REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE
UNIVERSITÉ ABDELHAMID IBN BADIS-MOSTAGANEM
FACULTE DES SCIENCES EXACTES ET INFORMATIQUE
DEPARTEMENT DE MATHEMATIQUES

Mémoire de Master en Mathématiques

OPTION : Modélisation, Contrôle et Optimisation

Sujet

Hessien généralisé et la méthode de Newton

Présenté par Mehallah fatima zohra

Soutenu le

26/05/2015

Devant le Jury

Mr BELGACEM Rachid	**Président**	**MCA**	**U. CHLEF**
Mr BOUKHARI Ahmed	**Examinateur**	**MCA**	**U. MEDIA**
Abdessamad AMIR	**Encadreur**	**MCA**	**U. MOSTAGANEM**

Table des matières

Remerciements

En préambule à ce mémoire, j'adresse ces quelques mots pour remercier notre grand Dieu tout puissant pour exprimer ma reconnaissance en vers sa grande générosité. Dieu m'a donné la volonté, la patience, la santé et la confiance durant toutes mes années d'études.

*Je tiens à remercier sincèrement Monsieur **AMIR Abdessamad**, qui, en tant que mon encadreur, s'est toujours montré favorable tout le long de la réalisation de ce mémoire, ainsi que pour son aide et le temps qu'il a bien voulu me consacrer.*

Je voudrais remercier également mes parents,mon époux, mon frère et mes deux sœurs et mes amies pour leur soutien moral, et tous ceux qui m'ont aidé dans cette expérience.

J'adresse mes remerciements aussi à notre chef de département de mathématiques, Mr Belhamiti Omar.

C'est, encore, un grand plaisir pour moi, d'adresser mes plus sincères remerciements à monsieurs : BELGACEM Rachid, BOUKHARI Ahmed d'avoir bien voulu présider mon jury, et d'en avoir fait partie.

J'adresse un autre remerciement à tous mes enseignants qui ont été toujours derrière nous.

Je remercie, enfin, toute personne qui m'a encouragé et m'a poussé de l'avant, sans oublié tous mes collègues de la promotion master 2 (2014-2015).

Résumé

Le but de ce travail est d'étudié la méthode de Newton avec une fonction objectif $C^{1.1}$(Fonctions différentiables avec Gradient Lipschitz). Pour se faire nous suivant un travail réalisé par O. Mangasarian ([5]) qui considère un Hessien généralisé. Nous avons fait une petite modification dans l'algorithme proposé par ce dernier qui concerne le calcul du Hessien généralisé. Les résultats obtenus étaient satisfaisantes.

Introduction

Une méthode classiquement utilisée pour résoudre le problème d'optimisation sans contraintes est la méthode de Newton, qui est une méthode d'optimisation d'ordre deux. L'idée derrière de cette méthode est étant donné un point de départ, nous construisons une approximation quadratique de la fonction objective. Nous minimisons alors la fonction approximative (quadratique) à la place de la fonction objective initiale. Nous utilisons ce minimum comme point de départ à l'itération suivante et en suite répéter la procédure itérative. Si la fonction objective est quadratique, l'approximation est exacte, et la méthode donne la solution en une seule itération. Si, d'autre part, la fonction objective n'est pas quadratique, et si le point de départ est suffisamment proche de la solution optimale, la méthode peut converger avec une vitesse quadratique.

Cette méthode suppose que la fonction f soit deux fois différentiable. Cependant, cette méthode demande à chaque itération, l'évaluation de la fonction objective f et l'évaluation de la matrice Hessienne de f.

Le but de ce travail est d'étudié la méthode de Newton avec une fonction objectif $C^{1.1}$(Fonctions différentiables avec Gradient Lipschitz). Pour se faire nous suivant un travail réalisé par O. Mangasarian [5] qui considère un Hessien généralisé. Nous avons fait une petite modification dans l'algorithme proposé par ce dernier qui concerne le calcul du Hessien généralisé. Les résultats obtenus étaient satisfaisantes.

Ce mémoire est organisé comme suit : Le premier chapitre est consacré à un rappel sur les fonctions convexes et calcul sous- différentielle, permettant de définir la matrice Hessienne généralisée. Au second chapitre, on présente la méthode de Newton pour l'optimisation sans contraintes. Nous verrons en dernier lieu, la méthode de Newton avec Hessien généralisé.

Chapitre 1

Hessien généralisé

L'objectif de ce chapitre est de présenter le concept de la Sous-différentiabilité pour les fonctions convexes.

1.1 Quelques propriétés des fonctions convexes

L'étude des fonctions convexes montrera que celles ci sont continues sur tout l'intérieur de leur domaine de définition et qu'elles sont presque partout différentiables.

Définition 1.1 (Fonctions convexes) *Soit une fonction $f : C \to \mathbb{R}$, où C un ensemble convexe non vide de $\mathbb{R}^n$. f est dite convexe sur C si :*

$$f(\alpha x_1 + (1-\alpha)x_2) \leq \alpha f(x_1) + (1-\alpha)f(x_2), \forall x_1, x_2 \in C, \forall \alpha \in [0,1]$$

f est dite strictement convexe si l'inégalité précédente est stricte pour $x_1 \neq x_2$.

