



Laplacien anisotropique

Soit un domaine 2D rectangulaire $\Omega = [0, L_1] \times [0, L_2]$, avec $(L_1, L_2) \in (\mathbb{R}_+^*)^2$.
On s'intéresse à l'équation du Laplacien 2D sur ce domaine :

$$\begin{cases} -\nu_1 \frac{\partial^2 u}{\partial x_1^2}(x) - \nu_2 \frac{\partial^2 u}{\partial x_2^2}(x) = c(x), & \forall x \in]0, L_1[\times]0, L_2[\\ u(x_1, 0) = u(x_1, L_2) = 0, & \forall x_1 \in]0, L_1[\\ u(0, x_2) = u(L_1, x_2) = 0, & \forall x_2 \in]0, L_2[\end{cases} \quad (1)$$

avec $\nu = (\nu_1, \nu_2) \in (\mathbb{R}_+^*)^2$ les coefficients de diffusivité dans les directions x_1 et x_2 , et c continue sur Ω .

L'objectif de ce TP est d'implanter la résolution numérique de ce problème par une méthode de différences finies.

On souhaite approcher u par la méthode des différences finies. Pour cela, on considère une discrétisation régulière de Ω , de pas constants h_1 et h_2 dans chacune des deux directions. Soit $(x_{i,j})_{i=0:N_1+1, j=0:N_2+1}$ les points de discrétisation du maillage. On approxime les dérivées partielles secondes par un schéma centré d'ordre 2, ce qui conduit au schéma numérique :

$$\begin{cases} \forall (i, j) \in \llbracket 1, N_1 \rrbracket \times \llbracket 1, N_2 \rrbracket \\ -\nu_1 \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h_1^2} - \nu_2 \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{h_2^2} = c(x_{i,j}) \\ \forall i \in \{0, N_1 + 1\}, \forall j \in \llbracket 1, N_2 \rrbracket, & u_{i,j} = 0 \\ \forall j \in \{0, N_2 + 1\}, \forall i \in \llbracket 1, N_1 \rrbracket, & u_{i,j} = 0 \end{cases} \quad (2)$$

Travail à réaliser

- 1- On pose $u_h = [u_{1,1}, \dots, u_{1,N_2}, u_{2,1}, \dots, u_{2,N_2}, \dots, u_{N_1,N_2}]^T \in \mathbb{R}^{N_1 N_2}$.
Ecrire le schéma sous la forme matricielle $A_h u_h = b_h$ en précisant A_h et b_h .
- 2- Implanter la construction de la matrice A_h dans le fichier `laplacian.m`.
Cette matrice étant creuse - elle présente un très grand nombre d'entrées nulles - il n'est pas envisagé de la construire/stocker sous la

forme d'un tableau bidimensionnel. Vous utiliserez plutôt des fonctions `matlab` dédiées à l'algèbre linéaire creuse telles que `sparse`, `spdiags`, `speye`, etc..

- 3- Implanter le terme de forçage défini pour la fonction $c(x) = -1, \forall x \in \Omega$ dans le fichier `forcing.m`.
- 4- Résoudre numériquement l'EDP (1) pour différentes valeurs de (L_1, L_2) , (N_1, N_2) et ν . La figure représente la solution obtenue avec le schéma numérique (2) pour le cas $(L_1, L_2) = (1, 2)$ et $\nu = (1, 4)$ pour différentes valeurs de N_1 et N_2 .

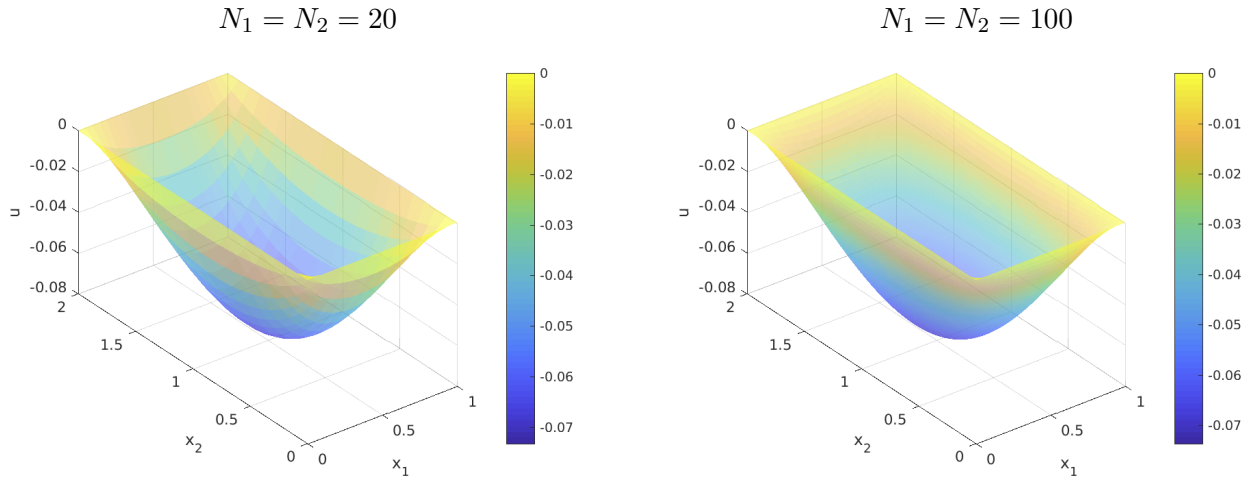


FIGURE 1 – Approximations de la solution de l'EDP pour $(L_1, L_2) = (1, 2)$ et $\nu = (1, 4)$

Bonus

B1- Soit $(A, B) \in \mathcal{M}_{m,n}(\mathbb{R}) \times \mathcal{M}_{p,q}(\mathbb{R})$. On définit le produit de Kronecker entre les matrices A et B par

$$A \otimes B = \begin{bmatrix} a_{1,1}B & a_{1,2}B & \cdots & a_{1,n}B \\ a_{2,1}B & a_{2,2}B & \cdots & a_{2,n}B \\ \vdots & \vdots & \ddots & \vdots \\ a_{m,1}B & a_{m,2}B & \cdots & a_{m,n}B \end{bmatrix} \in \mathcal{M}_{mp,nq}(\mathbb{R}).$$

Montrer que la matrice A_h s'écrit $A_h = \nu_1(B_1 \otimes C_1) + \nu_2(B_2 \otimes C_2)$, avec B_1, C_1, B_2, C_2 à définir. Implanter une nouvelle version de la construction de la matrice A_h dans le fichier `laplacian.m` basée sur la fonction `kron` de `matlab`, qui réalise le produit de Kronecker de deux matrices.