Mimetic Finite Difference Methods

Franco Brezzi

Dipartimento di Matematica, Università di Pavia, Italy

IMATI - CNR, Pavia, Italy

From joint works with K. Lipnikov, M. Shashkov, and V. Simoncini

INdAM Workshop on Multiscale Problems

Cortona, September 18, 2006

ESSENTIAL REFERENCES

- Brezzi-Fortin, Book 1991 (Mixed Finite Elements)
- Eymard-Gallouet-Herbin, Book 2003 (Finite Volumes)
- Hyman-Scovel LANL 1988, Hyman-Knapp-Scovel, Physica D 1992
- Hyman-Shashkov, SJNM 1999, JCP 1999 ($Maxwell$)
- Hiptmair, Math. Comp 1999, Num. Math 2001 ($MFE\&Hodge$)
- Arnold, IMU Congress 2002
- Kuznetsov-Lipnikov-Shashkov LANL 2003
- Kuznetsov-Repin, Russian JNAM 2003
- Klausen-Russel, Comp. Geos. 2004
- Nicolaides-Trapp, IMA 2004

ESSENTIAL REFERENCES (2)

- Bochev-Hyman, IMA 2004
- Brezzi-Lipnikov-Simoncini, M^3AS 2005
- Lipnikov-Shashkov-Yotov, preprint 2005
- Buffa-Christiansen, Num. Math 2005 (*Discrete Diff. Forms*)
- Wheeler-Yotov, preprint 2005
- Brezzi-Lipnikov-Shashkov, SINUM 2005, M^3AS 2006
- Brezzi-Lipnikov-Shashkov-Simoncini, $CMAME$ 2006
- Arnold-Falk-Winther, Acta Numerica 2006
- Christiansen M^3AS, to appear
- Christiansen, Num. Math., to appear (*Discrete Diff. Forms*)
- Bochev-Gunzburger, CMAME. to appear

PLAN OF THE TALK

- Darcy's law
- Inner products
- Curved Faces
- Numerical Experiments
- Conclusions
- Appendix (in whatever time is left)

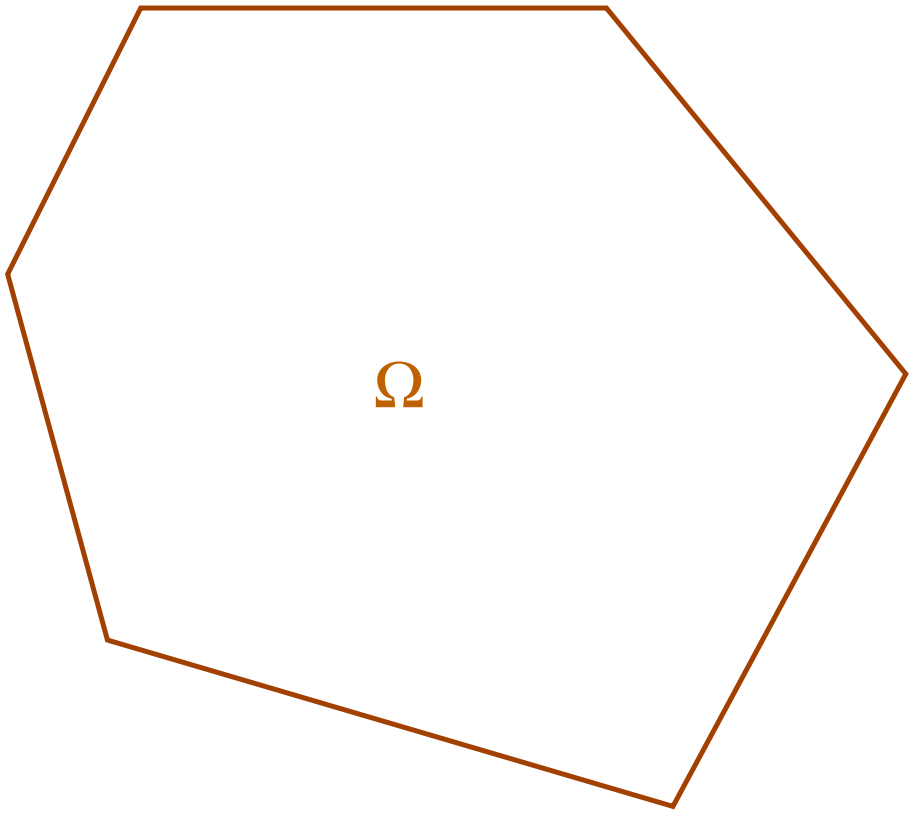
DARCY'S LAW

- p = pressure
- $\mathbf{u}$ = velocities
- b = source
- $\mathbb{K}$ = material-depending (full) tensor
- $\mathbf{u} = -\mathbb{K}\nabla p$ (Constitutive Equation)
- $\operatorname{div} \mathbf{u} = b$ (Conservation Equation)

$$-\operatorname{div}(\mathbb{K}\nabla p) = b \qquad \text{in } \Omega,$$

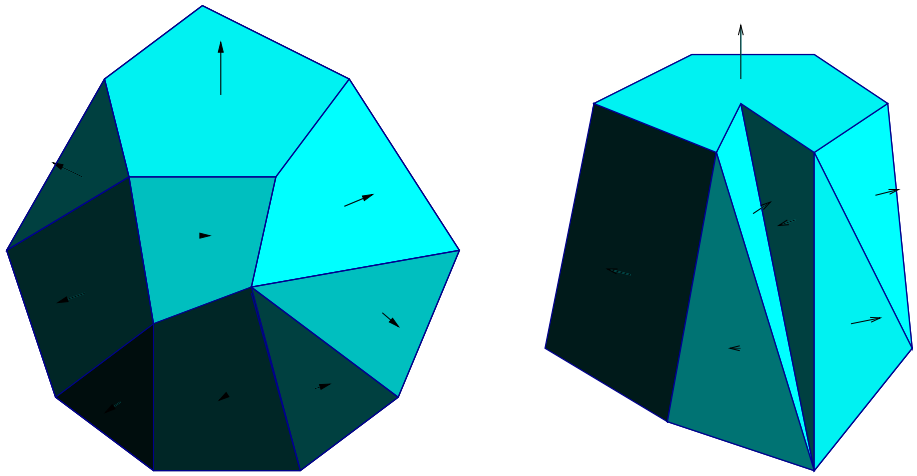
$$p = 0 \qquad \text{on } \partial\Omega, \qquad \text{for simplicity.}$$

THE DOMAIN Ω

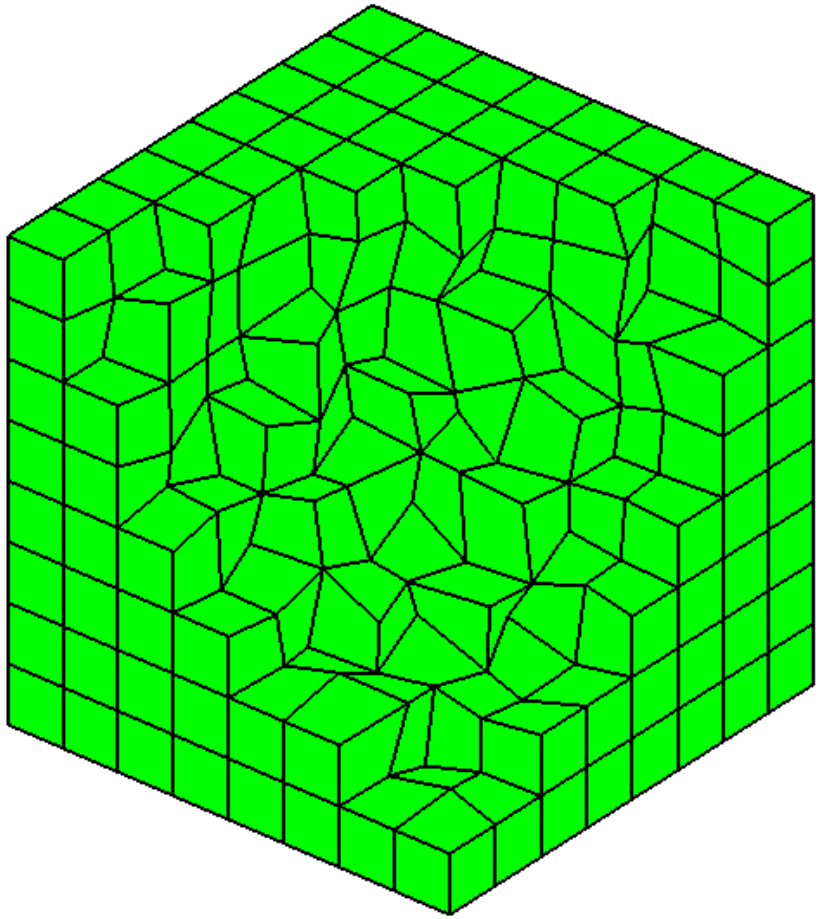


GENERALITY OF THE APPROACH

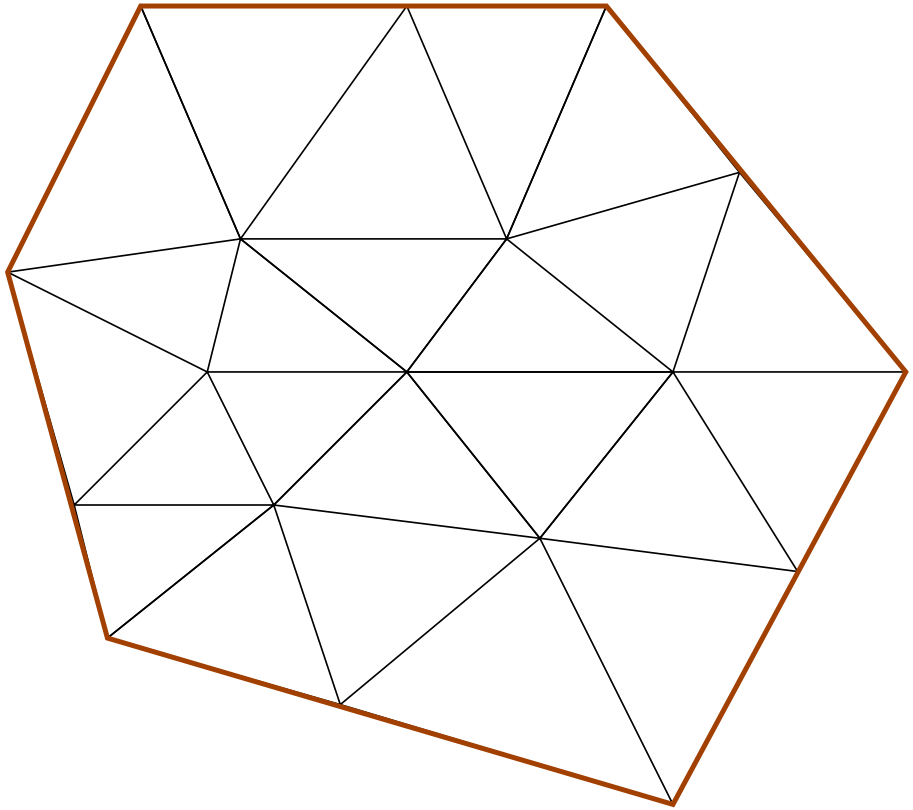
Note that, in the following discussion, many *figures* will be 2-dimensional. This corresponds to a *limitation of the speaker* and *not of the method*. Indeed, the method works in *very general* situations (including curved faces), and has actually been conceived in a three-dimensional framework.



