Discontinuous Galerkin method on 2-d Cartesian grids

Praveen. C
praveen@tifrbng.res.in

Tata Institute of Fundamental Research
Center for Applicable Mathematics
Bangalore 560065
http://praveen.tifrbng.res.in

NPDE-TCA Workshop on FEM for Navier-Stokes Equations
SERC, IISc
12 September, 2014

Euler equations

Consider the system of hyperbolic conservation laws

$$\frac{\partial \boldsymbol{w}}{\partial t} + \sum_{\alpha=1}^{d} \frac{\partial \boldsymbol{f}_\alpha}{\partial x_\alpha} = 0$$

where

- $\boldsymbol{w} \in \mathcal{U}_{\mathrm{ad}} \subset \mathbb{R}^m$ are the conserved variables
- $\boldsymbol{f}_\alpha : \mathcal{U}_{\mathrm{ad}} \to \mathbb{R}^m, \quad \alpha = 1, \ldots, d$ are the Cartesian components of the flux

For the Euler equations in 2-d we have $m = 4$ and the physically admissible set of states is given by

$$\mathcal{U}_{\mathrm{ad}} = \left\{ \boldsymbol{w} \in \mathbb{R}^4 : w_1 > 0, \quad w_4 - \frac{w_2^2 + w_3^2}{2w_1} > 0 \right\}$$

where

$$\boldsymbol{w} = \begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ w_4 \end{bmatrix} = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ E \end{bmatrix}, \qquad E = \frac{p}{\gamma - 1} + \frac{1}{2}\rho(u^2 + v^2)$$

Euler equations

and the flux is given by

$$\boldsymbol{f}_1 = \begin{bmatrix} \rho u \\ p + \rho u^2 \\ \rho u v \\ (E+p)u \end{bmatrix}, \qquad \boldsymbol{f}_2 = \begin{bmatrix} \rho v \\ \rho u v \\ p + \rho v^2 \\ (E+p)v \end{bmatrix}$$

Hyperbolicity: For any unit vector $\boldsymbol{n} = (n_1, n_2)$ the flux Jacobian

$$A(\boldsymbol{w}, \boldsymbol{n}) = \boldsymbol{f}_1'(\boldsymbol{w}) n_1 + \boldsymbol{f}_2'(\boldsymbol{w}) n_2$$

has real eigenvalues and complete set of eigenvectors. For Euler equations in 2-D, the eigenvalues are

$$u_n - c, \quad u_n, \quad u_n, \quad u_n + c$$

where

$$u_n = u n_1 + v n_2, \qquad c = \sqrt{\frac{\gamma p}{\rho}}$$

Left and right eigenvectors of Euler flux Jacobian

The right eigenvector matrices of the flux Jacobians $\frac{\partial \boldsymbol{f}_1}{\partial \boldsymbol{w}}$, $\frac{\partial \boldsymbol{f}_2}{\partial \boldsymbol{w}}$ are respectively

$$\mathcal{R}_x = \begin{bmatrix} 1 & 0 & 1 & 1 \\ u & 0 & u+c & u-c \\ v & -1 & v & v \\ k & -v & h+cu & h-cu \end{bmatrix}, \qquad \mathcal{R}_y = \begin{bmatrix} 1 & 0 & 1 & 1 \\ u & 1 & u & u \\ v & 0 & v+c & v-c \\ k & u & h+cv & h-cv \end{bmatrix}$$

The left eigenvector matrices are

$$\mathcal{L}_x = \begin{bmatrix} 1-\frac{\phi}{c^2} & \gamma_1\frac{u}{c^2} & \gamma_1\frac{v}{c^2} & -\frac{\gamma_1}{c^2} \\ v & 0 & -1 & 0 \\ \beta(\phi - cu) & \beta(c-\gamma_1 u) & -\beta\gamma_1 v & \beta\gamma_1 \\ \beta(\phi + cu) & -\beta(c+\gamma_1 u) & -\beta\gamma_1 v & \beta\gamma_1 \end{bmatrix}$$

$$\mathcal{L}_y = \begin{bmatrix} 1-\frac{\phi}{c^2} & \gamma_1\frac{u}{c^2} & \gamma_1\frac{v}{c^2} & -\frac{\gamma_1}{c^2} \\ -u & 1 & 0 & 0 \\ \beta(\phi - cv) & -\beta\gamma_1 u & \beta(c-\gamma_1 v) & \beta\gamma_1 \\ \beta(\phi + cv) & -\beta\gamma_1 u & -\beta(c+\gamma_1 v) & \beta\gamma_1 \end{bmatrix}$$

