Maîtrise de Mathématiques Fondamentales et Applications
Grands Systèmes Linéaires
feuille de TP 7

12 Novembre 2009

Exercice - 1 *Factorisation Cholesky incomplète*

On appelle factorisation Cholesky incomplète d'une matrice symétrique A définie positive l'écriture $A = \tilde{L}\tilde{L}^t$ où la matrice $\tilde{L}$ est calculée par la factorisation Cholesky modifiée ainsi : l' élément $\tilde{L}_{i,j}$ n'est calculé que si l'élément $A_{i,j}$ n'est pas nul. Si cet élément est nul on pose $\tilde{L}_{i,j} = 0$.
1- Écrire une fonction MATLAB de prototype `function [L] = factoCholI(A)` calculant la factorisation incomplète Cholesky de la matrice A. Prévoir de "relâcher" la condition $A_{i,j} = 0$ en $|A_{i,j}| < \varepsilon$, où ε est un seuil à définir.

2- Générer la matrice `A = laplace2D(10,10)` (*voir tp4*), de la discrétisation par différences finies du Laplacien sur le carré unité. Évaluer `[L] = factoCholI(A)`. Puis comparer les remplissages des matrices A et L.

3- Calculer $cond_2(A)$ et $cond_2(L^{-t}L^{-1}A)$. Expliquer.

4- Écrire une fonction MATLAB `function [IL,JL,VL] = morsefactoCholI(IA,JA,VA)` qui calcule et stocke sous format CSR (*voit tp 5*) une matrice symétrique définie positive dont seule la partie triangulaire inférieure est stockée au format CSR.

Exercice - 2 *Gradient conjugué préconditionné*

Soit $A \in \mathcal{M}_n(\mathbb{R})$ une matrice symétrique définie positive, et $b \in \mathbb{R}^n$. On s'intéresse au préconditionnement de la résolution du système linéaire $Ax = b$. Soit C une matrice symétrique inversible.
1- Écrire l'algorithme du gradient conjugué préconditionné, par C, pour la résolution du système linéaire $Ax = b$.

2- Programmer l'algorithme du gradient conjugué préconditionné à travers une fonction MATLAB `function [x,res]= GradientCP (A, b, x0, C, tol, iterMax)`. Les notations sont celles du TP 6, où ici C représente la matrice de préconditionnement.(*L'appel de C^{-1} ne devra pas se faire explicitement, au contraire, l'évaluation de $z = C^{-1}r$ doit correspondre à la résolution du système $Cz = r$*).

3- Soit $A = D - E - F$ la décomposition de la matrice A en une somme de sa matrice diagonale (D) et de ses matrices triangulaires inférieure $(-E)$ et supérieure $(-F)$. Il faut noter que $F = E^t$ lorsque la matrice A est symétrique.

On donne ici quelques matrices de préconditionnement :

- $C = D$, correspond au préconditionnement *diagonal*.

- $C = \frac{\omega}{2-\omega}\left(\frac{D}{\omega} - E\right) D^{-1} \left(\frac{D}{\omega} - E^t\right)$, correspond au préconditionnement $SSOR$.
- $C = I$, où I désigne la matrice identité, correspond au cas sans préconditionnement.

On considère A et b la matrice et le second membre obtenus par discrétisation par différences finies du laplacien sur $]0,1[\times]0,1[$ avec pour second membre $f = 1$ et des conditions aux limites de Dirichlet homogènes. La grille utilisée est définie par $h_x = \frac{1}{n+1}, h_y = \frac{1}{m+1}$.

a- Écrire un script MATLAB pour générer A et b. (*On pourra recourir aux précédents TPs*).

b- Pour les différentes matrices de préconditionnement ci-dessus, représenter sur un même graphique la courbe du logarithme des normes des résidus retournés. On prendra $n = m = 40$ et pour le préconditionnement SSOR $\omega = \frac{2}{1+2\sin\frac{\pi}{2n}}$. On fixera `tol = 1e-11` et `Itermax = 1000`.