Julia`
- LW-D2 and ADER equivalent for linear, and *close* for non-linear equations
- Mild instability for ADER and LW schemes for $N = 4$

Future Plans

- Extend the scheme to unstructured grids.

- *Continuum mechanics* by *Gonzales* and *Stuart*
- *A mathematical introduction to fluid mechanics* by *Chorin*
- *Gas Dynamics* at NPTEL-IISc
- *Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction* by *E.F. Toro*
- *Introduction to Numerical Methods for Time Dependent Differential Equations* by *Kreiss, Ortiz*

- *Continuum mechanics* by *Gonzales* and *Stuart*
- *A mathematical introduction to fluid mechanics* by *Chorin*
- *Gas Dynamics* at NPTEL-IISc
- *Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction* by *E.F. Toro*
- *Introduction to Numerical Methods for Time Dependent Differential Equations* by *Kreiss, Ortiz*

- *Continuum mechanics* by *Gonzales* and *Stuart*
- *A mathematical introduction to fluid mechanics* by *Chorin*
- *Gas Dynamics* at NPTEL-IISc
- *Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction* by *E.F. Toro*
- *Introduction to Numerical Methods for Time Dependent Differential Equations* by *Kreiss, Ortiz*

- *Continuum mechanics* by *Gonzales* and *Stuart*
- *A mathematical introduction to fluid mechanics* by *Chorin*
- *Gas Dynamics* at NPTEL-IISc
- *Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction* by *E.F. Toro*
- *Introduction to Numerical Methods for Time Dependent Differential Equations* by *Kreiss, Ortiz*

- *Continuum mechanics* by *Gonzales* and *Stuart*
- *A mathematical introduction to fluid mechanics* by *Chorin*
- *Gas Dynamics* at NPTEL-IISc
- *Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction* by *E.F. Toro*
- *Introduction to Numerical Methods for Time Dependent Differential Equations* by *Kreiss, Ortiz*

Publication

Arpit Babbar, Sudarshan Kumar Kenettinkara, and Praveen Chandrashekar. Lax-wendroff flux reconstruction method for hyperbolic conservation laws. *Journal of Computational Physics*, 2022 https://doi.org/10.1016/j.jcp.2022.111423, https://arxiv.org/abs/2207.02954

Bibliography

[1] Semih Akkurt, Freddie Witherden, and Peter Vincent. Cache blocking strategies applied to flux reconstruction. *Computer Physics Communications*, 271:108193, 2022.

[2] Arpit Babbar, Sudarshan Kumar Kenettinkara, and Praveen Chandrashekar. Lax-wendroff flux reconstruction method for hyperbolic conservation laws. *Journal of Computational Physics*, page 111423, 2022.

[3] Christophe Berthon. Why the MUSCL–Hancock scheme is l1-stable. *Numerische Mathematik*, 104(1):27–46, jun 2006.

[4] Jeff Bezanson, Alan Edelman, Stefan Karpinski, and Viral B Shah. Julia: a fresh approach to numerical computing. *SIAM Review*, 59(1):65–98, 2017.

[5] Michael Dumbser and Olindo Zanotti. Very high order PNPM schemes on unstructured meshes for the resistive relativistic MHD equations. *Journal of Computational Physics*, 228(18):6991–7006, oct 2009.

[6] Camille Felton, Mariana Harris, Caleb Logemann, Stefan Nelson, Ian Pelakh, and James A Rossmanith. A positivity-preserving limiting strategy for locally-implicit Lax-Wendroff discontinuous galerkin methods. Jun 2018.

[7] Youngsoo Ha, Carl Gardner, Anne Gelb, and ⟨keepcase|Chi Wang⟩ Shu. Numerical simulation of high mach number astrophysical jets with radiative cooling. *Journal of Scientific Computing*, 24(1):597–612, jul 2005. Funding Information: We would like to thank Jeff Hester for valuable discussions. The authors would like to acknowledge for the following: (1) Research supported in part by the BK21 project at KAIST (Youngsoo Ha); (2) Research supported in part by the Space Telescope Science Institute under grant HST-GO-09863.06-A (Carl L. Gardner); (3) Research supported in part by the National Science Foundation under grant DMS-0107428. (Anne Gelb); (4) Research supported in part by the Army Research Office under grant DAAD19-00-1-0405 and in part by the National Science Foundation under grant DMS-0207451 (Chi-Wang Shu).

[8] Sebastian Hennemann, Andrés M. Rueda-Ramírez, Florian J. Hindenlang, and Gregor J. Gassner. A provably entropy stable subcell shock capturing approach for high order split form dg for the compressible euler equations. *Journal of Computational Physics*, 426:109935, 2021.

[9] H. T. Huynh. A Flux Reconstruction Approach to High-Order Schemes Including Discontinuous Galerkin Methods. Miami, FL, jun 2007. AIAA.

[10] Haran Jackson. On the eigenvalues of the ADER-WENO galerkin predictor. *Journal of Computational Physics*, 333:409–413, mar 2017.

[11] Randall J. LeVeque. High-Resolution Conservative Algorithms for Advection in Incompressible Flow. *SIAM Journal on Numerical Analysis*, 33(2):627–665, apr 1996. Bibtex: LeVeque1996.

[12] Per-Olof Persson and Jaime Peraire. Sub-Cell Shock Capturing for Discontinuous Galerkin Methods. In *44th AIAA Aerospace Sciences Meeting and Exhibit*, Aerospace Sciences Meetings. American Institute of Aeronautics and Astronautics, jan 2006.

[13] Hendrik Ranocha, Michael Schlottke-Lakemper, Andrew Ross Winters, Erik Faulhaber, Jesse Chan, and Gregor Gassner. Adaptive numerical simulations with Trixi.jl: A case study of Julia for scientific computing. *Proceedings of the JuliaCon Conferences*, 1(1):77, 2022.

[14] A Rueda-Ramrez and G Gassner. A subcell finite volume positivity-preserving limiter for DGSEM discretizations of the euler equations. In *14th WCCM-ECCOMAS Congress*. CIMNE, 2021.

[15] Kevin Schaal, Andreas Bauer, Praveen Chandrashekar, Rüdiger Pakmor, Christian Klingenberg, and Volker Springel. Astrophysical hydrodynamics with a high-order discontinuous Galerkin scheme and adaptive mesh refinement. *Monthly Notices of the Royal Astronomical Society*, 453(4):4278–4300, 09 2015.

[16] L.I. SEDOV. Chapter iv - one-dimensional unsteady motion of a gas. In L.I. SEDOV, editor, *Similarity and Dimensional Methods in Mechanics*, pages 146–304. Academic Press, 1959.

[17] Chi-Wang Shu and Stanley Osher. Efficient implementation of essentially non-oscillatory shock-capturing schemes, II. *Journal of Computational Physics*, 83(1):32–78, jul 1989.

[18] Volker Springel. E pur si muove: Galilean-invariant cosmological hydrodynamical simulations on a moving mesh. *Monthly Notices of the Royal Astronomical Society*, 401(2):791–851, 01 2010.

[19] Paul Woodward and Phillip Colella. The numerical simulation of two-dimensional fluid flow with strong shocks. *Journal of Computational Physics*, 54(1):115–173, apr 1984.

[20] Yuan Xu, Qiang Zhang, Chi-wang Shu, and Haijin Wang. The L^2-norm Stability Analysis of Runge–Kutta Discontinuous Galerkin Methods for Linear Hyperbolic Equations. *SIAM Journal on Numerical Analysis*, 57(4):1574–1601, jan 2019. Bibtex: Xu2019.

[21] Xiangxiong Zhang and Chi-Wang Shu. On maximum-principle-satisfying high order schemes for scalar conservation laws. *Journal of Computational Physics*, 229(9):3091–3120, may 2010.

[22] D. Zorío, A. Baeza, and P. Mulet. An Approximate Lax–Wendroff-Type Procedure for High Order Accurate Schemes for Hyperbolic Conservation Laws. *Journal of Scientific Computing*, 71(1):246–273, apr 2017. Bibtex: Zorio2017.

Thank you

Joint Work With

Praveen Chandrashekar, TIFR-CAM, Bangalore

Sudarshan Kumar Kenettinkara, IISER-Trivandrum

$$(u_j^e)^{n+1} = (u_j^e)^n - \frac{\Delta t}{\Delta x_e} \frac{\partial F_h}{\partial \xi}(\xi_j), \qquad 1 \leq j \leq N+1.$$

$$(u_j^e)^{n+1} = (u_j^e)^n - \frac{\Delta t}{\Delta x_e} \frac{\partial F_h}{\partial \xi}(\xi_j), \qquad 1 \le j \le N+1.$$

For $\{w_j\}_{j=1}^{N+1}$ being the quadrature weights associated with solution points:

$$\sum_{j=1}^{N+1} w_j (u_j^e)^{n+1} = \sum_{j=1}^{N+1} (u_j^e)^n - \frac{\Delta t}{\Delta x_e} \sum_{j=1}^{N+1} w_j \frac{\partial F_h}{\partial \xi}(\xi_j),$$

$$(u_j^e)^{n+1} = (u_j^e)^n - \frac{\Delta t}{\Delta x_e} \frac{\partial F_h}{\partial \xi}(\xi_j), \qquad 1 \le j \le N+1.$$

For $\{w_j\}_{j=1}^{N+1}$ being the quadrature weights associated with solution points:

$$\sum_{j=1}^{N+1} w_j (u_j^e)^{n+1} = \sum_{j=1}^{N+1} (u_j^e)^n - \frac{\Delta t}{\Delta x_e} \sum_{j=1}^{N+1} w_j \frac{\partial F_h}{\partial \xi}(\xi_j),$$

$$\Rightarrow \overline{u}_e^{n+1} = \overline{u}_e^n - \frac{\Delta t}{\Delta x_e}\Big[F_{e+\frac{1}{2}} - F_{e-\frac{1}{2}} \Big].$$

High order

$$\overline{u}_e^{H,n+1} = \overline{u}_e^n - \frac{\Delta t}{\Delta x_e}\left(F_{e+\frac{1}{2}}^H - F_{e-\frac{1}{2}}^H\right)$$

Low-order

$$\overline{u}_e^{L,n+1} = \overline{u}_e^n - \frac{\Delta t}{\Delta x_e}\left(F_{e+\frac{1}{2}}^L - F_{e-\frac{1}{2}}^L\right).$$

For blended update

$$\overline{u}_e^{n+1} = \overline{u}_e^n - \Delta t\left(F_{e+\frac{1}{2}} - F_{e-\frac{1}{2}}\right),$$

$$\text{where} \quad F_{e+\frac{1}{2}} = \alpha_e F_{e+\frac{1}{2}}^L + (1-\alpha_e) F_{e+\frac{1}{2}}^H.$$

Conservation requires

$$\alpha_e F_{e+\frac{1}{2}}^L + (1-\alpha_e) F_{e+\frac{1}{2}}^H = \alpha_{e+1} F_{e+\frac{1}{2}}^L + (1-\alpha_{e+1}) F_{e+\frac{1}{2}}^H$$

$$\Rightarrow F_{e+\frac{1}{2}}^L = F_{e+\frac{1}{2}}^H$$

Initial candidate:
$$
\begin{aligned}
\tilde{F}_{e_x+\frac{1}{2},e_y,j} &= (1-\alpha_{e_x+\frac{1}{2},e_y})\,F^{\mathrm{LW}}_{e_x+\frac{1}{2},e_y,j} + \alpha_{e_x+\frac{1}{2},e_y}\, f_{\boldsymbol{e},N+\frac{3}{2},j}, \qquad & 1 \le j \le N+1,\\
\tilde{F}_{e_x,e_y+\frac{1}{2},i} &= (1-\alpha_{e_x,e_y+\frac{1}{2}})\,F^{\mathrm{LW}}_{e_x,e_y+\frac{1}{2},i} + \alpha_{e_x,e_y+\frac{1}{2}}\, f_{\boldsymbol{e},i,N+\frac{3}{2}}, \qquad & 1 \le i \le N+1.
\end{aligned}
$$