Théorème 1.1 ([1]) *Soit C un ensemble convexe non vide de $\mathbb{R}^n$, et soit $f : C \to \mathbb{R}$ une fonction convexe. Alors f est continue sur l'intérieur de C.*

Définition 1.2 (Dérivée directionnelle) *Soit une fonction $f : C \to \mathbb{R}$, où C est un ouvert de $\mathbb{R}^n$. Soit $\overline{x} \in C$ et $d \in \mathbb{R}^n$ un vecteur non nul. La dérivée directionnelle de f en $\overline{x} \in C$ dans la direction $d \in \mathbb{R}^n$, noté par $f'(\overline{x}; d)$, est définie par la limite quand elle existe de :*

$$f'(\overline{x}; d) = \lim_{t \downarrow 0^+} \frac{1}{t}[f(\overline{x} + td) - f(\overline{x})].$$

Si la fonction f est différentiable en $\overline{x} \in C$, alors

$$f'(\overline{x}; d) = \langle \nabla f(\overline{x}), d \rangle,$$

où $\nabla f(\overline{x})$ est le gradient de f en $\overline{x}$.

Théorème 1.2 ([1]) *Soit une fonctionf $: C \to \mathbb{R}$, où C est un ouvert convexe de $\mathbb{R}^n$. Soit $\overline{x} \in C$ et $d \in \mathbb{R}^n$ un vecteur non nul. Alors si f est convexe $f'(\overline{x}; d)$ existe.*

1.2 Calcul sous-différentiel

Définition 1.3 *Soit un ensemble C non vide et $f : C \subset \mathbb{R}^n \to \mathbb{R}$ convexe. le point $\xi \in \mathbb{R}^n$ est appelé sous gradient de f au point $\overline{x} \in C$ si*

$$f(x) \geq f(\overline{x}) + \xi^T(x - \overline{x}) \ \forall x \in C.$$

L'ensemble des sous gradient de f en $\overline{x}$ est appelé le sous-différentiel de f en $\overline{x}$ est noté $\partial f(\overline{x})$:

$$\partial f(\overline{x}) = \{\xi \in \mathbb{R}^n : f(x) \geq f(\overline{x}) + \xi^T(x - \overline{x}) \ \forall x \in C\}.$$

Proposition 1.1 *$f : C \to \mathbb{R}$, C un ouvert convexe de $\mathbb{R}^n$. Si f est convexe différentiable. Alors*

$$\partial f(x) = \{\nabla f(x)\}.$$

Preuve. Soit $\xi \in \partial f(x)$ et $d \in \mathbb{R}^n$, d'où

$$f(x + \alpha d) \geq f(x) + \alpha \xi^T d, \tag{1.1}$$

de la différentiabilité de f, on a

$$f(x + \alpha d) = f(x) + \alpha \nabla f(x)^T d + \alpha \|d\| \varepsilon(\alpha d), \tag{1.2}$$

en remplaçant (1.2) dans (1.1), on obtient

$$\alpha[(\xi - \nabla f(x))^T d] - \alpha \|d\| \xi(\alpha d) \leq 0,$$

en dévisant par α et en faisant tendre α vers 0, on trouve

$$[\xi - \nabla f(x)]^T d \leq 0,$$

on prend $d = [\xi - \nabla f(x)]$, ce qui implique

$$\|\xi - \nabla f(x)\|^2 \leq 0 \Rightarrow \xi = \nabla f(x).$$

■

Lemme 1.1 *Soit $f(x) = max\{f_i(x), i \in I\}$, et $f(x_0) = f_i(x_0) \ \forall i \in I$. Alors le sous-gradient de f au point x_0 est :*

$$\partial f(x_0) = conv\{\nabla f_i(x_0), i \in I(x_0)\}.$$

où conv désigne l'enveloppe convexe d'un ensemble donné.

Exemple 1.1 *Soit $f(x) = |x|$. Alors,*

$$\partial f(x) = \begin{cases} +1 & si\ x > 0 \\ -1 & si\ x < 0 \\ \quad conv\{-1, +1\} = [-1, +1] & si\ x = 0. \end{cases}$$

Exemple 1.2 *Soit* $g : \Omega \subset \mathbb{R}^p \to \mathbb{R}$ *de classe* C^2 *en* Ω $(g \in C^2(\Omega))$ *et* f *définit sur* Ω *par :*

$$f(x) = [g^+(x)]^2 \text{ où } g^+(x) = \max(g(x), 0).$$

Il est clair que f *est differentiable, donc*

$$\nabla f(x) = 2g^+(x)\nabla g(x),$$

en remarque que $\nabla f(x)$ *n'est pas différentiable en* $g(x_0) = 0$*. En appliquant le lemme (1.1) sur la fonction* $x \mapsto \nabla f(x)$*, nous obtenons le sous-différentiel de cette fonction appelé Hessien généralisé associé à la fonction* f*, que nous notons*

$$\partial^2 f(x) = \begin{cases} \{2g(x_0)\nabla^2 g(x_0) + \nabla g(x_0)\nabla g(x_0)^T\} & si\ g(x_0) > 0, \\ \{0\} & si\ g(x_0) < 0, \\ 2\{\alpha \nabla g(x_0)\nabla g(x_0)^T \mid \alpha \in [0,1]\} & si\ g(x_0) = 0. \end{cases}$$

Un sous-gradient n'est pas toujours simple à calculer. Dans ce mémoire, nous considérons la fonction f défine par :

$$f(x) = \epsilon c^T x + (1/2)\left\|(Ax - b)_+\right\|^2, \tag{1.3}$$

tel que $A \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$, $c \in \mathbb{R}^n$. Cette fonction apparait souvent en optimisation, quand on pénalise un problème de programmation linéaire, dont la fonction objectif est $c^T x$, et $Ax = b$ la contrainte associé. Nous montrons ici qu'il est possible de calculer le Hessien Généralisé de f.