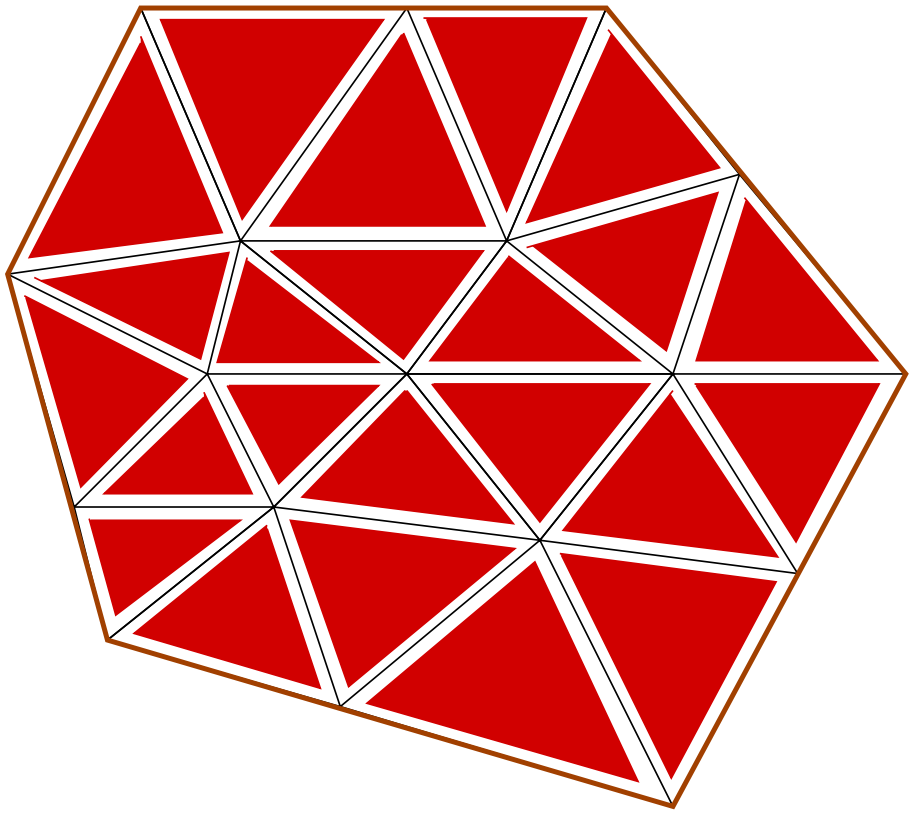
AN ACADEMIC BUT MORE REALISTIC DOMAIN



THE DECOMPOSITION OF Ω

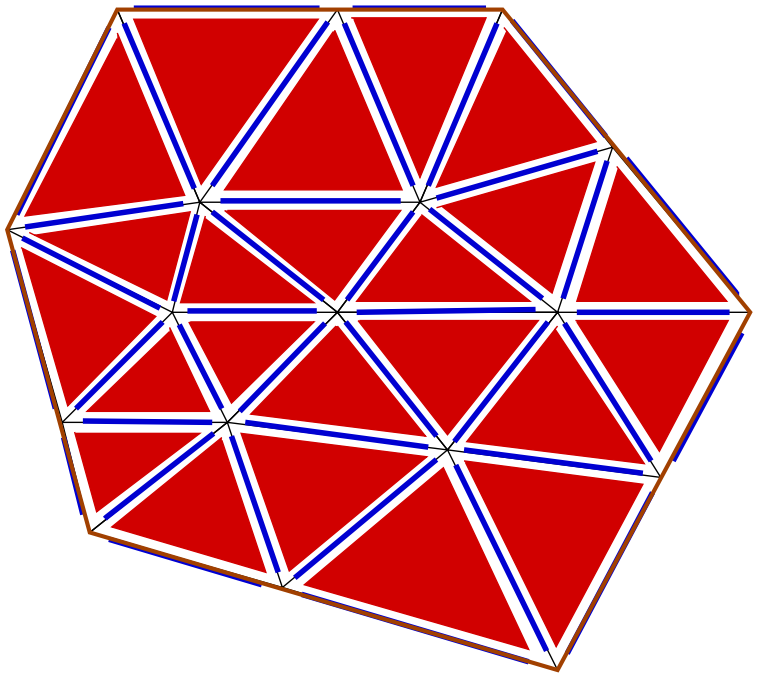


DISCRETE PRESSURES



Space Q: pressure is taken as constant in each element

DISCRETE FLUXES



Space $\mathbf{X}$: $\mathbf{F} = \mathbf{u} \cdot \mathbf{n}$ is taken constant on each edge (face)

Note that we *do not* discretize $\mathbf{u}$ inside each element, but only its *normal component* on the edges (faces). The orientation of the normal is chosen once and for all.

THE DISCRETE DIVERGENCE OPERATOR

On each element P, from the knowledge of $\mathbf{F} = \mathbf{u} \cdot \mathbf{n}$ on the boundary we can deduce "the average of div $\mathbf{u}$ inside" by Gauss theorem

$$\int_P \mathrm{div}\mathbf{u}\, \mathrm{d}V = \int_{\partial P} \mathbf{u} \cdot \mathbf{n}_{ext}\, \mathrm{d}S.$$

This suggests to introduce a *discrete divergence* operator $\mathcal{DIV}^h$ which associates, to every $\mathbf{G} \in \mathbf{X}$, a piecewise constant $\mathcal{DIV}^h\mathbf{G} \in Q'$ defined in each P as

$$\mathcal{DIV}^h\mathbf{G}_{|P} := \frac{1}{|P|} \int_{\partial P} \mathbf{G}_{ext}\, \mathrm{d}s.$$

with obvious meaning for $\mathbf{G}_{ext}$ (equal to plus or minus $\mathbf{G}$, according to the orientation that was chosen for the normal).

DISCRETIZATION OF THE CONSERVATION EQUATION

In order to discretize the equation

$$\operatorname{div} \mathbf{u} = b,$$

assuming that b is piecewise constant, we can simply use our *discrete divergence* operator $\mathcal{DIV}^h : \mathbf{X} \to \mathbf{Q}'$. The conservation equation is then discretized as

$$\mathcal{DIV}^h \mathbf{F} = b,$$

or, in variational form, as

$$_{\mathbf{Q}'}\langle \mathcal{DIV}^h \mathbf{F}, q \rangle_{\mathbf{Q}} = \int_\Omega b\, q \, \mathrm{d}V \qquad \forall q \in \mathbf{Q}.$$

This will be *the same* for Mimetic Finite Differences, Finite Volumes, and Mixed Finite Elements.

DISCRETIZATION OF THE CONSTITUTIVE EQUATION

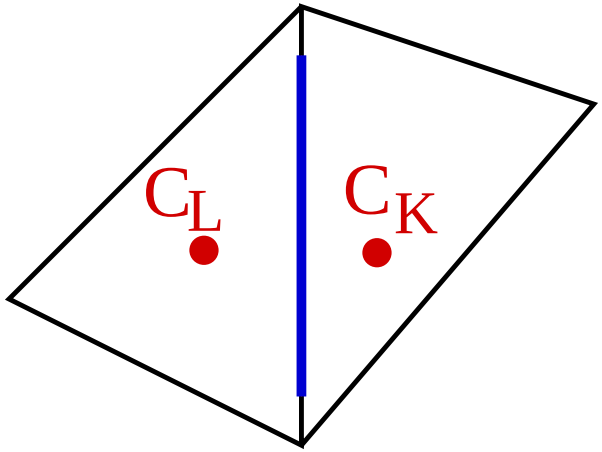
The problem is now to discretize the equation

$$\mathbf{u} = -\mathbb{K}\nabla p$$

{To be precise, we should actually define a *discrete gradient* operator $(-\mathcal{DIV}^h)^T : \mathbf{Q} \to \mathbf{X}'$ and then a (*- Hodge) operator $\mathbb{K} : \mathbf{X}' \to \mathbf{X}$.}

The discretization of the Constitutive Equation is done with three different approaches in Finite Volumes, in Mixed Finite Elements, and in Mimetic Finite Differences. As we shall see, Mimetic Finite Differences are somewhat *in between* Finite Volumes and Mixed Finite Elements.

FV DISCRETIZATION OF THE CONSTITUTIVE EQUATION



The points C_K and C_L are the *circumcenters* of the triangles.

Assuming for simplicity that $\mathbb{K} = \kappa\mathbb{I}$, the normal flux on the edge e_{KL} is defined, in (one of the most classical formulations of) Finite Volumes, as

$$F_{KL} := \kappa \frac{p_K - p_L}{|C_K - C_L|}$$

MFE DISCRETIZATION OF THE CONSTITUTIVE EQUATION

For Mixed Finite Elements, the passage from pressures to fluxes is made through a suitable reconstruction of the fluxes *inside* each element. More precisely in every triangle (tetrahedron) we consider the (Raviart-Thomas) space

$$RT := \{\mathbf{v}| \quad v_i = a_i + bx_i \quad (i = 1, 2, (3))\}.$$

Then for each $\mathbf{G} \in \mathbf{X}$ we reconstruct $\mathcal{R}(\mathbf{G})$ in Ω such that:

1) in every triangle, $\mathcal{R}(\mathbf{G})$ belongs to RT,

2) the normal components of $\mathcal{R}(\mathbf{G})$ on edges (faces) coincide with $\mathbf{G}$. As $\mathbf{G}$ is single valued, this will imply the continuity of the normal components of $\mathcal{R}(\mathbf{G})$.

3) div$\mathcal{R}(\mathbf{G})$ is a piecewise constant (hence **equal to** $\mathcal{DIV}^h\mathbf{G}$).

MFE DISCRETIZATION OF THE CONSTITUTIVE EQUATION

Once you have a reconstruction of the fluxes, you can introduce the inner product in $\mathbf{X}$, depending on $\mathbb{K}$, as

$$[\mathbf{F}, \mathbf{G}]_{\mathbf{X}} := \int_{\Omega} \mathbb{K}^{-1} \mathcal{R}(\mathbf{F}) \cdot \mathcal{R}(\mathbf{G}) \mathrm{d}V.$$

Note that $\mathcal{R}(\mathbf{F}) \sim \mathbf{u}$.

