



*République Algérienne Démocratique et Populaire*  
*Ministère de l'enseignement supérieur et de la*  
*recherche scientifique*

*Université Larbi Tébessi - Tébessa*

*Faculté des Sciences Exactes et des Sciences de la Nature et de la Vie*  
*Département : Mathématiques et Informatique*



*Mémoire de fin d'études*  
*Pour l'obtention du diplôme de MASTER*  
*Domaine : Mathématiques et Informatique*

*Filière : Mathématique*

*Option : Equation aux dérivées partielles et applications*

*Thème*

**Nouvelle preuve de la convergence globale de la méthode  
BFGS pour les problèmes de minimisations non convexes**

*Présenté Par :*

*Aouina Anfel*

*Merrouche Zineb*

*Devant le jury :*

*Dr. Messaoudene Hadia MCA Université Larbi Tébessi Présidente.*

*Dr. Mesloub Fatiha MCA Université Larbi Tébessi Examinatrice .*

*Dr. Degaichia Hakima MCA Université Larbi Tébessi Encadreuse*

*Date de soutenance : 25/6/2020*

# Remerciements

*Nous tenons tout d'abord à remercier Dieu le tout puissant, qui nous a donné la force et la patience d'accomplir ce Modeste travail.*

*Nous tenons à adresser notre remerciements les plus chaleureux et notre profonde gratitude à notre encadreur Dr. Degaichia Hakîma, MCA à l'Université Elarbi Tebessi-Tebessa pour nos avoir proposé le sujet de ce mémoire. C'est grâce à sa grande disponibilité, ses conseils, ses orientations et ses encouragements que nous avons pu mener à bien ce travail.*

*Nous sommes conscientes de l'honneur que nous a fait Dr. Messaoudene, MCA en étant président du jury et Dr. Mesloub Fatiha, MCA d'avoir accepté d'examiner ce travail.*

*Notre remerciement s'adresse également à tous nos professeurs pour leurs générosités et la grande patience dont ils ont su faire preuve malgré leurs charges académiques et professionnelles.*

*Nos profonds remerciements vont également à toutes les personnes qui nous ont aidés et soutenue de près ou de loin.*

*Enfin, Merci à tous mes amis et tous mes collègues sans exception, nous n'oublions pas de remercier toutes les personnes qui nous ont facilité la tâche*

# Résumé

**Dans ce mémoire on s'intéresse à une Nouvelle preuve de la convergence globale de la méthode BFGS pour les problèmes de minimisations non convexes avec des recherches linéaires inexacts. De plus, nous avons vérifié directement la convergence de cette méthode avec la recherche linéaire inexacte de Wolfe**

**Mots clés: R-L de Wolfe, convergence globale, méthode de BFGS, recherche linéaire inexacte, Problèmes de minimisations non convexe.**

# Abstract

**In this memory we give a new proof for the global convergence of the BFGS method for nonconvex unconstrained minimization problems . Furthermore, we have checked directly the convergence of this method with the inaccurate linear search of Wolfe.**  
**Key words : Wolfe, global convergence, BFGS method, inexact line search, nonconvex unconstrained minimization problems.**

## المخلص

في هذه الأطروحة سنقوم بإثبات جديد للتقارب الشامل لطريقة BFGS في حل المسائل الأمثلية غير المحدبة وذلك باستخدام عمليات البحث بالطرق الخطية غير الدقيقة الكلمات المفتاحية:  
وولف ، التقارب الشامل ، طريقة BFGS ، البحث الخطي غير الدقيق، المسائل الأمثلية غير المحدبة

# Table des matières

<b>1 Optimisation sans contraintes. Notions de base</b>	<b>5</b>
1.1 Différentiabilité . . . . .	5
1.1.1 Dérivée partielle . . . . .	5
1.1.2 Gradient . . . . .	5
1.1.3 Matrice Hessienne . . . . .	6
1.1.4 Dérivée directionnelle . . . . .	6
1.1.5 Fonction différentiable . . . . .	6
1.2 Définitions . . . . .	6
1.3 Direction de descente . . . . .	7
1.4 Schémas général des algorithmes . . . . .	8
1.4.1 Exemple de choix de directions de descente . . . . .	9
1.4.2 Exemple de choix de pas $\lambda_k$ . . . . .	9
1.5 Condition nécessaire d'optimalité . . . . .	11
1.5.1 Condition nécessaire d'optimalité du premier ordre . . . . .	11
1.5.2 Condition nécessaire d'optimalité du second ordre . . . . .	11
1.6 Condition suffisante d'optimalité . . . . .	12
1.7 Conditions nécessaires et suffisantes d'optimalité dans le cas convexe . . . . .	12
<b>2 Optimisation unidimensionnelle</b>	<b>14</b>
2.1 Recherches linéaires . . . . .	14
2.1.1 Principe des méthodes de descente . . . . .	14
2.1.2 Recherche linéaire . . . . .	16
2.1.3 Les recherches linéaires exactes . . . . .	17

2.1.4	Les recherches linéaires inexactes . . . . .	18
2.1.5	Convergence des méthodes utilisant des recherches linéaires inexactes et des directions de descente. Le théorème de Zoutendijk . . . . .	22
<b>3</b>	<b>Méthodes quasi-Newtoniennes</b>	<b>24</b>
3.1	Méthode de Newton . . . . .	24
3.1.1	Avantages et inconvénients . . . . .	25
3.2	Méthodes de Quasi-Newton . . . . .	26
3.2.1	Quelques propriétés des algorithmes de quasi-Newton . . . . .	27
3.2.2	Formules de mise à jour de l'approximation du Hessien . . . . .	27
3.2.3	Méthode de correction de rang un (SR1 [1960]) . . . . .	28
3.2.4	Méthode de Davidon-Fletcher-Powell (DFP [D-1959 et FP-1963]) . . . . .	30
3.2.5	Méthode de Broyden-Fletcher-Goldfard-Shanno (BFGS [1970]) . . . . .	32
<b>4</b>	<b>Nouvelle preuve de la convergence globale de la méthode BFGS dans le cas non convexe</b>	<b>36</b>
4.1	Méthode BFGS avec mise à jour appropriée . . . . .	36
4.2	Convergence globale de Méthode BFGS avec mise à jour appropriée . . . . .	38
4.3	Convergence globale de l'algorithme de Méthode BFGS avec mise à jour appropriée avec la recherche linéaire de Wolfe . . . . .	43
4.4	Tests numériques . . . . .	49
4.5	Conclusion . . . . .	55

---

# Introduction Générale

Les problèmes d'optimisation différentiable se posent lorsque l'on cherche à déterminer la valeur optimale d'un nombre fini de paramètres. L'optimalité signifie ici la minimalité d'un critère donné. La différentiable supposée des fonctions qui définissent le problème écarte d'emblée de notre propos l'optimisation combinatoire (les paramètres à optimiser ne prennent que des valeurs entières ou discrètes) et l'optimisation non lisse (les fonctions ont des irrégularités).

L'optimisation est un sujet très ancien. Taylor [1685-1731], Newton [1643-1727], Lagrange [1736-1813] et Cauchy [1789-1857] ont élaboré les bases des développements limités. L'optimisation a connu un nouvel essor depuis l'apparition des ordinateurs et s'applique désormais dans de très nombreux domaines : économie, gestion, planification, logistique, automatique, robotique, conception optimale, science de l'ingénieur, traitement du signal, etc.

Soit  $f : \mathbb{R}^n \rightarrow \mathbb{R}$ . Considérons le problème d'optimisation sans contraintes  $(P)$  suivant :

$$(P) : \underset{x \in \mathbb{R}^n}{\text{Min}} f(x)$$

Les méthodes quasi-Newton, et particulièrement la méthode BFGS est parmi les méthodes les plus célèbres pour résoudre le problème  $(P)$ . Cette méthode a été étudiée et développée par Broyden [6], Fletcher [32], Goldfarb [10] et Shanno [11]. La méthode BFGS est une solution souvent utilisée lorsque l'on veut un algorithme à directions de descente.

On s'intéresse dans ce mémoire aux problèmes de convergence de la méthode BFGS dans le cas non convexe avec des recherches linéaires inexacts de Wolfe [26].

On essaye de mettre en lumière et d'étudier de façon approfondie les principaux résultats de convergence dans le cas où la fonction objectif  $f$  n'est pas convexe.

Pour cela on construit une suite  $\{x_k\}_{k \in \mathbb{N}}$  de la forme suivante :

$$x_{k+1} = x_k + \lambda_k d_k, \quad (1)$$

où  $d_k$  vérifie :

$$M_k d_k + g_k = 0, \quad g_k = \nabla f(x_k), \quad (2)$$

Les matrices  $M_k$  sont calculées itérativement d'une façon semblable à celle de la méthode BFGS. Plus explicitement :

$$M_{k+1} = \begin{cases} M_k - \frac{M_k s_k s_k^T M_k}{s_k^T M_k s_k} + \frac{y_k y_k^T}{y_k^T s_k}, & \text{si } \frac{y_k^T s_k}{\|s_k\|^2} \geq \epsilon \|g_k\|^\alpha \\ M_k, & \text{sinon} \end{cases} \quad (3)$$

$$s_k = x_{k+1} - x_k \quad \text{et} \quad y_k = g_{k+1} - g_k,$$



---

$\alpha$  et  $\varepsilon$  sont des constantes positives.  $\lambda_k$  est un pas positif obtenu par une recherche linéaire de Wolfe .

Bien sur l'expression précédente est une modification de la formule classique de la méthode BFGS dont l'expression des matrices  $M_k$  est la suivante :

$$M_{k+1} = M_k - \frac{M_k s_k s_k^T M_k}{s_k^T M_k s_k} + \frac{y_k y_k^T}{y_k^T s_k}, \quad (4)$$

Le mémoire est divisé en 4 chapitres :

- Dans **le premier chapitre** on expose les notions de base pour étudier le problème (P).
- Dans **le deuxième chapitre** on étudie l'optimisation unidimensionnelle c-à-d les recherches linéaires, spécialement les recherches linéaires inexactes.
- Dans **le troisième chapitre** on donne un aperçu général sur les méthodes quasi-Newtoniennes surtout la méthode BFGS.
- La partie essentielle du mémoire se trouve dans **le chapitre 4** dans lequel on s'intéresse aux problèmes de convergence de la méthode BFGS avec la recherches linéaires inexactes de Wolfe, dans le cas non convexes.

On trouve à la fin du mémoire une conclusion.

# Chapitre 1

## Optimisation sans contraintes. Notions de base

### 1.1 Différentiabilité

On se place dans  $\mathbb{R}^n$ ,  $n < \infty$  considéré comme un espace vectoriel normé muni de la norme euclidienne notée  $\|\cdot\|$ .

Soit  $\Omega$  un ouvert de  $\mathbb{R}^n$ .

#### 1.1.1 Dérivée partielle

**Définition 1.1** Soit  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  une fonction continue. La fonction notée  $\nabla_i f(x) : \mathbb{R}^n \rightarrow \mathbb{R}$  également notée  $\frac{\partial f(x)}{\partial x_i}$  est appelée *i*<sup>ème</sup> dérivée partielle de  $f$  et est définie par :

$$\lim_{\alpha \rightarrow 0} \frac{f(x_1, \dots, x_i + \alpha, \dots, x_n) - f(x_1, \dots, x_i, \dots, x_n)}{\alpha}$$

Cette limite peut ne pas exister.

#### 1.1.2 Gradient

**Définition 1.2** soit  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  une fonction continue,  $\nabla f(x) : \mathbb{R}^n \rightarrow \mathbb{R}^n$  est appelé le gradient de  $f$  est définie par

$$\nabla f(x) = \begin{pmatrix} \frac{\partial f(x)}{\partial x_1} \\ \cdot \\ \cdot \\ \cdot \\ \frac{\partial f(x)}{\partial x_n} \end{pmatrix}$$

le gradient de  $f$  au point  $x = (x_1 \dots x_n)$

Le gradient jouera un rôle essentiel dans le développement et l'analyse des algorithmes d'optimisation.

### 1.1.3 Matrice Hessienne

**Définition 1.3** soit  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  fonction deux fois différentiable., la fonction notée  $\nabla^2 f(x) : \mathbb{R}^n \rightarrow \mathbb{R}^{n \times n}$  est appelé matrice hessienne symétrique de  $f$  définie par

$$H(x) = \nabla^2 f(x) = \left( \frac{\partial^2 f}{\partial x_i \partial x_j} \right) (x), i = 1, \dots, n; j = 1, \dots, n$$

alors

$$\begin{bmatrix} \frac{\partial^2 f(x)}{\partial x_1^2} & \frac{\partial^2 f(x)}{\partial x_1 \partial x_2} & \cdot & \cdot & \cdot & \frac{\partial^2 f(x)}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f(x)}{\partial x_2 \partial x_1} & \frac{\partial^2 f(x)}{\partial x_2^2} & \cdot & \cdot & \cdot & \frac{\partial^2 f(x)}{\partial x_2 \partial x_n} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \frac{\partial^2 f(x)}{\partial x_n \partial x_1} & \frac{\partial^2 f(x)}{\partial x_n \partial x_2} & \cdot & \cdot & \cdot & \frac{\partial^2 f(x)}{\partial x_n^2} \end{bmatrix}$$

### 1.1.4 Dérivée directionnelle

**Définition 1.4** soit  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  une fonction continue ,soit  $x \in \mathbb{R}^n$  et  $d \in \mathbb{R}^n$  , la dérivée directionnelle de  $f$  en  $x$  dans la direction  $d$  est donné par

$$\lim_{\alpha \rightarrow 0} \frac{f(x+\alpha d) - f(x)}{\alpha}$$

si la direction existe ,lorsque le gradient existe la dérivée directionnelle est le produit scalaire entre le gradient de  $f$  et la direction  $d$  c-a-d

$$\nabla f(x)^T \cdot d$$

### 1.1.5 Fonction différentiable

**Définition 1.5** Soit  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  une fonction continue. Si, pour tout  $d \in \mathbb{R}^n$  la dérivée directionnelle de  $f$  dans la direction  $d$  existe, alors la fonction  $f$  est dite différentiable.