La fonction f définit par (1.3) est convexe, continue et différentiable avec un gradient

$$\nabla f(x) = A^T (Ax - b)_+ + \epsilon c, \tag{1.4}$$

le Hessien généralisé de f est

$$\partial^2 f(x) = A^T diag(Ax - b)_* A, \tag{1.5}$$

où $(x)_*$ est le sous gradient de $(x)_+$ qui vaut :

$$\begin{cases} (x_*)_i = 1 \text{ si } x_i > 0, \\ (x_*)_i = 0 \text{ si } x_i < 0, \\ (x_*)_i \in [0,1] \text{ si } x_i = 0. \end{cases}$$

Chapitre 2

Méthode de Newton

2.1 Conditions d'optimalité

Un problème d'optimisation sans contraintes est formulé comme suit :

$$\begin{cases} \min f(x) \\ x \in \mathbb{R}^n \end{cases}, \tag{2.1}$$

où $f : \mathbb{R}^n \to \mathbb{R}$, une fonction non linéaire. Résoudre le problème (2.1) revient à chercher des points de minimum locaux ou globaux.

Définition 2.1 (Minimum local et global) *Soit la fonction $f : \mathbb{R}^n \to \mathbb{R}$.*

(i) *Un point x^* est dit minimum local pour le problème (2.1) si :*

$$\exists r > 0 : \forall y \in B(x^*, r),\ f(x^*) \leq f(y).$$

(ii) *Un point x^* est dit minimum global pour le problème (2.1) si :*

$$\forall y \in \mathbb{R}^n,\ f(x^*) \leq f(y).$$

Nous nous intéressons dans la suite aux hypothèses sur la fonction objective f donnant les conditions d'optimalités caractérisant des solutions pour le problème (2.1).

Théorème 2.1 (Conditions d'optimalité nécessaires du premier ordre) *Supposons que la fonction $f : \mathbb{R}^n \to \mathbb{R}$ est différentiable. Si $x^* \in \mathbb{R}^n$ est un minimum local de f, alors :*

$$\nabla f(x^*) = 0. \tag{2.2}$$

Preuve. Soit $h \in \mathbb{R}^n$, $h \neq 0$. Pour s assez petit, on définit la fonction

$$\varphi : s \in \mathbb{R} \mapsto \varphi(s) = f(x^* + sh)$$

φ admet donc un minimum local en $s = 0$ ($\varphi(0) = f(x^*) \leq f(x^* + sh) = \varphi(s)$). D'où

$$\varphi'(0) = 0.$$

De la règle de dérivation d'une fonction composée en déduit

$$\varphi'(0) = \nabla f(x^*)^T h = 0.$$

Ceci étant vrai pour tout h, la condition nécessaire (2.2) suit. ∎

Le résultat ci-dessous, donne une situation où la condition donnée dans (2.2) soit aussi suffisante.

Théorème 2.2 (Conditions d'optimalité suffisantes du premier ordre)
Soit $f : \mathbb{R}^n \to \mathbb{R}$ une application différentiable et convexe. Un point x^* est un minimum global du problème (2.1) si seulement si :

$$\nabla f(x^*) = 0$$

Preuve. La condition est bien evidemment nécessaire, montrons qu'elle est suffisante.

Soient $x \in \mathbb{R}^n$ et $\alpha \in [0, 1]$. Puisque f est convexe on a

$$f(\alpha x + (1-\alpha)x^*) \leq \alpha f(x) + (1-\alpha)f(x^*).$$

On retranche $f(x^*)$ de chaque côté de l'inégalité, on note que

$$\alpha x + (1-\alpha)x^* = x^* + \alpha(x - x^*),$$

en divisant par α, On obtient l'inégalité :

$$\frac{f(x^* + \alpha(x - x^*)) - f(x^*)}{\alpha} \leq f(x) - f(x^*),$$

en faisant tendre α vers 0, on obtient

$$\nabla f(x^*)^T(x - x^*) \leq f(x) - f(x^*),$$

d'où

$$0 \leq f(x) - f(x^*) \quad \forall x \in \mathbb{R}^n.$$

■

Quand la fonction objective est deux fois différentiable, on a le résultat ci-dessous donnant les conditions nécessaire d'optimalité d'ordre deux.

