L3 MINT | M325 Calcul Scientifique II
Université Paris–Sud | Centre d'Orsay
AN-EDP | Année 2016 – 2017

© J.-B. A. K. <jean-baptiste.apoung@math.u-psud.fr>

Examen : 15 juin 2017 (Durée 3h00)

CONSIGNES : Les notes de cours sur papier sont autorisées. Les téléphones portables ne sont pas autorisés La consultation des pages Internet, en particulier de votre message électronique, est interdite.

Le sujet est formé de 3 thèmes qu'on peut considérés indépendants.
On peut admettre les résultats des questions précédentes lors de la réponse à une question.

Thème - 1 *Inversion d'une matrice tridiagonale*

Note 1.

.

Dans cette partie, on s'intéresse à la résolution dans $\mathbb{R}^n$, d'un système de la forme

$$\mathbf{Ax} = \mathbf{f} \quad \text{où} \quad \mathbf{A} = \begin{pmatrix} a_1 & b_1 & 0 & \cdots & 0 & 0 \\ c_2 & a_2 & b_2 & 0 & & 0 \\ 0 & c_3 & a_3 & b_3 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & & \ddots & c_{n-1} & a_{n-1} & b_{n-1} \\ 0 & 0 & \cdots & 0 & c_n & a_n \end{pmatrix} \tag{1}$$

On posera $a = [a_1, \dots, a_n]^T$, $b = [b_1, \dots, b_n]^T$, $c = [c_1, \dots, c_n]^T$, où $c_1 = 0, b_n = 0$. [$(\cdot)^T$ est l'opérateur transposé].

Thème - 1- 1 *Cas particulier où le calcul explicite de A est symétrique définie positive*

On suppose que la matrice A est symétrique définie positive. L'application $(x, y) \longmapsto (Ax, y)$ définit alors un produit scalaire sur $\mathbb{R}^n$, qu'on note $(\cdot,\cdot)_A$ et désigne par $\|\cdot\|_A$ la norme associée. On désignera par V^* la transposée au sens du produit scalaire euclidien du vecteur ou matrice V.

$\boxed{\textbf{Q-1} \quad :}$ Soit $\{v_0, \dots v_{n-1}\}$ une famille de n vecteurs orthogonaux dans le produit scalaire $(\cdot,\cdot)_A$. On dit aussi que ces vecteurs sont A-conjugués. On les suppose en plus normés, toujours dans ce produit scalaire. On pose $M = [v_0 \dots v_{n-1}]$ la matrice dont la colonne i est v_i.
Soit x un vecteur quelconque de $\mathbb{R}^n$. Posons α ses composantes dans la base (v_i). On écrit alors $x = \sum_{i=0}^{n-1} \alpha_i v_i \equiv M\alpha$.

Q-1-1 : Montrer que $\alpha_i = (A\,x, v_i) \quad \forall\, 0 \le i \le n-1$. En déduire que $\alpha = M^* A x$, et conclure que $\underline{A^{-1} = MM^*}$.

Q-1-2 : En déduire que si l'on dispose de n vecteurs A-conjugués, le calcul de l'inverse de A est immédiat.

$\boxed{\textbf{Q-2} \quad :}$ On s'engage à présent à chercher n vecteurs A-conjugués, en orthonormalisant les vecteurs colonnes de A par le procédé d'orthonormalisation de Gram-Schmidt relativement au produit scalaire $(\cdot,\cdot)_A$.

Dans ce qui suit, A_i désignera la i-ème colonne de la matrice A, $i = 0, \dots, n-1$.

Q-2-1 : On pose $v_0 = A_0/\sqrt{(A_0, A_0)_A}$. Vérifier que $\|v_0\|_A = 1$.

Q-2-2 : On suppose construits $v_0, \dots v_{k-1}$, et on désire construire v_k. On écrit $v_k = A_k + \displaystyle\sum_{l=0}^{k-1} \gamma_l^k v_l$.

Montrer que $\gamma_l^k = -(A_k, Av_l) \quad \forall\, 0 \le l \le k-1$. On achève la construction de v_k en le normalisant.

