An invariant domain preserving FE technique for hyperbolic systems

Jean-Luc Guermond and Bojan Popov

Department of Mathematics
Texas A&M University

Advanced numerical methods:
Recent developments, analysis and applications
October 03-07, 2016
IHP, Paris

Acknowledgments

Collaborators:
Murtazo Nazarov (Uppsala University)
Vladimir Tomov (LLNL)
Young Yang (Penn State University)
Laura Saavedra (Universidad Politécnica de Madrid)

Support:

Hyperbolic systems

Hyperbolic systems

Hyperbolic systems

The PDEs

- Hyperbolic system

$$\partial_t \mathbf{u} + \nabla\cdot\mathbf{f}(\mathbf{u}) = 0, \qquad (\mathbf{x}, t) \in D\times\mathbb{R}_+.$$
$$u(\mathbf{x}, 0) = \mathbf{u}_0(\mathbf{x}), \qquad \mathbf{x} \in D.$$

- D open polyhedral domain in $\mathbb{R}^d$.
- $\mathbf{f} \in \mathcal{C}^1(\mathbb{R}^m; \mathbb{R}^{m\times d})$, the flux.
- $\mathbf{u}_0$, admissible initial data.
- Periodic BCs or $\mathbf{u}_0$ has compact support (to simplify BCs)

Hyperbolic systems

The PDEs

- Hyperbolic system

$$\partial_t \mathbf{u} + \nabla\cdot\mathbf{f}(\mathbf{u}) = 0, \qquad (\mathbf{x}, t) \in D\times\mathbb{R}_+.$$
$$u(\mathbf{x}, 0) = \mathbf{u}_0(\mathbf{x}), \qquad \mathbf{x} \in D.$$

- D open polyhedral domain in $\mathbb{R}^d$.
- $\mathbf{f} \in \mathcal{C}^1(\mathbb{R}^m; \mathbb{R}^{m\times d})$, the flux.
- $\mathbf{u}_0$, admissible initial data.
- Periodic BCs or $\mathbf{u}_0$ has compact support (to simplify BCs)

Hyperbolic systems

The PDEs

- Hyperbolic system

$$\partial_t \mathbf{u} + \nabla\cdot\mathbf{f}(\mathbf{u}) = 0, \qquad (\mathbf{x}, t) \in D\times\mathbb{R}_+.$$
$$u(\mathbf{x}, 0) = \mathbf{u}_0(\mathbf{x}), \qquad \mathbf{x} \in D.$$

- D open polyhedral domain in $\mathbb{R}^d$.
- $\mathbf{f} \in \mathcal{C}^1(\mathbb{R}^m;\mathbb{R}^{m\times d})$, the flux.
- $\mathbf{u}_0$, admissible initial data.
- Periodic BCs or $\mathbf{u}_0$ has compact support (to simplify BCs)

Hyperbolic systems

The PDEs

- Hyperbolic system

$$\partial_t \mathbf{u} + \nabla\cdot\mathbf{f}(\mathbf{u}) = 0, \qquad (\mathbf{x}, t) \in D\times\mathbb{R}_+.$$
$$u(\mathbf{x}, 0) = \mathbf{u}_0(\mathbf{x}), \qquad \mathbf{x} \in D.$$

- D open polyhedral domain in $\mathbb{R}^d$.
- $\mathbf{f} \in \mathcal{C}^1(\mathbb{R}^m; \mathbb{R}^{m\times d})$, the flux.
- $\mathbf{u}_0$, admissible initial data.
- Periodic BCs or $\mathbf{u}_0$ has compact support (to simplify BCs)

Hyperbolic systems

The PDEs

- Hyperbolic system

$$\partial_t \mathbf{u} + \nabla\cdot\mathbf{f}(\mathbf{u}) = 0, \qquad (\mathbf{x}, t) \in D\times\mathbb{R}_+.$$
$$u(\mathbf{x}, 0) = \mathbf{u}_0(\mathbf{x}), \qquad \mathbf{x} \in D.$$

- D open polyhedral domain in $\mathbb{R}^d$.
- $\mathbf{f} \in \mathcal{C}^1(\mathbb{R}^m; \mathbb{R}^{m\times d})$, the flux.
- $\mathbf{u}_0$, admissible initial data.
- Periodic BCs or $\mathbf{u}_0$ has compact support (to simplify BCs)

Formulation of the problem

Assumptions

- $\exists$ admissible set $\mathcal{A}$ s.t. for all $(\mathbf{u}_l, \mathbf{u}_r) \in \mathcal{A}$ the 1D Riemann problem

$$\partial_t \mathbf{v} + \partial_x(\mathbf{n}{\cdot}\mathbf{f}(\mathbf{v})) = 0, \quad \mathbf{v}(x,0) = \begin{cases} \mathbf{u}_l & \text{if } x < 0 \\ \mathbf{u}_r & \text{if } x > 0. \end{cases}$$

has a unique "entropy" solution $\mathbf{u}(\mathbf{u}_l, \mathbf{u}_r)(x,t)$ for all $\mathbf{n} \in \mathbb{R}^d$, $\|\mathbf{n}\|_{\ell^2} = 1$.

- There exists an invariant set $A \subset \mathcal{A}$, i.e.,

$$\mathbf{u}(\mathbf{u}_l, \mathbf{u}_r)(x,t) \in A, \quad \forall t \geq 0, \forall x \in \mathbb{R}, \quad \forall \mathbf{u}_l, \mathbf{u}_r \in A.$$

- A is convex.

Formulation of the problem

Assumptions

- $\exists$ admissible set $\mathcal{A}$ s.t. for all $(\mathbf{u}_l, \mathbf{u}_r) \in \mathcal{A}$ the 1D Riemann problem

$$\partial_t \mathbf{v} + \partial_x(\mathbf{n}\cdot\mathbf{f}(\mathbf{v})) = 0, \quad \mathbf{v}(x,0) = \begin{cases} \mathbf{u}_l & \text{if } x < 0 \\ \mathbf{u}_r & \text{if } x > 0. \end{cases}$$

  has a unique "entropy" solution $\mathbf{u}(\mathbf{u}_l, \mathbf{u}_r)(x,t)$ for all $\mathbf{n} \in \mathbb{R}^d$, $\|\mathbf{n}\|_{\ell^2} = 1$.
- There exists an invariant set $A \subset \mathcal{A}$, i.e.,

$$\mathbf{u}(\mathbf{u}_l, \mathbf{u}_r)(x,t) \in A, \quad \forall t \geq 0, \forall x \in \mathbb{R}, \quad \forall \mathbf{u}_l, \mathbf{u}_r \in A.$$

- A is convex.

Formulation of the problem

Assumptions

- $\exists$ admissible set $\mathcal{A}$ s.t. for all $(\mathbf{u}_l, \mathbf{u}_r) \in \mathcal{A}$ the 1D Riemann problem

$$\partial_t \mathbf{v} + \partial_x(\mathbf{n}\cdot\mathbf{f}(\mathbf{v})) = 0, \quad \mathbf{v}(x,0) = \begin{cases} \mathbf{u}_l & \text{if } x < 0 \\ \mathbf{u}_r & \text{if } x > 0. \end{cases}$$

  has a unique "entropy" solution $\mathbf{u}(\mathbf{u}_l, \mathbf{u}_r)(x,t)$ for all $\mathbf{n} \in \mathbb{R}^d$, $\|\mathbf{n}\|_{\ell^2} = 1$.
- There exists an invariant set $A \subset \mathcal{A}$, i.e.,

$$\mathbf{u}(\mathbf{u}_l, \mathbf{u}_r)(x,t) \in A, \quad \forall t \geq 0, \forall x \in \mathbb{R}, \quad \forall \mathbf{u}_l, \mathbf{u}_r \in A.$$

- A is convex.

Formulation of the problem

Examples of invariant sets

- Invariant domains are convex for genuinely nonlinear systems (**Hoff (1979, 1985)**, **Chueh, Conley, Smoller (1973)**).
- Scalar conservation in $\mathbb{R}^d$: $\boxed{A = [a, b], \quad \forall a \leq b \in \mathbb{R}}$.
- Euler: $\boxed{A = \{\rho > 0,\ e > 0,\ s \geq a\}, \quad \forall a \in \mathbb{R}}$, where s is the specific entropy.
- p-system (1D): etc. $\mathbf{U} = (v, u)^{\mathrm{T}}$

$$\boxed{A := \{\mathbf{U} \in \mathbb{R}_+ \times \mathbb{R} \mid a \leq W_2(\mathbf{U}) \leq \ W_1(\mathbf{U}) \leq b\}, \quad \forall a \leq b \in \mathbb{R}}$$

where W_1 and W_2 are the Riemann invariants

$$W_1(\mathbf{U}) = u + \int_v^\infty \sqrt{-p'(s)}\,\mathrm{d}s, \quad \text{and} \quad W_2(\mathbf{U}) = u - \int_v^\infty \sqrt{-p'(s)}\,\mathrm{d}s.$$

Formulation of the problem

Examples of invariant sets

- Invariant domains are convex for genuinely nonlinear systems (**Hoff (1979, 1985)**, **Chueh, Conley, Smoller (1973)**).
- Scalar conservation in $\mathbb{R}^d$: $\boxed{A = [a, b], \quad \forall a \leq b \in \mathbb{R}}$.
- Euler: $\boxed{A = \{\rho > 0,\ e > 0,\ s \geq a\}, \quad \forall a \in \mathbb{R}}$, where s is the specific entropy.
- p-system (1D): etc. $\mathbf{U} = (v, u)^{\mathrm{T}}$

$$\boxed{A := \{\mathbf{U} \in \mathbb{R}_+ \times \mathbb{R} \mid a \leq W_2(\mathbf{U}) \leq \ W_1(\mathbf{U}) \leq b\}, \quad \forall a \leq b \in \mathbb{R}}$$

where W_1 and W_2 are the Riemann invariants

$$W_1(\mathbf{U}) = u + \int_v^\infty \sqrt{-p'(s)}\,\mathrm{d}s, \quad \text{and} \quad W_2(\mathbf{U}) = u - \int_v^\infty \sqrt{-p'(s)}\,\mathrm{d}s.$$

Formulation of the problem

Examples of invariant sets

- Invariant domains are convex for genuinely nonlinear systems (**Hoff (1979, 1985)**, **Chueh, Conley, Smoller (1973)**).
- Scalar conservation in $\mathbb{R}^d$: $\boxed{A = [a, b], \quad \forall a \leq b \in \mathbb{R}}$.
- Euler: $\boxed{A = \{\rho > 0,\ e > 0,\ s \geq a\}, \quad \forall a \in \mathbb{R}}$, where s is the specific entropy.
- p-system (1D): etc. $\mathbf{U} = (v, u)^{\mathrm{T}}$

$$\boxed{A := \{\mathbf{U} \in \mathbb{R}_+ \times \mathbb{R} \mid a \leq W_2(\mathbf{U}) \leq W_1(\mathbf{U}) \leq b\}, \quad \forall a \leq b \in \mathbb{R}}$$

where W_1 and W_2 are the Riemann invariants

$$W_1(\mathbf{U}) = u + \int_v^\infty \sqrt{-p'(s)}\,\mathrm{d}s, \quad \text{and} \quad W_2(\mathbf{U}) = u - \int_v^\infty \sqrt{-p'(s)}\,\mathrm{d}s.$$

Formulation of the problem

Examples of invariant sets

- Invariant domains are convex for genuinely nonlinear systems (**Hoff (1979, 1985)**, **Chueh, Conley, Smoller (1973)**).
- Scalar conservation in $\mathbb{R}^d$: $\boxed{A = [a, b], \quad \forall a \le b \in \mathbb{R}}$.
- Euler: $\boxed{A = \{\rho > 0,\ e > 0,\ s \ge a\}, \quad \forall a \in \mathbb{R}}$, where s is the specific entropy.
- p-system (1D): etc. $\mathbf{U} = (v, u)^{\mathrm{T}}$

$$\boxed{A := \{\mathbf{U} \in \mathbb{R}_+ \times \mathbb{R} \mid a \le W_2(\mathbf{U}) \le W_1(\mathbf{U}) \le b\}, \quad \forall a \le b \in \mathbb{R}}$$

where W_1 and W_2 are the Riemann invariants

$$W_1(\mathbf{U}) = u + \int_v^\infty \sqrt{-p'(s)}\,\mathrm{d}s, \quad \text{and} \quad W_2(\mathbf{U}) = u - \int_v^\infty \sqrt{-p'(s)}\,\mathrm{d}s.$$

FE approximation

Hyperbolic systems

Approximation (time and space)

FE space/Shape functions

- $\{\mathcal{T}_h\}_{h>0}$ shape regular conforming mesh sequence
- $\{\varphi_1, \ldots, \varphi_N\}$, positive + partition of unity
- Ex: $\mathbb{P}_1$, $\mathbb{Q}_1$, Bernstein polynomials (any degree)
- $m_i := \int_D \varphi_i \, \mathrm{d}\mathbf{x}$, lumped mass matrix

Algorithm: Galerkin + First-order viscosity + Explicit Euler

$$m_i \frac{\mathbf{U}_i^{n+1} - \mathbf{U}_i^n}{\Delta t} + \int_D \nabla \cdot \left(\sum_{j \in \mathcal{I}(S_i)} (\mathbf{f}(\mathbf{U}_j^n)) \varphi_j \right) \varphi_i \, \mathrm{d}\mathbf{x} + \sum_{j \in \mathcal{I}(S_i)} d_{ij}^n (\mathbf{U}_i^n - \mathbf{U}_j^n) = 0.$$

- How should we choose artificial viscosity d_{ij}^n?

Approximation (time and space)

FE space/Shape functions

- $\{\mathcal{T}_h\}_{h>0}$ shape regular conforming mesh sequence
- $\{\varphi_1, \ldots, \varphi_N\}$, positive + partition of unity
- Ex: $\mathbb{P}_1$, $\mathbb{Q}_1$, Bernstein polynomials (any degree)
- $m_i := \int_D \varphi_i \, \mathrm{d}\mathbf{x}$, lumped mass matrix

Algorithm: Galerkin + First-order viscosity + Explicit Euler

$$m_i \frac{\mathbf{U}_i^{n+1} - \mathbf{U}_i^n}{\Delta t} + \int_D \nabla\cdot \left(\sum_{j\in\mathcal{I}(S_i)} (\mathbf{f}(\mathbf{U}_j^n))\varphi_j \right) \varphi_i \, \mathrm{d}\mathbf{x} + \sum_{j\in\mathcal{I}(S_i)} d_{ij}^n(\mathbf{U}_i^n - \mathbf{U}_j^n) = 0.$$

- How should we choose artificial viscosity d_{ij}^n?

Approximation (time and space)

FE space/Shape functions

- $\{\mathcal{T}_h\}_{h>0}$ shape regular conforming mesh sequence
- $\{\varphi_1, \ldots, \varphi_N\}$, positive + partition of unity
- Ex: $\mathbb{P}_1$, $\mathbb{Q}_1$, Bernstein polynomials (any degree)
- $m_i := \int_D \varphi_i \, \mathrm{d}\mathbf{x}$, lumped mass matrix

Algorithm: Galerkin + First-order viscosity + Explicit Euler

$$m_i \frac{\mathbf{U}_i^{n+1} - \mathbf{U}_i^n}{\Delta t} + \int_D \nabla\cdot \left(\sum_{j \in \mathcal{I}(S_i)} (\mathbf{f}(\mathbf{U}_j^n)) \varphi_j \right) \varphi_i \, \mathrm{d}\mathbf{x} + \sum_{j \in \mathcal{I}(S_i)} d_{ij}^n (\mathbf{U}_i^n - \mathbf{U}_j^n) = 0.$$

- How should we choose artificial viscosity d_{ij}^n?

Approximation (time and space)

FE space/Shape functions

- $\{\mathcal{T}_h\}_{h>0}$ shape regular conforming mesh sequence
- $\{\varphi_1, \ldots, \varphi_N\}$, positive + partition of unity
- Ex: $\mathbb{P}_1$, $\mathbb{Q}_1$, Bernstein polynomials (any degree)
- $m_i := \int_D \varphi_i \,\mathrm{d}\mathbf{x}$, lumped mass matrix

Algorithm: Galerkin + First-order viscosity + Explicit Euler

$$m_i \frac{\mathbf{U}_i^{n+1} - \mathbf{U}_i^n}{\Delta t} + \int_D \nabla\cdot \left(\sum_{j\in\mathcal{I}(S_i)} (\mathbf{f}(\mathbf{U}_j^n))\varphi_j \right) \varphi_i \,\mathrm{d}\mathbf{x} + \sum_{j\in\mathcal{I}(S_i)} d_{ij}^n (\mathbf{U}_i^n - \mathbf{U}_j^n) = 0.$$

- How should we choose artificial viscosity d_{ij}^n?

Approximation (time and space)

FE space/Shape functions

- $\{\mathcal{T}_h\}_{h>0}$ shape regular conforming mesh sequence
- $\{\varphi_1, \ldots, \varphi_N\}$, positive + partition of unity
- Ex: $\mathbb{P}_1$, $\mathbb{Q}_1$, Bernstein polynomials (any degree)
- $m_i := \int_D \varphi_i \,\mathrm{d}\mathbf{x}$, lumped mass matrix

Algorithm: Galerkin + First-order viscosity + Explicit Euler

$$m_i \frac{\mathbf{U}_i^{n+1} - \mathbf{U}_i^n}{\Delta t} + \int_D \nabla\cdot \left(\sum_{j\in\mathcal{I}(S_i)} (\mathbf{f}(\mathbf{U}_j^n))\varphi_j \right) \varphi_i \,\mathrm{d}\mathbf{x} + \sum_{j\in\mathcal{I}(S_i)} d_{ij}^n (\mathbf{U}_i^n - \mathbf{U}_j^n) = 0.$$

- How should we choose artificial viscosity d_{ij}^n?

