A mimetic finite difference method for anisotropic elliptic problems

Sven Marnach

IPVS, University of Stuttgart

Mimetic finite difference methods in general

Mimetic finite difference (MFD) methods are applicable to a broad range of partial differential equations. The general idea is to split the strong formulation of a PDE into a system of first-order PDEs and to discretize this system by giving discrete analoga of the usual first-order continuum differential operators gradient, curl and divergence. These analoga are constructed by *mimicking* important properties of the continuum operators like

- conservation laws
- operator symmetries
- kernels of operators
- basic theorems of vector calculus (Gauß, Stokes, Green)

MFD methods for elliptic problems

The diffusion problem is stated as a system of two first order PDEs

$$\begin{aligned} \operatorname{div} \mathbf{v} &= f && \text{in } \Omega \subset \mathbb{R}^2, \\ \mathbf{v} &= -\mathbb{K}\nabla u && \text{in } \Omega, \\ u &= \bar{u} && \text{on } \Gamma_D, \\ \mathbf{v} \cdot n &= -g && \text{on } \Gamma_N. \end{aligned}$$

We construct discrete analoga for the divergence and flux operators

$$\operatorname{div}: H(\operatorname{div}, \Omega) \to L^2(\Omega), \qquad -\mathbb{K}\nabla u: H^1(\Omega) \to \left(L^2(\Omega)\right)^2$$

that respect the followig continuum operator properties:

- Divergence theorem

$$\int_V \operatorname{div} \mathbf{v} = \int_{\partial V} \mathbf{v} \cdot \mathbf{n} \qquad \text{for } V \subset \Omega \tag{1}$$

  needed for local mass conservation

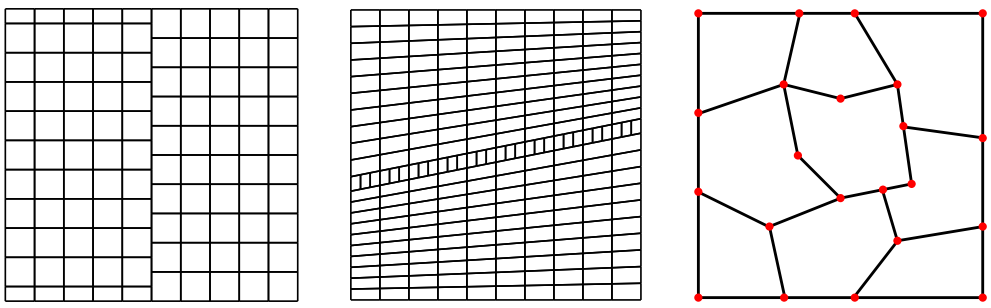
- Partial integration formula

$$\int_V u \ \operatorname{div} \mathbf{w} + \int_V \mathbf{w} \cdot \nabla u = \int_{\partial V} u \ \mathbf{w} \cdot \mathbf{n} \qquad \text{for all } u, \ \mathbf{w} \text{and } V \subset \Omega \tag{2}$$

  needed for the symmetry of the Laplacian

Triangulation of the domain

One of the strength of the MFD method is that it works on nearly all computational grids. There are no constraints on the number of vertices of a cell, nor on the angles. The grids are always considered to be conforming, but hanging nodes can simply be treated by considering them an additional vertex with an interior angle of 180°.



The software framework DUNE

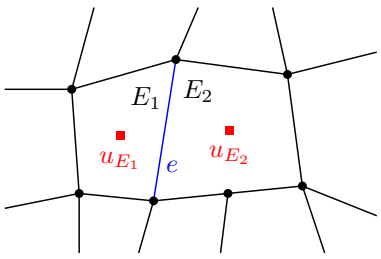
The mimetic finite difference method was implemented within the DUNE framework. DUNE is a numerical template library written in C++. It provides tools for the solution of partial differential equations with grid-based methods. DUNE defines interfaces for different components involved in the solution process, for example computational grids or linear solvers. Using the ideas of generic programming, these interfaces can be implemented by slim wrappers around existing libraries or by new code specifically written for DUNE. As a consequence, the same code can be used for a variety of grid managers, linear solvers etc.
DUNE consists of the following modules:

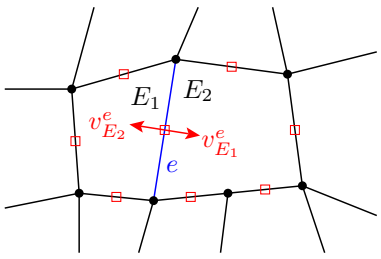
- `dune-common` contains the basic classes used by all DUNE-modules. It provides some infrastructural classes for debugging and exception handling as well as a library to handle dense matrices and vectors of fixed size.
- `dune-grid` defines a very elaborate and efficient interface for non-conforming, hierarchically nested, multi-element-type, parallel grids in arbitrary space dimensions. Various powerful grid managers are wrapped for this interface, including UG, ALBERTA and ALUGrid. An efficient structured parallel grid, the YaspGrid, is also included, as well as a generic data output for the visualization library VTK.
- `dune-istl`, the Iterative Solver Template Library, provides generic sparse matrix/vector classes and a variety of solvers based on these classes. A special feature is the use of templates to exploit the recursive block structure of finite element matrices at compile time. Available solvers include Krylov methods, (block-) incomplete decompositions and aggregation-based algebraic multigrid.

