

Mini-projet d'analyse numérique du cours MAP 431

Résolution de l'équation de Poisson par transformée de Fourier rapide

Sujet proposé par Gabriel Peyré

Le but de ce projet est de rendre la méthode des différences finies très efficace grâce à la transformée de Fourier rapide (FFT). L'idée est de résoudre une EDP dans le domaine des sinusoides de Fourier. Grâce à la FFT, on dispose d'un moyen très rapide de passer du domaine spatial au domaine de Fourier, et vice et versa.

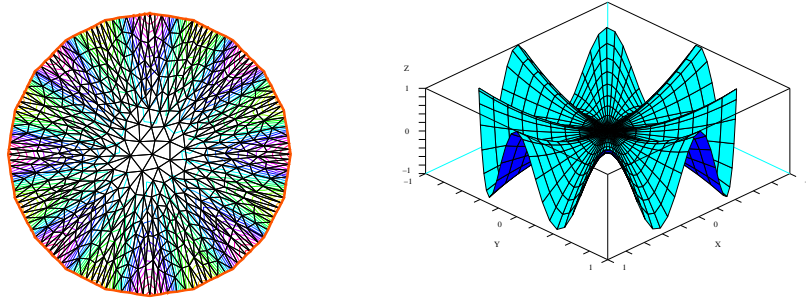


Figure 1: Solution par éléments finis (gauche) et par différences finies (droite)

On souhaite calculer, pour $x \in \overline{D} = \{y \in \mathbb{R}^2 \mid \|y\| \leq 1\}$, le déplacement vertical $u(x)$, d'une membrane élastique circulaire soumise à une force surfacique $f(x)$. Ce déplacement vérifie l'équation de Poisson

$$\begin{cases} \forall x \in D, & \Delta u(x) = f(x), \\ \forall x \in \partial D, & u(x) = g(x). \end{cases} \quad (1)$$

Partie 2 : différences finies et FFT (SCILAB).

Question 3 : Montrer qu'en coordonnées polaires, le problème (1) s'écrit

$$\begin{cases} \forall (r, \theta) \in]0, 1[\times [0, 2\pi[, & \frac{\partial^2 u}{\partial r^2} + \frac{1}{r} \frac{\partial u}{\partial r} + \frac{1}{r^2} \frac{\partial^2 u}{\partial \theta^2} = f(r, \theta), \\ \forall \theta \in [0, 2\pi[, & u(1, \theta) = g(\theta). \end{cases} \quad (2)$$

Question 4 : Sous l'hypothèse que la solution u est suffisamment régulière, on peut la développer en séries de Fourier de θ . On cherche donc, pour $N_\theta > 0$ une solution approchée sous la forme

$$u(r, \theta) = \sum_{n=-N_\theta/2}^{N_\theta/2-1} \widehat{u}_n(r) e^{in\theta} \quad \text{avec} \quad \widehat{u}_n(r) = \frac{1}{2\pi} \int_0^{2\pi} u(r, \theta) e^{-in\theta} d\theta.$$

On développe de même $\theta \mapsto f(r, \theta)$ et g en polynôme trigonométrique. Montrer alors que u vérifie (2) si et seulement si

$$\forall n \in \{-N_\theta/2, \dots, N_\theta/2 - 1\}, \forall r \in]0, 1[, \quad \frac{d^2 \widehat{u}_n}{dr^2} + \frac{1}{r} \frac{d\widehat{u}_n}{dr} - \frac{n^2}{r} \widehat{u}_n = \widehat{f}_n.$$

Question 5 : Soit $N_\theta > 0$ et $N_r > 0$ les nombres de points de discrétisation selon θ et selon r . On choisit les points de discrétisation

$$\forall j = 1, \dots, N_\theta, \quad \theta_j = \frac{2(j-1)\pi}{N_\theta}, \quad \text{et} \quad \forall i = 1, \dots, N_r, \quad r_i = (i-1/2)\Delta_r,$$

avec $\Delta_r = 2/(2N_r + 1)$. On définit les matrices et vecteurs SCILAB suivant, pour $n = 1, \dots, N_\theta$

$$\begin{aligned} \mathbf{U}(i, j) &= u(r_i, \theta_j), & \mathbf{f}(i, j) &= f(r_i, \theta_j), & \mathbf{g}(i, j) &= g(\theta_j), \\ \mathbf{UU}(i, \mathbf{n}) &= \widehat{u}_{\tilde{n}}(r_i), & \mathbf{ff}(i, \mathbf{n}) &= \widehat{u}_{\tilde{n}}(r_i), & \mathbf{gg}(\mathbf{n}) &= \widehat{g}_{\tilde{n}}, \end{aligned}$$

où $\tilde{n} = n - 1$ si $1 \leq n \leq N_\theta/2$ et $\tilde{n} = n - 1 - N_\theta$ si $N_\theta/2 < n \leq N_\theta$. Expliquer pourquoi on peut calculer, de façon approchée, \mathbf{gg} , \mathbf{ff} et \mathbf{U} de la façon suivante

$$\begin{aligned} \mathbf{gg}(\mathbf{n}) &= \frac{1}{N_\theta} \sum_{j=1}^{N_\theta} \mathbf{g}(j) e^{-\frac{2i\pi}{N_\theta}(j-1)(n-1)}, & \mathbf{ff}(\mathbf{n}) &= \frac{1}{N_\theta} \sum_{j=1}^{N_\theta} \mathbf{f}(j) e^{-\frac{2i\pi}{N_\theta}(j-1)(n-1)}, \\ \mathbf{UU}(i, \mathbf{n}) &= \sum_{j=1}^{N_\theta} \mathbf{U}(i, j) e^{\frac{2i\pi}{N_\theta}(j-1)(n-1)}. \end{aligned}$$