Initial candidate:
$$
\begin{aligned}
\tilde{F}_{e_x+\frac{1}{2},e_y,j} &= (1-\alpha_{e_x+\frac{1}{2},e_y})\, F^{\text{LW}}_{e_x+\frac{1}{2},e_y,j} + \alpha_{e_x+\frac{1}{2},e_y}\, f_{\boldsymbol{e},N+\frac{3}{2},j}, \qquad && 1 \le j \le N+1,\\
\tilde{F}_{e_x,e_y+\frac{1}{2},i} &= (1-\alpha_{e_x,e_y+\frac{1}{2}})\, F^{\text{LW}}_{e_x,e_y+\frac{1}{2},i} + \alpha_{e_x,e_y+\frac{1}{2}}\, f_{\boldsymbol{e},i,N+\frac{3}{2}}, \qquad && 1 \le i \le N+1.
\end{aligned}
$$

$$
\begin{aligned}
\tilde{\boldsymbol{u}}^{n+1} &= \boldsymbol{u}^n_{\boldsymbol{e},1,j} - \frac{\Delta t}{\Delta x_{\boldsymbol{e}} w_1}\Big(\tilde{F}_{e_x+\frac{1}{2},e_y,j} - f_{\boldsymbol{e},\frac{1}{2},j}\Big) - \frac{\Delta t}{\Delta y_e w_j}\Big(f_{\boldsymbol{e},1,j+\frac{1}{2}} - f_{\boldsymbol{e},1,j-\frac{1}{2}}\Big), \quad && 1 < j < N+1,\\
\tilde{\boldsymbol{u}}^{n+1} &= \boldsymbol{u}^n_{\boldsymbol{e},1,1} - \frac{\Delta t}{\Delta x_{\boldsymbol{e}} w_1}\Big(\tilde{F}_{e_x+\frac{1}{2},e_y,1} - f_{\boldsymbol{e},\frac{1}{2},1}\Big) - \frac{\Delta t}{\Delta y_e w_1}\Big(\tilde{F}_{e_x,e_y+\frac{1}{2},1} - f_{\boldsymbol{e},1,\frac{1}{2}}\Big), \quad && j = 1.
\end{aligned}
$$

Initial candidate:
$$\begin{aligned} \tilde{F}_{e_x+\frac{1}{2},e_y,j} &= (1-\alpha_{e_x+\frac{1}{2},e_y})\,F^{\text{LW}}_{e_x+\frac{1}{2},e_y,j} + \alpha_{e_x+\frac{1}{2},e_y}\,f_{\boldsymbol{e},N+\frac{3}{2},j}, && 1 \le j \le N+1,\\ \tilde{F}_{e_x,e_y+\frac{1}{2},i} &= (1-\alpha_{e_x,e_y+\frac{1}{2}})\,F^{\text{LW}}_{e_x,e_y+\frac{1}{2},i} + \alpha_{e_x,e_y+\frac{1}{2}}\,f_{\boldsymbol{e},i,N+\frac{3}{2}}, && 1 \le i \le N+1. \end{aligned}$$

$$\begin{aligned} \tilde{\boldsymbol{u}}^{n+1} &= \boldsymbol{u}^n_{\boldsymbol{e},1,j} - \frac{\Delta t}{\Delta x_{\boldsymbol{e}} w_1}\Big(\tilde{F}_{e_x+\frac{1}{2},e_y,j} - f_{\boldsymbol{e},\frac{1}{2},j}\Big) - \frac{\Delta t}{\Delta y_e w_j}\Big(f_{\boldsymbol{e},1,j+\frac{1}{2}} - f_{\boldsymbol{e},1,j-\frac{1}{2}}\Big), \quad 1 < j < N+1,\\ \tilde{\boldsymbol{u}}^{n+1} &= \boldsymbol{u}^n_{\boldsymbol{e},1,1} - \frac{\Delta t}{\Delta x_{\boldsymbol{e}} w_1}\Big(\tilde{F}_{e_x+\frac{1}{2},e_y,1} - f_{\boldsymbol{e},\frac{1}{2},1}\Big) - \frac{\Delta t}{\Delta y_e w_1}\Big(\tilde{F}_{e_x,e_y+\frac{1}{2},1} - f_{\boldsymbol{e},1,\frac{1}{2}}\Big), \quad j = 1. \end{aligned}$$

In the 2-D code, there's two separate face loops for vertical and horizontal faces. This poses a challenge because to ensure $\tilde{u}^{n+1}$ is admissible, we need to correct both $\tilde{F}_{e_x+\frac{1}{2},e_y,1}$ and $\tilde{F}_{e_x,e_y+\frac{1}{2},1}$ and these values are never available together.

Initial candidate:
$$
\begin{aligned}
\tilde{F}_{e_x+\frac{1}{2},e_y,j} &= (1-\alpha_{e_x+\frac{1}{2},e_y})\, F^{\text{LW}}_{e_x+\frac{1}{2},e_y,j} + \alpha_{e_x+\frac{1}{2},e_y}\, f_{\boldsymbol{e},N+\frac{3}{2},j}, \qquad 1 \le j \le N+1,\\
\tilde{F}_{e_x,e_y+\frac{1}{2},i} &= (1-\alpha_{e_x,e_y+\frac{1}{2}})\, F^{\text{LW}}_{e_x,e_y+\frac{1}{2},i} + \alpha_{e_x,e_y+\frac{1}{2}}\, f_{\boldsymbol{e},i,N+\frac{3}{2}}, \qquad 1 \le i \le N+1.
\end{aligned}
$$

$$
\begin{aligned}
\tilde{\boldsymbol{u}}^{n+1} &= \boldsymbol{u}^n_{\boldsymbol{e},1,j} - \frac{\Delta t}{\Delta x_{\boldsymbol{e}} w_1}\Big(\tilde{F}_{e_x+\frac{1}{2},e_y,j} - f_{\boldsymbol{e},\frac{1}{2},j}\Big) - \frac{\Delta t}{\Delta y_e w_j}\Big(f_{\boldsymbol{e},1,j+\frac{1}{2}} - f_{\boldsymbol{e},1,j-\frac{1}{2}}\Big), \quad 1<j<N+1,\\
\tilde{\boldsymbol{u}}^{n+1} &= \boldsymbol{u}^n_{\boldsymbol{e},1,1} - \frac{\Delta t}{\Delta x_{\boldsymbol{e}} w_1}\Big(\tilde{F}_{e_x+\frac{1}{2},e_y,1} - f_{\boldsymbol{e},\frac{1}{2},1}\Big) - \frac{\Delta t}{\Delta y_e w_1}\Big(\tilde{F}_{e_x,e_y+\frac{1}{2},1} - f_{\boldsymbol{e},1,\frac{1}{2}}\Big), \quad j=1.
\end{aligned}
$$

In the 2-D code, there's two separate face loops for vertical and horizontal faces. This poses a challenge because to ensure $\tilde{u}^{n+1}$ is admissible, we need to correct both $\tilde{F}_{e_x+\frac{1}{2},e_y,1}$ and $\tilde{F}_{e_x,e_y+\frac{1}{2},1}$ and these values are never available together.

To avoid having to store values and doing aposteriori correction, we find appropriate λ_x, λ_y such that

$$\lambda_x + \lambda_y = 1,$$

and then, following the 1-D procedure, construct corrected $F_{e_x+\frac{1}{2},e_y,1}$ and $F_{e_x,e_y+\frac{1}{2},1}$ such that

Initial candidate:

$$
\begin{aligned}
\tilde{F}_{e_x+\frac{1}{2},e_y,j} &= (1-\alpha_{e_x+\frac{1}{2},e_y})\, F^{\text{LW}}_{e_x+\frac{1}{2},e_y,j} + \alpha_{e_x+\frac{1}{2},e_y}\, f_{\boldsymbol{e},N+\frac{3}{2},j}, && 1 \le j \le N+1,\\
\tilde{F}_{e_x,e_y+\frac{1}{2},i} &= (1-\alpha_{e_x,e_y+\frac{1}{2}})\, F^{\text{LW}}_{e_x,e_y+\frac{1}{2},i} + \alpha_{e_x,e_y+\frac{1}{2}}\, f_{\boldsymbol{e},i,N+\frac{3}{2}}, && 1 \le i \le N+1.
\end{aligned}
$$

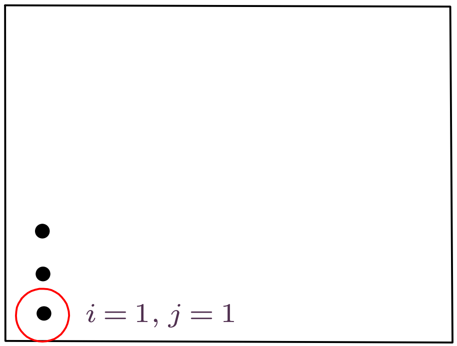
$$
\begin{aligned}
\tilde{\boldsymbol{u}}^{n+1} &= \boldsymbol{u}^n_{\boldsymbol{e},1,j} - \frac{\Delta t}{\Delta x_{\boldsymbol{e}} w_1}\Big(\tilde{F}_{e_x+\frac{1}{2},e_y,j} - f_{\boldsymbol{e},\frac{1}{2},j}\Big) - \frac{\Delta t}{\Delta y_{\boldsymbol{e}} w_j}\Big(f_{\boldsymbol{e},1,j+\frac{1}{2}} - f_{\boldsymbol{e},1,j-\frac{1}{2}}\Big), \quad 1 < j < N+1,\\
\tilde{\boldsymbol{u}}^{n+1} &= \boldsymbol{u}^n_{\boldsymbol{e},1,1} - \frac{\Delta t}{\Delta x_{\boldsymbol{e}} w_1}\Big(\tilde{F}_{e_x+\frac{1}{2},e_y,1} - f_{\boldsymbol{e},\frac{1}{2},1}\Big) - \frac{\Delta t}{\Delta y_{\boldsymbol{e}} w_1}\Big(\tilde{F}_{e_x,e_y+\frac{1}{2},1} - f_{\boldsymbol{e},1,\frac{1}{2}}\Big), \quad j = 1.
\end{aligned}
$$

In the 2-D code, there's two separate face loops for vertical and horizontal faces. This poses a challenge because to ensure $\tilde{u}^{n+1}$ is admissible, we need to correct both $\tilde{F}_{e_x+\frac{1}{2},e_y,1}$ and $\tilde{F}_{e_x,e_y+\frac{1}{2},1}$ and these values are never available together.