We then discretize the constitutive equation ($\sim \mathbb{K}^{-1} \mathcal{R}(\mathbf{F}) = -\nabla p$) using the variational formulation

$$[\mathbf{F}, \mathbf{G}]_{\mathbf{X}} = \int_{\Omega} p \, \mathrm{div} \mathcal{R}(\mathbf{G}) \mathrm{d}V \quad \forall \, \mathbf{G} \in \mathbf{X}.$$

FINAL MFE FORMULATION

Summarizing, the Mixed Finite Element formulation can be seen, in the present context, as: *find* $p \in Q$ *and* $\mathbf{F} \in \mathbf{X}$ *such that*:

$$\int_\Omega \mathbb{K}^{-1}\mathcal{R}(\mathbf{F}) \cdot \mathcal{R}(\mathbf{G})\mathrm{d}V = \int_\Omega p\,\mathrm{div}\mathcal{R}(\mathbf{G})\mathrm{d}V \quad \forall\, \mathbf{G} \in \mathbf{X}$$

and

$$\int_\Omega \mathrm{div}\mathcal{R}(\mathbf{F})\, q\mathrm{d}V = \int_\Omega b\, q\, \mathrm{d}V \qquad \forall q \in \mathbf{Q}.$$

Note that, in the usual presentation of MFE, the velocity unknown is considered to be, directly, $\mathbf{u} \sim \mathcal{R}(\mathbf{F})$ (while its normal components $\mathbf{F}$ on the edges are just *the degrees of freedom*). In MFD, as we shall see, we think instead that *the unknown is* $\mathbf{F}$, and the reconstruction $\mathbf{u} \sim \mathcal{R}(\mathbf{F})$ (if any) is used only to define the scalar product in $\mathbf{X}$.

MFD DISCRETIZATION OF THE CONSTITUTIVE EQUATION

We start by mimicking what is done in MFE. For this, we need a suitable *inner product* in $\mathbf{X}$ (depending on $\mathbb{K}$), that allow us to write

$$[\mathbf{F}, \mathbf{G}]_{\mathbf{X}} = \int_{\Omega} p\, \mathcal{D}\mathcal{I}\mathcal{V}^h \mathbf{G} \mathrm{d}V \quad \forall\, \mathbf{G} \in \mathbf{X}.$$

One possibility is again *to reconstruct*. For each element P, from $\mathbf{G}$ on ∂P you reconstruct (somehow) $\mathcal{R}_P(\mathbf{G})$ in P. Then you set

$$[\mathbf{F}, \mathbf{G}]_{\mathbf{X}} := \sum_P \int_P \mathbb{K}_P^{-1} \mathcal{R}_P(\mathbf{F}) \cdot \mathcal{R}_P(\mathbf{G}) \mathrm{d}V.$$

MFD IN VERY SIMPLE GEOMETRIES

If our elements are triangles (or tetrahedra), then the easiest way to define the inner product in $\mathbf{X}$ is surely to use the Raviart-Thomas (RT) reconstruction as in MFE (although other simple choices are available). In that case, the MFD formulation will *coincide* with MFE: the method will just be written differently:

$$[\mathbf{F},\mathbf{G}]_{\mathbf{X}} \quad \Big(= \sum_P \int_P \mathbb{K}_P^{-1} \mathcal{R}_P(\mathbf{F}) \cdot \mathcal{R}_P(\mathbf{G}) \mathrm{d}V \Big) = \int_\Omega p\, \mathcal{DIV}^h \mathbf{G} \mathrm{d}V$$

for all $\mathbf{G} \in \mathbf{X}$, and

$$\int_\Omega \mathcal{DIV}^h \mathbf{F}\, q \,\mathrm{d}V = \int_\Omega p\, q \,\mathrm{d}V$$

for all $q \in Q$.

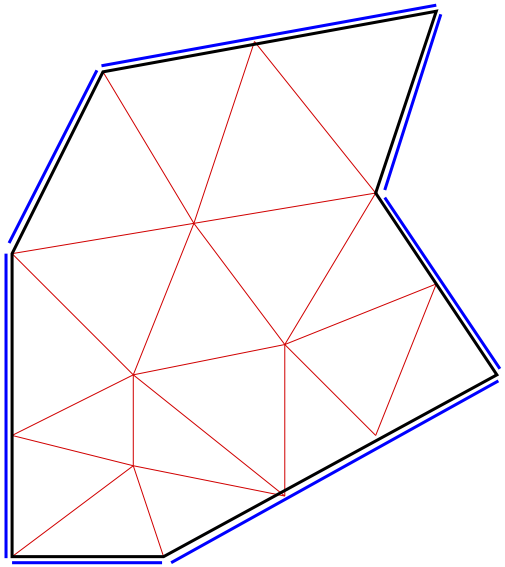
WHAT TO DO FOR COMPLEX GEOMETRIES

In more complex geometries, simple spaces (as *RT*) to be used for the reconstruction are *not available*. Hence to build a suitable reconstruction operator becomes *cumbersome*.

A very good idea to deal with the problem was proposed by Y. Kuznetsov-S. Repin (2004) and generalized by S. Christiansen (2006). It amounts to construct a *subgrid* made of triangles or tetrahedra and reconstruct the fluxes according to the following rules

- In each triangle/tetrahedron the flux is a Raviart-Thomas field.
- The divergence of the flux is *constant on the whole element*.

RECONSTRUCTION USING A SUBGRID



For each $\mathbf{G} \in \mathbf{X}$ and for each element P, we use the subgrid to construct a MFE (Raviart-Thomas) approximate solution $(\boldsymbol{\tau}_h, \phi_h)$ of the Neumann problem

$$-\mathrm{div}\mathbb{K}\nabla\phi = \mathcal{D}\mathcal{I}\mathcal{V}^h\mathbf{G}_P \quad \text{in } P \qquad -\mathbb{K}\nabla\phi\cdot\mathbf{n}_{ext} = (\mathbf{G}_P)_{ext} \quad \text{on } \partial P$$

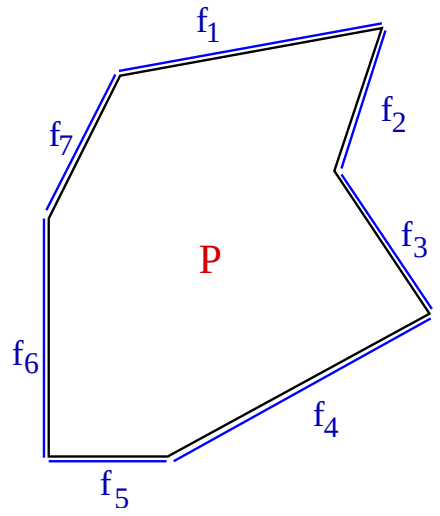
and then we set $\mathcal{R}_P(\mathbf{G}_P) := \boldsymbol{\tau}_h$.

THE BASIC IDEA

We saw that there are various possibilities to build a reconstruction operator, and then to define the scalar product in **X** accordingly.

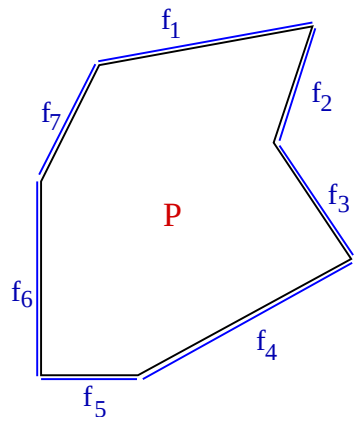
However here **the name of the game** is to **guess** how a "scalar product based on reconstruction" should be, and then **invent** a scalar product **without** actually building a reconstruction operator (and **to get away with that**).

Our scalar product in $\mathbf{X}$ will be defined as the sum of scalar products $[\cdot,\cdot]_{\mathbf{X}_P}$ on individual elements P. To fix ideas, we assume that we are in 2 dimensions, that P has 7 edges, and that $\mathbb{K}_P$ is constant in P.



Assume that you represent the elements of $\mathbf{X}_P$ in the canonical basis $\mathbf{E}^{(1)}, \mathbf{E}^{(2)}, ...\mathbf{E}^{(7)}$ by prescribing $\mathbf{E}^{(i)}{}_{|f_j} = \delta_{i,j}$. Then every element $\mathbf{G}$ in $\mathbf{X}_{|P}$ will be represented as an element of $\mathbb{R}^7$ with $\mathbf{G} = \sum_{i=1}^{7} \mathbf{G}_i \mathbf{E}^{(i)}$.

ASSOCIATED MATRIX



Every possible reconstruction $\mathcal{R}_P$ will produce a scalar product

$$[\mathbf{F}, \mathbf{G}]_{\mathbf{X}_P} = \int_P \mathbb{K}_P^{-1} \mathcal{R}_P(\mathbf{F}) \cdot \mathcal{R}_P(\mathbf{G}) \mathrm{d}V$$

which, in turn, will be representable as a 7×7 matrix $\mathbb{M}_P$, namely $[\mathbf{F}, \mathbf{G}]_{\mathbf{X}_P} = \sum_{i,j} \mathbb{M}_{P\,i,j} \mathbf{F}_i \mathbf{G}_j$, with

$$\mathbb{M}_{P\,i,j} := \int_P \mathbb{K}_P^{-1} \mathcal{R}_P(\mathbf{E}^{(i)}) \cdot \mathcal{R}_P(\mathbf{E}^{(j)}) \mathrm{d}V.$$

$\mathcal{P}_0$-COMPATIBLE RECONSTRUCTIONS

We shall now restrict our attention to *reasonable reconstructions* (that we shall call $\mathcal{P}_0$-*compatible reconstructions*). These are linear mappings $\mathcal{R}_P$ defined on $\mathbf{X}_P$ and having the following properties:

- For every $\mathbf{G} \in \mathbf{X}_P$, we have that $\mathcal{R}_P(\mathbf{G}) \in L^2(P)$.
- For every $\mathbf{G} \in \mathbf{X}_P$, we have that $\mathrm{div}\mathcal{R}_P(\mathbf{G})$ is constant in P.
- For every $\mathbf{G} \in \mathbf{X}_P$ and for every face f_i of ∂P, we have $\mathcal{R}_P(\mathbf{G})_{|f_i} \cdot \mathbf{n}_{|f_i} = \mathbf{G}_{|f_i}$ (hence $\mathrm{div}\mathcal{R}_P(\mathbf{G}) = \mathcal{DIV}^h\mathbf{G}$).
- For every constant vector $\mathbf{c}$, setting for all faces $\mathbf{G}^{\mathbf{c}}_{|f_i} := \mathbf{c} \cdot \mathbf{n}_{|f_i}$ we have that $\mathcal{R}_P(\mathbf{G}^{\mathbf{c}}) \equiv \mathbf{c}$. Note the difference between $\mathbf{c} \in \mathbb{R}^2$ and $\mathbf{G}^{\mathbf{c}} \in \mathbb{R}^7$!!!

SCALAR PRODUCTS ASSOCIATED WITH $\mathcal{P}_0$-COMPATIBLE RECONSTRUCTIONS

We **claim** now that: **if** $\mathbf{c}$ is a constant vector and $\mathbf{G^c}$ has been constructed as before, and **if** $\mathcal{R}_P$ is a $\mathcal{P}_0$-compatible reconstruction, **then** for every $\mathbf{G} \in \mathbf{X}_P$ the result of

$$\int_P \mathbb{K}_P^{-1} \mathcal{R}_P(\mathbf{G^c}) \cdot \mathcal{R}_P(\mathbf{G}) \mathrm{d}V$$

depends on P, $\mathbb{K}$, $\mathbf{c}$ and $\mathbf{G}$, but **not** on the choice of the reconstruction (among all possible $\mathcal{P}_0$-compatible reconstructions). Indeed...

