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

Submission date: 17 Jan 2022

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

Exercise 1 *The best approximation in the energy norm*

Let V be a Hilbert space, $a : V \times V \to \mathbb{R}$ be a coercive continuous bilinear form, and $l \in V'$. Let $V_h \subset V$ be closed. Let u be the solution of the problem

$$a(u, v) = l(v) \quad \forall v \in V,$$

and $u_h \in V_h$ be the solution of the problem

$$a(u_h, v) = l(v) \quad \forall v \in V_h.$$

Moreover, let a be symmetric. Then $a(\cdot, \cdot)$ is a scalar product, and we denote the norm generated by it with lower index a, i.e. $\|v\|_a = \sqrt{a(v, v)}$. This norm is called the energy norm.

Prove that

$$\|u - u_h\|_a = \inf_{v_h \in V_h} \|u - v_h\|_a.$$

(4 Points)

Exercise 2 *Interpolation on triangle*

Let $v \in C^2(K)$ and K be a triangle with vertices $a_1, a_2, a_3 \in \mathbb{R}^2$. Functions φ_i for $i = 1, 2, 3$ denote $P^1(K)$ basis functions satisfying $\varphi_i(a_j) = \delta_{ij}$. The longest side of triangle K is h_K and the smallest angle is τ_K. The P^1-interpolation function has a form

$$\Pi v(x) = \sum_{i=0}^{3} v(a_i)\varphi_i(x).$$

Prove the following estimations:

1. $$\|v - \Pi v\|_{L_\infty(K)} \leq \frac{1}{2} h_K^2 \|D^2 v\|_{L_\infty(K)}$$

   Hint: Use a Taylor expansion.

2. $$\|\nabla(v - \Pi v)\|_{L_\infty(K)} \leq \frac{3}{\sin \tau_K} h_K \|D^2 v\|_{L_\infty(K)}$$

   Hint: Similar approach. What is the maximum gradient a basis function can have on the given element?

(7 Points)

Exercise 3 *Loss of coercivity, cross-point problem*

In *uebungen/uebung10* of your *dune-npde* module you can find a program that solves so-called cross point problem (see lecture notes Example 8.20) with P^k finite element on a conforming trianglular grid (*UGGrid*) and with Q^k finite element on a conforming quadrilateral grid (*YaspGrid*).

You can change input parameters in file *uebung10.ini*.

1. In *problem.hh*, read the parameters *k1,k2* from the .ini file, replacing the hard-coded constants.
2. Now adjust the matrix A defined in the problem class to match the cross point problem. In particular, you have to include k appropriately.
3. In the definition of the boundary condition, complete the implementation of the analytical solution given in the script. Run the program and verify your code by checking via paraview if analytical and numerical solution are a close match.
4. Where in the domain is the singularity (look at $u - u_h$ in paraview)?
5. In homogeneous case ($k_1 = 1, k_2 = 1$), the convergence rates are kind of strange. Can you explain it?
6. Run the programm for different polynomial degrees 1 and 2 with different permeabilities ($k_1 = 1, k_2 = \{1, 100, 10000\}$). Which behaviour in convergence rates do you observe? Describe it qualitatively or create a table/plot as in lecture notes.
7. Change your grid to unstructured triangular grid and choose $k_1 = 1, k_2 = 10$. Why are convergence rates not as expected?

(10 Points)