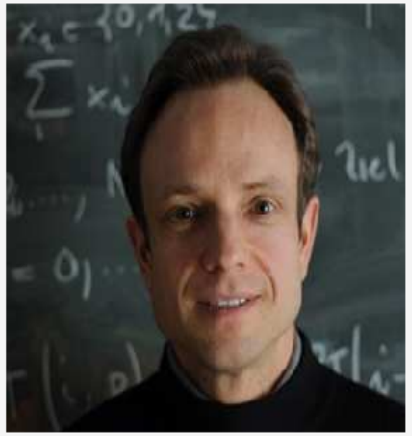
ETH Lecture 401-0674-00L Numerical Methods for Partial Differential Equations

Homework Problem Video Tutorial

Problem 3-1: Computing Averages over the Boundary

Prof. R. Hiptmair, SAM, ETH Zurich

Date: March 29, 2019



For $v : \Omega \to \mathbb{R}$:

$$F(v) = \int_{\partial\Omega} w(x)\, v(x)\, dS(x)$$

↑ fixed function

a) Show : $F : H^1(\Omega) \to \mathbb{R}$ continuous

$$|F(v)| \le C \|v\|_{H^1(\Omega)} \quad \forall v \in H^1(\Omega)$$

Hint: $F \sim$ r.h.s. functional for Neumann BVP

CSI in $L^2(\partial\Omega)$

$$|F(v)| \le \int_{\partial\Omega} |w(x)v(x)|\, dS(x) \le \|w\|_{L^2(\partial\Omega)} \|v\|_{L^2(\partial\Omega)}$$

$$\le C_1 C_2 \|v\|^{1/2}_{L^2(\Omega)} \|v\|^{1/2}_{H^1(\Omega)} \le C \|v\|_{H^1(\Omega)} .$$

Theorem 1.9.0.10. Multiplicative trace inequality

$$\exists C = C(\Omega) > 0: \ \|u\|^2_{L^2(\partial\Omega)} \le C \|u\|_{L^2(\Omega)} \cdot \|u\|_{H^1(\Omega)} \quad \forall u \in H^1(\Omega) .$$

Note: $\|w\|_{L^2(\Omega)}$ is incorporated into C, 2

Available :

```
template<typename FUNC_ALPHA, typename FUNC_GAMMA, typename
  FUNC_BETA>
 Eigen::SparseMatrix<double> compGalerkinMatrix(
     const lf::assemble::DofHandler &lfe_dofh,
     FUNC_ALPHA&& alpha, FUNC_GAMMA&& gamma, FUNC_BETA&& beta);
```

$$\underbrace{u \in H^1(\Omega): \int_\Omega \alpha(x)\,\mathbf{grad}\,u(x) \cdot \mathbf{grad}\,v(x) + \gamma(x)\,u(x)\,v(x)\,\mathrm{d}x + \int_{\partial\Omega} \beta(x)\,u(x)\,v(x)\,\mathrm{d}S(x)}_{a(u,v)} = \int_\Omega f(x)\,v(x)\,\mathrm{d}x \quad \forall v \in H^1(\Omega)\,, \tag{3.1.2}$$

b) Based on the function `compGalerkinMatrix()`, write a LEHRFEM++-based C++ function

```
double compH1seminorm(lf::assemble::DofHandler& dofh,
                      const Eigen::VectorXd& u);
```

that computes the norm $|v_h|_{H^1(\Omega)}$ for a function $v_h \in \mathcal{S}_1^0(\mathcal{M})$ passed through its vector u of expansion coefficients with respect to the customary "tent functions" nodal basis of $\mathcal{S}_1^0(\mathcal{M})$. The argument dofh passes an **lf::assemble::DofHandler** object providing the mesh $\mathcal{M}$ and the local-to-global index mapping for $\mathcal{S}_1^0(\mathcal{M})$.