Left and right eigenvectors of Euler flux Jacobian

where

$$\gamma_1 = \gamma - 1, \quad k = \frac{1}{2}(u^2 + v^2), \quad \phi = \gamma_1 k, \quad \beta = \frac{1}{2c^2}$$

$$h = \frac{c^2}{\gamma - 1} + k$$

These eigenvectors are orthonormal

$$\mathcal{R}_x \mathcal{L}_x = I = \mathcal{R}_y \mathcal{L}_y$$

Legendre polynomials

Legendre polynomials are obtained as the solution of Legendre's differential equation

$$\frac{\mathrm{d}}{\mathrm{d}\xi}\left[(1-\xi^2)\frac{\mathrm{d}}{\mathrm{d}\xi}P_n(\xi)\right] + n(n+1)P_n(\xi) = 0, \qquad n = 0, 1, 2, \ldots$$

A few of them are listed below, also see Fig. 1

$$\begin{array}{ll}
P_0(\xi) = 1 & P_1(\xi) = \xi \\
P_2(\xi) = \frac{1}{2}(3\xi^2 - 1) & P_3(\xi) = \frac{1}{2}(5\xi^3 - 3\xi) \\
P_4(\xi) = \frac{1}{8}(35\xi^4 - 30\xi^2 + 3) & P_5(\xi) = \frac{1}{8}(63\xi^5 - 70\xi^3 + 15\xi)
\end{array}$$

Legendre polynomials

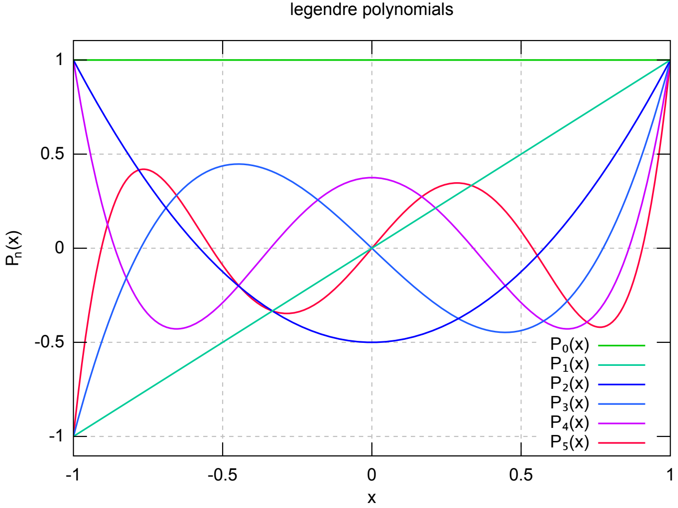


Figure: A few legendre polynomials

Legendre polynomials

A good way to compute them is by the following recursion relation

$$(n+1)P_{n+1}(\xi) = (2n+1)\xi P_n(\xi) - nP_{n-1}(\xi), \qquad n = 1, 2, \ldots$$

We notice that P_n is a polynomial of degree n. The Legendre polynomials have the orthogonality property

$$\int_{-1}^{+1} P_j(\xi)P_k(\xi)\mathrm{d}\xi = \begin{cases} 0 & \text{if } j \neq k \\ \frac{2}{2j+1} & \text{if } j = k \end{cases}$$

Another useful property is

$$P_n(1) = 1, \qquad P_n(-1) = (-1)^j, \qquad n = 0, 1, 2, \ldots$$

The derivatives of P_n can be computed from

$$P_0'(\xi) = 0, \qquad \frac{\xi^2 - 1}{n} P_n'(\xi) = nP_n(\xi) - P_{n-1}(\xi), \qquad \xi \in (-1, +1)$$

Legendre polynomials

Let us also define the scaled Legendre functions

$$\tilde{P}_n(\xi) = \sqrt{2n+1}P_n(\xi), \qquad n = 0, 1, 2, \dots$$

for which we have

$$\int_{-1}^{+1} \tilde{P}_j(\xi)\tilde{P}_k(\xi)\mathrm{d}\xi = \begin{cases} 0 & \text{if } j \neq k \\ 2 & \text{if } j = k \end{cases}$$

Gauss quadrature

To integrate a function $f : [-1, +1] \to \mathbb{R}$, the Gauss rule of N points is

$$\int_{-1}^{+1} f(x)\mathrm{d}x \approx \sum_{r=1}^{N} f(\xi_r^N)\omega_r^N$$

where

$$\{\xi_r^N\}_r \subset (-1, +1) \text{ are the Gauss quadrature nodes}$$

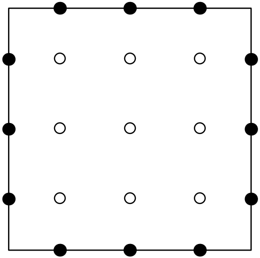
$$\{\omega_r^N\}_r \text{ are the corresponding weights}$$

The N point Gauss quadrature is exact for polynomials of degree $2N - 1$.

