Université Paris Sud
Modélisation et Calcul Scientifique
TP proposé par F. Dubois

Minimisation dans $\mathbb{R}$ et $\mathbb{R}^2$

20 et 27 janvier 2006

(1) Minimisation à une variable réelle.

- Soit $a < b$ deux nombres réels et $f : [a,b] \to \mathbb{R}$ une application qu'on supposera au moins continue. En suivant le cours de J.-C. Culioli [introduction à l'optimisation, Ellipses, 1994], on dit que f est <u>unimodale</u> sur $[a,b]$ si il existe $x^* \in]a,b[$ tel que f est décroissante strictement sur $]a, x^*[$ et croissante strictement sur $]x^*, b[$.

- on suppose f unimodale sur $[a,b]$. On pose $x_1 = a + \frac{1}{4}(b-a)$, $x_2 = a + \frac{1}{2}(b-a)$, $x_3 = a + \frac{3}{4}(b-a)$.
Montrer que selon la position relative de $f(x_1)$, $f(x_2)$, $f(x_3)$, on peut choisir $a' < b'$ tels que f est unimodale sur $[a', b']$ et $b' - a' = \frac{1}{2}(b-a)$.
En déduire un algorithme, dit de dichotomie, pour minimiser une fonction unimodale sur

un intervalle de $\mathbb{R}$. Le programmer avec le logiciel Matlab. Au cours d'une itération courante combien d'évaluations de la fonction f sont nécessaires à chaque étape ? Quelle est la vitesse de convergence de l'algorithme de dichotomie ?

- On suppose toujours f unimodale sur $[a,b]$. On note ϕ le nombre d'or $\phi = \frac{1}{2}(1+\sqrt{5})$,
  $$\varsigma = b + \frac{a-b}{\phi} < \xi = a + \frac{b-a}{\phi}.$$
  Montrer que selon la position relative de $f(\varsigma)$ et $f(\xi)$, on peut choisir $a' < b'$ de sorte que f soit unimodale sur $[a', b']$ et $b'-a' = \frac{1}{\phi}(b-a)$. En déduire un algorithme dit "de la section dorée" pour calculer le minimum de la fonction f. Le programmer avec Matlab, le tester. Combien d'évaluations de f sont nécessaires à chaque étape de l'algorithme (hors initialisation) ? Comparer l'efficacité de l'algorithme de la section dorée et l'algorithme de dichotomie.

- La minimisation dans $\mathbb{R}^n$ utilise en général une étape de minimisation à une variable. Les algorithmes précédents sont donc des briques de base pour des problèmes plus complexes. La difficulté pratique est d'assurer que f est bien unimodale sur $[a,b]$

(2) Minimisation dans $\mathbb{R}^2$.

- On appliquera les divers algorithmes présentés ci-dessous à la fonction quadratique

(1) $f(x,y) = (x-1)^2 + p\,(y-1)^2$, $p \simeq 10$

puis à la fonction de Rosenbrock

(2) $f(x,y) = (x-1)^2 + p\,((x^2-y))^2$, $p \simeq 10$.

- La méthode la plus simple, et parfois la plus efficace est la méthode du gradient à pas constant ρ :

(3) $u^{k+1} = u^k - \rho\, \nabla f(u^k)$, $k \in \mathbb{N}$, $\rho > 0$ fixé.

où $u^k = (x^k, y^k)$ représente un point de $\mathbb{R}^2$. Programmer cet algorithme et représenter à chaque itération d'une part l'état u^k et d'autre part le segment $[u^k, u^{k+1}]$. Pourquoi le paramètre ρ ne doit-il pas être choisi trop grand?