Théorème 2.3 (Conditions d'optimalité nécessaires d'ordre deux) *Soit $f : \mathbb{R}^n \to \mathbb{R}$ une application deux fois différentiable sur un ouvert contenant x^*. Si x^* est un minimum local alors,*

(i) *On a la condition du premier ordre $\nabla f(x^*) = 0$.*

(ii) *La matrice Hessienne $D^2 f(x^*)$ est semi définie positive.*

Preuve. Supposons qu'il existe $d \in \mathbb{R}^n$ avec $d \neq 0$ tel que $\langle D^2 f(x^*)d, d\rangle < 0$. Le développement de Taylor à l'ordre deux en x^* donne

$$f(x^* + td) = f(x^*) + t\ \nabla f(x^*)^T d\ + \frac{1}{2}t^2 d^T D^2 f(x)d + o(t^2)$$

d'où

$$f(x^* + td) - f(x^*) = \frac{1}{2}t^2 d^T D^2 f(x)d + o(t^2)$$

puisque $o(t^2) \to 0$ quand $t \to 0$, en déduit que

$$f(x^* + td) < f(x^*)$$

pour des valeurs de t suffisamment petites, ceci est en contradiction avec le fait que x^* est un minimum local. ■

Théorème 2.4 (Conditions d'optimalité suffisantes du second ordre) *soit* $f : \mathbb{R}^n \to \mathbb{R}$, *et supposons que* $D^2 f$ *est continue sur un ouvert contenant* x^*,*tel que :*

(i) $\nabla f(x^*) = 0$.

(ii) $D^2 f(x^*)$ *est définie positive.*

Alors x^* *est un minimum local strict.*

Preuve. Puisque $D^2 f(x^*)$ est définie positive en x^*, alors vue la continuité de $D^2 f$ il existe $r > 0$, tel que $D^2 f$ reste définie positive pour tout x dans la boule ouverte $D = \{z \mid \|z - x^*\| < r\}$. on prend $d \in \mathbb{R}^n$ avec $\|d\| < r$, on a $x^* + d \in D$ et donc

$$\begin{aligned} f(x^* + d) &= f(x^*) + d^T \nabla f(x^*) + \frac{1}{2} d^T \nabla^2 f(z) d \\ &= f(x^*) + \frac{1}{2} d^T \nabla^2 f(z) d, \end{aligned}$$

avec $z = x^* + td$ pour $t \in [0, 1]$. Comme $z \in D$, on a $d^T \nabla^2 f(z) d > 0$ et donc

$$f(x^* + d) > f(x^*).$$

■

Théorème 2.5 ([4]) *Lorsque* f *est convexe, tout minimum local* x^* *est un minimum global de* f. *En plus si* f *est différentiable, alors tout point stationnaire* x^* *est un minimum global de* f.

Afin de donner le théorème de convergence de la méthode de Newton, nous donnons le résultat classique suivant.

Théorème 2.6 *Soit la fonction* $f : \mathbb{R}^n \to \mathbb{R}$ *de classe* C^1 *et* $d \in \mathbb{R}^n$, *alors on a*

$$f(x + d) = f(x) + \nabla f(x + td)^T d,$$

pour $t \in [0, 1]$. *Si* f *est de classe* C^2, *on a*

$$\nabla f(x + d) = \nabla f(x) + \int_0^1 \nabla^2 f(x + td) d \, dt,$$

et

$$f(x + d) = f(x) + \nabla f(x)^T d + \frac{1}{2} d^T \nabla^2 f(x + td) d.$$

pour tout $t \in [0, 1]$.

2.2 La vitesse de convergence

Définition 2.2 *Soit* $\left\{x^{(k)}\right\}_{k\in\mathbb{N}}$*une suite d'itéré générées par l'algorithme (2.3) qui converge vers la limite* x^**. Supposons que* $\forall k \in \mathbb{N}$, $x^{(k)} \neq x^*$*, La convergence de l'algorithme est dite :*

-Linéaire si l'erreur $e^{(k)} = \|x^{(k+1)} - x^{(k)}\|$ *décroit linéairement (i.e.), s'il existe* $\tau \in]0,1[$ *tel que :*

$$\lim_{k\longmapsto+\infty} \frac{\left\|x^{(k+1)} - x^{(k)}\right\|}{\left\|x^{(k)} - x^{(*)}\right\|} = \tau.$$

-Superlinéaire si :

$$\lim_{k\longmapsto+\infty} \frac{\left\|x^{(k+1)} - x^{(k)}\right\|}{\left\|x^{(k)} - x^{*}\right\|} = 0.$$

-d'ordre γ *s'il existe* $\tau \geq 0$ *tel que :*

$$\lim_{k\longmapsto+\infty} \frac{\left\|x^{(k+1)} - x^{(k)}\right\|}{\left\|x^{(k)} - x^{(*)}\right\|^{\gamma}} = \tau.$$

En particulier, si $\gamma = 2$ la convergence est dite quadratique.

