Numerical Algorithms

Winter semester 2018/2019
Prof. Dr. Marc Alexander Schweitzer
Denis Düsseldorf

universitätbonn

Exercise sheet 2. Submission on **Tuesday, 2018-10-30, before the lecture**.

Exercise 7. (Converse of a generalized Céa lemma)

Let U, V be Hilbert spaces and $a : U \times V \to \mathbb{R}$ be a bilinear form fulfilling

1. Continuity
$$\exists C > 0 : \forall u \in U : \forall v \in V : \qquad |a(u,v)| \leq C \|u\|_U \|v\|_V \ .$$
2. inf − sup condition
$$\exists \alpha > 0 : \qquad \inf_{u \in U} \sup_{v \in V} \frac{a(u,v)}{\|u\|_U \|v\|_V} \geq \alpha \ .$$
3. and
$$\forall v \in V : \exists u \in U : \qquad a(u,v) \neq 0 \ .$$

Further let $f \in V'$ be a linear functional from V's dual and let $u \in U$ be a solution of

$$a(u,v) = f(v) \ , \qquad \forall v \in V \ . \tag{1}$$

If $U_h \subset U$ and $V_h \subset V$ are finite dimensional subspaces, chosen such that 1.–3. also hold when replacing U by U_h and V by V_h, then for the finite dimensional solution $u_h \in U_h$ of

$$a(u_h, v_h) = f(v_h) \ , \qquad \forall v_h \in V_h \tag{2}$$

we have a generalized Céa lemma

$$\|u - u_h\|_U \leq \left(1 + \frac{C}{\alpha}\right) \inf_{w_h \in U_h} \|u - w_h\|_U \ .$$

Show the following converse result:
Let

$$L : U \to V' \ , \qquad (Lu)(v) := a(u,v) \ .$$

Assume that $\forall f \in V'$ and $\forall h$ there exist solutions $u = L^{-1} f \in U$ of (1), and u_h of (2) and further assume that

$$\lim_{h \to 0} u_h = u := L^{-1} f \ .$$

Then it holds that

$$\inf_h \inf_{u_h U_h} \sup_{v_h \in V_h} \frac{a(u_h, v_h)}{\|u_h\|_U \|v_h\|_V} > 0 \ .$$

Hint: Consider the linear and bounded operators (why?) $K_h : V' \to U_h, f \mapsto u_h$ and use the uniform boundedness principle, i.e.

$$\left(\forall f \in V' : \sup_h \|K_h f\| < \infty\right) \Rightarrow \sup_h \|K_h\| < \infty \ .$$

(4 Points)

Exercise 8. (inf − sup condition and ellipticity)

Let V be a Hilbert space and $a : V \times V \to \mathbb{R}$ be a positive, symmetric, bilinear form. Assume further that

1. a is continuous, i.e.
$$\exists C > 0 : \forall u, v \in V : \qquad |a(u,v)| \leq C\|u\|_V \|v\|_V \; .$$
2. a, V fulfill an inf − sup-condition
$$\exists \alpha > 0 : \qquad \inf_{u \in V} \sup_{v \in V} \frac{a(u,v)}{\|u\|_V \|v\|_V} \geq \alpha \; .$$

Show that a is elliptic.

(3 Points)

Exercise 9. (P^1 assembly via reference element)

We revisit stiffness matrix assembly for the regular Lagrange P^1 triangle, i.e. linear polynomials on triangles with interpolation conditions in the corner. Let a_0, a_1, a_2 be the corners of an arbitrary non-degenerate (a_i linearly independent) T and $r_0 = (0,0), r_1 = (1,0), r_2 = (0,1)$ the corners of the reference triangle T_{ref}.

a) Explicitly state the affine map $F_T : T_{ref} \to T$ such that $F_T(r_i) = a_i$, i.e. explicitly find $B_T \in \mathbb{R}^{2\times 2}, x_T \in \mathbb{R}^2$ such that
$$B_T r_i + x_T = a_i$$
and give F_T^{-1}.

b) The shape functions on T_{ref} are given by
$$\varphi_0(x,y) = 1 - x - y \; , \qquad \varphi_1(x,y) = x \; , \qquad \varphi_2(x,y) = y \; .$$
and hence the shape functions on T are given by $L_{i,T} = \varphi_i \circ F_T^{-1}$. Compute $L_{i,T}$ and $\nabla L_{i,T}$ in global coordinates, i.e. expand $L_{i,T}, \nabla L_{i,T}$.

c) Show that
$$\int_T \nabla L_{i,T} \cdot \nabla L_{j,T} = \frac{1}{2} |\det B| \nabla \varphi_i \cdot (B^T B)^{-1} \nabla \varphi_j \; ,$$

d) Show that the local stiffness matrix K_T for a single triangle with vertices
$$a_0 = (x_0, y_0) \; , \qquad a_1 = (x_1, y_1) \; , \qquad a_2 = (x_2, y_2)$$
can be computed via
$$K_T = \frac{1}{2} \left| \det \begin{pmatrix} x_1 - x_0 & x_2 - x_0 \\ y_1 - y_0 & y_2 - y_0 \end{pmatrix} \right| G G^T \qquad G := \begin{pmatrix} 1 & 1 & 1 \\ x_0 & x_1 & x_2 \\ y_0 & y_1 & y_2 \end{pmatrix}^{-1} \begin{pmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{pmatrix}$$

(5 Points)

Programming exercise 3. (Matrix assembly)

Please use a kind of standard editor to create regular .py files (see Virtual-Studio Code, Sublime, Notepad++, Vim, ...) - no notebook files!

Using the Python knowledge and data structures established programming exercises 1) and 2) and mainly put exercise 9 into code.

a) Write a function to determine the free degrees of freedom, i.e. indices of vertices with are not part of any edge on Dirichlet or Neumann boundary. Determine the free nodes from the example mesh from programming exercise 1.

b) Write a function to assemble the local stiffness matrix K_T given the indices of the vertices.

   Hint: `vstack` can be used to add lines to a given matrix.

c) Write a function to loop over all triangles adding contributions to assemble the global stiffness matrix K.

(4 Points)

Send to `duesseldorf@ins.uni-bonn.de`