Set $q^1(\mathbf{x}) := (\mathbb{K}_P^{-1}\mathbf{c}) \cdot (\mathbf{x} - \mathbf{x}_B)$ (where $\mathbf{x_B}$ is the barycenter of P). Then we have

$$\int_P \mathbb{K}_P^{-1}\mathcal{R}_P(\mathbf{G^c}) \cdot \mathcal{R}_P(\mathbf{G})\mathrm{d}V =$$

$$\int_P \mathbb{K}_P^{-1}\mathbf{c} \cdot \mathcal{R}_P(\mathbf{G})\mathrm{d}V = \int_P \nabla q^1 \cdot \mathcal{R}_P(\mathbf{G})\mathrm{d}V =$$

$$-\int_P \operatorname{div}\mathcal{R}_P(\mathbf{G})\, q^1 \mathrm{d}V + \int_{\partial P} q^1\, \mathcal{R}_P(\mathbf{G}) \cdot \mathbf{n}_{ext}\, \mathrm{d}S =$$

$$-\int_P \mathcal{DIV}^h\mathbf{G}\, q^1 \mathrm{d}V + \int_{\partial P} q^1\, \mathbf{G}_{ext}\, \mathrm{d}S$$

$$= 0 + \int_{\partial P} (\mathbb{K}_P^{-1}\mathbf{c}) \cdot (\mathbf{x} - \mathbf{x}_B)\, \mathbf{G}_{ext}\, \mathrm{d}S.$$

Let us summarize the previous result. We found that if the scalar product in $\mathbf{X}_P$ is obtained through a $\mathcal{P}_0$-compatible reconstruction, and if $\mathbf{G}^{\mathbf{c}}$ is associated to *a constant vector*, $\mathbf{c}$, then

$$[\mathbf{G}^{\mathbf{c}}, \mathbf{G}]_{\mathbf{X}_P} = \int_{\partial P} (\mathbb{K}_P^{-1}\mathbf{c}) \cdot (\mathbf{x} - \mathbf{x}_B)\, \mathbf{G}_{ext}\, \mathrm{d}S.$$

It is also simple to check that taking two constant vectors in the canonical basis of $\mathbb{R}^2$, $\mathbf{e}^1 = (1, 0)$ and $\mathbf{e}^2 = (0, 1)$, then we *must* have

$$[\mathbf{G}^{\mathbf{e}^i}, \mathbf{G}^{\mathbf{e}^j}]_{\mathbf{X}_P} = \int_P \mathbb{K}_P^{-1}\mathcal{R}_P(\mathbf{G}^{\mathbf{e}^i}) \cdot \mathcal{R}_P(\mathbf{G}^{\mathbf{e}^j})\mathrm{d}V =$$

$$\int_P \mathbb{K}_P^{-1}\mathbf{e}^i \cdot \mathbf{e}^j \mathrm{d}V = (\mathbb{K}_P^{-1})_{i,j}|P|.$$

It seems now natural to *change the basis in* $\mathbf{X}_P$ We take

$$\widetilde{\mathbf{E}}^1 := \mathbf{G}^{\mathbf{e}^1}, \quad \widetilde{\mathbf{E}}^2 := \mathbf{G}^{\mathbf{e}^2},$$

and then we complete the basis with vectors in $\mathbb{R}^7$

$$\widetilde{\mathbf{E}}^3, \quad \widetilde{\mathbf{E}}^4, \ldots \quad \widetilde{\mathbf{E}}^7$$

such that

$$[\mathbf{G}^{\mathbf{e}^i}, \widetilde{\mathbf{E}}^j]_{\mathbf{X}_P} = \int_{\partial P} (\mathbb{K}_P^{-1}\mathbf{e}^i)\cdot(\mathbf{x}-\mathbf{x}_B)\, \widetilde{\mathbf{E}}^j_{ext}\, \mathrm{d}S = 0 \quad (i = 1, 2 \quad j = 3, .., 7)$$

Note that all this *does not* depend on the choice of the reconstruction.

In the new basis $\widetilde{\mathbf{E}^1}, ..., \widetilde{\mathbf{E}^7}$ the matrix associated to **any** scalar product obtained with a $\mathcal{P}_0$-compatible reconstruction will then have the form

$\mathrm{K}^{-1}\lvert\mathrm{P}\rvert$	0
0	?

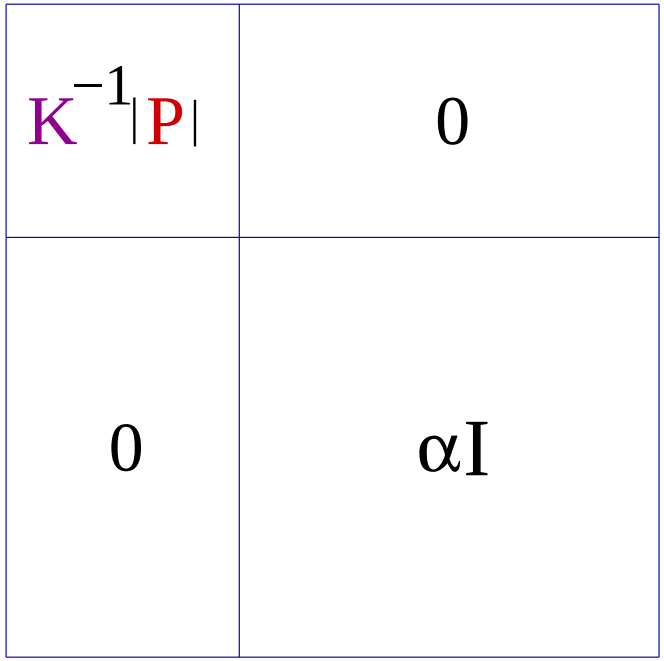
and the 5×5 diagonal block "**?**" will depend on the reconstruction.

Theorem. There exists an $\alpha_0 > 0$ such that: for every symmetric and positive definite 5×5 matrix $\mathbf{S}$ with smallest eigenvalue $\geq \alpha_0$ there exists a $\mathcal{P}_0$-compatible reconstruction whose associated scalar product corresponds, in the basis $\widetilde{\mathbf{E}^1}, ..., \widetilde{\mathbf{E}^7}$, to the matrix

$\mathrm{K}^{-1}\lvert\mathrm{P}\rvert$	0
0	S

In other words: I know that the matrix comes from **a** reconstruction. I don't care to know "which one".

In all our experiments we took the matrix as



with $\alpha = |P|\text{trace}(\mathbb{K}^{-1})$, and we got very good results.

ERROR ESTIMATES (B.-Lipnikov-Shashkov-Simoncini)

Assume that:

- Ω has a Lipschitz continuous boundary.
- Every element is *uniformly strictly* starshaped.
- The number of faces per element is uniformly bounded.
- Every face is *uniformly strictly* starshaped.
- The number α of the previous slide verifies $\alpha \simeq |P|$.

Then

$$\|p - p_h\|_0 + \|\mathbf{F}^I - \mathbf{F}_h\|_{\mathbf{X}} \leq C\, h.$$

If moreover Ω is convex and $\alpha \geq \alpha_0$, then

$$\|p^I - p_h\|_0 \leq C\, h^2,$$

where p^I is the (element by element) mean value of p.

CURVED FACES IN 3 DIMENSIONS

To deal with *curved faces* we must use, inside each domain, *three unknowns per face* (the three components of a constant vector).

The face e is called *moderately curved* if, *at every point* of e, we have

$$|\widetilde{\mathbf{n}} - \mathbf{n}| \leq \frac{\sigma_*}{L}|e|^{1/2} \quad \text{where} \quad \widetilde{\mathbf{n}} := \frac{\int_e \mathbf{n} \mathrm{d}S}{|e|}.$$

Here σ_* is a (chosen) positive number and L a characteristic length. A face that is not moderately curved will be called *strongly curved*.

Clearly, the bigger we choose σ_*, the bigger will be the number of moderately curved faces, and the smaller that of *strongly curved* ones.

In assembling the global matrix, we require only the continuity of the *averaged normal component* on moderately curved faces, but we require the continuity of all three components on the strongly curved faces.

CURVED FACES

The total number of flux unknowns will then equal the number of moderately curved faces plus three times the number of strongly curved faces.

Hence, strongly curved faces are more expensive. From the point of view of cost we have then the convenience to take σ_* very big, so that the number of strongly curved faces will be small.

As usual, however, you pay somewhere else: actually the error estimates go like

$$\|p - p_h\|_0 + \|\mathbf{F}^I - \mathbf{F}_h\|_{\mathbf{X}} \leq C\,(1 + \sigma_*)\,h.$$

Finally, we point out that the same trick (actually, **better**) for computing the scalar products holds for curved faces.

NUMERICAL EXPERIMENTS

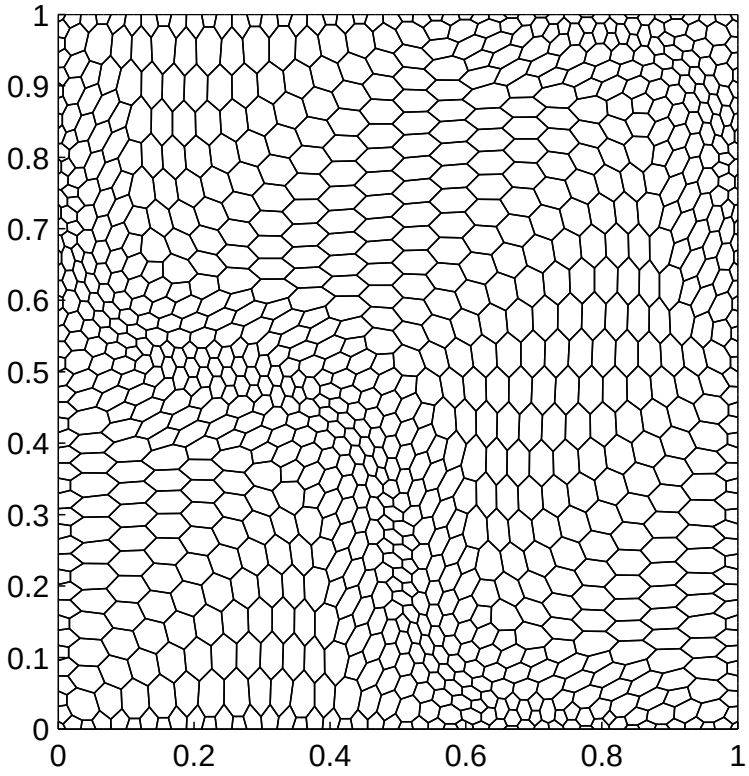
Example 1. Let us consider the Dirichlet boundary value problem in the unit square $[0,\,1]^2$ with the exact solution

$$p(x,\,y) = x^3y^2 + x\sin(2\pi xy)\sin(2\pi y)$$

and the full diffusion tensor

$$\mathbb{K} = \begin{pmatrix} (x+1)^2 + y^2 & -xy \\ -xy & (x+1)^2 \end{pmatrix}.$$

Polygonal mesh

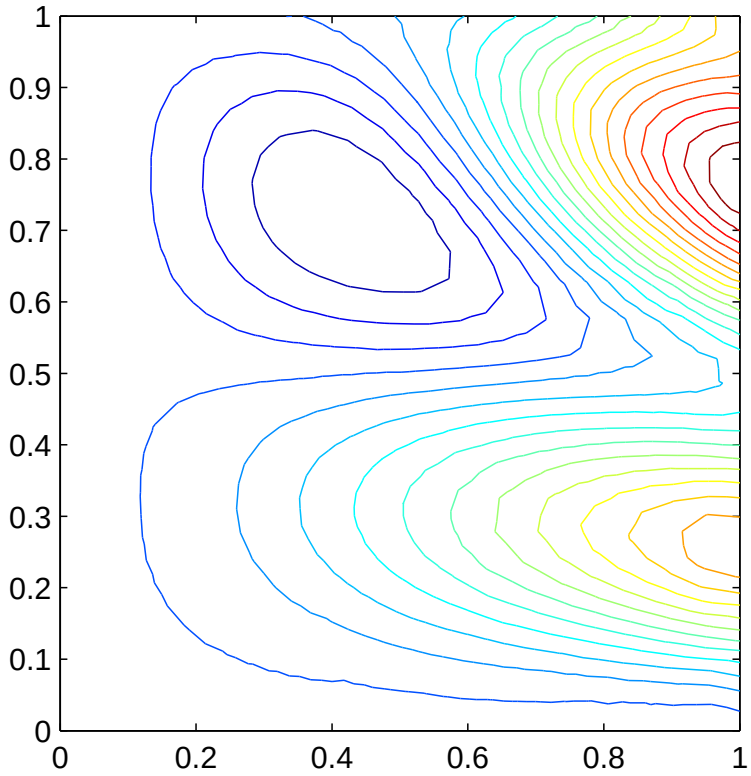


The table below shows the errors for different refinements.