On admet que ces calculs se réalisent de façon très rapide (en temps $O(N_\theta \log(N_\theta))$) grâce aux instructions SCILAB suivantes

$$\mathbf{gg} = \mathbf{fft}(\mathbf{g}, -1); \quad \mathbf{ff} = \mathbf{fft}(\mathbf{f}, -1); \quad \mathbf{U}(i, :) = \mathbf{fft}(\mathbf{U}(i, :), 1);$$

Question 6 : Montrer que l'on peut calculer de façon approchée les vecteurs $v = \mathbf{UU}(:, \mathbf{n})$ en résolvant, pour $\mathbf{n} = 1, \dots, N_\theta$ l'équation aux différences finies

$$\frac{v_{i+1} - 2v_i + v_{i-1}}{(\Delta_r)^2} + \frac{1}{r_i} \frac{v_{i+1} - v_{i-1}}{2\Delta_r} + \frac{\tilde{n}}{r_i^2} v_i = \mathbf{ff}(i, \mathbf{n}). \quad (1)$$

Ecrire le système matriciel correspondant. Programmer en SCILAB les différentes étapes

- Calcul de \mathbf{f} et \mathbf{g} pour la fonction étudiée à la question 2.
- Calcul de \mathbf{ff} et \mathbf{gg} avec la commande `fft`.
- Calcul des $\mathbf{UU}(:, \mathbf{n})$ en résolvant (1).
- Calcul de \mathbf{U} avec la commande `fft`.
- Affichage du résultat et comparaison avec la solution exacte.

Question bonus : Dans la suite, on suppose que $N = 2^p$.

Soit $f = \{f[0], \dots, f[N-1]\} \in \mathbb{C}^N$; on note \hat{f} la *transformée de Fourier discrète* (TFD) de f , qui est définie par

$$\hat{f}[k] = \sum_{n=0}^{N-1} f[n] e^{-\frac{2i\pi}{N} kn}.$$

Quelle est la matrice Ω_N de la transformation $f \mapsto \hat{f}$? Montrer que $\frac{1}{\sqrt{N}}\Omega_N$ est une matrice unitaire, i.e. $\Omega_N \Omega_N^* = N \text{Id}_N$. En déduire comment on peut calculer f à partir de \hat{f} . Combien d'opération faut-il pour calculer le produit d'une matrice générique M de taille $N \times N$ par un vecteur ?

On va voir qu'on peut en fait calculer le produit $\hat{f} = \Omega_N f$ de façon beaucoup plus rapide. On note $f_0 = \{f[2k]\}_{k=0}^{N/2-1}$ et $f_1 = \{f[2k+1]\}_{k=0}^{N/2-1}$; de même, on définit $\hat{f}_g = \{\hat{f}[k]\}_{k=0}^{N/2-1}$ et $\hat{f}_d = \{\hat{f}[k]\}_{k=N/2}^{N-1}$. Montrer que l'on a

$$\hat{f}_g = \hat{f}_0 + \mathcal{S}_{N/2}^{1/2}(\hat{f}_1) \quad \text{et} \quad \hat{f}_d = \hat{f}_0 - \mathcal{S}_{N/2}^{1/2}(\hat{f}_1),$$

où l'on a noté \mathcal{S}_n^x l'opérateur qui envoie $y \in \mathbb{C}^N$ sur $\{y[n]e^{-\frac{2i\pi}{N}xn}\}_n$. On note $C(N)$ le nombre d'opérations nécessaires pour calculer la TFD d'un vecteur de taille N . En déduire que l'on a la relation de récurrence $C(N) \leq C(N/2) + KN$ où K est une certaine constante. En déduire un algorithme (FFT) qui calcule \hat{f} en $O(N \log(N))$ opérations.

Quel gain l'algorithme FFT apporte-t-il à l'ensemble de la résolution numérique ? On devra prendre en compte le temps nécessaire à l'inversion du système linéaire (1).

Partie 1 : formulation variationnelle (FREEFEM++).

Question 1 : Donner la formulation variationnelle du problème (1). Rappeler pourquoi, sous la condition que $f \in L^2(D)$ et que g est la trace sur ∂D d'une fonction de $H^1(D)$, alors il existe une unique solution $u \in H^1(D)$ qui vérifie (1) presque partout.

Question 2 : Résoudre numériquement ce problème avec FREEFEM++, en étudiant un problème dont on connaît la solution, par exemple, en notation polaire, $u(r, \theta) = r^2 \cos(2k\theta)$, $k \in \mathbb{N}^*$, avec f à déterminer.