- Afin de s'affranchir de la contrainte du choix de ρ pour toutes les itérations de l'algorithme, la méthode du <u>gradient à pas optimal</u> consiste, une fois u^k et $\nabla f(u^k)$ connus, à chercher $\overline{\rho}_k > 0$ de sorte que

$$[0, \overline{\rho}_k] \ni t \mapsto f(u^k + t\,\nabla f(u^k)) \in \mathbb{R}$$

soit unimodale sur $[0, \overline{\rho}_k]$ puis à poser

$$(4) \quad u^{k+1} = u^k - \rho_k \nabla f(u^k), \quad k \in \mathbb{N}$$

avec $\rho_k > 0$ choisi tel que

$$(5) \quad f(u^k + \rho_k \nabla f(u^k)) \leq f(u^k + t\,\nabla f(u^k)), \quad \forall t \in [0, \overline{\rho}_k].$$

Programmer l'algorithme du gradient à pas optimal, avec un post-traitement analogue à la question précédente. Le tester pour la fonction quadratique et la fonction de Rosenbrock.

- L'étape suivante, avec l'<u>algorithme de Newton</u> suppose que pour tout $u \in \mathbb{R}^2$, le calcul de la Hessienne $H(u)_{ij} = \dfrac{\partial^2 f}{\partial x_i \partial x_j}(u)$ est facile, ce qui est loin d'être le cas dans beaucoup d'applications en grandes dimensions. Si f dérivable a un minimum u^* sur $\mathbb{R}^2$, il vérifie

$$(6) \quad \nabla f(u^*) = 0$$

et cette équation caractérise u^* si f a un unique minimum. Montrer que l'algorithme de Newton pour l'équation (6) s'écrit

(7) $$u^{k+1} = u^k - (H(u^k))^{-1} \cdot \nabla f(u^k), \quad k \geq 0$$

où $H(u^k)$ est la hessienne de f au point u^k.
Programmer l'algorithme de Newton comme pouvant être utilisé <u>après</u> l'algorithme du gradient à pas constant ou à pas optimal.
Le tester pour les deux fonctions précédentes.

- Comme la Hessienne est souvent impossible à calculer et parfois même il est impossible de résoudre en un temps raisonnable le système linéaire qui permet de calculer u^{k+1} avec la relation (7). On remplace donc les itérations de Newton par des itérations de <u>"quasi-Newton"</u>

(8) $$u^{k+1} = u^k - \rho^k d^k, \quad d^k = S^k \cdot \nabla f(u^k).$$

La direction de descente d^k n'est pas le gradient comme pour le gradient à pas optimal, mais ρ^k doit être calculé par les mêmes méthodes : trouver $\bar{\rho}_k$ de sorte que
$g(t) \equiv f(u^k - t d^k)$ soit unimodale sur $[0, \bar{\rho}_k]$, où $\bar{\rho}_k$ doit être déterminé au mieux.
D'ailleurs qu'aviez-vous proposé pour l'algo.

rithme de gradient à pas optimal?
Puis chercher ρ^k tel que

(9) $f(u^k - \rho^k d^k) \leq f(u^k - \rho d^k)$, $\forall \rho \in [0, \bar{\rho}_k]$.

- A l'étape 0, on choisit

(10) $S^0 = Id$.

ce qui permet d'initialiser l'algorithme de quasi Newton par une étape de gradient à pas optimal. Montrer par un développement de Taylor que si S^k doit approcher $(H(x^k))^{-1}$, il est raisonnable de supposer

(11) $S^{k+1}\gamma^k = \delta^k$,

avec

(12) $\gamma^k = \nabla f(u^{k+1}) - \nabla f(u^k)$, $\delta^k = u^{k+1} - u^k$.

On suppose que (11) a lieu pour les approches dites de "quasi-Newton".

- L'algorithme de Davidon-Fletcher-Powell s'écrit

(13) $$S^{k+1} = S^k + \frac{1}{(\delta^k, \gamma^k)} \delta^k (\delta^k)^t - \frac{1}{(\gamma^k, S^k\gamma^k)} S^k \gamma^k (\gamma^k)^t S^k ,$$

en plus de toutes les hypothèses précédentes. Vérifier que la relation (11) a bien lieu. Programmer l'algorithme de DFP et comparer son efficacité pour la fonction quadratique puis celle de Rosenbrock.

- Que proposez-vous pour améliorer l'efficacité de l'algorithme de quasi-Newton?

Le 16/01/06.

D.