## 1.2 Définitions

Soit  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  qui à tout  $x \in \mathbb{R}^n$ ,  $x = (x_1, x_2, \dots, x_n)^t$ , associe la valeur réelle

$$f(x) = f(x_1, x_2, \dots, x_n) .$$

On cherche à résoudre le problème (P) :

$$(P) \quad \min \{f(x) : x \in \mathbb{R}^n\} . \tag{1.1}$$

Il s'agit donc de déterminer un point  $\hat{x}$  de  $\mathbb{R}^n$  tel que :

**Définition 1.6** 1.  $\hat{x} \in \mathbb{R}^n$  s'appelle solution minimum global de (P) si et seulement si

$$f(\hat{x}) \leq f(x) : \forall x \in \mathbb{R}^n$$

$f(\hat{x})$  s'appelle valeur minimum global .

2.  $\hat{x} \in \mathbb{R}^n$  s'appelle solution minimum local de (P) si et seulement s'il existe un voisinage  $V_\varepsilon(\hat{x})$  tel que

$$f(\hat{x}) \leq f(x) : \forall x \in V_\varepsilon(\hat{x})$$

$f(\hat{x})$  s'appelle valeur minimum local

3.  $\hat{x} \in \mathbb{R}^n$  s'appelle solution minimum local strict de (P) si et seulement s'il existe un voisinage  $V_\varepsilon(\hat{x})$  tel que

$$f(\hat{x}) < f(x) : \forall x \in V_\varepsilon(\hat{x}), x \neq \hat{x}$$

$f(\hat{x})$  s'appelle valeur minimum local strict.

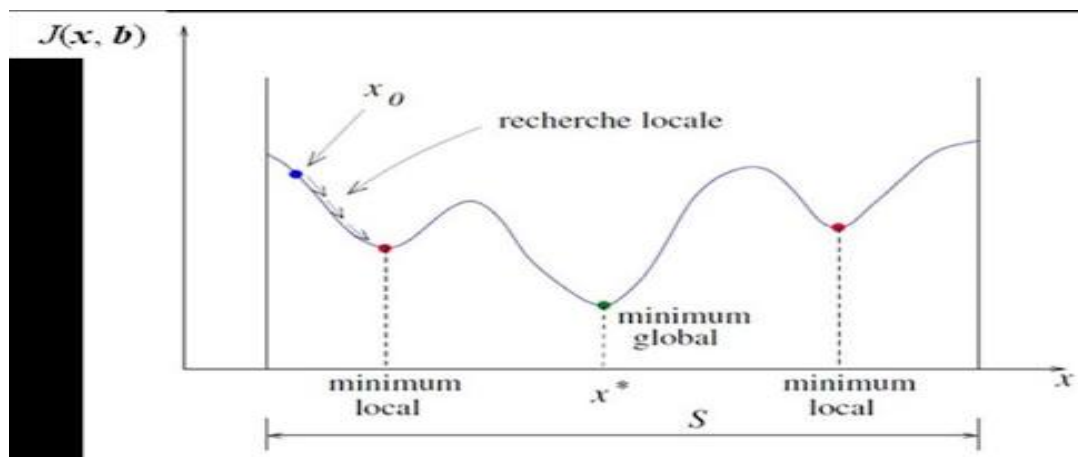


Fig.1 optimum local, optimum global

### 1.3 Direction de descente

**Définition 1.7** Soit  $f : \mathbb{R}^n \rightarrow \mathbb{R}$ ,  $\hat{x} \in \mathbb{R}^n$ ,  $d \in \mathbb{R}^n$  est dite direction de descente au point  $\hat{x}$  si et seulement si il existe un nombre strictement positive ( $\delta > 0$ ) tel que :

$$f(\hat{x} + \lambda d) < f(\hat{x}) \quad : \forall \lambda \in ]0, \delta[.$$

Donnons une condition suffisante pour que  $d$  soit une direction de descente.

**Théorème 1.1** Soit  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  différentiable au point  $\hat{x} \in \mathbb{R}^n$  et  $d \in \mathbb{R}^n$  une direction vérifiant la condition suivante :

$$f'(\hat{x}, d) = \nabla f(\hat{x})^t \cdot d < 0$$

alors  $d$  est une direction de descente au point  $\hat{x}$ .

**Preuve.**  $f$  est différentiable au point  $\hat{x}$  alors  $f$  continue et  $\nabla f(\hat{x})$  existe, donc

$$f(\hat{x} + \lambda d) = f(\hat{x}) + \lambda \nabla f(\hat{x})^t \cdot d + \lambda \|d\| \alpha(\hat{x}, \lambda d)$$

alors

$$\begin{aligned} f(\hat{x} + \lambda d) - f(\hat{x}) &= \lambda \nabla f(\hat{x})^t \cdot d + \lambda \|d\| \alpha(\hat{x}, \lambda d) \\ \Rightarrow \frac{f(\hat{x} + \lambda d) - f(\hat{x})}{\lambda} &= \nabla f(\hat{x})^t \cdot d + \|d\| \alpha(\hat{x}, \lambda d) \\ \Rightarrow \lim_{\lambda \rightarrow 0} \frac{f(\hat{x} + \lambda d) - f(\hat{x})}{\lambda} &= \lim_{\lambda \rightarrow 0} (\nabla f(\hat{x})^t \cdot d + \|d\| \alpha(\hat{x}, \lambda d)) \end{aligned}$$

avec

$$\alpha(\hat{x}, \lambda d) \xrightarrow{\lambda \rightarrow 0} 0$$

donc

$$f'(\hat{x}, d) = \lim_{\lambda \rightarrow 0} \frac{f(\hat{x} + \lambda d) - f(\hat{x})}{\lambda} = \nabla f(\hat{x})^t \cdot d < 0$$

la limite étant strictement négative, alors il existe un voisinage de zéro  $V(0) = ]-\delta, +\delta[$  tel que

$$\frac{f(\hat{x} + \lambda d) - f(\hat{x})}{\lambda} < 0, \forall \lambda \in ]-\delta, +\delta[ \quad (1.2)$$

la relation (1.2) est particulièrement vraie pour tout  $\lambda \in ]0, +\delta[$ . on obtient le résultat cherché en multipliant la relation (1.2) par  $\lambda > 0$ . ■

## 1.4 Schémas général des algorithmes

Soit  $d_k$  une direction de descente aux point  $x_k$  on peut considérer le point  $x_{k+1}$  le successeur de  $x_k$  comme suit :

$$x_{k+1} = x_k + \lambda_k d_k, \lambda_k \in ]0, +\delta[.$$

**Démarrage :**  $x_0 \in \mathbb{R}^n, d_0 :$

$$\nabla f(x_0)^t \cdot d_0 < 0$$

$$x_1 = x_0 + \lambda_0 d_0$$

$\lambda_0$  vérifie :

$$f(x_0 + \lambda_0 d_0) < f(x_0)$$

**Itération k** :  $x_k, d_k$  telle que  $\nabla f(x_k)^t \cdot d_k < 0$  et  $\lambda_k$  telle que :

$$f(x_k + \lambda_k d_k) < f(x_k)$$

alors

$$x_{k+1} = x_k + \lambda_k d_k$$

Le bon choix de  $d_k$ , et  $\lambda_k$  permet de construire une multitude d'algorithmes d'optimisation.

### 1.4.1 Exemple de choix de directions de descente

- Si on choisit :  $d_k = -\nabla f(x_k)$ , avec  $\nabla f(\hat{x}_k) \neq 0$ , on obtient la méthode de gradient.

Bien sur  $d_k = -\nabla f(x_k)$  est une direction de descente, en effet :

$$\nabla f(x_k)^t d_k = \nabla f(x_k)^t (-\nabla f(x_k)) = -\nabla f(x_k)^t \cdot \nabla f(x_k) = -\|\nabla f(x_k)^t\|^2 < 0$$

- Aussi si on choisit :  $d_k = -(H(x_k))^{-1} \nabla f(x_k)$  tel que :

$H(x_k)$  la matrice Hessienne. ( $H(x_k) \in \mathcal{M}_{n \times n}$ ),  $\nabla f(x_k)$  le vecteur de gradient. ( $\nabla f(x_k) \in \mathcal{M}_{n \times 1}$ ), on obtient la méthode de Newton.

si la matrice  $H(x_k)$  est définie positive, alors

$$\nabla f(x_k)^t d_k = -\nabla f(x_k)^t (H(x_k))^{-1} \nabla f(x_k) < 0$$

### 1.4.2 Exemple de choix de pas $\lambda_k$

On choisit  $\lambda_k$  vérifier

$$f(x_k + \lambda_k d_k) \leq f(x_k + \lambda d_k), \quad \forall \lambda \in ]0, \delta[$$

la recherche d'une variable réelle  $\lambda_k$  qui s'appelle la recherche linéaire.

#### Notion de convergence globale

**Définition 1.8** Soit  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  différentiable. Supposons que l'on a construit une suite  $\{x_k\}$ , à l'aide d'un algorithme d'optimisation sans contraintes décrit dans le modèle (modèle algorithme). Nous dirons que l'algorithme converge globalement si on a :

$$\liminf_{k \rightarrow \infty} \|\nabla f(x_k)\| = 0$$

**Remarque 1.1** Des auteurs exigent parfois pour la même définition la relation plus forte suivante :

$$\lim_{k \rightarrow \infty} \|\nabla f(x_k)\| = 0$$

### Notion de vitesse de convergence

La convergence globale d'un algorithme ayant été établie, nous nous intéressons maintenant à l'évaluation de son *efficacité*. D'un point de vue pratique, l'efficacité d'un algorithme dépend du nombre d'itérations nécessaires pour obtenir une approximation à  $\varepsilon$  près ( $\varepsilon$  fixé à l'avance) de l'optimum  $x^*$ .

Si l'on compare entre eux, plusieurs algorithmes, et si l'on admet que le temps de calcul par itération est sensiblement le même pour tous, le meilleur est celui qui nécessitera le plus petit nombre d'itérations.

Malheureusement, il se révèle impossible de dégager des conclusions générales de ce genre de comparaison.

Suivant le point de départ choisi, la nature de la fonction à optimiser, la valeur de la tolérance choisie, la hiérarchie des algorithmes peut varier considérablement.

Si l'on veut dégager un critère ayant une certaine valeur d'absolu, il faut par conséquent recourir à un autre type d'analyse : c'est l'objet de l'étude de la *convergence asymptotique*, c'est-à-dire du comportement de la suite  $\{x_k\}$  au voisinage du point limite  $x^*$ .

Ceci conduit à attribuer à chaque algorithme un indice d'efficacité appelé sa *vitesse de convergence*.

**Remarque 1.2** on est quelque fois amené à exprimer la vitesse de convergence de la suite  $\{x_k\}$  en étudiant, non pas la façon dont  $\|x_k - x^*\|$  tend vers 0, mais la façon dont la suite  $\{f(x_k)\}$  tend vers  $f(x^*)$  où  $f$  la fonction que l'on minimise.

### Les modes de convergence

**Définition 1.9** Soit  $\{x_k\}_{k \in \mathbb{N}}$  une suite dans  $\mathbb{R}^n$  convergeant vers  $x^*$

1° si

$$\limsup_{k \rightarrow \infty} \frac{\|x_{k+1} - x^*\|}{\|x_k - x^*\|} = \alpha < 1,$$

on dit que la convergence est linéaire et  $\alpha$  est le taux de convergence associé.

2° si

$$\limsup_{k \rightarrow \infty} \frac{\|x_{k+1} - x^*\|}{\|x_k - x^*\|^\gamma} < +\infty; \quad \gamma > 1,$$

On dit que la convergence est superlinéaire d'ordre  $\gamma$ .

3° si

$$\lim_{k \rightarrow \infty} \frac{\|z_{k+1} - z^*\|}{\|z_k - z^*\|} = 0,$$

On dit que  $\{z_k\}$  tend vers  $z^*$  de façon superlinéaire.

## 1.5 Condition nécessaire d'optimalité

### 1.5.1 Condition nécessaire d'optimalité du premier ordre

**Théorème** Soit  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  différentiable au point  $z^* \in \mathbb{R}^n$ . Si  $z^*$  est un minimum local de  $(P)$  alors  $\nabla f(z^*) = 0$ .

**Preuve.** Supposons le contraire c. a. d que  $\nabla f(z^*) \neq 0$  ce qui implique que  $d = -\nabla f(z^*)$  est une direction de descente. Donc  $\exists \delta > 0$  tel que

$$\forall \lambda \in ]0, \delta[ : f(z^* + \lambda d) < f(z^*)$$

Ceci est contradiction avec le fait que  $z^*$  est une solution optimale locale de  $(P)$ . ■

### 1.5.2 Condition nécessaire d'optimalité du second ordre

**Théorème 1.2** Soit  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  deux fois différentiable au point  $\hat{x} \in \mathbb{R}^n$ , si  $\hat{x}$  est un minimum local de  $(P)$  alors  $\nabla f(\hat{x}) = 0$  et la matrice hessienne de  $f$  au point  $\hat{x}$ , qu'on note  $H(\hat{x})$ , est semi définie positive.