To avoid having to store values and doing aposteriori correction, we find appropriate λ_x, λ_y such that

$$\lambda_x + \lambda_y = 1,$$

and then, following the 1-D procedure, construct corrected $F_{e_x+\frac{1}{2},e_y,1}$ and $F_{e_x,e_y+\frac{1}{2},1}$ such that



$$
\begin{aligned}
\boldsymbol{u}^n_{\boldsymbol{e},1,1} - \frac{\Delta t}{\Delta x_{\boldsymbol{e}} \lambda_x w_1}\Big(\tilde{F}_{e_x+\frac{1}{2},e_y,1} - f_{\boldsymbol{e},\frac{1}{2},1}\Big) \in \Omega,\\
\boldsymbol{u}^n_{\boldsymbol{e},1,1} - \frac{\Delta t}{\Delta y_{\boldsymbol{e}} \lambda_y w_1}\Big(\tilde{F}_{e_x,e_y+\frac{1}{2},1} - f_{\boldsymbol{e},1,\frac{1}{2}}\Big) \in \Omega.
\end{aligned}
$$

Initial candidate:

$$
\begin{aligned}
\tilde{F}_{e_x+\frac{1}{2},e_y,j} &= (1-\alpha_{e_x+\frac{1}{2},e_y})\, F^{\text{LW}}_{e_x+\frac{1}{2},e_y,j} + \alpha_{e_x+\frac{1}{2},e_y}\, f_{\boldsymbol{e},N+\frac{3}{2},j}, \qquad 1 \le j \le N+1,\\
\tilde{F}_{e_x,e_y+\frac{1}{2},i} &= (1-\alpha_{e_x,e_y+\frac{1}{2}})\, F^{\text{LW}}_{e_x,e_y+\frac{1}{2},i} + \alpha_{e_x,e_y+\frac{1}{2}}\, f_{\boldsymbol{e},i,N+\frac{3}{2}}, \qquad 1 \le i \le N+1.
\end{aligned}
$$

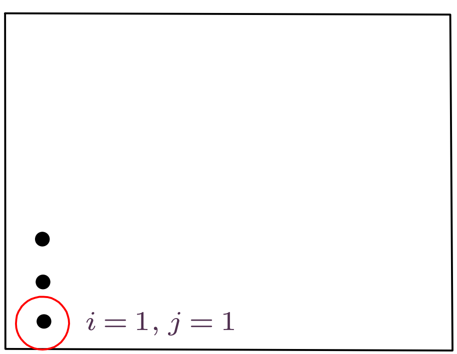
$$
\begin{aligned}
\tilde{\boldsymbol{u}}^{n+1} &= \boldsymbol{u}^n_{\boldsymbol{e},1,j} - \frac{\Delta t}{\Delta x_{\boldsymbol{e}} w_1}\Big(\tilde{F}_{e_x+\frac{1}{2},e_y,j} - f_{\boldsymbol{e},\frac{1}{2},j}\Big) - \frac{\Delta t}{\Delta y_{\boldsymbol{e}} w_j}\Big(f_{\boldsymbol{e},1,j+\frac{1}{2}} - f_{\boldsymbol{e},1,j-\frac{1}{2}}\Big), \quad 1 < j < N+1,\\
\tilde{\boldsymbol{u}}^{n+1} &= \boldsymbol{u}^n_{\boldsymbol{e},1,1} - \frac{\Delta t}{\Delta x_{\boldsymbol{e}} w_1}\Big(\tilde{F}_{e_x+\frac{1}{2},e_y,1} - f_{\boldsymbol{e},\frac{1}{2},1}\Big) - \frac{\Delta t}{\Delta y_{\boldsymbol{e}} w_1}\Big(\tilde{F}_{e_x,e_y+\frac{1}{2},1} - f_{\boldsymbol{e},1,\frac{1}{2}}\Big), \quad j = 1.
\end{aligned}
$$

In the 2-D code, there's two separate face loops for vertical and horizontal faces. This poses a challenge because to ensure $\tilde{u}^{n+1}$ is admissible, we need to correct both $\tilde{F}_{e_x+\frac{1}{2},e_y,1}$ and $\tilde{F}_{e_x,e_y+\frac{1}{2},1}$ and these values are never available together.

To avoid having to store values and doing aposteriori correction, we find appropriate λ_x, λ_y such that

$$\lambda_x + \lambda_y = 1,$$

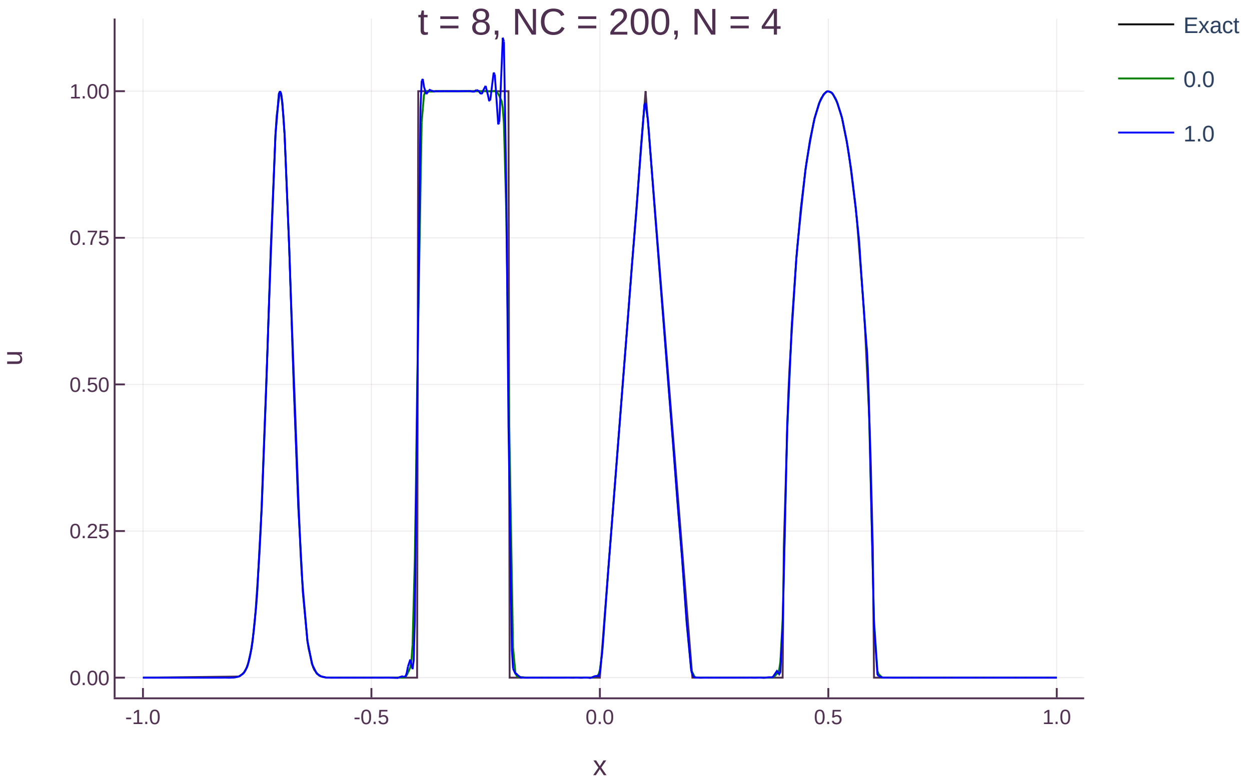
and then, following the 1-D procedure, construct corrected $F_{e_x+\frac{1}{2},e_y,1}$ and $F_{e_x,e_y+\frac{1}{2},1}$ such that



$$
\begin{aligned}
\boldsymbol{u}^n_{\boldsymbol{e},1,1} - \frac{\Delta t}{\Delta x_{\boldsymbol{e}} \lambda_x w_1}\Big(\tilde{F}_{e_x+\frac{1}{2},e_y,1} - f_{\boldsymbol{e},\frac{1}{2},1}\Big) \in \Omega,\\
\boldsymbol{u}^n_{\boldsymbol{e},1,1} - \frac{\Delta t}{\Delta y_{\boldsymbol{e}} \lambda_y w_1}\Big(\tilde{F}_{e_x,e_y+\frac{1}{2},1} - f_{\boldsymbol{e},1,\frac{1}{2}}\Big) \in \Omega.
\end{aligned}
$$

$$
\lambda_x = \frac{|s_x^{\boldsymbol{e}}|\,/\,\Delta x_{\boldsymbol{e}}}{|s_x^{\boldsymbol{e}}|\,/\,\Delta x_{\boldsymbol{e}} + |s_y^{\boldsymbol{e}}|\,/\,\Delta y_{\boldsymbol{e}}}, \qquad \lambda_y = \frac{|s_y^{\boldsymbol{e}}|\,/\,\Delta y_{\boldsymbol{e}}}{|s_x^{\boldsymbol{e}}|\,/\,\Delta x_{\boldsymbol{e}} + |s_y^{\boldsymbol{e}}|\,/\,\Delta y_{\boldsymbol{e}}}
$$

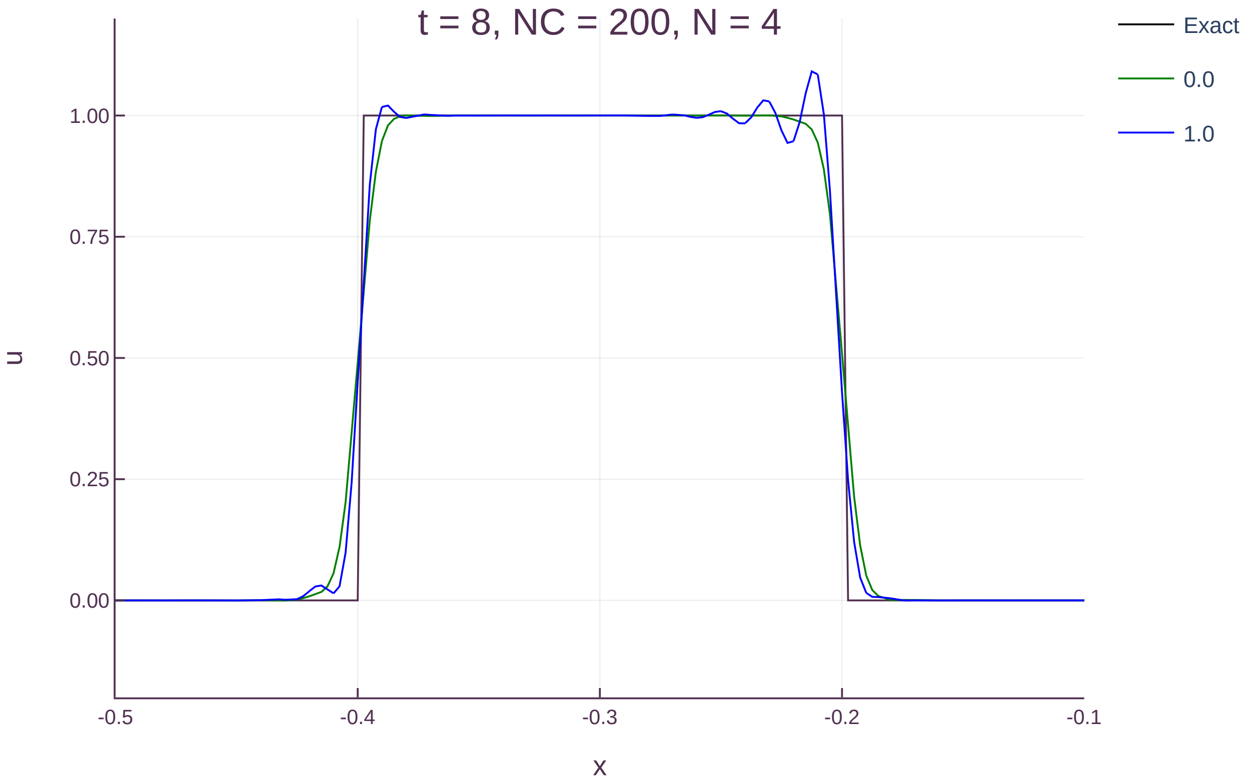
$$\mathbb{E} = \max\left(\frac{m_{N+1}^2}{\beta_1 m_1 + \sum_{j=2}^{N+1} m_j^2}, \frac{m_N^2}{\beta_2 m_1 + \sum_{j=2}^{N} m_j^2} \right)$$