$$a(v_h,v_h) = |v_h|^2_{H^1(\Omega)} \quad \text{for } \alpha \equiv 1,\ \gamma,\beta \equiv 0$$

$$[v_h \in \mathcal{S}_1^0(\mathcal{M})] \quad = \vec{\nu}^T \underset{\text{Galerkin matrix}}{A}\, \vec{\nu}$$

c) Again based on the function `compGalerkinMatrix`, write a LEHRFEM++-based C++ function

```
template<typename FUNCTION>
 double compBoundaryFunctional(
       const lf::assemble::DofHandler &dofh_lfe,
       const Eigen::VectorXd& u, const FUNCTION& w);
```

that evaluates the functional F from (3.1.1) for a function $v_h \in \mathcal{S}_1^0(\mathcal{M})$ passed through its expansion coefficient vector u with respect to the customary "tent function" nodal basis of $\mathcal{S}_1^0(\mathcal{M})$. The weight function w is made available through the functor w, while argument fe_test passes an object describing the piecewise linear Lagrangian finite element space.

```
template<typename FUNC_ALPHA, typename FUNC_GAMMA, typename
  FUNC_BETA>
 Eigen::SparseMatrix<double> compGalerkinMatrix(
     const lf::assemble::DofHandler &lfe_dofh,
     FUNC_ALPHA&& alpha, FUNC_GAMMA&& gamma, FUNC_BETA&& beta);
```

$$u \in H^1(\Omega): \int_\Omega \alpha(x)\,\mathbf{grad}\,\cancel{u(x)} \cdot \mathbf{grad}\,v(x) + \gamma(x)\,\cancel{u(x)}\,v(x)\,\mathrm{d}x + \int_{\partial\Omega} \beta(x)\,u(x)\,v(x)\,\mathrm{d}S(x) = \int_\Omega f(x)\,v(x)\,\mathrm{d}x \quad \forall v \in H^1(\Omega)\,, \tag{3.1.2}$$

$$F(v_h) = \int_{\partial\Omega} w\, v_h \,\mathrm{d}S = a(\mathbb{1}, v_h) \quad \text{for } \beta \equiv w,\ \alpha,\gamma \equiv 0$$

$$= \mathbb{1}^T \underset{\text{Gal. Mat}}{A}\, \vec{\nu}$$

$\mathbb{1} \in \mathcal{S}_1^0(\mathcal{M})$, w/ coefficient vector $[1,\dots,1]^T$

③ Now: $\mapsto \|\cdot\|_a = \|\cdot\|_{H^1(\Omega)}$ Var. Form. of a Neumann problem

$$u \in H^1(\Omega): \int_\Omega \mathbf{grad}\, u(x) \cdot \mathbf{grad}\, v(x) + u(x)v(x)\,dx = \int_\Omega f(x)\,v(x)\,dx \quad \forall v \in H^1(\Omega)\,. \tag{3.1.6}$$

$\hookrightarrow$ convex $\quad \hookrightarrow \in L^2(\Omega)$

d)

Based on the multiplicative trace estimate from [Lecture → Thm. 1.9.0.10] and assuming $f \in L^2(\Omega)$, describe qualitatively and quantitatively the expected asymptotic convergence of

$$\|u - u_h\|_{L^2(\partial\Omega)} := \left(\int_{\partial\Omega} |u(x) - u_h(x)|^2\,dS(x)\right)^{\frac{1}{2}} \tag{3.1.8}$$

in terms of the meshwidth $h := h_\mathcal{M} \to 0$. [h-refinement setting]

Theorem 1.9.0.10. Multiplicative trace inequality

$$\exists C = C(\Omega) > 0: \quad \|u\|^2_{L^2(\partial\Omega)} \le C\|u\|_{L^2(\Omega)} \cdot \|u\|_{H^1(\Omega)} \quad \forall u \in H^1(\Omega)\,.$$

$$\hookrightarrow \|u - u_h\|^2_{L^2(\Omega)} \le C\|u - u_h\|_{L^2} \cdot \|u - u_h\|_{\cancel{H^1}\, a}$$

$$\|u - u_h\|_a \le \inf_{v \in S_1^0(\mathcal{M})} \|u - v_h\|_{a/H^1} \le C h_\mathcal{M} |u|_{H^2(\Omega)}$$

Theorem 3.1.7. Regularity of solutions of Neumann problems

Assume that Ω is a convex bounded Lipschitz domain. If $f \in L^2(\Omega)$ and $g \in H^1(\partial\Omega)$ such that $\int_\Omega f\,dx + \int_{\partial\Omega} g\,dS = 0$, then every solution $u \in H^1(\Omega)$, $\int_\Omega u\,dx = 0$, of

$$-\Delta u + u = f \quad \text{in} \quad \Omega\,, \quad \mathbf{grad}\, u \cdot \boldsymbol{n} = g \quad \text{on} \quad \partial\Omega\,,$$

satisfies

$$u \in H^2(\Omega) \quad \text{and} \quad \|u\|_{H^2(\Omega)} \le C(\|f\|_{L^2(\Omega)} + \|g\|_{H^1(\partial\Omega)})\,,$$

with a constant $C > 0$ depending only on Ω.

