1 – DG Methods for Diffusion Problems

Per-Olof Persson

Department of Mathematics, University of California, Berkeley
Mathematics Department, Lawrence Berkeley National Laboratory

Summer School on Discontinuous Galerkin Methods
International Center for Numerical Methods in Engineering
Barcelona, Spain

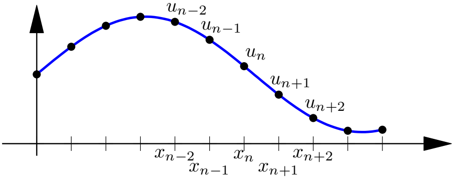
June 11-15, 2012

Outline

The Finite Difference Method (FDM)

- Consider linear convection: $u_t + u_x = 0$ for $x \in [0,1]$, $u(0) = u(1)$
- Approximate u_x point-wise using difference formulas:

$$\frac{d}{dx}u(x_n) \approx \frac{u_{n+1} - u_{n-1}}{2\Delta x}$$



or high-order:

$$\frac{d}{dx}u(x_n) \approx \frac{u_{n+2} - 8u_{n+1} + 8u_{n-1} - u_{n-1}}{12\Delta x}$$

or one-sided (e.g. for stability, “upwinding”):

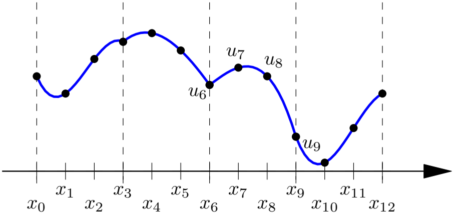
$$\frac{d}{dx}u(x_n) \approx \frac{3u_n - 16u_{n-1} + 36u_{n-2} - 48u_{n-3} + 25u_{n-4}}{25\Delta x}$$

- Simple, efficient, flexible
- Needs *structured neighborhood* of nodes – hard to generalize to unstructured grids in 2-D and 3-D

The Finite Element Method (FEM)

- Discretize domain into *elements* (intervals)
- Seek approximate solution in space of piecewise polynomials $\hat{X}$
- Impose equation weakly: Seek $\hat{u} \in \hat{X}$ such that for all $v \in \hat{X}$:

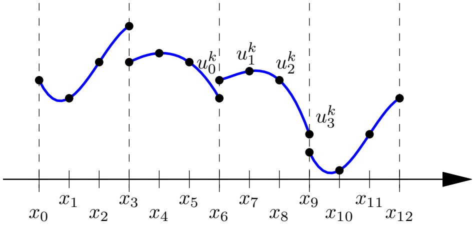
$$\begin{aligned}&\int_0^1 (\hat{u}_t + \hat{u}_x) v \, dx \\ &= \int_0^1 \hat{u}_t v \, dx + \int_0^1 \hat{u}_x v \, dx \\ &= \int_0^1 \hat{u}_t v \, dx - \int_0^1 \hat{u} v_x \, dx = 0\end{aligned}$$

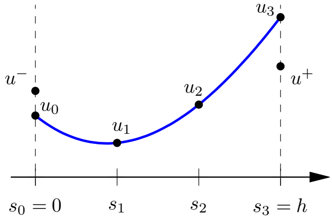


- Leads to semi-discrete system $M\boldsymbol{u}_t + K\boldsymbol{u} = 0$, with element-wise local M, K matrices
- M^{-1} dense $\Longrightarrow$ Explicit methods for $\boldsymbol{u}_t = -M^{-1}K\boldsymbol{u}$ not practical
- Also, unclear how to stabilize by upwinding (but other techniques exist, such as Streamline Upwind Petrov-Galerkin)

The Discontinuous Galerkin Method

- Do not enforce continuity – allow “jumps” between elements





- Galerkin formulation for single element $\kappa = [0, h]$: For all $v \in P^p(\kappa)$,

$$\int_0^h (\hat{u}_t + \hat{u}_x) v \, dx = \int_0^h \hat{u}_t v \, dx + \int_0^h \hat{u}_x v \, dx$$

$$= \int_0^h \hat{u}_t v \, dx - \int_0^h \hat{u} v_x \, dx + \mathcal{U}(u^+, u_p) v(h) - \mathcal{U}(u_0, u^-) v(0)$$

- *Numerical flux function* $\mathcal{U}(u_R, u_L)$ allows for stabilization by high-order upwinding, e.g. $\mathcal{U}(u_R, u_L) = u_L$