**Preuve.** Soit  $x \in \mathbb{R}^n$  quelconque,  $f$  étant deux fois différentiable au point  $\hat{x}$  on aura pour tout  $\lambda \neq 0$

$$f(\hat{x} + \lambda x) = f(\hat{x}) + \frac{1}{2} \lambda^2 x^t H(\hat{x}) x + \lambda^2 \|x\|^2 \alpha(\hat{x}, \lambda x), \quad \alpha(\hat{x}, \lambda x) \xrightarrow{\lambda \rightarrow 0} 0$$

Ceci implique

$$\frac{f(\hat{x} + \lambda x) - f(\hat{x})}{\lambda^2} = \frac{1}{2} x^t H(\hat{x}) x + \|x\|^2 \alpha(\hat{x}, \lambda x) \quad (1.3)$$

$\hat{x}$  est un optimum local, il existe alors  $\delta > 0$  tel que

$$\frac{f(\hat{x} + \lambda x) - f(\hat{x})}{\lambda^2} \geq 0, \quad \forall \lambda \in ]-\delta, +\delta[$$

si on prend en considération (1.3) et on passe à la limite quand  $\lambda \rightarrow 0, \lambda \neq 0$ , on obtient

$$x^t H(\hat{x}) x \geq 0, \quad \forall x \in \mathbb{R}^n.$$

■



## 1.6 Condition suffisante d'optimalité

**Théorème 1.3** soit  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  deux fois différentiable au point  $\hat{x} \in \mathbb{R}^n$ , Si  $\nabla f(\hat{x}) = 0$  et  $H(\hat{x})$  est définie positive alors  $\hat{x}$  est un minimum local strict de  $(P)$

**Preuve.**  $f$  étant deux fois différentiable au point  $\hat{x}$ , on aura pour tout  $x \in \mathbb{R}^n$

$$f(x) = f(\hat{x}) + \frac{1}{2}(x - \hat{x})^t H(\hat{x})(x - \hat{x}) + \|(x - \hat{x})\|^2 \alpha(\hat{x}, (x - \hat{x})), \alpha(\hat{x}, (x - \hat{x})) \xrightarrow{x \rightarrow \hat{x}} 0, (\nabla f(\hat{x}) = 0) \quad (1.4)$$

supposons que  $\hat{x}$  n'est pas un optimum local strict .

Alors il existe une suite  $\{x_k\}_{k \in \mathbb{N}^*}$  telle que  $x_k \neq \hat{x} : \forall k$  et

$$x_k \neq \hat{x} : \forall k, x_k \xrightarrow{k \rightarrow \infty} \hat{x} \text{ et } f(x_k) \leq f(\hat{x}). \quad (1.5)$$

Dans (1.4) prenons  $x = x_k$ , division le tout par  $\|(x - \hat{x})\|^2$  et notons  $d_k = \frac{(x_k - \hat{x})}{\|(x_k - \hat{x})\|}$ , on obtient

$$\frac{f(x_k) - f(\hat{x})}{\|(x_k - \hat{x})\|^2} = \frac{1}{2} d_k^t H(\hat{x}) d_k + \alpha(\hat{x}, (x_k - \hat{x})), \alpha(\hat{x}, (x_k - \hat{x})) \xrightarrow{k \rightarrow \infty} 0. \quad (1.6)$$

(1.5) et (1.6) impliquent

$$\frac{1}{2} d_k^t H(\hat{x}) d_k + \alpha(\hat{x}, (x_k - \hat{x})) \leq 0, \quad \forall k.$$

d'autre part la suite  $\{d_k\}_{k \in \mathbb{N}^*}$  est bornée ( $\|d_k\| = 1, \forall n$ ). Donc il existe une sous suite  $\{d_k\}_{k \in N_1 \subset \mathbb{N}}$  telle que

$$d_k \xrightarrow{k \rightarrow \infty, k \in N_1} \bar{d}.$$

Finalement lorsque  $k \rightarrow \infty, k \in N_1$ , on obtient

$$\frac{1}{2} \bar{d}^t H(\hat{x}) \bar{d} \leq 0.$$

La dernière relation et le fait que  $\bar{d} \neq 0$  ( $\|\bar{d}\| = 1$ ) impliquent que la matrice hessienne  $H(\hat{x})$  n'est pas définie positive. Ceci est en contradiction avec l'hypothèse. ■

## 1.7 Conditions nécessaires et suffisantes d'optimalité dans le cas convexe

**Définition 1.10** soit  $f : \mathbb{R}^n \rightarrow \mathbb{R}$ .

a)  $f$  est dite convexe dans  $\mathbb{R}^n$  si :

$$f(t\alpha_1 + (1-t)\alpha_2) \leq tf(\alpha_1) + (1-t)f(\alpha_2)$$

pour tous points  $x_1$  et  $x_2 \in \mathbb{R}^n$  et pour tout  $t \in [0, 1]$ .

b) si l'inégalité précédente est stricte pour tous points  $x_1$  et  $x_2$  distincts et pour tout  $t \in ]0, 1[$ , alors  $f$  est dite strictement convexe.

c) supposons que  $f$  soit différentiable,  $f$  est dite fortement convexe si :

$$f((1-t)x_1 + tx_2) \leq (1-t)f(x_1) + tf(x_2) - \frac{\alpha}{2}t(1-t)\|x_1 - x_2\|^2$$

pour tous points  $x_1$  et  $x_2 \in \mathbb{R}^n$  et pour tout  $t \in [0, 1]$

# Chapitre 2

## Optimisation unidimensionnelle

### 2.1 Recherches linéaires

La recherche linéaire consiste à trouver  $\lambda_k$  de façon à diminuer la fonction  $f$  *Suffisamment* le long de cette direction.

Ce " *suffisamment* " sera quantifié dans la suite dans la description des conditions dites d' Armijo, Wolfe, Goldstein & Price(recherches linéaires inexactes).

#### 2.1.1 Principe des méthodes de descente

Le principe d'une méthode de descente consiste à faire les itérations suivantes :

$$x_{k+1} = x_k + \lambda_k d_k, \quad k > 0 \quad (2.1)$$

tout en assurant la propriété

$$f(x_{k+1}) < f(x_k) .$$

Le vecteur  $d_k$  est la direction de descente en  $x_k$  . Le scalaire  $\lambda_k$  est appelé le pas de la méthode à l'itération  $k$ .

On peut caractériser les directions de descente en  $x_k$  à l'aide du gradient :

**Proposition 2.1** Soit  $d \in \mathbb{R}^n$  vérifiant

$$\nabla f(x)^t . d < 0$$

alors  $d$  est une direction de descente en  $x$ .

**Preuve.** on a pour  $\lambda > 0$

$$f(x + \lambda d) = f(x) + \lambda \nabla f(x)^t d + \lambda \varepsilon(\lambda)$$

donc si on écrit

$$\frac{f(x + \lambda d) - f(x)}{\lambda} = \nabla f(x)^t d + \varepsilon(\lambda)$$

on voit bien que pour  $\lambda$  suffisamment petit on aura

$$f(x + \lambda d) - f(x) < 0.$$

■

**Remarque 2.1**  $d$  fait avec l'opposé du gradient  $-\nabla f(x)$  un angle strictement plus petit que  $90^\circ$  :

$$\theta := \arccos \frac{-\nabla f(x)^t d}{\|\nabla f(x)\| \|d\|} \in \left] 0, \frac{\pi}{2} \right[$$

L'ensemble des directions de descente de  $f$  en  $x$

$$\{d \in \mathbb{R}^n : \nabla f(x)^t d < 0\}$$

forme un demi-espace ouvert de  $\mathbb{R}^n$

Pour définir une direction de descente il faut donc spécifier deux choses :

- \* choisir  $d_k$  est dire comment la direction est calculée . Ce choix influe directement dans la nomination de l'algorithme.
- \* détermine le pas  $\lambda_k$ ,

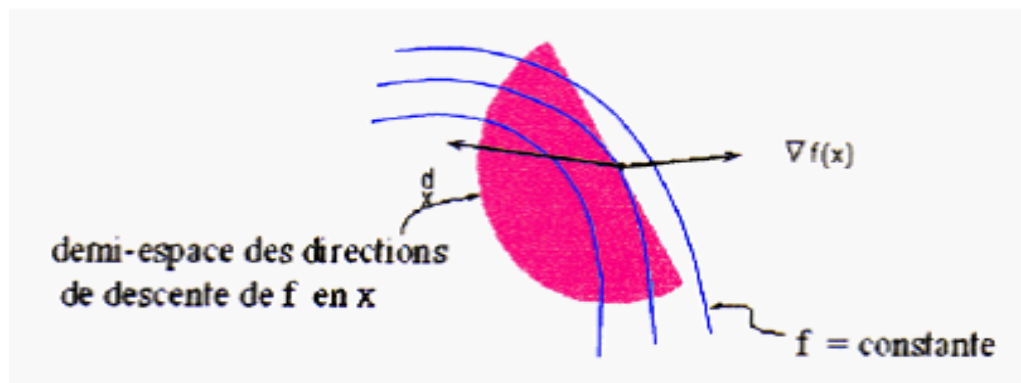


Fig.2 Demi-espace (translaté) des directions de descente  $d$  de  $f$  en  $x$

**Algorithme (méthode à directions de descente – une itération)**

**Etape 0 : (initialisation)**

On suppose qu'au début de l'itération  $k$ , on dispose d'un itéré

$$x_k \in \mathbb{R}^n$$

**Etape 1 :**

Test d'arrêt : si  $\|\nabla f(x_k)\| \simeq 0$ , d'arrêt de l'algorithme ;

**Etape 2 :**

Choix d'une direction de descente  $d_k \in \mathbb{R}^n$  ;

**Etape 3 :**

Recherche linéaire : déterminer un pas  $\lambda_k > 0$  le long de  $d_k$  de manière à "faire décroître  $f$  suffisamment" ;

**Etape 4 :**

Si la recherche linéaire est finie  $x_{k+1} = x_k + \lambda_k d_k$  ;

### 2.1.2 Recherche linéaire

Considérons le problème d'optimisation sans contraintes ( $P$ )

$$(p) \text{ minimiser } f(x) \quad x \in \mathbb{R}^n$$

où  $f : \mathbb{R}^n \rightarrow \mathbb{R}$

Les algorithmes qu'on étudie par la suite suivent les schémas généraux suivants :

$$x_{k+1} = x_k + \lambda_k d_k$$

où  $\lambda_k$  est solution optimale du problème d'optimisation unidimensionnel suivant :

$$\min_{\lambda > 0} f(x_k + \lambda d_k)$$

$x_k$  ;  $d_k$  sont fixé et la fonction à minimiser est une fonction d'une variable réelle définie comme suit :

$$\lambda \rightarrow \varphi(\lambda) = f(x_k + \lambda d_k)$$

Il faut noter que dans les problèmes d'optimisation sans contraintes on a besoin de résoudre à chaque itération  $x_k$ , un problème d'optimisation dans  $\mathbb{R}$ .

Dans ce chapitre nous allons décrire les différentes manières de déterminer un pas  $\lambda_k > 0$  le long d'une direction de descente  $d_k$ . C'est ce que l'on appelle faire de la recherche linéaire.

## Objectifs de la recherche linéaire

Il s'agit de réaliser deux objectifs :

### Le premier objectif

Faire décroître  $f$  suffisamment, cela se traduit le plus souvent par la réalisation d'une inégalité de la forme

$$f(x_k + \lambda_k d_k) \leq f(x_k) + \text{"un terme négatif"} \quad (2.2)$$

Le terme négatif, disons  $\nu_k$ , joue un rôle-clé dans la convergence de l'algorithme utilisant cette recherche linéaire.

L'argument est le suivant .

Si  $f(x_k)$  est minorée ( $\exists c$  telle que  $f(x_k) \geq c$  pour tout  $k$ ), alors  $\nu_k$  tend nécessairement vers zéro ( $\nu_k \rightarrow 0$ ). c'est souvent à partir de la convergence vers zéro de cette suite que l'on parvient à montrer que le gradient lui-même doit tendre vers zéro. Le terme négatif devra prendre une forme bien particulière si on veut pouvoir en tirer de l'information .

En particulier, il ne suffit pas d'imposer  $f(x_k + \lambda_k d_k) < f(x_k)$ .

### Le second objectif

Empêcher le pas  $\lambda_k > 0$  d'être trop petit, trop proche de zéro.

Le premier objectif n'est en effet pas suffisant car l'inégalité (2.3) est en général satisfaite par des pas  $\lambda_k > 0$  arbitrairement petit .

Or ceci peut entraîner une "fausse convergence", c'est-à-dire la convergence des itérés vers un point non stationnaire.

## Types de recherches linéaires

Il existe deux grandes classes de méthodes qui s'intéressent à l'optimisation unidimensionnelle :

- les recherches linéaires exactes.
- les recherches linéaires inexactes.

### 2.1.3 Les recherches linéaires exactes

Dans ce cas la solution optimale  $\lambda_k$  est calculée de façon exacte (d'un point de vue théorique car pratiquement on n'obtient en général qu'une approximation).