To integrate a 2-d function $f : [-1, +1]^2 \to \mathbb{R}$ we use the tensor product of the N Gauss points

$$\int_{-1}^{+1}\int_{-1}^{+1} f(x, y)\mathrm{d}x\mathrm{d}y \approx \sum_{r=1}^{N}\sum_{s=1}^{N} f(\xi_r^N, \xi_s^N)\omega_r^N\omega_s^N$$

Gauss quadrature



$\circ$ cell quadrature points

$\bullet$ face quadrature points

Gauss quadrature points for degree $k = 2$

From a programming point of view, one may want to arrange the Gauss points in a 1-d array indexed by a single integer q; there is a unique mapping from $q \to (r, s)$. Then we can write the quadrature rule as

$$\int_{-1}^{+1} \int_{-1}^{+1} f(x, y) \mathrm{d}x \mathrm{d}y \approx \sum_{q=1}^{N^2} f(\vec{\xi}_q^N) \tilde{\omega}_q^N$$

where

$$\vec{\xi}_q^N = (\xi_r^N, \xi_s^N), \qquad \tilde{\omega}_q^N = \omega_r^N \omega_s^N$$

Basis functions in 2-D

Let

$$\mathbb{P}_k = \text{space of polynomials on } [-1,+1]\times[-1,+1] \text{ of degree at most } k$$

We will form a basis for this space using scaled Legendre polynomials $\tilde{P}_n$ as follows

$$\{\phi_j(\xi,\eta)\}_{j=1}^{N(k)} = \begin{cases} \tilde{P}_0(\xi)\tilde{P}_0(\eta), \quad \tilde{P}_1(\xi)\tilde{P}_0(\eta), \quad \ldots, \quad \tilde{P}_{k-1}(\xi)\tilde{P}_0(\eta), \quad \tilde{P}_k(\xi)\tilde{P}_0(\eta) \\ \tilde{P}_0(\xi)\tilde{P}_1(\eta), \quad \tilde{P}_1(\xi)\tilde{P}_1(\eta), \quad \ldots, \quad \tilde{P}_{k-1}(\xi)\tilde{P}_1(\eta) \\ \vdots \\ \tilde{P}_0(\xi)\tilde{P}_{k-1}(\eta), \quad \tilde{P}_1(\xi)\tilde{P}_{k-1}(\eta) \\ \tilde{P}_0(\xi)\tilde{P}_k(\eta) \end{cases}$$

where

$$N(k) = \frac{1}{2}(k+1)(k+2)$$

DG solution representation

Let

$$\mathcal{T}_h = \text{triangulation of the domain } \Omega \text{ into disjoint rectangular cells}$$

Consider cell $K \in \mathcal{T}_h$ whose size is $\Delta x_K, \Delta y_K$ and center (x_K, y_K); the i'th component of the solution $\boldsymbol{w}$ is represented inside K as

$$\vec{x} = (x, y) \in K : \qquad w_i(x, y, t) = \sum_{j=1}^{N(k)} w_{i,j}^K(t) \phi_j^K(x, y)$$

where

$$\phi_j^K(x, y) = \phi_j(\xi, \eta), \qquad \xi = \frac{x - x_K}{\frac{1}{2}\Delta x_K}, \quad \eta = \frac{y - y_K}{\frac{1}{2}\Delta y_K}$$

The basis functions ϕ_j^K are also orthogonal

$$\int_K \phi_j^K \phi_l^K \mathrm{d}\vec{x} = |K| \delta_{jl} = \Delta x_K \Delta y_K \delta_{jl}$$

DG solution representation

where $\mathrm{d}\vec{x} = \mathrm{d}x\mathrm{d}y$ is the 2-d measure. We will sometimes also write the solution in terms of the scaled variables

$$(x, y) \in K : \qquad w_i(\xi, \eta, t) = \sum_{j=1}^{N(k)} w_{i,j}^K(t)\phi_j(\xi, \eta)$$

Note that since

$$\phi_1(\xi, \eta) = \tilde{P}_0(\xi)\tilde{P}_0(\eta) \equiv 1$$

then $w_{i,1}^K$ is the cell average value of w_i on cell K

$$w_{i,1}^K = \frac{1}{|K|}\int_K w_i \mathrm{d}\vec{x}$$

DG formulation

To derive the DG scheme on cell K we multiply the i'th conservation law by the basis function ϕ_l^K

$$\int_K \left(\frac{\partial w_i}{\partial t} + \sum_\alpha \frac{\partial f_{\alpha,i}}{\partial x_\alpha} \right) \phi_l^K \mathrm{d}\vec{x} = 0$$

Integrate by parts on the flux divergence term

$$\frac{\mathrm{d}}{\mathrm{d}t} \int_K w_i \phi_l^K \mathrm{d}\vec{x} - \sum_\alpha \int_K f_{\alpha,i} \frac{\partial \phi_l^K}{\partial x_\alpha} \mathrm{d}\vec{x} + \int_{\partial K} \sum_\alpha f_{\alpha,i} n_\alpha \phi_l^K \mathrm{d}s = 0$$