2.3 Principe de la méthode

Supposons à présent que f est de classe C^2 et remplaçons f au voisinage de l'itéré courant x_k par son développement de Taylor de second ordre :

$$f(y) \sim q(y) = f(x_k) + \langle \nabla f(x_k), y - x_k\rangle + \frac{1}{2}\left\langle \nabla^2 f(x_k)(y - x_k), y - x_k\right\rangle$$

Le nouveau itéré x_{k+1} est choisi comme minimum de l'approximé quadratique q de f. Or le minimum de q est réalisé par x_{k+1} solution de l'équation $\nabla q(x_{k+1}) = 0$, ceci revient à résoudre

$$\nabla f(x_k) + \nabla^2 f(x_k)(x_{k+1} - x_k) = 0,$$

en supposant que $\nabla^2 f(x_k)$ est définie positive

$$x_{k+1} = x_k - \nabla^2 f(x_k)^{-1}\nabla f(x_k). \tag{2.3}$$

La méthode de Newton consiste alors à prendre pour direction de descente la direction

$$d_k = -\nabla^2 f(x_k)^{-1}\nabla f(x_k). \tag{2.4}$$

Cette méthode est bien définie si à chaque itération, la matrice Hessienne $\nabla^2 f(x_k)$ est définie positive.

Théorème 2.7 (convergence de la méthode de Newton) *Soit* f *est une fonction de classe* C^2*, dont la fonction* $x \mapsto \nabla^2 f(x)$ *est de lipshitz dans un voisinage de la solution* x^* *(vérifiant les conditions suffisantes d'odre 2). on considère l'itération* $x_{k+1} = x_k + d_k$*, où* d_k *est donné (2.4). Alors*

1. *si le point de départ x_0 est suffisamment proche de x^*, $\{x_k\}$ la suite converge vers x^* ;*
2. *la vitesse de convergence de $\{x_k\}$ est quadratique ; et*
3. *la suite des normes gradient $\{\|\nabla f_k\|\}$ converge quadratiquement vers zéro.*

Preuve. De la formule de Newton (2.4) et la condition d'optimalité $\nabla f(x^*) = 0$, on a

$$x_k + d_k - x^* = x_k - x^* - \nabla^2 f(x_k)^{-1} \nabla f(x_k)$$

$$= \nabla^2 f(x_k)^{-1}[\nabla^2 f(x_k)(x_k - x^*) - (\nabla f(x_k) - \nabla f(x^*))]. \tag{2.5}$$

D'après le théorème de Taylor : $\nabla f(x_k) - \nabla f(x^*) = \int\limits_0^1 \nabla^2 f(x_k + t(x^* - x_k))(x_k - x^*))dt,$

on a

$$\begin{aligned} & \left\|\nabla^2 f(x_k)(x_k - x_*) - (\nabla f(x_k) - \nabla f(x^*))\right\| \\ = & \left\| \int\limits_0^1 [\nabla^2 f(x_k) - \nabla^2 f(x_k + t(x^* - x_k))](x_k - x^*)dt \right\| \\ \leq & \int\limits_0^1 \left\|[\nabla^2 f(x_k) - \nabla^2 f(x_k + t(x^* - x_k))]\right\| \|x_k - x^*\| \, dt \end{aligned}$$

$$\leq \|x_k - x^*\|^2 \int\limits_0^1 Lt \; dt = \frac{1}{2} L \|x_k - x^*\|^2 , \tag{2.6}$$

où L est la constante de Lipschitz de $\nabla^2 f(x)$ pour x proche de x^*, comme $\nabla^2 f(x)$ est inversible, il existe un rayon $r > 0$ tel que $\left\|\nabla^2 f(x_k)^{-1}\right\| \leq 2 \left\|\nabla^2 f(x^*)^{-1}\right\|$ pour tout x_k avec $\|x_k - x^*\| < r$.

Par substitution de (2.5) et (2.6), on obtient

$$\|x_k + d_k - x^*\| \leq L \left\|\nabla^2 f(x^*)^{-1}\right\| \|x_k - x^*\|^2 = \acute{L} \|x_k - x^*\|^2 , \tag{2.7}$$

où $\acute{L} = L \left\|\nabla^2 f(x^*)^{-1}\right\|$. On choisit x_0 tel que $\|x_k - x^*\| < \min(r, 1/(2\acute{L}))$, on peut utiliser cette inégalité pour déduire par induction la convergence vers x^* avec une vitesse quadratique, d'où 1) et 2).

Montrons 3), d'après la relation $x_{k+1} - x_k = d_k$ et $\nabla f(x_k) + \nabla^2 f(x_k) d_k = 0$, on obtient

$$\begin{aligned}
\|\nabla f(x_{k+1})\| &= \left\|\nabla f(x_{k+1}) - \nabla f(x_k) - \nabla^2 f(x_k) d_k\right\| \\
&= \int_0^1 \left\|\nabla^2 f(x_k + t d_k)(x_{k+1} - x_k) dt - \nabla^2 f(x_k) d_k\right\| \\
&\leq \int_0^1 \left\|\nabla^2 f(x_k + t d_k) - \nabla^2 f(x_k)\right\| \|d_k\| \, dt \\
&\leq \frac{1}{2} L \|d_k\|^2 \\
&\leq \frac{1}{2} L \left\|\nabla^2 f(x_k)^{-1}\right\|^2 \|\nabla f(x_k)\|^2 \\
&\leq 2L \left\|\nabla^2 f(x^*)^{-1}\right\|^2 \|\nabla f(x_k)\|^2 .
\end{aligned}$$

Ce qui prouve que la suite des normes du gradient converge quadratiquement vers 0. ■

Algorithm 2.1 (Algorithme de Newton de base) *1-**Initialisation***

$k = 0$, choisir x_0 dans un voisinage de x^*(x^*est le minimum de f) et une précision $\varepsilon > 0$.

