Technische Universität Berlin Winter term 2025/26
Institute of Mathematics
Prof. Dr. T. Breiten, Prof. Dr. S. May, Dr. Juliette Dubois

Numerical Mathematics II for Engineers
Tutorial 4

Topics : M-matrices, 2D Poisson equation with inhomogeneous boundary conditions

Discussion in the tutorials of the week 17–21 November 2025

Disclaimers:

- To test your code, we provide tests. This should help you to locate precisely your errors, and code more efficiently. To run the test, you need to install pytest.
- Exercises should be solved in **fixed groups of 3 students**.
- Hand in the solution in **one folder** labeled **hw[hw_number]_group[group_number]** and containing:
  - **One pdf** for the theoretical questions and comments on the numerical results,
  - **One python file per programming exercise**.
  - Write the group number and all names of your members **in each file**.

Exercise 4.1: M-Matrices

Recall the definition of an M-matrix, seen in the lecture: We say that $M \in \mathbb{R}^{N\times N}$ is an M-matrix when it holds

(a) $a_{ii} > 0$ for all $i = 1, \dots, N$ and $a_{ij} < 0$ for all $i \neq j$.

(b) $\det(A) \neq 0$,

(c) All entries in A^{-1} are non-negative:$A^{-1} \geq 0$.

We show that the matrix obtained from the discretization of the 2D Laplace equation with the 5-points stencil is an M-matrix. We focus on the case with 3 inner points in each direction. The reduced matrix is a 9×9 matrix:

$$A = \frac{1}{h^2}\left(\begin{array}{ccc|ccc|ccc}
4 & -1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\
-1 & 4 & -1 & 0 & -1 & 0 & 0 & 0 & 0 \\
0 & -1 & 4 & 0 & 0 & -1 & 0 & 0 & 0 \\
\hline
-1 & 0 & 0 & 4 & -1 & 0 & -1 & 0 & 0 \\
0 & -1 & 0 & -1 & 4 & -1 & 0 & -1 & 0 \\
0 & 0 & -1 & 0 & -1 & 4 & 0 & 0 & -1 \\
\hline
0 & 0 & 0 & -1 & 0 & 0 & 4 & -1 & 0 \\
0 & 0 & 0 & 0 & -1 & 0 & -1 & 4 & -1 \\
0 & 0 & 0 & 0 & 0 & -1 & 0 & -1 & 4
\end{array}\right),$$

and define $\tilde{A} = h^2 A$ to get rid of the factor $1/h^2$ for the computations.

(a) Among the three conditions (a)-(c), which one can be directly seen on the matrix A?

(b) Using the `numpy.linalg` library, compute the determinant and inverse of A. Is A and M-matrix?

Hint: use the `numpy.linalg.det` and `numpy.linalg.inv` functions.

(c) One can show that $\det(A) \neq 0$ (that is, A is invertible) without actually computing it. From linear algebra, you may know that positive definite matrices are invertible. In Lecture 2, it was shown that the matrix of the discretized 1D Poisson problem is positive definite. Sketch how you would adapt the proof from Lecture 2 to show that A is positive definite.

Exercise 4.2: 2D Poisson equation, non-homogenous Dirichlet boundary conditions

We solve again the 2D Poisson problem, this time with non-homogeneous Dirichlet boundary conditions

$$-\Delta u(x,y) = f(x,y) \quad \text{in } \Omega = (0,1)^2 \subset \mathbb{R}^2 \tag{1}$$
$$u(x,y) = g(x,y) \quad \text{on } \Gamma = \partial\Omega. \tag{2}$$

Let $\Gamma_L, \Gamma_R, \Gamma_B$ and Γ_T the left, right, bottom and top boundary, respectively:

$$\Gamma_L = \{(0,y), y \in (0,1)\}, \quad \Gamma_R = \{(1,y), y \in (0,1)\},$$
$$\Gamma_B = \{(x,0), x \in (0,1)\}, \quad \Gamma_T = \{(x,1), x \in (0,1)\}.$$

The function g is defined using four functions, one for each boundary

$$g(x,y) = \begin{cases} g_L(y), & (x,y) \in \Gamma_L \\ g_R(y), & (x,y) \in \Gamma_R \\ g_B(x), & (x,y) \in \Gamma_B \\ g_T(x), & (x,y) \in \Gamma_T \end{cases}.$$

The template `Poisson2D_bc_template.py` and the test `Poisson2D_bc_test.py` are provided.

