Prof. Dr. Peter Bastian, Michal Tóth
IWR, Universität Heidelberg

Submission date: 24 Jan 2022

---

Note: Do not forget to update your dune installation as described in exercise sheet 2.

**Exercise 1** *Inverse error estimate for finite elements*

This exercise is about proving a generalization of Proposition 8.12 (Inverse Estimate) from the lecture notes. Let $\{\mathcal{T}_\nu\}$ be a family of affine and shape regular triangulations of domain $\Omega$ with corresponding finite element spaces $P_k(\mathcal{T}_\nu)$, and $h_\nu$ be the length of the shortest edge from the set of longest edges of simplices of $\mathcal{T}_\nu$.

Prove, that there exists a constant $c$ (independent of $h_\nu$, but depending on $k$, $l$, $m$, grid regularity, dimension, etc.) such that

$$|v_h|_{l,\Omega} \leq c\, h_\nu^{m-l} |v_h|_{m,\Omega} \quad \forall v_h \in P_k(\mathcal{T}_\nu)$$

for any integers $0 \leq m < l$.

This is an inverse error estimate, meaning that the higher order seminorm is estimated by the lower order seminorm. Such estimates can be made only on finite-dimensional spaces (like $P_k$). You can use the claim that norms on these spaces are equivalent, i.e.

$$\|w_h\|_{l,\hat{T}} \leq c \|w_h\|_{m,\hat{T}} \quad \forall w_h \in P_k(\hat{T})$$

with $c$ depending on $k, m, l$ and $\hat{T}$, where $\hat{T}$ is the reference element.

Another theorem you might find useful is Bramble-Hilbert lemma.

**( 6 Points )**

**Exercise 2** *Convection-Diffusion Problem*

In *uebungen/uebung11* of your *dune-npde* module you can find a program that solves a convection-diffusion problem

$$\begin{aligned} -\nabla \cdot (k(x)\nabla u) + a(x) \cdot \nabla u = 0 \qquad & x \in \Omega \\ u(x) = g(x), \quad & x \in \partial\Omega_D \\ (a(x) - k(x)\nabla u) \cdot n = j(x), \quad & x \in \partial\Omega_N \end{aligned}$$

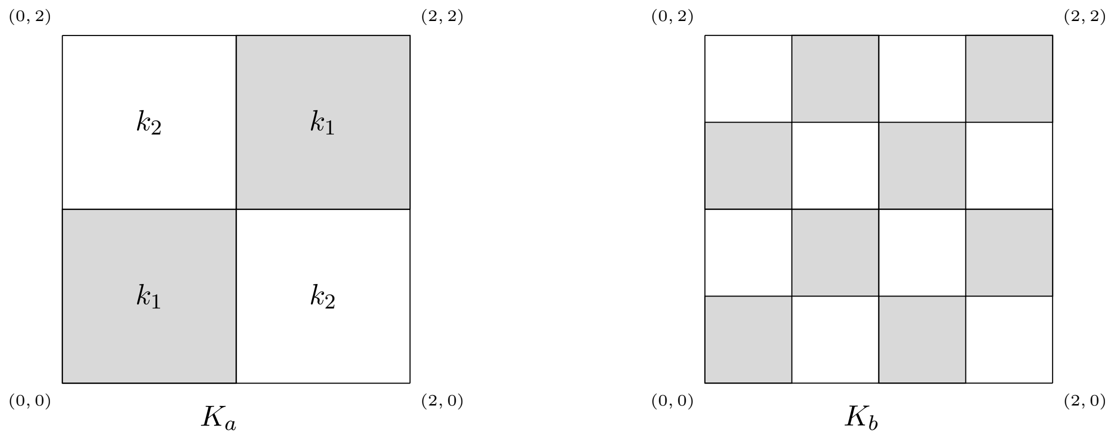
using $Q^1$ and $Q^2$ finite elements on domain $\Omega = [0,2] \times [0,2] \subset \mathbb{R}^2$.

1. The main difference of this program compared to previous ones is that it implements its own *LocalOperator* in *convectiondiffusion.hh*, instead of using a predefined one. Figure out how the *LocalOperator* retrieves the parameters of the PDE it represents. Also, write down the numerical integrations it performs and check if it matches what you expect from the bilinear form.

2. First of all we will solve only a diffusion problem ($a(x) = \vec{0}$) with boundary conditions

   $$u = 1 \text{ for } x_1 = 0,\ u = -1 \text{ for } x_1 = 2,\ \text{(Dirichlet on the left and on the right side)}$$
   $$\nabla u \cdot n = 0 \text{ otherwise.}$$

   The permeability field is heterogeneous. In the program we used permeability field $K_a$. Your task is to implement permeability field $K_b$, see picture.




3. Have a look at the function *flux* (you can find it in the file *utilities.hh*). What does the function compute?

4. Compare the results of *flux* function for $K_a$ and $K_b$ with coefficients $k_1 = 3.10^{-4}$, $k_2 = 10^{-1}$ for different refinement level and polynomial degrees. Does it converge to some value?

5. Now we will add advection. Set $a = (1,0)^T$ (parameter $convection = 1$ in *uebung11.ini* and $k_1 = k_2 = 10^{-3}$. What do you observe in the solution for different grid refinements? Can you explain these phenomena?

**( 10 Points )**