

NUMERICS OF PARTIAL DIFFERENTIAL EQUATIONS

Series 10

1. Discretisation error for the L^2 -projection

For the finite element space V_n we define the L^2 -projection $Q_n : L^2(\Omega) \rightarrow V_n$ by

$$\int_{\Omega} Q_n u v_n \, dx = \int_{\Omega} u v_n \, dx \quad \forall v_n \in V_n, \quad (1)$$

where Ω is a open bounded subset of \mathbb{R}^2 .

Let $\mathcal{S}_1(\mathcal{M})$ be the finite element space of piecewise linear, continuous finite elements on a triangular mesh \mathcal{M} .

- a) Show existence and uniqueness of the L^2 -projection $Q_n u$ for any $u \in L^2(\Omega)$.
Hint: Regard (1) as variational formulation: Seek $u_n = Q_n u \in V_n$ such that

$$b(u_n, v_n) = \langle f, v_n \rangle_{L^2(\Omega)} \quad \forall v_n \in V_n,$$

with $f = u$.

- b) Show that $Q_n : L^2(\Omega) \rightarrow \mathcal{S}_1(\mathcal{M})$ is a projection, i. e. , it is a linear bounded operator with $\|Q_n\|_{L^2(\Omega) \rightarrow L^2(\Omega)} = 1$ and $Q_n^2 = Q_n$.
- c) Give a bound of the discretisation error $\|u - Q_n u\|_{L^2(\Omega)}$ by the best approximation error $\inf_{v_n \in \mathcal{S}_1(\mathcal{M})} \|u - v_n\|_{L^2(\Omega)}$ with an optimal constant.
Is $Q_n u$ the best-approximant of u in $\mathcal{S}_1(\mathcal{M})$? If not, explain this observation.
- d) Show that the following estimate

$$\|u - Q_n u\|_{L^2(\Omega)} \leq \gamma h_{\mathcal{M}}^2 |u|_{H^2(\Omega)}$$

holds true for all $u \in L^2(\Omega)$, with a constant γ independent of the mesh width $h_{\mathcal{M}}$.

Hint: Show the result for one triangle K using the interpolation operator l_n and its local version \hat{I} and the transformation techniques between the reference triangle \hat{K} and K .

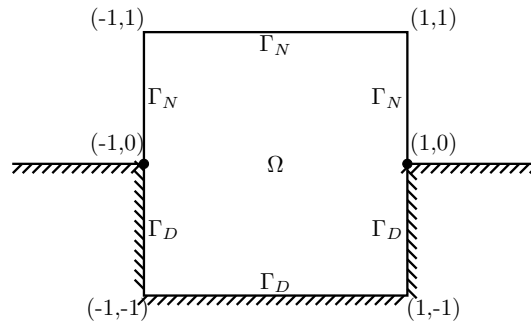
2. Singular functions for the mixed problem

Let us assume the problem

$$-\Delta u = 1,$$

where u is the displacement in the domain $\Omega = (-1, 1)^2$:

See next page!



The problem is completed by homogeneous Neumann boundary conditions on Γ_N and homogeneous Dirichlet boundary conditions on Γ_D .

- a) Compute the complete series of singular functions close to the points where the boundary conditions change.
- b) Are these all potentially occurring singular functions for the problem?

3. Solution of the problem with mixed boundary on uniform meshes

Solve the problem in task 2. with your linear (and voluntary quadratic) finite element code on regular meshes with $n = 3$ the order of the numerical quadrature. Perform numerical experiments with varying mesh widths $h_j = h_0 2^{-j/2}$ and investigate how the discrete solution converges in energy norm to the exact solution. Show the convergence in a double logarithmic plot with the number of degrees of freedom on the abscissa and estimate the order of convergence.

Hint: See task 3. in series 8 for the computation of the energy norm.

Hint: We obtain that the energy norm of the exact solution $|u|_{H^1(\Omega)} = 1.548888$ (rounded to six digits).

