Méthodes numériques : optimisation.
Préparation à l'examen de mai 2015 — Éléments de correction

Amic Frouvelle

5 mai 2015

1 Laplacien discrétisé sur un carré

1. Il suffit de le montrer pour $h > 0$ puisque la formule ne change pas lorsqu'on remplace h par $-h$ (et n'a pas de sens pour $h = 0$, il y avait une petite coquille dans l'énoncé). On écrit les développements de Taylor à l'ordre 4 en $h > 0$ pour $g(x+h)$ et $g(x-h)$:

$$\begin{aligned} g(x+h) &= g(x) + hg'(x) + \tfrac{1}{2}h^2 g''(x) + \tfrac{1}{6}h^3 g^{(3)}(x) + \tfrac{1}{24}h^4 g^{(4)}(c_+) \\ g(x-h) &= g(x) - hg'(x) + \tfrac{1}{2}h^2 g''(x) - \tfrac{1}{6}h^3 g^{(3)}(x) + \tfrac{1}{24}h^4 g^{(4)}(c_-), \end{aligned}$$

avec $c_+ \in [x, x+h]$ et $c_- \in [x-h, x]$. En sommant les deux lignes et divisant par h^2, on obtient

$$\left| g''(x) - \frac{g(x+h) - 2g(x) + g(x-h)}{h^2} \right| = \tfrac{1}{24}h^2 |g^{(4)}(c_+) + g^{(4)}(c_-)| \leqslant \frac{C_4}{12}h^2. \tag{1}$$

2. On remplace $\partial^2_{xx} u(x,y)$ par $\frac{1}{h^2}[u(x+h,y) - 2u(x,y) + u(x-h,y)]$ et de même on remplace $\partial^2_{yy} u(x,y)$ par $\frac{1}{h^2}[u(x,y+h) - 2u(x,y) + u(x,y-h)]$ dans l'équation

$$\begin{cases} \partial^2_{xx} u(x,y) + \partial^2_{yy} u(x,y) = f(x,y) & \text{si } (x,y) \in]0,1[\times]0,1[\\ u(x,y) = 0 & \text{si } x = 0 \text{ ou } x = 1 \text{ ou } y = 0 \text{ ou } y = 1, \end{cases}$$

et on cherche à satisfaire les égalités seulements aux points (x_i, y_j). En notant $u_{i,j}$ l'approximation de $u(x_i, y_j)$, comme on a $x_i + h = x_{i+1}$ et $x_i - h = x_{i-1}$, l'approximation de $u(x_i + h, y_j)$ est donc $u_{i+1,j}$ et celle de $u(x_i - h, y_j)$ est $u_{i-1,j}$, de même l'approximation de $u(x_i, y_j + h)$ est donc $u_{i,j+1}$ et celle de $u(x_i, y_j - h)$ est $u_{i,j-1}$. On obtient donc le problème approché suivant :

$$\begin{cases} \frac{1}{h^2}[u_{i+1,j} - 2u_{i,j} + u_{i-1,j}] + \frac{1}{h^2}[u_{i+1,j} - 2u_{i,j} + u_{i-1,j}] = f(x_i, y_j) & \text{pour } 1 \leqslant i,j \leqslant n \\ u_{i,j} = 0 & \text{si } i \text{ ou } j \text{ vaut } 0 \text{ ou } n+1. \end{cases}$$

En écrivant

$$\Delta_1 = \begin{pmatrix} 2 & -1 & 0 & \cdots & \cdots & 0 \\ -1 & 2 & -1 & \ddots & & \vdots \\ 0 & -1 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & -1 & 0 \\ \vdots & & \ddots & -1 & 2 & -1 \\ 0 & \cdots & \cdots & 0 & -1 & 2 \end{pmatrix},$$

et U la matrice des $u_{i,j}$ pour $1 \leqslant i,j \leqslant n$, on obtient que $(\Delta_1 U)_{i,j} = -u_{i-1,j} + 2u_{i,j} - u_{i+1,j}$ dès que $1 \leqslant j \leqslant n$ et $2 \leqslant i \leqslant n-1$, mais la formule est aussi vraie pour $i = 1$ ou $i = n$ si on

suppose $u_{0,j} = 0$ et $u_{n+1,j} = 0$. De même, on obtient que $(U\Delta_1)_{i,j} = -u_{i,j-1} + 2u_{i,j} - u_{i,j+1}$, qui de nouveau est valable pour $1 \leqslant i, j \leqslant n$ si on suppose $u_{i,0} = u_{i,n+1} = 0$.

Le problème approché devient donc

$$\frac{1}{h^2}(-\Delta_1 U)_{i,j} + \frac{1}{h^2}(-U\Delta_1)_{i,j} = f(x_i, y_j)$$

pour $1 \leqslant i, j \leqslant n$, c'est à dire $\Delta_1 U + U\Delta_1 + B = 0$, en notant $B_{i,j} = h^2 f(x_i, y_j)$.