$\|u - u_h\|_{L^2} \le C h_\mathcal{M}^2$ for 2-regular BVP [by Thm. 3.1.7]

$$\|u - u_h\|^2_{L^2(\partial\Omega)} \le C\,\|u - u_h\|_{L^2(\Omega)}\|u - u_h\|_{H^1(\Omega)}\,.$$

$$\|u - u_h\|_{H^1(\Omega)} \le C \inf_{v_h \in V_{0,N}} \|u - v_h\|_{H^1(\Omega)}\,.$$

$$\|u - u_h\|_{L^2(\Omega)} \le C h_\mathcal{M} \cdot \inf_{v_h \in V_{0,N}} \|u - v_h\|_{H^1(\Omega)}\,.$$

$$\inf_{v_h \in V_{0,N}} \|u - v_h\|_{H^1(\Omega)} \le C h_\mathcal{M} |u|_{H^2(\Omega)}\,.$$

$$\|u - u_h\|^2_{L^2(\partial\Omega)} \le C h_\mathcal{M}^3 \quad \Rightarrow \quad \|u - u_h\|_{L^2(\partial\Omega)} = O\left(h_\mathcal{M}^{\frac{3}{2}}\right) \quad \text{for} \quad h_\mathcal{M} \to 0\,.$$

e) Dual problem for output functional F

Theorem 3.6.1.7. Duality estimate for linear functional output

Define the dual solution $g_F \in V_0$ to F as solution of the dual variational problem

$$g_F \in V_0: \quad \mathsf{a}(v, g_F) = F(v) \quad \forall v \in V_0\,.$$

Then

$$|F(u) - F(u_h)| \le \|u - u_h\|_\mathsf{a} \inf_{v_h \in V_{0,h}} \|g_F - v_h\|_\mathsf{a}\,. \tag{3.6.1.8}$$

(4) Dual V.P.:

$$a(v,g_F) := \int_\Omega \mathbf{grad}\, v \cdot \mathbf{grad}\, g_F + v\, g_F \,dx = F(v) := \int_{\partial\Omega} w\, v\, dx \quad \forall v \in H^1(\Omega). \tag{3.1.9}$$

↳ Weak form of Neumann BVP
[→ Sect. 1.8.]

Dual BVP: $\begin{cases} -\Delta g_F + g_F = 0 & \text{in } \Omega \\ \mathbf{grad}\, g_F \cdot n = w & \text{on } \partial\Omega \end{cases}$

f)

Which asymptotic convergence of $|F(u) - F(u_h)|$ in terms of $h := h_\mathcal{M} \to 0$ is predicted by duality estimates for a piecewise continuously differentiable weight function $w \in C^1_{pw}(\partial\Omega)$? Justify your answer.

↳ $w \in H^1(\partial\Omega)$

Theorem 3.1.7. Regularity of solutions of Neumann problems

Assume that Ω is a convex bounded Lipschitz domain. If $f \in L^2(\Omega)$ and $g \in H^1(\partial\Omega)$ such that $\int_\Omega f\,dx + \int_{\partial\Omega} g\,dS = 0$, then every solution $u \in H^1(\Omega)$, $\int_\Omega u\,dx = 0$, of

$$-\Delta u + u = f \ \text{ in } \ \Omega, \quad \mathbf{grad}\, u \cdot n = \cancel{g} \ W \ \text{ on } \ \partial\Omega,$$

satisfies

$$u \in H^2(\Omega) \quad \text{and} \quad \|u\|_{H^2(\Omega)} \le C(\cancel{\|f\|_{L^2(\Omega)}} + \|g\|_{H^1(\partial\Omega)}),$$

with a constant $C > 0$ depending only on Ω.