Table 1: Convergence analysis on polygonal meshes.

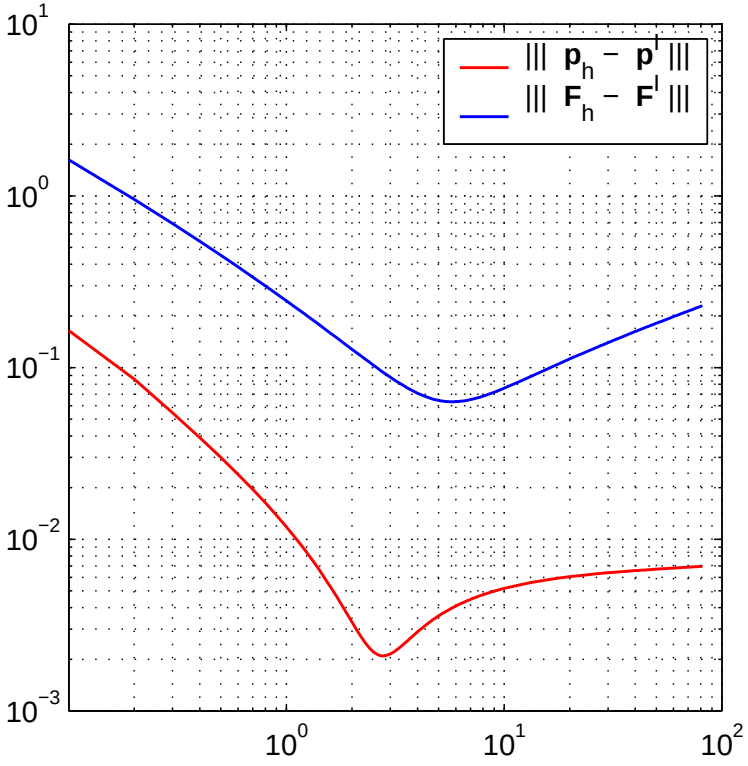
$1/h$	$\|\|p^I - p_h\|\|_0$	$\|\|\mathbf{F}^I - \mathbf{F}_h\|\|_{\mathbf{x}}$	$\|\|p^I - p_h\|\|_\infty$	$\|\|\mathbf{F}^I - \mathbf{F}_h\|\|_\infty$
16	5.17e-2	7.38e-1	1.61e-1	5.25e-0
32	1.18e-2	2.44e-1	4.54e-2	2.80e-0
64	2.76e-3	8.45e-2	1.28e-2	1.46e-0
128	6.65e-4	2.89e-2	3.06e-3	7.79e-1
rate	2.09	1.56	1.90	0.92

Isolines of the solution



Errors as a function of α

The spectral properties of the matrix $\mathbb{M}_P$ defining the scalar product in P depend on a balance between the extreme eigenvalues of $\mathbb{K}$ and α. The figure in the next slide shows errors (vertical axis) as functions of α^{-1} (horizontal axis) for the case $1/h = 32$. There is a quite big interval $\alpha^{-1} \in [2, 80]$ where the errors vary only 3 times. What is remarkable here is that for all values of α we observed second order convergence rate for p_h in L^2 and 1.5 convergence rate for $\mathbf{F}_h$ in L^2.

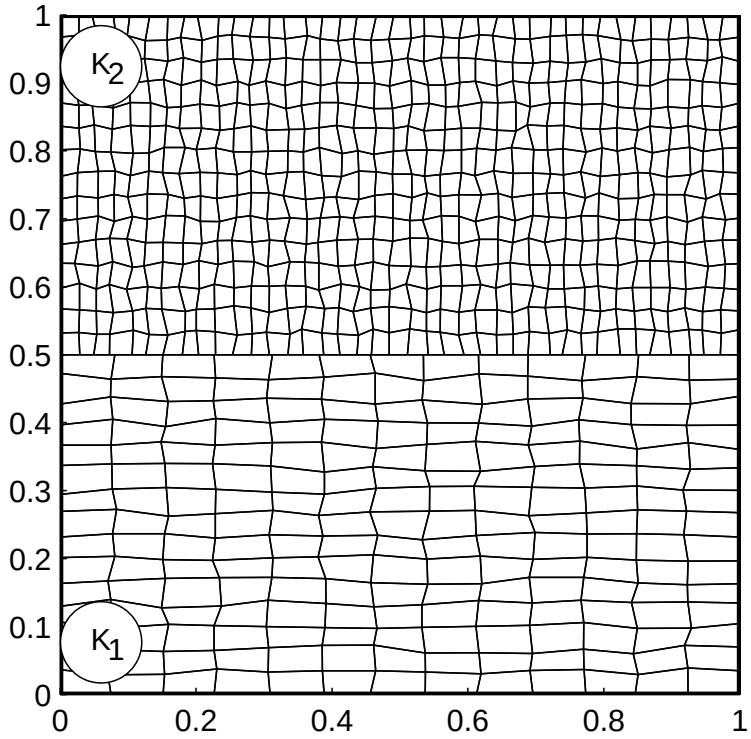

10^{1}
10^{0}
10^{-1}
10^{-2}
10^{-3}
10^{0}
10^{1}
10^{2}
$|||\ \mathbf{p}_h - \mathbf{p}^I\ |||$
$|||\ \mathbf{F}_h - \mathbf{F}^I\ |||$

Example 2. Let us consider a problem in the unit square $[0, 1]^2$ with *mixed* boundary conditions. On the bottom ($y = 0$) and top ($y = 1$) boundaries, we impose the Dirichlet boundary condition. Homogeneous Neumann boundary conditions are set on the other (vertical) boundaries. Let the tensor $\mathbb{K}$ be scalar and equal to $K_1\mathbb{I}$ in the region defined by $y < 0.5$ and $K_2\mathbb{I}$ in the rest of the domain. The source term is chosen in such a way that the exact solution is given by

$$p(x, y) = \begin{cases} \frac{7}{16} - \frac{K_2}{12K_1} + \frac{2K_2}{3K_1}\, y^3, & y < 0.5, \\ \\ y - y^4, & y \geq 0.5 \end{cases}$$

and hence, in particular, depends only on y.

The meshes we used come from non-matching mesh methods but we treat them as conformal polygonal meshes. The mesh below is obtained as a random perturbation of an originally "uniform, non matching" mesh.

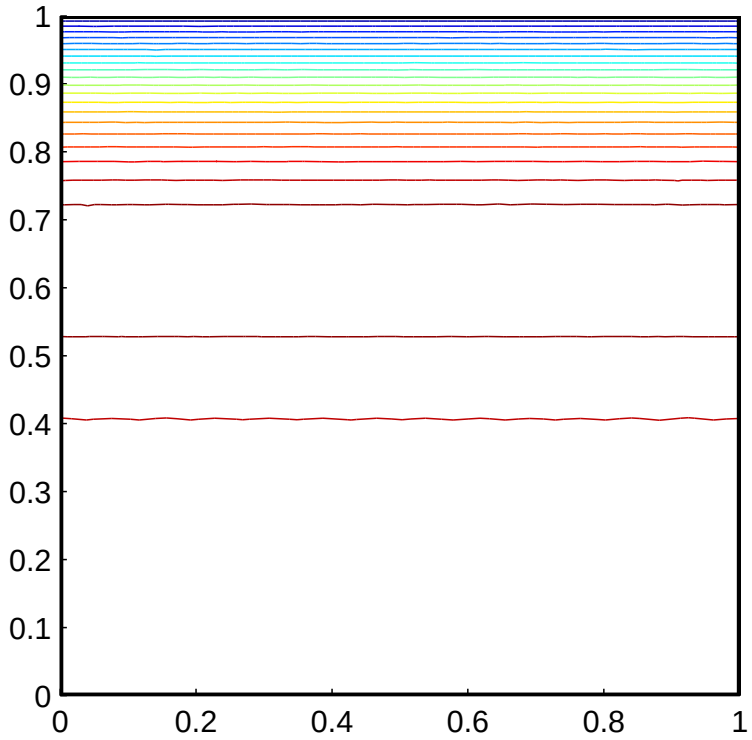


In the numerical experiments we used $K_1 = 10$, $K_2 = 1$. We observe superconvergence of the scalar variable in both norms. The lack of flux superconvergence is typical for random meshes and is observed in other similar discretization schemes on simplicial meshes.

Table 2: Convergence analysis on non-matching meshes.

#cells	$\|\|p^I - p_h\|\|_0$	$\|\|\mathbf{F}^I - \mathbf{F}_h\|\|_{\mathbf{X}}$	$\|\|p^I - p_h\|\|_\infty$	$\|\|\mathbf{F}^I - \mathbf{F}_h\|\|_\infty$
780	1.01e-2	1.12e-1	2.82e-2	7.80e-1
3286	2.36e-3	4.72e-2	6.70e-3	3.51e-1
13482	5.73e-4	2.24e-2	1.78e-3	1.38e-1
54610	1.41e-4	1.09e-2	4.37e-4	7.70e-2
rate	2.01	1.09	1.95	1.11

Isoline plot of the solution



Example 3. Let us consider the Dirichlet boundary value problem with the exact solution

$$p(x,\, y,\, z) = x^3 y^2 z + x \sin(2\pi xy) \sin(2\pi yz) \sin(2\pi z).$$

We take the diffusion tensor as:

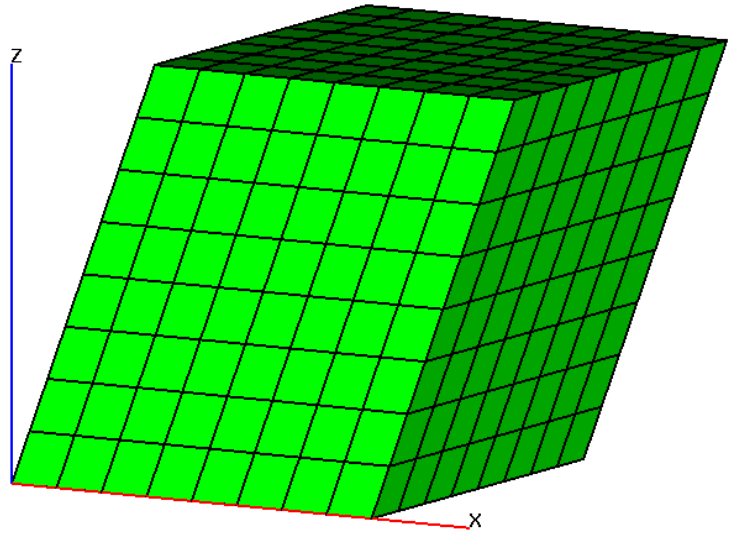
$$\mathbb{K} = \begin{pmatrix} 1 + y^2 + z^2 & -xy & -xz \\ -xy & 1 + x^2 + z^2 & -yz \\ -xz & -yz & 1 + x^2 + y^2 \end{pmatrix}.$$

It is not difficult to check that $\mathbb{K}$ is a positive definite matrix for all values of x, y and z.