3. On écrit

$$\begin{aligned}\langle A(U), V\rangle &= \mathrm{Tr}(\Delta_1 UV^T + U\Delta_1 V^T)\\ &= \mathrm{Tr}(UV^T\Delta_1) + \mathrm{Tr}U(V\Delta_1)^T)\\ &= \mathrm{Tr}(U[(\Delta_1 V)^T + (V\Delta_1)^T])\\ &= \langle U, A(V)\rangle,\end{aligned}$$

où on a utilisé la formule $Tr(MN) = Tr(NM)$ avec $N = UV^T$ et $M = \Delta_1$ pour passer de la première ligne à la deuxième, ainsi que le fait que $\Delta_1^T = \Delta_1$.

4. On utilise les formules $\sin(\theta - \varphi) = \sin\theta\cos\varphi - \cos\theta\sin\varphi$ et $\sin(\theta + \varphi) = \sin\theta\cos\varphi + \cos\theta\sin\varphi$, les termes en $\cos\theta\sin\varphi$ se simplifient et on obtient la formule demandée.

En prenant $\theta = \frac{jk\pi}{n+1}$ et $\varphi = \frac{k\pi}{n+1}$, on obtient $\sin\theta = (v_k)_j$, $\sin(\theta - \varphi) = (v_k)_{j-1}$ si $j \geqslant 2$ (et vaut $\sin 0 = 0$ si $j = 1$) et $(v_k)_{j+1} = sin(\theta + \varphi)$ si $j \leqslant n-1$ (et vaut $\sin k\pi = 0$ si $j = n$). La formule s'écrit donc

$$\begin{cases} 2(v_k)_1 - (v_k)_2 = \lambda_k(v_k)_1 & (\text{cas } j = 1)\\ -(v_k)_{j-1} + 2(v_k)_j - (v_k)_{j+1} = \lambda_k(v_k)_j & \text{pour } 2 \leqslant j \leqslant n-1\\ -(v_k)_{n-1} + 2(v_k)_n = \lambda_k(v_k)_n & (\text{cas } j = n),\end{cases}$$

avec $\lambda_k = 2(1 - \cos\varphi) = 2(1 - \cos(\frac{k\pi}{n+1}))$. Cela correspond exactement à $\Delta_1 v_k = \lambda_k v_k$.

On a donc obtenu n vecteurs propres v_k pour $1 \leqslant k \leqslant n$, associés aux valeurs propres λ_k. Pour $1 \leqslant k \leqslant n$, on a $0 < \frac{k\pi}{n+1} < \pi$, et comme la fonction cos est strictement décroissante sur $[0, \pi]$, les $\cos(\frac{k\pi}{n+1})$ sont tous distincts et dans $]-1, 1[$. On en déduit que les valeurs propres λ_k sont toutes distinctes et dans $]0, 4[$. Donc les vecteurs propres sont indépendants, on a donc une base de vecteurs propres associés à des valeurs propres strictement positives. La matrice Δ_1 est donc définie positive, et on sait alors que les différents espaces propres sont orthogonaux, ce qui nous donne que les v_k sont orthogonaux entre eux (rappel de la méthode : si $\lambda_k \neq \lambda_j$ et $\Delta_1 v_k = \lambda_k v_k$ et $\Delta_1 v_j = \lambda_j v_j$, on peut écrire $\lambda_k\langle v_k, v_j\rangle = \langle\Delta_1 v_k, v_j\rangle = \langle v_k, \Delta_1 v_j\rangle = \lambda_j\langle v_k, v_j\rangle$ et donc $\langle v_k, v_j\rangle = 0$).

Enfin pour calculer le nombre de conditionnement de la matrice Δ_1, on sait que la plus petite valeur propre est pour $k = 1$ (on a alors $\cos(\frac{k\pi}{n+1})$ le plus proche possible de 1) et la plus grande est pour $k = n$. On a $\lim_{n\to\infty}\cos(\frac{n\pi}{n+1}) = -1$ donc $\lim_{n\to\infty}\lambda_n = 4$ et en utilisant le développement limité $\cos t = 1 - \frac{1}{2}t^2 + O(t^4)$, on obtient que $2(1 - \cos t) \sim t^2$ lorsque $t \to 0$, donc $\lambda_1 \sim (\frac{\pi}{n+1})^2 \sim \frac{\pi^2}{n^2}$ lorsque $n \to \infty$. En conclusion, le nombre de conditionnement κ de la matrice Δ_1, donné par $\kappa = \frac{\lambda_n}{\lambda_1} = \frac{1-\cos(\frac{n\pi}{n+1})}{1-\cos(\frac{\pi}{n+1})}$, est équivalent à $\frac{4}{\pi^2}n^2$.

