Prof. Dr. Carsten Burstedde
Uta Seidler and Denis Düsseldorf

UNIVERSITÄT BONN

Exercise sheet 7.

Submission (theory) by **08.12.2022**.

Exercise 18. (Sobolev Spaces)

(6 Points)

a) Let $\Omega \subset \mathbb{R}^d$ be an open set. Show that $H^1(\Omega)$ is a Hilbert space.

b) Show that there is no constant $C > 0$ such that

$$\|f\|_{L^2(\Omega)} \leq C\|\nabla f\|_{L^2(\Omega)} \quad \text{for all } f \in H^1(\Omega).$$

c) Suppose that Ω is a bounded Lipschitz domain and let $1 \leq p \leq \infty$. We define $|\cdot|_{W^{1,p}(\Omega)}$ as the seminorm

$$|f|_{W^{1,p}(\Omega)} = \left(\sum_{\alpha\in\mathbb{N}_0^d, |\alpha|=1} \left\| \frac{\partial^{|\alpha|} f}{\partial x^\alpha} \right\|_{L^p(\Omega)}^p \right)^{1/p} \qquad \text{for } p < \infty,$$

$$|f|_{W^{1,p}(\Omega)} = \max_{\alpha\in\mathbb{N}_0^d, |\alpha|=1} \left\| \frac{\partial^{|\alpha|} f}{\partial x^\alpha} \right\|_{L^\infty(\Omega)} \qquad \text{for } p = \infty.$$

Show that the $|\cdot|_{W^{1,p}(\Omega)}$ and $\|\cdot\|_{W^{1,p}(\Omega)}$ are equivalent on $W_0^{1,p}(\Omega)$, i.e., there are constants $C, c > 0$ which can depend on the domain Ω and p such that

$$c|f|_{W^{1,p}(\Omega)} \leq \|f\|_{W^{1,p}(\Omega)} \leq C|f|_{W^{1,p}(\Omega)} \quad \text{for all } f \in W_0^{1,p}(\Omega).$$

Exercise 19. (Singularities of H^1 functions)

(0 Points)

Let $\Omega = B_1(0) \subset \mathbb{R}^2$ be the open unit ball around the origin. Show that

$$u(x) = \ln\left(\ln\left(\frac{2}{|x|}\right)\right)$$

is in $H^1(\Omega)$.

Programming exercise 2. (Piecewise linear finite elements)

(12 Points)

The goal of this programming exercise is to solve the Poisson problem with mixed boundary conditions,

$$\begin{aligned} -\Delta u &= f \quad \text{in } \Omega, \\ u &= g \quad \text{on } \partial\Omega_D, \\ \nabla u \cdot \nu &= 0 \quad \text{on } \partial\Omega_N, \end{aligned} \tag{1}$$

using a piecewise linear finite element method. The domain $\Omega \subset \mathbb{R}^d$ and our degrees of freedom shall be encoded by lookup tables, and we would like to write code only once that can be adapted easily to 1D and 2D as well as to new choices for the Dirichlet and Neumann boundaries.

a) Consider a triangulation $\mathcal{T}$ of Ω and store the relevant information for this triangulation $\mathcal{T}$. By this we mean a division of Ω into line segments (1D) or triangles (2D) T_k, $k = 1, \ldots, K$ where each simplex is defined by $M = d + 1$ vertices $z_{i(k,r)}$, $r = 1, \ldots, M$.

   We encode the topology of the discretized domain by the map from local to global indices. For each element T_k, we can identify any local index $r \in \{1 \ldots, M\}$ with one matching global index $i \in \{1, \ldots, I\}$ where I is the total number of vertices in $\mathcal{T}$. To this end, define the following table in the program:

$$i = i_k(r) = i(k, r) \quad \text{for } k = 1, \ldots, K \text{ and } r = 1, \ldots, M. \tag{2}$$

   The global indices are used to compose the numerical solution using global basis functions,

$$u_h(x) = \sum_{i=1}^{I} u_i \psi_i(x). \tag{3}$$

   We will also separate the global degrees of freedom into two nonempty sets. The first set of cardinality $L < I$ contains interior (volume) vertices and vertices on the Neumann boundary and the second set the complementary Dirichlet set. To this end, we need to configure two non-intersecting lists,

$$i_V = i_V(j) \quad \text{where } j = 1, \ldots, L,\ i_V \in 1, \ldots, I, \tag{4a}$$

$$i_D = i_D(j') \quad \text{where } j' = 1, \ldots, I - L,\ i_D \in 1, \ldots, I. \tag{4b}$$

   Consider different discretizations of one- and two-dimensional domains with different boundary conditions. Create the lists described in (2) and (4). For example, in the case where $\Omega = (0, 1)^2$, you may consider the discretization where the square is divided into $2n^2$ similar triangles (see Figure 1).

b) We pull each of the T_k back to the reference element T_0 with vertices ξ_r, $r = 1, \ldots, M$. In 2D we choose as reference element the convex hull of the points $\xi_1 = (0, 0)$, $\xi_2 = (1, 0)$ and $\xi_3 = (0, 1)$, and for the 1D case the line segment between $\xi_1 = 0$ and $\xi_2 = 1$.