Discrete function spaces

To construct discrete analoga of the continuum differential operators, we first need discrete function spaces.

- The space Q_h of *discrete scalar functions* consists of functions which are constant on each cell. The degrees of freedom of the functions are located at the barycenters of the cells. The value of a function $u \in Q_h$ on the cell E is denoted by u_E.
- The elements of the space X_h of *discrete vector functions* are given by their normal components on the faces of the grid. The degrees of freedom are located at the barycenters of the faces. The value of the normal component of a function $\mathbf{v} \in X_h$ on the face e with respect to the outer normal vector of cell E is denoted by v_E^e. For a face e shared by the cells E_1 and E_2 we have the compatibility condition $v_{E_1}^e = -v_{E_2}^e$.





Discrete operators

The *discrete divergence operator* $\operatorname{div}_h : X_h \to Q_h$ is defined to comply with the divergence theorem (1) on each cell,

$$(\operatorname{div}_h \mathbf{v})_E = \frac{1}{|E|} \sum_{e \subset \partial E} |e| \ v_E^e.$$

A discrete analogon of the partial integration formula (2) reads (assuming homogeneous Dirichlet boundary conditions)

$$\sum_E |E| \ u_E \operatorname{div}_h \mathbf{w} = \sum_E [\mathbf{w}, (-\mathbb{K}\nabla)_h u]_E \qquad \text{for all } u, \ \mathbf{w}$$

with some scalar product $[\mathbf{v}, \mathbf{w}]_E$ on each cell E. This relation can be used as an implicit definition of the *discrete flux operator* $(-\mathbb{K}\nabla)_h$, giving rise to a linear system in saddle point form with one pressure unknown on each cell and one flux unknown on each face. By the usual hybridization process, this system can be transformed into a positive definite system with pressure unknowns on cells and faces.
The hybridized linear system is given by the equations

$$\operatorname{div}_h(-\mathbb{K}\nabla)_h u = \frac{1}{|E|} \sum_{e \subset \partial E} |e| \sum_{f \subset \partial E} \mathbb{W}_E^{e,f} |f| (u_E - u^f) = q_E$$

for each cell E and the equations

$$\left((-\mathbb{K}\nabla)_h u\right)_{E_1}^e + \left((-\mathbb{K}\nabla)_h u\right)_{E_2}^e = \sum_{f \subset \partial E} \left\{ \mathbb{W}_{E_1}^{e,f} |f| (u_{E_1} - u^f) + \mathbb{W}_{E_2}^{e,f} |f| (u_{E_2} - u^f) \right\} = 0$$

for each face e. The construction of the matrix $\mathbb{W}_E$ is described in the next box.

Construction of the matrix $\mathbb{W}_E$

For a cell E, let k_E denote the number of faces of E. Define the $k_E \times 2$ matrices $\mathbb{R}$ and $\mathbb{N}$ by

$$\mathbb{R}_{e,i} = \int_e (x_i - x_{E,i}), \quad \mathbb{N}_{e,i} = \mathbf{e}_i \cdot n_E^e, \tag{3}$$

where e ranges over the faces of E, $i = 1, 2$, x_i is the i-th coordinate function, x_E is the center of mass of E and $\mathbf{e}_i$ is the unit vector in the direction of the i-th axis. We construct a $k_E \times k_E$ matrix $\mathbb{W}_E$ according to algorithm 1 in [3]. In short, that means the following:

1. Orthonormalize the columns of the matrix $\mathbb{R}$ and call the resulting matrix $\tilde{\mathbb{R}}$.
2. Set $\mathbb{D} = \mathbb{I} - \tilde{\mathbb{R}}\tilde{\mathbb{R}}^{\mathrm{T}}$, where $\mathbb{I}$ denotes the $k_E \times k_E$ unit matrix.
3. Define $\mathbb{W}_E = \frac{1}{|E|} \mathbb{N}\mathbb{K}\mathbb{N}^{\mathrm{T}} + \omega\mathbb{D}$, where ω is an arbitrary positive real number and $\mathbb{K}$ is simply evaluated at the cell center x_E.

As numerical evidence indicates, a good choice for ω is

$$\omega = 2 \cdot \frac{\operatorname{trace} \mathbb{K}}{|E|} \cdot \frac{d_{\max}}{d_{\min}}, \tag{4}$$

where $d_{\max}$ and $d_{\min}$ denote the maximum and minimum distances of the center of mass to the centers of mass of the neighbouring cells.

References

[1] P. Bastian, M. Blatt, A. Dedner, C. Engwer, R. Klöfkorn, R. Kornhuber, M. Ohlberger, O. Sander: *A Generic Grid Interface for Parallel and Adaptive Scientific Computing. Part II: Implementation and Tests in DUNE*, to appear in *Computing*, 2007

[2] F. Brezzi, K. Lipnikov, M. Shashkov: *Convergence of the mimetic finite difference method for diffusion problems on polyhedral meshes*, SIAM Journal on Numerical Analysis **43**, num. 5, 2005

[3] F. Brezzi, K. Lipnikov, V. Simoncini: *A family of mimetic finite difference methods on polygonal and polyhedral meshes*, Mathematical Models and Methods in Applied Sciences **15**, num. 10, 2005

[4] A. Favorskii, A. Samarskii, M. J. Shashkov, V. Tishkin: *Operational Finite-Difference Schemes*, Differential Equations **17**, 1981