$\beta_2 = 1.0$

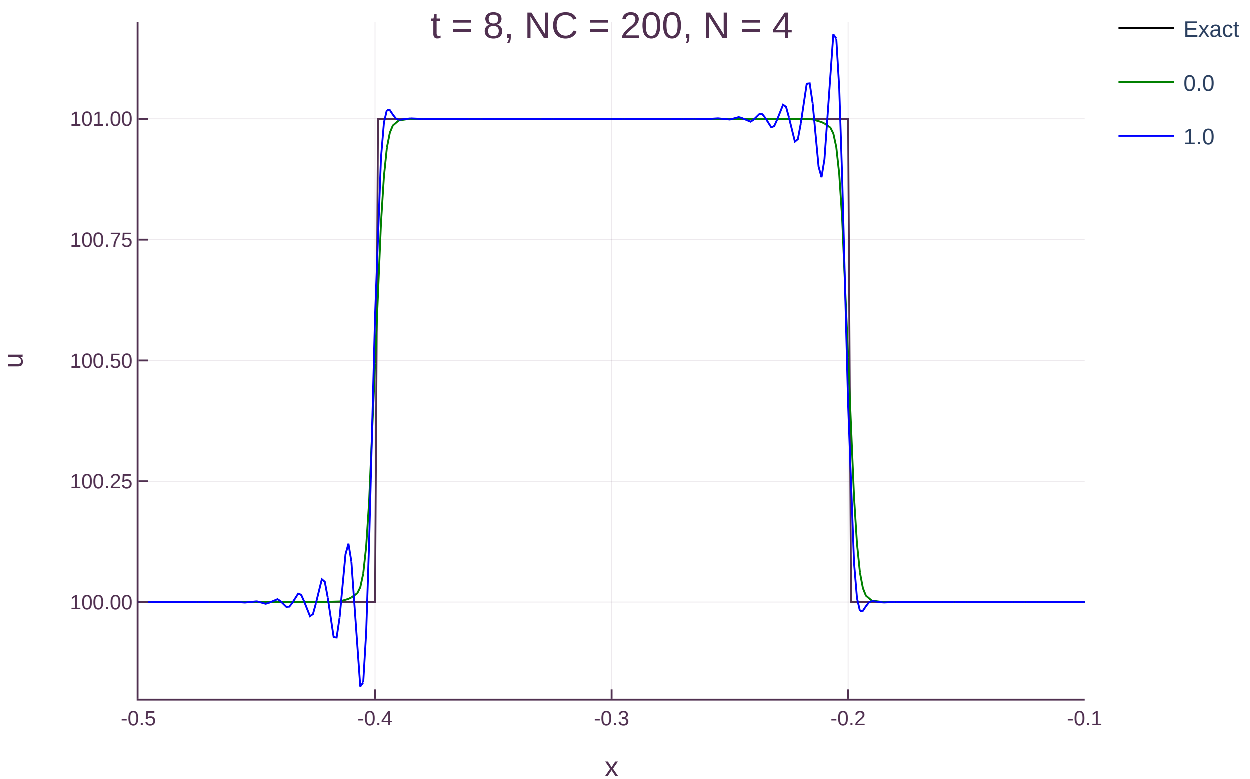
Choosing β_1, β_2

$$\mathbb{E} = \max\left(\frac{m_{N+1}^2}{\beta_1 m_1 + \sum_{j=2}^{N+1} m_j^2}, \frac{m_N^2}{\beta_2 m_1 + \sum_{j=2}^{N} m_j^2}\right)$$



$\beta_2 = 1.0$

$$\mathbb{E} = \max\left(\frac{m_{N+1}^2}{\beta_1 m_1 + \sum_{j=2}^{N+1} m_j^2}, \frac{m_N^2}{\beta_2 m_1 + \sum_{j=2}^{N} m_j^2}\right)$$



$\beta_2 = 1.0$

Lemma. *Consider the 1-D Riemann problem*

$$\begin{aligned} \boldsymbol{u}_t + \boldsymbol{f}(\boldsymbol{u})_x &= 0, \\ \boldsymbol{u}(x,0) &= \begin{cases} \boldsymbol{u}_l, & x < 0, \\ \boldsymbol{u}_r, & x > 0, \end{cases} \end{aligned}$$

in $[-h, h] \times [0, \Delta t]$ *where*

$$\frac{\Delta t}{h} |\sigma_e(\boldsymbol{u}_l, \boldsymbol{u}_r)| \leq \frac{1}{2},$$

where $\sigma_e(\boldsymbol{u}_1, \boldsymbol{u}_2)$ *denotes all eigenvalues of all Jacobian matrices at the states between* $\boldsymbol{u}_1$ *and* $\boldsymbol{u}_2$*. Then, for all* $t \leq \Delta t$*, we have*

$$\int_{-h}^{h} \boldsymbol{u}(x,t)\, dx = h\,(\boldsymbol{u}_l + \boldsymbol{u}_r) - t(\boldsymbol{f}(\boldsymbol{u}_r) - \boldsymbol{f}(\boldsymbol{u}_l)).$$

Lemma 7. ***(Evolution)*** *Pick*

$$\mu^- = \frac{x_{i+1/2} - x_i}{x_{i+1/2} - x_{i-1/2}}, \qquad \mu^+ = \frac{x_i - x_{i-1/2}}{x_{i+1/2} - x_{i-1/2}},$$

so that

$$\frac{\mu^-}{2}\boldsymbol{u}_i^{n,-} + \frac{1}{2}\boldsymbol{u}_i^{*,\pm} + \frac{\mu^+}{2}\boldsymbol{u}_i^{n,+} = \boldsymbol{u}_i^{n,\pm}.$$

Then, assume that

$$\boldsymbol{u}_i^{n,\pm} \in \mathcal{U}_{\mathrm{ad}} \quad \textit{and} \quad \boldsymbol{u}_i^{*,\pm} \in \mathcal{U}_{\mathrm{ad}},$$

and the CFL restrictions

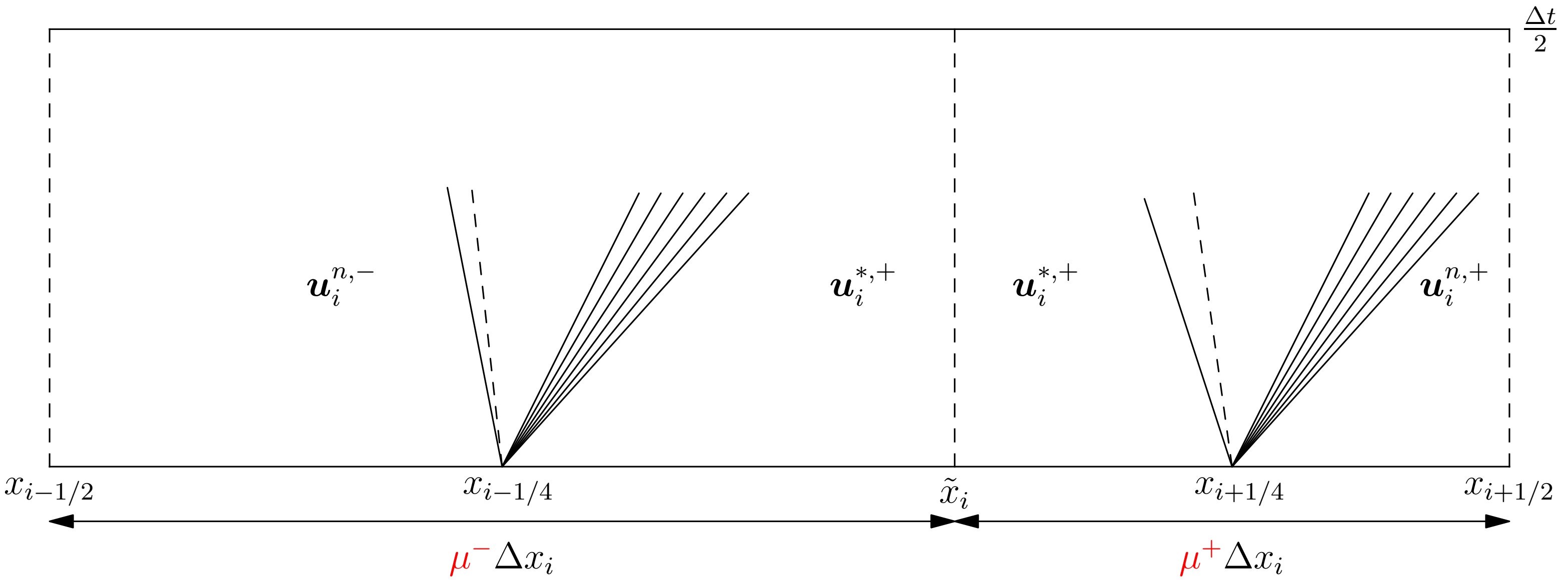
$$\begin{aligned} \frac{\Delta t/2}{\mu^- \Delta x/2} \max_{i \in \mathbb{Z}} (|\sigma_e(\boldsymbol{u}_i^{n,-}, \boldsymbol{u}_i^{*,\pm})|) \le \frac{1}{2}, \\ \frac{\Delta t/2}{\mu^+ \Delta x/2} \max_{i \in \mathbb{Z}} (|\sigma_e(\boldsymbol{u}_i^{*,\pm}, \boldsymbol{u}_i^{n,+})|) \le \frac{1}{2}, \end{aligned} \tag{5}$$

where $\sigma_e(\boldsymbol{u}_1, \boldsymbol{u}_2)$ denotes the maximum spectral radius among all Jacobian matrices at states between $\boldsymbol{u}_1$ and $\boldsymbol{u}_2$.