*2-**l'itération principale***

*a)**Critère d'arrêt***

si $\|\nabla f(x_k)\| < \varepsilon$ stop et $x^* = x_k$ sinon

*b)**l'itération k***

$$x_{k+1} = x_k - [\nabla^2 f(x_k)]^{-1} \nabla f(x_k),$$

on pose $k = k + 1$ et on retourne à 2.

Avantages et Inconvénients de la méthode de Newton

– La méthode ne converge pas globalement pour certains problèmes,

– Elle demande l'évaluation de la matrice Hessienne $\nabla^2 f(x_k)$ à chaque itération,

– À chaque itération, elle demande la résolution d'un système linéaire impliquant la matrice Hessienne qui peut être singulière ou mal-conditionnée.

– La puissance de la méthode de Newton réside dans sa convergence quadratique si x_0 est suffisamment proche de x^* pour vu que $\nabla^2 f(x^*)$ soit inversible. Cette difficulté et voir même les autres difficultés peuvent être surmonté en remplaçant la matrice Hessienne par une approximation définie positive comme on va le voir ci-dessous.

Chapitre 3

La méthode de Newton généralisée

Ce chapitre concerne le comportement de la méthode de Newton avec le Hessien généralisé. A présent, revenons à notre fonction f définie par la relation 1.3, avec $m > n$, son gradient est définie par la relation 1.4 et sa matrice Hessienne par la relation 1.5.

L'idée de La méthode de Newton généralisée est que le Hessien généralisé sera utilisée pour générer la direction de Newton donnée dans (2.5).

Proposition 3.1 *Soit A une matrice de dimension (m, n), le rang de A est égale à n et $diag(Ax - b)_*$ est semi définie positive , alors $A^T diag(Ax - b)_* A$ est semi définie positive.*

Preuve. On a, $m > n$, $A^T A$ est définie positive et $D = diag(Ax - b)_*$ est une matrice diagonale dont les éléments diagonaux sont donnés par

$$d_i = \begin{cases} 1, \, si \, [Ax - b]_i > 0, \\ \in [0, 1], \, si \, \, [Ax - b]_i = 0, \\ 0, \, si \, \, [Ax - b]_i < 0. \end{cases}$$

il est clair que les élément $d_i \geq 0$, alors D est semi défini positif. D'où

$$\left\langle A^T DAx, x \right\rangle = \langle DAx, Ax \rangle \geq 0.$$

■

Comme la matrice Hessienne généralisée de f est semi définie positive. Alors la direction de descente d_k ne peut être pas une direction de descente, on propose une méthode de Newton avec Hessien modifié.

3.1 Newton avec une matrice Hessienne Modifiée

L'algorithme de Newton exige que le Hessien de f soit défini positive pour que la direction choisie soit une direction de descente, on doit proposer une alternative pratique lorsque le Hessien ne vérifie pas cette condition. Cette dernière consiste à remplacer la matrice Hessienne définit dans (1.5) par

$$(\partial^2 f(x) + \lambda Id),$$

où $\lambda = \sqrt{\delta}$, pour une précision machine δ, qu'on peut prendre égale à10^{-16}. Comme on a démontré que le Hessien généralisé est semi défini positif, la direction modifiée est une direction de descente. D'ou

$$d_k = -(\partial^2 f(x_k) + \lambda Id)^{-1} \nabla f(x_k).$$

3.2 L'algorithme de la méthode de Newton généralisée :

Algorithm 3.1 *1-**Initialisation***

$k = 0$ choisir $x_0 \in \mathbb{R}^n$ ($x_0 = (\overline{A}^T A + \epsilon Id)^{-1}\overline{A}^T\overline{b}$ où ($\overline{A}$une sous matrice de A et $\overline{b}$ est le sou vecteur de b) et ε,$\lambda > 0$ (généralement $\varepsilon = 10^{-4}$,$\lambda = 10^{-12}$)

*2-**l'itération principale***

*a)**Critère d'arrêt***

si $\|\nabla f(x_k)\| < \varepsilon$ stop et $x^* = x_k$ sinon

*b)**l'itération k***

$$x_{k+1} = x_k - (\lambda Id + \partial^2 f(x_k))^{-1}\nabla f(x_k)$$

on pose $k = k + 1$ et on retourne à 2.

Lemme 3.1 *Soit x^* un point stationnaire de la fonction f. Alors il existe un $\epsilon > 0$ tel que*

$$\nabla f(x) - H(x - x^*) = 0 \tag{3.1}$$

pour tout $x \in B_\epsilon(x^*)$ et pour tout $H \in \partial(\nabla f(x))$.