Since the solution is discontinuous across ∂K, we introduce a numerical flux function $\hat{\boldsymbol{f}}(\boldsymbol{w}^-, \boldsymbol{w}^+, n)$ for the interface flux leading to the semi-discrete DG scheme

$$\frac{\mathrm{d}}{\mathrm{d}t} \int_K w_i \phi_l^K \mathrm{d}\vec{x} - \sum_\alpha \int_K f_{\alpha,i} \frac{\partial \phi_l^K}{\partial x_\alpha} \mathrm{d}\vec{x} + \int_{\partial K} \hat{f}_i \phi_l^K \mathrm{d}s = 0 \tag{1}$$

DG formulation

In order to derive equations more suitable for numerical implementation, we substitute the solution representation in terms of basis functions to get

$$\sum_{j=1}^{N(k)} \frac{\mathrm{d}w_{i,j}^K}{\mathrm{d}t} \int_K \phi_j^K \phi_l^K \mathrm{d}\vec{x} - \sum_\alpha \int_K f_{\alpha,i} \frac{\partial \phi_l^K}{\partial x_\alpha} \mathrm{d}\vec{x} + \int_{\partial K} \hat{f}_i \phi_l^K \mathrm{d}s = 0$$

Finally, using the orthogonality of the basis functions

$$|K| \frac{\mathrm{d}w_{i,l}^K}{\mathrm{d}t} - \sum_\alpha \int_K f_{\alpha,i} \frac{\partial \phi_l^K}{\partial x_\alpha} \mathrm{d}\vec{x} + \int_{\partial K} \hat{f}_i \phi_l^K \mathrm{d}s = 0$$

To approximate the integrals we transform the cell and edge to the reference domains of $[-1,+1]^2$ and $[-1,+1]$ respectively. For example,

$$\begin{aligned} \int_K f_{1,i} \frac{\partial \phi_l^K}{\partial x_1} \mathrm{d}\vec{x} &= \int_{[-1,+1]^2} f_{1,i} \left(\frac{2}{\Delta x_K} \frac{\partial \phi_l}{\partial \xi} \right) \left(\frac{1}{4} \Delta x_K \Delta y_K \mathrm{d}\vec{\xi} \right) \\ &= \frac{1}{2} \Delta y_K \int_{[-1,+1]^2} f_{1,i} \frac{\partial \phi_l}{\partial \xi} \mathrm{d}\vec{\xi} \end{aligned}$$

DG formulation

The cell integrals are approximated by tensor product of Gauss quadrature rule with $k+1$ points while the edge integral is approximated with a Gauss rule of $k+1$ points (see Fig. 2), leading to

$$
\begin{aligned}
|K|\frac{\mathrm{d}w_{i,l}^K}{\mathrm{d}t} \quad & - \quad \frac{1}{2}\Delta y_K \sum_{q=1}^{M_k} f_{1,i}(\boldsymbol{w}(\vec{\xi}_q,t))\frac{\partial \phi_l}{\partial \xi}(\vec{\xi}_q)\tilde{\omega}_q^{k+1} \\
& - \quad \frac{1}{2}\Delta x_K \sum_{q=1}^{M_k} f_{2,i}(\boldsymbol{w}(\vec{\xi}_q,t))\frac{\partial \phi_l}{\partial \eta}(\vec{\xi}_q)\tilde{\omega}_q^{k+1} \\
& + \quad \frac{1}{2}\sum_{e\in\partial K}|e|\sum_{q=1}^{k+1}\hat{f}_i(\vec{\xi}_q^e,t)\phi_l(\vec{\xi}_q^e)\omega_q^{k+1} = 0 \\
& \qquad \text{for } i=1,\ldots,4, \quad l=1,\ldots,N(k)
\end{aligned}
$$

We have a total of $M_k = (k+1)^2$ quadrature nodes denoted by $\vec{\xi}_q = \vec{\xi}_q^{k+1} = (\xi_r^{k+1},\xi_s^{k+1}), q = 1,\ldots,M_k$

DG formulation

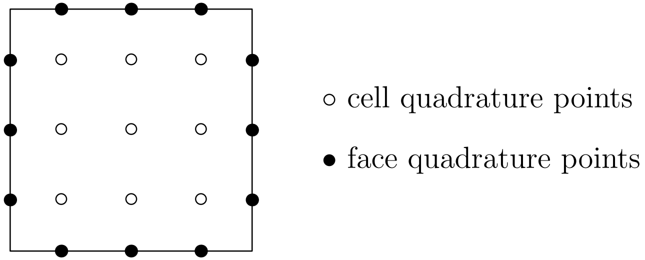


Figure: Gauss quadrature points for degree $k = 2$

Define the vector of degrees of freedom associated to cell K by

$$w^K = [w^K_{1,1}, \dots, w^K_{1,N(k)}, \dots, w^K_{4,1}, \dots, w^K_{4,N(k)}]^\top$$

Then we have an ODE of the form

$$\frac{\mathrm{d}w^K}{\mathrm{d}t} + R_K(w) = 0$$

which will be solved by a Runge-Kutta method.