#### L'intervalle d'incertitude

**Définition 2.1** *Considérons le problème unidimensionnel suivant :*

$$\underset{\lambda \in [a, b]}{\text{Minimiser}} \varphi(\lambda) . \quad (2.3)$$

L'intervalle  $[a, b]$  est dit *intervalle d'incertitude* si le minimum  $\bar{\lambda}$  de  $\varphi(\lambda)$  appartient à  $[a, b]$ , mais sa valeur exacte n'est pas connue .

**Théorème 2.1** *soit  $\varphi : \mathbb{R} \rightarrow \mathbb{R}$  strictement quasi-convexe sur  $[a, b]$  .*

*soient  $\lambda, \mu \in ]a, b[$ ,  $\lambda < \mu$*

**1)** *si  $\varphi(\lambda) > \varphi(\mu)$ , alors  $\varphi(z) \geq \varphi(\mu); \forall z \in [a, \lambda]$  .*

**2)** *si  $\varphi(\lambda) \leq \varphi(\mu)$ , alors  $\varphi(z) \geq \varphi(\lambda); \forall z \in [\mu, b]$  .*

**Conséquences importante du théorème :**

1. Si  $\varphi(\lambda) > \varphi(\mu)$ , alors le nouveau intervalle d'incertitude est :  $[\lambda, b]$  (On supprime  $[a, \lambda[$ ).
2. Si  $\varphi(\lambda) \leq \varphi(\mu)$ , alors le nouveau intervalle d'incertitude est :  $[a, \mu]$  (On supprime  $[\mu, b[$ ).

Ceci est l'idée de base pour la construction d'algorithmes d'optimisation unidimensionnelle sans calcul de dérivées. A chaque itération on fait diminuer l'intervalle d'incertitude jusqu'à ce qu'on arrive à un intervalle final de longueur inférieure à une tolérance fixée à l'avance. Bien sur une valeur quelconque de ce dernier intervalle conviendrait comme approximation de notre solution optimale.

On a deux méthodes d'optimisation unidimensionnelle sans dérivées :

**La méthode de dichotomie**

**La méthode du nombre d'or**

## 2.1.4 Les recherches linéaires inexactes

Les recherches linéaires exactes, malgré qu'elles n'aboutissent qu'à une solution optimale approchée, elle nécessitent beaucoup d'observations à chaque itération de l'algorithme principal Pour cela de nombreux mathématiciens (Armijo, Goldstein, Wolfe, Albaali, Lemaréchal, Fletcher...) ont élaboré plusieurs règles.

l'objectif de cette section consiste à présenter les principales tests.

D'abord présentons La règle d'Armijo, règle de Goldstein et La règle de Wolfe

### La recherche linéaire inexacte d'Armijo

Soit  $f : \mathbb{R}^n \rightarrow \mathbb{R}$ ,  $x \in \mathbb{R}^n$ ,  $d \in \mathbb{R}^n$  tel que

$$\nabla f^T(x_k) d_k < 0 \quad (2.4)$$

Définissons la fonction  $\varphi : \mathbb{R} \rightarrow \mathbb{R}$  avec

$$\varphi(\lambda_k) = f(x_k + \lambda_k d_k) \quad \lambda_k \geq 0$$

Notons que :

$$\begin{aligned} \varphi'(\lambda) &= \nabla f^T(x_k + \lambda_k d_k) d_k, \\ \varphi'(0) &= \nabla f^T(x_k) d_k < 0, \\ \varphi(0) &= f(x_k) \end{aligned}$$

L'équation de la tangente au point  $(0, \varphi(0))$  est la suivante

$$\begin{aligned} \{\lambda, y\} &: y = \varphi(0) + \varphi'(0)(\lambda - 0). \\ \varphi_1(\lambda_k) &= f(x_k) + \nabla f^T(x_k) d_k \lambda_k \end{aligned}$$

Posons

$$\varphi_1(\lambda) = \varphi(0) + \varphi'(0) \lambda.$$

L'équation de la tangente devient :

$$\varphi_1(\lambda) = f(x_k) + \nabla f^T(x_k) d_k \lambda$$

Définissons maintenant la fonction  $\hat{\varphi}(\lambda)$  comme suit :

$$\hat{\varphi}(\lambda) = f(x_k) + \varepsilon \lambda \hat{\varphi}(0) = f(x_k) + \varepsilon \lambda \nabla f^T(x_k) d_k; \quad 0 < \varepsilon < 1 \quad (2.5)$$

On cherche  $\bar{\lambda}_k$  tel que

$$\varphi(\bar{\lambda}_k) \leq \hat{\varphi}(\bar{\lambda}_k)$$



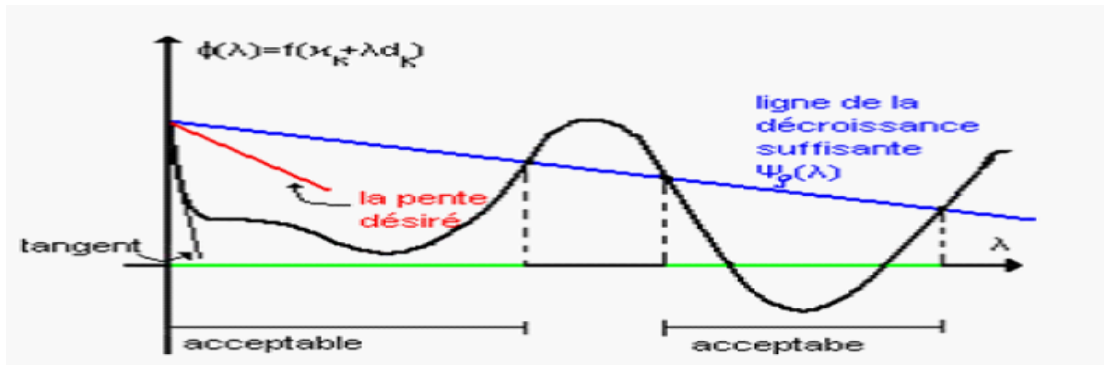


Fig 3 -Règle d'Armijo

### Remarques

1-La condition  $\varphi(\bar{\lambda}_k) \leq \hat{\varphi}(\bar{\lambda}_k)$  implique la décroissance de la fonction  $f$ .

En effet

$$\begin{aligned} \varphi(\bar{\lambda}_k) &\leq \hat{\varphi}(\bar{\lambda}_k) \\ f(x_k + \bar{\lambda}_k d_k) &\leq f(x_k) + \varepsilon \bar{\lambda}_k \nabla f(x_k) d_k < f(x_k) \end{aligned}$$

car la direction  $d$  est une direction de descente.

2-Il ne faut pas prendre  $\bar{\lambda}_k$  très proche de zéro car cela va altérer la convergence et la vitesse de convergence.

En effet

$$\begin{aligned} f(x_k + \bar{\lambda}_k d_k) &= f(x_k) + \bar{\lambda}_k \nabla f(x_k) d_k + \bar{\lambda}_k \alpha(x_k, \bar{\lambda}_k d_k) \\ f(x_{k+1}) - f(x_k) &= \bar{\lambda}_k [\nabla f(x_k) d_k + \alpha(x_k, \bar{\lambda}_k d_k)] \\ \text{si } \bar{\lambda}_k &\rightarrow 0 \quad \alpha(x_k, \bar{\lambda}_k d_k) \xrightarrow{\bar{\lambda}_k \rightarrow 0} 0 \quad \text{donc} \quad f(x_{k+1}) \simeq f(x_k) \end{aligned}$$

### Algorithme d'Armijo

- $\lambda^*$  initial
- Si  $\varphi(\lambda^*) \leq \hat{\varphi}(\lambda^*)$ , alors choisi le plus grand entier  $t \geq 0$  tel que  $\varphi(2^t \lambda^*) \leq \hat{\varphi}(2^t \lambda^*)$   
 $\bar{\lambda}_k = 2^t \lambda^*$
- Sinon choisir le plus petit entier  $t \geq 0$  tel que  $\varphi(\frac{\lambda^*}{2^t}) < \hat{\varphi}(\frac{\lambda^*}{2^t})$   
 $\bar{\lambda}_k = \frac{\lambda^*}{2^t}$

### La recherche linéaire inexacte de Goldstein

En ajoutant une deuxième inégalité  $\varphi(\lambda) \geq \varphi(0) + \sigma_2 \dot{\varphi}(0)$  à la règle d'Armijo on obtient la règle de Goldstein, avec  $\sigma_1, \sigma_2$  sont deux constantes vérifiant  $0 < \sigma_1 < \frac{1}{2} < \sigma_2 < 1$ . Les deux inégalités de la règle de Goldstein sont donc :

$$\begin{aligned}\varphi(\bar{\lambda}_k) &\leq \hat{\varphi}(\bar{\lambda}_k) \text{ et} \\ \varphi(\bar{\lambda}_k) &\geq \varphi(0) + \sigma_2 \dot{\varphi}(0).\end{aligned}$$

Soit en remplaçant

$$\begin{aligned}f(x_k + \bar{\lambda}_k d_k) &\leq f(x_k) + \sigma_1 \bar{\lambda}_k \nabla f(x_k) d_k \\ &\text{et} \\ f(x_k + \bar{\lambda}_k d_k) &\geq f(x_k) + \sigma_2 \bar{\lambda}_k \nabla f(x_k) d_k\end{aligned}$$

### La recherche linéaire inexacte de Wolfe

La règle de Wolfe fait appel au calcul de  $\varphi'(\lambda)$ , elle est donc en théorie plus coûteuse que la règle de Goldstein. Cependant dans de nombreuses applications, le calcul du gradient  $\nabla f(x)$  représente un faible coût additionnel en comparaison du coût d'évaluations de  $f(x)$ , c'est pourquoi cette règle est très utilisée.

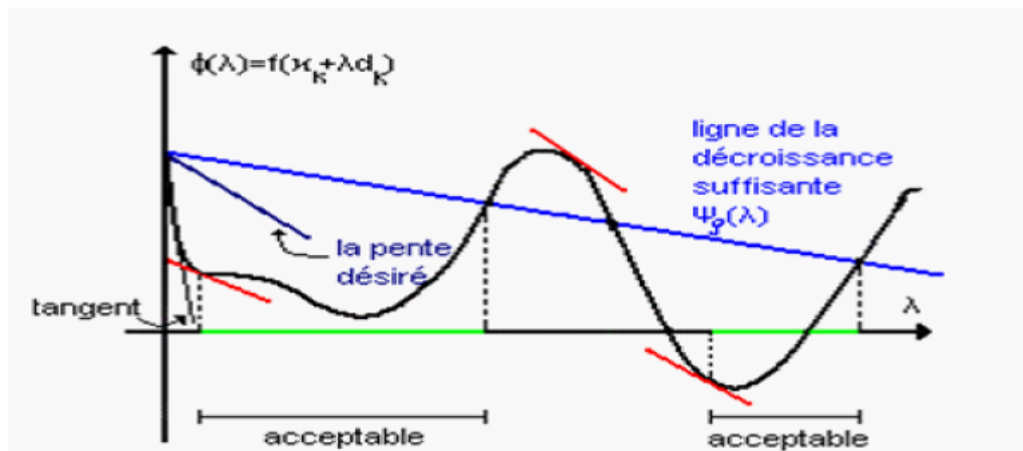


Fig 4 -Règle de Wolfe

Les deux conditions de la règle Wolfe sont donc les suivantes :

$$f(x_k + \bar{\lambda}_k d_k) \leq f(x_k) + \sigma_1 \bar{\lambda}_k \nabla f(x_k) d_k \quad (2.6)$$

et

$$\nabla f(x_k + \bar{\lambda}_k d_k)^T d_k \geq \sigma_2 \nabla f(x_k)^T d_k \quad (2.7)$$

avec

$$0 < \sigma_1 < \sigma_2 < 1.$$

Le théorème suivant est important car il nous explicite les conditions suffisantes pour l'existence des scalaires  $\bar{\lambda}_k$ .

**Théorème 2.2** *Si  $d_k$  est une direction de descente,  $f$  est continument différentiable et bornée, alors il existe toujours un scalaire  $\bar{\lambda}_k$  satisfait les conditions (2.6) et (2.7).*

## 2.1.5 Convergence des méthodes utilisant des recherches linéaires inexactes et des directions de descente. Le théorème de Zoutendijk

### La condition de Zoutendijk

Maintenant on va étudier la contribution de la recherche linéaire inexacte dans la convergence des algorithmes à directions de descente. Ce n'est qu'une contribution, parce que la recherche linéaire ne peut à elle seule assurer la convergence des itérés. On comprend bien que le choix de la direction de descente joue aussi un rôle. Cela se traduit par une condition, dite de Zoutendijk, dont on peut tirer quelques informations qualitatives intéressantes.

On dit qu'une règle de recherche linéaire inexacte satisfait la condition de Zoutendijk s'il existe une constante  $C > 0$  telle que pour tout indice  $k \geq 1$  on ait

$$f(x_{k+1}) \leq f(x_k) - C \|\nabla f(x_k)\|^2 \cos^2 \theta_k \quad (2.8)$$

où  $\theta_k$  est l'angle que fait  $d_k$  avec  $-\nabla f(x_k)$ , défini par

$$\cos \theta_k = \frac{-\nabla^T f(x_k) d_k}{\|d_k\| \|d_k\|}. \quad (2.9)$$

Voici comment on se sert de la condition de condition de Zoutendijk.