We consider a sequence of uniform cubic meshes in the unit cube $[0,\,1]^3$ and generate a corresponding sequence of hexahedral meshes using the following linear transformation:

$$\tilde{x} = x + \varepsilon z, \qquad \tilde{y} = y + \varepsilon z, \qquad \tilde{z} = z.$$

An example of a modified mesh is shown in the next figure. It corresponds to $\varepsilon = 0.25$ and $h = 1/8$, where h is the size of a cubic element in the original mesh.

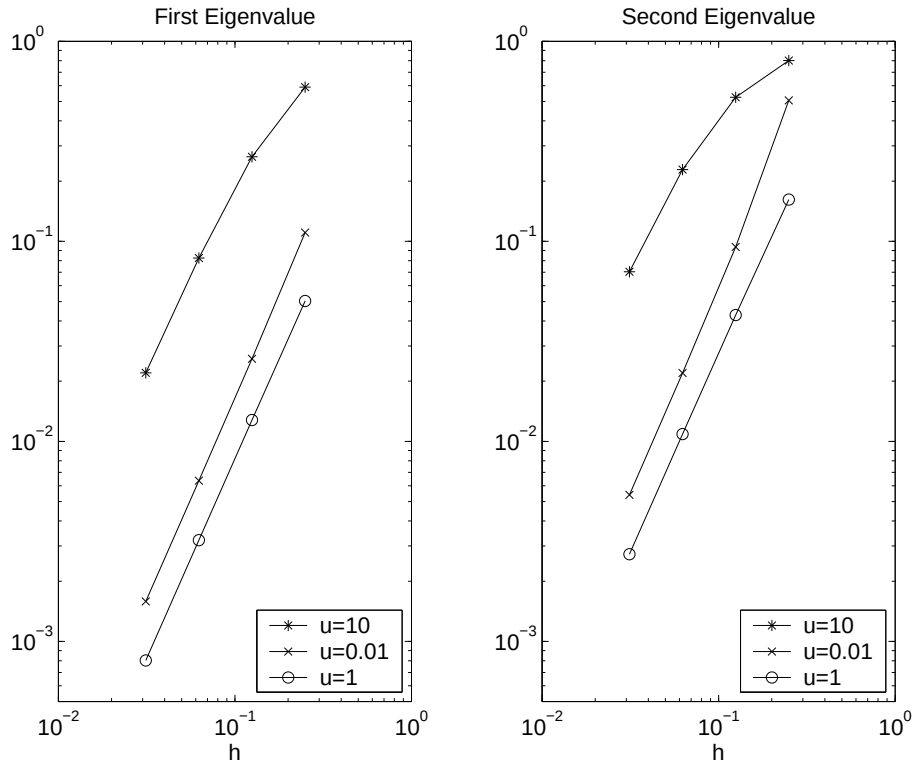

Z
X

The results presented in the table below show the superconvergence of the scalar variable in both norms and the superconvergence of the vector variable in the discrete L_2-norm.

Table 3: Convergence analysis on polyhedral meshes.

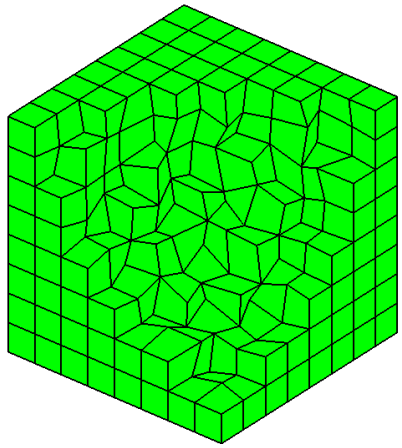
$1/h$	$\|\|p^I - p_h\|\|_0$	$\|\|\mathbf{F}^I - \mathbf{F}_h\|\|_{\mathbf{X}}$	$\|\|p^I - p_h\|\|_\infty$	$\|\|\mathbf{F}^I - \mathbf{F}_h\|\|_\infty$
8	3.83e-2	5.35e-1	1.55e-1	6.07e-0
16	1.10e-2	1.43e-1	4.83e-2	2.48e-0
32	2.86e-3	3.58e-2	1.26e-2	1.11e-0
64	7.21e-4	8.94e-3	3.28e-3	5.42e-1
rate	1.91	1.97	1.86	1.16

Approximation of eigenvalues



Approximation of the first two eigenvalues of Laplace operator on the square $]0,1[^2$, on a uniform mesh of squares, for different values of $\alpha = 0.01,\ 1,\ 10$. By A. Cangiani and M. Manzini.

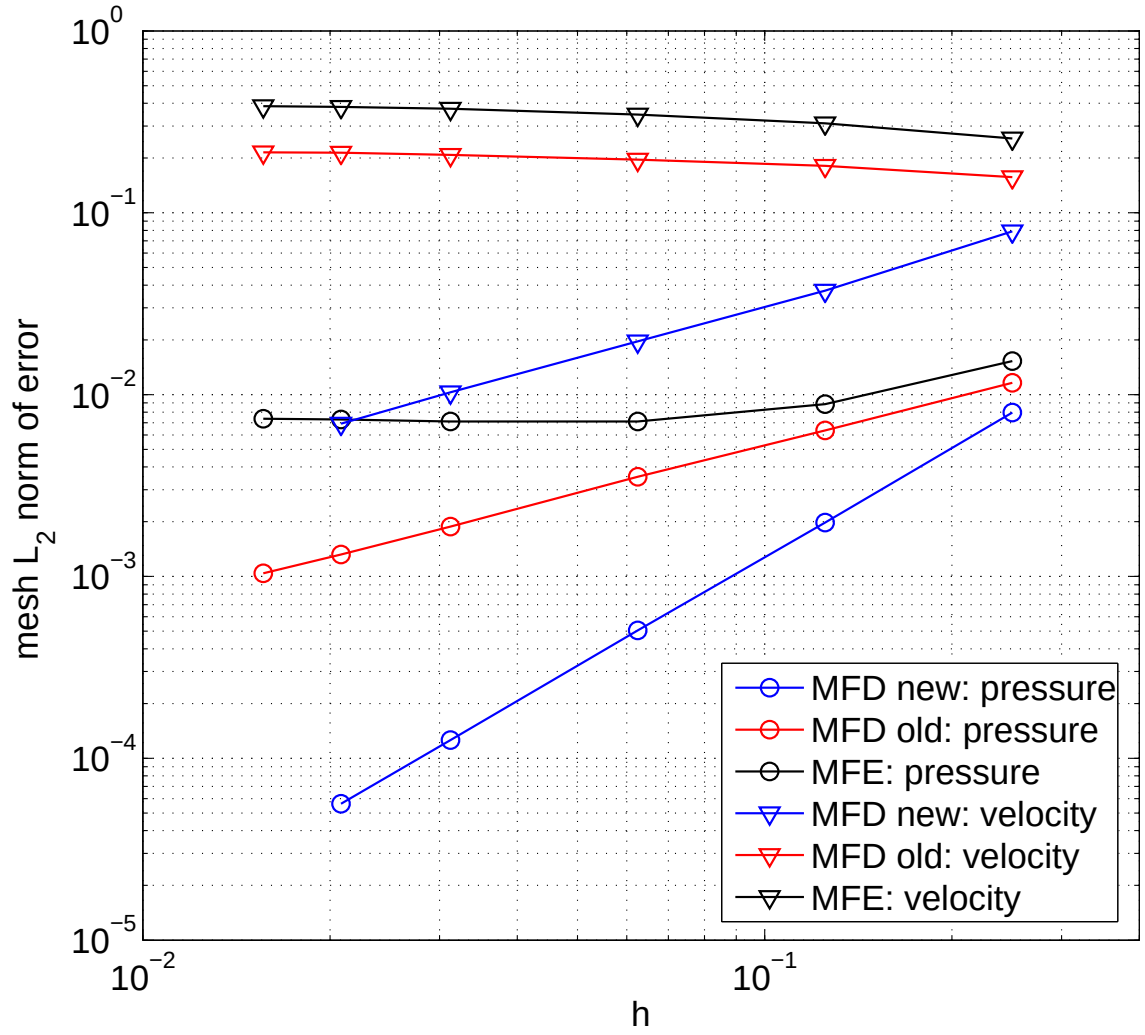
3D-Curved faces

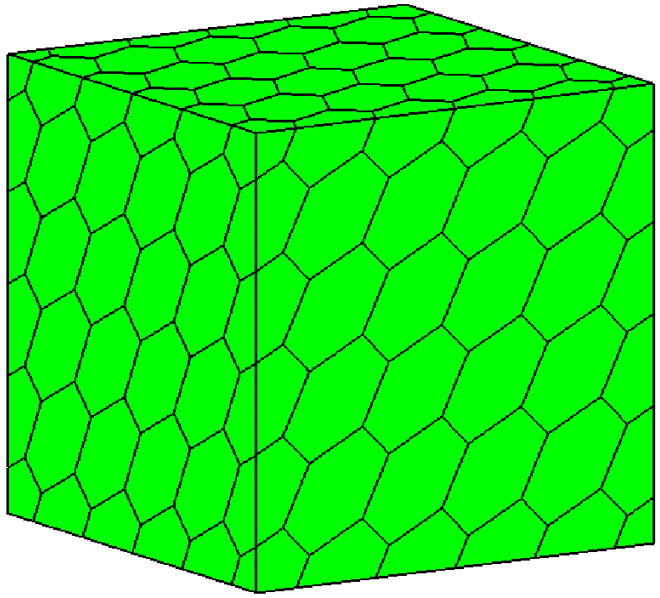


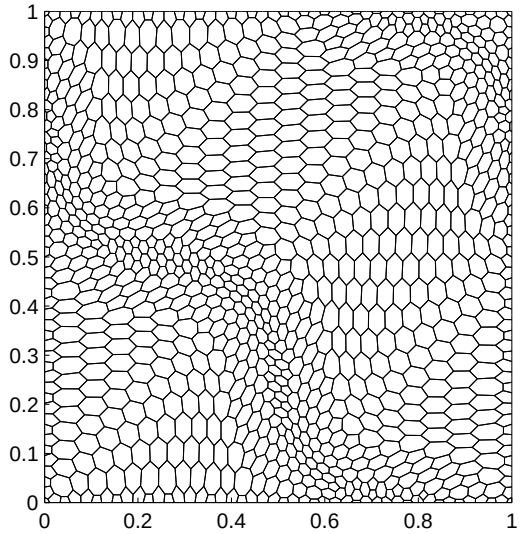
Here we have $\mathbb{K}$=Identity. The exact solution is