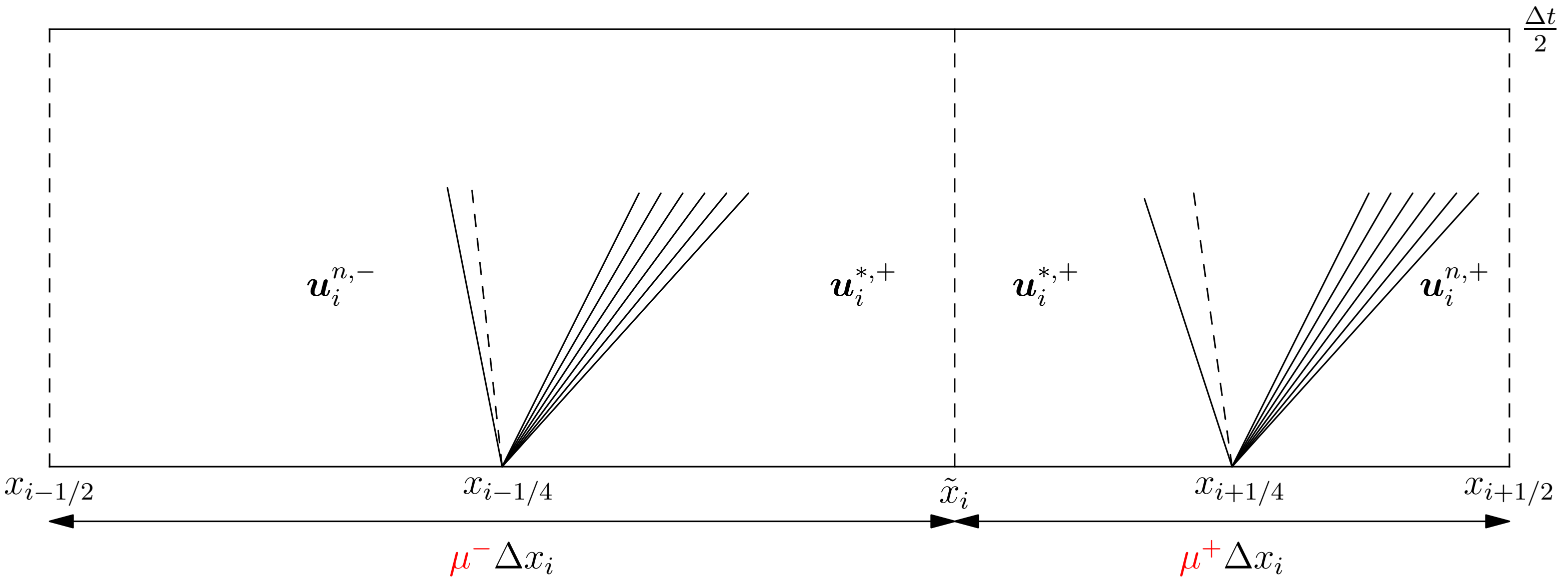
Then, we have invariance of $\mathcal{U}_{\mathrm{ad}}$ under the first step of MUSCL-Hancock scheme, i.e.,

$$\boldsymbol{u}_i^{n+1/2,\pm} \in \mathcal{U}_{\mathrm{ad}}.$$

Proof


$\frac{\Delta t}{2}$
$\boldsymbol{u}_i^{n,-}$
$\boldsymbol{u}_i^{*,+}$
$\boldsymbol{u}_i^{*,+}$
$\boldsymbol{u}_i^{n,+}$
$x_{i-1/2}$
$x_{i-1/4}$
$\tilde{x}_i$
$x_{i+1/4}$
$x_{i+1/2}$
$\mu^- \Delta x_i$
$\mu^+ \Delta x_i$

Proof



$$
\begin{aligned}
\frac{1}{\Delta x_i}\int_{x_{i-1/2}}^{x_{i+1/2}} \boldsymbol{u}^h(x,\Delta t/2)\,dx &= \frac{1}{2}(\mu^-\boldsymbol{w}_i^{n,-}+\boldsymbol{u}_i^{*,+}+\mu^+\boldsymbol{u}_i^{n,+}) - \frac{\Delta t/2}{\Delta x_i}(f(\boldsymbol{u}_i^{n,+})-f(\boldsymbol{u}_i^{n,-}))\\
&= \boldsymbol{u}_i^{n,+} - \frac{\Delta t/2}{\Delta x_i}(f(\boldsymbol{u}_i^{n,+})-f(\boldsymbol{u}_i^{n,-})) = \boldsymbol{u}_i^{n+\frac{1}{2},+}
\end{aligned}
$$

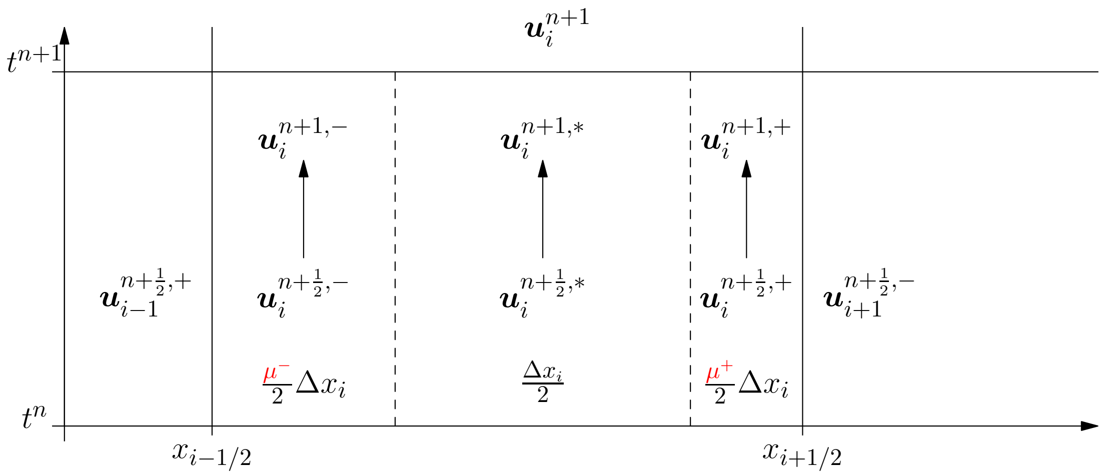
$\square$

Define $\boldsymbol{u}_i^{n+\frac{1}{2},*}$

$$\frac{\mu^-}{2}\boldsymbol{u}_i^{n+\frac{1}{2},-}+\frac{1}{2}\boldsymbol{u}_i^{n+\frac{1}{2},*}+\frac{\mu^+}{2}\boldsymbol{u}_i^{n+\frac{1}{2},+}=\boldsymbol{u}_i^n.$$

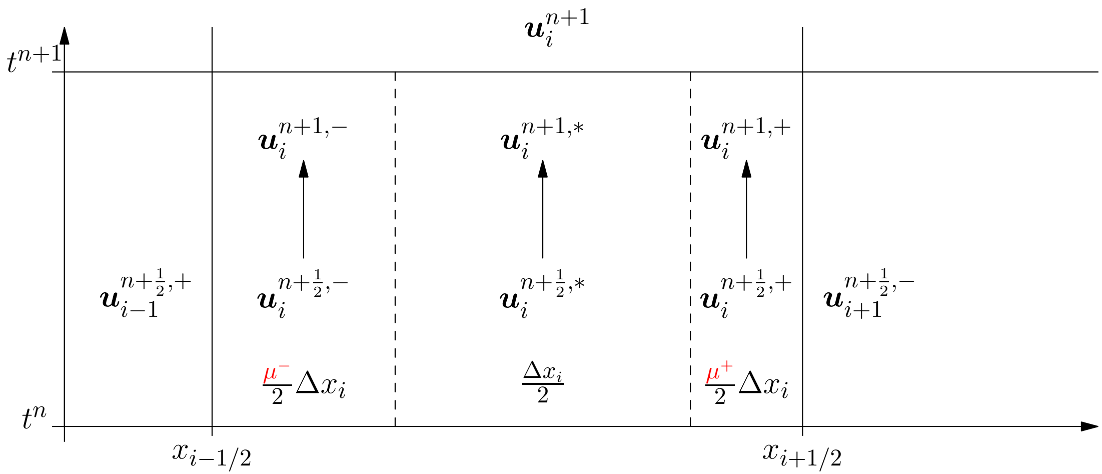
Define $\boldsymbol{u}_i^{n+\frac{1}{2},*}$

$$\frac{\mu^-}{2}\boldsymbol{u}_i^{n+\frac{1}{2},-} + \frac{1}{2}\boldsymbol{u}_i^{n+\frac{1}{2},*} + \frac{\mu^+}{2}\boldsymbol{u}_i^{n+\frac{1}{2},+} = \boldsymbol{u}_i^n.$$



Define $\boldsymbol{u}_i^{n+\frac{1}{2},*}$

$$\frac{\mu^-}{2}\boldsymbol{u}_i^{n+\frac{1}{2},-}+\frac{1}{2}\boldsymbol{u}_i^{n+\frac{1}{2},*}+\frac{\mu^+}{2}\boldsymbol{u}_i^{n+\frac{1}{2},+}=\boldsymbol{u}_i^n.$$



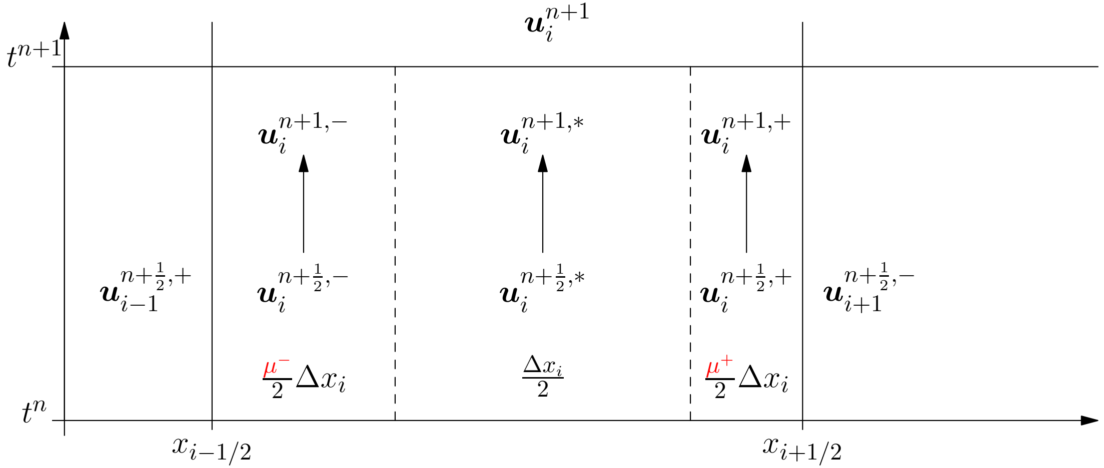
$$\boldsymbol{u}_i^{n+1,-} := \boldsymbol{u}_i^{n+\frac{1}{2},-}-\frac{\Delta t}{\mu^-\Delta x_i/2}\left(f\left(\boldsymbol{u}_i^{n+\frac{1}{2},-},\boldsymbol{u}_i^{n+\frac{1}{2},*}\right)-f\left(\boldsymbol{u}_{i-1}^{n+\frac{1}{2},+},\boldsymbol{u}_i^{n+\frac{1}{2},-}\right)\right)$$

$$\boldsymbol{u}_i^{n+1,*} := \boldsymbol{u}_i^{n+\frac{1}{2},*}-\frac{\Delta t}{\Delta x_i/2}\left(f\left(\boldsymbol{u}_i^{n+\frac{1}{2},*},\boldsymbol{u}_i^{n+\frac{1}{2},+}\right)-f\left(\boldsymbol{u}_i^{n+\frac{1}{2},-},\boldsymbol{u}_i^{n+\frac{1}{2},*}\right)\right)$$

$$\boldsymbol{u}_i^{n+1,+} := \boldsymbol{u}_i^{n+\frac{1}{2},+}-\frac{\Delta t}{\mu^+\Delta x_i/2}\left(f\left(\boldsymbol{u}_i^{n+\frac{1}{2},+},\boldsymbol{u}_{i+1}^{n+\frac{1}{2},-}\right)-f\left(\boldsymbol{u}_i^{n+\frac{1}{2},*},\boldsymbol{u}_i^{n+\frac{1}{2},+}\right)\right)$$

