Prof. Ch. Schwab
L. Herrmann, J. Zech

Fall Term 2017

Numerical Analysis of High-Dimensional Problems

ETH Zürich
D-MATH

Exercise 2

Problem 2.1 Continuous dependence on data

Let $\mathrm{D} \subseteq \mathbb{R}^d$ be a bounded Lipschitz domain and let $a \in L^\infty(\mathrm{D}, \mathbb{R})$ such that for two constants $0 < \underline{a} \leq \overline{a} < \infty$ it holds $\underline{a} \leq a \leq \overline{a}$ a.e. in D. Moreover let $f \in H^{-1}(\mathrm{D})$.

(2.1a) Denote by $u_{a,f} \in H_0^1(\mathrm{D})$ the unique solution of the variational problem

$$\int_{\mathrm{D}} a(x) \nabla u_{a,f}(x)^\top \nabla v(x) \, \mathrm{d}x = {}_{H^{-1}}\langle f, v \rangle_{H_0^1} \qquad \forall v \in H_0^1(\mathrm{D}). \tag{2.1.1}$$

Fix $R > 0$. Prove that $u_{a,f}$ is locally Lipschitz continuous as a function of $a \in L^\infty(\mathrm{D})$ and $f \in H^{-1}(\mathrm{D})$: there exists a constant C depending on $\underline{a}$ and R such that

$$\|u_{a,f} - u_{b,g}\|_{H_0^1(\mathrm{D})} \leq C(\|a - b\|_{L^\infty(\mathrm{D})} + \|f - g\|_{H^{-1}(\mathrm{D})}), \tag{2.1.2}$$

whenever $\|a - b\|_{L^\infty(\mathrm{D})} \leq \underline{a}/2$ and $\|g\|_{H^{-1}} \leq R$.

(2.1b) Let now $d \leq 3$ and consider the nonlinear problem of finding $u_{a,f} \in H_0^1(\mathrm{D})$ such that

$$\int_{\mathrm{D}} a(x) \nabla u_{a,f}(x)^\top \nabla v(x) + u_{a,f}(x)^3 v(x) \, \mathrm{d}x = {}_{H^{-1}}\langle f, v \rangle_{H_0^1} \qquad \forall v \in H_0^1(\mathrm{D}). \tag{2.1.3}$$

Show that there exists $\varepsilon > 0$ and a constant C depending on ε and $\underline{a}$ such that for all f, $g \in H^{-1}(\mathrm{D})$ with $\|f\|_{H^{-1}}, \|g\|_{H^{-1}} \leq \varepsilon$ and all $b \in L^\infty$ with $\|a - b\| \leq \underline{a}/2$ it holds (2.1.2).

Problem 2.2 Linear Finite Elements for univariate elliptic equations

For given $f \in L^2(\mathrm{D})$, $u_1 \in C^1(\overline{\mathrm{D}})$ and u_2, $u_3 \in C^0(\overline{\mathrm{D}})$ consider the linear elliptic equation

$$\begin{aligned} -(u_1 q')' + u_2 q' + u_3 q &= f \quad \text{in D}, \\ q &= 0 \quad \text{on } \partial\mathrm{D}, \end{aligned} \tag{2.2.1}$$

where $\mathrm{D} = (a, b)$ and $\operatorname{essinf}_{x \in \mathrm{D}} u_1(x) > 0$.

(2.2a) Derive the variational formulation of (2.2.1): reformulate the problem to find $q \in V := H_0^1(\mathrm{D})$ such that

$$a(u; q, v) = {}_{V^*}\langle f, v \rangle_V \quad \forall v \in V. \tag{2.2.2}$$

Give sufficient conditions on u_2 and u_3 such that (2.2.2) is uniquely solvable (use the Lax–Milgram lemma).

(2.2b) Consider the uniform mesh $x_i = a + hi$, $i = 0, \dots, N$ on D, where $h = (b-a)/N$, $N \in \mathbb{N}$. With $x_{-1} := a - h$, $x_{N+1} := b + h$, the *hat functions* are defined as

$$b_i(x) = \begin{cases} \frac{1}{h}(x - x_{i-1}) & \text{if } x \in [a,b] \text{ and } x \in (x_{i-1}, x_i), \\ \frac{1}{h}(-x + x_{i+1}) & \text{if } x \in [a,b] \text{ and } x \in (x_i, x_{i+1}), \\ 0 & \text{else,} \end{cases}$$

for $i = 0, \dots, N$. Define the Finite Element (FE) space $\widetilde{V}_N := \operatorname{span}\{b_0, \dots, b_N\} \subset H^1(\mathrm{D})$. Which functions lie in the space $V_N := \widetilde{V}_N \cap H_0^1(\mathrm{D})$? The variational formulation over V_N reads: find (the unique) $q_N \in V_N$ such that

$$a(u; q_N, v_N) = \langle f, v_N \rangle_{V^*,V}, \quad \forall v_N \in V_N. \tag{2.2.3}$$