Approximation (time and space)

FE space/Shape functions

- $\{\mathcal{T}_h\}_{h>0}$ shape regular conforming mesh sequence
- $\{\varphi_1, \ldots, \varphi_N\}$, positive + partition of unity
- Ex: $\mathbb{P}_1$, $\mathbb{Q}_1$, Bernstein polynomials (any degree)
- $m_i := \int_D \varphi_i \, \mathrm{d}\mathbf{x}$, lumped mass matrix

Algorithm: Galerkin + First-order viscosity + Explicit Euler

$$m_i \frac{\mathbf{U}_i^{n+1} - \mathbf{U}_i^n}{\Delta t} + \int_D \nabla \cdot \left(\sum_{j \in \mathcal{I}(S_i)} (\mathbf{f}(\mathbf{U}_j^n)) \varphi_j \right) \varphi_i \, \mathrm{d}\mathbf{x} + \sum_{j \in \mathcal{I}(S_i)} d_{ij}^n (\mathbf{U}_i^n - \mathbf{U}_j^n) = 0.$$

- How should we choose artificial viscosity d_{ij}^n?

Approximation (time and space)

Algorithm: Galerkin + First-order viscosity + Explicit Euler

- Introduce

$$\mathbf{c}_{ij} = \int_D \varphi_i(\mathbf{x}) \nabla \varphi_j(\mathbf{x}) \, \mathrm{d}\mathbf{x}.$$

- Then

$$m_i \frac{\mathbf{U}_i^{n+1} - \mathbf{U}_i^n}{\Delta t} = \sum_j \left(-\mathbf{c}_{ij} \cdot \mathbf{f}(\mathbf{U}_j) + d_{ij}^n \mathbf{U}_j \right).$$

- Observe that conservation implies $\sum_j \mathbf{c}_{ij} = 0$, (partition of unity)
- We define d_{ij}^n such that $\sum_j d_{ij}^n = 0$, (conservation).

Remark

- *Rest of the talk applies to any method that can be formalized as above.*

Approximation (time and space)

Algorithm: Galerkin + First-order viscosity + Explicit Euler

- Introduce

$$\mathbf{c}_{ij} = \int_D \varphi_i(\mathbf{x}) \nabla \varphi_j(\mathbf{x})\, \mathrm{d}\mathbf{x}.$$

- Then

$$m_i \frac{\mathbf{U}_i^{n+1} - \mathbf{U}_i^n}{\Delta t} = \sum_j \left(-\mathbf{c}_{ij} \cdot \mathbf{f}(\mathbf{U}_j) + d_{ij}^n \mathbf{U}_j \right).$$

- Observe that conservation implies $\sum_j \mathbf{c}_{ij} = 0$, (partition of unity)
- We define d_{ij}^n such that $\sum_j d_{ij}^n = 0$, (conservation).

Remark

- *Rest of the talk applies to any method that can be formalized as above.*

Approximation (time and space)

Algorithm: Galerkin + First-order viscosity + Explicit Euler

- Introduce

$$\mathbf{c}_{ij} = \int_D \varphi_i(\mathbf{x}) \nabla \varphi_j(\mathbf{x}) \, \mathrm{d}\mathbf{x}.$$

- Then

$$m_i \frac{\mathbf{U}_i^{n+1} - \mathbf{U}_i^n}{\Delta t} = \sum_j \left(-\mathbf{c}_{ij} \cdot \mathbf{f}(\mathbf{U}_j) + d_{ij}^n \mathbf{U}_j \right).$$

- Observe that conservation implies $\sum_j \mathbf{c}_{ij} = 0$, (partition of unity)
- We define d_{ij}^n such that $\sum_j d_{ij}^n = 0$, (conservation).

Remark

- *Rest of the talk applies to any method that can be formalized as above.*

Approximation (time and space)

Algorithm: Galerkin + First-order viscosity + Explicit Euler

- Introduce
$$\mathbf{c}_{ij} = \int_D \varphi_i(\mathbf{x}) \nabla \varphi_j(\mathbf{x})\, \mathrm{d}\mathbf{x}.$$
- Then
$$m_i \frac{\mathbf{U}_i^{n+1} - \mathbf{U}_i^n}{\Delta t} = \sum_j \left(-\mathbf{c}_{ij} \cdot \mathbf{f}(\mathbf{U}_j) + d_{ij}^n \mathbf{U}_j \right).$$
- Observe that conservation implies $\sum_j \mathbf{c}_{ij} = 0$, (partition of unity)
- We define d_{ij}^n such that $\sum_j d_{ij}^n = 0$, (conservation).

Remark

- *Rest of the talk applies to any method that can be formalized as above.*

Approximation (time and space)

Algorithm: Galerkin + First-order viscosity + Explicit Euler

- Introduce

$$\mathbf{c}_{ij} = \int_D \varphi_i(\mathbf{x}) \nabla \varphi_j(\mathbf{x}) \, \mathrm{d}\mathbf{x}.$$

- Then

$$m_i \frac{\mathbf{U}_i^{n+1} - \mathbf{U}_i^n}{\Delta t} = \sum_j \left(-\mathbf{c}_{ij} \cdot \mathbf{f}(\mathbf{U}_j) + d_{ij}^n \mathbf{U}_j \right).$$

- Observe that conservation implies $\sum_j \mathbf{c}_{ij} = 0$, (partition of unity)
- We define d_{ii}^n such that $\sum_j d_{ij}^n = 0$, (conservation).

Remark

- *Rest of the talk applies to any method that can be formalized as above.*

Approximation (time and space)

Algorithm: Galerkin + First-order viscosity + Explicit Euler

- Observe that conservation implies $\sum_j \mathbf{c}_{ij} = 0$ and $\sum_j d_{ij}^n = 0$.

$$m_i \frac{\mathbf{U}_i^{n+1} - \mathbf{U}_i^n}{\Delta t} = \sum_j \left(\mathbf{c}_{ij} \cdot (\mathbf{f}(\mathbf{U}_i) - \mathbf{f}(\mathbf{U}_j)) + d_{ij}^n (\mathbf{U}_i + \mathbf{U}_j) \right).$$

- Try to construct convex combination . . .

$$\mathbf{U}_i^{n+1} = \mathbf{U}_i^n (1 + 2\frac{\Delta t}{m_i} D_{ii}) + \sum_{j \neq i} \frac{\Delta t}{m_i} \left(\mathbf{c}_{ij} \cdot (\mathbf{f}(\mathbf{U}_i) - \mathbf{f}(\mathbf{U}_j)) + d_{ij}^n (\mathbf{U}_i + \mathbf{U}_j) \right)$$

$$= \mathbf{U}_i^n (1 - \sum_{j \neq i} 2\frac{\Delta t}{m_i} d_{ij}^n) + \sum_{j \neq i} \frac{2\Delta t}{m_i} d_{ij}^n \left(\frac{1}{2}(\mathbf{U}_i + \mathbf{U}_j) + \frac{\mathbf{c}_{ij}}{2 d_{ij}^n} \cdot (\mathbf{f}(\mathbf{U}_i) - \mathbf{f}(\mathbf{U}_j)) \right)$$

- Introduce intermediate states $\overline{\mathbf{U}}(\mathbf{U}_i, \mathbf{U}_j)$

$$\overline{\mathbf{U}}(\mathbf{U}_i, \mathbf{U}_j) := \frac{1}{2}(\mathbf{U}_i + \mathbf{U}_j) + \frac{\mathbf{c}_{ij}}{2 d_{ij}^n} \cdot (\mathbf{f}(\mathbf{U}_i) - \mathbf{f}(\mathbf{U}_j)).$$

Approximation (time and space)

Algorithm: Galerkin + First-order viscosity + Explicit Euler

- Observe that conservation implies $\sum_j \mathbf{c}_{ij} = 0$ and $\sum_j d_{ij}^n = 0$.

$$m_i \frac{\mathbf{U}_i^{n+1} - \mathbf{U}_i^n}{\Delta t} = \sum_j \Big(\mathbf{c}_{ij}\cdot(\mathbf{f}(\mathbf{U}_i) - \mathbf{f}(\mathbf{U}_j)) + d_{ij}^n(\mathbf{U}_i + \mathbf{U}_j) \Big).$$

- Try to construct convex combination ...

$$\mathbf{U}_i^{n+1} = \mathbf{U}_i^n(1 + 2\frac{\Delta t}{m_i} D_{ii}) + \sum_{j\neq i} \frac{\Delta t}{m_i} \Big(\mathbf{c}_{ij}\cdot(\mathbf{f}(\mathbf{U}_i) - \mathbf{f}(\mathbf{U}_j)) + d_{ij}^n(\mathbf{U}_i + \mathbf{U}_j) \Big)$$

$$= \mathbf{U}_i^n(1 - \sum_{j\neq i} 2\frac{\Delta t}{m_i} d_{ij}^n) + \sum_{j\neq i} \frac{2\Delta t}{m_i} d_{ij}^n \left(\frac{1}{2}(\mathbf{U}_i + \mathbf{U}_j) + \frac{\mathbf{c}_{ij}}{2d_{ij}^n}\cdot(\mathbf{f}(\mathbf{U}_i) - \mathbf{f}(\mathbf{U}_j)) \right)$$

- Introduce intermediate states $\overline{\mathbf{U}}(\mathbf{U}_i, \mathbf{U}_j)$

$$\overline{\mathbf{U}}(\mathbf{U}_i, \mathbf{U}_j) := \frac{1}{2}(\mathbf{U}_i + \mathbf{U}_j) + \frac{\mathbf{c}_{ij}}{2d_{ij}^n}\cdot(\mathbf{f}(\mathbf{U}_i) - \mathbf{f}(\mathbf{U}_j)).$$

Approximation (time and space)

Algorithm: Galerkin + First-order viscosity + Explicit Euler

- Observe that conservation implies $\sum_j \mathbf{c}_{ij} = 0$ and $\sum_j d_{ij}^n = 0$.

$$m_i \frac{\mathbf{U}_i^{n+1} - \mathbf{U}_i^n}{\Delta t} = \sum_j \left(\mathbf{c}_{ij} \cdot (\mathbf{f}(\mathbf{U}_i) - \mathbf{f}(\mathbf{U}_j)) + d_{ij}^n (\mathbf{U}_i + \mathbf{U}_j) \right).$$

- Try to construct convex combination ...

$$\mathbf{U}_i^{n+1} = \mathbf{U}_i^n (1 + 2\frac{\Delta t}{m_i} D_{ii}) + \sum_{j \neq i} \frac{\Delta t}{m_i} \left(\mathbf{c}_{ij} \cdot (\mathbf{f}(\mathbf{U}_i) - \mathbf{f}(\mathbf{U}_j)) + d_{ij}^n (\mathbf{U}_i + \mathbf{U}_j) \right)$$

$$= \mathbf{U}_i^n (1 - \sum_{j \neq i} 2\frac{\Delta t}{m_i} d_{ij}^n) + \sum_{j \neq i} \frac{2\Delta t}{m_i} d_{ij}^n \left(\frac{1}{2}(\mathbf{U}_i + \mathbf{U}_j) + \frac{\mathbf{c}_{ij}}{2 d_{ij}^n} \cdot (\mathbf{f}(\mathbf{U}_i) - \mathbf{f}(\mathbf{U}_j)) \right)$$

- Introduce intermediate states $\overline{\mathbf{U}}(\mathbf{U}_i, \mathbf{U}_j)$

$$\overline{\mathbf{U}}(\mathbf{U}_i, \mathbf{U}_j) := \frac{1}{2}(\mathbf{U}_i + \mathbf{U}_j) + \frac{\mathbf{c}_{ij}}{2 d_{ij}^n} \cdot (\mathbf{f}(\mathbf{U}_i) - \mathbf{f}(\mathbf{U}_j)).$$

Approximation (time and space)

Algorithm: Galerkin + First-order viscosity + Explicit Euler

- Observe that conservation implies $\sum_j \mathbf{c}_{ij} = 0$ and $\sum_j d_{ij}^n = 0$.

$$m_i \frac{\mathbf{U}_i^{n+1} - \mathbf{U}_i^n}{\Delta t} = \sum_j \left(\mathbf{c}_{ij} \cdot (\mathbf{f}(\mathbf{U}_i) - \mathbf{f}(\mathbf{U}_j)) + d_{ij}^n (\mathbf{U}_i + \mathbf{U}_j) \right).$$

- Try to construct convex combination . . .

$$\mathbf{U}_i^{n+1} = \mathbf{U}_i^n (1 + 2\frac{\Delta t}{m_i} D_{ii}) + \sum_{j \neq i} \frac{\Delta t}{m_i} \left(\mathbf{c}_{ij} \cdot (\mathbf{f}(\mathbf{U}_i) - \mathbf{f}(\mathbf{U}_j)) + d_{ij}^n (\mathbf{U}_i + \mathbf{U}_j) \right)$$

$$= \mathbf{U}_i^n (1 - \sum_{j \neq i} 2\frac{\Delta t}{m_i} d_{ij}^n) + \sum_{j \neq i} \frac{2\Delta t}{m_i} d_{ij}^n \left(\frac{1}{2}(\mathbf{U}_i + \mathbf{U}_j) + \frac{\mathbf{c}_{ij}}{2 d_{ij}^n} \cdot (\mathbf{f}(\mathbf{U}_i) - \mathbf{f}(\mathbf{U}_j)) \right)$$

- Introduce intermediate states $\overline{\mathbf{U}}(\mathbf{U}_i, \mathbf{U}_j)$

$$\overline{\mathbf{U}}(\mathbf{U}_i, \mathbf{U}_j) := \frac{1}{2}(\mathbf{U}_i + \mathbf{U}_j) + \frac{\mathbf{c}_{ij}}{2 d_{ij}^n} \cdot (\mathbf{f}(\mathbf{U}_i) - \mathbf{f}(\mathbf{U}_j)).$$

Approximation (time and space)

Algorithm: Galerkin + First-order viscosity + Explicit Euler

- Now construct convex combination

$$\mathbf{U}_i^{n+1} = \mathbf{U}_i^n(1 - \sum_{j\neq i} 2\frac{\Delta t}{m_i} d_{ij}^n) + \sum_{j\neq i} \frac{2\Delta t}{m_i} d_{ij}^n \overline{\mathbf{U}}(\mathbf{U}_i, \mathbf{U}_j)$$

- Are the states $\overline{\mathbf{U}}(\mathbf{U}_i, \mathbf{U}_j)$ good objects?

Approximation (time and space)

Algorithm: Galerkin + First-order viscosity + Explicit Euler

- Now construct convex combination

$$\mathbf{U}_i^{n+1} = \mathbf{U}_i^n \Big(1 - \sum_{j\neq i} 2\frac{\Delta t}{m_i} d_{ij}^n\Big) + \sum_{j\neq i} \frac{2\Delta t}{m_i} d_{ij}^n \overline{\mathbf{U}}(\mathbf{U}_i, \mathbf{U}_j)$$

- Are the states $\overline{\mathbf{U}}(\mathbf{U}_i, \mathbf{U}_j)$ good objects?

Approximation (time and space)

Algorithm: Galerkin + First-order viscosity + Explicit Euler

- Define $\mathbf{n}_{ij} = \mathbf{c}_{ij}/\|\mathbf{c}_{ij}\|_{\ell^2} \in \mathbb{R}^d$, (unit vector).
- $\mathbf{f}_{ij}(\mathbf{U}) := \mathbf{n}_{ij}\cdot\mathbf{f}(\mathbf{U})$ is an hyperbolic flux by definition of hyperbolicity!
- Then

$$\overline{\mathbf{U}}(\mathbf{U}_i, \mathbf{U}_j) := \frac{1}{2}(\mathbf{U}_i + \mathbf{U}_j) + \frac{\|\mathbf{c}_{ij}\|_{\ell^2}}{2d_{ij}^n}(\mathbf{f}_{ij}(\mathbf{U}_i) - \mathbf{f}_{ij}(\mathbf{U}_j)).$$

Approximation (time and space)

Algorithm: Galerkin + First-order viscosity + Explicit Euler

- Define $\mathbf{n}_{ij} = \mathbf{c}_{ij}/\|\mathbf{c}_{ij}\|_{\ell^2} \in \mathbb{R}^d$, (unit vector).
- $\mathbf{f}_{ij}(\mathbf{U}) := \mathbf{n}_{ij} \cdot \mathbf{f}(\mathbf{U})$ is an hyperbolic flux by definition of hyperbolicity!
- Then

$$\overline{\mathbf{U}}(\mathbf{U}_i, \mathbf{U}_j) := \frac{1}{2}(\mathbf{U}_i + \mathbf{U}_j) + \frac{\|\mathbf{c}_{ij}\|_{\ell^2}}{2d_{ij}^n}(\mathbf{f}_{ij}(\mathbf{U}_i) - \mathbf{f}_{ij}(\mathbf{U}_j)).$$

Approximation (time and space)

Algorithm: Galerkin + First-order viscosity + Explicit Euler

- Define $\mathbf{n}_{ij} = \mathbf{c}_{ij}/\|\mathbf{c}_{ij}\|_{\ell^2} \in \mathbb{R}^d$, (unit vector).
- $\mathbf{f}_{ij}(\mathbf{U}) := \mathbf{n}_{ij}\cdot\mathbf{f}(\mathbf{U})$ is an hyperbolic flux by definition of hyperbolicity!
- Then

$$\overline{\mathbf{U}}(\mathbf{U}_i, \mathbf{U}_j) := \frac{1}{2}(\mathbf{U}_i + \mathbf{U}_j) + \frac{\|\mathbf{c}_{ij}\|_{\ell^2}}{2d_{ij}^n}(\mathbf{f}_{ij}(\mathbf{U}_i) - \mathbf{f}_{ij}(\mathbf{U}_j)).$$

Approximation (time and space)

Lemma (GP (2015))