### Théorème 2.3 (de Zoutendijk)

*Si la suite  $\{x_k\}$  générée par un algorithme d'optimisation vérifie la condition de Zoutendijk (2.8) et si la suite  $\{f(x_k)\}$  est minorée, alors*

$$\sum_{k \geq 1} \|\nabla f(x_k)\|^2 \cos^2 \theta_k < \infty \quad (2.10)$$

**Preuve.** En sommant les quantités inégalités  $\|\nabla f(x_k)\|^2 \cos^2 \theta_k$  tout en prenant en considération (2.9), on a

$$\sum_{k \geq 1}^l \|\nabla f(x_k)\|^2 \cos^2 \theta_k \leq \frac{1}{C'} (f(x_1) - f(x_{l+1}))$$

La série est donc convergente puisqu'il existe une constante  $C'$  telle que pour tout  $k$ ,  $f(x_k) \geq C'$ . ■

**Proposition 2.2** Soit  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  une fonction continument différentiable dans un voisinage de  $T = \{x \in \mathbb{R}^n : f(x) \leq f(x_1)\}$ .

On considère un algorithme à directions de descente  $d_k$ , qui génère une suite  $\{x_k\}$  en utilisant la recherche linéaire d'Armijo, avec  $\alpha_1 > 0$ .

Alors il existe une constante  $C > 0$  telle que, pour tout  $k \geq 1$ , l'une des conditions

$$f(x_{k+1}) \leq f(x_k) - C \nabla^T f(x_k) d_k$$

ou

$$f(x_{k+1}) \leq f(x_k) - C \|\nabla f(x_k)\|^2 \cos^2 \theta_k$$

est vérifiée.

**Proposition 2.3** Soit  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  une fonction continument différentiable dans un voisinage de  $T = \{x \in \mathbb{R}^n : f(x) \leq f(x_1)\}$ .

On considère un algorithme à directions de descente  $d_k$ , qui génère une suite  $\{x_k\}$  en utilisant la recherche linéaire de Wolfe (Wolfe1) et (Wolfe2).

Alors il existe une constante  $C > 0$  telle que, pour tout  $k \geq 1$ , la condition de Zoutendijk (2.9) est vérifiée.

### Conséquence importante du théorème de Zoutendijk

La condition (2.10) implique

$$\|\nabla f(x_k)\|^2 \cos^2 \theta_k \rightarrow 0 \quad (k \rightarrow \infty) \quad (2.11)$$

Cette limite peut être utilisée pour en déduire la convergence de l'algorithme.

En effet si notre algorithme génère une suite  $\{x_k\}$  de la forme :

$$x_{k+1} = x_k + \lambda_k d_k.$$

Si le choix de  $d_k$  est tel que

$$\cos \theta_k \geq \delta > 0, \quad \forall k$$

alors il découle de (2.11) que

$$\lim \|\nabla f(x_k)\| = 0 \quad (2.12)$$

# Chapitre 3

## Méthodes quasi-Newtoniennes

### 3.1 Méthode de Newton

Considérons le problème d'optimisation sans contraintes  $(P)$

$$(P) : \underset{x \in \mathbb{R}^n}{\text{Min}} f(x)$$

où  $f : \mathbb{R}^n \rightarrow \mathbb{R}$

Le principe de la méthode de Newton consiste à minimiser successivement les approximations du second ordre de  $f$ , plus précisément si

$$f(x) = f(x_k) + \nabla f(x_k)^t (x - x_k) + (x - x_k)^t H(x_k) (x - x_k) + o\|x - x_k\|^2,$$

posons

$$q(x) = f(x_k) + \nabla f(x_k)^t (x - x_k) + (x - x_k)^t H(x_k) (x - x_k).$$

Soit  $x_{k+1}$  l'optimum de  $q$ , alors il vérifie  $\nabla q(x_{k+1}) = 0$ , soit en remplaçant :

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

ou encore

$$H(x_k) (x_{k+1} - x_k) = -\nabla f(x_k)$$

donc

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

Donc pour obtenir une méthode qui converge superlinéairement, il est nécessaire d'approximer l'étape de Newton asymptotiquement. C'est le principe de Dennis et Moré. Comment peut-on-y aboutir sans évaluer la matrice Hessienne dans chaque itération ?

La réponse a été découverte par Davidon en 1959 ([33]) et a été développée et popularisée par Fletcher et Powell en 1963 ([31]). Elle consiste à commencer par n'importe quelle approximation de la matrice Hessienne et à chaque itération, on améliore la matrice en introduisant la courbure du problème mesuré tous au long de l'étape. Si cette amélioration est faite correctement, on obtient quelques méthodes remarquablement robustes et efficaces, qu'on appelle les méthodes de la variable métrique ou quasi Newton. Ils ont libéré l'optimisation non linéaire en procurant une alternative à la méthode de Newton, qui est très coûteuse pour plusieurs applications.

Il y a plusieurs méthodes de variable métrique, on s'étalera particulièrement sur les trois plus importantes, la méthode de correction de rang un, la méthode DFP (Davidon, Fletcher, Powell), et la méthode BFGS (Broyden, Fletcher, Goldfarb, Shanno).

**Algorithme (méthode de Newton)**

**Étape initiale :**

Soit  $\varepsilon > 0$ , critère d'arrêt. Choisir  $\mathcal{x}_1$  point initial, poser  $k = 1$  et aller à l'étape principale.

**Étape principale :**

Si  $\|\nabla f(\mathcal{x}_k)\| \leq \varepsilon$  STOP, sinon poser  $\mathcal{x}_{k+1} = \mathcal{x}_k - [H(\mathcal{x}_k)]^{-1} \nabla f(\mathcal{x}_k)$  remplacer  $k$  par  $k + 1$  et aller à l'étape principale.

### 3.1.1 Avantages et inconvénients

#### ♣ Avantages

- Si le point  $\mathcal{x}_0$  est assez proche de la solution optimale locale  $\mathcal{x}^*$  telle que  $H(\mathcal{x}^*)$  soit définie positive, alors l'algorithme de Newton converge de façon quadratique vers la solution  $\mathcal{x}^*$  c'est-à-dire que l'on a

$$\|\mathcal{x}_{k+1} - \mathcal{x}^*\| \leq \gamma \|\mathcal{x}_k - \mathcal{x}^*\|^2 \quad \gamma \geq 0$$

- La méthode de Newton fonctionne bien si  $1 < n < 10$  car il est facile de déterminer  $\nabla^2 f(\mathcal{x}_k)$ .
- La méthode de Newton converge en une seule itération lorsqu'elle est appliquée à une fonction quadratique strictement convexe.

### ♣ Inconvénients

- Il faut calculer les dérivées secondes de  $f$  qui peut être très coûteux en temps de calcul
- L'algorithme n'est pas globalement convergent si le premier itéré  $x_0$  est éloigné de  $x^*$ .
- Le successeur  $x_{k+1}$  de  $x_k$  n'est pas toujours bien défini, si  $\nabla^2 f(x) = H(x)$  est singulière.
- Si  $f$  n'est pas strictement convexe, l'algorithme ne génère pas nécessairement des directions de descente de  $f$ .
- Un système linéaire d'ordre  $n$  doit être résolu à chaque itération, et on ne dispose pas de place mémoire pour stocker les  $\mathcal{O}(n^2)$  éléments d'une matrice.

## 3.2 Méthodes de Quasi-Newton

Le couplage de la méthode de Newton avec la recherche linéaire de Wolfe a permis de construire une méthode globalement convergente. Les méthodes de quasi-Newton ont été développées pour pallier d'autres inconvénients de la méthode de Newton : en particulier

le problème du calcul de la matrice hessienne qui n'est pas toujours possible ou conseillé. Ces méthodes se concentrent donc sur la construction itérative de matrices  $H_k$  approchant la hessienne, ou de matrices  $B_k$  approchant l'inverse de la hessienne

Une méthode de quasi Newton est une méthode de type :

$$\begin{cases} x_{k+1} = x_k + \lambda_k d_k \\ d_k = -M_k g_k \end{cases} \quad (3.1)$$

ou bien :

$$\begin{cases} x_{k+1} = x_k + \lambda_k d_k \\ d_k = -S_k^{-1} g_k \end{cases} \quad (3.2)$$

ou  $M_k$  (respectivement  $S_k$ ) est une matrice destinée à approcher l'inverse du Hessien de  $f$  (respectivement le Hessien) de  $f$  en  $x_k$ . Le problème posé est : quelle stratégie à adopter pour faire cette approximation ? On peut par exemple poser  $M_1 = I$ , mais comment ensuite mettre à jour l'approximation  $M_k$  au cours des itérations ?

L'idée est la suivante :

Prenons  $f \in \mathcal{C}^2(\mathbb{R}^n)$  et faisons un développement de  $\nabla f(x)$  au voisinage de  $x_k$ .

$$\begin{aligned} \nabla f(x) &= \nabla f(x_k) + H(x_k)(x - x_k) + o(\|x - x_k\|) \\ &= \nabla f(x_k) + H(x_k)(x - x_k) \end{aligned}$$

ce qui implique

$$H(x_k)^{-1} = [\nabla f(x) - \nabla f(x_k)] (x - x_k)$$

Les approximations sont exacts si  $f$  est quadratique. En particulier avec  $x = x_{k+1}$  et si  $M_k$  était une bonne approximation de  $[H(x_k)]^{-1}$  alors

$$M_k [g(x_{k+1}) - g(x_k)] = x_{k+1} - x_k$$

On peut imposer que  $M_{k+1}$  satisfait cette équation exactement d'où :

$$M_{k+1} [g(x_{k+1}) - g(x_k)] = x_{k+1} - x_k$$

### 3.2.1 Quelques propriétés des algorithmes de quasi-Newton

- Des méthodes quasi-newtoniennes, comme la plupart des autres méthodes, sont développées pour le problème quadratique convexe et puis sont prolongées au problème général. Elles sont considérées comme étant parmi les méthodes les plus efficaces et sont employées très intensivement dans de nombreuses applications. Plusieurs méthodes quasi-newtoniennes distinctes ont évolué ces dernières années. Les trois méthodes les plus importantes de cette classe sont :

1- *Méthode de correction de rang un (SR1 [1960])*

2- *Méthode de Davidon-Fletcher-Powell (DFP [D-1959 et FP-1963])*

3- *Méthode de Broyden-Fletcher-Goldfarb-Shanno (BFGS [1970])*

- Les algorithmes quasi-newtoniens convergent moins rapidement que l'algorithme de Newton. Dans les implémentations correctes, les itérés convergent toutefois q-superlinéairement.

- Chaque itération demande moins de calcul au simulateur que dans l'algorithme de Newton : il n'est pas nécessaire d'évaluer les dérivées secondes.

- Dans leur version standard, ces algorithmes peuvent être utilisés pour un nombre de variables qui n'est pas trop grand, disons  $n \leq 500$  pour fixer les idées, cependant on a développé ces dernières années (à partir de 1990) des algorithmes qui sont des variantes de la méthode BFGS et qui prennent en charge les problèmes de grande taille (citons particulièrement la méthode BFGS à mémoire limitée, La LBFGS).

Dans ce chapitre, nous discuterons en détail sur la méthode BFGS.

### 3.2.2 Formules de mise à jour de l'approximation du Hessien

Le principe de la mise à jour consiste à une itération donnée de l'algorithme



$$\begin{cases} \varkappa_{k+1} = \varkappa_k + \lambda_k d_k \\ d_k = -M_k g_k \end{cases}$$

Nous cherchons  $M_k$  sous la forme suivante :

$$M_{k+1} = M_k + \Delta_k \quad (3.3)$$

avec  $\Delta_k$  symétrique, assurant la relation de quasi-Newton. On exige aussi que  $M_{k+1}$  soit définie positive, sous l'hypothèse que  $M_k$  est définie positive.

La formule (3.3) permet d'utiliser les nouvelles informations obtenues lors de l'étape  $k$  de l'algorithme, c'est-à-dire essentiellement le gradient  $g_{k+1} = \nabla f(\varkappa_{k+1})$  au point  $\varkappa_{k+1}$ , obtenu par une recherche linéaire (exacte ou approchée) dans la direction  $d_k$ . Il existe différentes formules de type (3.1). Suivant que  $\Delta_k$  est de rang un ou deux, on parlera de correction de rang un (SR1) ou de rang deux (DFP, BFGS).

### 3.2.3 Méthode de correction de rang un (SR1 [1960])

Cette méthode a été proposée par Broyden ([6]), Davidon ([33]), Fiacco et McCormick ([3]), Murtagh et Sargent ([4]), et Wolfe ([25]).