Define $\boldsymbol{u}_i^{n+\frac{1}{2},*}$

$$\frac{\mu^-}{2}\boldsymbol{u}_i^{n+\frac{1}{2},-}+\frac{1}{2}\boldsymbol{u}_i^{n+\frac{1}{2},*}+\frac{\mu^+}{2}\boldsymbol{u}_i^{n+\frac{1}{2},+}=\boldsymbol{u}_i^n.$$



$$\begin{aligned}
\boldsymbol{u}_i^{n+1,-} &:= \boldsymbol{u}_i^{n+\frac{1}{2},-}-\frac{\Delta t}{\mu^-\Delta x_i/2}\left(f\left(\boldsymbol{u}_i^{n+\frac{1}{2},-},\boldsymbol{u}_i^{n+\frac{1}{2},*}\right)-f\left(\boldsymbol{u}_{i-1}^{n+\frac{1}{2},+},\boldsymbol{u}_i^{n+\frac{1}{2},-}\right)\right)\\
\boldsymbol{u}_i^{n+1,*} &:= \boldsymbol{u}_i^{n+\frac{1}{2},*}-\frac{\Delta t}{\Delta x_i/2}\left(f\left(\boldsymbol{u}_i^{n+\frac{1}{2},*},\boldsymbol{u}_i^{n+\frac{1}{2},+}\right)-f\left(\boldsymbol{u}_i^{n+\frac{1}{2},-},\boldsymbol{u}_i^{n+\frac{1}{2},*}\right)\right)\\
\boldsymbol{u}_i^{n+1,+} &:= \boldsymbol{u}_i^{n+\frac{1}{2},+}-\frac{\Delta t}{\mu^+\Delta x_i/2}\left(f\left(\boldsymbol{u}_i^{n+\frac{1}{2},+},\boldsymbol{u}_{i+1}^{n+\frac{1}{2},-}\right)-f\left(\boldsymbol{u}_i^{n+\frac{1}{2},*},\boldsymbol{u}_i^{n+\frac{1}{2},+}\right)\right)
\end{aligned}$$

$$\begin{aligned}
\boldsymbol{u}_i^{n+1} &= \boldsymbol{u}_i^n-\frac{\Delta t}{\Delta x}\left(\boldsymbol{f}\left(\boldsymbol{u}_i^{n+\frac{1}{2},+},\boldsymbol{u}_{i+1}^{n+\frac{1}{2},-}\right)-\boldsymbol{f}\left(\boldsymbol{u}_{i-1}^{n+\frac{1}{2},+},\boldsymbol{u}_i^{n+\frac{1}{2},-}\right)\right)\\
&= \frac{\mu^-}{2}\boldsymbol{u}_i^{n+1,-}+\frac{1}{2}\boldsymbol{u}_i^{n+1,*}+\frac{\mu^+}{2}\boldsymbol{u}_i^{n+1,+}
\end{aligned}$$

Lemma 8. (***Riemann solver***) *Assume that the states* $\left(\boldsymbol{u}_i^{n+\frac{1}{2},\pm}\right)_{i\in\mathbb{Z}}, \left(\boldsymbol{u}_i^{n+\frac{1}{2},*}\right)_{i\in\mathbb{Z}}$ *belong to* $\mathcal{U}_{\text{ad}}$*, where* $\boldsymbol{u}_i^{n+\frac{1}{2},*}$ *was defined above as*

$$\mu^{-}\boldsymbol{u}_i^{n+\frac{1}{2},-} + \boldsymbol{u}_i^{n+\frac{1}{2},*} + \mu^{+}\boldsymbol{u}_i^{n+\frac{1}{2},+} = 2\boldsymbol{u}_i^n.$$

Then, the updated solution of MUSCL-Hancock scheme is in Ω *under the CFL conditions*

$$\begin{aligned}
&\frac{\Delta t}{\mu^{-}\Delta x_i/2}\max_{i\in\mathbb{Z}}\left(\left|\sigma_e\left(\boldsymbol{u}_i^{n+\frac{1}{2},-}, \boldsymbol{u}_i^{n+\frac{1}{2},*}\right)\right|\right) \le \frac{1}{2},\\
&\frac{\Delta t}{\mu^{-}\Delta x_i/2}\max_{i\in\mathbb{Z}}\left(\left|\sigma_e\left(\boldsymbol{u}_{i-1}^{n+\frac{1}{2},+}, \boldsymbol{u}_i^{n+\frac{1}{2},-}\right)\right|\right) \le \frac{1}{2},\\
&\frac{\Delta t}{\Delta x_i/2}\max_{i\in\mathbb{Z}}\left(\left|\sigma_e\left(\boldsymbol{u}_i^{n+\frac{1}{2},*}, \boldsymbol{u}_i^{n+\frac{1}{2},+}\right)\right|\right) \le \frac{1}{2},\\
&\frac{\Delta t}{\Delta x_i/2}\max_{i\in\mathbb{Z}}\left(\left|\sigma_e\left(\boldsymbol{u}_i^{n+\frac{1}{2},-}, \boldsymbol{u}_i^{n+\frac{1}{2},*}\right)\right|\right) \le \frac{1}{2},\\
&\frac{\Delta t}{\mu^{+}\Delta x_i/2}\max_{i\in\mathbb{Z}}\left(\left|\sigma_e\left(\boldsymbol{u}_i^{n+\frac{1}{2},+}, \boldsymbol{u}_{i+1}^{n+\frac{1}{2},-}\right)\right|\right) \le \frac{1}{2},\\
&\frac{\Delta t}{\mu^{+}\Delta x_i/2}\max_{i\in\mathbb{Z}}\left(\left|\sigma_e\left(\boldsymbol{u}_i^{n+\frac{1}{2},*}, \boldsymbol{u}_i^{n+\frac{1}{2},+}\right)\right|\right) \le \frac{1}{2}.
\end{aligned} \tag{6}$$

$$\mu^{-}\boldsymbol{u}_i^{n+\frac{1}{2},-} + \boldsymbol{u}_i^{n+\frac{1}{2},*} + \mu^{+}\boldsymbol{u}_i^{n+\frac{1}{2},+} = 2\boldsymbol{u}_i^{n}$$

Final admissibility condition

$$\boldsymbol{u}_i^{n+\frac{1}{2},*} = (2\boldsymbol{u}_i^n - (\mu^- \boldsymbol{u}_i^{n,-} + \mu^+ \boldsymbol{u}_i^{n,+})) - \frac{\Delta t}{2\Delta x_i}[(f(\boldsymbol{u}_i^{n,-}) - f(\boldsymbol{u}_i^{n,+}))]$$

$$\boldsymbol{u}_i^{n+\frac{1}{2},*} = (2\boldsymbol{u}_i^n - (\mu^- \boldsymbol{u}_i^{n,-} + \mu^+ \boldsymbol{u}_i^{n,+})) - \frac{\Delta t}{2\Delta x_i}[(f(\boldsymbol{u}_i^{n,-}) - f(\boldsymbol{u}_i^{n,+}))]$$

Lemma 13. (***Link previous lemmas***) *Define* $\boldsymbol{u}_i^{*,*}$ *to satisfy*

$$\mu^- \boldsymbol{u}_i^{n,-} + \boldsymbol{u}_i^{*,*} + \mu^+ \boldsymbol{u}_i^{n,+} = 2(2\boldsymbol{u}_i^n - (\mu^- \boldsymbol{u}_i^{n,-} + \mu^+ \boldsymbol{u}_i^{n,+})), \tag{11}$$

where, as defined before,

$$\mu^- = \frac{x_{i+1/2} - x_i}{x_{i+1/2} - x_{i-1/2}}, \qquad \mu^+ = \frac{x_i - x_{i-1/2}}{x_{i+1/2} - x_{i-1/2}}.$$

Assume that $\boldsymbol{u}_i^{n,\pm}$ *and* $\boldsymbol{u}_i^{*,*}$ *are in* $\mathcal{U}_{\mathrm{ad}}$. *Consider the CFL conditions*

$$\begin{aligned} \frac{\Delta t/2}{\mu^- \Delta x/2} \max_{i\in\mathbb{Z}}(|\sigma_e(\boldsymbol{u}_i^{*,*}, \boldsymbol{u}_i^{n,-})|) \le \frac{1}{2}, \\ \frac{\Delta t/2}{\mu^+ \Delta x/2} \max_{i\in\mathbb{Z}}(|\sigma_e(\boldsymbol{u}_i^{n,+}, \boldsymbol{u}_i^{*,*})|) \le \frac{1}{2}, \end{aligned} \tag{12}$$

then $\boldsymbol{u}_i^{n+\frac{1}{2},*} \in \mathcal{U}_{\mathrm{ad}}$.

$$\boldsymbol{u}_i^{n+\frac{1}{2},*} = (2\boldsymbol{u}_i^n - (\mu^- \boldsymbol{u}_i^{n,-} + \mu^+ \boldsymbol{u}_i^{n,+})) - \frac{\Delta t}{2\Delta x_i}[(f(\boldsymbol{u}_i^{n,-}) - f(\boldsymbol{u}_i^{n,+}))]$$

Lemma 15. (***Link previous lemmas***) *Define $\boldsymbol{u}_i^{*,*}$ to satisfy*

$$\mu^- \boldsymbol{u}_i^{n,-} + \boldsymbol{u}_i^{*,*} + \mu^+ \boldsymbol{u}_i^{n,+} = 2(2\boldsymbol{u}_i^n - (\mu^- \boldsymbol{u}_i^{n,-} + \mu^+ \boldsymbol{u}_i^{n,+})), \tag{13}$$

where, as defined before,

$$\mu^- = \frac{x_{i+1/2} - x_i}{x_{i+1/2} - x_{i-1/2}}, \qquad \mu^+ = \frac{x_i - x_{i-1/2}}{x_{i+1/2} - x_{i-1/2}}.$$

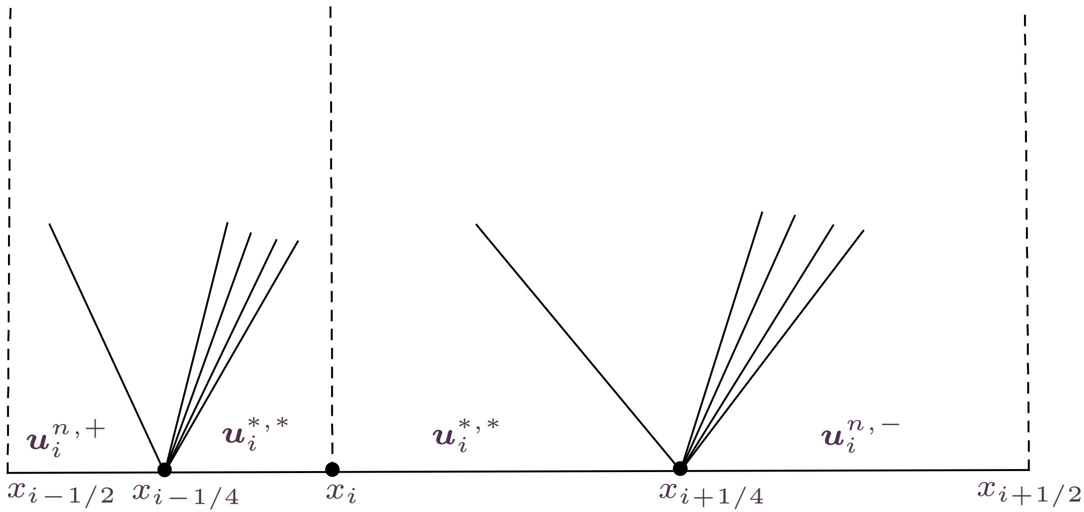
Assume that $\boldsymbol{u}_i^{n,\pm}$ and $\boldsymbol{u}_i^{*,*}$ are in $\mathcal{U}_{\mathrm{ad}}$. Consider the CFL conditions

$$\begin{aligned} \frac{\Delta t/2}{\mu^- \Delta x/2} \max_{i\in\mathbb{Z}}(|\sigma_e(\boldsymbol{u}_i^{*,*}, \boldsymbol{u}_i^{n,-})|) \leq \frac{1}{2}, \\ \frac{\Delta t/2}{\mu^+ \Delta x/2} \max_{i\in\mathbb{Z}}(|\sigma_e(\boldsymbol{u}_i^{n,+}, \boldsymbol{u}_i^{*,*})|) \leq \frac{1}{2}, \end{aligned} \tag{14}$$