Q-2-3 : Écrire une fonction C (ou un pseudo-code) qui prend en entrée la matrice A et qui retourne la matrice M.

Thème - 1- 2 A^{-1} *par décomposition LU (Cas où $a_i, b_i, c_i, i = 1\ldots, n$ sont quelconques)*

Q-3 : **Stockage optimal :** Montrer que si on a la décomposition LU de $A = LU$ alors on dispose de A^{-1}. Et que les facteurs L et U offrent en eux-mêmes un stockage avantageux par rapport au calcul explicite de A^{-1}.

Q-4 : **Construction effective**

On pose

$$L = \begin{pmatrix} d_1 & 0 & \cdots & \cdots & 0 \\ l_2 & d_2 & \ddots & & \vdots \\ 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & l_{n-1} & d_{n-1} & 0 \\ 0 & \cdots & 0 & l_n & d_n \end{pmatrix} \qquad U = \begin{pmatrix} 1 & r_1 & 0 & \cdots & 0 \\ 0 & 1 & r_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ \vdots & & \ddots & 1 & r_{n-1} \\ 0 & \cdots & \cdots & 0 & 1 \end{pmatrix}$$

Q-4-1 : Montrer que $LU = A$ si et seulement si

$$\begin{array}{lll} & d_1 = a_1, & d_1\, r_1 = b_1 \\ l_k = c_k, & d_k = -l_k\, r_{k-1} + a_k, & d_k\, r_k = b_k \quad k = 2, \ldots, n-1, \\ l_n = c_n, & d_n = -l_n\, r_{n-1} + a_n. & \end{array}$$

Q-4-2 : En résolvant ces équations d'inconnues $l_k, d_k, r_k, k = 1, \ldots, n$, déduire que l'on peut stocker A à l'aide de trois vecteurs c, a, b, de taille au plus n chacun. Déduire aussi que l'on peut encore stocker dans ces trois vecteurs la décomposition LU de A. Implémenter alors en langage C ()ou en pseudo-code) un algorithme de décomposition LU de la matrice tridiagonale A dont les diagonales principales sont stockées dans c, a, b comme décrit précédemment.

Q-4-3 : Expliquer comment calculer $z = A^{-1}y$.

Thème - 2 *Une méthode de projection 1D pour matrices de partie symétrique définie positive*

On suppose dans cette partie que la matrice A n'est pas nécessairement symétrique mais sa partie symétrique $A_S = \frac{A+A^T}{2}$ est définie positive.

Thème - 2- 1 *Dérivation d'une méthode de projection 1D*

Q-5 : **Forme générale de la méthode de projection** (*voir cours*)

Q-5-1 : Rappeler la forme générale de la méthode de projection sur les espaces $\mathcal{K}_k$ (support) et $\mathcal{L}_k$ (direction) pour la résolution du système linéaire $Ax = b$.

Q-5-2 : Montrer que dans le cas présent, la méthode n'est définie que si $\mathcal{L}_k = A\,\mathcal{K}_k$.

Q-5-3 : Critiquer cette méthode en terme de complexité spatiale.

Q-6 : **Projection 1D avec redémarrage**

Afin de remédier au problème d'espace mémoire posé par cette méthode, on opte pour une approche avec redémarrage : on fixe une dimension maximale des espaces de projection à m. Et lorsque $k = m$, on pose $x_0 = x_m$ et on recommence.

Q-6-1 : Montrer que lorsqu'on fixe $m = 1$ (c'est-à-dire $\mathcal{K}_k$ est de dimension 1 quelque soit k) la méthode devient

$$x_0 \quad \text{donné, et} \quad \forall\, k = 0, \ldots, \left\{ \begin{array}{lll} x_{k+1} & \in & x_k + \mathcal{K}_k \\ r_{k+1} & \perp & \mathcal{L}_k \end{array} \right. \qquad \text{où} \quad r_{k+1} = b - Ax_{k+1} \tag{2}$$

On parle alors de projection 1D, car ici dim$\mathcal{K}_k$ = dim$\mathcal{L}_k = 1$

Q-7 : On pose $\mathcal{K}_k =< r_k >$ le sous-espace vectoriel engendré par le vecteur $r_k = b - Ax_k$.