Etant donné que  $[H(\varkappa_k)]^{-1}$  est symétrique, la formule de mise à jour de l'approximation du Hessian est la suivante :

$$M_{k+1} = M_k + a_k u_k u_k^\top \quad (3.4)$$

où  $u_k$  est un vecteur de  $IR^n$  et  $a_k$  constante. La matrice de correction  $\Delta_k = a_k u_k u_k^\top$  est symétrique et de rang un.

on a :

$$s_k = M_{k+1} y_k \quad (3.5)$$

d'après (3.4) on a

$$s_k = M_k y_k + a_k u_k u_k^\top y_k \quad (3.6)$$

et par conséquent

$$\begin{aligned} y_k^\top (s_k - M_k y_k) &= a_k y_k^\top u_k u_k^\top y_k \\ &= a_k (u_k^\top y_k)^2 \end{aligned} \quad (3.7)$$

Alternativement, de l'équation (3.6) on a :

$$(s_k - M_k y_k) = a_k u_k u_k^\top y_k = a_k (u_k^\top y_k) u_k$$

$$(s_k - M_k y_k)^\top = a_k y_k^\top u_k u_k^\top = a_k (u_k^\top y_k) u_k^\top$$

Si on prend

$$u_k^\top = (s_k - M_k y_k),$$

alors

$$a_k (u_k^\top y_k) = 1.$$

Soit en remplaçant :

$$a_k \cdot (s_k - M_k y_k)^\top \cdot y_k = 1$$

ou encore :

$$a_k = \frac{1}{(s_k - M_k y_k)^\top \cdot y_k},$$

et (3.4) devient :

$$M_{k+1} = M_k + \frac{(s_k - M_k y_k) \cdot (s_k - M_k y_k)^\top}{(s_k - M_k y_k)^\top \cdot y_k}$$

Cette formule porte le nom de formule SR1 (*système de rang 1*).

### Algorithme (Méthode de SR1)

#### Etape initiale :

Soit  $\varepsilon > 0$  (critère d'arrêt). Choisir un point initial  $x_1$  et une matrice symétrique définie positive  $M_1$  quelconque (par exemple  $M_1 = I$ ) poser  $k = 1$ , et aller aux étapes principales

#### Etapes principales.

##### Etape 1 :

Si  $\|\nabla f(x_k)\| < \varepsilon$  stop ; sinon, poser  $d_k = -M_k g_k$  et déterminer le pas optimal  $\lambda_k$  solution optimale du problème  $\min f(x_k + \lambda d_k)$ ,  $\lambda \geq 0$ . et poser  $x_{k+1} = x_k + \lambda_k d_k$

##### Etape 2 :

Construire  $M_{k+1}$  comme suit :

$$M_{k+1} = M_k + \frac{(s_k - M_k y_k) \cdot (s_k - M_k y_k)^\top}{(s_k - M_k y_k)^\top \cdot y_k}$$

avec

$$\begin{aligned} s_k &= \mathcal{X}_{k+1} - \mathcal{X}_k \\ y_k &= \nabla f(\mathcal{X}_{k+1}) - \nabla f(\mathcal{X}_k) \end{aligned}$$

Remplacer  $k$  par  $k + 1$  et aller à l'étape 1.  $\square$

Cet algorithme a un comportement remarquable dans le cas où  $f$  est une fonction quadratique.

### Avantages et inconvénients de la méthode de SR1

#### ♣ Avantages

- Cette méthode présente l'avantage, que le point  $\mathcal{X}_{k+1}$  n'a pas besoin d'être choisi comme le minimum exact, c'est à dire qu'on n'a pas besoin d'effectuer des recherches linéaire exactes.

#### ♣ Inconvénients

- Même si la fonction est quadratique, et même si son Hessien est défini positif, la matrice  $M_k$  n'est pas forcément définie positive.
- Le dénominateur  $(s_k - M_k y_k)^\top y_k$  peut devenir nul ou très petit, ce qui rend le procédé instable c'est-à-dire, la méthode n'est pas bien définie.

### 3.2.4 Méthode de Davidon-Fletcher-Powell (DFP [D-1959 et FP-1963])

Cette méthode a été proposée par Davidon en 1959 ([33]) et développé plus tard en 1963 par Fletcher et Powell ([31]). La formule de mise à jour de DFP est une formule de correction de rang deux. De façon plus précise construisons  $M_{k+1}$  en fonction de  $M_k$  de la forme :

$$M_{k+1} = M_k + A_k + \Delta_k \quad (3.8)$$

Bien sur le choix de  $A_k$  et  $\Delta_k$  doit prendre en considération l'équation de quasi-Newton

$$M_{k+1} y_k = s_k \quad (3.8\text{bis})$$

Si on écrit  $A_k$  sous la forme  $A_k = a_k u_k u_k^\top$  et  $\Delta_k = b_k v_k v_k^\top$ , alors (3.8 bis) prend la forme :

$$(a_k u_k u_k^\top) \cdot y_k + (b_k v_k v_k^\top) \cdot y_k = s_k - M_k \cdot y_k.$$

Comme dans le cas de la formule SR1, un bon choix de  $a_k$ ,  $b_k$ ,  $u_k$  et  $v_k$  conduit à la formule de correction de rang 2 suivante :

$$M_{k+1} = M_k + \frac{s_k s_k^\top}{s_k^\top y_k} - \frac{M_k y_k y_k^\top M_k}{y_k^\top M_k y_k} \quad (3.9)$$

### Algorithme (Méthode de DFP)

#### Etape initiale :

Soit  $\varepsilon > 0$ , déterminer le critère d'arrêt. Choisir un point initial  $x_1$  et une matrice symétrique définie positive  $M_1$  quelconque (par exemple  $M_1 = I$ ) poser  $k = 1$ , et aller aux étapes principales

#### Etapes principales.

##### Etape 1 :

Si  $\|\nabla f(x_k)\| < \varepsilon$  STOP ; sinon, poser  $d_k = -M_k g_k$  et déterminer le pas optimal  $\lambda_k$  solution optimale du problème

$\min f(x_k + \lambda d_k)$ ,  $\lambda \geq 0$ . et poser  $x_{k+1} = x_k + \lambda_k d_k$

##### Etape 2 :

Construire  $M_{k+1}$  comme suit :

$$M_{k+1} = M_k + \frac{s_k s_k^T}{s_k^T y_k} - \frac{M_k y_k y_k^T M_k}{y_k^T M_k y_k}$$

avec

$$s_k = x_{k+1} - x_k$$

$$y_k = \nabla f(x_{k+1}) - \nabla f(x_k)$$

Remplacer  $k$  par  $k + 1$  et aller à l'étape 1.  $\square$

Cet algorithme a un comportement remarquable dans le cas où  $f$  est une fonction quadratique.

Avantages et inconvénients de la méthode de DFP

#### ♣ Avantages

1- Pour des fonctions quadratiques (avec une recherche linéaire exacte) :

- L'algorithme converge dans au plus  $n$  étapes avec  $M_{n+1} = H^{-1}$ .
- Elles engendrent des directions conjuguées.

2- Pour les fonctions quelconques :

- la matrice  $M_k$  reste définie positive, ce qui est nécessaire pour que la direction soit une direction de descente.

### 3.2.5 Méthode de Broyden-Fletcher-Goldfard-Shanno (BFGS [1970])

La méthode de BFGS est une méthode quasi-newtonienne bien connue pour résoudre des problèmes d'optimisation sans contraintes, elle a été proposée par : Broyden ([6]), Fletcher ([32]), Goldfarb ([10]), et Shanno ([11]). Elle est devenue une méthode de choix pour les ingénieurs et les mathématiciens qui sont intéressés à résoudre des problèmes d'optimisation.

Une autre manière d'imposer à  $M_{k+1}$  d'être proche de  $M_k$  est de minimiser l'"écart" entre  $M_{k+1}$  et  $M_k$ , toujours en requérant que  $M_{k+1}$  soit symétrique et vérifie l'équation de quasi-Newton (3.5). On est donc conduit à considérer le problème en la variable matricielle  $M \in IR^{n \times n}$  suivant :

$$\begin{cases} \min \text{"écart"}(M, M_k) \\ y_k = M s_k \\ M = M^\top \end{cases} ; M \in IR^{n \times n} \quad (3.10)$$

On dit alors que la matrice est obtenue par une *approche variationnelle*.

**Notations :** Pour indiquer qu'une matrice carrée  $M$  est symétrique semi-définie positive (resp. définie positive), on notera  $M \geq 0$  (resp.  $M > 0$ ). L'ensemble des matrices symétriques d'ordre  $n$  est noté  $S^n$ ,

$$S_+^n := \{M \in S^n : M \geq 0\} \text{ et } S_{++}^n := \{M \in S^n : M > 0\}.$$

Il est souvent intéressant d'imposer également la définie positivité des matrices  $M_k$ , car pour que  $d_k = -M_k^{-1}g_k$  soit une direction de descente, on a besoin que  $M_k$  soit une matrice symétrique, définie positive (en effet : on sait que, toute matrice symétrique réelle et définie positive est inversible, et son inverse est elle aussi définie positive, c'est-à-dire on peut écrire :

$$g_k^\top d_k = -g_k^\top M_k^{-1}g_k < 0 \text{ donc, } d_k \text{ est une direction de descente}).$$

Cette condition définissant un ensemble ouvert ne peut être utilisée directement comme contrainte dans le problème définissant  $M_{k+1}$ ; et dans ce but, on commence par introduire une fonction :  $\psi : S^n \rightarrow IR$  dont le domaine est  $S_{++}^n$  et qui forme une "barrière" au bord du cône  $S_{++}^n$  (elle tend vers l'infini lorsque son argument se rapproche du bord de  $S_{++}^n$ ) ainsi qu'à l'infini :

$$\psi(\Upsilon) = tr\Upsilon + ld\Upsilon \quad (3.11)$$

où la fonction log-déterminant  $ld : S^n \rightarrow IR \cup \{+\infty\}$  est définie en  $\Upsilon \in S^n$  par

$$ld(\Upsilon) = \begin{cases} -\log \det \Upsilon & \text{si } \Upsilon \in S_{++}^n \\ +\infty & \text{sinon} \end{cases}$$

Les propriétés annoncées de  $\psi$  peuvent se voir sur son expression suivante. Si on note  $\{\alpha_i\}$  les valeurs propres de  $\Upsilon$ , on a  $\text{tr}\Upsilon = \sum_{i=1}^n \alpha_i$ ,  $\det \Upsilon = \prod_{i=1}^n \alpha_i$  et donc

$$\psi(\Upsilon) = \sum_{i=1}^n (\alpha_i - \log \alpha_i), \quad \text{si } \Upsilon \in S_{++}^n \quad (3.12)$$

Etant donné l'allure de la fonction  $t \in IR_{++} \mapsto t - \log t$ ,  $\psi(\Upsilon)$  tend vers l'infini si l'une des valeurs propres de  $\Upsilon$  tend vers zéro ou vers l'infini.

( $\exists j \in \{1, \dots, n\}$ ;  $\lim_{\alpha_j \rightarrow 0 \text{ ou } \infty} \psi(\Upsilon) = \infty$ ).

La formule (3.12) montre aussi que l'unique minimiseur de  $\psi$  est  $\Upsilon = I$  la matrice identité ( $\alpha_i = 1$  pour tout  $i$ ).

Si  $M_k$  est une matrice réelle symétrique alors :

la matrice  $M_k$  est définie positive si et seulement s'il existe une matrice définie positive  $A_k$  telle que  $A_k^2 = M_k$ ; dans ce cas, la matrice définie positive  $A_k$  et on peut la noter  $A_k = M_k^{\frac{1}{2}}$  est unique. en effet :

Si  $M_k$  est une matrice réelle symétrique alors on peut écrire

$$U^\top M_k U = \Lambda$$

où  $U$  est vérifiée

$$U^\top U = U U^\top = I$$

et  $\Lambda$  est une matrice diagonalisée où les éléments du diagonale sont les valeurs propres de  $M_k$  qui sont strictement positives car  $M_k$  est définie positive. Donc on peut écrire

$$M_k = U \Lambda U^\top = U \Lambda^{\frac{1}{2}} \Lambda^{\frac{1}{2}} U^\top = \left( U \Lambda^{\frac{1}{2}} U^\top \right) \left( U \Lambda^{\frac{1}{2}} U^\top \right) = M_k^{\frac{1}{2}} M_k^{\frac{1}{2}}. \quad \square$$

On veut dire, qu'il faut trouver une matrice  $M$  symétrique et définie positive être proche de  $M_k$  c'est-à-dire

$$M \approx M_k^{\frac{1}{2}} M_k^{\frac{1}{2}}$$

implique

$$M_k^{-\frac{1}{2}} M M_k^{-\frac{1}{2}} \approx I$$

Donc afin de minimisée l'écart entre  $M$  et  $M_k$ , on va chercher à ce que  $M_k^{-\frac{1}{2}} M M_k^{-\frac{1}{2}}$  soit proche de  $I$ ; et ceci peut être obtenu en minimisant  $\psi \left( M_k^{-\frac{1}{2}} M M_k^{-\frac{1}{2}} \right)$ , si bien que l'on aura  $M_{k+1}$  proche de  $M_k$  en résolvant :

$$\begin{cases} \min \psi \left( M_k^{-\frac{1}{2}} M M_k^{-\frac{1}{2}} \right) \\ y_k = M s_k \\ M \in S_{++}^n \quad (\text{contrainte implicite}). \end{cases} \quad (3.13)$$

Si  $s_k = 0$  alors  $\begin{cases} \text{si } y_k \neq 0 \text{ alors (3.13) n'a pas de solutions} \\ \text{ou bien} \\ \text{si } y_k = 0 \text{ alors la solution du (3.13) est } M = M_k \end{cases}$  ;

Sinon; le cas non trivial où  $s_k \neq 0$  est examiné dans la proposition suivante.