   On this reference simplex, we define nodal linear functions Ψ_r, that is, $\Psi_r(\xi_s) = \delta_{rs}$, $r, s \in \{1, \ldots, M\}$. Moreover, we define for each element T_k an affine linear transformation

$$F_k : T_0 \to T_k \tag{5}$$

   such that the basis functions ψ_i over this element satisfy

$$\Psi_r \circ F_k^{-1} = \psi_{i(k,r)}|_{T_k}. \tag{6}$$

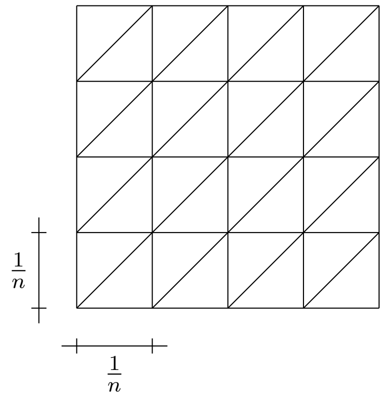

Figure 1: (Uniform) Triangulation of the unit square $(0,1)^2$.

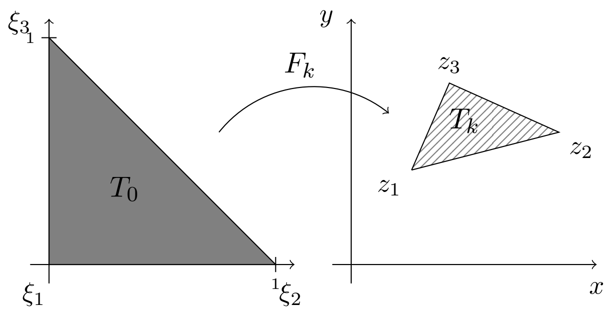

Figure 2: Reference element T_0 and transformation F_k onto a triangle T_k.

For each element T_k compute the local mass and stiffness matrices M_{T_k}, $A_{T_k} \in \mathbb{R}^{M\times M}$ as

$$(M_{T_k})_{rs} = \int_{T_k} \psi_{i(k,r)}\psi_{i(k,s)}\,\mathrm{d}x, \tag{7a}$$

$$(A_{T_k})_{rs} = \int_{T_k} \nabla\psi_{i(k,r)} \cdot \nabla\psi_{i(k,s)}\,\mathrm{d}x \tag{7b}$$

for $r, s = 1, \ldots, M$, using the transformation theorem to integrate over T_0. Write a routine that carries out the matrix-vector multiplication for the global stiffness matrix $\mathbf{A} \in \mathbb{R}^{I\times I}$ given by

$$\mathbf{A}_{i(k,r),i(k,s)} = \sum_{k=1}^{K} (A_{T_k})_{r,s}, \tag{8}$$

and analogously for the global mass matrix $\mathbf{M}$. Test your implementation by verifying that for the I-vector $\mathbf{e}$ of all ones

$$\mathbf{e}^T\mathbf{M}\mathbf{e} = |\Omega|, \qquad \mathbf{e}^T\mathbf{A}\mathbf{e} = 0. \tag{9}$$

c) Write a routine that returns the local load vector $f_{T_k} \in \mathbb{R}^M$ for each element T_k, $k = 1, \ldots, K$ with

$$(f_{T_k})_r = \int_{T_k} f\psi_{i(k,r)}\,\mathrm{d}x \tag{10}$$

for $r = 1, \ldots, M$. To this end, transform the integral to the reference triangle T_0 and approximate it with the quadrature

$$\int_{T_0} f(x,y)\,\mathrm{d}x\,\mathrm{d}y \approx \frac{1}{6}\left[f\left(\frac{1}{6},\frac{1}{6}\right) + f\left(\frac{4}{6},\frac{1}{6}\right) + f\left(\frac{1}{6},\frac{4}{6}\right)\right] \quad \text{in 2D and} \tag{11a}$$

$$\int_{T_0} f(x)\,\mathrm{d}x \approx \frac{1}{2}\left[f\left(\frac{\sqrt{3}-1}{2\sqrt{3}}\right) + f\left(\frac{\sqrt{3}+1}{2\sqrt{3}}\right)\right] \quad \text{in 1D.} \tag{11b}$$

Combine them into a global load vector $\mathbf{f} \in \mathbb{R}^I$.

d) Use the conjugate gradient method from the last programming exercise to solve the discrete problem. We utilize the projector $\mathbf{P}_V \in \mathbb{R}^{I \times I}$ that sets all Dirichlet entries of a vector to zero and its complement projector $\mathbf{P}_D = \mathrm{id} - \mathbf{P}_V$. With given Dirichlet values $(\mathbf{u}_D)_i = g(z_i)$, $i = i_D(j')$, and $(\mathbf{u}_D)_i = 0$, $i = i_V(j)$, solve the equation

$$(\mathbf{P}_V \mathbf{A} \mathbf{P}_V + \mathbf{P}_D)\, \mathbf{u} = \mathbf{P}_V\, (\mathbf{f} - \mathbf{A}\mathbf{u}_D) + \mathbf{u}_D. \tag{12}$$

Plot the solution that you obtained. Try your code for different problems. To this end, you can choose a true solution u and define the right hand side and boundary conditions such that (1) is satisfied. Then use your code to find an approximation u_h to u. Use the triangulation shown in Figure 1 and also consider other discretizations.

e) Bonus: Now we want to measure the error between the approximate solution u_h and the true solution u. First, use the quadrature rule in part (c) to evaluate $\|\nabla(u - u_h)\|_{L^2(\Omega)}$. Alternatively, interpolate the true solution u to the finite element space and use an application of the stiffness matrix to compute $\|\nabla(u - u_h)\|_{L^2(\Omega)}$. Compare the two errors. Vary the number of elements and examine the order of convergence for different choices of u.

f) Bonus: Run your code also for non-rectangular domains.

You have two weeks to turn in the programming exercises, in this case until Monday, December 19th, 2022.