5. On prend $(k_1, \ell_1) \neq (k_2, \ell_2)$. On a

$$\langle V_{k_1,\ell_1}, V_{k_2,\ell_2}\rangle = \mathrm{Tr}(V_{k_1,\ell_1}, V_{k_2,\ell_2}^T) = \mathrm{Tr}(v_{k_1}v_{\ell_1}^T(v_{k_2}v_{\ell_2}^T)^T) = \mathrm{Tr}(v_{k_1}v_{\ell_1}^T v_{\ell_2}v_{k_2}^T).$$

Mais comme $v_{\ell_1}^T v_{\ell_2}$ est une matrice 1×1 (c'est en fait le produit scalaire $\langle v_{\ell_1}, v_{\ell_2}\rangle$ dans $\mathbb{R}^n$), c'est un scalaire que l'on peut sortir de la trace par linéarité, on obtient donc

$$\langle V_{k_1,\ell_1}, V_{k_2,\ell_2}\rangle = \langle v_{\ell_1}, v_{\ell_2}\rangle\mathrm{Tr}(v_{k_1}v_{k_2}^T) = \langle v_{\ell_1}, v_{\ell_2}\rangle\mathrm{Tr}(v_{k_2}^T v_{k_1}) = \langle v_{\ell_1}, v_{\ell_2}\rangle\langle v_{k_1}, v_{k_2}\rangle,$$

puisque la formule $\mathrm{Tr}(MN) = \mathrm{Tr}(NM)$ est également vraie pour les matrices rectangulaires.

Par conséquent, si $(k_1, \ell_1) \neq (k_2, \ell_2)$, un des deux produits scalaires $\langle v_{k_1}, v_{k_2}\rangle$ ou $\langle v_{\ell_1}, v_{\ell_2}\rangle$ est nul (les vecteurs propres distincts sont orthogonaux deux à deux), et donc $\langle V_{k_1,\ell_1}, V_{k_2,\ell_2}\rangle = 0$. Les $V_{k,\ell}$ pour $1 \leqslant k, \ell \leqslant n$ forment donc une base orthogonale de $M_n(\mathbb{R})$.

On calcule ensuite

$$A(V_{k,\ell}) = \Delta_1 v_k v_\ell^T + v_k v_\ell^T \Delta_1 = \lambda_k v_k v_\ell^T + v_k (\Delta_1 v_\ell)^T = (\lambda_k + \lambda_\ell) v_k v_\ell^T = (\lambda_k + \lambda_\ell) V_{k,\ell}.$$

6. Les $V_{k,\ell}$ pour $1 \leqslant k, \ell \leqslant n$ sont donc une base orthogonale de vecteurs propres pour l'opérateur A, associés aux valeurs propres $\lambda_k + \lambda_\ell$. Comme les valeurs propres λ_k sont toutes strictement positives, les valeurs propres de l'opérateur A sont toutes aussi strictement positives, donc l'opérateur A est symétrique défini positif, donc en particulier il est inversible. Donc le problème $A(U) + B = 0$ a une unique solution, donnée par $-A^{-1}(B)$.
7. Étant donné que le problème $A(U) + B = 0$ était déjà une approximation de notre problème initial, cela ne sert à rien de vouloir la solution exacte. Une solution approchée pour laquelle l'erreur est de l'ordre de l'erreur faite en discrétisant le problème continu est satisfaisante.

   Résoudre le problème $A(U) + B = 0$ est équivalent à minimiser la fonction $U \mapsto \frac{1}{2}\langle U, A(U)\rangle + \langle B, U\rangle$. La méthode du gradient conjuguée est adaptée à ce problème, car c'est un problème de minimisation d'une fonction quadratique avec une matrice symétrique définie positive, et que c'est un problème en grande dimension (l'espace est $M_n(\mathbb{R})$, de dimension n^2).

   À chaque étape, le coût correspond à des calculs de produits scalaires entre matrices (qui nécessitent de faire n^2 multiplications et additions) ainsi qu'à un calcul de l'application A sur la matrice U. Ce dernier calcul, s'il est effectué en ne faisant que les opérations nécessaires (en ne multipliant pas par tous les zéros dans la matrice Δ_1) peut se faire en $2n^2$ multiplications, $4n^2$ soustractions et n^2 additions. En effet par exemple on a $(\Delta_1 U)_{i,j} = -u_{i-1,j} + 2u_{i,j} - u_{i+1,j}$ qui nécessite une multiplication (par 2) et deux soustractions (ou une seule pour $i = 1$ ou n), et on doit faire ce calcul pour $1 \leqslant i, j \leqslant n$, c'est à dire n^2 fois. De même pour le calcul de $U\Delta_1$, et il faut faire ensuite n^2 additions pour faire la somme de $\Delta_1 U$ et $U\Delta_1$.

   Ce que l'on retient c'est qu'il faut de l'ordre de n^2 opérations élémentaires par itération. Pour la méthode de gradient à pas fixe ou optimal, c'est la même chose, puisqu'à chaque étape, il faut calculer une fois le gradient, qui est donné par $A(U) + B$ (et calculer un produit scalaire dans le cas de la descente à pas optimal).
8. Le nombre de conditionnement de l'application A est donné par le quotient de sa plus grande valeur propre sur sa plus petite. Comme les valeurs propres sont de la forme $\lambda_k + \lambda_j$, la plus grande, que l'on note L est obtenue quand $k = j = n$ (on obtient donc $L = 2\lambda_n$), et la plus petite, notée ℓ, est obtenue quand $k = j = 1$ (on o. Le nombre de conditionnement est donc $\frac{L}{\ell} = \frac{2\lambda_n}{2\lambda_1}$, c'est donc le même nombre de conditionnement κ que celui de la matrice Δ_1.