The Discontinuous Galerkin Method

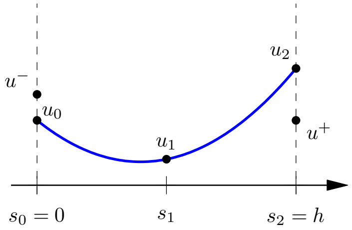
- The DG formulation leads to linear system of equations:

$$M\boldsymbol{u}_t + K\boldsymbol{u} + \begin{pmatrix} -u^- & 0 & \dots & 0 & u_p \end{pmatrix}^T = 0$$

- For example, with $p = 2$:

$$\begin{aligned} \boldsymbol{u}_t &= -M^{-1}K\boldsymbol{u} - M^{-1}\begin{pmatrix} -u^- & 0 & u_2 \end{pmatrix}^T \\ &= \frac{1}{h}\begin{pmatrix} -6 & -4 & 1 \\ 2.5 & 0 & -1 \\ -4 & 4 & -3 \end{pmatrix}\begin{pmatrix} u_0 \\ u_1 \\ u_2 \end{pmatrix} + \frac{1}{h}\begin{pmatrix} 9 \\ -1.5 \\ 3 \end{pmatrix} u^- \end{aligned}$$

- Element-wise local FD-type stencil
- Stabilized, "upwinded" through u^-
- Extends naturally to other PDEs, N-D, unstructured meshes



The DG Scheme – Details of Discretization

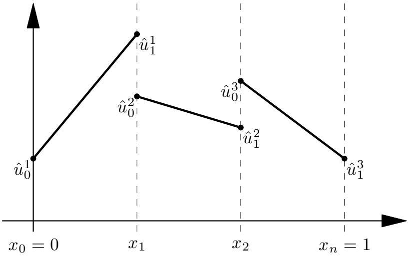
- Consider the 1-D conservation law

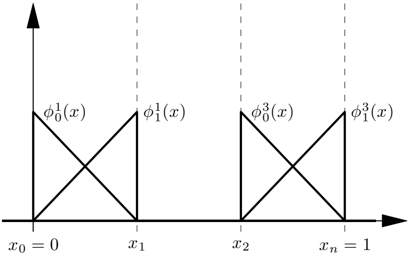
$$\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = 0$$

- Seek a solution in space of piecewise polynomial functions X_h
- Nodal representation with values u_i^k for local node i in element k:

$$u_h(x) = \sum_{k=1}^{n} \sum_{i=0}^{p} u_i^k \phi_i^k(x)$$

- Example, piecewise linear functions ($p = 1$):





The DG Scheme – Details of Discretization

- Galerkin formulation: Find $u_h \in X_h$ such that

$$\int_0^1 \frac{\partial u_h}{\partial t} v\,dx + \int_0^1 \frac{\partial f(u_h)}{\partial x} v\,dx = 0$$

- Set $v = \phi_i^k$ and integrate by parts

$$\int_{x_{k-1}}^{x_k} \frac{\partial u_h}{\partial t} \phi_i^k\,dx + \left[f(u_h(x))\phi_i^k(x)\right]_{x_{k-1}}^{x_k} - \int_{x_{k-1}}^{x_k} f(u_h) \frac{d\phi_i^k}{dx}\,dx = 0$$

- Use a numerical flux function $F(u_R, u_L)$ at the discontinuities

$$\int_{x_{k-1}}^{x_k} \frac{\partial u_h}{\partial t} \phi_i^k\,dx + F(u_0^{k+1}, u_p^k)\phi_i^k(x_k) - F(u_0^k, u_p^{k-1})\phi_i^k(x_k)$$
$$- \int_{x_{k-1}}^{x_k} f(u_h) \frac{d\phi_i^k}{dx}\,dx = 0$$

The DG Scheme – Details of Discretization

- Example: $f(u) = u$, $F(u_R, u_L) = u_L$

$$\int_{x_{k-1}}^{x_k} \frac{\partial}{\partial t} \left(\sum_{k=1}^{n} \sum_{j=0}^{p} u_j^k \phi_j^k(x) \right) \phi_i^k \, dx - \int_{x_{k-1}}^{x_k} \left(\sum_{k=1}^{n} \sum_{j=0}^{p} u_j^k \phi_j^k(x) \right) \frac{d\phi_i^k}{dx} \, dx$$
$$+ u_p^k \phi_i^k(x_k) - u_p^{k-1} \phi_i^k(x_{k-1}) = 0$$

- Rearrange to obtain a linear system of equations

$$M^k \dot{u}^k - C^k u^k + \begin{bmatrix} -u_p^{k-1} & 0 & \cdots & 0 & u_0^k \end{bmatrix}^T = 0$$