Theorem 3.6.1.7. Duality estimate for linear functional output

Define the dual solution $g_F \in V_0$ to F as solution of the dual variational problem

$$g_F \in V_0: \quad a(v, g_F) = F(v) \quad \forall v \in V_0.$$

Then

$$|F(u) - F(u_h)| \le \|u - u_h\|_a \inf_{v_h \in V_{0,h}} \|g_F - v_h\|_a. \tag{3.6.1.8}$$

$$|F(u) - F(u_h)| \le \|u - u_h\|_a \cdot \inf_{v_h \in V_{0,N}} \|g_F - v_h\|_a$$

[Cea's Lemma] $= \inf_{v_h \in V_{0,h}} \|u - v_h\|_{H^1(\Omega)} \inf_{v_h \in V_{0,h}} \|g_F - v_h\|_{H^1(\Omega)}$

$\le C\, h_\mathcal{M} \inf_{v_h \in V_{0,h}} \|g_F - v_h\|_{H^1(\Omega)} \le C\, h_\mathcal{M} \inf_{v_h \in V_{0,N}} \|g_F - v_h\|_{H^1(\Omega)}.$

↑ Approximation estimates for $\mathcal{S}^0_1(\mathcal{M})$, exploit $u \in H^2(\Omega)$

Theorem 3.3.5.6. Best approximation error estimates for Lagrangian finite elements

Let $\Omega \subset \mathbb{R}^d$, $d = 1,2,3$, be a bounded polygonal/polyhedral domain equipped with a mesh $\mathcal{M}$ consisting of simplices or parallelepipeds. Then, for each $k \in \mathbb{N}$, there is a constant $C > 0$ depending only on k and the shape regularity measure $\rho_\mathcal{M}$ such that

$$\inf_{v_h \in \mathcal{S}^0_p(\mathcal{M})} \|u - v_h\|_{H^1(\Omega)} \le C \left(\frac{h_\mathcal{M}}{p}\right)^{\min\{p+1,k\}-1} \|u\|_{H^k(\Omega)} \quad \forall u \in H^k(\Omega). \tag{3.3.5.7}$$

for $p = 1$ for $k = 2$

⑤ $g_F \in H^2(\Omega)$

$$\inf_{v_h \in V_{0,h}} \|g_F - v_h\|_{H^1(\Omega)} \leq C h_{\mathcal{M}} \|g_F\|_{H^2(\Omega)} \,.$$

$$|F(u) - F(u_h)| \leq C\, h_{\mathcal{M}}^2 \, \|g_F\|_{H^2(\Omega)} \|u\|_{H^2(\Omega)} = O(h_{\mathcal{M}}^2) \quad \text{for} \quad h_{\mathcal{M}} \to 0 \,,$$

g) Finally, we want to test the predictions of our theoretical estimates in a numerical experiment. To that end the LEHRFEM++ function

```
std::shared_ptr<MeshHierarchy>
  generateTestMeshSequence(unsigned int L);
```

in the file `avg_val_boundary.cc` provides a (pointer to a) **lf::refinement::MeshHierarchy** object that contains a sequence $\mathcal{M}_0, \mathcal{M}_1, \ldots, \mathcal{M}_L$ of $L+1 \in \mathbb{N}$ meshes created by successive regular refinement of a triangular mesh of the unit square $\Omega =]0,1[^2$. In this domain we consider (3.1.6) with $f(x) = \|x\|$ and its Galerkin finite element discretization based on $\mathcal{S}_1^0(\mathcal{M})$.

We write $u_\ell \in \mathcal{S}_1^0(\mathcal{M}_\ell)$ for the finite element solution on $\mathcal{M}_\ell$. Based on `compGalerkinMatrix()` (also for the computation of the right-hand-side vector of the Galerkin linear system of equations) and `compBoundaryFunctional()` implement a C++ function (also in the file `avg_val_boundary.cc`)

```
std::vector<std::pair<unsigned int,double>>
  approxBoundaryFunctionalValues(unsigned int L);
```

that computes the values $F(u_\ell)$ for $\ell = 0, \ldots, L$ and returns them in a vector together with the dimension of the finite element space $\mathcal{S}_1^0(\mathcal{M}_\ell)$.

Compute r.h.s. vector from A with suitable α, β, γ.

$$\int_\Omega f\, b_h^i \, dx = a(1, b_h^i) \quad \text{for} \quad \begin{array}{l} \alpha \equiv 0 \\ \beta \equiv 0 \\ \gamma \equiv f \end{array}$$