Derive an equivalent matrix formulation $\mathbf{A}\mathbf{q}_N = \mathbf{F}$ of (2.2.3): here the vector $\mathbf{q}_N = (q_{N,i})_{i=0}^N \in \mathbb{R}^{N+1}$ is such that $q_N = \sum_{i=0}^N q_{N,i} b_i$. Give exact formulas for the entries of the "stiffness matrix" $\mathbf{A} \in \mathbb{R}^{N+1 \times N+1}$ and the "load vector" $\mathbf{F} \in \mathbb{R}^{N+1}$.

(2.2c) Implement a function `stiff.m` in `Matlab` which takes the functions $u = (u_1, u_2, u_3)$ and the vector $\mathbf{x} = (x_i)_{i=0}^N$ (representing the mesh) as input and returns $\mathbf{A}$ as output. Moreover implement `load_vec.m` which takes the functions f and the vector $\mathbf{x}$ as input and returns the load vector $\mathbf{F}$. (If you prefer you can use another programming language to solve this exercise.)

(2.2d) Test your code with the data $\mathrm{D} = (0,1)$, $u_1(x) = 1 + x$, $u_2(x) = x$, $u_3(x) = 2$ and $q(x) = x\cos(x\pi/2)$: Solve the FE system for $N = 2^j$, $j = 4, \dots, 12$ and plot the error in the $H^1(\mathrm{D})$ and $L^2(\mathrm{D})$ norm. Which convergence rate do you observe?

(2.2e) Consider a parametric coefficient $u_1 = 2 + y\sin(2\pi x)$ for $y \in [-1,1]$ and $u_2 = u_3 = 0$ as well as the right-hand side $f(x) = 1$. Write a function which takes as input N and y and returns the FEM solution $q_{N;y}$ (in the form of a vector containing its coefficients).

(2.2f) Now we want to approximate the Bochner integral $\int_{-1}^1 q_{N;y}\,\mathrm{d}y \in V_N$ with a Gauss rule: Use the given MATLAB function `gauleg.m` to obtain the quadrature weights α_i and quadrature points $x_i \in [-1,1]$ for $i = 0, \dots, M$. Then approximate the integral via the quadrature formula

$$Q_M(q_N) := \sum_{i=0}^M \alpha_i q_{N;x_i}. \tag{2.2.4}$$

For $N := 2^8$ plot the error decay of $\left\| \int_{-1}^1 u_{N;y}\,\mathrm{d}y - Q_M(q_N) \right\|_{H^1(\mathrm{D})}$ for $M = 1, \dots, 15$. What asymptotic behaviour do you observe as $M \to \infty$?

HINT: As a reference value for $\int_{-1}^1 u_{N;y}\,\mathrm{d}y$ use $Q_{20}(q_N)$.

(2.2g) Consider the non-linear problem from the previous exercise sheet

$$\mathcal{R}(q) = -q'' + q^3 - f = 0. \tag{2.2.5}$$

The goal of this exercise is to approximate the FE solution q_N with a Newton iteration:

Algorithm 2.1 Newton Iteration.

Require: Absolute tolerance $\tau > 0$, max. number of iterations k_{max}

given q_0, $s_0 \leftarrow 1$, $k \leftarrow 0$

while $\|s_k\|_{L^2(\mathrm{D})} > \tau$ and $k \leq k_{max}$ **do**

solve $\mathrm{D}\mathcal{R}(q_k)s_{k+1} = \mathcal{R}(q_k)$ for s_{k+1}

$q_{k+1} \leftarrow q_k - s_{k+1}$

$k \leftarrow k + 1$

end while

return q_k

1. State the variational formulation of (2.2.3).
2. Implement a function `FE_func_eval.m` in `Matlab` that takes the coefficients of a FE function v_N, the mesh points $\{x_0, \ldots, x_N\}$, and a certain point $x \in \mathrm{D}$ as input and returns the value of this FE function at x as output.
3. State the elements of the given Newton iteration in variational form with respect to the FE space V_N.
4. Implement the Newton iteration to approximate q_N.

Test your implementation with the function $q(x) = cx\cos(\pi x/2)$ for different values of $c > 0$. Comment on the number of steps needed in the Newton iteration for a certain chosen accuracy.

Published on October 12.

To be submitted on October 26. Last modified on October 20, 2017