UNIVERSITÉ DE RENNES 1 **Année 2005-2006**
U.F.R. Mathématiques

TRAVAUX PRATIQUES D'OPTIMISATION

4. Méthode du gradient conjugué

Remarque : Utiliser la fonction MATLAB `fminbnd` pour la détermination du minimum d'une fonction définie sur $\mathbf{R}$.

Méthode du gradient conjugué pour les fonctions quadratiques

Soit J la fonctionnelle quadratique

$$J(u) = \frac{1}{2} u^t A u - b^t u + c,$$

où $A \in \mathbf{R}^{n,n}$ est symétrique définie positive, $b \in \mathbf{R}^n$, $c \in \mathbf{R}$.

On note $\nabla J(u)$ le gradient relatif au produit scalaire usuel $(.,.)$ de $\mathbf{R}^n$, et $J'(u)$ le gradient relatif au produit scalaire $< .,. >$ défini par

$$< u, v >= (C^{-1}u, v),$$

où C est une matrice symétrique définie positive. On a donc $J'(u) = C\nabla J(u)$.

Algorithme :

- On choisit la valeur initiale u^0. On calcule le résidu $r^0 = \nabla J(u^0) = Au^0 - b$.
La première direction de descente est $v^0 = Cr^0$.
- à l'itération k :
on définit le point $u^{k+1} = u^k - \mu_k v^k$
où μ_k réalise le minimum sur $\mathbf{R}$ de $g(\mu) = J(u^k - \mu v^k)$, c'est-à-dire

$$\mu_k = \frac{(Cr^k, r^k)}{(Av^k, v^k)};$$

on calcule ensuite le résidu $\qquad r^{k+1} = Au^{k+1} - b, \qquad$ (arrêt si $\|r^{k+1}\| \leq \tau$),

$$\lambda_{k+1} = \frac{(Cr^{k+1}, r^{k+1})}{(Cr^k, r^k)},$$

et la direction de descente $\qquad v_{k+1} = \lambda_{k+1} v^k + Cr^{k+1}$

- On prendra pour le test d'arrêt la valeur $\tau = 10^{-6}$.

1 - Écrire un programme MATLAB mettant en œuvre la méthode du gradient conjugué pour une fonctionnelle quadratique elliptique. On utilisera, comme au TP précédent, trois fichiers MATLAB : `J.m` qui servira à définir la fonctionnelle J, `GJ.m` qui contiendra l'expression du gradient de J et `Gcquad.m` qui contiendra le programme principal mettant en œuvre la méthode du gradient conjugué. Un paramètre `numex` sera utilisé afin de distinguer les différents exemples.

2 - Inclure dans votre programme les fonctions graphiques `/usr/local/ananum/opti/tp3/visiso.m` et `/usr/local/ananum/opti/tp4/vitergc.m`. Visualiser les lignes de niveaux et les directions de descente

pour les exemples 1, 2 et 3 du TP précédent. Comparer aux résultats obtenus par la méthode du gradient à pas optimal. (On prendra d'abord ici pour C la matrice identité, puis $C = \texttt{inv}(A)$).

3 - Reprendre ces essais en utilisant la fonction MATLAB `fminbnd` pour effectuer la recherche de μ_k. Dans le cas de l'exemple 3, comparer la convergence de la méthode sans réinitialisation, avec réinitialisation toutes les n itérations, toutes les $n+1$ itérations, toutes les $n+2$ itérations.

Pour assurer une meilleure convergence des méthodes de gradients conjugués, (surtout dans le cas non quadratique) il convient de briser la suite des directions conjuguées. Cela consiste, à intervalles réguliers, à prendre pour direction de descente la direction $v^k = C\nabla J(u^k)$ (on réinitialise l'algorithme). La réinitialisation est effectuée avec une périodicité qui est ajustée manuellement en fonction du problème considéré (en général autour de n itérations, n étant le nombre de variables).

Méthode du gradient conjugué pour une fonctionnelle quelconque

Pour une fonctionnelle quelconque, l'algorithme du gradient conjugué admet plusieurs formulations (qui sont équivalentes pour une fonctionnelle quadratique).

Méthode de Flechter-Reeves (1964): on prend $\lambda_{k+1} = \dfrac{(Cr^{k+1}, r^{k+1})}{(Cr^k, r^k)}$, où $r^k = \nabla J(u^k)$.

Méthode de Polak-Ribière (1971): on prend $\lambda_{k+1} = \dfrac{(Cr^{k+1}, r^{k+1} - r^k)}{(Cr^k, r^k)}$.

4 - Mettre en œuvre ces 2 méthodes. On utilisera la fonction MATLAB `fminbnd` pour effectuer la recherche du paramètre μ_k. tout d'abord sur les deux problèmes suivants:

$$(\mathcal{P}_6) \ : \ \min \ (x_1 - 1)^2 + p(x_1^2 - x_2)^2,$$

Il s'agit de la fonction de Rosenbrock. On prendra $p = 10$ et $u_0 = (0, 1)$ (le minimum global est atteint en $u^* = (1, 1)$). On affichera la suite des points (u_k) et les directions de descente successives et on comparera au résultat obtenu avec la méthode du gradient à pas optimal (TP précédent). Pour le préconditionnement, on utilisera d'abord $C = Id$, et ensuite $C = diag([1, 4])$.

Exemple de Colville:

$$(\mathcal{P}_7) \ : \ \min \ 100(x_2-x_1^2)^2+(1-x_1)^2+90(x_4-x_3^2)^2+(1-x_3)^2+10.1[(x_2-1)^2+(x_4-1)^2]+19.8(x_2-1)(x_4-1).$$

Le minimum est atteint au point $(1, 1, 1, 1)$; on prendra pour point de départ $u_0 = (-3, -1, -3, -1)$ et pour le préconditionnement, on utilisera d'abord $C = Id$, et ensuite $C = diag([1/4, 1, 2/7, 1])$.

5 - Comparer le nombre d'itérations nécessaire pour obtenir la solution du problème de Colville par
- la méthode de Flechter-Reeves sans réinitialisation;
- la méthode de Flechter-Reeves avec réinitialisation toutes les $n+2$ itérations;
- la méthode de Polak-Ribière sans réinitialisation;
- la méthode de Polak-Ribière avec réinitialisation toutes les $3n$ itérations.

6 - Mettre en œuvre la méthode du gradient conjugué sur l'exemple du TP 2

$$J(x) = \frac{n+1}{2} \sum_{i=1}^{n+1} (x_i - x_{i-1})^2 + \frac{1}{4(n+1)} \sum_{i=1}^{n} x_i^4 - \frac{1}{(n+1)} \sum_{i=1}^{n} x_i \ b_i.$$

on prendra ici $n = 20$, $b_i = 1 \ \forall i$. On comparera la méthode sans préconditionnement, $C = Id$, et ensuite avec $C = \texttt{inv}\ M$, où la matrice M est définie par

$m_{ii} = 2$ pour $i = 1, \cdots, n$; $\quad m_{i,i+1} = m_{i+1,i} = -1$ pour $i = 1, \cdots, n-1$; $\quad m_{ij} = 0$ autrement.