$$p(x,\, y,\, z) = x^2 y^3 z + 3x \sin(yz).$$

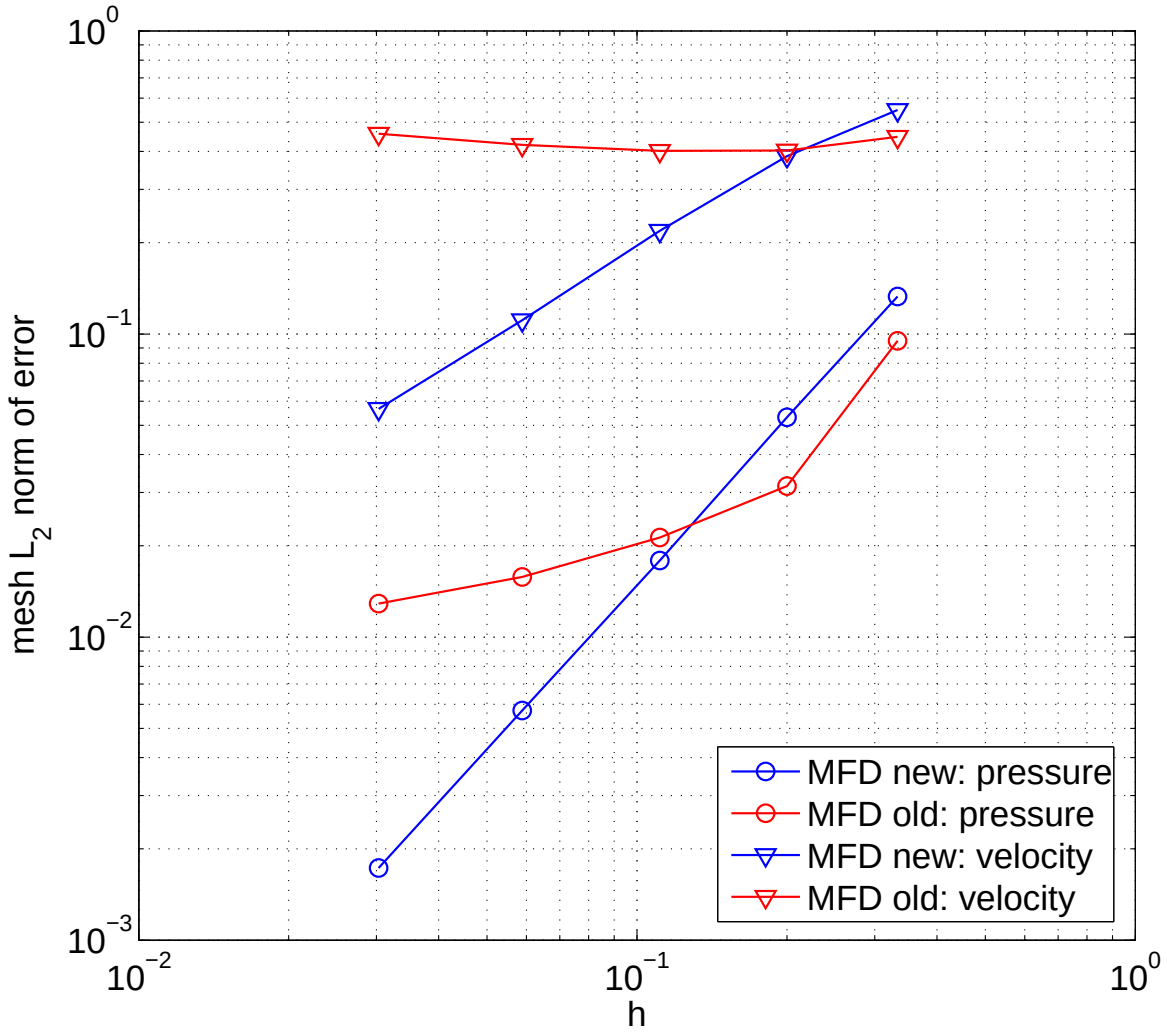
Now we compare: Mixed Finite Elements, Mimetic Finite Differences treating all faces as moderately curved, and Mimetic Finite Differences with $\sigma^* = 0.2$.


10^{0}
10^{-1}
10^{-2}
10^{-3}
10^{-4}
10^{-5}
mesh L_2 norm of error
10^{-2}
10^{-1}
h
MFD new: pressure
MFD old: pressure
MFE: pressure
MFD new: velocity
MFD old: velocity
MFE: velocity





We do the same for a polyhedral mesh (distorted inside as in the 2-d case). Here we have no Mixed Finite Elements available for comparison.


10^0
10^{-1}
10^{-2}
10^{-3}
mesh L_2 norm of error
10^{-2}
10^{-1}
10^0
h
MFD new: pressure
MFD old: pressure
MFD new: velocity
MFD old: velocity

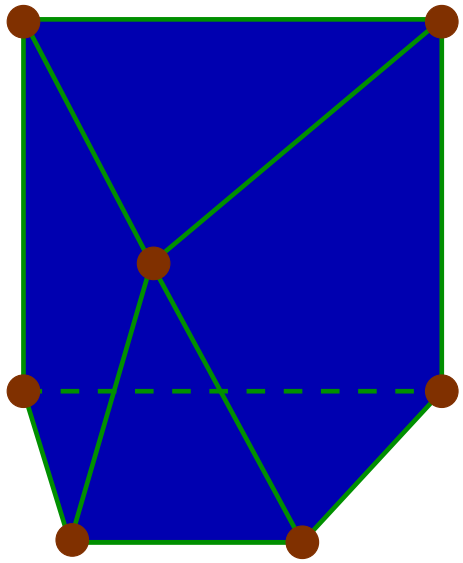
CONCLUSIONS

- Mimetic Finite Differences aim at a better representation of fundamental physical laws.
- The present approach allows an enormous freedom in the type of decomposition. The treatment of non-matching grids and of curved faces is also allowed.
- The basic ideas could be generalized to other problems, involving *gradients* and *curls* (in the framework of *cochain approximations of differential forms*).
- We still have to understand how to make the best use of the freedom allowed by our construction: indeed S does not need to be a diagonal matrix αI.
- For curved faces we need more indications on the optimal choice for the parameter σ_*.

APPENDIX

A ONE-DOMAIN SETTING

We consider a Polyhedron, with its Vertices, its Edges, and its Faces



● Vertices — Edges ▰ Faces

7 Vertices 11 Edges 6 Faces

We consider, to start with, a single polyhedron P with vertices $V_1, V_2, ...V_7$, edges $e_1, e_2, ..., e_{11}$, and faces $f_1, f_2, ..., f_6$. The numbers (7, 11, 6) are taken from the example of the previous figure, but of course they change from one geometry to another.

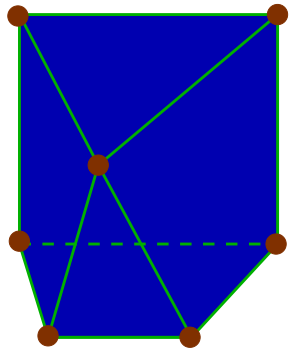
The motivation for working on a single element is that we want to discuss *scalar products*. All the scalar products that are going to be used on the whole structure (made, obviously, of several polyhedra) will be constructed *separately* in each element, and then summed over the elements.

We are going to consider four types of unknowns:

- node unknowns (whose values are defined only at the vertices and are to be interpreted as the values of a scalar function at each node)
- edge unknowns (whose value is defined only on the edges and are to be interpreted as the *integral of the tangential component* of a vector valued-function on each edge)
- face unknowns (whose values are defined only on the faces and are to be interpreted as the *integral of the outward normal component* of a vector-valued function on each face)
- element unknowns (whose values are defined only inside the element and are to be interpreted as the *integral* of a scalar function over the element)

Accordingly, we denote

- by $\mathcal{N}$ the space of all node unknowns
- by $\mathcal{E}$ the space of all edge unknowns
- by $\mathbf{F}$ the space of all face unknowns
- by $\mathcal{P}$ the space of all element unknowns



The dimensions of the above spaces will be, in our case, 7, 11, 6 and 1, respectively.

These spaces can obviously be interpreted as well as discretizations of the spaces of $0-forms$, $1-forms$, $2-forms$, and $3-forms$, respectively.

Although we did not say it explicitly, the space $\mathcal{E}$ will require an orientation of the tangent vector to edges, and, if more than one element is considered, the space $\mathbf{F}$ will require an orientation of faces. Here we have only one element, and we can choose the outward normal to each face, once and for all.

THE $\mathcal{GRAD}^h$ operator

It is now natural (according to our *interpretation*) to construct three basic differential operators:

- The $\mathcal{GRAD}^h$ operator, from $\mathcal{N}$ to $\mathcal{E}$, defined as follows:

  for each edge e with vertices V_1 and V_2, and for each element $u \in \mathcal{N}$, we define $\mathcal{GRAD}^h u$ *on* e *in the direction from* V_1 *to* V_2 *as*

$$\mathcal{GRAD}^h u_{|e} = u(V_2) - u(V_1)$$

THE $\mathcal{CURL}^h$ operator

- The $\mathcal{CURL}^h$ operator, from $\mathcal{E}$ to $\mathbf{F}$, defined as follows:

  for each element $\varphi \in \mathcal{E}$ and for each face f we orient the edges $e_1, e_2, ..., e_n$ of f in the counterclockwise direction (seen from outside the element), and then we consider the corresponding values $\varphi_1, \varphi_2, ..., \varphi_n$ of φ with the sign corresponding to the orientation just chosen. Then *the value of* $\mathcal{CURL}^h \varphi$ *on the face* f *is defined as*

$$\mathcal{CURL}^h \varphi_{|f} = \sum_{i=1}^{n} \varphi_i$$

THE $\mathcal{DIV}^h$ operator

- The $\mathcal{DIV}^h$ operator, from $\mathbf{F}$ to $\mathcal{P}$, defined as follows:

  let $f_1, ..., f_k$ be all the faces of our element P, and for each $\sigma \in \mathbf{F}$ let $\sigma_1, ..., \sigma_k$ be its values on each face. Then $\mathcal{DIV}^h\sigma$ *is defined as*

$$\mathcal{DIV}^h\sigma = \sum_{i=1}^{k} \sigma_i$$

  Note: in our example, $k = 6$. Of course. If we had more than one element, the same construction would be done for each element, taking care to use the values of σ corresponding to the *outward* direction.

THE ASSOCIATED MATRICES

It is interesting to note that, taking in the spaces $\mathcal{N}$, $\mathcal{E}$, $\mathbf{F}$, $\mathcal{P}$ the obvious canonical basis (after choosing an orientation of the edges in an arbitrary way, but once and for all), then the matrices associated with each one of the operators $\mathcal{GRAD}^h$, $\mathcal{CURL}^h$, and $\mathcal{DIV}^h$ are just made of zeroes, ones, and minus ones.