Preuve. On a

$$\nabla f(x) = A^T[Ax - b]_+ + \epsilon c.$$

De plus, chaque élément $H \in \partial(\nabla f(x))$ est donné par

$$H = A^T DA$$

pour une matrice diagonale $D = diag(d_1, ..., d_n)$ avec des entrées d_i de telle sorte que :

$$d_i = \begin{cases} 1, \ si \ [Ax - b]_i > 0, \\ \in [0, 1], \ si \ \ [Ax - b]_i = 0, \\ 0, \ si \ \ [Ax - b]_i < 0. \end{cases}$$

Puisque x^* est un point stationnaire de la fonction f, il résulte que

$$A^T[A_i x^* - b]_+ = -\epsilon c_i \quad \forall j = 1,...,n.$$

Utilisons cette relation, on obtient pour tout $j \in \{1, ..., n\}$:

$$\begin{aligned} & [\nabla f(x) - H(x - x^*)]_j \\ = \ & A'_j[Ax - b]_+ + \epsilon c_j - H_j(x - x^*) \\ = \ & A'_j[Ax - b]_+ - A'_j[Ax^* - b]_+ - A'_j DA(x - x^*) \\ = \ & \sum_{i=1}^{n} a_{ji} \left(\max\{0, [Ax - b]_i\} - \max\{0, [Ax^* - b]_i\} - d_i \left[A(x - x^*)\right]_i\right) \\ = \ & : \sum_{i=1}^{n} a_{ji} S_i. \end{aligned}$$

Considérons cinq cas pour montrer que chaque S_i dans l'expression précédente est égale à zéro à condition que x est assez proche de x^*.

a) Si $[Ax^* - b]_i > 0$, alors $[Ax - b]_i > 0$ pour tout x assez proche de x^*, cela implique que $d_i = 1$. Donc nous obtenons immédiatement $S_i = 0$ dans ce cas.

b) Si $[Ax^* - b]_i < 0$, alors $[Ax - b]_i < 0$ pour tout x assez proche de x^* cela implique que $d_i = 0$. Et cela montre qu'on a $S_i = 0$ dans ce cas.

c) Si $[Ax^* - b]_i = 0$, et $[Ax - b]_i > 0$, on a $d_i = 1$ et donc $S_i = 0$.

d) Si $[Ax^* - b]_i = 0$, et $[Ax - b]_i < 0$, on a $d_i = 0$ et donc $S_i = 0$.

e) Si $[Ax^* - b]_i = 0$, alors $[Ax - b]_i = 0$, on a $d_i \in [0, 1]$, dans ce cas, nous avons

$$[A(x - x^*)]_i = [Ax + \epsilon c]_i - [Ax^* + \epsilon c]_i = 0 - 0 = 0,$$

nous voyons que $S_i = 0$ quelle que soit la valeur de d_i. ■

Théorème 3.1 (théorème de convergence de la méthode de Newton généralisée) *Chaque point d'accumulation x^* par la suite généré d'un algorithme de Newton généralisé est un point stationnaire et donc un minimum global de f.*

Preuve. Soit ϵ est la constante donnée dans (lemme 3.1), alors il résulte que chaque matrice $H \in \partial(\nabla f(x_k))$ est non singulière pour tout $x \in B_\epsilon(x^*)$, donc la matrice $H_k = \partial^2 f(x_k) \in \partial(\nabla f(x_k))$ est non singulière. De la direction de Newton généralisé, et d'après (3.1), on a

$$x_k + d_k - x^* = x_k - x^* - H_k^{-1}\nabla f(x_k) = -H_k^{-1}(\nabla f(x_k) - H_k(x_k - x^*)) = 0.$$

Par conséquent, l'algorithme de Newton généralisé prend $x_{k+1} = x_k + d_k = x^*$ comme une prochaine itération de l'itération principale, donc x^* est un point stationnaire de f. ■

3.3 Etude numérique

Mangasarian programme la méthode de Newton généralisée pour la fonction f, où il considère les éléments de $diag(Ax - b)_*$ comme suit

$$d_i = \begin{cases} 1, \; si \; [Ax - b]_i > 0, \\ 0, \; si \; [Ax - b]_i \leq 0. \end{cases}$$

Autrement dit, il prend toujours la valeur 0 pour les $[Ax - b]_i$ nuls. Le choix de Mangasarian est fait de sorte que l'algorithme ne perd pas du temps en calculant la direction de descente. Alors qu'en réalité d_i prend une valeur α dans $[0, 1]$. Notre étude pour cette problématique est uniquement numérique. Pour se faire on a considéré une fonction objective qui s'annule à l'optimum. Soit la fonction

$$f(x) = (1/2) \left\| (Ax - b)_+ \right\|^2, \tag{3.2}$$

où la matrice A est de rang maximal. Nous avons considéré une classe de problème dont la fonction objectif est de type (3.2), plus précisément on a considéré l'instance LPNET [6]. Nous avons testé sur cette instance deux algorithmes ; celui de Mangasarian qui prend toujours la valeur 0 pour les $[Ax - b]_i$ nuls, le nôtre prendra le meilleur α, en faisant varié α de 0 à 1 avec un pas qui vaut 0.05. Les résultats sont donnés dans le tableau ci-dessous. On a calculé aussi les valeurs données à ces problèmes par la fonction *fminsearch* de Matlab.

Les algorithmes sont exécutés sur un PC hp disposant d'un processeur Pentium (R) Dual-Core CPU avec une fréquence de 2.30GHz et une mémoire vive de 2 Go tournant sous Microsoft Windows 7.