- *Consider the fake 1D Riemann problem!*

$$\partial_t \mathbf{v} + \partial_x(\mathbf{n}_{ij}\cdot\mathbf{f}(\mathbf{v})) = 0, \quad \mathbf{v}(x,0) = \begin{cases} \mathbf{U}_i & \text{if } x < 0 \\ \mathbf{U}_j & \text{if } x > 0. \end{cases}$$

- *Let* $\lambda_{\max}(\mathbf{f}, \mathbf{n}_{ij}, \mathbf{U}_i, \mathbf{U}_j)$ *be maximum wave speed in 1D Riemann problem*
- *Then* $\overline{\mathbf{U}}(\mathbf{U}_i, \mathbf{U}_j) = \int_{-\frac{1}{2}}^{\frac{1}{2}} \mathbf{v}(x,t)\,\mathrm{d}x$ *with fake time* $t = \frac{\|\mathbf{c}_{ij}\|_{\ell^2}}{2d_{ij}^n}$, *provided*

$$\frac{\|\mathbf{c}_{ij}\|_{\ell^2}}{2d_{ij}^n}\lambda_{\max}(\mathbf{f}, \mathbf{n}_{ij}, \mathbf{U}_i, \mathbf{U}_j) = \boxed{t\lambda_{\max}(\mathbf{f}, \mathbf{n}_{ij}, \mathbf{U}_i, \mathbf{U}_j) \leq \frac{1}{2}}$$

- Define viscosity coefficient

$$\boxed{d_{ij}^n := \lambda_{\max}(\mathbf{f}, \mathbf{n}_{ij}, \mathbf{U}_i, \mathbf{U}_j)\|\mathbf{c}_{ij}\|_{\ell^2}, \quad j \neq i.}$$

Approximation (time and space)

Lemma (GP (2015))

- *Consider the fake 1D Riemann problem!*

$$\partial_t \mathbf{v} + \partial_x(\mathbf{n}_{ij}\cdot\mathbf{f}(\mathbf{v})) = 0, \quad \mathbf{v}(x,0) = \begin{cases} \mathbf{U}_i & \text{if } x < 0 \\ \mathbf{U}_j & \text{if } x > 0. \end{cases}$$

- *Let* $\lambda_{\max}(\mathbf{f}, \mathbf{n}_{ij}, \mathbf{U}_i, \mathbf{U}_j)$ *be maximum wave speed in 1D Riemann problem*
- *Then* $\overline{\mathbf{U}}(\mathbf{U}_i, \mathbf{U}_j) = \int_{-\frac{1}{2}}^{\frac{1}{2}} \mathbf{v}(x,t)\,\mathrm{d}x$ *with fake time* $t = \frac{\|\mathbf{c}_{ij}\|_{\ell^2}}{2d_{ij}^n}$, *provided*

$$\frac{\|\mathbf{c}_{ij}\|_{\ell^2}}{2d_{ij}^n}\lambda_{\max}(\mathbf{f}, \mathbf{n}_{ij}, \mathbf{U}_i, \mathbf{U}_j) = \boxed{t\lambda_{\max}(\mathbf{f}, \mathbf{n}_{ij}, \mathbf{U}_i, \mathbf{U}_j) \leq \frac{1}{2}}$$

- Define viscosity coefficient

$$\boxed{d_{ij}^n := \lambda_{\max}(\mathbf{f}, \mathbf{n}_{ij}, \mathbf{U}_i, \mathbf{U}_j)\|\mathbf{c}_{ij}\|_{\ell^2}, \quad j \neq i.}$$

Approximation (time and space)

Lemma (GP (2015))

- *Consider the fake 1D Riemann problem!*

$$\partial_t \mathbf{v} + \partial_x(\mathbf{n}_{ij}\cdot\mathbf{f}(\mathbf{v})) = 0, \quad \mathbf{v}(x,0) = \begin{cases} \mathbf{U}_i & \text{if } x < 0 \\ \mathbf{U}_j & \text{if } x > 0. \end{cases}$$

- *Let* $\lambda_{\max}(\mathbf{f}, \mathbf{n}_{ij}, \mathbf{U}_i, \mathbf{U}_j)$ *be maximum wave speed in 1D Riemann problem*
- *Then* $\overline{\mathbf{U}}(\mathbf{U}_i, \mathbf{U}_j) = \int_{-\frac{1}{2}}^{\frac{1}{2}} \mathbf{v}(x,t)\,\mathrm{d}x$ *with fake time* $t = \frac{\|\mathbf{c}_{ij}\|_{\ell^2}}{2d_{ij}^n}$, *provided*

$$\frac{\|\mathbf{c}_{ij}\|_{\ell^2}}{2d_{ij}^n}\lambda_{\max}(\mathbf{f}, \mathbf{n}_{ij}, \mathbf{U}_i, \mathbf{U}_j) = \boxed{t\lambda_{\max}(\mathbf{f}, \mathbf{n}_{ij}, \mathbf{U}_i, \mathbf{U}_j) \le \frac{1}{2}}$$

- Define viscosity coefficient

$$\boxed{d_{ij}^n := \lambda_{\max}(\mathbf{f}, \mathbf{n}_{ij}, \mathbf{U}_i, \mathbf{U}_j)\|\mathbf{c}_{ij}\|_{\ell^2}, \quad j \ne i.}$$

Approximation (time and space)

Lemma (GP (2015))

- *Consider the fake 1D Riemann problem!*

$$\partial_t \mathbf{v} + \partial_x(\mathbf{n}_{ij} \cdot \mathbf{f}(\mathbf{v})) = 0, \quad \mathbf{v}(x, 0) = \begin{cases} \mathbf{U}_i & \text{if } x < 0 \\ \mathbf{U}_j & \text{if } x > 0. \end{cases}$$

- *Let* $\lambda_{\max}(\mathbf{f}, \mathbf{n}_{ij}, \mathbf{U}_i, \mathbf{U}_j)$ *be maximum wave speed in 1D Riemann problem*
- *Then* $\overline{\mathbf{U}}(\mathbf{U}_i, \mathbf{U}_j) = \int_{-\frac{1}{2}}^{\frac{1}{2}} \mathbf{v}(x, t)\,\mathrm{d}x$ *with fake time* $t = \frac{\|\mathbf{c}_{ij}\|_{\ell^2}}{2d_{ij}^n}$*, provided*

$$\frac{\|\mathbf{c}_{ij}\|_{\ell^2}}{2d_{ij}^n} \lambda_{\max}(\mathbf{f}, \mathbf{n}_{ij}, \mathbf{U}_i, \mathbf{U}_j) = \boxed{t\lambda_{\max}(\mathbf{f}, \mathbf{n}_{ij}, \mathbf{U}_i, \mathbf{U}_j) \leq \frac{1}{2}}$$

- Define viscosity coefficient

$$\boxed{d_{ij}^n := \lambda_{\max}(\mathbf{f}, \mathbf{n}_{ij}, \mathbf{U}_i, \mathbf{U}_j)\|\mathbf{c}_{ij}\|_{\ell^2}, \quad j \neq i.}$$

Approximation (time and space)

Theorem (GP (2015))

Provided CFL condition, $(1 - 2\frac{\Delta t}{m_i}|D_{ii}|) \geq 0$.

- *Local invariance:* $\mathbf{U}_i^{n+1} \in Conv\{\overline{\mathbf{U}}(\mathbf{U}_i^n, \mathbf{U}_j^n) \mid j \in \mathcal{I}(S_i)\}$.
- *Global invariance: The scheme preserves all the convex invariant sets.* *(Let A be a convex invariant set, assume $\mathbf{U}_0 \in A$, then $\mathbf{U}_i^{n+1} \in A$ for all $n \geq 0$.)*
- *Discrete entropy inequality for all the entropy pairs (η, q):*

$$\frac{m_i}{\Delta t}(\eta(\mathbf{U}_i^{n+1}) - \eta(\mathbf{U}_i^n)) + \int_D \nabla\cdot(\Pi_h \mathbf{q}(\mathbf{u}_h^n))\varphi_i \,\mathrm{d}x + \sum_{i\neq j\in\mathcal{I}(S_i)} d_{ij}\eta(\mathbf{U}_j^n) \leq 0.$$

Approximation (time and space)

Theorem (GP (2015))

Provided CFL condition, $(1 - 2\frac{\Delta t}{m_i}|D_{ii}|) \geq 0$.

- *Local invariance:* $\mathbf{U}_i^{n+1} \in Conv\{\overline{\mathbf{U}}(\mathbf{U}_i^n, \mathbf{U}_j^n) \mid j \in \mathcal{I}(S_i)\}$.
- *Global invariance: The scheme preserves all the convex invariant sets.*
  (Let A be a convex invariant set, assume $\mathbf{U}_0 \in A$, then $\mathbf{U}_i^{n+1} \in A$ for all $n \geq 0$.)
- *Discrete entropy inequality for all the entropy pairs* (η, q):

$$\frac{m_i}{\Delta t}(\eta(\mathbf{U}_i^{n+1}) - \eta(\mathbf{U}_i^n)) + \int_D \nabla\cdot(\Pi_h \mathbf{q}(\mathbf{u}_h^n))\varphi_i \,\mathrm{d}x + \sum_{i\neq j\in\mathcal{I}(S_i)} d_{ij}\eta(\mathbf{U}_j^n) \leq 0.$$

Approximation (time and space)

Theorem (GP (2015))

Provided CFL condition, $(1 - 2\frac{\Delta t}{m_i}|D_{ii}|) \geq 0$.

- *Local invariance:* $\mathbf{U}_i^{n+1} \in Conv\{\overline{\mathbf{U}}(\mathbf{U}_i^n, \mathbf{U}_j^n) \mid j \in \mathcal{I}(S_i)\}$.
- *Global invariance: The scheme preserves all the convex invariant sets.* *(Let A be a convex invariant set, assume $\mathbf{U}_0 \in A$, then $\mathbf{U}_i^{n+1} \in A$ for all $n \geq 0$.)*
- *Discrete entropy inequality for all the entropy pairs (η, q):*

$$\frac{m_i}{\Delta t}(\eta(\mathbf{U}_i^{n+1}) - \eta(\mathbf{U}_i^n)) + \int_D \nabla\cdot(\Pi_h \mathbf{q}(\mathbf{u}_h^n))\varphi_i \,\mathrm{d}x + \sum_{i\neq j\in\mathcal{I}(S_i)} d_{ij}\eta(\mathbf{U}_j^n) \leq 0.$$

Approximation (time and space)

Theorem (GP (2015))

Provided CFL condition, $(1 - 2\frac{\Delta t}{m_i}|D_{ii}|) \geq 0$.

- *Local invariance:* $\mathbf{U}_i^{n+1} \in Conv\{\overline{\mathbf{U}}(\mathbf{U}_i^n, \mathbf{U}_j^n) \mid j \in \mathcal{I}(S_i)\}$.
- *Global invariance: The scheme preserves all the convex invariant sets.*
  (Let A be a convex invariant set, assume $\mathbf{U}_0 \in A$*, then* $\mathbf{U}_i^{n+1} \in A$ *for all* $n \geq 0$*.)*
- *Discrete entropy inequality for all the entropy pairs* (η, q)*:*

$$\frac{m_i}{\Delta t}(\eta(\mathbf{U}_i^{n+1}) - \eta(\mathbf{U}_i^n)) + \int_D \nabla\cdot(\Pi_h \mathbf{q}(\mathbf{u}_h^n))\varphi_i \,\mathrm{d}x + \sum_{i\neq j\in\mathcal{I}(S_i)} d_{ij}\eta(\mathbf{U}_j^n) \leq 0.$$

Approximation (time and space)

Theorem (GP (2015))

Provided CFL condition, $(1 - 2\frac{\Delta t}{m_i}|D_{ii}|) \geq 0$.

- *Local invariance:* $\mathbf{U}_i^{n+1} \in Conv\{\overline{\mathbf{U}}(\mathbf{U}_i^n, \mathbf{U}_j^n) \mid j \in \mathcal{I}(S_i)\}$.
- *Global invariance: The scheme preserves all the convex invariant sets.* *(Let A be a convex invariant set, assume $\mathbf{U}_0 \in A$, then $\mathbf{U}_i^{n+1} \in A$ for all $n \geq 0$.)*
- *Discrete entropy inequality for all the entropy pairs* (η, q):

$$\frac{m_i}{\Delta t}(\eta(\mathbf{U}_i^{n+1}) - \eta(\mathbf{U}_i^n)) + \int_D \nabla\cdot(\Pi_h \mathbf{q}(\mathbf{u}_h^n))\varphi_i \,\mathrm{d}x + \sum_{i \neq j \in \mathcal{I}(S_i)} d_{ij}\eta(\mathbf{U}_j^n) \leq 0.$$

Approximation (time and space)

Is it new?

- Loose extension of non-staggered Lax-Friedrichs to FE.
- Similar results proved by **Hoff (1979, 1985)**, **Perthame-Shu (1996)**, **Frid (2001)** in FV context and compressible Euler.
- Not aware of similar results for arbitrary hyperbolic systems and continuous FE.

Approximation (time and space)

Is it new?

- Loose extension of non-staggered Lax-Friedrichs to FE.
- Similar results proved by **Hoff (1979, 1985)**, **Perthame-Shu (1996)**, **Frid (2001)** in FV context and compressible Euler.
- Not aware of similar results for arbitrary hyperbolic systems and continuous FE.

Approximation (time and space)

Is it new?

- Loose extension of non-staggered Lax-Friedrichs to FE.
- Similar results proved by **Hoff (1979, 1985)**, **Perthame-Shu (1996)**, **Frid (2001)** in FV context and compressible Euler.
- Not aware of similar results for arbitrary hyperbolic systems and continuous FE.

A priori error estimate for scalar equations: Definition of mollifiers

- Let $\delta > 0$ and $\epsilon = \|\mathbf{f}\|_{\mathrm{Lip}}\,\delta$
- Consider mollifiers ω_δ and $\boldsymbol{\omega}_\epsilon$

$$\omega_\delta(t) := \begin{cases} \frac{1}{3\delta} & |t| \le \delta, \\ \frac{2\delta - |t|}{3\delta^2} & \delta \le |t| \le 2\delta, \\ 0 & \text{otherwise}, \end{cases} \qquad \boldsymbol{\omega}_\epsilon(\mathbf{x}) := \Pi_{l=1}^{d}\omega_\epsilon(x_l), \quad \mathbf{x} := (x_1, \ldots, x_d).$$

- Following Kruskov (1970), define

$$\phi(\mathbf{x}, \mathbf{y}, t, s) := \boldsymbol{\omega}_\epsilon(\mathbf{x} - \mathbf{y})\omega_\delta(t - s), \qquad \forall(\mathbf{y}, s) \in D\times[0, T].$$

- Following Cockburn Gremaud (1996,1998), define

$$\Gamma_\delta(t) := \int_0^t \omega_\delta(s)\,\mathrm{d}s.$$

A priori error estimate for scalar equations: Definition of mollifiers

- Let $\delta > 0$ and $\epsilon = \|\mathbf{f}\|_{\text{Lip}}\,\delta$
- Consider mollifiers ω_δ and $\boldsymbol{\omega}_\epsilon$

$$\omega_\delta(t) := \begin{cases} \frac{1}{3\delta} & |t| \le \delta, \\ \frac{2\delta - |t|}{3\delta^2} & \delta \le |t| \le 2\delta, \\ 0 & \text{otherwise,} \end{cases} \qquad \boldsymbol{\omega}_\epsilon(\mathbf{x}) := \Pi_{l=1}^{d} \omega_\epsilon(x_l), \quad \mathbf{x} := (x_1, \ldots, x_d).$$

- Following Kruskov (1970), define

$$\phi(\mathbf{x}, \mathbf{y}, t, s) := \boldsymbol{\omega}_\epsilon(\mathbf{x} - \mathbf{y})\omega_\delta(t - s), \qquad \forall (\mathbf{y}, s) \in D \times [0, T].$$

- Following Cockburn Gremaud (1996,1998), define

$$\Gamma_\delta(t) := \int_0^t \omega_\delta(s)\,\mathrm{d}s.$$

A priori error estimate for scalar equations: Definition of mollifiers

- Let $\delta > 0$ and $\epsilon = \|\mathbf{f}\|_{\text{Lip}}\,\delta$
- Consider mollifiers ω_δ and $\boldsymbol{\omega}_\epsilon$

$$\omega_\delta(t) := \begin{cases} \frac{1}{3\delta} & |t| \le \delta, \\ \frac{2\delta - |t|}{3\delta^2} & \delta \le |t| \le 2\delta, \\ 0 & \text{otherwise,} \end{cases} \qquad \boldsymbol{\omega}_\epsilon(\mathbf{x}) := \Pi_{l=1}^{d} \omega_\epsilon(x_l), \quad \mathbf{x} := (x_1, \ldots, x_d).$$

- Following **Kruskov (1970)**, define

$$\phi(\mathbf{x}, \mathbf{y}, t, s) := \boldsymbol{\omega}_\epsilon(\mathbf{x} - \mathbf{y})\omega_\delta(t - s), \qquad \forall(\mathbf{y}, s) \in D \times [0, T].$$

- Following **Cockburn Gremaud (1996,1998)**, define

$$\Gamma_\delta(t) := \int_0^t \omega_\delta(s)\,\mathrm{d}s.$$

A priori error estimate for scalar equations: Definition of mollifiers

- Let $\delta > 0$ and $\epsilon = \|\mathbf{f}\|_{\text{Lip}}\,\delta$
- Consider mollifiers ω_δ and $\boldsymbol{\omega}_\epsilon$

$$\omega_\delta(t) := \begin{cases} \frac{1}{3\delta} & |t| \le \delta, \\ \frac{2\delta - |t|}{3\delta^2} & \delta \le |t| \le 2\delta, \\ 0 & \text{otherwise,} \end{cases} \qquad \boldsymbol{\omega}_\epsilon(\mathbf{x}) := \Pi_{l=1}^{d} \omega_\epsilon(x_l), \quad \mathbf{x} := (x_1, \ldots, x_d).$$

- Following **Kruskov (1970)**, define

$$\phi(\mathbf{x}, \mathbf{y}, t, s) := \boldsymbol{\omega}_\epsilon(\mathbf{x} - \mathbf{y})\omega_\delta(t - s), \qquad \forall(\mathbf{y}, s) \in D \times [0, T].$$

- Following **Cockburn Gremaud (1996,1998)**, define

$$\Gamma_\delta(t) := \int_0^t \omega_\delta(s)\,\mathrm{d}s.$$

A priori error estimate for scalar equations: A useful lemma

Lemma (Guermond, Popov (2014-15))

Assume $u_0 \in BV(\Omega)$. Let $\widetilde{u}_h : D\times[0,T] \longrightarrow \mathbb{R}$ be any approximate solution. Assume that there is Λ a bounded functional on Lipschitz functions so that $\forall k \in [u_{\min}, u_{\max}]$, $\forall \psi \in W^{1,\infty}_c(D\times[0,T];\mathbb{R}^+)$:

$$-\int_0^T\int_D \left(|\widetilde{u}_h - k|\partial_t\psi + sgn(\widetilde{u}_h - k)(\mathbf{f}(\widetilde{u}_h) - \mathbf{f}(k))\cdot\nabla\psi\right) \mathrm{d}\mathbf{x}\,\mathrm{d}t$$
$$+ \|\pi_h((\widetilde{u}_h(T) - k)\bar{\pi}_h\psi(\cdot,\mathcal{T}_h))\|_{\ell^1_h} - \|\pi_h((\widetilde{u}_h(0) - k)\bar{\pi}_h\psi(\cdot,\sigma_h))\|_{\ell^1_h} \le \Lambda(\psi),$$

where $\|\cdot\|_{\ell^1_h}$ is the discrete L^1-norm and $|T - \mathcal{T}_h| \le \gamma\Delta t$, $|0 - \sigma_h| \le \gamma\Delta t$, $\gamma > 0$ is a uniform constant. Then the following estimate holds

$$\|u(\cdot,T) - \tilde{u}_h(\cdot,T)\|_{L^1(\Omega)} \le c\left((\epsilon + h)|u_0|_{BV(\Omega)} + \Lambda^*\right)$$

where $\Lambda^* := \sup_{0\le t\le T} \frac{\int_0^t\int_D \Lambda(\phi)\,\mathrm{d}y\,\mathrm{d}s}{\Gamma_\delta(t)}$.