Remarks

1. For $l = 1$, we have $\phi_1 \equiv 1$ and we obtain the equation for the cell average value
$$|K|\frac{\mathrm{d}w_{i,1}^K}{\mathrm{d}t} + \frac{1}{2}\sum_{e\in\partial K}|e|\sum_{q=1}^{k+1}\hat{f}_i(\vec{\xi}_q^e, t)\phi_1(\vec{\xi}_q^e)\omega_q^{k+1} = 0$$
2. If the cell face lies on the boundary of the domain, then the numerical flux $\hat{\boldsymbol{f}}$ must account for the boundary condition.
3. There are a large number of numerical flux functions that can be used. For the Lax-Friedrich's flux, see appendix (21).
4. To perform the quadature, we need to evaluate the solution $\boldsymbol{w}$ at the quadrature points $\vec{\xi}_q = (\xi_r, \eta_s)$, see appendix (10)
$$w_i(\vec{\xi}_q, t) = \sum_{j=1}^{N(k)} w_{i,j}(t)\phi_j(\vec{\xi}_q)$$
The set of numbers
$$\phi_j(\vec{\xi}_q), \quad j = 1, \ldots, N(k), \quad q = 1, \ldots, M_k$$

Remarks

do not depend on the element K; hence they can be computed and stored in a common location that can be used by all elements.

5. Each basis function ϕ_l is of the form

$$\phi_l(\vec{\xi}) = P_m(\xi)P_n(\eta) \quad \text{for some } m, n \in \{0, 1, \ldots, k\}$$

and hence

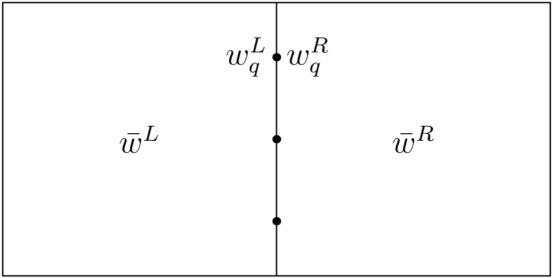
$$\frac{\partial}{\partial \xi}\phi_l(\vec{\xi}) = P_m'(\xi)P_n(\eta), \qquad \frac{\partial}{\partial \eta}\phi_l(\vec{\xi}) = P_m(\xi)P_n'(\eta)$$

The set of numbers

$$\frac{\partial}{\partial \xi}\phi_l(\vec{\xi}_q), \quad \frac{\partial}{\partial \eta}\phi_l(\vec{\xi}_q), \quad l = 1, \ldots, N(k), \quad q = 1, \ldots, M_k$$

do not depend on the element and can be computed and stored once.

Lax-Friedrich's flux



Consider the face between cells L and R as shown in Fig. **??**. The quantities $\boldsymbol{w}_q^L$, $\boldsymbol{w}_q^R$ are the two trace values at the q'th quadrature point and $\bar{\boldsymbol{w}}^L$, $\bar{\boldsymbol{w}}^R$ are the cell average values. The Lax-Friedrich's flux at the q'th quadrature point on this face is given by

$$\hat{\boldsymbol{f}}_q = \frac{1}{2}[\boldsymbol{f}_1(\boldsymbol{w}_q^L) + \boldsymbol{f}_1(\boldsymbol{w}_q^R)] - \frac{1}{2}\lambda_{LR}(\boldsymbol{w}_q^R - \boldsymbol{w}_q^L)$$

where

$$\lambda_{LR} = \max\{\lambda_1(\bar{\boldsymbol{w}}^L), \lambda_1(\bar{\boldsymbol{w}}^R)\}, \qquad \lambda_1(\boldsymbol{w}) = |u(\boldsymbol{w})| + c(\boldsymbol{w})$$

Limiting

High order schemes can generate oscillations in the solutions which are physically incorrect. Limiting tries to eliminate such oscillations. Let $\boldsymbol{w}$ be the solution obtained after applying the RK scheme. The i'th component of solution inside cell K is of the form

$$(x,y) \in K: \qquad w_i = w_{i,1}^K + \underbrace{w_{i,2}^K \phi_2^K + w_{i,3}^K \phi_3^K}_{\text{linear terms}} + \text{higher order terms}$$

The quantities $w_{i,2}^K$, $w_{i,3}^K$ are proportional to the derivatives along x and y directions respectively. Let E_K, W_K, N_K, S_K denote the cells to the east, west, north and south of cell K, see Fig. 3. We first compute the limited derivatives using minmod function[1]