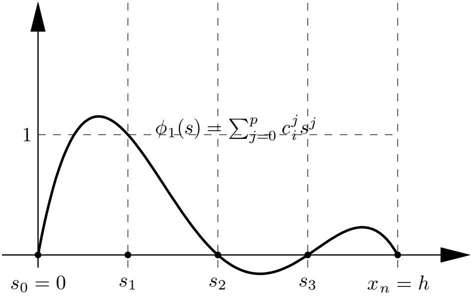
for element k, with elementary matrices

$$M_{ij}^k = \int_{x_{k-1}}^{x_k} \phi_i^k \phi_j^k \, dx \text{ and } C_{ij}^k = \int_{x_{k-1}}^{x_k} \frac{d\phi_i^k}{dx} \phi_j^k \, dx$$

Calculating Elementary Matrices

- Consider an element of degree p, width h, and a nodal basis at the points $s_i = h_i/p$, $i = 0, \ldots, p$
  - For p high (> 4), use Gauss-Lobatto points instead
- Write basis functions in monomial form $\phi_i(s) = \sum_{j=0}^{p} c_i^j s^j$
  - For p high (> 4), use an orthogonal basis instead
- Nodal basis functions are defined by

$$\phi_i(s_k) = \delta_{ij} = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}$$



- Produces a linear system of equations

Calculating Elementary Matrices

- The linear system of equations has the form

$$\begin{pmatrix} 1 & s_0 & s_0^2 & \cdots & s_0^p \\ 1 & s_1 & s_1^2 & \cdots & s_1^p \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & s_p & s_p^2 & \cdots & s_p^p \end{pmatrix} \begin{pmatrix} c_0^0 & c_1^0 & \cdots & c_p^0 \\ c_0^1 & c_1^1 & \cdots & c_p^1 \\ \vdots & \vdots & \ddots & \vdots \\ c_0^p & c_1^p & \cdots & c_p^p \end{pmatrix} = \begin{pmatrix} 1 & & & \\ & 1 & & \\ & & \ddots & \\ & & & 1 \end{pmatrix}$$

  or $VC = I$, which gives the coefficient matrix $C = V^{-1}$

- Use Gaussian quadrature or explicit polynomial integration to compute the elementary matrices

$$M_{ij} = \int_0^h \phi_i(s)\phi_j(s)\, ds$$

$$C_{ij} = \int_0^h \phi_i'(s)\phi_j(s)\, ds$$

Outline

DG for Elliptic Problems – Historical Overview

- Enforcing Dirichlet conditions by penalties
  - Lions (1968), Babuška (1973) – Penalty term
  - Nitsche (1971) – additional terms in bilinear form for consistency
- Interior Penalty (IP) methods
  - Babuška and Zlámal (1973) – enforce C^1-continuity by penalties
  - Wheeler (1978), Arnold (1979) – Nitsche's method for spaces of discontinuous piecewise polynomials
- DG methods
  - Bassi and Rebay (1997) – apply RKDG to unknown and its gradient
  - Cockburn and Shu (1998) – generalized the ideas, the LDG method
- Unification
  - Arnold, Brezzi, Cockburn, Marini (2000,2002) – showed that most methods fit in a unified framework by choosing appropriate numerical fluxes

Second-order Equations

- Consider the 1-D Poisson equation

$$-\frac{d^2u}{dx^2} = f(x) \quad \text{in } [0,1]$$

  with homogeneous Dirichlet conditions $u(0) = u(1) = 0$

- Standard Continuous Galerkin FEM would consider the space $X_{h,0}$ of continuous piecewise polynomials satisfying the Dirichlet conditions, and solve for $u_h \in X_{h,0}$ s.t.

$$\int_0^1 -\frac{d^2u_h}{dx^2} v\, dx = \int_0^1 \frac{du_h}{dx}\frac{dv}{dx}\, dx - \left[\frac{du_h}{dx} v\right]_0^1 = \int_0^1 \frac{du_h}{dx}\frac{dv}{dx}\, dx = \int_0^1 f\, v\, dx$$

  for all $v \in X_{h,0}$

- With discontinuous functions, appropriate numerical fluxes must be chosen at all element boundaries