   Pour le cas du gradient conjugué, le théorème d'estimation de l'erreur $e_k = \|U_k - U_*\|_A$ nous donne

$$e_k \leqslant 2\left(\frac{\sqrt{L} - \sqrt{\ell}}{\sqrt{L} + \sqrt{\ell}}\right)^k e_0.$$

   Pour le cas du gradient à pas optimal, le théorème nous donne $e_{k+1} \leqslant (\frac{L-\ell}{L+\ell})e_k$, donc on obtient

$$e_k \leqslant \left(\frac{L - \ell}{L + \ell}\right)^k e_0.$$

   Enfin pour le cas du gradient à pas fixe, on sait que le taux de convergence est $r(\alpha) = \max(|1 - \alpha\ell|, |1 - \alpha L|)$ et que le meilleur taux que l'on peut obtenir avec cette formule est le même que le taux de convergence pour le gradient à pas optimal, lorsque $\alpha = \frac{2}{L+\ell}$, pour lequel $r(\alpha) = \frac{L-\ell}{L+\ell}$.

9. Si on veut $\frac{e_k}{e_0} \leqslant \frac{1}{n^2}$, cela correspond (dans le cas du gradient conjugué), dans le pire des cas, à

$$2\left(\frac{\sqrt{L}-\sqrt{\ell}}{\sqrt{L}+\sqrt{\ell}}\right)^k \leqslant \frac{1}{n^2}.$$

En divisant par $\sqrt{\ell}$ au dénominateur et au numérateur, on obtient $2(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1})^k \leqslant \frac{1}{n^2}$, que l'on peut réécrire $2(1-\frac{2}{\sqrt{\kappa}+1})^k \leqslant \frac{1}{n^2}$. En prenant le logarithme, cela revient à $\ln 2 + k\ln(1-\frac{2}{\sqrt{\kappa}+1}) \leqslant -2\ln n$, c'est à dire que dès que k satisfait

$$k \geqslant \frac{-\ln 2 - 2\ln n}{\ln(1-\frac{2}{\sqrt{\kappa}+1})},$$

on est sûr qu'on atteint la tolérance voulue au bout de k itérations.

On peut calculer un équivalent de ce nombre maximal d'itérations, vu qu'on sait que $\kappa \sim \frac{4}{\pi^2}n^2$, on obtient que $\frac{2}{\sqrt{\kappa}+1} \sim \frac{\pi}{n}$, et que donc

$$\frac{-\ln 2 - 2\ln n}{\ln(1-\frac{2}{\sqrt{\kappa}+1})} \sim \frac{2}{\pi} n\ln n \text{ quand } n \to \infty.$$

Le nombre maximal d'itérations est donc de l'ordre de grandeur de $n\ln n$. Comme il y a de l'ordre de n^2 multiplications par itération, on aura donc fait au total de l'ordre de $n^3\ln n$ multiplications. Pour ce qui est du nombre de réels à stocker, il est de l'ordre de la dimension du problème (on doit stocker le point courant, ainsi que la direction de descente), ici n^2.

Pour la méthode de descente de gradient à pas optimal ou à pas fixe (avec le meilleur taux), on fait les mêmes calculs et on obtient que dès que k satisfait

$$k \geqslant \frac{-2\ln n}{\ln(1-\frac{2}{\kappa+1})},$$

on est sûr qu'on atteint la tolérance voulue au bout de k itérations. On obtient cette fois-ci

$$\frac{-2\ln n}{\ln(1-\frac{2}{\kappa+1})} \sim \frac{4}{\pi^2} n^2\ln n \text{ quand } n \to \infty.$$

Le nombre maximal d'itérations est donc dans ce cas de l'ordre de grandeur de $n^2\ln n$. Comme il y a de l'ordre de n^2 multiplications par itération, on aura donc fait au total de l'ordre de $n^4\ln n$ multiplications. De même, le nombre de réels à stocker est de l'ordre de n^2.

10. La première difficulté est d'écrire la matrice correspondant au système, qui est en dimension n^2. Dans le cas des méthodes simples présentées dans la question, on observe qu'il y a au plus $2n+1$ éléments non-nuls par ligne. On obtient donc au total de l'ordre de n^3 éléments non-nuls (il y a n^2 lignes). Il faut donc stocker au moins n^3 réels.

Application numérique : pour $n = 10^3$, une telle méthode prendrait de l'ordre de $8n^3$ octets d'espace mémoire, soit environ 8 Gigaoctets, et nécessiterait de l'ordre de 10^{12} opérations, soit une centaine de secondes environ.

Par comparaison, la méthode du gradient conjugué (ou à pas fixe ou optimal) nécessite de stocker de l'ordre de $8n^2$ octets d'espace mémoire, soit 8 Mégaoctets. La tolérance voulue est atteinte, pour la méthode du gradient conjugué, après de l'ordre de $10^9\ln 10^3$ opérations soit environ $7\cdot 10^9$ opérations, qui prenent huit dixièmes de secondes, alors que pour la méthode de descente de gradient à pas optimal (ou fixe avec le meilleur taux possible), on a de l'ordre de $10^{12}\ln 10^3$ opérations, soit de l'ordre de 700 secondes.