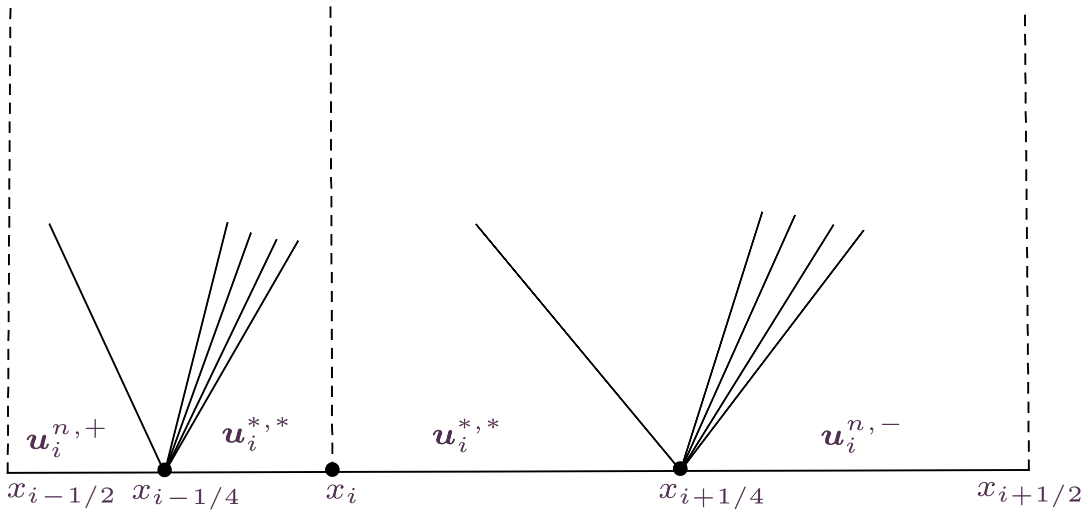
then $\boldsymbol{u}_i^{n+\frac{1}{2},*} \in \mathcal{U}_{\mathrm{ad}}$.

Remark 16. For conservative reconstruction, we actually have $\boldsymbol{u}_i^{*,*} = \boldsymbol{u}_i$. So, this lemma isn't placing new restrictions.

Proof.


$\boldsymbol{u}_i^{n,+}$
$\boldsymbol{u}_i^{*,*}$
$\boldsymbol{u}_i^{*,*}$
$\boldsymbol{u}_i^{n,-}$
$x_{i-1/2}$
$x_{i-1/4}$
x_i
$x_{i+1/4}$
$x_{i+1/2}$

Proof.



$$\begin{aligned}\frac{1}{\Delta x_i}\int_{x_{i-1/2}}^{x_{i+1/2}} \boldsymbol{u}^h\left(x, \frac{\Delta t}{2}\right) dx &= \frac{1}{2}(\mu^+ \boldsymbol{u}_i^{n,+} + \boldsymbol{u}_i^{*,*} + \mu^- \boldsymbol{u}_i^{n,-}) - \frac{\Delta t}{2\,\Delta x_i}(f(\boldsymbol{u}_i^{n,-}) - f(\boldsymbol{u}_i^{n,+})) \\ &= \boldsymbol{u}_i^{n+\frac{1}{2},*}\end{aligned}$$

Theorem 17. *Consider the conservation law*

$$\boldsymbol{u}_t + \boldsymbol{f}(\boldsymbol{u})_x = 0$$

which preserves a convex set $\mathcal{U}_{\mathrm{ad}}$*. Let* $\{\boldsymbol{u}_i^n\}_{i\in\mathbb{Z}}$ *be the approximate solution at time level* n *and assume that* $\boldsymbol{u}_i^n \in \mathcal{U}_{\mathrm{ad}}$ *for all* $i \in \mathbb{Z}$*. Consider conservative reconstructions*

$$\boldsymbol{u}_i^{n,+} = \boldsymbol{u}_i + (x_{i+1/2} - x_i)\boldsymbol{\sigma}_i, \quad \boldsymbol{u}_i^{n,-} = \boldsymbol{u}_i + (x_{i-1/2} - x_i)\boldsymbol{\sigma}_i.$$

Define $\boldsymbol{u}_i^{*,\pm}$ *to be*

$$\boldsymbol{u}_i^{*,\pm} = \boldsymbol{u}_i^n + 2\Big(x_{i\pm\frac{1}{2}} - x_i\Big)\,\boldsymbol{\sigma}_i$$

and assume that the slope σ_i *is chosen so that*

$$\boldsymbol{u}_i^{*,\pm} \in \mathcal{U}_{\mathrm{ad}}.$$

Then, under time step restrictions (5), (6), (14), the updated solution $\boldsymbol{u}_i^{n+1}$*, defined by the MUSCL-Hancock procedure is in* $\mathcal{U}_{\mathrm{ad}}$*.*

Theorem 18. *Consider the conservation law*

$$\boldsymbol{u}_t + \boldsymbol{f}(\boldsymbol{u})_x = 0$$

which preserves a convex set $\mathcal{U}_{\text{ad}}$*. Let* $\{\boldsymbol{u}_i^n\}_{i\in\mathbb{Z}}$ *be the approximate solution at time level* n *and assume that* $\boldsymbol{u}_i^n \in \mathcal{U}_{\text{ad}}$ *for all* $i \in \mathbb{Z}$*. Consider conservative reconstructions*

$$\boldsymbol{u}_i^{n,+} = \boldsymbol{u}_i + (x_{i+1/2} - x_i)\boldsymbol{\sigma}_i, \quad \boldsymbol{u}_i^{n,-} = \boldsymbol{u}_i + (x_{i-1/2} - x_i)\boldsymbol{\sigma}_i.$$

Define $\boldsymbol{u}_i^{*,\pm}$ *to be*

$$\boldsymbol{u}_i^{*,\pm} = \boldsymbol{u}_i^n + 2\Big(x_{i\pm\frac{1}{2}} - x_i\Big)\boldsymbol{\sigma}_i$$

and assume that the slope σ_i *is chosen so that*

$$\boldsymbol{u}_i^{*,\pm} \in \mathcal{U}_{\text{ad}}.$$

Then, under time step restrictions (5), (6), (14), the updated solution $\boldsymbol{u}_i^{n+1}$*, defined by the MUSCL-Hancock procedure is in* $\mathcal{U}_{\text{ad}}$*.*

Proof. We only need $\boldsymbol{u}_i^{n,\pm} \in \mathcal{U}_{\text{ad}}$ to apply the Lemmas 7, 8, 15. To that end, notice

$$\boldsymbol{u}_i^{n,\pm} = \frac{1}{2}\boldsymbol{u}_i^{*,\pm} + \frac{1}{2}\boldsymbol{u}_i^n. \qquad \square$$

Given candidate slope $\boldsymbol{\sigma}_i$,

$$\boldsymbol{u}_i^{*,\pm} := \boldsymbol{u}_i^n + 2\,(x_{i\pm1/2} - x_i)\,\boldsymbol{\sigma}_i.$$

Find $\theta \in [0,1]$

$$\boldsymbol{u}_i^n + 2\,(x_{i\pm1/2} - x_i)\,\theta\,\boldsymbol{\sigma}_i \in \mathcal{U}_{\mathrm{ad}}. \tag{15}$$

For concave $p = p(\boldsymbol{u})$, assume

$$\mathcal{U}_{\mathrm{ad}} = \{\boldsymbol{u} \in \Omega \colon p(\boldsymbol{u}) > 0\}$$

We pick

$$\theta_{\pm} = \min\left(\left|\frac{\epsilon - p(\boldsymbol{u}_i^n)}{p(\boldsymbol{u}_i^{*,\pm}) - p(\boldsymbol{u}_i^n)}\right|, 1\right)$$

and

$$\theta = \min\,(\theta_+, \theta_-).$$

By concavity,

$$p(\theta\,\boldsymbol{u}_i^{*,\pm} + (1-\theta)\,\boldsymbol{u}_i^n) > \theta\,p(\boldsymbol{u}_i^{*,\pm}) + (1-\theta)\,p(\boldsymbol{u}_i^n) > \epsilon > 0.$$

Thus, this θ will give us (15).

Consider non-conservative variables

$$\boldsymbol{U}_i^n = \kappa(\boldsymbol{u}_i^n),$$

so that reconstruction is given by

$$\boldsymbol{U}^n(x) = \boldsymbol{U}_i^n + \sigma_i(x - x_i)$$

$$\boldsymbol{u}_i^{n,\pm} \; := \; \kappa^{-1}(\boldsymbol{U}_i^{n,\pm}) \tag{16}$$

Theorem 19. *Assume that $\boldsymbol{u}_i^n \in \Omega$ for all $i \in \mathbb{Z}$. Consider $\boldsymbol{u}_i^{n,\pm}$ defined in (16), $\boldsymbol{u}_i^{*,\pm}$ defined so that*

$$\mu^- \boldsymbol{u}_i^{n+\frac{1}{2},-} + \boldsymbol{u}_i^{n+\frac{1}{2},*} + \mu^+ \boldsymbol{u}_i^{n+\frac{1}{2},+} = 2\boldsymbol{u}_i^n,$$

and $\boldsymbol{u}_i^{*,*}$ defined explicitly as

$$\boldsymbol{u}_i^{*,*} = 4\boldsymbol{u}_i^n - 3(\mu^- \boldsymbol{u}_i^{n,-} + \mu^+ \boldsymbol{u}_i^{n,+}).$$

Assume that the slope is chosen so that

$$\boldsymbol{u}_i^{n,\pm} \in \Omega, \quad \boldsymbol{u}_i^{*,\pm} \in \Omega \quad \text{and} \quad \boldsymbol{u}_i^{*,*} \in \Omega.$$

Consider the same CFL conditions (5),(6),(14). Then the updated solution $\boldsymbol{u}_i^{n+1}$ of MUSCL-Hancock procedure is in Ω.