Limiting

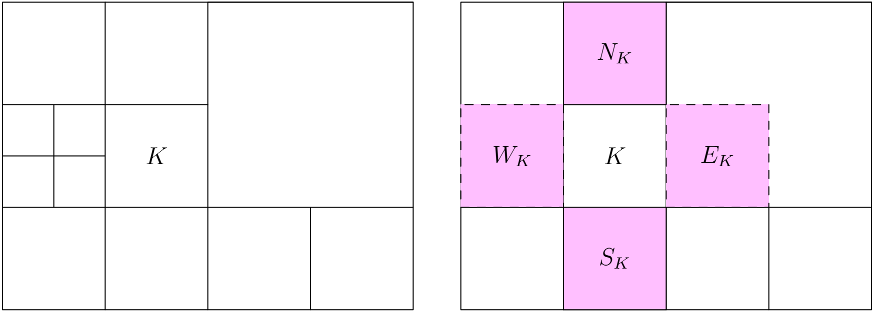


Figure: Definition of neighbours for TVD limiter

Limiting

$$\tilde{w}_{i,2}^K = \frac{1}{\sqrt{3}} \text{minmod}\left(\sqrt{3} w_{i,2}^K, \beta(w_{i,1}^K - w_{i,1}^{W_K}), \beta(w_{i,1}^{E_K} - w_{i,1}^K)\right)$$

$$\tilde{w}_{i,3}^K = \frac{1}{\sqrt{3}} \text{minmod}\left(\sqrt{3} w_{i,3}^K, \beta(w_{i,1}^K - w_{i,1}^{S_K}), \beta(w_{i,1}^{N_K} - w_{i,1}^K)\right)$$

where $\beta \in [1/2, 1]$ and

$$\text{minmod}\,(a, b, c) = \begin{cases} s \min(|a|, |b|, |c|) & s = \text{sign}(a) = \text{sign}(b) = \text{sign}(c) \\ 0 & \text{otherwise} \end{cases}$$

If $\tilde{w}_{i,2}^K = w_{i,2}^K$ and $\tilde{w}_{i,3}^K = w_{i,3}^K$ then the limited solution is same as the original solution

$$\Lambda\Pi_h(w_i) = w_i$$

else the limited solution is an affine solution of the form

$$\Lambda\Pi_h(w_i) = w_{i,1}^K + \tilde{w}_{i,2}^K \phi_2^K + \tilde{w}_{i,3}^K \phi_3^K$$

Note that if the limiter modifies the solution, then it preserves the cell average value. What we have described above is sometimes refered to as the

Limiting

component-wise limiter since we limit each component w_i, $i = 1, \ldots, 4$ individually.

Remark: The parameter β controls the amount of limiting; $\beta = \frac{1}{2}$ leads to a TVD scheme for a scalar problem but is more dissipative. A value of $\beta = 1$ leads to a less restrictive limiter which may give more accurate solutions.

[1] The factors involving $\sqrt{3}$ are necessary because we use scaled Legendre polynomials.

Characteristic limiter

Define

$$\bar{\boldsymbol{w}}^K = \begin{bmatrix} w_{1,1}^K \\ w_{2,1}^K \\ w_{3,1}^K \\ w_{4,1}^K \end{bmatrix}, \qquad \boldsymbol{w}_2^K = \begin{bmatrix} w_{1,2}^K \\ w_{2,2}^K \\ w_{3,2}^K \\ w_{4,2}^K \end{bmatrix}, \qquad \boldsymbol{w}_3^K = \begin{bmatrix} w_{1,3}^K \\ w_{2,3}^K \\ w_{3,3}^K \\ w_{4,3}^K \end{bmatrix}$$

Compute the right eigenvector matrices $\mathcal{R}_x^K, \mathcal{R}_y^K$ and left eigenvector matrices $\mathcal{L}_x^K, \mathcal{L}_y^K$ of the corresponding flux Jacobian based on the cell average value $\bar{\boldsymbol{w}}_K$, see appendix (4). Define

$$\boldsymbol{c}_2^K = \mathcal{L}_x^K \boldsymbol{w}_2^K, \qquad \boldsymbol{c}_3^K = \mathcal{L}_y^K \boldsymbol{w}_3^K$$

Compute limited quantities by applying minmod function component-wise

$$\tilde{\boldsymbol{c}}_2^K = \frac{1}{\sqrt{3}} \mathrm{minmod}\left(\sqrt{3}\boldsymbol{c}_2^K, \beta\mathcal{L}_x^K(\bar{\boldsymbol{w}}^K - \bar{\boldsymbol{w}}^{W_K}), \beta\mathcal{L}_x^K(\bar{\boldsymbol{w}}^{E_K} - \bar{\boldsymbol{w}}^K)\right)$$

$$\tilde{\boldsymbol{c}}_3^K = \frac{1}{\sqrt{3}} \mathrm{minmod}\left(\sqrt{3}\boldsymbol{c}_3^K, \beta\mathcal{L}_y^K(\bar{\boldsymbol{w}}^K - \bar{\boldsymbol{w}}^{S_K}), \beta\mathcal{L}_y^K(\bar{\boldsymbol{w}}^{N_K} - \bar{\boldsymbol{w}}^K)\right)$$

