ECP, semestre 4, Electif 11 Châtenay-Malabry, le 27 mai 2009
Numerical Optimization and applications
Mr Dumas

EXAMEN NUMERICAL OPTIMIZATION 2009

(**3 heures**, aucun document la première heure et demie, ordinateur la deuxième heure et demie)

A. PARTIE THEORIQUE

L'objectif de cette partie est de définir une nouvelle règle de recherche linéaire et de démontrer qu'elle génère une méthode de descente convergente.

On considère f une fonction C^1 de $\mathbb{R}^n$ dans $\mathbb{R}$, telle que $\lim\limits_{||x|| \to +\infty} f(x) = +\infty$. On note g la fonction gradient de f définie de $\mathbb{R}^n$ dans $\mathbb{R}^n$. On suppose que g est Lipschitzienne sur tout ensemble $S_{x_0} = \{x \in \mathbb{R}^n, f(x) \leq f(x_0)\}$.

1. Montrer que f possède un minimum global x^* pour lequel $g(x^*) = 0$.

2. On cherche à construire une méthode de descente $(x_k)_{k \in \mathbb{N}}$ définie à partir de $x_0 \in \mathbb{R}^n$ par la formule :

$$x_{k+1} = x_k + t_k d_k$$

où d_k désigne une direction de descente (c'est à dire telle que $(d_k, g(x_k)) < 0$) et t_k le pas dans cette direction déterminé par les conditions suivantes : on note $q(t) = f(x_k + td_k)$ et on choisit deux réels m_1 et m_2 tels que $0 < m_1 < m2 < 1$. Alors :

a) t_k est satisafaisant si $q(t_k) \leq q(0) + m_1 t_k q'(0)$ et $q'(t_k) \geq m_2 q'(0)$.
b) t_k est trop grand si $q(t_k) > q(0) + m_1 t_k q'(0)$.
c) t_k est trop petit si $q(t_k) \leq q(0) + m_1 t_k q'(0)$ et $q'(t_k) < m_2 q'(0)$

Donner un exemple graphique des valeurs de t_k admissibles dans le cas d'une fonction tracée 'à la main'. Comment interpréter la condition a) ?

3. Pour déterminer une telle valeur de t_k, on utilise l'algorithme suivant (de paramètres $t_{init} > 0$ et $\lambda > 1$) :

```
- Etape 0 : on part de t = t_init > 0. On note t_g = 0 et t_d = 0.
- Etape 1 : On teste t suivant les critères a) b) et c) :
  - si a), c'est fini
  - si b) (t trop grand), on note t_d = t et on passe à l'étape 2.
  - si c) (t trop petit), on note t_g = t et on passe à l'étape 2.
- Etape 2
  - si t_d = 0, on calcule t_new = λt
  - Si t_d > 0, on calcule t_new = (t_g+t_d)/2.
- Etape 3 : on boucle avec l'étape 1 avec le nouveau t.
```

On cherche à montrer que cet algorithme converge en un nombre fini d'étapes pour la fonction f choisie en préambule.

3.1 Montrer qu'il est impossible de répéter une infinité de fois la première condition de l'étape 2.

3.2 On suppose qu'on répète une infinité de fois la deuxième condition de l'étape 2. Montrer dans ce cas que t_g et t_d sont deux suites adjacentes de limite t^*.

3.3 Montrer alors qu'en ce point, $q(t^*) = q(0) + m_1 t^* q'(0)$. En utilisant un taux d'accroissement bien choisi, en déduire que $q'(t^*) \geq m_1 q'(0)$ ainsi que $q'(t^*) \leq m_2 q'(0)$. Conclure.

4. On cherche à montrer que la méthode de descente ainsi construite est convergente vers un point critique de f lorsque $d_k = -g(x_k) = -g_k$ (direction du gradient).

4.1 Montrer que $q'(0) = -||g_k||^2$ dans ce cas.

4.2 Montrer que $m_1 ||g_k|| \cdot ||x_{k+1} - x_k|| \leq f(x_k) - f(x_{k+1})$.

4.3 Montrer que $(1 - m_2)||g_k||^2 \leq (g_k - g_{k+1}, g_k)$ puis $(1 - m_2)||g_k|| \leq L||x_{k+1} - x_k||$ où L désigne la constante de Lipschitz de g dans S_{x_0}.

4.4 En composant et en sommant ces relations, montrer que $\lim_{k \to +\infty} g_k = 0$.

B. PARTIE APPLIQUEE

Le contenu de cette partie doit être réalisé dans un language à choisir entre Matlab et Scilab. Au terme de l'examen, le candidat enverra ses 2 programmes à l'adresse suivante : `laurent.dumas@upmc.fr` avec pour titre de mail 'ECP2009' et des noms de fichier du type 'ecp2009-prenom.nom.numero.sci'. En cas de défaillance du réseau, il est demandé de rendre ses programmes sur une clé USB, qui vous sera restituée par la suite.

Programme 1. Il existe peu de résultats de convergence des méthodes évolutionnaires. Neanmoins, on peut citer celui-ci datant de 2007 :

Soit une stratégie d'évolution $(1, 1)$ *représentée par la suite de vecteurs aléatoires* (X_k) *et ayant une mutation gaussienne de variance* $\sigma||X_k||$. *Sous certaines hypothèses techniques et pour une fonction sphérique, c'est à dire :*

$$J(x) = g(||x||)$$

avec $g : \mathbb{R}_+ \to \mathbb{R}_+$ *croissante,* X_k *converge presque surement log-linéairement vers le minimum global,* $X^* = 0$ *de* J, *c'est à dire :*

$$\lim_{k\to+\infty} \frac{1}{k} \ln\left(\frac{||X_k||}{||X_0||}\right) = c < 0 \quad (*)$$

L'objectif est de vérifier numériquement ce résultat.

1.1 Ecrire un programme Scilab (ou Matlab) simulant une stratégie d'évolution $(1, 1)$ pour une fonction J sphérique définie sur $\mathbb{R}^n$ avec une mutation gaussienne de variance $\sigma||X_n||$.

On rappelle que cela revient à construire une suite (X_k) telle que X_0 est choisie alétoirement (par exemple dans $[-A, A]^n$) et à contruire au rang k

$$\tilde{X}_k = X_k + \sigma||X_k||\mathcal{N}(0, 1) \qquad (**)$$

et à prendre $X_{k+1} = \tilde{X}_k$ si $J(\tilde{X}_k) < J(X_k)$ et $X_{k+1} = X_k$ sinon. Dans la relation (**), $\mathcal{N}(0, 1)$ désigne la réalisation d'une variable aléatoire gaussienne centrée réduite.

1.2 Vérifier graphiquement le résultat (*) dans le cas de la fonction $J(x) = ||x||^2$ avec $A = 10$, et $\sigma = 1$. Quelle valeur de c trouve t-on approximativement ?

Programme 2. Programmer la méthode de recherche linéaire de la partie théorique dans le cas de la direction de descente du gradient. On pourra créer une fonction

`t=recherchelineaire(x,f,g,tinit,lambda,m1,m2)`

qui renvoie la valeur de t trouvée au terme de l'algorithme proposé précédemment, au point x, pour une fonction f et sa fonction gradient associée g.

Donner la valeur de cette fonction dans le cas de la fonction de Rastrigin en dimension n :

$$f_n(x) = \sum_{i=1}^{n}(x_i^2 - \cos(2\pi x_i)) + n$$

au point $x = (1, 1, 1..., 1)$ avec les paramètres :

$$(t_{init}, lambda, m_1, m_2) = (10, 1.2, 0.1, 0.5)$$

.