Évidemment ces calculs sont assez approximatifs, et ne donnent de résultat que dans le nombre maximal d'itérations, il se peut que cela soit plus rapide. Ils ont pour mérite de se donner une première idée du gain d'espace mémoire et de temps que l'on peut avoir en appliquant la méthode du gradient conjugué.

2 Taux de convergence de la méthode du gradient conjugué

1. On a $Ax_* + b = 0$. Donc on peut remplacer b par $-Ax_*$ dans les expressions de $f(x)$ et $f(x_*)$. On obtient

$$f(x) - f(x_*) = \tfrac{1}{2}\langle x, Ax\rangle + \langle b, x\rangle - \tfrac{1}{2}\langle x_*, Ax_*\rangle - \langle b, x_*\rangle = \tfrac{1}{2}\langle x, Ax\rangle - \langle Ax_*, x\rangle + \tfrac{1}{2}\langle x_*, Ax_*\rangle$$

   Et d'autre part

$$\tfrac{1}{2}\|x - x_*\|_A = \tfrac{1}{2}\langle x - x_*, A(x - x_*)\rangle = \tfrac{1}{2}\langle x, Ax\rangle - \tfrac{1}{2}\langle x, Ax_*\rangle - \tfrac{1}{2}\langle x_*, Ax\rangle + \tfrac{1}{2}\langle x_*, Ax_*\rangle,$$

   qui est bien égal au terme de droite de l'équation précédente puisque A est symétrique et qu'on a donc $\langle x, Ax_*\rangle = \langle x_*, Ax\rangle$.

2. On sait que $r_k = \nabla f(x_k)$ dans la méthode du gradient conjugué (avec les notations du cours). On a donc $r_k = Ax_k + b = Ax_k - Ax_* = Ae_k$.

   On a $x_{k+1} = x_k + \alpha_k p_k$, donc on peut écrire $x_k = x_0 + \sum_{i=0}^{k-1} \alpha_i p_i$. On a donc que $x_k - x_0 \in \mathrm{Vect}(p_0, p_1, \ldots, p_{k-1})$. Mais cet espace est le même que $\mathrm{Vect}(r_0, Ar_0, \ldots, A^{k-1}r_0)$ d'après le cours. Et donc comme $e_k - e_0 = x_k - x_* - x_0 + x_* = x_k - x_0$, on obtient que $e_k - e_0 \in \mathrm{Vect}(r_0, Ar_0, \ldots, A^{k-1}r_0)$. En remplaçant r_0 par Ae_0, on obtient le résultat voulu.

   On a donc des réels $a_1, a_2, \ldots, a_k$ tels que $e_k - e_0 = \sum_{i=1}^{k} a_i A^i e_0$, donc en notant $a_0 = 1$ et en posant $P_k = \sum_{i=0}^{k} a_i X^i$, on obtient que $P(0) = a_0 = 1$ et que $e_k = P_k(A)e_0$.

3. On veut montrer que e_k est l'unique minimiseur du problème suivant

$$\inf_{e \in e_0 + \mathrm{Vect}(p_0, \ldots, p_{k-1})} \|e\|_A^2,$$

   puisque $\mathrm{Vect}(p_0, \ldots, p_{k-1}) = \mathrm{Vect}(Ae_0, \ldots, A^k e_0)$ comme on l'a vu dans la question précédente.
   On a

$$e_k - e_0 = x_k - x_0 = x_{k-1} + \alpha_{k-1}p_{k-1} - x_0 = \cdots = x_0 + \sum_{i=0}^{k-1} \alpha_i p_i - x_0 = \sum_{i=0}^{k-1} \alpha_i p_i,$$

   On sait également que l'algorithme s'arrête au bout de n itérations, ce qui signifie que $e_n = 0$, donc en appliquant la formule précédente, on obtient donc $e_0 = -\sum_{i=0}^{n-1} \alpha_i p_i$.
   Pour $e - e_0 \in \mathrm{Vect}(p_0, \ldots, p_{k-1})$, on a $e - e_0 = \sum_{i=0}^{k-1} t_i p_i$, avec des réels $t_0, \ldots, t_{k-1}$. On a donc

$$e = e - e_0 + e_0 = \sum_{i=0}^{k-1} t_i p_i - \sum_{i=0}^{n-1} \alpha_i p_i = \sum_{i=0}^{k-1} (t_i - \alpha_i) p_i + \sum_{i=k}^{n-1} \alpha_i p_i$$

   Et donc, comme les vecteurs p_i sont conjugués deux à deux pour A, on obtient

$$\|e\|^2 = \langle e, Ae\rangle = \sum_{i=0}^{k-1} (t_i - \alpha_i)^2 \|p_i\|_A^2 + \sum_{i=k}^{n-1} \alpha_i^2 \|p_i\|_A^2.$$

   Cette somme (de termes positifs) atteint son unique minimum lorsque tous les t_i sont égaux à α_i, c'est à dire lorsque $e - e_0 = e_k - e_0$, ce qui est bien le résultat voulu.