Characteristic limiter

If $\tilde{\boldsymbol{c}}_2^K = \boldsymbol{c}_2^K$ and $\tilde{\boldsymbol{c}}_3^K = \boldsymbol{c}_3^K$ then

$$\Lambda\Pi_h(\boldsymbol{w}) = \boldsymbol{w}$$

else

$$\tilde{\boldsymbol{w}}_2^K = \mathcal{R}_x^K \tilde{\boldsymbol{c}}_2^K, \qquad \tilde{\boldsymbol{w}}_3^K = \mathcal{R}_y^K \tilde{\boldsymbol{c}}_3^K, \qquad \Lambda\Pi_h(\boldsymbol{w}) = \bar{\boldsymbol{w}}^K + \tilde{\boldsymbol{w}}_2^K \phi_2 + \tilde{\boldsymbol{w}}_3^K \phi_3$$

TVB version

The minmod limiter is too strict and leads to clipping of smooth extrema which are identified as shocks by the limiter. In order to avoid such effects, we can use a less strict limiter which involves replacing the minmod function with the following function

$$\mathrm{minmodB}\,(a, b, c) = \begin{cases} a & \text{if } \ |a| < Mh^2 \\ \mathrm{minmod}\,(a, b, c) & \text{otherwise} \end{cases}$$

The parameter M is related to the second derivative of the solution at smooth extrema. This is usually not known and a proper choice of the parameter M has to be done for each problem.

Time integration

After discretizing in space using the DG scheme, we end with a system of coupled ODEs of the the form

$$\frac{\mathrm{d}w}{\mathrm{d}t} + R(w) = 0$$

We will solve them using the 3'rd order SSP Runge-Kutta scheme which is made of three stages:

$$\begin{aligned}
w^{(0)} &= w^n \\
w^{(1)} &= w^{(0)} - \Delta t^n R(w^{(0)}) \\
w^{(2)} &= \frac{3}{4}w^n + \frac{1}{4}\left[w^{(1)} - \Delta t^n R(w^{(1)})\right] \\
w^{(3)} &= \frac{1}{3}w^n + \frac{2}{3}\left[w^{(2)} - \Delta t^n R(w^{(2)})\right] \\
w^{n+1} &= w^{(3)}
\end{aligned}$$

Time integration

The time step is computed based on a CFL condition. For each cell K, compute the local time step

$$\Delta t_K^n = \frac{\text{cfl}}{2k+1} \left(\frac{|u_K^n| + c_K^n}{\Delta x_K} + \frac{|v_K^n| + c_K^n}{\Delta y_K} \right)^{-1}, \qquad 0 < \text{cfl} \le 1$$

where (u_K, v_K) is the velocity and c_K is the sound speed corresponding to the cell average value $\bar{\boldsymbol{w}}^K$. The global time step is

$$\Delta t^n = \min_{K \in \mathcal{T}_h} \Delta t_K^n$$

Setting the initial condition

Given the initial condition for the i'th component of the solution

$$w_i = g_i \qquad \text{at} \qquad t = 0$$

we approximate the initial condition on cell K by an L^2-projection

$$\min_{\{w_{i,j}^K\}_j} \int_K (w_i - g_i)^2 \mathrm{d}\vec{x}$$

which leads to

$$w_{i,l}^K = \frac{1}{|K|} \int_K g_i \phi_l^K \mathrm{d}\vec{x} = \frac{1}{4} \int_{[-1,+1]^2} g_i \phi_l \mathrm{d}\vec{\xi}$$

The integral is approximated by a tensor product Gauss quadrature of $M_k = (k+1)^2$ points

$$w_{i,l}^K = \frac{1}{4} \sum_{q=1}^{M_k} g_i(\vec{x}_q) \phi_l(\vec{\xi}_q) \tilde{\omega}_q^{k+1}, \qquad l = 1, 2, \ldots, N(k)$$

Setting the initial condition

where the real location of quadrature nodes $\vec{\xi}_q = (\xi_r, \eta_s)$ is given by

$$\vec{x}_q = \left(\frac{1}{2} \xi_r^{k+1} \Delta x_K + x_K, \frac{1}{2} \xi_s^{k+1} \Delta y_K + y_K \right)$$

Summary of algorithm

1. Compute projection matrices, quadrature data, etc.
2. Set initial condition by L^2-projection, $t = 0$
3. Apply limiter and positivity limiter (if needed)
4. While $t < T$
   1. Compute time step
   2. For $r = 1, 2, 3$
      1. Compute rhs
      2. Update solution to next RK stage
      3. Apply limiter
      4. Apply positivity limiter
   3. $t = t + \Delta t$

Adaptive grids

The DG scheme can be easily applied on adapted grids. It is usual practice to keep the level difference between any two neighbouring cells to be at most one. When two neighbouring cells are at different level of refinement, then the face quadrature is performed on the smaller faces.