The 1-D Poisson Equation

- To define a DG discretization, first split into first order system:

$$-\sigma' = f(x), \quad u' = \sigma$$

- Multiply by test functions v, τ, integrate over an element, and integrate by parts to obtain the weak form

$$\int_{x_k}^{x_{k+1}} f(x)\, v\, dx = \int_{x_k}^{x_{k+1}} -\sigma' v\, dx = \int_{x_k}^{x_{k+1}} \sigma v'\, dx - [\hat{\sigma} v]_0^1$$

$$\int_{x_k}^{x_{k+1}} \sigma\, \tau\, dx = \int_{x_k}^{x_{k+1}} u'\, \tau\, dx = -\int_{x_k}^{x_{k+1}} u\, \tau'\, dx + [\hat{u}\tau]_0^1$$

- Galerkin formulation: Find $u_h, \sigma_h \in X_h$ s.t. for all elements k

$$\int_{x_k}^{x_{k+1}} \sigma_h v'\, dx = \int_{x_k}^{x_{k+1}} f(x)\, v\, dx + [\hat{\sigma}(u_h, \sigma_h) v]_0^1\,, \qquad \forall v \in X_h$$

$$\int_{x_k}^{x_{k+1}} \sigma_h\, \tau\, dx = -\int_{x_k}^{x_{k+1}} u_h\, \tau'\, dx + [\hat{u}(u_h)\tau]_0^1\,, \qquad \forall \tau \in X_h$$

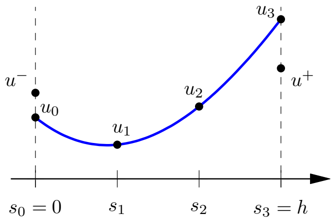
- Remains only to define the *numerical fluxes* $\hat{u}(u_h), \hat{\sigma}(u_h, \sigma_h)$

The BR1 Fluxes

- The BR1 fluxes:

$$\hat{u} = \{u_h\}, \qquad \hat{\sigma} = \{\sigma_h\}$$

  where $\{\cdot\}$ is the *averaging* operator



- For example, with notation according to the figure:

$$\hat{u}(0) = (u^- + u_0)/2 \text{ and } \hat{u}(h) = (u_3 + u^+)/2$$
$$\hat{\sigma}(0) = (\sigma^- + \sigma_0)/2 \text{ and } \hat{\sigma}(h) = (\sigma_3 + \sigma^+)/2$$

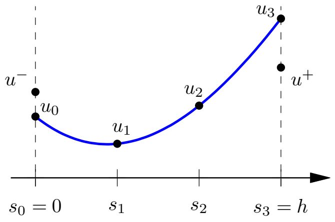
- Simple, intuitive (no preference to direction in equation)
- However, unstable and non-compact stencil

Interior Penalty (IP)

- In the *interior penalty* method, we set

$$\hat{u} = \{u_h\}$$
$$\hat{\sigma} = \{\nabla u_h\} + C_{11}[\![u_h]\!]$$

for some $C_{11} > 0$, where $\{\cdot\}$ is the *averaging* operator and $[\![\cdot]\!]$ is the *jump* operator



- For example, with notation according to the figure:

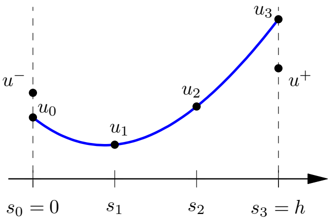
$$\hat{u}(0) = (u^- + u_0)/2 \text{ and } \hat{u}(h) = (u_3 + u^+)/2$$
$$\hat{\sigma}(0) = (u_h'|_{x=0^-} + u_h'|_{x=0^+})/2 + C_{11}(u^- - u_0)$$
$$\hat{\sigma}(h) = (u_h'|_{x=h^-} + u_h'|_{x=h^+})/2 + C_{11}(u_3 - u^+)$$

- Convergent with optimal order of accuracy
- However, C_{11} is problem dependent, introduces stiffness

The Local Discontinuous Galerkin (LDG) Method

- In the *LDG* method, we set

$$\hat{u} = \{u_h\} + C_{12}[\![u_h]\!]$$
$$\hat{\sigma} = \{\sigma_h\} + C_{11}[\![u_h]\!] - C_{12}[\![\sigma_h]\!]$$



- For the special cases $C_{11} = 0$ (minimal dissipation LDG) and $C_{12} = 1/2$ we get a simple upwind/downwind structure
- For example, with notation according to the figure:

$$\hat{u}(0) = u_0 \text{ and } \hat{u}(h) = u^+$$
$$\hat{\sigma}(0) = \sigma^- \text{ and } \hat{\sigma}(h) = \sigma_3$$

- Simple and general
- Convergent with optimal order of accuracy
- However, in general a non-compact stencil in higher dimensions