   Comme dans la question précédente, le fait que $e - e_0 \in \mathrm{Vect}(Ae_0, \ldots, A^k e_0)$ est équivalent au fait que $e = P(A)e_0$ avec P un polynôme de $\mathbb{R}_k[X]$ vérifiant $P(0) = 1$, ce qui donne le deuxième résultat.

4. On écrit $e_0 = \sum_{i=1}^{n} t_i v_i$, et on a donc $Ae_0 = \sum_{i=1}^{n} \lambda_i t_i v_i$, et plus généralement $A^j e_0 = \sum_{i=1}^{n} \lambda_i^j t_i v_i$, et donc $P(A)e_0 = \sum_{i=1}^{n} P(\lambda_i) t_i v_i$.
   La base v_i étant orthogonale, on a

$$\|P(A)e_0\|_A^2 = \langle P(A)e_0, AP(A)e_0\rangle = \sum_{i=1}^{n} \lambda_i P(\lambda_i)^2 t_i^2 \|v_i\|^2.$$

5. On a aussi d'après la question précédente $\|e_0\|_A^2 = \sum_{i=1}^n \lambda_i t_i^2 \|v_i\|^2$ (en prenant $P = 1$). Et donc

$$\|P(A)e_0\|_A^2 \leqslant \sum_{i=1}^n \lambda_i \Big(\max_{\lambda\in\Lambda} |P(\lambda)|^2 \Big) t_i^2 \|v_i\|^2 = \max_{\lambda\in\Lambda} |P(\lambda)|^2 \|e_0\|_A^2.$$

D'après la question 3, on sait que pour tout polynôme P de $\mathbb{R}_k[X]$ tel que $P(0) = 1$, on a

$$\|e_k\|_A^2 = \|P_k(A)e_0\|_A^2 \leqslant \|P(A)e_0\|_A^2.$$

On obtient donc bien le résultat voulu.

6. *Application :* on suppose que la matrice A a une seule « petite » valeur propre $\frac{1}{\kappa} > 0$ (avec $\kappa \gg 1$) et que toutes ses autres valeurs propres sont dans $[1-\rho, 1]$ avec $\rho \in]0, 1 - \frac{1}{\kappa}[$.

   (a) Le pire des cas est quand la plus grande valeur propre vaut 1, alors le nombre de conditionnement vaut κ puisque la plus petite valeur propre est $\frac{1}{\kappa}$. D'après le cours, on sait qu'on a cette estimation :

$$\|e_k\|_A \leqslant 2 \left(\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1} \right)^k \|e_0\|_A.$$

   Si on prend le polynôme $P(X) = (1 - \kappa X)(1 - X)^{k-1}$, on a bien $P(0) = 1$. Pour $\lambda \in \Lambda$, on a donc soit $\lambda = \frac{1}{\kappa}$ et alors $P(\lambda) = 0$, soit $\lambda \in [1-\rho, 1]$ et alors $P(\lambda) \in [(1 - \kappa(1-\rho))\rho^{k-1}, 0]$, donc $|P(\lambda)| \leqslant |\kappa - \kappa\rho - 1|\rho^{k-1} \leqslant \kappa\rho^{k-1}$. Et donc on a $\max_{\lambda\in\Lambda} |P(\lambda)| \leqslant \kappa\rho^{k-1}$, ce qui nous donne bien

$$\|e_k\|_A \leqslant \kappa\rho^{k-1}\|e_0\|_A.$$

   La convergence est donc plus rapide que ce qui est donné par le théorème si ρ est suffisamment petit (le taux de convergence est ρ, alors qu'il serait proche de 1 si κ est grand). Par exemple, si $\rho = \frac{1}{10}$ et $\kappa = 100$, le théorème donnerait $\|e_k\|_A \leqslant 2(\frac{9}{11})^{k-1}\|e_0\|_A$ alors qu'on a en fait au moins $\|e_k\|_A \leqslant \frac{1}{10^{k-1}}\|e_0\|_A$, qui est une meilleure estimation dès que $k \geqslant 1$ et qui converge bien plus rapidement vers 0.

   (b) On a $M^{-1}Av_i = M^{-1}\lambda_i v_i = \lambda_i[(\kappa - 1)(v_i \cdot v_1)v_1 + v_i]$, qui vaut $\lambda_i v_i$ si $i \neq 1$ (puisque v_i et v_1 sont orthogonaux), et qui vaut $\lambda_1\kappa v_1 = v_1$ si $i = 1$. Donc v_i est un vecteur propre de A associé à λ_i si $i \neq 1$ et associé à 1 si $i = 1$. On a donc une base orthogonale de vecteurs propres pour $M^{-1}A$, et donc le nombre de conditionnement est dans le pire des cas $\frac{1}{1-\rho}$, si la plus grande valeur propre λ_i pour $i \neq 1$ vaut 1 et la plus petite vaut $1 - \rho$.

   On peut avoir intérêt à utiliser M^{-1} comme préconditionneur, puisque le calcul de $M^{-1}x$ est peu couteux, et qu'on obtient une matrice avec un bien meilleur conditionnement (si par exemple $\kappa \gg 1$ mais que ρ est de l'ordre de grandeur de 1). Donc on peut espérer avoir une convergence plus rapide.

