Scientific Computing 1

Winter term 2017/18
Priv.-Doz. Dr. Christian Rieger
Christopher Kacwin

Sheet 8

Submission on **Thursday, 14.12.17**.

Exercise 1. (Friedrich's inequality)

Suppose Ω is an open and bounded domain, as well as starshaped with respect to a ball B. Show that for all $u \in H^1(\Omega)$ one has

$$\|u - [u]\|_{L^2(\Omega)} \leq C \operatorname{diam}(\Omega) \, |u|_{H^1(\Omega)} ,$$

where $[u] = \frac{1}{|\Omega|} \int_\Omega u(x) \,\mathrm{d}x$.

(4 points)

Exercise 2. (continuous data discretization)

Let $\Omega = [0, 1]^2$ and consider the PDE

$$\begin{aligned} -\Delta u &= f \qquad && \text{in } \Omega \\ u &= 0 && \text{on } \partial\Omega \end{aligned}$$

with $f \in H^1(\Omega)$ and solution $u \in H^3(\Omega)$. Moreover, let $\mathcal{T}_h$ be a quasi-uniform, nondegenerate triangulation of Ω with $h = \max_{T \in \mathcal{T}} \operatorname{diam}(T)$. Define $[f](x) \in L^2(\Omega)$ as the piecewise constant function

$$[f]|_T = \frac{1}{|T|} \int_T f(y) \,\mathrm{d}y \,.$$

Let V_h be the linear finite element space with respect to $\mathcal{T}_h$ and replace f with $[f]$ in the above PDE, i.e. we consider:
Find $u_h \in V_h$ s.t. for all $v_h \in V_h$

$$\int_\Omega \nabla u_h(x) \cdot \nabla v_h(x) \,\mathrm{d}x = \int_\Omega [f](x) v_h(x) \,\mathrm{d}x \,.$$

Prove the convergence rate

$$\|u - u_h\|_{H^1(\Omega)} = \mathcal{O}(h) \,.$$

(4 points)

Programmieraufgabe 1. (local load vector/ finite element mesh)

We consider the reference triangle element $T_0 = \{(x, y)^\top \in \mathbb{R}^2 \mid x \geq 0, y \geq 0, x + y \leq 1\}$ with the three nodes $a_1 = (0, 0)^\top$, $a_2 = (1, 0)^\top$ and $a_3 = (0, 1)^\top$. Furthermore, let ϕ_1, ϕ_2, ϕ_3 be the linear nodal basis with respect to a_1, a_2, a_3. For an arbitrary nondegenerate triangle $T \subset \mathbb{R}^2$ with corners $b_1, b_2, b_3 \in \mathbb{R}^2$, we consider the affine linear map $J(x, y) = C(x, y)^\top + d$ which maps a_i to b_i for $i = 1, 2, 3$. With the new coordinates $(\hat{x}, \hat{y})^\top = J(x, y)$, the nodal basis ψ_1, ψ_2, ψ_3 on T is given via

$$\psi_i(\hat{x}, \hat{y}) = \phi_i(x, y) \,, \quad i = 1, 2, 3 \,.$$

a) Write a routine that takes $b_1, b_2, b_3 \in \mathbb{R}^2$ as input and returns the local load vector $l_T \in \mathbb{R}^3$, given by

$$(l_T)_i = \int_T \psi_i(\hat{x}, \hat{y}) \, \mathrm{d}(\hat{x}, \hat{y})$$

for $i = 1, 2, 3$.

Let $\Omega = [0,1]^2 \subset \mathbb{R}^2$ and consider the PDE

$$\begin{aligned} -m\Delta u + cu &= f & \text{in } \Omega \\ u &= 0 & \text{on } \partial\Omega \end{aligned}$$

with $f \in L^2(\Omega)$ and $m, c \in L^\infty(\Omega)$ (uniformly bounded from below by a constant $\epsilon > 0$). We discretize Ω with a triangular finite element mesh and assume the functions m, c, f to be piecewise constant on elements. Such a discretization is given to us as the datafile `mesh.txt` (on the homepage), which has the following format:

```
#Some comments
#...
#NODES:
#globalNodeNumber #xCoordinate #yCoordinate #onBoundary
0 0.0 0.0 1
...
7 1.0 0.5 1
#ELEMENTS:
#globalElementNumber #node1 #node2 #node3 #mValue #cValue #fValue
0 1 3 5 1.0 1.0 1.0
...
7 0 2 4 1.0 1.0 1.0
#
```

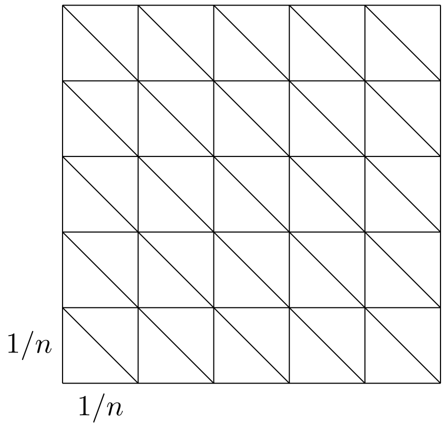
The # symbol denotes comment lines. The first part of the file lists all nodes with their global reference number, coordinates and whether they lie on the boundary ('1') or not ('0'). The second part lists all elements with their global reference number, the corner node numbers and the constant values for m, c, f.

b) Construct an object 'Node' that can contain the information given by the datafile, i.e. global reference number (nonnegative integer), x and y coordinate (double precision type) and boundary classification (boolean). Add a routine that prints out the contained information to the console.

c) Construct an object 'Element' that can contain the information given by the datafile, i.e. global reference number (nonnegative integer), three corner node numbers (nonnegative integer) and three constants for m, c, f (double precision type). Moreover, it should be able to contain the local stiffness matrix, local mass matrix and the local load vector (for linear elements). Add a routine that prints out the contained information to the console.

d) Write a routine that reads the data from the file and produces an array of 'Nodes' as well as an array of 'Elements' with complete information. Here, the node reference number should coincide with the position of the 'Node' in the array, same for the elements. Print the nodes and elements to the console to test your implementation.

e) Write a routine that takes functions $m, c, f\colon [0,1]^2 \to \mathbb{R}$ as well as a refinement parameter $n \in \mathbb{N}$ as input and produces a datafile in the same format as `mesh.txt` for the following mesh on $[0,1]^2$:



Here, the `mValue,cValue,fValue` parameters for an element should be computed as the average over the function values at the corner nodes, i.e. for an element T with corner nodes $b_1, b_2, b_3 \in \mathbb{R}^2$, we approximate m, c, f with piecewise constant functions defined via

$$\begin{aligned} m|_T &= 1/3(m(b_1) + m(b_2) + m(b_3))\,, \\ c|_T &= 1/3(c(b_1) + c(b_2) + c(b_3))\,, \\ f|_T &= 1/3(f(b_1) + f(b_2) + f(b_3))\,. \end{aligned}$$

Test your routine with $n = 3$ and arbitrary nonconstant functions m, c, f. Apply your routine from Exercise d) and print the nodes' and elements' information to the console.

(24 points)

The programming exercise should be handed in during the exercise classes (bring your own laptop!) right after the christmas break, i.e. on 11.1.18/12.1.18. All group members need to attend the presentation of your solution to get points.