**Proposition** *Supposons que  $M_k$  soit symétrique définie positive et que  $s_k \neq 0$ . Alors, le problème (3.14) a une solution si et seulement si*

$$y_k^\top s_k > 0.$$

*Sous cette condition la solution  $M_{k+1}$  de (2.36) est unique et est donnée par l'une des formules suivantes :*

$$M_{k+1} = M_k + \frac{y_k y_k^\top}{y_k^\top s_k} - \frac{M_k s_k s_k^\top M_k}{s_k^\top M_k s_k} \quad (3.14)$$

$$B_{k+1} = \left( I - \frac{s_k y_k^\top}{y_k^\top s_k} \right) B_k \left( I - \frac{y_k s_k^\top}{y_k^\top s_k} \right) + \left( I - \frac{s_k s_k^\top}{y_k^\top s_k} \right) \quad (3.15)$$

où on a noté  $B_k := M_k^{-1}$  et  $B_{k+1} := M_{k+1}^{-1}$ .

**Algorithme (Méthode de BFGS)**

**Etape initiale :**

Soit  $\varepsilon > 0$ , déterminer le critère d'arrêt. Choisir  $\varkappa_1$  point initial et  $M_1$  définie positive quelconque (par exemple  $M_1 = I$ ).

Poser  $k = 1$  et aller aux étapes principales

**Etapes principales.**

**Etape 1 :**

Si  $\|\nabla f(\varkappa_k)\| < \varepsilon$  STOP; sinon, poser  $d_k = -M_k g_k$  et déterminer le pas optimal  $\lambda_k$  solution optimale du problème  $\min f(\varkappa_k + \lambda d_k)$ ,  $\lambda \geq 0$  et poser

$$\varkappa_{k+1} = \varkappa_k + \lambda_k d_k$$

**Etape 2 :**

Construire  $M_{k+1}$  comme suit :

$$M_{k+1} = M_k + \frac{y_k y_k^\top}{y_k^\top s_k} - \frac{M_k s_k s_k^\top M_k}{s_k^\top M_k s_k}$$

avec

$$s_k = \mathcal{X}_{k+1} - \mathcal{X}_k$$

$$y_k = \nabla f(\mathcal{X}_{k+1}) - \nabla f(\mathcal{X}_k)$$

Remplacer  $k$  par  $k + 1$  et aller à l'étape 1.

### Propriétés

- La condition  $y_k^\top s_k > 0$  assure la défini-positivité de  $M_k$ .
- La condition  $y_k^\top s_k > 0$  est satisfaite lorsqu' on utilise la recherche linéaire de Wolfe.
- Les directions  $s_1, s_2, \dots, s_n$  engendrées par la formule de BFGS sont conjuguées.
- Dans le cas quadratique la méthode de BFGS possède les mêmes propriétés que DFP.
- Dans le cas non quadratique, il faut procéder à des réinitialisations périodiques pour assurer la convergence globale.



# Chapitre 4

## Nouvelle preuve de la convergence globale de la méthode BFGS dans le cas non convexe

### 4.1 Méthode BFGS avec mise à jour appropriée

On a vu qu'une des propriétés de la formule de BFGS est que la matrice  $M_{k+1}$  hérite de la définie positivité de  $M_k$  si la condition  $y_k^\top s_k > 0$  est vérifiée. Remarquons que si on utilise une recherche linéaire exacte ou une recherche linéaire inexacte de Wolfe, alors la condition  $y_k^\top s_k > 0$  est vérifiée. Par contre, la recherche linéaire d'Armijo [19] n'assure pas cette condition, et par conséquent  $M_{k+1}$  n'est pas nécessairement définie positive même si  $M_k$  est définie positive. Afin d'assurer la définie positivité de  $M_{k+1}$ , la condition  $y_k^\top s_k > 0$  est parfois utilisée pour décider si  $M_{k+1}$  est une mise à jour ou non, c'est-à-dire on pose

$$M_{k+1} = \begin{cases} M_k + \frac{y_k y_k^\top}{y_k^\top s_k} - \frac{M_k s_k s_k^\top M_k}{s_k^\top M_k s_k}, & \text{si } y_k^\top s_k > 0 \\ M_k & , \text{ ailleurs} \end{cases} \quad (4.1)$$

La condition  $y_k^\top s_k > 0$  est souvent remplacée par  $y_k^\top s_k > \eta$  où  $\eta > 0$  est une petite constante. Li et Masao Fukushima [7] et [8] proposent une mise à jour appropriée à la méthode BFGS semblable à ce qui précède et patent de cela ils établissent un théorème de convergence globale pour des problèmes non convexes. Avant de décrire cette mise à jour appropriée, voyons d'abord un lemme important dû à Powell ([21]) et qui nous sera utile par la suite

**Lemme 4.1** (Powell ([21])) *Si la méthode de BFGS avec la recherche linéaire de Wolfe (2.6)-(2.7)[24] est appliquée à une fonction  $f$  continument différentiable ; et s'il existe une constante  $c > 0$*

telle que :

$$\frac{\|y_k\|^2}{y_k^\top s_k} \leq c \quad \text{pour tout } k \quad (4.2)$$

Alors

$$\liminf_{k \rightarrow \infty} \|g_k\| = 0. \quad (4.3)$$

Si  $f$  est deux fois continument différentiable et strictement convexe, alors on obtient toujours l'inégalité (4.2) mais, dans le cas où  $f$  est non convexe, il est difficile de garantir l'inégalité (4.2). Peut-être c'est une des raisons pour laquelle la convergence globale de la méthode de BFGS n'a pas été prouvée.

Nous allons maintenant présenter la méthode BFGS avec mise à jour appropriée et montrer plus tard qu'elle est globalement convergente sans que la fonction économique soit convexe.

Déterminons  $M_{k+1}$  en fonction de  $M_k$  par :

$$M_{k+1} = \begin{cases} M_k + \frac{y_k y_k^\top}{y_k^\top s_k} - \frac{M_k s_k s_k^\top M_k}{s_k^\top M_k s_k}, & \text{si } \frac{y_k^\top s_k}{\|s_k\|^2} \geq \varepsilon \|g_k\|^\alpha \\ M_k, & \text{sinon} \end{cases} \quad (4.4)$$

où  $\varepsilon$  et  $\alpha$  sont des constantes positives.

### Algorithme de Méthode BFGS avec mise à jour appropriée

**Etape 0** choisir un point initial  $\varkappa_0 \in IR^n$  et une matrice initiale  $M_0 \in IR^{n \times n}$  symétrique, définie positive ; choisir les constantes

$$0 < \sigma_1 < \sigma_2 > 1, \alpha > 0 \text{ et } \varepsilon > 0.$$

soit  $k = 0$

**Etape 1** résoudre l'équation linéaire  $M_k d_k + g_k = 0$  pour obtenir  $d_k$ .

**Etape 2** déterminer le pas  $\lambda_k > 0$  par la recherche linéaire inexacte de Wolfe ou d'Armijo.[24]

**Etape 3** Calculer  $\varkappa_{k+1} := \varkappa_k + \lambda_k d_k$ .

**Etape 4** déterminer  $M_{k+1}$  par

$$M_{k+1} = \begin{cases} M_k + \frac{y_k y_k^\top}{y_k^\top s_k} - \frac{M_k s_k s_k^\top M_k}{s_k^\top M_k s_k}, & \text{si } \frac{y_k^\top s_k}{\|s_k\|^2} \geq \varepsilon \|g_k\|^\alpha \\ M_k, & \text{sinon} \end{cases} \quad (4.5)$$

avec

$$\begin{aligned} s_k &= \varkappa_{k+1} - \varkappa_k \\ y_k &= \nabla f(\varkappa_{k+1}) - \nabla f(\varkappa_k) \end{aligned} \quad (4.6)$$

**Etape 5** Remplacer  $k$  par  $k + 1$  et aller à l'étape 1.

**Remarque 4.1** *il n'est pas difficile de voir que les matrices  $M_k$  générés par l'algorithme de Méthode BFGS avec mise à jour appropriée sont symétriques et défini-positives pour tout  $k$ . Ceci implique qu'avec l'utilisation d'une recherche linéaire inexacte de Wolfe ou d'Armijo, qu'on obtienne que la suite  $\{f(\mathcal{x}_k)\}_{k \in \mathbb{N}}$  est décroissante.*

Nous avons aussi en considération (wolfe1) ou (2.6)-(2.7) et si  $f$  est bornée inférieurement :

$$-\sum_{k=0}^{\infty} g_k^\top s_k < \infty. \quad (4.7)$$

Ceci implique que

$$-\lim_{k \rightarrow \infty} (-g_k^\top s_k) = 0 \quad (4.8)$$

et puisque

$$s_k = \mathcal{x}_{k+1} - \mathcal{x}_k = \lambda_k d_k. \quad (4.9)$$

Alors

$$-\lim_{k \rightarrow \infty} (\lambda_k g_k^\top d_k) = 0 \quad (4.10)$$

## 4.2 Convergence globale de Méthode BFGS avec mise à jour appropriée

Dans cette section, nous prouvons la convergence globale de l'algorithme de Méthode BFGS sous l'hypothèse suivante :

### Hypothèse 1

Considérons l'ensemble  $\Omega$  suivant :

$$\Omega = \{\mathcal{x} \in IR^n / f(\mathcal{x}) \leq f(\mathcal{x}_0)\}. \quad (4.11)$$

Nous supposons que  $\Omega$  est contenu dans un ensemble convexe et bornée  $D$  et que la fonction économique  $f$  est continument différentiable sur  $D$  et qu'il existe une constante  $l > 0$  telle que :

$$\|g(\mathcal{x}) - g(y)\| \leq l \|\mathcal{x} - y\|, \quad \forall \mathcal{x}, y \in D \quad (4.12)$$

c'est-à-dire  $f \in C^{1,1}(D)$ .

Comme la suite  $\{f(\mathcal{x}_k)\}$  est décroissante, il est clair que la suite  $\{\mathcal{x}_k\}$  générée par l'algorithme de Méthode BFGS avec mise à jour appropriée est contenue dans  $\Omega$  car

$$f(\mathcal{x}_{k+1}) < f(\mathcal{x}_k) < \dots < f(\mathcal{x}_1) < f(\mathcal{x}_0)$$

Pour des raisons de convenance et pour être plus pratique, définissons les ensembles d'indices suivants :

$$\bar{K} = \left\{ i \mid \frac{y_i^\top s_i}{\|s_i\|^2} \geq \varepsilon \|g_i\|^\alpha \right\} . \quad (4.13)$$

et

$$\bar{K}_k = \left\{ i \in \bar{K} \mid i \leq k \right\}$$

Notons par  $i_k$ , l'ensemble des indices  $i \in \bar{K}_k$

on peut récrire (4.4) sous la forme :

$$M_{k+1} = \begin{cases} M_k + \frac{y_k y_k^\top}{y_k^\top s_k} - \frac{M_k s_k s_k^\top M_k}{s_k^\top M_k s_k}, & \text{si } k \in \bar{K} \\ M_k & , \text{ sinon} \end{cases} \quad (4.14)$$

et en considérant la trace des deux membres de (4.14), on peut écrire pour n'importe quel  $k \in \mathbb{N}$

$$Tr(M_{k+1}) = Tr(M_1) + \sum_{i \in \bar{K}_k} \frac{\|y_i\|^2}{y_i^\top s_i} - \sum_{i \in \bar{K}_k} \frac{\|M_i s_i\|^2}{s_i^\top M_i s_i} \quad (4.15)$$

**Preuve.** pour des matrices A, B de,  $Tr(A + B) = Tr(A) + Tr(B)$

$$\begin{aligned} Tr(M_{k+1}) &= Tr\left(M_k + \frac{y_k y_k^\top}{y_k^\top s_k} - \frac{M_k s_k s_k^\top M_k}{s_k^\top M_k s_k}\right) \\ &= Tr(M_k) + Tr\left(\frac{y_k y_k^\top}{y_k^\top s_k}\right) - Tr\left(\frac{M_k s_k s_k^\top M_k}{s_k^\top M_k s_k}\right) \\ &= Tr(M_k) + \frac{1}{y_k^\top s_k} Tr(y_k y_k^\top) - \frac{1}{s_k^\top M_k s_k} Tr(M_k s_k s_k^\top M_k) \\ &= Tr(M_k) + \frac{\|y_k\|^2}{y_k^\top s_k} - \frac{\|M_k s_k\|^2}{s_k^\top M_k s_k} \\ &= Tr(M_1) + \sum_{i \in \bar{K}_k} \frac{\|y_i\|^2}{y_i^\top s_i} - \sum_{i \in \bar{K}_k} \frac{\|M_i s_i\|^2}{s_i^\top M_i s_i} \end{aligned}$$

■

**Théorème 4.1** . On suppose vraie l'hypothèse 1 et soit  $\{z_k\}$  la suite générée par l'algorithme de Méthode BFGS avec mise à jour appropriée

Si  $\bar{K}$  est un ensemble fini, alors

$$\lim_{k \rightarrow \infty} \|g_k\| = 0. \quad (4.16)$$

**Preuve.** si  $\bar{K}$  est un ensemble fini alors il existe un indice  $k_0$  tel que

$$M_k = M_{k_0} \triangleq M \quad \text{pour tout } k \geq k_0$$

Par la défini- positivité de  $M$ , il existe des constantes positives  $c_1 \leq C_1$  telles que

$$\begin{cases} c_1 \|d\|^2 \leq d^\top M d \leq C_1 \|d\|^2 \\ c_1 \|d\|^2 \leq d^\top M^{-1} d \leq C_1 \|d\|^2 \end{cases} \quad (4.17)$$