Hint: For handling mixed boundary conditions you may follow the approach proposed in task 4.b) of series 7, i. e. , first assembling the system matrix and load vector also for the degrees of freedom on the Dirichlet boundary, and then reduce the system. You may use the coordinates of the vertices to distinguish whether a vertex or edge is on on the Dirichlet boundary or not.

Alternatively, you can consider that there are no degrees of freedom on nodes and edges on the Dirichlet boundary by not giving a number. For the quadratic elements this means to ignore those edges in the “edge numbering”, which is then an “edge dof numbering”. For the linear elements you may create a mapping of the node number of the “node dof number”, for which an array can be used. Then, build the full element matrices and check if for the local degrees of freedom there is a global one in the assembling process. For T-matrices this means just not to have an entry 1 where there is no global degree of freedom.

4. Solution of the problem with mixed boundary on graded meshes

Problem in task 3. shall be solved on meshes which are adaptively refined towards the singular points. For this construct meshes in Gmsh as follows. The geometry of the domain in the geo-file should be defined with explicitly given points $(-1, 0)^T$ and $(1, 0)^T$ and given line connecting these two points. The first guarantees that the singular points are resolved by the meshes and for the adaptive scheme described below. The second helps the mesh generator.

There are two variants:

See next page!

- A. The easiest way to obtain an adaptive mesh with Gmsh is to define the element size at a point of the geometry. For that in Gmsh select either Mesh -> Define -> Element size at point or give the desired value already in the geo-file as follows.

```
hmax = 0.5;
hmin = 0.125;

Point(1) = {-1, -1, 0, hmax};
Point(2) = {-1, 0, 0, hmin};
...
```

Gmsh uses a interpolation between the given points to decide for the element sizes.

If you set at the singular points the element size to be $\frac{1}{2}h_{\mathcal{M}_n}^2$ and $h_{\mathcal{M}_n}$ on the other points (corners of the square) you shall obtain a graded mesh with grading factor $\beta = 2$.

- B. More freedom to generate adaptive meshes is to give a desired element size as a function in space, which is given as sampled values on a so-called background mesh. Take the background meshes given in `bgmeshes.zip` we have created for $\beta = 4$ and mesh widths which decay with a factor $\sqrt{2}$.

The background mesh is used when calling Gmsh on the command line as follows.

```
gmsh square.geo -2 -bgm bgmesh0.pos -o square0.msh
```

In the Gmsh GUI open first the geo-file and merge the background mesh via File -> Merge. Then the background mesh is shown, it can be suppressed when clicking on Post-processing -> [0] background mesh. To use it as a background mesh click on the arrow besides and choose Apply as Background Mesh.

- a) Verify that the number of mesh nodes and elements are approximately doubled when the mesh width is divided by $\sqrt{2}$, for variant A. or B.
- b) (optional) Visualize the varying element sizes by plotting a piecewise linear continuous function, which takes on each node of the mesh the maximum of the element sizes of the elements sharing this node, for variant A. or B. Add the respective function to your module `meshes.py`. Compare the two variants.
- c) Perform numerical experiments with varying mesh widths $h_{\mathcal{M}_n} = h_{\mathcal{M}_0}2^{-n/2}$ and investigate how the discrete solution converges in energy norm to the exact solution, for variant A. or B.. Show the convergence in a double logarithmic plot with the number of degrees of freedom on the abscissa and estimate the order of convergence. Compare the results with those obtained in task 3..

What is the best refinement strategy for linear finite elements ?

(optional) What is the best refinement strategy for quadratic finite elements ?

To be handed in by: February 4th, 2016 (9 a.m.)

This exercise series will be discussed in the tutorial class on February 9th, 2016, 10.15 a.m. in A 052.

Coordinator: Anastasia Thöns, MA 365, 030/314-79369, anastasia.thoens@math.tu-berlin.de

Website: <http://www.tu-berlin.de/?NumPDE>