Outline

Higher Space Dimensions

- From Arnold, Brezzi, Cockburn, Marini (2002)
- Model problem:

$$-\Delta u = f \quad \text{in } \Omega, \qquad u = 0 \quad \text{on } \partial\Omega$$

- Rewrite as first-order system

$$\sigma = \nabla u, \quad -\nabla \cdot \sigma = f \quad \text{in } \Omega, \quad u = 0 \quad \text{on } \partial\Omega$$

- Multiply by test functions τ, v, integrate over element K, integrate by parts $\Rightarrow$ weak formulation:

$$\int_K \sigma \cdot \tau \, dx = -\int_K u \nabla \cdot \tau \, dx + \int_{\partial K} u \, n_K \cdot \tau \, ds$$
$$\int_K \sigma \cdot \nabla v \, dx = \int_K f \, v \, dx + \int_{\partial K} \sigma \cdot n_K \, v \, ds$$

Higher Space Dimensions

- Introduce finite element spaces for triangulation $\mathcal{T}_h = \{K\}$:

$$V_h := \{v \in L^2(\Omega) : v|_K \in \mathcal{P}_p(K) \quad \forall K \in \mathcal{T}_h \}$$
$$\Sigma_h := \{\tau \in [L^2(\Omega)]^2 : \tau|_K \in [\mathcal{P}_p(K)]^2 \quad \forall K \in \mathcal{T}_h \}$$

- The flux formulation: Find $u_h \in V_h$ and $\sigma_h \in \Sigma_h$ s.t.

$$\begin{aligned} \int_K \sigma_h \cdot \tau \, dx &= -\int_K u_h \nabla \cdot \tau \, dx + \int_{\partial K} \hat{u}_K \, n_K \cdot \tau \, ds, && \forall \tau \in [\mathcal{P}_p(K)]^2 \\ \int_K \sigma_h \cdot \nabla v \, dx &= \int_K f \, v \, dx + \int_{\partial K} \hat{\sigma}_K \cdot n_K \, v \, ds, && \forall v \in \mathcal{P}_p(K) \end{aligned}$$

  for all elements $K \in \mathcal{T}_h$

- Need to define the *numerical fluxes* $\hat{u}_K$ and $\hat{\sigma}_K$

Higher Space Dimensions

- Denote the union of the element edges Γ, the interior edges $\Gamma^0 := \Gamma \backslash \partial\Omega$, and the trace space $T(\Gamma) := \prod_{K \in \mathcal{T}_h} L^2(\partial K)$
- For an interior edge e, with unit normal vectors n_1, n_2 define the *jump* and *average* of $q \in T(\Gamma)$ by

$$\{q\} = \frac{1}{2}(q_1 + q_2), \quad [\![q]\!] = q_1 n_1 + q_2 n_2$$

  and for $\sigma \in [T(\Gamma)]^2$ by

$$\{\sigma\} = \frac{1}{2}(q_1 + q_2), \quad [\![\sigma]\!] = \sigma_1 \cdot n_1 + \sigma_2 \cdot n_2$$

- For boundary edges, set

$$[\![q]\!] = qn, \quad \{\sigma\} = \sigma$$

- Note: The jump of a scalar is vector valued (in the normal direction), the jump of a vector is scalar

The Primal Formulation

- Summing over all K, the flux formulation can be written

$$\int_\Omega \sigma_h \cdot \tau \, dx = -\int_\Omega u_h \nabla_h \cdot \tau \, dx + \int_\Gamma [\![\hat{u}]\!] \cdot \{\tau\} \, ds + \int_{\Gamma^0} \{\hat{u}\} [\![\tau]\!] \, ds$$

$$\int_\Omega \sigma_h \cdot \nabla_h v \, dx - \int_\Gamma \{\hat{\sigma}\} \cdot [\![v]\!] \, ds - \int_{\Gamma^0} [\![\hat{\sigma}]\!] \{v\} = \int_\Omega f \, v \, dx$$

- With some manipulations, σ_h can be expressed as

$$\sigma_h = \sigma_h(u_h) := \nabla_h u_h - r([\![\hat{u}(u_h) - u_h]\!]) - l(\{\hat{u}(u_h) - u_h\})$$

where r, l are *lifting operators* defined by

$$\int_\Omega r(\phi) \cdot \tau \, dx = -\int_\Gamma \phi \cdot \{\tau\} \, ds, \qquad \int_\Omega l(q) \cdot \tau \, dx = -\int_{\Gamma^0} q [\![\tau]\!] \, ds \quad \forall \tau \in \Sigma_h$$