We recall the discrete system with the two-dimensional indexing:

$$-\frac{1}{h^2}\left(u_{k+1,l} + u_{k,l+1} - 4u_{k,l} + u_{k-1,l} + u_{k,l-1}\right) = f_{k,l}, \qquad k = 1,\dots,N, l = 1,\dots,N. \tag{3}$$

Remark: Note that the intervals are discretized in $N+1$ sub-intervals in each directions, instead of N points in each direction in the lecture.

(a) Write the discrete system (3) for the points next to the boundary: $(1,l), (N,l)$, $(k,1), (k,N)$ with $l = 1,\dots,N$ and $k = 1,\dots,N$. How should you modify the right-hand side of the discrete system to include the non-homogeneous Dirichet Boundary conditions?

Hint: For the values of u on the boundary points $k = 0, k = N+1, l = 0, l = N+1$, use the boundary condition (2). The functions g_L, g_R, g_B, g_T should appear.

(b) The non-homogeneous boundary conditions appear at two different places in the implementation: in the right-hand side for computing the solution at interior points, and at the boundary points. Hence, you will need an indexing function that finds the indices either at the boundary, or next to the boundary. We provide a function `get_boundary_separate` that returns four arrays of booleans `boundaryL, boundaryR, boundaryB, boundaryT`:

- If `flag=="boundary"`, the array `boundaryL` has the value `True` at indices on the boundary Γ_L and False everywhere else.
- If `flag=="interior"`, the array `boundaryL` has the value `True` at indices next to the boundary Γ_L and False everywhere else.

The other arrays are defined similarly for $\Gamma_R, \Gamma_B, \Gamma_T$.

Write a function `get_boundary_values` that returns the evaluation of g_L, g_R, g_B, g_T at the boundary points. The function should return four arrays `aL, aR, aB, aT`, each of size $(N+2)$.

Hint: Use the function `get_boundary_separate` to get the boolean array of indices for each boundary. Should you use this function with the full mesh `X_full, Y_full`, or with the mesh of interior points `X, Y`?

(c) Complete the function `get_matrix_rhs` that returns the discrete Laplace operator and the right-hand side.

Hint: For the right-hand side, you can use the arrays `aL, aR, aB, aT` and the function `get_boundary_values`. Should you use this function with the full mesh `X_full, Y_full`, or with the mesh of interior points `X, Y`?

(d) The function `my_driver` has almost the same structure as in Homework 03. Complete this function to fill the solution vector with the boundary conditions.

(e) For the right-hand side and boundary conditions

$$\begin{aligned} f_3(x,y) &= \frac{\pi^2}{2}\sin\left(\frac{\pi}{2}(x+1)\right)\sin\left(\frac{\pi}{2}(y+1)\right), \\ g_L(y) &= \sin\left(\frac{\pi}{2}(y+1)\right), \quad g_R(y) = 0 \\ g_B(x) &= \sin\left(\frac{\pi}{2}(x+1)\right), \quad g_T(x) = 0. \end{aligned}$$

the exact solution is

$$u_3(x,y) = \sin\left(\frac{\pi}{2}(x+1)\right)\sin\left(\frac{\pi}{2}(y+1)\right).$$

Test your scheme with the right-hand side f_2 from Homework 03, and with right-hand side f_3 and boundary conditions g_L, g_R, g_B, g_T. Plot the solution for each case for $N = 40$.