7. * On veut démontrer le résultat donné dans le cours. On suppose donc que les valeurs propres de A sont dans $[L, \ell]$. On considère le polynôme de Tchebychev $T_k \in \mathbb{R}^k[X]$ donné par les relations suivantes

$$T_0 = 1, \quad T_1 = X, \quad T_{n+1} = 2XT_n - T_{n-1} \text{ pour } n \geqslant 1.$$

   (a) Le résultat est vrai pour $k = 0$ et $k = 1$, on le montre par récurrence, en supposant que c'est vrai aux rangs k et $k - 1$.

   On a $T_{k+1}(\cos\theta) = 2\cos\theta\, T_k(\cos\theta) - T_{k-1}(\cos\theta) = 2\cos\theta\cos(k\theta) - \cos((k-1)\theta)$. Mais on a aussi $\cos((k+1)\theta) + \cos((k-1)\theta) = \cos(k\theta + \theta) - \cos(k\theta - \theta) = 2\cos\theta\cos(k\theta)$, et donc on obtient bien $T_{k+1}(\cos\theta) = \cos((k+1)\theta)$.

   Donc pour $x \in [-1, 1]$, on prend $\theta = \arccos x$ et on obtient $T_k(x) = \cos(k\theta) \in [-1, 1]$.

   (b) On obtient immédiatement que $P_k(0) = 1$ (attention ici, petit conflit de notation avec la question 2.), et si $x \in [\ell, L]$, alors $\frac{L+\ell-2x}{L-\ell} \in [-1, 1]$, et donc $|P_k(x)| \leqslant \frac{1}{T_k(\frac{L+\ell}{L-\ell})}$.

(c) On le fait encore par récurrence. C'est bon pour $k = 0$ et $k = 1$. Si on suppose que c'est vrai aux rangs k et $k-1$, on obtient

$$\begin{aligned} T_{k+1}(x) &= 2xT_k(x) - T_{k-1}(x) \\ &= \tfrac{1}{2}[(2x^2 + 2x\sqrt{x^2-1}) - 1)(x+\sqrt{x^2-1})^{k-1} + (2x^2 - 2x\sqrt{x^2-1} - 1)(x-\sqrt{x^2-1})^{k-1}]. \end{aligned}$$

Mais on a $(x \pm \sqrt{x^2-1})^2 = 2x^2 - 1 \pm 2x\sqrt{x^2-1}$ et donc on obtient bien

$$T_{k+1}(x) = \tfrac{1}{2}[(x+\sqrt{x^2-1})^{k+1} + (2x(x-\sqrt{x^2-1}) - 1)(x-\sqrt{x^2-1})^{k+1}].$$

(d) Si toutes les valeurs propres sont dans $[\ell, L]$, on a donc que $|P_k(\lambda)| \leqslant |T_k(\frac{L+\ell}{L-\ell})|^{-1}$ dès que $\lambda \in \Lambda$. On a donc $\|e_k\|_A \leqslant |T_k(\frac{L+\ell}{L-\ell})|^{-1}\|e_0\|_A$.

On prend $x = \frac{L+\ell}{L-\ell}$. On a $x \pm \sqrt{x^2-1} = \frac{(L+\ell)\pm 2\sqrt{L\ell}}{L-\ell} = \frac{(\sqrt{L}\pm\sqrt{\ell})^2}{(\sqrt{L}+\sqrt{\ell})(\sqrt{L}-\sqrt{\ell})}$, et donc

$$T_k(\frac{L+\ell}{L-\ell}) = \frac{1}{2}\left[\left(\frac{\sqrt{L}-\sqrt{\ell}}{\sqrt{L}+\sqrt{\ell}}\right)^k + \left(\frac{\sqrt{L}+\sqrt{\ell}}{\sqrt{L}-\sqrt{\ell}}\right)^k\right],$$

ce qui donne la première partie du résultat. La deuxième inégalité provient simplement du fait que $\left(\frac{\sqrt{L}-\sqrt{\ell}}{\sqrt{L}+\sqrt{\ell}}\right)^k + \left(\frac{\sqrt{L}+\sqrt{\ell}}{\sqrt{L}-\sqrt{\ell}}\right)^k \geqslant \left(\frac{\sqrt{L}+\sqrt{\ell}}{\sqrt{L}-\sqrt{\ell}}\right)^k$, et donc

$$\left[\left(\frac{\sqrt{L}-\sqrt{\ell}}{\sqrt{L}+\sqrt{\ell}}\right)^k + \left(\frac{\sqrt{L}+\sqrt{\ell}}{\sqrt{L}-\sqrt{\ell}}\right)^k\right]^{-1} \leqslant \left(\frac{\sqrt{L}+\sqrt{\ell}}{\sqrt{L}-\sqrt{\ell}}\right)^{-k} = \left(\frac{\sqrt{L}-\sqrt{\ell}}{\sqrt{L}+\sqrt{\ell}}\right)^k.$$

3 Méthode de moindres carrés non-linéaires (Gauss-Newton) [1]

Soient $r_i : \mathbb{R}^n \to \mathbb{R}$ pour $1 \leqslant i \leqslant k$ des fonctions de classe C^2. On notera $J(x) \in M_{k,n}(\mathbb{R})$ la matrice Jacobienne au point x de l'application $r : x \in \mathbb{R}^n \mapsto (r_i(x))_{1\leqslant i\leqslant k} \in \mathbb{R}^k$ (vu comme un vecteur colonne), c'est-à-dire que $J_{ij}(x) = \partial_{x_j} r_i(x)$. On cherche à minimiser $f(x) = \frac{1}{2}\sum_{i=1}^k r_i^2(x) = \frac{1}{2}\|r(x)\|^2$.