Q-7-1 : Montrer que la méthode s'écrit :

$$x_0 \quad \text{donné, et} \quad \forall\, k = 0, \ldots, \left\{ \begin{array}{l} x_{k+1} = x_k + \alpha_k\, r_k \\ \alpha_k = \dfrac{(r_k, Ar_k)}{(Ar_k, Ar_k)} \end{array} \right. \tag{3}$$

Q-7-2 : Fournir alors une fonction C ou un pseudo-code implémentant cet algorithme.

Q-8 : **Préliminaires**

Dans tout ce qui suit on désignera par $\|y\|$ la norme euclidienne du vecteur y. Et pour une matrice B, $\|B\|$ désignera la norme de la matrice B subordonnée à la norme euclidienne. Le *conditionnement* la matrice inversible B est défini par $cond(B) = \|B^{-1}\|\|B\|$.

Soi B une matrice symétrique définie positive, on définit la plus petite respectivement la plus grande valeur propre de B par

$$\lambda_{\min}(B) = \min_{x\in R^n, x\neq 0} \frac{(Bx,x)}{(x,x)}, \quad \text{resp.} \quad \lambda_{\max}(B) = \max_{x\in R^n, x\neq 0} \frac{(Bx,x)}{(x,x)}$$

Q-8-1 : Montrer que pour une matrice B symétrique réelle, on a

$$\lambda_{\min}(B)\,\|y\|^2 \leq (By,y) \leq \lambda_{\max}(B)\,\|y\|^2, \quad \forall y \in \mathbb{R}^n$$

Q-8-2 : Montrer que pour une matrice B quelconque, $\|By\| \leq \|B\|\,\|y\| \quad \forall y \in \mathbb{R}^n$

Q-9 : **Propriété de minimisation de la méthode de projection (3)**

Q-9-1 : Soit $\alpha \in \mathbb{R}$, on pose $x_{k+1} = x_k + \alpha r_k$. et on désigne par $r_{k+1} = b - Ax_{k+1}$.

Montrer qu'on a

$$\|r_{k+1}\|^2 = \|r_k\|^2 + \|Ar_k\|^2 \left(\alpha - \frac{(r_k, Ar_k)}{\|Ar_k\|^2}\right)^2 - \frac{(r_k, Ar_k)^2}{\|Ar_k\|^2} \tag{4}$$

Q-9-2 : En déduire que la méthode (3) vérifie la propriété suivante :

$$x_{k+1} \quad \text{minimise la fonction} \quad f(y) = \|b - Ay\|^2 \quad \text{sur} \quad x_k + \mathcal{K}_k$$

Q-10 : **Convergence et estimation d'erreur**

Q-10-1 : montrer que pour la méthode (3) on a $\|r_{k+1}\|^2 = \|r_k\|^2 - \frac{(r_k, Ar_k)^2}{\|Ar_k\|^2}$

Q-10-2 : En déduire qu'on a $\|r_{k+1}\|^2 \leq \|r_k\|^2 \left(1 - \frac{\lambda^2_{\min}(\frac{A+A^T}{2})}{\lambda_{\max}(A^TA)}\right)$

Q-10-3 : On pose $e_k = x_k - x$, l'erreur à l'itération k. En remarquant que $A\,e_k = -r_k$, montrer que

$$\|e_k\| \leq cond(A) \left(\sqrt{1 - \frac{\lambda^2_{\min}(\frac{A+A^T}{2})}{\lambda_{\max}(A^TA)}}\right)^k \|e_0\| \tag{5}$$

Q-10-4 : En déduire que si on pose $\mu = \lambda_{\min}\left(\frac{A+A^T}{2}\right)$ et $\sigma = \|A\|$, on a

$$\|x_k - x\| \leq cond(A) \left(\sqrt{1 - \frac{\mu^2}{\sigma^2}}\right)^k \|x_0 - x\| \tag{6}$$

Q-10-5 : Conclure que cette méthode converge pour tout x_0

Q-11 : **Comparaisons**

On considère la méthode 3 lorsque la matrice est symétrique définie positive.