All this can by now be considered as *classical*. I will just mention the works of Shashkov and co-authors, Kuznetsov, Bochev, Yotov, Wheeler, Christiansen, Buffa, and many others.

THE INTERPOLANTS

Along the same way of thought, for each smooth scalar function u and for each smooth vector valued function $\boldsymbol{\theta}$ we can define the following *interpolants*

- $u_{I,node} \in \mathcal{N}$ defined by

$$u_{I,node}(V) = u(V) \qquad \text{for all vertex } V$$

- $\theta_{I,edge} \in \mathcal{E}$ defined by

$$\theta_{I,edge}|_e = \int_e \boldsymbol{\theta} \cdot \mathbf{t} \mathrm{d}s \qquad \text{for all edge } e$$

  where the unit tangent vector $\mathbf{t}$ indicates the orientation of e

THE INTERPOLANTS-2

- $\theta_{I,face} \in \mathbf{F}$ defined by

$$\theta_{I,face}|_f = \int_f \boldsymbol{\theta} \cdot \mathbf{n} \mathrm{d}S \qquad \text{for all face } f$$

  where $\mathbf{n}$ is the unit normal outward vector to f

- $u_{I,poly} \in \mathcal{P}$ defined by

$$u_{I,poly} = \int_P u \mathrm{d}P.$$

THE COMMUTATION PROPERTIES

Note that the interpolation operators and the differential operators introduced above have interesting commutation properties. Namely

- $\mathcal{GRAD}^h(u_{I,node}) = (\text{grad}u)_{I,edge}$
- $\mathcal{CURL}^h(\boldsymbol{\theta}_{I,edge}) = (\text{curl}\boldsymbol{\theta})_{I,face}$
- $\mathcal{DIV}^h(\boldsymbol{\theta}_{I,face}) = (\text{div}\boldsymbol{\theta})_{I,poly}$.

THE COMMUTING DIAGRAM

The above properties could be summarized saying that the following diagram

$$
\begin{array}{ccccccc}
C^\infty(P) & \xrightarrow{\text{grad}} & \left(C^\infty(P)\right)^3 & \xrightarrow{\text{curl}} & \left(C^\infty(P)\right)^3 & \xrightarrow{\text{div}} & C^\infty(P) \\
\downarrow & & \downarrow & & \downarrow & & \downarrow \\
\mathcal{N} & \xrightarrow{\mathcal{GRAD}^h} & \mathcal{E} & \xrightarrow{\mathcal{CURL}^h} & \mathbf{F} & \xrightarrow{\mathcal{DIV}^h} & \mathcal{P}
\end{array}
$$

(where the vertical arrows are the interpolation operators just defined), commutes.

All this *should* be related to the approximation of differential complexes intensively studied, among others, by Arnold, Falk, and Winther. See the recent review in *Acta Numerica*.

TRANSPOSED OPERATORS

As we are dealing with finite dimensional spaces, we can introduce immediately the dual spaces of the spaces $\mathcal{N}$, $\mathcal{E}$, $\mathbf{F}$, $\mathcal{P}$, and transpose the operators $\mathcal{GRAD}^h$, $\mathcal{CURL}^h$, and $\mathcal{DIV}^h$.

It seems rather natural (although one might question it) to denote by $\widetilde{\mathcal{GRAD}}^h : \mathcal{P}' \to \mathbf{F}'$ the transposed of the operator $-\mathcal{DIV}^h$, and by $\widetilde{\mathcal{DIV}}^h : \mathcal{E}' \to \mathcal{N}'$ the transposed of the operator $-\mathcal{GRAD}^h$. Similarly we shall denote by $\widetilde{\mathcal{CURL}}^h : \mathbf{F}' \to \mathcal{E}'$ the transposed of $\mathcal{CURL}^h$.

The dual spaces $\mathcal{P}'$, $\mathbf{F}'$, $\mathcal{E}'$ $\mathcal{N}'$ could then be thought as (*weird*) approximations of $0-forms$, $1-forms$, $2-forms$, and $3-forms$, respectively.

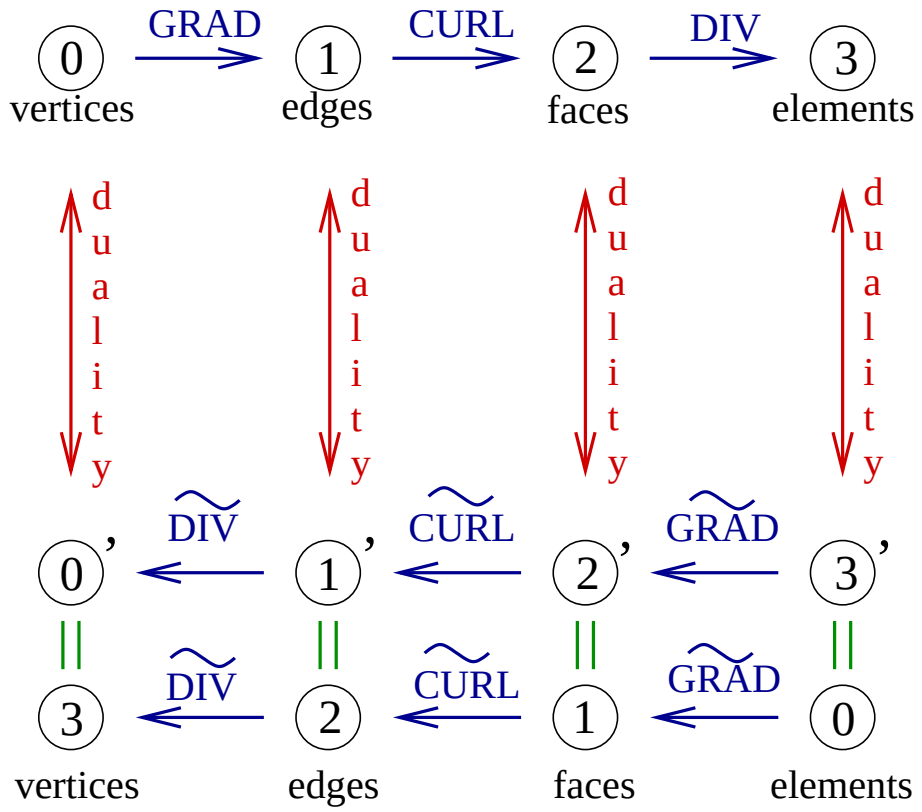
TRANSPOSED OPERATORS

In this way, we have the following diagram

$$
\begin{array}{ccccccc}
C^\infty(P) & \xrightarrow{\text{grad}} & \left(C^\infty(P)\right)^3 & \xrightarrow{\text{curl}} & \left(C^\infty(P)\right)^3 & \xrightarrow{\text{div}} & C^\infty(\\
\downarrow & & \downarrow & & \downarrow & & \downarrow \\
\mathcal{N} & \xrightarrow{\mathcal{GRAD}^h} & \mathcal{E} & \xrightarrow{\mathcal{CURL}^h} & \mathbf{F} & \xrightarrow{\mathcal{DIV}^h} & \mathcal{P} \\
\updownarrow & & \updownarrow & & \updownarrow & & \updownarrow \\
\mathcal{N}' & \xleftarrow{\widetilde{\mathcal{DIV}}^h} & \mathcal{E}' & \xleftarrow{\widetilde{\mathcal{CURL}}^h} & \mathbf{F}' & \xleftarrow{\widetilde{\mathcal{GRAD}}^h} & \mathcal{P}'
\end{array}
$$

DISCRETE DIFFERENTIAL FORMS

The previous diagram can be thought as made of *discretized differential forms*



INNER PRODUCTS

The previous discretizations of differential forms is quite appealing (and meaningful) on the first line. In order to transform the last two lines in a useful instrument, we need however to introduce suitable inner products in the spaces $\mathcal{N}$, $\mathcal{E}$, $\mathbf{F}$, $\mathcal{P}$.

If these inner products will be *consistent* with the $L^2(\Omega)$ inner product (for scalars) or with the $(L^2(\Omega))^2$ inner product (for vectors), then the adjoint operators will start to make sense in a more familiar way.

In doing this, however, it will be more convenient to insert the *material properties* into our inner products. This will lead us to a discretization of the $*-Hodge$ operator.

All this is vague. Let us consider a particular case in order to make it clearer.

SCALAR PRODUCTS in $\mathcal{N}$ (L^2-TYPE)

We already saw convenient scalar products in the spaces of $3-forms$ (elements) and of $2-forms$ (faces). Let us see some other scalar products.

First one might think of an L^2-inner product. For that it seems reasonable to request a 7×7 matrix. Denoting by $M_{\mathcal{N}}$ such matrix, it seems reasonable to require that

$$\sum_{i=1}^{7}\sum_{j=1}^{7}(M_{\mathcal{N}})_{i,j}u(V_i)v(V_j) = \int_P u\,v\,\mathrm{d}P$$

whenever u and v are the interpolant of two polynomials of degree ≤ 1 (u and v, respectively). To construct such a matrix $M_{\mathcal{N}}$ we can proceed as before (solving explicitly with a semi-positive definite matrix, and then adding a suitable correction that makes it positive definite).

SCALAR PRODUCTS in $\mathcal{N}$ (H^1-TYPE)

We proceed to introduce a scalar product in $\mathcal{E}$ for gradients. The dimension of the subspace of gradients is clearly 6 (or, the number of vertices minus one). We denote by $\mathcal{E}^G$ the subspace of $\mathcal{E}$ made of gradients. We would like to construct a symmetric 11×11 matrix $M^G_{\mathcal{E}}$ (SPD on $\mathcal{E}^G$) such that: for every polynomial v of degree ≤ 1, and for every $\varphi = \mathcal{GRAD}^h u$ (with $u \in \mathcal{N}$), setting $\psi := (\mathrm{grad} v)_{I,edge}$ $\equiv \mathcal{GRAD}^h v_{I,node}$ we have

$$\sum_{i=1}^{11}\sum_{j=1}^{11}(M^G_{\mathcal{E}})_{i,j}\varphi_i\,\psi_j = \int_{\partial P} u\frac{\partial v}{\partial \mathbf{n}}.$$

There is a problem with the above formula: Since u is only defined at the nodes, we cannot evaluate the integral over the faces. Hence we need to pick up a numerical integration formula. In most cases we only need such formula to be exact for linear functions.

SCALAR PRODUCTS in $\mathcal{N}$ (H^1-TYPE)

If the matrix $M_{\mathcal{E}}^G$ satisfies the previous formula one can prove that: there exists a lifting linear mapping $r : \mathcal{N} \to C^0(\bar{P})$ such that

i) $(r(w))_{I,node} \equiv w$ for each $w \in \mathcal{N}$,

ii) the value of $r(u)$ on each face f depends only on the values of u at the nodes belonging to f,

iii) for every linear function v we have $r(v_{I,node}) = v$,

iv) and moreover:

$$\sum_{i=1}^{11}\sum_{j=1}^{11}(M_{\mathcal{E}}^G)_{i,j}(\mathcal{GRAD}^h(u))_i\,(\mathcal{GRAD}^h(v_{I,node}))_j = \int_{\partial P} r(u)\frac{\partial v}{\partial \mathbf{n}}.$$

The best way to complete the scalar product in $\mathcal{E}$ beyond the gradients is still a rather open question.

MUMBLE GRUMBLE.