C++11 code 3.1.12: Sub-problem (3-1.g): C++ function `approxBoundaryFunctionalValues()` ➺ GITLAB

```
std::vector<std::pair<unsigned int, double>> approxBoundaryFunctionalValues(
    unsigned int L) {
  std::vector<std::pair<unsigned int, double>> result{};
  /* SOLUTION_BEGIN */
  auto meshes = generateTestMeshSequence(L - 1);
  int num_meshes = meshes->NumLevels();
  for (int level = 0; level < num_meshes; ++level) {
    auto mesh_p = meshes->getMesh(level);
    // Set up global FE space; lowest order Lagrangian finite elements
    auto fe_space =
        std::make_shared<lf::uscalfe::FeSpaceLagrangeO1<double>>(mesh_p);
    // Obtain local->global index mapping for current finite element
    // space
    const lf::assemble::DofHandler& dofh{fe_space->LocGlobMap()};
    const lf::base::size_type N_dofs(dofh.NoDofs());
    lf::uscalfe::MeshFunctionConstant mf_identity{1.};
    // compute galerkin matrix
    auto A = AvgValBoundary::compGalerkinMatrix(dofh, mf_identity,
      mf_identity,
                                                mf_identity);
    // compute load vector for f(x) = x.norm()
    auto f = [](Eigen::Vector2d x) -> double { return x.norm(); };
    lf::uscalfe::MeshFunctionGlobal mf_f{f};
    Eigen::Matrix<double, Eigen::Dynamic, 1> phi(N_dofs);
    phi.setZero();
    lf::uscalfe::ScalarLoadElementVectorProvider<double,
      decltype(mf_identity)>
        elvec_builder(fe_space, mf_identity);
    AssembleVectorLocally(0, dofh, elvec_builder, phi);

    // solve system of equations
    Eigen::SparseLU<Eigen::SparseMatrix<double>> solver;
    solver.compute(A);
    Eigen::VectorXd u = solver.solve(phi);

    // set up weight function
    auto w = [](Eigen::Vector2d x) -> double { return 1.; };
    lf::uscalfe::MeshFunctionGlobal mf_w{w};
    double functional_value = compBoundaryFunctional(dofh, u, mf_w);

    result.push_back({N_dofs, functional_value});
  }
  /* SOLUTION_END */
  return result;
}
```

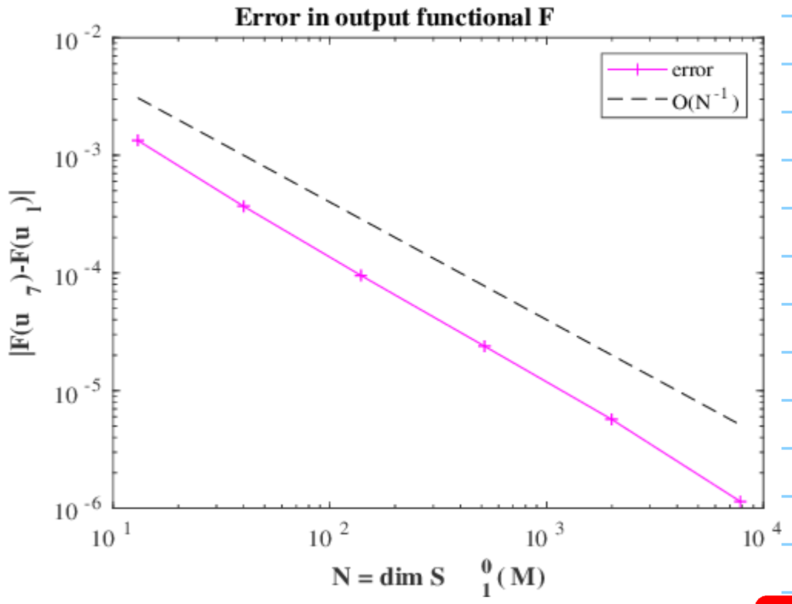
⑥ h, Converge study

In the setting of the Sub-problem (3-1.g) we choose $L = 7$ and use $F(u_7)$ as a substitute for the unknown value $F(u)$.

- Extend the `main()` function in `main.cc` so that it tabulates $F(u_\ell) - F(u_7)$ as a function of $N_\ell := \dim \mathcal{S}_1^0(\mathcal{M}_{ell})$.
- Describe qualitatively and quantitatively the "convergence" of $F(u_\ell) - F(u_7)$ as a function of $N_\ell := \dim \mathcal{S}_1^0(\mathcal{M}_{ell})$ for $\ell = 0, \ldots, 6$ as ℓ increases.

reference value
[replacement for $F(u)$]

N	$\lvert F(u_l)F(u_7)\rvert$
13	1.3E-03
40	3.6E-04
139	9.5E-05
517	2.3E-05
1993	5.6E-06
7825	1.1E-06



Asymptotic alg. cvg. with rate 1 w.r.t. $N \to \infty$ $\approx$ $O(h_\mathcal{M}^2)$ w.r.t. $h_\mathcal{M} \to 0$