- Generalization of results by **Cockburn Gremaud (1996)** and **Bouchut, Perthame (1998)** based on **Kruskov (1970)**, **Kuznecov (1976)**.

A priori error estimate for scalar equations: A useful lemma

Lemma (Guermond, Popov (2014-15))

Assume $u_0 \in BV(\Omega)$. Let $\tilde{u}_h : D\times[0,T] \longrightarrow \mathbb{R}$ be any approximate solution. Assume that there is Λ a bounded functional on Lipschitz functions so that $\forall k \in [u_{\min}, u_{\max}]$, $\forall \psi \in W^{1,\infty}_c(D\times[0,T];\mathbb{R}^+)$:

$$-\int_0^T\int_D \left(|\tilde{u}_h - k|\partial_t\psi + sgn(\tilde{u}_h - k)(\mathbf{f}(\tilde{u}_h) - \mathbf{f}(k))\cdot\nabla\psi\right) \mathrm{d}\mathbf{x}\,\mathrm{d}t$$
$$+ \|\pi_h((\tilde{u}_h(T) - k)\bar{\pi}_h\psi(\cdot,\mathcal{T}_h))\|_{\ell^1_h} - \|\pi_h((\tilde{u}_h(0) - k)\bar{\pi}_h\psi(\cdot,\sigma_h))\|_{\ell^1_h} \leq \Lambda(\psi),$$

where $\|\cdot\|_{\ell^1_h}$ is the discrete L^1-norm and $|T - \mathcal{T}_h| \leq \gamma\Delta t$, $|0 - \sigma_h| \leq \gamma\Delta t$, $\gamma > 0$ is a uniform constant. Then the following estimate holds

$$\boxed{\|u(\cdot,T) - \tilde{u}_h(\cdot,T)\|_{L^1(\Omega)} \leq c\left((\epsilon + h)|u_0|_{BV(\Omega)} + \Lambda^*\right)}$$

where $\Lambda^* := \sup_{0\leq t\leq T} \dfrac{\int_0^t\int_D \Lambda(\phi)\,\mathrm{d}\mathbf{y}\,\mathrm{d}s}{\Gamma_\delta(t)}$.

- Generalization of results by Cockburn Gremaud (1996) and Bouchut, Perthame (1998) based on Kruskov (1970), Kuznecov (1976).

A priori error estimate for scalar equations: A useful lemma

Lemma (Guermond, Popov (2014-15))

Assume $u_0 \in BV(\Omega)$. Let $\widetilde{u}_h : D\times[0,T] \longrightarrow \mathbb{R}$ be any approximate solution. Assume that there is Λ a bounded functional on Lipschitz functions so that $\forall k \in [u_{\min}, u_{\max}]$, $\forall \psi \in W^{1,\infty}_c(D\times[0,T];\mathbb{R}^+)$:

$$
\begin{aligned}
&-\int_0^T\int_D \left(|\widetilde{u}_h - k|\partial_t\psi + sgn(\widetilde{u}_h - k)(\mathbf{f}(\widetilde{u}_h) - \mathbf{f}(k))\cdot\nabla\psi\right) \mathrm{d}\mathbf{x}\,\mathrm{d}t \\
&\qquad + \|\pi_h((\widetilde{u}_h(T) - k)\bar{\pi}_h\psi(\cdot,\mathcal{T}_h))\|_{\ell^1_h} - \|\pi_h((\widetilde{u}_h(0) - k)\bar{\pi}_h\psi(\cdot,\sigma_h))\|_{\ell^1_h} \le \Lambda(\psi),
\end{aligned}
$$

where $\|\cdot\|_{\ell^1_h}$ is the discrete L^1-norm and $|T - \mathcal{T}_h| \le \gamma\Delta t$, $|0-\sigma_h| \le \gamma\Delta t$, $\gamma > 0$ is a uniform constant. Then the following estimate holds

$$
\boxed{\|u(\cdot,T) - \tilde{u}_h(\cdot,T)\|_{L^1(\Omega)} \le c\left((\epsilon + h)|u_0|_{BV(\Omega)} + \Lambda^*\right)}
$$

where $\Lambda^* := \sup_{0\le t\le T} \dfrac{\int_0^t\int_D \Lambda(\phi)\,\mathrm{d}\mathbf{y}\,\mathrm{d}s}{\Gamma_\delta(t)}$.

- Generalization of results by **Cockburn Gremaud (1996)** and **Bouchut, Perthame (1998)** based on **Kruskov (1970)**, **Kuznecov (1976)**.

A priori error estimate for scalar equations

English translation

Control on all the Kruskov entropies $\Rightarrow$ Convergence estimate.

Theorem (Guermond, Popov (2014-15))

Assume $u_0 \in BV$ and $\mathbf{f}$ Lipschitz. Let u_h be the first-order viscosity solution. Then there is c_0, uniform, such that the following holds if CFL$\leq c_0$:

(i) $\|u(T) - u_h(T)\|_{L^\infty((0,T);L^1)} \leq ch^{\frac{1}{2}}$ if a priori BV estimate on u_h.

(ii) $\|u(T) - u_h(T)\|_{L^\infty((0,T);L^1)} \leq ch^{\frac{1}{4}}$ otherwise.

- BV estimate is trivial in 1D (Harten's lemma).
- BV estimate can be proved in nD on special meshes.
- Similar results for FV Chainais-Hillairet (1999), Eymard et al (1998)
- First error estimates for explicit continuous FE method (as far as we know).

A priori error estimate for scalar equations

English translation

Control on all the Kruskov entropies ⇒ Convergence estimate.

Theorem (Guermond, Popov (2014-15))

Assume $u_0 \in BV$ and $\mathbf{f}$ Lipschitz. Let u_h be the first-order viscosity solution. Then there is c_0, uniform, such that the following holds if CFL$\leq c_0$:

(i) $\|u(T) - u_h(T)\|_{L^\infty((0,T);L^1)} \leq ch^{\frac{1}{2}}$ if a priori BV estimate on u_h.

(ii) $\|u(T) - u_h(T)\|_{L^\infty((0,T);L^1)} \leq ch^{\frac{1}{4}}$ otherwise.

- BV estimate is trivial in 1D (Harten's lemma).
- BV estimate can be proved in nD on special meshes.
- Similar results for FV Chainais-Hillairet (1999), Eymard et al (1998)
- First error estimates for explicit continuous FE method (as far as we know).

A priori error estimate for scalar equations

English translation

Control on all the Kruskov entropies $\Rightarrow$ Convergence estimate.

Theorem (Guermond, Popov (2014-15))

Assume $u_0 \in BV$ and $\mathbf{f}$ Lipschitz. Let u_h be the first-order viscosity solution. Then there is c_0, uniform, such that the following holds if CFL$\le c_0$:
(i) $\|u(T) - u_h(T)\|_{L^\infty((0,T);L^1)} \le ch^{\frac{1}{2}}$ if a priori BV estimate on u_h.
(ii) $\|u(T) - u_h(T)\|_{L^\infty((0,T);L^1)} \le ch^{\frac{1}{4}}$ otherwise.

- BV estimate is trivial in 1D (Harten's lemma).
- BV estimate can be proved in nD on special meshes.
- Similar results for FV Chainais-Hillairet (1999), Eymard et al (1998)
- First error estimates for explicit continuous FE method (as far as we know).

A priori error estimate for scalar equations

English translation

Control on all the Kruskov entropies $\Rightarrow$ Convergence estimate.

Theorem (Guermond, Popov (2014-15))

Assume $u_0 \in BV$ and $\mathbf{f}$ Lipschitz. Let u_h be the first-order viscosity solution. Then there is c_0, uniform, such that the following holds if CFL$\leq c_0$:

(i) $\|u(T) - u_h(T)\|_{L^\infty((0,T);L^1)} \leq ch^{\frac{1}{2}}$ if a priori BV estimate on u_h.

(ii) $\|u(T) - u_h(T)\|_{L^\infty((0,T);L^1)} \leq ch^{\frac{1}{4}}$ otherwise.

- BV estimate is trivial in 1D (Harten's lemma).
- BV estimate can be proved in nD on special meshes.
- Similar results for FV **Chainais-Hillairet (1999), Eymard et al (1998)**
- First error estimates for explicit continuous FE method (as far as we know).

A priori error estimate for scalar equations

English translation

Control on all the Kruskov entropies $\Rightarrow$ Convergence estimate.

Theorem (Guermond, Popov (2014-15))

Assume $u_0 \in BV$ and $\mathbf{f}$ Lipschitz. Let u_h be the first-order viscosity solution. Then there is c_0, uniform, such that the following holds if CFL$\leq c_0$:

(i) $\|u(T) - u_h(T)\|_{L^\infty((0,T);L^1)} \leq ch^{\frac{1}{2}}$ if a priori BV estimate on u_h.

(ii) $\|u(T) - u_h(T)\|_{L^\infty((0,T);L^1)} \leq ch^{\frac{1}{4}}$ otherwise.

- BV estimate is trivial in 1D (Harten's lemma).
- BV estimate can be proved in nD on special meshes.
- Similar results for FV **Chainais-Hillairet (1999), Eymard et al (1998)**
- First error estimates for explicit continuous FE method (as far as we know).

A priori error estimate for scalar equations

English translation

Control on all the Kruskov entropies $\Rightarrow$ Convergence estimate.

Theorem (Guermond, Popov (2014-15))

Assume $u_0 \in BV$ and $\mathbf{f}$ Lipschitz. Let u_h be the first-order viscosity solution. Then there is c_0, uniform, such that the following holds if CFL$\le c_0$:

(i) $\|u(T) - u_h(T)\|_{L^\infty((0,T);L^1)} \le ch^{\frac{1}{2}}$ if a priori BV estimate on u_h.

(ii) $\|u(T) - u_h(T)\|_{L^\infty((0,T);L^1)} \le ch^{\frac{1}{4}}$ otherwise.

- BV estimate is trivial in 1D (Harten's lemma).
- BV estimate can be proved in nD on special meshes.
- Similar results for FV **Chainais-Hillairet (1999), Eymard et al (1998)**
- First error estimates for explicit continuous FE method (as far as we know).

High-order extension

Higher-order in time

- Use SSP method to get higher-order in time.
- Strong Stability Preserving methods (SSP), **Kraaijevanger (1991)** (amazing paper), **Gottlieb-Shu-Tadmor (2001), Spiteri-Ruuth (2002) Ferracina-Spijker (2005), Higueras (2005), etc.**:

Remark on SSP

- SSP is not about positivity, it is about convexity.
- Let A be a convex set and assume that $\mathbf{U} \longmapsto S_{\Delta t}(\mathbf{U})$ is an SSP scheme based on Euler step $\mathbf{U} \longmapsto E_{\Delta t}(\mathbf{U})$ for all $\Delta t \leq \Delta t_0$, then

$$\Big(\text{If Euler step } E_{\Delta t}(\mathbf{U}) \text{ is invariant domain preserving in } A\Big)$$
$$\text{then } \Big(\text{SSP step } S_{\Delta t}(\mathbf{U}) \text{ is invariant domain preserving in } A\Big)$$

- SSP methods preserve convex domains that are invariant for forward Euler time stepping (It's all about convexity).

High-order extension

Higher-order in time

- Use SSP method to get higher-order in time.
- Strong Stability Preserving methods (SSP), **Kraaijevanger (1991)** (amazing paper), **Gottlieb-Shu-Tadmor (2001), Spiteri-Ruuth (2002) Ferracina-Spijker (2005), Higueras (2005), etc.**:

Remark on SSP

- SSP is not about positivity, it is about convexity.
- Let A be a convex set and assume that $\mathbf{U} \longmapsto S_{\Delta t}(\mathbf{U})$ is an SSP scheme based on Euler step $\mathbf{U} \longmapsto E_{\Delta t}(\mathbf{U})$ for all $\Delta t \leq \Delta t_0$, then

$$\Big(\text{If Euler step } E_{\Delta t}(\mathbf{U}) \text{ is invariant domain preserving in } A\Big)$$

$$\text{then } \Big(\text{SSP step } S_{\Delta t}(\mathbf{U}) \text{ is invariant domain preserving in } A\Big)$$

- SSP methods preserve convex domains that are invariant for forward Euler time stepping (It's all about convexity).

High-order extension

Higher-order in time

- Use SSP method to get higher-order in time.
- Strong Stability Preserving methods (SSP), **Kraaijevanger (1991)** (amazing paper), **Gottlieb-Shu-Tadmor (2001), Spiteri-Ruuth (2002) Ferracina-Spijker (2005), Higueras (2005), etc.**:

Remark on SSP

- SSP is not about positivity, it is about convexity.
- Let A be a convex set and assume that $\mathbf{U} \longmapsto S_{\Delta t}(\mathbf{U})$ is an SSP scheme based on Euler step $\mathbf{U} \longmapsto E_{\Delta t}(\mathbf{U})$ for all $\Delta t \leq \Delta t_0$, then

$$\Big(\text{If Euler step } E_{\Delta t}(\mathbf{U}) \text{ is invariant domain preserving in } A\Big)$$

$$\text{then } \Big(\text{SSP step } S_{\Delta t}(\mathbf{U}) \text{ is invariant domain preserving in } A\Big)$$

- SSP methods preserve convex domains that are invariant for forward Euler time stepping (It's all about convexity).

High-order extension

Higher-order in time

- Use SSP method to get higher-order in time.
- Strong Stability Preserving methods (SSP), **Kraaijevanger (1991)** (amazing paper), **Gottlieb-Shu-Tadmor (2001), Spiteri-Ruuth (2002) Ferracina-Spijker (2005), Higueras (2005), etc.**:

Remark on SSP

- SSP is not about positivity, it is about convexity.
- Let A be a convex set and assume that $\mathbf{U} \longmapsto S_{\Delta t}(\mathbf{U})$ is an SSP scheme based on Euler step $\mathbf{U} \longmapsto E_{\Delta t}(\mathbf{U})$ for all $\Delta t \leq \Delta t_0$, then

$$\Big(\text{If Euler step } E_{\Delta t}(\mathbf{U}) \text{ is invariant domain preserving in } A\Big)$$

$$\text{then } \Big(\text{SSP step } S_{\Delta t}(\mathbf{U}) \text{ is invariant domain preserving in } A\Big)$$

- SSP methods preserve convex domains that are invariant for forward Euler time stepping (It's all about convexity).

High-order extension

Higher-order in time

- Use SSP method to get higher-order in time.
- Strong Stability Preserving methods (SSP), **Kraaijevanger (1991)** (amazing paper), **Gottlieb-Shu-Tadmor (2001), Spiteri-Ruuth (2002) Ferracina-Spijker (2005), Higueras (2005), etc.**:

Remark on SSP

- SSP is not about positivity, it is about convexity.
- Let A be a convex set and assume that $\mathbf{U} \longmapsto S_{\Delta t}(\mathbf{U})$ is an SSP scheme based on Euler step $\mathbf{U} \longmapsto E_{\Delta t}(\mathbf{U})$ for all $\Delta t \leq \Delta t_0$, then

$$\Big(\text{If Euler step } E_{\Delta t}(\mathbf{U}) \text{ is invariant domain preserving in } A\Big)$$

$$\text{then } \Big(\text{SSP step } S_{\Delta t}(\mathbf{U}) \text{ is invariant domain preserving in } A\Big)$$

- SSP methods preserve convex domains that are invariant for forward Euler time stepping (It's all about convexity).