Q-11-1 : Montrer que l'estimation d'erreur devient

$$\|x_k - x\| \leq cond(A) \left(\sqrt{1 - \frac{1}{cond(A)^2}}\right)^k \|x_0 - x\| \tag{7}$$

Q-11**-**2 **:** Comparer dans ce cas cette méthode à la méthode du gradient à pas variable. *On prendra en compte la vitesse de convergence théorique, et les complexités spatiales et temporelles.*

Q-12 **:** **Accélération de la convergence**

Q-12**-**1 **:** Quels sont les avantages de cette méthode ?

Q-12**-**2 **:** Cette méthode peut-elle être améliorée par préconditionnement ?

Thème - 3 *Un exemple Exemple d'application*

On considère l'équation aux dérivées partielles suivante : $\Omega \subset \mathbb{R}^2$.

$$\left\{ \begin{array}{l} -\Delta u(x,y) + \alpha \frac{\partial u}{\partial x}(x,y) = f(x,y), \quad \forall (x,y) \in \Omega =]a,b[\times]c,d[. \\ u(x,y) = g(x,y), \quad \forall (x,y) \in \partial\Omega, \end{array} \right. \tag{8}$$

où, f et g sont des fonctions données continues, et $\alpha \geq 0$ un réel donné.

Nous nous intéressons à sa discrétisation par différences finies sur une grille de $\bar{\Omega}$ de pas h_x et h_y dans les directions x et y, et de sommets $(x_i, y_j), 0 \leq i \leq n_x + 1, 0 \leq j \leq n_y + 1$ avec $x_0 = a, x_{n_x+1} = b, y_0 = c, y_{n_y+1} = d.$

On considère à présent le schéma suivant :

$$\left\{ \begin{array}{rcl} -\Delta_h(u)_{i,j} + \alpha \dfrac{u_{i,j} - u_{i-1,j}}{h_x} & = & f(x_i, y_j), \quad 1 \leq i \leq n_x \quad 1 \leq j \leq n_y \\ u_{i,j} & = & g(x_i, y_j) \quad i,j \text{ tels que } (x_i, y_j) \in \partial\Omega, \end{array} \right. \tag{9}$$

où $\Delta_h(v)_{i,j} = \dfrac{1}{h_x^2} v_{i-1,j} + \dfrac{1}{h_x^2} v_{i+1,j} + \dfrac{1}{h_y^2} v_{i,j+1} + \dfrac{1}{h_y^2} v_{i,j-1} - (\dfrac{2}{h_x^2} + \dfrac{2}{h_y^2}) v_{i,j} \quad 1 \leq i \leq n_x,\ 1 \leq j \leq n_y.$

Q-13 **:** **Analyse du schéma**

Q-13**-**1 **:** **Consistance.** Donner l'ordre de consistance de ce schéma en x et en y.

Q-13**-**2 **:** **Stabilité** Ce schéma est-il stable ?

Q-14 **:** **Résolution**

On pose pour inconnu le vecteur $U = (u_{i,j})$ formé de tous les $u_{i,j}$ y compris ceux sur le bord du domaine.

Le système (9) s'écrit donc $AU = F$.

Q-14**-**1 **:** Quel avantage offre cette approche ? (*Répondre en tenant compte des difficultés que peut poser la numérotation des noeuds internes du maillage.*)

Q-14**-**2 **:** Sans construire la matrice A montrer que le problème (9) rentre dans le cadre décrit par l'algorithme (3).

Q-14**-**3 **:** Est-il nécessaire de construire la matrice A pour appliquer l'algorithme (3) ? (*Justifiez en montrant comment à partir de (3) on peut calculer produit matrice vecteur AV et le résidu associé $R = F - AV$*)

Q-14**-**4 **:** Expliquer comment faire usage des résultats du **Thème 1** pour accélérer la convergence de la méthode (3) pour ce problème.