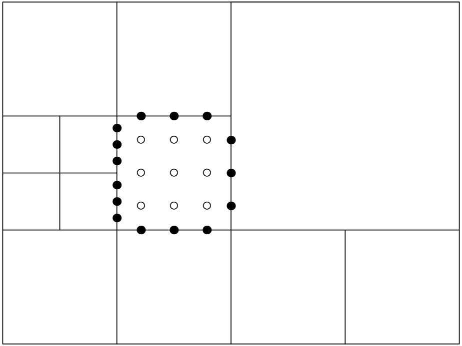


Figure: Example of Gauss quadrature points for degree $k = 2$ on adapted grids

Refinement

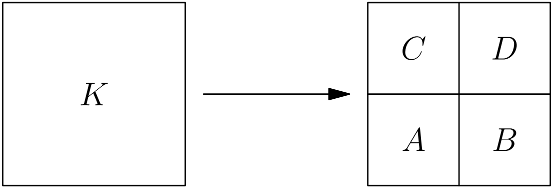


Figure: Refinement of cell K into four cells

When a cell K is divided into four smaller cells, say A, B, C, D, see Fig. 5, then the solution from the parent cell K has to be projected onto the child cells. We have the i'th component of the solution

$$\vec{x} \in K : \qquad w_i^K = \sum_j w_{i,j}^K \phi_j^K$$

and we want to compute

$$\vec{x} \in A : \qquad w_i^A = \sum_j w_{i,j}^A \phi_j^A$$

Refinement

We determine $\{w_{i,j}^A, j = 1, \ldots, N(k)\}$ by performing an L^2-projection, i.e.,

$$\min_{\{w_{i,j}^A\}_j} \int_A \left(w_i^K - w_i^A\right)^2 \mathrm{d}\vec{x}$$

which leads to

$$w_{i,l}^A = \frac{1}{|A|} \int_A w_i^K \phi_l^A \mathrm{d}\vec{x} = \frac{1}{|A|} \sum_{j=1}^{N(k)} w_{i,j}^K \int_A \phi_j^K \phi_l^A \mathrm{d}\vec{x}, \qquad l = 1, \ldots, N(k)$$

This can be written as a matrix vector product

$$w_i^A = P_1 w_i^K, \qquad (P_1)_{jl} = \frac{1}{|A|} \int_A \phi_j^K \phi_l^A \mathrm{d}\vec{x}$$

By transforming the integral on A to the reference cell $[-1, +1] \times [-1 + 1]$, we can rewrite P_1 as

$$(P_1)_{jl} = \frac{1}{4} \int_{-1}^{+1} \int_{-1}^{+1} \phi_j\left((\xi+1)/2, (\eta+1)/2\right) \phi_l(\xi, \eta) \mathrm{d}\xi \mathrm{d}\eta$$

Refinement

which is independent of the element K, A; the above integrals can be computed exactly using a Gauss quadrature of $(k+1)^2$ points. Similarly, the computation of w_i^B, w_i^C, w_i^D can be expressed in terms of matrices P_2, P_3, P_4 respectively, which are independent of the elements.

Coarsening

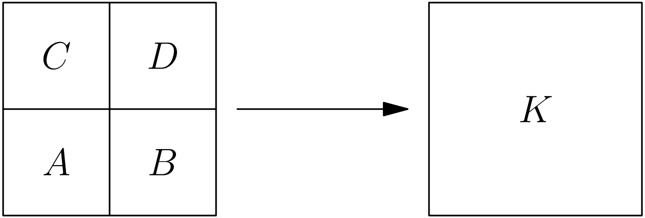


Figure: Coarsening of cells A, B, C, D into cell K

Consider the case when cells A, B, C, D are merged into one cell K, see Fig. 6. We determine the solution on K by performing an L^2-projection of the solutions on A, B, C, D, i.e., we solve the following minimization problem,

$$\min_{\{w_{i,j}^K\}_j} \int_A \left(w_i^K - w_i^A\right)^2 \mathrm{d}\vec{x} + \int_B \left(w_i^K - w_i^B\right)^2 \mathrm{d}\vec{x} + \int_C \left(w_i^K - w_i^C\right)^2 \mathrm{d}\vec{x} + \int_D \left(w_i^K - w_i^D\right)^2 \mathrm{d}\vec{x}$$

Coarsening

which leads to

$$w_{i,l}^K = \frac{1}{|K|} \left(\int_A w_i^A \phi_l^K \mathrm{d}\vec{x} + \int_B w_i^B \phi_l^K \mathrm{d}\vec{x} + \int_C w_i^C \phi_l^K \mathrm{d}\vec{x} + \int_D w_i^D \phi_l^K \mathrm{d}\vec{x} \right)$$
$$\text{for } l = 1, \ldots, N(k)$$

The above equations can be expressed as a matrix-vector product

$$w_i^K = \frac{1}{16} \left(P_1^\top w_i^A + P_2^\top w_i^B + P_3^\top w_i^C + P_4^\top w_i^D \right)$$

where the matrices P_1, P_2, P_3, P_4 are same as those described in the previous section. Moreover we have used the property that

$$|A| = |B| = |C| = |D| = \frac{1}{4}|K|$$