High-order extension

Higher-order in time

- Use SSP method to get higher-order in time.
- Strong Stability Preserving methods (SSP), **Kraaijevanger (1991)** (amazing paper), **Gottlieb-Shu-Tadmor (2001), Spiteri-Ruuth (2002) Ferracina-Spijker (2005), Higueras (2005), etc.**:

Remark on SSP

- SSP is not about positivity, it is about convexity.
- Let A be a convex set and assume that $\mathbf{U} \longmapsto S_{\Delta t}(\mathbf{U})$ is an SSP scheme based on Euler step $\mathbf{U} \longmapsto E_{\Delta t}(\mathbf{U})$ for all $\Delta t \leq \Delta t_0$, then

$$\Big(\text{If Euler step } E_{\Delta t}(\mathbf{U}) \text{ is invariant domain preserving in } A\Big)$$

$$\text{then } \Big(\text{SSP step } S_{\Delta t}(\mathbf{U}) \text{ is invariant domain preserving in } A\Big)$$

- SSP methods preserve convex domains that are invariant for forward Euler time stepping (It's all about convexity).

High-order extension

Higher-order in time

- Use SSP method to get higher-order in time.
- Strong Stability Preserving methods (SSP), **Kraaijevanger (1991)** (amazing paper), **Gottlieb-Shu-Tadmor (2001), Spiteri-Ruuth (2002) Ferracina-Spijker (2005), Higueras (2005), etc.**:

Remark on SSP

- SSP is not about positivity, it is about convexity.
- Let A be a convex set and assume that $\mathbf{U} \longmapsto S_{\Delta t}(\mathbf{U})$ is an SSP scheme based on Euler step $\mathbf{U} \longmapsto E_{\Delta t}(\mathbf{U})$ for all $\Delta t \leq \Delta t_0$, then

$$\Big(\text{If Euler step } E_{\Delta t}(\mathbf{U}) \text{ is invariant domain preserving in } A\Big)$$
$$\text{then } \Big(\text{SSP step } S_{\Delta t}(\mathbf{U}) \text{ is invariant domain preserving in } A\Big)$$

- SSP methods preserve convex domains that are invariant for forward Euler time stepping (It's all about convexity).

High-order extension

Higher-order in time

- Use SSP method to get higher-order in time.
- Strong Stability Preserving methods (SSP), **Kraaijevanger (1991)** (amazing paper), **Gottlieb-Shu-Tadmor (2001), Spiteri-Ruuth (2002) Ferracina-Spijker (2005), Higueras (2005), etc.**:

Remark on SSP

- SSP is not about positivity, it is about convexity.
- Let A be a convex set and assume that $\mathbf{U} \longmapsto S_{\Delta t}(\mathbf{U})$ is an SSP scheme based on Euler step $\mathbf{U} \longmapsto E_{\Delta t}(\mathbf{U})$ for all $\Delta t \leq \Delta t_0$, then

$$\Big(\text{If Euler step } E_{\Delta t}(\mathbf{U}) \text{ is invariant domain preserving in } A\Big)$$

$$\text{then } \Big(\text{SSP step } S_{\Delta t}(\mathbf{U}) \text{ is invariant domain preserving in } A\Big)$$

- SSP methods preserve convex domains that are invariant for forward Euler time stepping (It's all about convexity).

High-order extension

Higher-order in space: Entropy viscosity

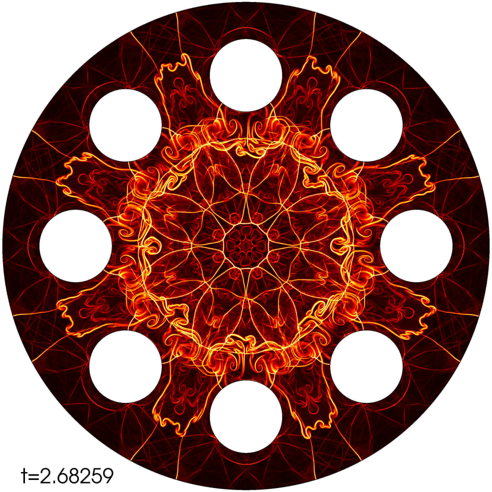
- Use entropy viscosity (or something else)
- FCT or other limitation (work in progress)

High-order extension

Higher-order in space: Entropy viscosity

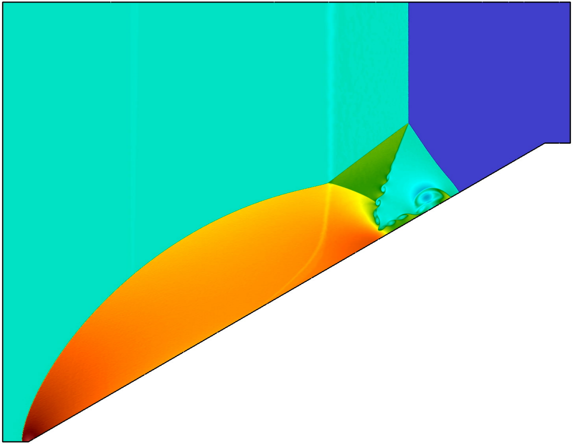
- Use entropy viscosity (or something else)
- FCT or other limitation (work in progress)

Strong explosion; ent. vis. sol. 1.5 million $\mathbb{P}_2$ nodes



(author: Murtazo Nazarov; 1.5 million $\mathbb{P}_2$ nodes)

Mach 10 ramp, ent. vis. sol. 1.2 million $\mathbb{P}_2$ nodes



(author: Murtazo Nazarov; 1.2 millions $\mathbb{P}_2$ nodes)

Hyperbolic systems + ALE

Hyperbolic systems

ALE formulation

- Instead of tracking the characteristics (there are too many), we want to move the mesh.

ALE formulation

- Let $\mathbf{\Phi} : \mathbb{R}^d \times \mathbb{R}_+ \longrightarrow \mathbb{R}^d$ be a uniformly Lipschitz mapping ($\mathbb{R}^d \ni \boldsymbol{\xi} \longmapsto \mathbf{\Phi}(\boldsymbol{\xi}, t) \in \mathbb{R}^d$ invertible on $[0, t^*]$)
- Let $\mathbf{v}_{\mathrm{A}}(\mathbf{x}, t) = \partial_t \mathbf{\Phi}(\mathbf{\Phi}_t^{-1}(\mathbf{x}), t)$ Arbitrary Lagrangian Eulerian velocity

- We are going to use $\mathbf{v}_{\mathrm{A}}$ to move the mesh.

Lemma

The following holds in the distribution sense (in time) over $[0, t^*]$ for every function $\psi \in C^0_0(\mathbb{R}^d; \mathbb{R})$ (with the notation $\varphi(\mathbf{x}, t) := \psi(\mathbf{\Phi}_t^{-1}(\mathbf{x}))$):

$$\partial_t \int_{\mathbb{R}^d} \mathbf{u}(\mathbf{x}, t)\varphi(\mathbf{x}, t)\,\mathrm{d}\mathbf{x} = \int_{\mathbb{R}^d} \nabla\cdot(\mathbf{u} \otimes \mathbf{v}_{\mathrm{A}} - \mathbf{f}(\mathbf{u}))\varphi(\mathbf{x}, t)\,\mathrm{d}\mathbf{x}.$$

ALE formulation

- Instead of tracking the characteristics (there are too many), we want to move the mesh.

ALE formulation

- Let $\mathbf{\Phi} : \mathbb{R}^d \times \mathbb{R}_+ \longrightarrow \mathbb{R}^d$ be a uniformly Lipschitz mapping ($\mathbb{R}^d \ni \boldsymbol{\xi} \longmapsto \mathbf{\Phi}(\boldsymbol{\xi}, t) \in \mathbb{R}^d$ invertible on $[0, t^*]$)
- Let $\mathbf{v}_{\mathrm{A}}(\mathbf{x}, t) = \partial_t \mathbf{\Phi}(\mathbf{\Phi}_t^{-1}(\mathbf{x}), t)$ Arbitrary Lagrangian Eulerian velocity

- We are going to use $\mathbf{v}_{\mathrm{A}}$ to move the mesh.

Lemma

The following holds in the distribution sense (in time) over $[0, t^*]$ for every function $\psi \in C_0^0(\mathbb{R}^d; \mathbb{R})$ (with the notation $\varphi(\mathbf{x}, t) := \psi(\mathbf{\Phi}_t^{-1}(\mathbf{x}))$):

$$\partial_t \int_{\mathbb{R}^d} \mathbf{u}(\mathbf{x}, t)\varphi(\mathbf{x}, t)\,\mathrm{d}\mathbf{x} = \int_{\mathbb{R}^d} \nabla\cdot(\mathbf{u} \otimes \mathbf{v}_{\mathrm{A}} - \mathbf{f}(\mathbf{u}))\varphi(\mathbf{x}, t)\,\mathrm{d}\mathbf{x}.$$

ALE formulation

- Instead of tracking the characteristics (there are too many), we want to move the mesh.

ALE formulation

- Let $\mathbf{\Phi} : \mathbb{R}^d \times \mathbb{R}_+ \longrightarrow \mathbb{R}^d$ be a uniformly Lipschitz mapping ($\mathbb{R}^d \ni \boldsymbol{\xi} \longmapsto \mathbf{\Phi}(\boldsymbol{\xi}, t) \in \mathbb{R}^d$ invertible on $[0, t^*]$)
- Let $\mathbf{v}_{\mathrm{A}}(\mathbf{x}, t) = \partial_t \mathbf{\Phi}(\mathbf{\Phi}_t^{-1}(\mathbf{x}), t)$ Arbitrary Lagrangian Eulerian velocity

- We are going to use $\mathbf{v}_{\mathrm{A}}$ to move the mesh.

Lemma

The following holds in the distribution sense (in time) over $[0, t^*]$ for every function $\psi \in C_0^0(\mathbb{R}^d; \mathbb{R})$ (with the notation $\varphi(\mathbf{x}, t) := \psi(\mathbf{\Phi}_t^{-1}(\mathbf{x}))$):

$$\partial_t \int_{\mathbb{R}^d} \mathbf{u}(\mathbf{x}, t)\varphi(\mathbf{x}, t)\,\mathrm{d}\mathbf{x} = \int_{\mathbb{R}^d} \nabla\cdot(\mathbf{u} \otimes \mathbf{v}_{\mathrm{A}} - \mathbf{f}(\mathbf{u}))\varphi(\mathbf{x}, t)\,\mathrm{d}\mathbf{x}.$$

ALE formulation

- Instead of tracking the characteristics (there are too many), we want to move the mesh.

ALE formulation

- Let $\mathbf{\Phi} : \mathbb{R}^d \times \mathbb{R}_+ \longrightarrow \mathbb{R}^d$ be a uniformly Lipschitz mapping ($\mathbb{R}^d \ni \boldsymbol{\xi} \longmapsto \mathbf{\Phi}(\boldsymbol{\xi}, t) \in \mathbb{R}^d$ invertible on $[0, t^*]$)
- Let $\mathbf{v}_{\mathrm{A}}(\mathbf{x}, t) = \partial_t \mathbf{\Phi}(\mathbf{\Phi}_t^{-1}(\mathbf{x}), t)$ Arbitrary Lagrangian Eulerian velocity

- We are going to use $\mathbf{v}_{\mathrm{A}}$ to move the mesh.

Lemma

The following holds in the distribution sense (in time) over $[0, t^*]$ for every function $\psi \in C_0^0(\mathbb{R}^d; \mathbb{R})$ (with the notation $\varphi(\mathbf{x}, t) := \psi(\mathbf{\Phi}_t^{-1}(\mathbf{x}))$):

$$\partial_t \int_{\mathbb{R}^d} \mathbf{u}(\mathbf{x}, t)\varphi(\mathbf{x}, t)\,\mathrm{d}\mathbf{x} = \int_{\mathbb{R}^d} \nabla\cdot(\mathbf{u} \otimes \mathbf{v}_{\mathrm{A}} - \mathbf{f}(\mathbf{u}))\varphi(\mathbf{x}, t)\,\mathrm{d}\mathbf{x}.$$

ALE formulation

- Instead of tracking the characteristics (there are too many), we want to move the mesh.

ALE formulation

- Let $\mathbf{\Phi} : \mathbb{R}^d \times \mathbb{R}_+ \longrightarrow \mathbb{R}^d$ be a uniformly Lipschitz mapping ($\mathbb{R}^d \ni \boldsymbol{\xi} \longmapsto \mathbf{\Phi}(\boldsymbol{\xi}, t) \in \mathbb{R}^d$ invertible on $[0, t^*]$)
- Let $\mathbf{v}_{\mathrm{A}}(\mathbf{x}, t) = \partial_t \mathbf{\Phi}(\mathbf{\Phi}_t^{-1}(\mathbf{x}), t)$ Arbitrary Lagrangian Eulerian velocity

- We are going to use $\mathbf{v}_{\mathrm{A}}$ to move the mesh.

Lemma

The following holds in the distribution sense (in time) over $[0, t^*]$ *for every function* $\psi \in C_0^0(\mathbb{R}^d; \mathbb{R})$ *(with the notation* $\varphi(\mathbf{x}, t) := \psi(\mathbf{\Phi}_t^{-1}(\mathbf{x}))$*):*

$$\partial_t \int_{\mathbb{R}^d} \mathbf{u}(\mathbf{x}, t)\varphi(\mathbf{x}, t)\,\mathrm{d}\mathbf{x} = \int_{\mathbb{R}^d} \nabla\cdot(\mathbf{u} \otimes \mathbf{v}_{\mathrm{A}} - \mathbf{f}(\mathbf{u}))\varphi(\mathbf{x}, t)\,\mathrm{d}\mathbf{x}.$$

Finite elements

Geometric Finite elements

- Let $(\mathcal{T}_h^0)_{h>0}$ be a shape-regular sequence of matching meshes.
- Reference Lagrange finite element $(\widehat{K}, \widehat{P}^{\rm geo}, \widehat{\Sigma}^{\rm geo})$ for geometry
- Lagrange nodes $\{\widehat{\mathbf{a}}_i\}_{i\in\{1:n_{\rm sh}^{\rm geo}\}}$ and Lagrange shape functions $\{\widehat{\theta}_i^{\rm geo}\}_{i\in\{1:n_{\rm sh}^{\rm geo}\}}$
- $\{\mathbf{a}_i^n\}_{i\in\{1:I^{\rm geo}\}}$ collection of all the Lagrange nodes in the mesh $\mathcal{T}_h^n$
- $\mathrm{j}^{\rm geo} : \mathcal{T}_h^n \times \{1:n_{\rm sh}^{\rm geo}\} \longrightarrow \{1:I^{\rm geo}\}$ geometric connectivity array
- Geometric transformation $T_K^n : \widehat{K} \longrightarrow K$ defined by

$$T_K^n(\widehat{\mathbf{x}}) = \sum_{i\in\{1:n_{\rm sh}^{\rm geo}\}} \mathbf{a}^n_{\mathrm{j}^{\rm geo}(i,K)} \widehat{\theta}_i^{\rm geo}(\widehat{\mathbf{x}}).$$

$\Rightarrow$ Mesh motion controlled by motion of Lagrange nodes

Finite elements

Geometric Finite elements

- Let $(\mathcal{T}_h^0)_{h>0}$ be a shape-regular sequence of matching meshes.
- Reference Lagrange finite element $(\widehat{K}, \widehat{P}^{\rm geo}, \widehat{\Sigma}^{\rm geo})$ for geometry
- Lagrange nodes $\{\widehat{\mathbf{a}}_i\}_{i\in\{1:n_{\rm sh}^{\rm geo}\}}$ and Lagrange shape functions $\{\widehat{\theta}_i^{\rm geo}\}_{i\in\{1:n_{\rm sh}^{\rm geo}\}}$
- $\{\mathbf{a}_i^n\}_{i\in\{1:I^{\rm geo}\}}$ collection of all the Lagrange nodes in the mesh $\mathcal{T}_h^n$
- $\mathrm{j}^{\rm geo} : \mathcal{T}_h^n \times \{1:n_{\rm sh}^{\rm geo}\} \longrightarrow \{1:I^{\rm geo}\}$ geometric connectivity array
- Geometric transformation $T_K^n : \widehat{K} \longrightarrow K$ defined by

$$T_K^n(\widehat{\mathbf{x}}) = \sum_{i\in\{1:n_{\rm sh}^{\rm geo}\}} \mathbf{a}_{\mathrm{j}^{\rm geo}(i,K)}^n \widehat{\theta}_i^{\rm geo}(\widehat{\mathbf{x}}).$$

$\Rightarrow$ Mesh motion controlled by motion of Lagrange nodes

Finite elements

Geometric Finite elements

- Let $(\mathcal{T}_h^0)_{h>0}$ be a shape-regular sequence of matching meshes.
- Reference Lagrange finite element $(\widehat{K}, \widehat{P}^{\text{geo}}, \widehat{\Sigma}^{\text{geo}})$ for geometry
- Lagrange nodes $\{\widehat{\mathbf{a}}_i\}_{i\in\{1:n_{\text{sh}}^{\text{geo}}\}}$ and Lagrange shape functions $\{\widehat{\theta}_i^{\text{geo}}\}_{i\in\{1:n_{\text{sh}}^{\text{geo}}\}}$
- $\{\mathbf{a}_i^n\}_{i\in\{1:I^{\text{geo}}\}}$ collection of all the Lagrange nodes in the mesh $\mathcal{T}_h^n$
- $\text{j}^{\text{geo}} : \mathcal{T}_h^n \times \{1:n_{\text{sh}}^{\text{geo}}\} \longrightarrow \{1:I^{\text{geo}}\}$ geometric connectivity array
- Geometric transformation $T_K^n : \widehat{K} \longrightarrow K$ defined by

$$T_K^n(\widehat{\mathbf{x}}) = \sum_{i\in\{1:n_{\text{sh}}^{\text{geo}}\}} \mathbf{a}^n_{\text{j}^{\text{geo}}(i,K)} \widehat{\theta}_i^{\text{geo}}(\widehat{\mathbf{x}}).$$