LPNET	$\alpha = 0$			$\alpha Best$				fminsearch
	f	i	time	$\alpha Best$	f	i	time	fval
25FV47	1.3846e-031	500	3.1216	0.1500	6.1870e-033	500	5.0418	3.3716e+003
ADLITTLE	2.5244e-029	500	0.3106	0.3500	0	49	0.0395	6.8160e+006
AGG	0	53	0.1560	0.2500	0	50	0.2011	4.4901e+004
BEACONFD	0	95	0.1342	0	0	95	0.1643	6.6561e-009
BLEND	4.5214e-033	500	0.2734	0.2500	1.4694e-039	500	0.2302	27.7562
BNL1	6.0190e-036	500	2.0648	0.1500	0	74	0.4965	2.5778
BOEING2	0	58	0.0626	0.2500	0	51	0.0611	2.0393
DEGEN2	3.9443e-031	500	0.9562	0.8000	0	66	0.1676	4.0741e+003
E226	6.1630e-033	500	0.5973	0.2500	0	91	0.1470	115.8892
FFFFF800	0	0	0.0730	0	0	0	0.0518	0
FORPLAN	2.5364e-034	500	0.6336	0.0015	0	63	0.0931	0.0015
GANGES	1.6853e-030	500	3.1255	0.2500	8.7889e-032	500	3.3331	28.1405
LOTFI	1.1767e-038	500	0.4591	0.8500	0	55	0.0518	3.6644e-007
MAROS-R7	0	0	10.7120	0.1500	0	0	9.7253	0
NESM	0	191	2.4876	0.7000	0	143	1.9920	3.8157e+003
PEROLD	1.3560e-006	500	3.6354	0.2500	1.3019e-006	500	3.9778	0.2372
PILOT4	5.7126e-007	500	3.5891	0	5.7126e-007	500	3.1625	0.5036
SC50A	1.2278e-033	500	0.2173	0.3500	0	30	0.0168	0.0106
SC205	3.4115e-029	500	0.5234	0.3000	2.0925e-033	500	0.5596	0.0011
SCFXM1	4.7963e-028	500	0.7607	0.2500	1.6759e-029	500	0.9379	3.5731
SCORPION	5.4852e-028	500	4.1795	0.2500	5.7716e-030	500	0.5846	1.1342e+003
SCRS8	7.1882e-005	500	1.6162	0	7.1882e-005	500	2.5746	1.9571e+004
SCSD6	0	0	0.0065	0.7500	0	0	0.0019	0
SCTAP2	0	50	0.7954	0.5000	0	47	0.7772	9.7981
SEBA	1.2714e-031	500	6.3670	0.3000	1.4187e-032	500	10.4077	4.3423e+003
SHARE1B	4.4006e-006	500	0.3860	0.4000	4.4006e-006	500	0.4841	161.2134

LPNET	$\alpha = 0$			$\alpha Best$				fminsearch
	f	i	time	$\alpha Best$	f	i	time	fval
SHIP12L	3.9304e-026	500	19.1075	0.7500	0	48	2.0059	8.5967e+007
SIERRA	2.8966e-008	500	6.4090	0.0500	8.6519e-011	500	6.4056	7.8690e+008
STANDATA	0	67	0.2386	0.9000	0	67	0.2179	0.0098
STOCFOR1	1.4981e-010	500	0.4280	0.0500	1.2639e-010	500	0.4060	2.2739e+005
TRUSS	0	0	0.3614	0.6000	0	0	0.2885	0
VTP-BASE	5.8769e-039	500	0.5243	0.3000	3.5873e-043	500	0.4367	0.0011
WOOD1P	0	0	0.0216	0.9000	0	0	0.0089	0
WOODW	0	0	0.4038	0.0500	0	0	0.4358	0

CONCLUSION

Nous avons présenté un algorithme de Newton généralisé pour résoudre un problème de minimisation sans contraintes pour une fonction de classe $C^{1.1}$. Nous avons essentiellement pris en considération le calcul approximatif du Hessien généralisé. L'étude numérique a donnée grande satisfaction. Reste à confirmer ceci par des résultats théorique.

Bibliographie

[1] Mokhtar S. Bazaraa D. Sherali C.M. Shetty. Nonlinear Programming. Theory and Algorithms. Third Edition 2006.

[2] Christian Kanzow, Houduo Qi, and Liqun Qi. On the minimum norm solution of linear programs. (Revised version, September 2001).

[3] Jean-Baptiste Hiriart-Urruty, Jean-Jacques Strodiot, and V. Hien Nguyen. Generalized Hessian Matrix and Second-Order Optimality Conditions for Problems with $C^{1,1}$ Data. Appl. Math. Optim. 11 :43-56 (1984) Springer-Verlag New York Inc.

[4] G. Nocedal, and S. J. Wright. Numerical Optimization. Springer Series in Operation Research and Financial Engineering. Second Edition 2006.

[5] O. L. MANGASARIAN. A Newton Method for Linear Programming. Journal of optimization theory and applications : Vol. 121, No. 1, pp. 1–18, April 2004.

[6] http ://www.cl.uni-heidelberg.de/