The Primal Formulation

- This leads to the *primal formulation*

$$B_h(u_h, v) = \int_\Omega f\, v\, dx \qquad \forall v \in V_h$$

with the *primal form*

$$B_h(u_h, v) = \int_\Omega \nabla_h u_h \cdot \nabla_h v\, dx + \int_\Gamma ([\![\hat{u} - u_h]\!] \cdot \{\nabla_h v\} - \{\hat{\sigma}\} \cdot [\![v]\!])\, ds$$
$$+ \int_{\Gamma^0} (\{\hat{u} - u_h\}[\![\nabla_h v]\!] - [\![\hat{\sigma}]\!]\{v\})\, ds$$

- Standard FEM formulation without σ_h
- In implementations it is often easier to work directly with the flux formulation

Consistency and Conservation

- The numerical fluxes are *consistent* if for smooth functions v

$$\hat{u}(v) = v|_\Gamma, \qquad \hat{\sigma}(v, \nabla v) = \nabla v|_\Gamma$$

  $\Rightarrow$ consistency of the primal formulation and Galerkin orthogonality $B_h(u - u_h, v) = 0$, $\forall v \in V_h$

- The numerical fluxes are *conservative* if $\hat{u}(\cdot)$ and $\hat{\sigma}(\cdot, \cdot)$ are single-valued on Γ

  $\Rightarrow$ adjoint consistency of the primal form

Some DG Methods

- Some of the most important schemes are summarized below:

Method	$\hat{u}_K$	$\hat{\sigma}_K$	Stable
Bassi-Rebay (BR1)	$\{u_h\}$	$\{\sigma_h\}$	$\times$
Bassi-Rebay (BR2)	$\{u_h\}$	$\{\nabla_h u_h\} - \alpha_r([\![u_h]\!])$	$\inf_e \eta_e > 3$
Interior Penalty	$\{u_h\}$	$\{\nabla_h u_h\} + C_{11}[\![u_h]\!]$	$C_{11} > C_{11}^*$
LDG	$\{u_h\} + \boldsymbol{C}_{12} \cdot [\![u_h]\!]$	$\{\sigma_h\} + C_{11}[\![u_h]\!] - \boldsymbol{C}_{12}[\![\sigma_h]\!]$	$C_{11} > 0$

- $\alpha_r(\phi) = -\eta_e\{r_e(\phi)\}$ on an edge e, where r_e is defined by

$$\int_\Omega r_e(\varphi) \cdot \tau \, dx = -\int_e \varphi \cdot \{\tau\} \, ds, \qquad \forall \tau \in \Sigma_h, \varphi \in [L^1(e)]^2$$

- C_{11}^* is mesh dependent, explicit form derived by Shahbazi (2005)
- The methods BR2, IP, and LDG are all commonly used

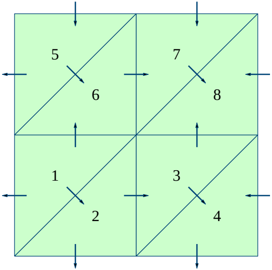
Outline

The LDG Method

- In the LDG method, we use the fluxes

$$\hat{\sigma}_K = \{\sigma_h\} + C_{11}[\![u_h]\!] - \boldsymbol{C}_{12}[\![\sigma_h]\!]$$

$$\hat{u}_K = \{u_h\} + \boldsymbol{C}_{12} \cdot [\![u_h]\!]$$



- Here, $C_{11} > 0$ (or zero for the *minimal dissipation LDG method*, Cockburn and Dong 2007)

- An important special case for $\boldsymbol{C}_{12}$ is the choice

$$\boldsymbol{C}_{12} = \frac{1}{2}(S_{K^+}^{K^-1} n^+ + S_{K^-}^{K^+} n^-)$$

where $S_{K^+}^{K^-} \in \{0, 1\}$ is a *switch* for the edge shared by K^- and K^+

- This leads to a simple upwind/downwind scheme:

$$\hat{\sigma}_K = C_{11}[\![u_h]\!] + \begin{cases} \sigma_h^+ & \text{if } S_{K^+}^{K^-} = 0 \\ \sigma_h^- & \text{if } S_{K^+}^{K^-} = 1 \end{cases}, \qquad \hat{u}_K = \begin{cases} u_h^- & \text{if } S_{K^+}^{K^-} = 0 \\ u_h^+ & \text{if } S_{K^+}^{K^-} = 1 \end{cases}$$

LDG Switch Functions

- *Natural switch*: Order the elements, let N_K be the index of element K, and set