Remark 20. The definition of $\boldsymbol{u}_i^{*,*}$ comes from

$$\mu^- \boldsymbol{u}_i^{n,-} + \boldsymbol{u}_i^{*,*} + \mu^+ \boldsymbol{u}_i^{n,+} = 2(2\boldsymbol{u}_i^n - (\mu^- \boldsymbol{u}_i^{n,-} + \mu^+ \boldsymbol{u}_i^{n,+}))$$

$$\int_{t_n}^{t_{n+1}} \int_{I_e} (\partial_t u_h + \partial_x f(u_h)) \ell_i(\xi)\, dx\, dt = 0$$

$$\int_{t_n}^{t_{n+1}}\int_{I_e}(\partial_t u_h+\partial_x f(u_h))\ell_i(\xi)\,dx\,dt=0$$

Integrate by parts in space

$$\begin{aligned}\int_{I_e} u_h^{n+1}\ell_i(\xi)\,dx \;=\; & \int_{I_e} u_h^n\,\ell_i(\xi)\,dx+\int_{t^n}^{t^{n+1}}\int_{I_e}\tilde{f}_h\partial_x\ell_i(\xi)\,dx\,dt\\ & +\int_{t^n}^{t^{n+1}} f(\tilde{u}^-_{e-\frac{1}{2}}(t),\tilde{u}^+_{e-\frac{1}{2}}(t))\ell_i(0)dt-\int_{t^n}^{t^{n+1}} f(\tilde{u}^-_{e+\frac{1}{2}}(t),\tilde{u}^+_{e+\frac{1}{2}}(t))\ell_i(1)dt.\end{aligned}$$

$$\int_{t_n}^{t_{n+1}} \int_{I_e} (\partial_t u_h + \partial_x f(u_h)) \ell_i(\xi)\, dx\, dt = 0$$

Integrate by parts in space

$$\begin{aligned}\int_{I_e} u_h^{n+1} \ell_i(\xi)\, dx &= \int_{I_e} u_h^n\, \ell_i(\xi)\, dx + \int_{t^n}^{t^{n+1}} \int_{I_e} \tilde{f}_h \partial_x \ell_i(\xi)\, dx\, dt \\ &+ \int_{t^n}^{t^{n+1}} f(\tilde{u}^-_{e-\frac{1}{2}}(t), \tilde{u}^+_{e-\frac{1}{2}}(t)) \ell_i(0) dt - \int_{t^n}^{t^{n+1}} f(\tilde{u}^-_{e+\frac{1}{2}}(t), \tilde{u}^+_{e+\frac{1}{2}}(t)) \ell_i(1) dt.\end{aligned}$$

Another integration by parts

$$\begin{aligned}\int_{I_e} u_h^{n+1} \ell_i(\xi) dx &= \int_{I_e} u_h^n \ell_i(\xi) dx - \int_{t^n}^{t^{n+1}} \int_{I_e} \partial_x \tilde{f}_h \ell_i(\xi)\, dx\, dt \\ &+ \int_{t^n}^{t^{n+1}} \left(f(\tilde{u}^-_{e-\frac{1}{2}}(t), \tilde{u}^+_{e-\frac{1}{2}}(t)) - \tilde{f}_h(0, \tau) \right) \ell_i(0) dt \\ &- \int_{t^n}^{t^{n+1}} \left(f(\tilde{u}^-_{e+\frac{1}{2}}(t), \tilde{u}^+_{e+\frac{1}{2}}(t)) - \tilde{f}_h(1, \tau) \right) \ell_i(1) dt,\end{aligned}$$

$$\int_{t_n}^{t_{n+1}}\int_{I_e}(\partial_t u_h + \partial_x f(u_h))\ell_i(\xi)\,dx\,dt = 0$$

Integrate by parts in space

$$\begin{aligned}\int_{I_e} u_h^{n+1}\ell_i(\xi)\,dx &= \int_{I_e} u_h^n\,\ell_i(\xi)\,dx + \int_{t^n}^{t^{n+1}}\int_{I_e}\tilde{f}_h\partial_x\ell_i(\xi)\,dx\,dt \\ &+ \int_{t^n}^{t^{n+1}} f(\tilde{u}^-_{e-\frac{1}{2}}(t), \tilde{u}^+_{e-\frac{1}{2}}(t))\ell_i(0)dt - \int_{t^n}^{t^{n+1}} f(\tilde{u}^-_{e+\frac{1}{2}}(t), \tilde{u}^+_{e+\frac{1}{2}}(t))\ell_i(1)dt.\end{aligned}$$

Another integration by parts

$$\begin{aligned}\int_{I_e} u_h^{n+1}\ell_i(\xi)dx &= \int_{I_e} u_h^n\ell_i(\xi)dx - \int_{t^n}^{t^{n+1}}\int_{I_e}\partial_x\tilde{f}_h\ell_i(\xi)\,dx\,dt \\ &+\int_{t^n}^{t^{n+1}}\Big(f(\tilde{u}^-_{e-\frac{1}{2}}(t), \tilde{u}^+_{e-\frac{1}{2}}(t)) - \tilde{f}_h(0,\tau)\Big)\ell_i(0)dt \\ &-\int_{t^n}^{t^{n+1}}\Big(f(\tilde{u}^-_{e+\frac{1}{2}}(t), \tilde{u}^+_{e+\frac{1}{2}}(t)) - \tilde{f}_h(1,\tau)\Big)\ell_i(1)dt,\end{aligned}$$

Quadrature on solution points

$$\begin{aligned} & u_i^n - \int_{t^n}^{t^{n+1}}\partial_x\tilde{f}_h(\xi_i,\tau)\,dt \\ \Rightarrow\ u_i^{n+1} \ = \ & + \frac{\ell_i(0)}{w_i}\int_{t^n}^{t^{n+1}} f(\tilde{u}^-_{e-\frac{1}{2}}(t), \tilde{u}^+_{e-\frac{1}{2}}(t)) - \tilde{f}_h(0,\tau)\,dt \\ & - \frac{\ell_i(1)}{w_i}\int_{t^n}^{t^{n+1}} f(\tilde{u}^-_{e+\frac{1}{2}}(t), \tilde{u}^+_{e+\frac{1}{2}}(t)) - \tilde{f}_h(1,\tau)\,dt\end{aligned}$$

$$\int_{t_n}^{t_{n+1}} \int_{I_e} (\partial_t u_h + \partial_x f(u_h))\ell_i(\xi)\, dx\, dt = 0$$

Integrate by parts in space

$$\begin{aligned}\int_{I_e} u_h^{n+1}\ell_i(\xi)\, dx &= \int_{I_e} u_h^n\, \ell_i(\xi)\, dx + \int_{t^n}^{t^{n+1}} \int_{I_e} \tilde{f}_h \partial_x \ell_i(\xi)\, dx\, dt \\ &+ \int_{t^n}^{t^{n+1}} f(\tilde{u}^-_{e-\frac{1}{2}}(t), \tilde{u}^+_{e-\frac{1}{2}}(t))\ell_i(0) dt - \int_{t^n}^{t^{n+1}} f(\tilde{u}^-_{e+\frac{1}{2}}(t), \tilde{u}^+_{e+\frac{1}{2}}(t))\ell_i(1) dt.\end{aligned}$$

Another integration by parts

$$\begin{aligned}\int_{I_e} u_h^{n+1}\ell_i(\xi) dx &= \int_{I_e} u_h^n \ell_i(\xi) dx - \int_{t^n}^{t^{n+1}} \int_{I_e} \partial_x \tilde{f}_h \ell_i(\xi)\, dx\, dt \\ &+ \int_{t^n}^{t^{n+1}} \left(f(\tilde{u}^-_{e-\frac{1}{2}}(t), \tilde{u}^+_{e-\frac{1}{2}}(t)) - \tilde{f}_h(0,\tau) \right) \ell_i(0) dt \\ &- \int_{t^n}^{t^{n+1}} \left(f(\tilde{u}^-_{e+\frac{1}{2}}(t), \tilde{u}^+_{e+\frac{1}{2}}(t)) - \tilde{f}_h(1,\tau) \right) \ell_i(1) dt,\end{aligned}$$

Quadrature on solution points

$$\begin{aligned} & \quad u_i^n - \int_{t^n}^{t^{n+1}} \partial_x \tilde{f}_h(\xi_i, \tau)\, dt \\ \Rightarrow\ u_i^{n+1} &= \quad - g_L'(\xi_i) \int_{t^n}^{t^{n+1}} f(\tilde{u}^-_{e-\frac{1}{2}}(t), \tilde{u}^+_{e-\frac{1}{2}}(t)) - \tilde{f}_h(0,\tau) dt \\ & \quad - g_R'(\xi_i) \int_{t^n}^{t^{n+1}} f(\tilde{u}^-_{e+\frac{1}{2}}(t), \tilde{u}^+_{e+\frac{1}{2}}(t)) - \tilde{f}_h(1,\tau) dt \end{aligned}$$

Theorem 21. *If the predictor solution of ADER satisfies*

$$\|U^n\|_{C^{N+1}} \leq C$$

where C *is a constant independent of* $n, \Delta x, \Delta t$, *then the ADER and LW solution will satisfy*

$$\|u^{n+1}_{\text{ADER}} - u^{n+1}_{\text{LW}}\|_\infty = O(\Delta t^{N+1}).$$

Theorem 22. *If the predictor solution of ADER satisfies*

$$\|U^n\|_{C^{N+1}} \leq C$$

where C *is a constant independent of* $n, \Delta x, \Delta t$*, then the ADER and LW solution will satisfy*

$$\|u_{\mathrm{ADER}}^{n+1} - u_{\mathrm{LW}}^{n+1}\|_{\infty} = O(\Delta t^{N+1}).$$

Idea of proof. The weak formulation will give us

$$\tilde{u}_t + (\tilde{f}_h)_x = 0, \tag{18}$$

for all (x_r, t_s) where $s > 0$, i.e., $t_s > t^n$. Then, we can extrapolate to $t = t^n$ as

Theorem 23. *If the predictor solution of ADER satisfies*

$$\|U^n\|_{C^{N+1}} \leq C$$

where C *is a constant independent of* $n, \Delta x, \Delta t$*, then the ADER and LW solution will satisfy*

$$\|u_{\text{ADER}}^{n+1} - u_{\text{LW}}^{n+1}\|_\infty = O(\Delta t^{N+1}).$$

Idea of proof. The weak formulation will give us

$$\tilde{u}_t + (\tilde{f}_h)_x = 0, \tag{19}$$

for all (x_r, t_s) where $s > 0$, i.e., $t_s > t^n$. Then, we can extrapolate to $t = t^n$ as

$$\tilde{u}_t + (\tilde{f}_h)_x = O(\Delta t^N) \qquad \text{at} \quad t = t^n,$$

Theorem 24. *If the predictor solution of ADER satisfies*

$$\|U^n\|_{C^{N+1}} \leq C$$

where C *is a constant independent of* $n, \Delta x, \Delta t$*, then the ADER and LW solution will satisfy*

$$\|u_{\text{ADER}}^{n+1} - u_{\text{LW}}^{n+1}\|_\infty = O(\Delta t^{N+1}).$$

Idea of proof. The weak formulation will give us

$$\tilde{u}_t + (\tilde{f}_h)_x = 0, \tag{20}$$

for all (x_r, t_s) where $s > 0$, i.e., $t_s > t^n$. Then, we can extrapolate to $t = t^n$ as

$$\tilde{u}_t + (\tilde{f}_h)_x = O(\Delta t^N) \qquad \text{at} \quad t = t^n,$$

so that we will have

$$\tilde{u}_h(x, t^n) \;=\; \tilde{u}_h(x, t^n) + \Delta t\, (\tilde{f}_h)_x + \ldots + \frac{\Delta t^N}{N!} \frac{\partial^{N-1}}{\partial t^{N-1}} (\tilde{f}_h)_x + O(\Delta t^{N+1}).$$