$\Rightarrow$ Mesh motion controlled by motion of Lagrange nodes

Finite elements

Geometric Finite elements

- Let $(\mathcal{T}_h^0)_{h>0}$ be a shape-regular sequence of matching meshes.
- Reference Lagrange finite element $(\widehat{K}, \widehat{P}^{\text{geo}}, \widehat{\Sigma}^{\text{geo}})$ for geometry
- Lagrange nodes $\{\widehat{\mathbf{a}}_i\}_{i\in\{1:n_{\text{sh}}^{\text{geo}}\}}$ and Lagrange shape functions $\{\widehat{\theta}_i^{\text{geo}}\}_{i\in\{1:n_{\text{sh}}^{\text{geo}}\}}$
- $\{\mathbf{a}_i^n\}_{i\in\{1:I^{\text{geo}}\}}$ collection of all the Lagrange nodes in the mesh $\mathcal{T}_h^n$
- $\mathrm{j}^{\text{geo}} : \mathcal{T}_h^n \times \{1:n_{\text{sh}}^{\text{geo}}\} \longrightarrow \{1:I^{\text{geo}}\}$ geometric connectivity array
- Geometric transformation $T_K^n : \widehat{K} \longrightarrow K$ defined by

$$T_K^n(\widehat{\mathbf{x}}) = \sum_{i\in\{1:n_{\text{sh}}^{\text{geo}}\}} \mathbf{a}^n_{\mathrm{j}^{\text{geo}}(i,K)} \widehat{\theta}_i^{\text{geo}}(\widehat{\mathbf{x}}).$$

$\Rightarrow$ Mesh motion controlled by motion of Lagrange nodes

Finite elements

Geometric Finite elements

- Let $(\mathcal{T}_h^0)_{h>0}$ be a shape-regular sequence of matching meshes.
- Reference Lagrange finite element $(\widehat{K}, \widehat{P}^{\rm geo}, \widehat{\Sigma}^{\rm geo})$ for geometry
- Lagrange nodes $\{\widehat{\mathbf{a}}_i\}_{i\in\{1:n_{\rm sh}^{\rm geo}\}}$ and Lagrange shape functions $\{\widehat{\theta}_i^{\rm geo}\}_{i\in\{1:n_{\rm sh}^{\rm geo}\}}$
- $\{\mathbf{a}_i^n\}_{i\in\{1:I^{\rm geo}\}}$ collection of all the Lagrange nodes in the mesh $\mathcal{T}_h^n$
- $\mathrm{j}^{\rm geo} : \mathcal{T}_h^n \times \{1:n_{\rm sh}^{\rm geo}\} \longrightarrow \{1:I^{\rm geo}\}$ geometric connectivity array
- Geometric transformation $T_K^n : \widehat{K} \longrightarrow K$ defined by

$$T_K^n(\widehat{\mathbf{x}}) = \sum_{i\in\{1:n_{\rm sh}^{\rm geo}\}} \mathbf{a}^n_{\mathrm{j}^{\rm geo}(i,K)} \widehat{\theta}_i^{\rm geo}(\widehat{\mathbf{x}}).$$

$\Rightarrow$ Mesh motion controlled by motion of Lagrange nodes

Finite elements

Geometric Finite elements

- Let $(\mathcal{T}_h^0)_{h>0}$ be a shape-regular sequence of matching meshes.
- Reference Lagrange finite element $(\widehat{K}, \widehat{P}^{\mathrm{geo}}, \widehat{\Sigma}^{\mathrm{geo}})$ for geometry
- Lagrange nodes $\{\widehat{\mathbf{a}}_i\}_{i\in\{1:n_{\mathrm{sh}}^{\mathrm{geo}}\}}$ and Lagrange shape functions $\{\widehat{\theta}_i^{\mathrm{geo}}\}_{i\in\{1:n_{\mathrm{sh}}^{\mathrm{geo}}\}}$
- $\{\mathbf{a}_i^n\}_{i\in\{1:I^{\mathrm{geo}}\}}$ collection of all the Lagrange nodes in the mesh $\mathcal{T}_h^n$
- $\mathrm{j}^{\mathrm{geo}} : \mathcal{T}_h^n \times \{1:n_{\mathrm{sh}}^{\mathrm{geo}}\} \longrightarrow \{1:I^{\mathrm{geo}}\}$ geometric connectivity array
- Geometric transformation $T_K^n : \widehat{K} \longrightarrow K$ defined by

$$T_K^n(\widehat{\mathbf{x}}) = \sum_{i\in\{1:n_{\mathrm{sh}}^{\mathrm{geo}}\}} \mathbf{a}^n_{\mathrm{j}^{\mathrm{geo}}(i,K)} \widehat{\theta}_i^{\mathrm{geo}}(\widehat{\mathbf{x}}).$$

$\Rightarrow$ Mesh motion controlled by motion of Lagrange nodes

Finite elements

Geometric Finite elements

- Let $(\mathcal{T}_h^0)_{h>0}$ be a shape-regular sequence of matching meshes.
- Reference Lagrange finite element $(\widehat{K}, \widehat{P}^{\rm geo}, \widehat{\Sigma}^{\rm geo})$ for geometry
- Lagrange nodes $\{\widehat{\mathbf{a}}_i\}_{i\in\{1:n_{\rm sh}^{\rm geo}\}}$ and Lagrange shape functions $\{\widehat{\theta}_i^{\rm geo}\}_{i\in\{1:n_{\rm sh}^{\rm geo}\}}$
- $\{\mathbf{a}_i^n\}_{i\in\{1:I^{\rm geo}\}}$ collection of all the Lagrange nodes in the mesh $\mathcal{T}_h^n$
- $\mathrm{j}^{\rm geo} : \mathcal{T}_h^n \times \{1:n_{\rm sh}^{\rm geo}\} \longrightarrow \{1:I^{\rm geo}\}$ geometric connectivity array
- Geometric transformation $T_K^n : \widehat{K} \longrightarrow K$ defined by

$$T_K^n(\widehat{\mathbf{x}}) = \sum_{i\in\{1:n_{\rm sh}^{\rm geo}\}} \mathbf{a}^n_{\mathrm{j}^{\rm geo}(i,K)} \widehat{\theta}_i^{\rm geo}(\widehat{\mathbf{x}}).$$

$\Rightarrow$ Mesh motion controlled by motion of Lagrange nodes

Finite elements

Approximating Finite elements

- Reference finite element $(\widehat{K}, \widehat{P}, \widehat{\Sigma})\}$
- Shape functions $\widehat{\theta}_i(\mathbf{x}) \geq 0$, $\sum_{i\in\{1:n_{\rm sh}\}} \widehat{\theta}_i(\widehat{\mathbf{x}}) = 1$
- Finite element spaces

$$P(\mathcal{T}_h^n) := \{v \in \mathcal{C}^0(D^n;\mathbb{R});\ v_{|K} \circ T_K^n \in \widehat{P},\ \forall K \in \mathcal{T}_h^n\},$$
$$\mathbf{P}_d(\mathcal{T}_h^n) := [P(\mathcal{T}_h^n)]^d,$$
$$\mathbf{P}_m(\mathcal{T}_h^n) := [P(\mathcal{T}_h^n)]^m.$$

- $\{\psi_i^n\}_{i\in\{1:I\}}$ global shape functions in $P(\mathcal{T}_h^n)$.

Finite elements

Approximating Finite elements

- Reference finite element $(\widehat{K}, \widehat{P}, \widehat{\Sigma})\}$
- Shape functions $\widehat{\theta}_i(\mathbf{x}) \geq 0$, $\sum_{i\in\{1:n_{\rm sh}\}} \widehat{\theta}_i(\widehat{\mathbf{x}}) = 1$
- Finite element spaces

$$P(\mathcal{T}_h^n) := \{v \in \mathcal{C}^0(D^n;\mathbb{R});\ v_{|K}\circ T_K^n \in \widehat{P},\ \forall K \in \mathcal{T}_h^n\},$$
$$\mathbf{P}_d(\mathcal{T}_h^n) := [P(\mathcal{T}_h^n)]^d,$$
$$\mathbf{P}_m(\mathcal{T}_h^n) := [P(\mathcal{T}_h^n)]^m.$$

- $\{\psi_i^n\}_{i\in\{1:I\}}$ global shape functions in $P(\mathcal{T}_h^n)$.

Finite elements

Approximating Finite elements

- Reference finite element $(\widehat{K}, \widehat{P}, \widehat{\Sigma})\}$
- Shape functions $\widehat{\theta}_i(\mathbf{x}) \geq 0$, $\sum_{i\in\{1:n_{\text{sh}}\}} \widehat{\theta}_i(\widehat{\mathbf{x}}) = 1$
- Finite element spaces

$$
\begin{aligned}
P(\mathcal{T}_h^n) &:= \{v \in \mathcal{C}^0(D^n;\mathbb{R});\ v_{|K}\circ T_K^n \in \widehat{P},\ \forall K \in \mathcal{T}_h^n\}, \\
\mathbf{P}_d(\mathcal{T}_h^n) &:= [P(\mathcal{T}_h^n)]^d, \\
\mathbf{P}_m(\mathcal{T}_h^n) &:= [P(\mathcal{T}_h^n)]^m.
\end{aligned}
$$

- $\{\psi_i^n\}_{i\in\{1:I\}}$ global shape functions in $P(\mathcal{T}_h^n)$.

Finite elements

Approximating Finite elements

- Reference finite element $(\widehat{K}, \widehat{P}, \widehat{\Sigma})\}$
- Shape functions $\widehat{\theta}_i(\mathbf{x}) \geq 0$, $\sum_{i\in\{1:n_{\rm sh}\}} \widehat{\theta}_i(\widehat{\mathbf{x}}) = 1$
- Finite element spaces

$$
\begin{aligned}
P(\mathcal{T}_h^n) &:= \{v \in \mathcal{C}^0(D^n;\mathbb{R});\ v_{|K} \circ T_K^n \in \widehat{P},\ \forall K \in \mathcal{T}_h^n\},\\
\mathbf{P}_d(\mathcal{T}_h^n) &:= [P(\mathcal{T}_h^n)]^d,\\
\mathbf{P}_m(\mathcal{T}_h^n) &:= [P(\mathcal{T}_h^n)]^m.
\end{aligned}
$$

- $\{\psi_i^n\}_{i\in\{1:I\}}$ global shape functions in $P(\mathcal{T}_h^n)$.

Finite elements

The algorithm

- Initialization: $\mathfrak{m}_i^0 := \int_{\mathbb{R}^d} \psi_i^n(\mathbf{x})\,\mathrm{d}\mathbf{x}$ $\mathbf{u}_{h0} := \sum_{i\in\{1:I\}} \mathbf{U}_i^0 \psi_i^0 \in \mathbf{P}_m(\mathcal{T}_h^0)$
- ALE velocity field given: $\mathbf{w}^n = \sum_{i\in\{1:I\}} \mathbf{W}_i^n \psi_i^n \in \mathbf{P}_d(\mathcal{T}_h^n)$,
- Mesh motion:
$$\mathbf{a}_i^{n+1} = \mathbf{a}_i^n + \Delta t \mathbf{w}^n(\mathbf{a}_i^n).$$
- Mass update: (do not use $\mathfrak{m}_i^{n+1} = \int_D \psi_i^{n+1}\,\mathrm{d}\mathbf{x}$!)
$$\mathfrak{m}_i^{n+1} = \mathfrak{m}_i^n + \Delta t \int_{S_i^n} \psi_i^n(\mathbf{x}) \nabla\cdot\mathbf{w}^n(\mathbf{x})\,\mathrm{d}\mathbf{x}.$$
- Update approximation field $\mathbf{u}_h^{n+1}$
$$\frac{\mathfrak{m}_i^{n+1}\mathbf{U}_i^{n+1} - \mathfrak{m}_i^n\mathbf{U}_i^n}{\Delta t} - \sum_{j\in\mathcal{I}(S_i^n)} d_{ij}^n \mathbf{U}_j^n$$
$$+ \int_{\mathbb{R}^d} \nabla\cdot\Big(\sum_{j\in\{1:I\}} (\mathbf{f}(\mathbf{U}_j^n) - \mathbf{U}_j^n \otimes \mathbf{W}_j^n)\psi_j^n(\mathbf{x})\Big)\psi_i^n(\mathbf{x})\,\mathrm{d}\mathbf{x} = 0,$$

Finite elements

The algorithm

- Initialization: $m_i^0 := \int_{\mathbb{R}^d} \psi_i^n(\mathbf{x})\,\mathrm{d}\mathbf{x}$ $\mathbf{u}_{h0} := \sum_{i\in\{1:I\}} \mathbf{U}_i^0 \psi_i^0 \in \mathbf{P}_m(\mathcal{T}_h^0)$
- ALE velocity field given: $\mathbf{w}^n = \sum_{i\in\{1:I\}} \mathbf{W}_i^n \psi_i^n \in \mathbf{P}_d(\mathcal{T}_h^n)$,
- Mesh motion:
$$\mathbf{a}_i^{n+1} = \mathbf{a}_i^n + \Delta t \mathbf{w}^n(\mathbf{a}_i^n).$$
- Mass update: (do not use $m_i^{n+1} = \int_D \psi_i^{n+1}\,\mathrm{d}\mathbf{x}$!)
$$m_i^{n+1} = m_i^n + \Delta t \int_{S_i^n} \psi_i^n(\mathbf{x}) \nabla\cdot \mathbf{w}^n(\mathbf{x})\,\mathrm{d}\mathbf{x}.$$
- Update approximation field $\mathbf{u}_h^{n+1}$
$$\frac{m_i^{n+1}\mathbf{U}_i^{n+1} - m_i^n \mathbf{U}_i^n}{\Delta t} - \sum_{j\in\mathcal{I}(S_i^n)} d_{ij}^n \mathbf{U}_j^n + \int_{\mathbb{R}^d} \nabla\cdot\Big(\sum_{j\in\{1:I\}} (\mathbf{f}(\mathbf{U}_j^n) - \mathbf{U}_j^n \otimes \mathbf{W}_j^n)\psi_j^n(\mathbf{x})\Big)\psi_i^n(\mathbf{x})\,\mathrm{d}\mathbf{x} = 0,$$

Finite elements

The algorithm

- Initialization: $\mathfrak{m}_i^0 := \int_{\mathbb{R}^d} \psi_i^n(\mathbf{x})\,\mathrm{d}\mathbf{x}$ $\mathbf{u}_{h0} := \sum_{i\in\{1:I\}} \mathbf{U}_i^0 \psi_i^0 \in \mathbf{P}_m(\mathcal{T}_h^0)$
- ALE velocity field given: $\mathbf{w}^n = \sum_{i\in\{1:I\}} \mathbf{W}_i^n \psi_i^n \in \mathbf{P}_d(\mathcal{T}_h^n)$,
- Mesh motion:

$$\mathbf{a}_i^{n+1} = \mathbf{a}_i^n + \Delta t \mathbf{w}^n(\mathbf{a}_i^n).$$

- Mass update: (do not use $\mathfrak{m}_i^{n+1} = \int_D \psi_i^{n+1}\,\mathrm{d}\mathbf{x}$!)

$$\mathfrak{m}_i^{n+1} = \mathfrak{m}_i^n + \Delta t \int_{S_i^n} \psi_i^n(\mathbf{x}) \nabla\cdot\mathbf{w}^n(\mathbf{x})\,\mathrm{d}\mathbf{x}.$$

- Update approximation field $\mathbf{u}_h^{n+1}$

$$\frac{\mathfrak{m}_i^{n+1}\mathbf{U}_i^{n+1} - \mathfrak{m}_i^n\mathbf{U}_i^n}{\Delta t} - \sum_{j\in\mathcal{I}(S_i^n)} d_{ij}^n \mathbf{U}_j^n$$
$$+ \int_{\mathbb{R}^d} \nabla\cdot\Big(\sum_{j\in\{1:I\}} (\mathbf{f}(\mathbf{U}_j^n) - \mathbf{U}_j^n \otimes \mathbf{W}_j^n)\psi_j^n(\mathbf{x}) \Big) \psi_i^n(\mathbf{x})\,\mathrm{d}\mathbf{x} = 0,$$

Finite elements

The algorithm

- Initialization: $\mathfrak{m}_i^0 := \int_{\mathbb{R}^d} \psi_i^n(\mathbf{x})\,\mathrm{d}\mathbf{x}$ $\mathbf{u}_{h0} := \sum_{i\in\{1:I\}} \mathbf{U}_i^0 \psi_i^0 \in \mathbf{P}_m(\mathcal{T}_h^0)$
- ALE velocity field given: $\mathbf{w}^n = \sum_{i\in\{1:I\}} \mathbf{W}_i^n \psi_i^n \in \mathbf{P}_d(\mathcal{T}_h^n)$,
- Mesh motion:

$$\mathbf{a}_i^{n+1} = \mathbf{a}_i^n + \Delta t \mathbf{w}^n(\mathbf{a}_i^n).$$

- Mass update: (do not use $\mathfrak{m}_i^{n+1} = \int_D \psi_i^{n+1}\,\mathrm{d}\mathbf{x}$!)

$$\mathfrak{m}_i^{n+1} = \mathfrak{m}_i^n + \Delta t \int_{S_i^n} \psi_i^n(\mathbf{x}) \nabla\cdot \mathbf{w}^n(\mathbf{x})\,\mathrm{d}\mathbf{x}.$$

- Update approximation field $\mathbf{u}_h^{n+1}$

$$\frac{\mathfrak{m}_i^{n+1}\mathbf{U}_i^{n+1} - \mathfrak{m}_i^n \mathbf{U}_i^n}{\Delta t} - \sum_{j\in\mathcal{I}(S_i^n)} d_{ij}^n \mathbf{U}_j^n$$
$$+ \int_{\mathbb{R}^d} \nabla\cdot\Big(\sum_{j\in\{1:I\}} (\mathbf{f}(\mathbf{U}_j^n) - \mathbf{U}_j^n \otimes \mathbf{W}_j^n)\psi_j^n(\mathbf{x})\Big)\psi_i^n(\mathbf{x})\,\mathrm{d}\mathbf{x} = 0,$$