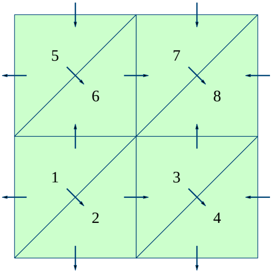
$$S_{K^+}^{K^-} = 1 \text{ if } N_{K^+} > N_{K^-}, \quad 0 \text{ otherwise.}$$

  Simple, leads to beneficial matrix structure, but unstable if $C_{11} = 0$ in the original LDG method

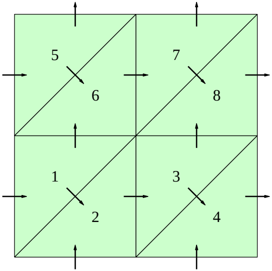
- *Consistent switch*: For example define any constant vector β and set

$$S_{K^+}^{K^-} = 1 \text{ if } n^+ \cdot \beta > 0, \quad 0 \text{ otherwise.}$$

- In general, any choice of switch leads to a *non-compact stencil*



Natural switch



Consistent switch

Outline

The Compact DG (CDG) Method

- To address the non-compactness of the LDG method and its sensitivity to the switch, Peraire and Persson developed the *Compact DG* method (2008)
- Recall the original LDG fluxes:

$$\hat{\sigma}_K = \{\sigma_h\} + C_{11}[\![u_h]\!] - \boldsymbol{C}_{12}[\![\sigma_h]\!]$$
$$\hat{u}_K = \{u_h\} + \boldsymbol{C}_{12} \cdot [\![u_h]\!]$$

- Now, introduce the *edge fluxes* σ_h^e on edge e by

$$\int_K \sigma_h^e \cdot \tau \, dx = -\int_K u_h \nabla \cdot \tau \, dx + \int_{\partial K} \hat{u}_K^e n_K \cdot \tau \, ds, \quad \forall \tau \in [\mathcal{P}_p(K)]^2$$

where

$$\hat{u}_K^e = \begin{cases} \hat{u}_K & \text{on edge } e\text{, as defined above} \\ u_h & \text{otherwise} \end{cases}$$

The Compact DG (CDG) Method

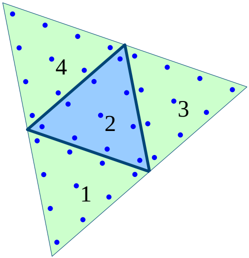
- The numerical fluxes for CDG are then simply given by

$$\hat{\sigma}_K^e = \{\sigma_h^e\} + C_{11}[\![u_h]\!] - \boldsymbol{C}_{12}[\![\sigma_h^e]\!]$$

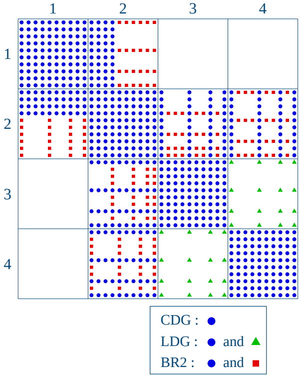
  on edge e

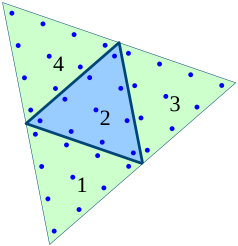
- The modification eliminates the non-compact terms in the primal form, while retaining all the good properties of the LDG scheme

- In addition, better stability properties are observed with in particular less sensitivity to the choice of switch function

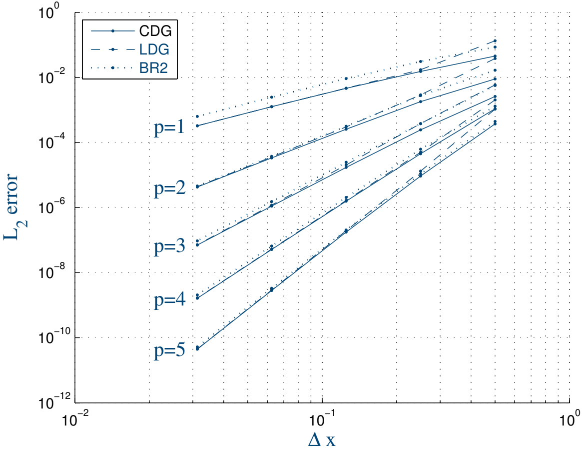


The CDG Method – Summary





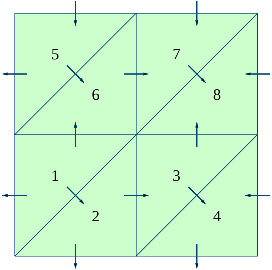
- Element-wise compact stencil
- Less connectivities than LDG/BR2/IP
- More accurate than LDG and BR2