1. On a $\partial_{x_j} f(x) = \sum_{i=1}^k \partial_{x_j} r_i(x)\, r_i(x) = \sum_{i=1}^k J_{ij}(x)\, r_i(x) = (J^T(x)r(x))_j$ donc $\nabla f(x) = J^T(x)r(x)$.
2. On a

$$\begin{aligned} \partial_{x_k}\partial_{x_j} f &= \sum_{i=1}^k \partial_{x_j} r_i(x)\partial_{x_k} r_i(x) + \partial_{x_k}\partial_{x_j} r_i(x)\, r_i(x) \\ &= \sum_{i=1}^k J_{ij}(x)J_{ik}(x) + \sum_{i=1}^k (H_{r_i}(x))_{jk}\, r_i(x) \\ &= (J^T(x)J(x))_{jk} + \sum_{i=1}^k (H_{r_i}(x))_{jk}\, r_i(x), \end{aligned}$$

Ce qui nous donne que $H_f(x) = J^T(x)J(x) + \sum_{i=1}^k r_i(x)H_{r_i}(x)$.

3. L'équation vérifiée par la direction de descente d_k au point x_k est $H_f(x_k)d_k + \nabla f(x_k) = 0$. Dans ce cas particulier, en remplaçant les expressions de $H_f(x_k)$ et $\nabla f(x_k)$ par les résultats des deux premières questions, on obtient

$$J^T(x_k)J(x_k)d_k + \sum_{i=1}^k r_i(x_k)H_{r_i}(x_k)d_k + J^T(x_k)r(x_k).$$

4. Si les r_i sont linéaires, on obtient $H_{r_i} = 0$ et donc les deux formules coïncident.

1. Origine : examen 2014

5. La fonction f est de classe C^2 sur $\mathbb{R}^n$, donc S_0 est fermé. Comme il est borné, il est donc compact. La fonction H_f est bornée sur ce compact par une constante L, donc ∇_f est L-Lipschitz sur S_0. La fonction f atteint ses bornes sur le compact S_0, donc elle est bien bornée inférieurement. Comme on a pris le pas α_k satisfaisant la règle de Wolfe, on a bien toutes les hypothèses du théorème.

6. On a $\|\nabla f(x_k)\| = \|J^T(x_k)r(x_k)\| = \|J^T(x_k)J(x_k)d_k\| \leqslant \|J^T(x_k)J(x_k)\|\|d_k\|$ (on prend la norme matricielle subordonnée pour la matrice $J^T(x_k)J(x_k)$). La fonction $x \mapsto \|J^T(x_k)J(x_k)\|$ est continue sur le compact S_0, donc majorée par un réel $K > 0$.
   On a ensuite $\langle -\nabla f(x_k), d_k\rangle = \langle J^T(x_k)J(x_k)d_k, d_k\rangle = \|J(x_k)d_k\|^2 \geqslant \gamma^2\|d_k\|^2$
   Et donc $|\cos\theta_k| \geqslant \frac{\gamma^2}{K}$. On obtient donc $\frac{K^2}{\gamma^4}|\cos\theta_k|^2 \geqslant 1$ et au final

$$\sum_k \|\nabla f(x_k)\| \leqslant \frac{K^2}{\gamma^4}\sum_k \cos^2\theta_k\|\nabla f(x_k)\| < +\infty.$$

   La série de termes positifs est convergente, son terme général tend vers 0 : $\lim_{k\to\infty}\nabla f(x_k) = 0$.

7. Si $J(x)$ n'est pas injective, on ne peut pas avoir l'hypothèse que $\|J(x)v\| \geqslant \gamma\|v\|$, puisque le terme de gauche est nul lorsque $v \in \ker J(x)$. Et de toute façon, la solution de l'équation pour d_k n'est pas unique, donc on n'est pas assuré de la convergence de l'algorithme.

8. L'intérêt de prendre cette formule est plutôt que la direction de descente de Newton est que la matrice $J^T(x_k)J(x_k)$ pour résoudre le système linéaire est symétrique positive (et définie positive si $J(x_k)$ est injective), alors qu'on n'est pas sûr que $H_f(x_k)$ le soit.
   Cela permet d'assurer que la direction d_k est bien une direction de descente : en effet on a

$$\langle d_k, \nabla f(x_k)\rangle = \langle d_k, -J^T(x_k)J(x_k)d_k\rangle = -\|J_k(x_k)d_k\|^2,$$

   qui est négatif (et même strictement si $d_k \notin \ker J(x_k)$). On n'est pas sûr que ce soit le cas pour $H_f(x_k)$ si elle n'est pas symétrique définie positive.