Finite elements

The algorithm

- Initialization: $\mathfrak{m}_i^0 := \int_{\mathbb{R}^d} \psi_i^n(\mathbf{x})\,\mathrm{d}\mathbf{x}$ $\mathbf{u}_{h0} := \sum_{i\in\{1:I\}} \mathbf{U}_i^0\psi_i^0 \in \mathbf{P}_m(\mathcal{T}_h^0)$
- ALE velocity field given: $\mathbf{w}^n = \sum_{i\in\{1:I\}} \mathbf{W}_i^n\psi_i^n \in \mathbf{P}_d(\mathcal{T}_h^n)$,
- Mesh motion:

$$\mathbf{a}_i^{n+1} = \mathbf{a}_i^n + \Delta t \mathbf{w}^n(\mathbf{a}_i^n).$$

- Mass update: (do not use $\mathfrak{m}_i^{n+1} = \int_D \psi_i^{n+1}\,\mathrm{d}\mathbf{x}$!)

$$\mathfrak{m}_i^{n+1} = \mathfrak{m}_i^n + \Delta t \int_{S_i^n} \psi_i^n(\mathbf{x})\nabla{\cdot}\mathbf{w}^n(\mathbf{x})\,\mathrm{d}\mathbf{x}.$$

- Update approximation field $\mathbf{u}_h^{n+1}$

$$\frac{\mathfrak{m}_i^{n+1}\mathbf{U}_i^{n+1} - \mathfrak{m}_i^n\mathbf{U}_i^n}{\Delta t} - \sum_{j\in\mathcal{I}(S_i^n)} d_{ij}^n\mathbf{U}_j^n$$
$$+ \int_{\mathbb{R}^d} \nabla{\cdot}\Big(\sum_{j\in\{1:I\}} (\mathbf{f}(\mathbf{U}_j^n) - \mathbf{U}_j^n\otimes\mathbf{W}_j^n)\psi_j^n(\mathbf{x})\Big)\psi_i^n(\mathbf{x})\,\mathrm{d}\mathbf{x} = 0,$$

Finite elements

Definition of d_{ij}^n

- Consider flux $\mathbf{g}_j^n(\mathbf{v}) := \mathbf{f}(\mathbf{v}) - \mathbf{v} \otimes \mathbf{W}_j^n$, $j \in \{1{:}I\}$
- Consider one-dimensional Riemann problem:

$$\partial_t \mathbf{v} + \partial_x(\mathbf{g}_j^n(\mathbf{v}) \cdot \mathbf{n}_{ij}^n) = 0, \quad (x,t) \in \mathbb{R} \times \mathbb{R}_+, \quad \mathbf{v}(x,0) = \begin{cases} \mathbf{U}_i^n & \text{if } x < 0 \\ \mathbf{U}_j^n & \text{if } x > 0. \end{cases}$$

- Define d_{ij}^n by

$$\boxed{d_{ij}^n = \max(\lambda_{\max}(\mathbf{g}_j^n, \mathbf{n}_{ij}^n, \mathbf{U}_i^n, \mathbf{U}_j^n)\|\mathbf{c}_{ij}^n\|_{\ell^2}, \lambda_{\max}(\mathbf{g}_i^n, \mathbf{n}_{ji}^n, \mathbf{U}_j^n, \mathbf{U}_i^n)\|\mathbf{c}_{ji}^n\|_{\ell^2}).}$$

- Note that

$$\lambda_{\max}(\mathbf{g}_j^n, \mathbf{n}_{ij}^n, \mathbf{U}_i^n, \mathbf{U}_j^n) = \max(|\lambda_L(\mathbf{f}, \mathbf{n}_{ij}^n, \mathbf{U}_i^n, \mathbf{U}_j^n) - \mathbf{W}_j^n \cdot \mathbf{n}_{ij}^n|, \\ |\lambda_R(\mathbf{f}, \mathbf{n}_{ij}^n, \mathbf{U}_i^n, \mathbf{U}_j^n) - \mathbf{W}_j^n \cdot \mathbf{n}_{ij}^n|).$$

Finite elements

Definition of d_{ij}^n

- Consider flux $\mathbf{g}_j^n(\mathbf{v}) := \mathbf{f}(\mathbf{v}) - \mathbf{v} \otimes \mathbf{W}_j^n$, $j \in \{1:I\}$
- Consider one-dimensional Riemann problem:

$$\partial_t \mathbf{v} + \partial_x(\mathbf{g}_j^n(\mathbf{v}) \cdot \mathbf{n}_{ij}^n) = 0, \quad (x,t) \in \mathbb{R} \times \mathbb{R}_+, \quad \mathbf{v}(x,0) = \begin{cases} \mathbf{U}_i^n & \text{if } x < 0 \\ \mathbf{U}_j^n & \text{if } x > 0. \end{cases}$$

- Define d_{ij}^n by

$$d_{ij}^n = \max(\lambda_{\max}(\mathbf{g}_j^n, \mathbf{n}_{ij}^n, \mathbf{U}_i^n, \mathbf{U}_j^n) \|\mathbf{c}_{ij}^n\|_{\ell^2}, \lambda_{\max}(\mathbf{g}_i^n, \mathbf{n}_{ji}^n, \mathbf{U}_j^n, \mathbf{U}_i^n) \|\mathbf{c}_{ji}^n\|_{\ell^2}).$$

- Note that

$$\lambda_{\max}(\mathbf{g}_j^n, \mathbf{n}_{ij}^n, \mathbf{U}_i^n, \mathbf{U}_j^n) = \max(|\lambda_L(\mathbf{f}, \mathbf{n}_{ij}^n, \mathbf{U}_i^n, \mathbf{U}_j^n) - \mathbf{W}_j^n \cdot \mathbf{n}_{ij}^n|, \\ |\lambda_R(\mathbf{f}, \mathbf{n}_{ij}^n, \mathbf{U}_i^n, \mathbf{U}_j^n) - \mathbf{W}_j^n \cdot \mathbf{n}_{ij}^n|).$$

Finite elements

Definition of d_{ij}^n

- Consider flux $\mathbf{g}_j^n(\mathbf{v}) := \mathbf{f}(\mathbf{v}) - \mathbf{v} \otimes \mathbf{W}_j^n$, $j \in \{1{:}I\}$
- Consider one-dimensional Riemann problem:

$$\partial_t \mathbf{v} + \partial_x(\mathbf{g}_j^n(\mathbf{v}) \cdot \mathbf{n}_{ij}^n) = 0, \quad (x,t) \in \mathbb{R} \times \mathbb{R}_+, \quad \mathbf{v}(x,0) = \begin{cases} \mathbf{U}_i^n & \text{if } x < 0 \\ \mathbf{U}_j^n & \text{if } x > 0. \end{cases}$$

- Define d_{ij}^n by

$$\boxed{d_{ij}^n = \max(\lambda_{\max}(\mathbf{g}_j^n, \mathbf{n}_{ij}^n, \mathbf{U}_i^n, \mathbf{U}_j^n)\|\mathbf{c}_{ij}^n\|_{\ell^2}, \lambda_{\max}(\mathbf{g}_i^n, \mathbf{n}_{ji}^n, \mathbf{U}_j^n, \mathbf{U}_i^n)\|\mathbf{c}_{ji}^n\|_{\ell^2}).}$$

- Note that

$$\lambda_{\max}(\mathbf{g}_j^n, \mathbf{n}_{ij}^n, \mathbf{U}_i^n, \mathbf{U}_j^n) = \max(|\lambda_L(\mathbf{f}, \mathbf{n}_{ij}^n, \mathbf{U}_i^n, \mathbf{U}_j^n) - \mathbf{W}_j^n \cdot \mathbf{n}_{ij}^n|, \; |\lambda_R(\mathbf{f}, \mathbf{n}_{ij}^n, \mathbf{U}_i^n, \mathbf{U}_j^n) - \mathbf{W}_j^n \cdot \mathbf{n}_{ij}^n|).$$

Finite elements

Definition of d_{ij}^n

- Consider flux $\mathbf{g}_j^n(\mathbf{v}) := \mathbf{f}(\mathbf{v}) - \mathbf{v} \otimes \mathbf{W}_j^n$, $j \in \{1{:}I\}$
- Consider one-dimensional Riemann problem:

$$\partial_t \mathbf{v} + \partial_x(\mathbf{g}_j^n(\mathbf{v}) \cdot \mathbf{n}_{ij}^n) = 0, \quad (x,t) \in \mathbb{R} \times \mathbb{R}_+, \quad \mathbf{v}(x,0) = \begin{cases} \mathbf{U}_i^n & \text{if } x < 0 \\ \mathbf{U}_j^n & \text{if } x > 0. \end{cases}$$

- Define d_{ij}^n by

$$\boxed{d_{ij}^n = \max(\lambda_{\max}(\mathbf{g}_j^n, \mathbf{n}_{ij}^n, \mathbf{U}_i^n, \mathbf{U}_j^n)\|\mathbf{c}_{ij}^n\|_{\ell^2}, \lambda_{\max}(\mathbf{g}_i^n, \mathbf{n}_{ji}^n, \mathbf{U}_j^n, \mathbf{U}_i^n)\|\mathbf{c}_{ji}^n\|_{\ell^2}).}$$

- Note that

$$\lambda_{\max}(\mathbf{g}_j^n, \mathbf{n}_{ij}^n, \mathbf{U}_i^n, \mathbf{U}_j^n) = \max(|\lambda_L(\mathbf{f}, \mathbf{n}_{ij}^n, \mathbf{U}_i^n, \mathbf{U}_j^n) - \mathbf{W}_j^n \cdot \mathbf{n}_{ij}^n|,$$
$$|\lambda_R(\mathbf{f}, \mathbf{n}_{ij}^n, \mathbf{U}_i^n, \mathbf{U}_j^n) - \mathbf{W}_j^n \cdot \mathbf{n}_{ij}^n|).$$

Conservation and invariant domain property

Theorem (GPSY (2015))

- *The total mass* $\sum_{i\in\{1:I\}} \mathfrak{m}_i^n \mathbf{U}_i^n$ *is conserved.*

Provided CFL condition, $(1 - 2\frac{\Delta t}{\mathfrak{m}_i^n}|D_{ii}|) \geq 0$.

- *Local invariance:* $\mathbf{U}_i^{n+1} \in Conv\{\overline{\mathbf{U}}(\mathbf{U}_i^n, \mathbf{U}_j^n) \mid j \in \mathcal{I}(S_i)\}$.
- *Global invariance. Let A be a convex invariant set, assume $\mathbf{U}_0 \in A$, then $\mathbf{U}_i^{n+1} \in A$ for all $n \geq 0$. The scheme preserves all the convex invariant sets.*
- *Discrete entropy inequality for any entropy pair* $(\eta, \mathbf{q})$

$$\frac{1}{\Delta t}\left(\mathfrak{m}_i^{n+1}\eta(\mathbf{U}_i^{n+1}) - \mathfrak{m}_i^n\eta(\mathbf{U}_i^n)\right) \leq -\sum_{j\in\mathcal{I}(S_i^n)} d_{ij}^n \eta(\mathbf{U}_j^n)$$
$$-\int_{\mathbb{R}^d} \nabla\cdot\left(\sum_{j\in\mathcal{I}(S_i^n)} (\mathbf{q}(\mathbf{U}_j^n) - \eta(\mathbf{U}_j^n)\mathbf{W}_j^n)\psi_j^n(\mathbf{x})\right)\psi_i^n(\mathbf{x})\,\mathrm{d}\mathbf{x}$$

Corollary (GPSY (2015))

The scheme preserves constant states (Discrete Global Conservation Law (DGCL))

Conservation and invariant domain property

Theorem (GPSY (2015))

- *The total mass* $\sum_{i\in\{1:I\}} \mathrm{m}_i^n \mathbf{U}_i^n$ *is conserved.*

Provided CFL condition, $(1 - 2\frac{\Delta t}{\mathrm{m}_i^n}|D_{ii}|) \geq 0$.

- *Local invariance:* $\mathbf{U}_i^{n+1} \in Conv\{\overline{\mathbf{U}}(\mathbf{U}_i^n, \mathbf{U}_j^n) \mid j \in \mathcal{I}(S_i)\}$.
- *Global invariance. Let A be a convex invariant set, assume $\mathbf{U}_0 \in A$, then $\mathbf{U}_i^{n+1} \in A$ for all $n \geq 0$. The scheme preserves all the convex invariant sets.*
- *Discrete entropy inequality for any entropy pair* $(\eta, \mathbf{q})$

$$\frac{1}{\Delta t}(\mathrm{m}_i^{n+1}\eta(\mathbf{U}_i^{n+1}) - \mathrm{m}_i^n\eta(\mathbf{U}_i^n)) \leq -\sum_{j\in\mathcal{I}(S_i^n)} d_{ij}^n \eta(\mathbf{U}_j^n)$$
$$-\int_{\mathbb{R}^d} \nabla\cdot\Big(\sum_{j\in\mathcal{I}(S_i^n)} (\mathbf{q}(\mathbf{U}_j^n) - \eta(\mathbf{U}_j^n)\mathbf{W}_j^n)\psi_j^n(\mathbf{x})\Big)\psi_i^n(\mathbf{x})\,\mathrm{d}\mathbf{x}$$

Corollary (GPSY (2015))

The scheme preserves constant states (Discrete Global Conservation Law (DGCL))

Conservation and invariant domain property

Theorem (GPSY (2015))

- *The total mass* $\sum_{i\in\{1:I\}} \mathrm{m}_i^n \mathbf{U}_i^n$ *is conserved.*

Provided CFL condition, $(1 - 2\frac{\Delta t}{\mathrm{m}_i^n}|D_{ii}|) \geq 0$.

- *Local invariance:* $\mathbf{U}_i^{n+1} \in Conv\{\overline{\mathbf{U}}(\mathbf{U}_i^n, \mathbf{U}_j^n) \mid j \in \mathcal{I}(S_i)\}$.
- *Global invariance. Let A be a convex invariant set, assume $\mathbf{U}_0 \in A$, then $\mathbf{U}_i^{n+1} \in A$ for all $n \geq 0$. The scheme preserves all the convex invariant sets.*
- *Discrete entropy inequality for any entropy pair* $(\eta, \mathbf{q})$

$$\frac{1}{\Delta t}(\mathrm{m}_i^{n+1}\eta(\mathbf{U}_i^{n+1}) - \mathrm{m}_i^n\eta(\mathbf{U}_i^n)) \leq -\sum_{j\in\mathcal{I}(S_i^n)} d_{ij}^n \eta(\mathbf{U}_j^n)$$
$$- \int_{\mathbb{R}^d} \nabla\cdot\Big(\sum_{j\in\mathcal{I}(S_i^n)} (\mathbf{q}(\mathbf{U}_j^n) - \eta(\mathbf{U}_j^n)\mathbf{W}_j^n)\psi_j^n(\mathbf{x})\Big)\psi_i^n(\mathbf{x})\,\mathrm{d}\mathbf{x}$$

Corollary (GPSY (2015))

The scheme preserves constant states (Discrete Global Conservation Law (DGCL))

Conservation and invariant domain property

Theorem (GPSY (2015))

- *The total mass* $\sum_{i\in\{1:I\}} \mathrm{m}_i^n \mathbf{U}_i^n$ *is conserved.*

Provided CFL condition, $(1 - 2\frac{\Delta t}{\mathrm{m}_i^n}|D_{ii}|) \geq 0$.

- *Local invariance:* $\mathbf{U}_i^{n+1} \in Conv\{\overline{\mathbf{U}}(\mathbf{U}_i^n, \mathbf{U}_j^n) \mid j \in \mathcal{I}(S_i)\}$.
- *Global invariance. Let* A *be a convex invariant set, assume* $\mathbf{U}_0 \in A$*, then* $\mathbf{U}_i^{n+1} \in A$ *for all* $n \geq 0$. *The scheme preserves all the convex invariant sets.*
- *Discrete entropy inequality for any entropy pair* $(\eta, \mathbf{q})$

$$\frac{1}{\Delta t}(\mathrm{m}_i^{n+1}\eta(\mathbf{U}_i^{n+1}) - \mathrm{m}_i^n\eta(\mathbf{U}_i^n)) \leq -\sum_{j\in\mathcal{I}(S_i^n)} d_{ij}^n \eta(\mathbf{U}_j^n)$$
$$-\int_{\mathbb{R}^d} \nabla\cdot\Big(\sum_{j\in\mathcal{I}(S_i^n)} (\mathbf{q}(\mathbf{U}_j^n) - \eta(\mathbf{U}_j^n)\mathbf{W}_j^n)\psi_j^n(\mathbf{x})\Big)\psi_i^n(\mathbf{x})\,\mathrm{d}\mathbf{x}$$

Corollary (GPSY (2015))

The scheme preserves constant states (Discrete Global Conservation Law (DGCL))

Conservation and invariant domain property

Theorem (GPSY (2015))

- *The total mass* $\sum_{i\in\{1:I\}} \mathfrak{m}_i^n \mathbf{U}_i^n$ *is conserved.*

Provided CFL condition, $(1 - 2\frac{\Delta t}{\mathfrak{m}_i^n}|D_{ii}|) \geq 0$.