Outline

Switches and Null-space Dimensions

- Unlike the LDG scheme, the CDG scheme appears to be stable for $C_{11} = 0$ and an *inconsistent switch* such as highest element number
- Simple test [Sherwin et al 05]: Poisson problem, periodic boundary conditions, expected nullspace dimension = 1

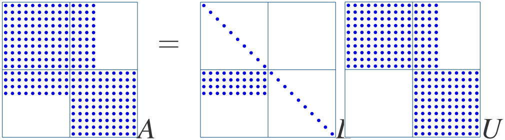


Nullspace dimension

Polynomial order p		1	2	3	4	5	6	7
Consistent switch	CDG	1	1	1	1	1	1	1
	LDG	1	1	1	1	1	1	1
Natural switch	CDG	1	1	1	1	1	1	1
	LDG	3	4	5	6	7	8	9

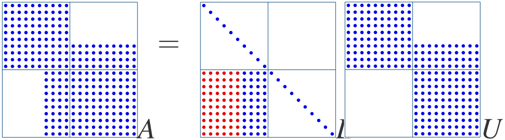
ILU and Switch Orientation

- Orientation of lower-triangular blocks important for ILU sparsity
- Take advantage of CDG's insensitivity to orientation



Switch 1:

Same LU storage



Switch 2:

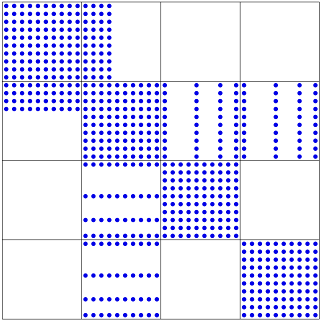
More LU storage

Switches and Null-space Dimensions

- No additional non-zeros in block-ILU(0) factorization using CDG
- Dense lower-triangular blocks using BR2 / IP

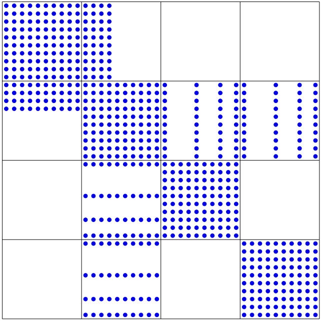
CDG

Stiffness Matrix



640 non-zeros

Block ILU(0)



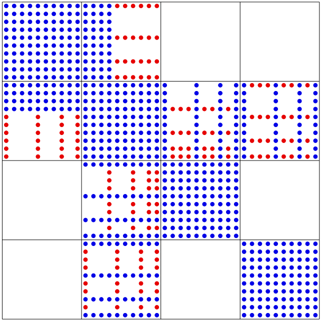
640 non-zeros

Switches and Null-space Dimensions

- No additional non-zeros in block-ILU(0) factorization using CDG
- Dense lower-triangular blocks using BR2 / IP

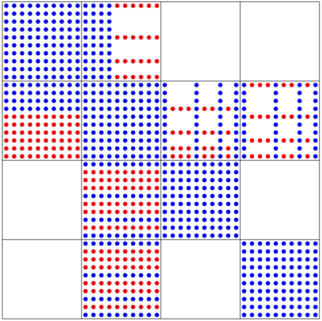
BR2 / IP

Stiffness Matrix



784 non-zeros

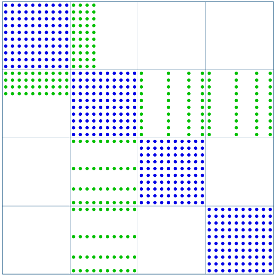
Block ILU(0)



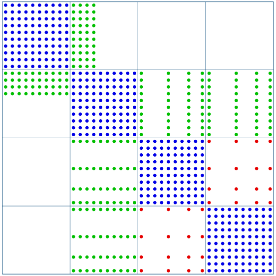
892 non-zeros

Matrix Representation

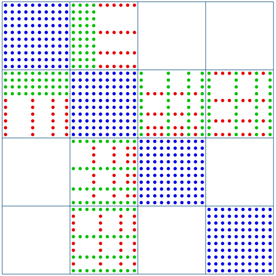
- Block matrix representation *fundamental for high performance*
  - Solver algorithms based on blocks
  - Up to 10 times higher performance with optimized BLAS
- Compact stencil $\Longrightarrow$ Matrix structure given by mesh connectivities
- Hard to store LDG/BR2/IP efficiently



CDG – 2 arrays



LDG – 3 arrays + struct



BR2 / IP – 3 arrays