Master de sciences et technologie 1 | MU4MA029
Mention : Mathématiques et applications | 2023–2024

Project

This project is an individual work. All implementation work should be done in Python 3.0 and rely on the libraries written for scientific computing, namely NumPy, SciPy and Matplotlib. The files submitted should include your Python scripts and any additional files (figures, ...) that you judge necessary. All figures should be self-explanatory (labels on every axis, legend, ...). **Some level of testing is expected in the implementation and you will be specifically asked questions on the tests.** This project is largely an adaptation of the code written during the programming sessions so that you should re-use some of the code already written.

1. **Geometry.** Let $L_x > 0$ and $L_y > 0$. We consider the following rectangular domain

$$\Omega := \{(x, y) \in (0, L_x) \times (0, L_y)\}. \tag{1}$$

   (a) Write a routine `GenerateRectangleMesh` adapted from the one from TP2 that generates a **uniform** structured triangular mesh for Ω. The inputs should be: L_x (horizontal length), L_y (vertical length), N_x (number of horizontal subdivisions) and N_y (number of vertical subdivisions). The outputs should be: `vtx` (coordinate array) and `elt` (connectivity array) in the format of TP2.

   (b) Write a routine `GeometricRefinement` that performs a refinement of the mesh **locally** near the origin. The refinement should be done according to the sketch in Figure 1.

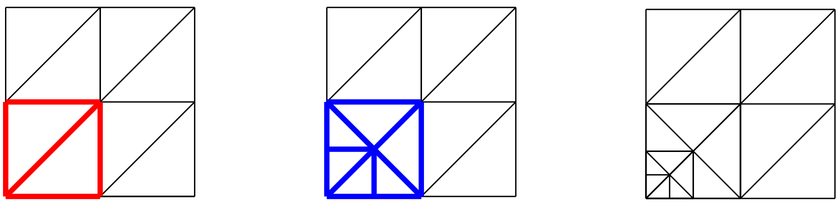

Figure 1: Mesh before refinement (left), after one level of refinement (middle) and after two levels of refinement (right).

   The 2 triangles (in red on the left) containing the origin are replaced by 6 new triangles (in blue in the middle), introducing 3 new nodes. All other triangles remain.

   Several levels of refinement can be performed iteratively, by refining on the next step the 2 new small triangles containing the origin (see right panel of Figure 1).

   The inputs should be the coordinate array `vtx` and connectivity array `elt` from `GenerateRectangleMesh` and the outputs should be the updated arrays. Besides, the total number of refinement levels will be controlled by a parameter $\mathtt{r} \in \mathbb{N}$.

   (c) Write a routine `PlotMesh` that can represent a triangular mesh of the domain Ω and plot the new mesh.

2. **PDE problem.** Let $f \in L^2(\Omega)$: and $\mu : \overline{\Omega} \to (0, +\infty)$ such that

$$\inf_\Omega \mu > 0, \qquad \sup_\Omega \mu < +\infty. \tag{2}$$

We consider the following second-order PDE model:

$$\begin{cases} \text{Find } u \in H^1(\Omega) \text{ such that :} & \\ -\operatorname{div}(\mu \operatorname{grad} u) + u = f, & \text{in } \Omega, \\ u = 0, & \text{on } \partial\Omega. \end{cases} \tag{3}$$

We want to compute an approximate solution u_h of the above problem using a conforming Galerkin method. The finite dimensional approximation space V_h is constructed using $\mathbb{P}_1$ Lagrange finite elements on triangular meshes.

(a) Write the variational formulation associated to (3).

(b) Write two routines to assemble the elementary matrices associated to each term appearing in the sesquilinear form for the numerical method considered, on the model of what was done in TP5. The variable coefficient μ will be approximated by $\Pi_h \mu$, where Π_h is the $\mathbb{P}_1$ Lagrange finite element interpolation operator.

(c) Write the routine to assemble the full matrix of the associated linear system.

3. **Right-hand-side.** For some $\alpha \in \mathbb{R}$ such that $1/2 < \alpha$, we consider the following solution

$$u_{\mathrm{ex}}(x, y) := (xy)^{\alpha}(x - L_x)(y - L_y), \qquad \forall (x, y) \in \Omega, \tag{4}$$

and the particular coefficient

$$\mu(x, y) = 2 + \sin(2\pi x/L_x)\sin(4\pi y/L_y), \qquad \forall (x, y) \in \Omega. \tag{5}$$

We want to have a routine to compute the right-hand-side of the linear system for the source term f such that u_{ex} given in (4) is the exact solution of the problem (3) for this μ given in (5).

(a) Compute an analytical expression of each of the following functions

$$\frac{\partial u_{\mathrm{ex}}}{\partial x}, \quad \frac{\partial u_{\mathrm{ex}}}{\partial y}, \quad \frac{\partial^2 u_{\mathrm{ex}}}{\partial x^2}, \quad \frac{\partial^2 u_{\mathrm{ex}}}{\partial y^2}, \quad \frac{\partial \mu}{\partial x}, \quad \frac{\partial \mu}{\partial y}, \tag{6}$$

and implement routines able to evaluate these expressions and the one of u_{ex} and μ at a point $(x, y) \in \Omega$. Pay attention to the case $\alpha = 1$.

(b) Give an expression of f in terms of the previous functions and implement a routine able to evaluate f at a point $(x, y) \in \Omega$.

(c) Write a routine to assemble (a numerical approximation of) the right-hand-side vector of the linear system for the source term f. Beware of division by 0 on $\partial\Omega$.

4. **Resolution.**

(a) Solve numerically (3) with a uniform mesh and the source term previously computed, for $\alpha = 2/3$ and $\alpha = 1$. Take sensible values for L_x and L_y.

(b) Write a routine `PlotApproximation` that can represent a piecewise affine field $v_h \in V_h$ in the domain Ω for some triangular mesh. Using this routine, represent the numerical solution u_h from the previous question and the associated error $u_h - \Pi_h u_{\mathrm{ex}}$.

(c) Plot the convergence of both errors

$$\frac{\|u_h - \Pi_h u_{\mathrm{ex}}\|_{L^2(\Omega)}}{\|\Pi_h u_{\mathrm{ex}}\|_{L^2(\Omega)}}, \qquad \frac{\|u_h - \Pi_h u_{\mathrm{ex}}\|_{H^1(\Omega)}}{\|\Pi_h u_{\mathrm{ex}}\|_{H^1(\Omega)}}, \tag{7}$$

with respect to the mesh parameter h for various uniform mesh refinements (increasing N_x and N_y). What is the order of convergence for $\alpha = 1$? How does the method converges for $\alpha = 2/3$?