- *Local invariance:* $\mathbf{U}_i^{n+1} \in Conv\{\overline{\mathbf{U}}(\mathbf{U}_i^n, \mathbf{U}_j^n) \mid j \in \mathcal{I}(S_i)\}$.
- *Global invariance. Let* A *be a convex invariant set, assume* $\mathbf{U}_0 \in A$*, then* $\mathbf{U}_i^{n+1} \in A$ *for all* $n \geq 0$. *The scheme preserves all the convex invariant sets.*
- *Discrete entropy inequality for any entropy pair* $(\eta, \mathbf{q})$

$$\frac{1}{\Delta t}(\mathfrak{m}_i^{n+1}\eta(\mathbf{U}_i^{n+1}) - \mathfrak{m}_i^n\eta(\mathbf{U}_i^n)) \leq -\sum_{j\in\mathcal{I}(S_i^n)} d_{ij}^n \eta(\mathbf{U}_j^n)$$
$$-\int_{\mathbb{R}^d} \nabla\cdot\Big(\sum_{j\in\mathcal{I}(S_i^n)} (\mathbf{q}(\mathbf{U}_j^n) - \eta(\mathbf{U}_j^n)\mathbf{W}_j^n)\psi_j^n(\mathbf{x})\Big)\psi_i^n(\mathbf{x})\,\mathrm{d}\mathbf{x}$$

Corollary (GPSY (2015))

The scheme preserves constant states (Discrete Global Conservation Law (DGCL))

2D Burgers

$$\partial_t u + \nabla\cdot(\tfrac{1}{2}u^2\boldsymbol{\beta}) = 0, \quad u_0(\mathbf{x}) = \mathbb{1}_S, \quad \text{with} \quad \boldsymbol{\beta} := (1,1)^{\mathrm{T}}, \quad S := (0,1)^2$$

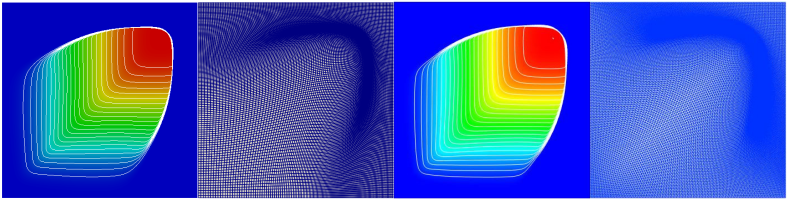


Figure: Burgers equation, 128×128 mesh. Left: $\mathbb{Q}_1$ FEM with 25 contours; Center left: Final $\mathbb{Q}_1$ mesh; Center right: $\mathbb{P}_1$ FEM with 25 contours; Right: Final $\mathbb{P}_1$ mesh.

Nonconvex flux (KPP problem)

$$\partial_t u + \nabla\cdot\mathbf{f}(u) = 0, \quad u_0(\mathbf{x}) = 3.25\pi\mathbb{1}_{\|\mathbf{x}\|_{\ell^2}<1} + 0.25\pi, \quad \text{with} \quad \mathbf{f}(u) = (\sin u, \cos u)^{\mathrm{T}}$$

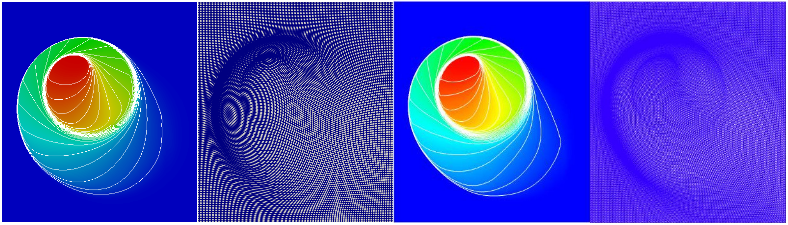


Figure: KPP problem, 128×128 mesh. Left: $\mathbb{Q}_1$ FEM with 25 contours; Center left: Final $\mathbb{Q}_1$ mesh; Center right: $\mathbb{P}_1$ FEM with 25 contours; Right: Final $\mathbb{P}_1$ mesh.

Euler

- Compressible Euler, 2D Noh problem, $\gamma = \frac{5}{3}$
- Initial data

$$\rho_0(\mathbf{x}) = 1.0, \quad \mathbf{u}_0(\mathbf{x}) = -\frac{\mathbf{x}}{\|\mathbf{x}\|_{\ell^2}} \mathbb{1}_{\mathbf{x}\neq\mathbf{0}}, \quad p_0(\mathbf{x}) = 10^{-15}.$$

	$\mathbb{Q}_1$				$\mathbb{P}_1$			
# dofs	L^2-norm		L^1-norm		L^2-norm		L^1-norm	
961	2.60	-	1.44	-	2.89	-	1.71	-
3721	1.81	0.52	8.45E-01	0.77	2.21	0.39	1.09	0.64
14641	1.16	0.64	4.21E-01	1.01	1.42	0.64	5.15E-01	1.08
58081	7.66E-01	0.60	2.10E-01	0.99	9.39E-01	0.59	2.60E-01	0.99
231361	5.21E-01	0.56	1.06E-01	0.98	6.33E-01	0.57	1.28E-01	1.02

Table: Noh problem, convergence test, $T = 0.6$, $CFL = 0.2$

Euler

- Compressible Euler, 2D Noh problem, $\gamma = \frac{5}{3}$
- Initial data

$$\rho_0(\mathbf{x}) = 1.0, \quad \mathbf{u}_0(\mathbf{x}) = -\frac{\mathbf{x}}{\|\mathbf{x}\|_{\ell^2}} \mathbb{1}_{\mathbf{x}\neq\mathbf{0}}, \quad p_0(\mathbf{x}) = 10^{-15}.$$

	$\mathbb{Q}_1$				$\mathbb{P}_1$			
# dofs	L^2-norm		L^1-norm		L^2-norm		L^1-norm	
961	2.60	-	1.44	-	2.89	-	1.71	-
3721	1.81	0.52	8.45E-01	0.77	2.21	0.39	1.09	0.64
14641	1.16	0.64	4.21E-01	1.01	1.42	0.64	5.15E-01	1.08
58081	7.66E-01	0.60	2.10E-01	0.99	9.39E-01	0.59	2.60E-01	0.99
231361	5.21E-01	0.56	1.06E-01	0.98	6.33E-01	0.57	1.28E-01	1.02

Table: Noh problem, convergence test, $T = 0.6$, $CFL = 0.2$

Compressible Euler, 2D Noh problem, $\gamma = \frac{5}{3}$

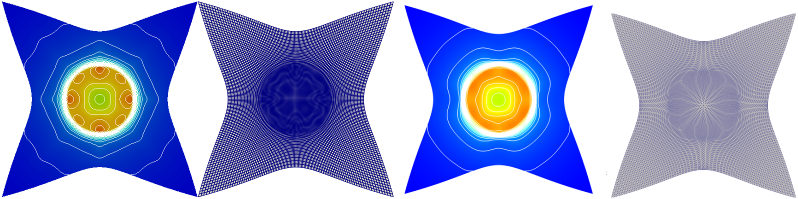


Figure: Noh problem at $t = 0.6$, 96×96 mesh. From left to right: density field with $\mathbb{Q}_1$ approximation (25 contour lines); mesh with $\mathbb{Q}_1$ approximation; density field with $\mathbb{P}_1$ approximation (25 contour lines); mesh with $\mathbb{P}_1$ approximation.

Compressible Euler, 3D Noh problem, $\gamma = \frac{5}{3}$

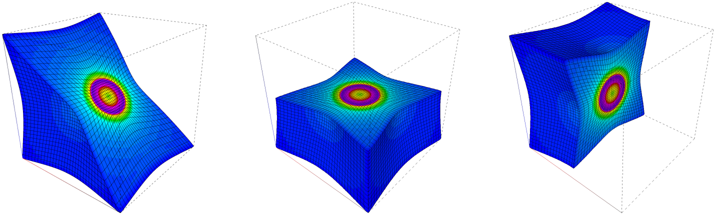


Figure: Density cuts for the 3D Noh problem at $t = 0.6$.

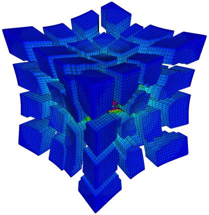


Figure: 3D Noh problem at $t = 0.6$. 64 MPI tasks division.

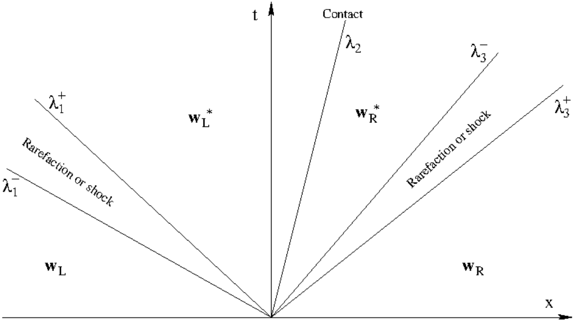
Maximum wave speed

Hyperbolic systems

How to compute local viscosity?

- $d_{ij}^n := 2\lambda_{\max}(\mathbf{f}, \mathbf{n}_{ij}, \mathbf{U}_i, \mathbf{U}_j)\|\mathbf{c}_{ij}\|_{\ell^2}$, for $j \neq i$.
- $\lambda_{\max}(\mathbf{f}, \mathbf{n}_{ij}, \mathbf{U}_i, \mathbf{U}_j)$ is max wave speed for Riemann problem

Riemann fan for Euler, $p = (\gamma - 1)\rho e$



- Structure of the Riemann problem (**Lax (1957), Bressan (2000), Toro (2009)**).
- Waves 1 and 3 are genuinely nonlinear (either shock or rarefaction)
- Wave 2 is linearly degenerate (contact)
- $\mathbf{w}_L = (\rho_L, u_L, p_L)$, $\mathbf{w}_L^* = (\rho_L^*, u^*, p^*)$, $\mathbf{w}_R^* = (\rho_R^*, u^*, p^*)$, $\mathbf{w}_R = (\rho_R, u_R, p_R)$,

Maximum wave speed bound

Euler system, $p = (\gamma - 1)\rho e$

- Given the states U_L and U_R, we have

$$\lambda_1 = u_L - a_L \left(1 + \frac{(p^* - p_L)_+}{p_L}\frac{\gamma+1}{2\gamma}\right)^{\frac{1}{2}} < \lambda_3 = u_R + a_R \left(1 + \frac{(p^* - p_R)_+}{p_R}\frac{\gamma+1}{2\gamma}\right)^{\frac{1}{2}}$$

  where p^* is the pressure of the intermediate state.

- Then and define

$$\lambda_{\max}(\mathbf{U}_L, \mathbf{U}_R) = \max(|\lambda_1|, |\lambda_3|).$$

- In practice we just need a good upper bound of p^*: $\overline{p}^* \geq p^*$. Then

$$\lambda_{\max}(\mathbf{U}_L, \mathbf{U}_R) = \max(|\lambda_1(\overline{p}^*)|, |\lambda_3(\overline{p}^*)|).$$

Maximum wave speed bound

Euler system, $p = (\gamma - 1)\rho e$

- Given the states U_L and U_R, we have

$$\lambda_1 = u_L - a_L \left(1 + \frac{(p^* - p_L)_+}{p_L}\frac{\gamma+1}{2\gamma}\right)^{\frac{1}{2}} < \lambda_3 = u_R + a_R \left(1 + \frac{(p^* - p_R)_+}{p_R}\frac{\gamma+1}{2\gamma}\right)^{\frac{1}{2}}$$

where p^* is the pressure of the intermediate state.

- Then and define

$$\lambda_{\max}(\mathbf{U}_L, \mathbf{U}_R) = \max(|\lambda_1|, |\lambda_3|).$$

- In practice we just need a good upper bound of p^*: $\overline{p}^* \geq p^*$. Then

$$\lambda_{\max}(\mathbf{U}_L, \mathbf{U}_R) = \max(|\lambda_1(\overline{p}^*)|, |\lambda_3(\overline{p}^*)|).$$

Maximum wave speed bound

- To avoid computing p^*, it is a common practice to estimate $\lambda_{\max}$ by $\max(|u_L| + a_L, |u_R| + a_R)$
- This estimate is inaccurate and can be wrong.

Maximum wave speed bound

- Counter-example 1: 1-wave and the 3-wave are both shocks **Toro 2009, §4.3.3**

ρ_L	ρ_R	u_L	u_R	p_L	p_R
5.99924	5.99242	19.5975	-6.19633	460.894	46.0950

- $\lambda_{\max} \approx 12.25$ but $\max(|u_L| + a_L, |u_R| + a_R) \approx 29.97$, large overestimation

Maximum wave speed bound

- Counter-example 2: 1-wave is a shock and the 3-wave is an expansion

ρ_L	ρ_R	u_L	u_R	p_L	p_R
0.01	1000	0	0	0.01	1000

- $\lambda_{\text{max}} \approx 5.227$ but $\max(|u_L| + a_L, |u_R| + a_R) \approx 1.183$, large underestimation

Definition of $\tilde{p}^*$

- Let $\tilde{p}^*$ be the zero of ϕ_R, then

$$\tilde{p}^* = \left(\frac{a_L + a_R - \frac{\gamma-1}{2}(u_R - u_L)}{a_L p_L^{-\frac{\gamma-1}{2\gamma}} + a_R p_R^{-\frac{\gamma-1}{2\gamma}}} \right)^{\frac{2\gamma}{\gamma-1}}$$

Lemma (GP (2016))

We have $p^* < \tilde{p}^*$ in the physical range of γ, $1 < \gamma \leq \frac{5}{3}$.

- $\tilde{p}^*$ is an upper bound on p^*.
- $\min(p_L, p_R) \leq p^* \leq \tilde{p}^*$ (starting guess for cubic Newton alg., **GP (2016)**)

Conclusions

Continuous finite elements

- Continuous FE are viable tools to solve hyperbolic systems.
- Continuous FE are viable alternatives to DG and FV.
- Continuous FE are easy to implement and parallelize.
- Exa-scale computing will need simple, robust, methods.

Current and future work

- Convergence analysis, error estimates beyond first-order.
- Extension to DG.
- Extension of BBZ to higher-order polynomials (order 3 and higher).
- Extension of BBZ to systems (Shallow water, Euler).
- Extension to equations with source terms (Radiative transport, Radiative hydrodynamics).

Conclusions

Continuous finite elements

- Continuous FE are viable tools to solve hyperbolic systems.
- Continuous FE are viable alternatives to DG and FV.
- Continuous FE are easy to implement and parallelize.
- Exa-scale computing will need simple, robust, methods.

Current and future work

- Convergence analysis, error estimates beyond first-order.
- Extension to DG.
- Extension of BBZ to higher-order polynomials (order 3 and higher).
- Extension of BBZ to systems (Shallow water, Euler).
- Extension to equations with source terms (Radiative transport, Radiative hydrodynamics).

Conclusions

Continuous finite elements

- Continuous FE are viable tools to solve hyperbolic systems.
- Continuous FE are viable alternatives to DG and FV.
- Continuous FE are easy to implement and parallelize.
- Exa-scale computing will need simple, robust, methods.

Current and future work

- Convergence analysis, error estimates beyond first-order.
- Extension to DG.
- Extension of BBZ to higher-order polynomials (order 3 and higher).
- Extension of BBZ to systems (Shallow water, Euler).
- Extension to equations with source terms (Radiative transport, Radiative hydrodynamics).

Conclusions

Continuous finite elements

- Continuous FE are viable tools to solve hyperbolic systems.
- Continuous FE are viable alternatives to DG and FV.
- Continuous FE are easy to implement and parallelize.
- Exa-scale computing will need simple, robust, methods.

Current and future work

- Convergence analysis, error estimates beyond first-order.
- Extension to DG.
- Extension of BBZ to higher-order polynomials (order 3 and higher).
- Extension of BBZ to systems (Shallow water, Euler).
- Extension to equations with source terms (Radiative transport, Radiative hydrodynamics).

Conclusions

Continuous finite elements

- Continuous FE are viable tools to solve hyperbolic systems.
- Continuous FE are viable alternatives to DG and FV.
- Continuous FE are easy to implement and parallelize.
- Exa-scale computing will need simple, robust, methods.

Current and future work

- Convergence analysis, error estimates beyond first-order.
- Extension to DG.
- Extension of BBZ to higher-order polynomials (order 3 and higher).
- Extension of BBZ to systems (Shallow water, Euler).
- Extension to equations with source terms (Radiative transport, Radiative hydrodynamics).

Conclusions

Continuous finite elements

- Continuous FE are viable tools to solve hyperbolic systems.
- Continuous FE are viable alternatives to DG and FV.
- Continuous FE are easy to implement and parallelize.
- Exa-scale computing will need simple, robust, methods.

Current and future work

- Convergence analysis, error estimates beyond first-order.
- Extension to DG.
- Extension of BBZ to higher-order polynomials (order 3 and higher).
- Extension of BBZ to systems (Shallow water, Euler).
- Extension to equations with source terms (Radiative transport, Radiative hydrodynamics).

Conclusions

Continuous finite elements

- Continuous FE are viable tools to solve hyperbolic systems.
- Continuous FE are viable alternatives to DG and FV.
- Continuous FE are easy to implement and parallelize.
- Exa-scale computing will need simple, robust, methods.

Current and future work

- Convergence analysis, error estimates beyond first-order.
- Extension to DG.
- Extension of BBZ to higher-order polynomials (order 3 and higher).
- Extension of BBZ to systems (Shallow water, Euler).
- Extension to equations with source terms (Radiative transport, Radiative hydrodynamics).

Conclusions

Continuous finite elements

- Continuous FE are viable tools to solve hyperbolic systems.
- Continuous FE are viable alternatives to DG and FV.
- Continuous FE are easy to implement and parallelize.
- Exa-scale computing will need simple, robust, methods.

Current and future work

- Convergence analysis, error estimates beyond first-order.
- Extension to DG.
- Extension of BBZ to higher-order polynomials (order 3 and higher).
- Extension of BBZ to systems (Shallow water, Euler).
- Extension to equations with source terms (Radiative transport, Radiative hydrodynamics).

Conclusions

Continuous finite elements

- Continuous FE are viable tools to solve hyperbolic systems.
- Continuous FE are viable alternatives to DG and FV.
- Continuous FE are easy to implement and parallelize.
- Exa-scale computing will need simple, robust, methods.

Current and future work

- Convergence analysis, error estimates beyond first-order.
- Extension to DG.
- Extension of BBZ to higher-order polynomials (order 3 and higher).
- Extension of BBZ to systems (Shallow water, Euler).
- Extension to equations with source terms (Radiative transport, Radiative hydrodynamics).