✂ Si la recherche linéaire inexacte utilisée est de Wolfe, on obtient de (4.17)

$$y_k^\top s_k \leq \|y_k^\top\| \|s_k\| \leq l \|s_k\|^2$$

on utilise la condition de Wolfe :

$$g(x_{k+1})^\top s_k \geq \sigma_2 g(x_k)^\top s_k$$

donc :

$$\begin{aligned} y_k^\top s_k &= (g^\top(x_{k+1}) - g^\top(x_k)) s_k \\ &= g^\top(x_{k+1}) s_k - g^\top(x_k) s_k \\ &\geq \sigma_2 g_k^\top s_k - g_k^\top s_k \\ &\geq -(1 - \sigma_2) g_k^\top s_k \end{aligned}$$

d'autre part :

$$\begin{aligned} l \|s\|^2 &\geq y_k^\top s_k \geq -(1 - \sigma_2) g_k^\top s_k \\ &= (1 - \sigma_2) \lambda_k^{-1} s_k^\top M^{-1} s_k \\ &\geq (1 - \sigma_2) \lambda_k^{-1} c_1 \|s_k\|^2 \end{aligned}$$

ceci implique :

$$\lambda_k \geq (1 - \sigma_2) c_1 l^{-1}, \quad \forall k \geq k_0$$

par conséquent on obtient de (4.12) :

$$\lim_{k \rightarrow \infty} \lambda_k g^\top d_k = \lim_{k \rightarrow \infty} (-g_k^\top s_k) = 0$$

avec  $\lambda_k \geq (1 - \sigma_2) c_1 l^{-1} > 0, \quad \forall k \geq k_0$ , on peut écrire

$$g_k^\top M^{-1} g_k = -g_k^\top d_k \rightarrow 0$$

et de (4.17) on a :

$$c_1 \|g_k\|^2 \leq g_k^\top M^{-1} g_k \rightarrow 0$$

alors

$$c_1 \|g_k\|^2 \rightarrow 0$$

avec  $c_1 \neq 0$ , donc

$$\lim_{k \rightarrow \infty} \|g_k\| = 0. \quad \square$$

■

Maintenant nous allons prouver la convergence globale de **l'algorithme de Méthode BFGS avec mise à jour appropriée** dans le cas où  $\bar{K}$  est un ensemble infini.

Si on suppose le contraire, alors il existe une constante  $\delta > 0$  telle que

$$\|g_k\| \geq \delta \quad \text{pour tout } k \quad (4.18)$$

et nous verrons que cela produit une contradiction. Avant d'établir le théorème de convergence globale de **l'algorithme de Méthode BFGS avec mise à jour appropriée** nous montrons d'abord quelques lemmes utiles.

**Lemme 4.2** *Supposons que l'hypothèse 1 soit vraie et que  $\{\varkappa_k\}$  est une suite générée par l'algorithme de Méthode BFGS avec mise à jour appropriée.*

*Supposons aussi que la relation (4.18) soit vraie pour tout  $k$ . alors il existe une constante  $c_2 > 0$  telle que :*

$$Tr(M_{k+1}) \leq c_2 i_k$$

et

$$\sum_{i \in \bar{K}_k} \frac{\|M_i s_i\|^2}{s_i^\top M_i s_i} \leq c_2 i_k$$

pour  $k$  suffisamment grand.

**Preuve.** on a d'après (4.13) et (4.18)

$$y_i^\top s_i \geq \varepsilon \|g_i\|^\alpha \|s_i\|^2 \geq \varepsilon \delta^\alpha \|s_i\|^2 \quad \text{pour tout } i \in \bar{K}$$

implique

$$y_i^\top s_i \geq \varepsilon \delta^\alpha \|s_i\|^2 \quad \text{pour tout } i \in \bar{K}$$

$$\frac{1}{y_i^\top s_i} \leq \frac{1}{\varepsilon \delta^\alpha \|s_i\|^2} \dots (*)$$

$$\begin{aligned} \|y_i\| \|s_i\| &\leq l \|s_i\|^2 \\ \|y_i\| &\leq l \|s_i\| \\ \|y_i\|^2 &\leq l^2 \|s_i\|^2 \dots (**) \end{aligned}$$

du (\*) et (\*\*) on obtient :

$$\frac{\|y_i\|^2}{y_i^\top s_i} \leq \frac{l^2}{\varepsilon \delta^\alpha} \triangleq c'_2$$

on obtient

$$\begin{aligned} Tr(M_{k+1}) &= Tr(M_1) + \sum_{i \in \bar{K}} \frac{\|y_i\|^2}{y_i^\top s_i} - \underbrace{\sum_{i \in \bar{K}} \frac{\|M_i s_i\|^2}{s_i^\top M_i s_i}}_{\text{terme positive}} \\ &\leq Tr(M_1) + \sum_{i \in \bar{K}} \frac{\|y_i\|^2}{y_i^\top s_i} \\ &\leq i_k (c_0 + c'_2) = i_k c_2 \end{aligned}$$

où on pose  $c_2 = \max(c_0, c'_2)$ . On obtient donc :

$$Tr(M_{k+1}) \leq i_k c_2.$$

$$\frac{\|y_i\|^2}{y_i^\top s_i} \leq \frac{l^2}{\varepsilon \delta^\alpha} \triangleq c'_2$$

Puisque  $Tr(M_{k+1}) > 0$  pour n'importe quel  $k$ , on obtient,

$$0 < Tr(M_1) + \sum_{i \in \bar{K}_k} \frac{\|y_i\|^2}{y_i^\top s_i} - \sum_{i \in \bar{K}_k} \frac{\|M_i s_i\|^2}{s_i^\top M_i s_i}$$

ceci implique

$$\sum_{i \in \bar{K}_k} \frac{\|M_i s_i\|^2}{s_i^\top M_i s_i} < Tr(M_1) + i_k c'_2 < i_k c_2$$

c'est-à-dire

$$\sum_{i \in \bar{K}_k} \frac{\|M_i s_i\|^2}{s_i^\top M_i s_i} \leq c_2 i_k.$$

■

### 4.3 Convergence globale de l'algorithme de Méthode BFGS avec mise à jour appropriée avec la recherche linéaire de Wolfe

Pour cet effet, nous prouvons d'abord un lemme semblable au lemme 2

**Lemme 4.3** *Supposons que l'hypothèse 1 soit vraie. Soit  $\{\alpha_k\}_{k \in \mathbb{N}}$  une suite générée par l'algorithme de Méthode BFGS avec mise à jour appropriée avec  $\lambda_k$  déterminé par la recherche linéaire de Wolfe (2.6)-(2.7)[24]. Si on a (4.15) pour tout  $k \in \mathbb{N}$ , alors il existe une constante  $c_3 > 0$  tel que pour tout  $k$  assez grand on ait :*

$$\prod_{i \in \tilde{K}_k} \lambda_i \geq c_3^{i_k} \quad (4.19)$$

**Preuve.** On a La formule de la trace se déduit directement de la relation BFGS suivante :

$$M_{k+1} = M_k + \frac{\|y_k\|^2}{y_k^\top s_k} - \frac{\|M_k s_k\|^2}{s_k^\top M_k s_k} \quad (BFGS),$$

celle du déterminant utilise la formule suivante :

$$\det(M_{k+1}) = \det(M_k) \cdot \det\left(I + \frac{M_k^{-1} y_k y_k^\top}{y_k^\top s_k} - \frac{s_k s_k^\top M_k}{s_k^\top M_k s_k}\right)$$

avec  $M_k^{-1} y_k = s_k$ , implique

$$\det(M_{k+1}) = \det(M_k) \cdot \det\left(\frac{y_k^\top s_k s_k^\top M_k s_k + s_k y_k^\top s_k^\top M_k s_k - y_k^\top s_k s_k s_k^\top M_k}{y_k^\top s_k s_k^\top M_k s_k}\right)$$

implique

$$\begin{aligned} \det(M_{k+1}) &= \det(M_k) \cdot \det\left(\frac{y_k^\top s_k y_k^\top s_k + s_k y_k^\top y_k^\top s_k - y_k^\top s_k s_k y_k^\top}{y_k^\top s_k s_k^\top M_k s_k}\right) \\ &= \det(M_k) \cdot \det\left(\frac{s_k y_k^\top + s_k y_k^\top - s_k y_k^\top}{s_k^\top M_k s_k}\right) \\ &= \det(M_k) \cdot \left(\frac{y_k^\top s_k}{s_k^\top M_k s_k}\right) \quad .\text{pour tout } i \in \bar{K} \end{aligned}$$

et on a

$$\det(M_{i+1}) = \det(M_i) \quad \forall i \notin \bar{K}$$

Si on note par  $n_k$  le plus grand indice dans l'ensemble  $\bar{K}$  alors on peut écrire

$$\det(M_{n_k+1}) = \det(M_1) \prod_{i \in \bar{K}_k} \frac{y_i^\top s_i}{s_i^\top M_i s_i}$$



D'autre part, on utilise la condition de (2.8) :

$$g^\top (\boldsymbol{x}_i + \lambda_i d_i) d_i \geq \sigma_2 g^\top (\boldsymbol{x}_i) d_i$$

implique

$$\begin{aligned} y_i^\top s_i &= (g_{i+1} - g_i)^\top s_i \\ &= g_{i+1}^\top s_i - g_i^\top s_i \\ &\geq \sigma_2 g_i^\top s_i - g_i^\top s_i \end{aligned}$$

ceci implique :

$$y_i^\top s_i \geq -(1 - \sigma_2) g_i^\top s_i$$

avec

$$g^\top = -d_i^\top M_i = -\lambda_i^{-1} s_i^\top M_i$$

donc

$$\begin{aligned} y_i^\top s_i &\geq (1 - \sigma_2) \lambda_i^{-1} s_i^\top M_i s_i, \\ \frac{y_i^\top s_i}{s_i^\top M_i s_i} &\geq (1 - \sigma_2) \lambda_i^{-1} = \frac{(1 - \sigma_2)}{\lambda_i} \end{aligned}$$

on peut écrire

$$\prod_{i \in \bar{K}_k} \frac{y_i^\top s_i}{s_i^\top M_i s_i} \geq \prod_{i \in \bar{K}_k} \frac{1 - \sigma_2}{\lambda_i}$$

on obtient

$$\det(M_{n_k+1}) \geq \det(M_1) \prod_{i \in \bar{K}_k} \frac{1 - \sigma_2}{\lambda_i}$$

ou

$$\det(M_{n_k+1}) \leq \left[ \frac{\text{Tr}(M_{n_k+1})}{n} \right]^n,$$

par conséquent,

$$\prod_{i \in \bar{K}_k} \frac{1 - \sigma_2}{\lambda_i} \leq \frac{\det(M_{n_k+1})}{\det(M_1)} \leq \frac{1}{\det(M_1)} \left[ \frac{\text{Tr}(M_{n_k+1})}{n} \right]^n \leq \frac{1}{\det(M_1)} \left[ \frac{c_2 i_k}{n} \right]^n$$

Donc

$$\begin{aligned} \prod_{i \in \bar{K}_k} \frac{1 - \sigma_2}{\lambda_i} &\leq \frac{1}{\det(M_1)} \left[ \frac{c_2 i_k}{n} \right]^n \\ \prod_{i \in \bar{K}_k} \frac{1 - \sigma_2}{\lambda_i} &\leq \frac{1}{\det(M_1) n^n} [c_2^{i_k}]^n \\ &\leq \frac{1}{\det(M_1) n^n} [c_2^n]^{i_k} \\ \prod_{i \in \bar{K}_k} \frac{\lambda_i}{1 - \sigma_2} &\geq \det(M_1) n^n [c_2^{i_k}]^n \\ \prod_{i \in \bar{K}_k} \lambda_i &\geq (1 - \sigma_2) \det(M_1) n^n [c_2^n]^{i_k} \end{aligned}$$

donc il existe une constante  $c_3 = (1 - \sigma_2) \det(M_1) n^n [c_2^n]$   
telle que

$$\prod_{i \in \tilde{K}_k} \lambda_i \geq c_3^{i_k}.$$

■

Maintenant nous sommes en mesure de prouver **la convergence globale de l'algorithme de Méthode BFGS** avec la **recherche linéaire de Wolfe** donnée par le théorème suivant :

**Théorème 4.2** [13] *Supposons que l'hypothèse 1 soit vraie. Soit  $\{\nu_k\}_{k \in \mathbb{N}}$  une suite générée par l'algorithme de Méthode BFGS avec  $\lambda_k, k \in \mathbb{N}$  déterminé par la recherche linéaire de Wolfe (2.6)-(2.7). Alors on a*

$$\liminf_{k \rightarrow \infty} \|g_k\| = 0 \quad (4.20)$$

**Preuve.** En prenant en considération le théorème 1, il suffit de vérifier (4.20) dans le cas où  $\bar{K}$  est infini. Notons  $\bar{K}$  par

$$\bar{K} = \{k_1 < k_2 < \dots < k_n\}.$$

Remarquons que (4.7) donne particulièrement

$$-\sum_{j=0}^{\infty} g_{k_j}^\top s_{k_j} < \infty$$

$$\begin{aligned} -g_{k_j}^\top s_{k_j} &= -\|g_{k_j}\|^2 \lambda_{k_j}^2 \frac{g_{k_j}^\top s_{k_j}}{\|g_{k_j}\|^2 \lambda_{k_j}^2} \\ &= \|g_{k_j}\|^2 \lambda_{k_j} \frac{-\lambda_{k_j} g_{k_j}^\top s_{k_j}}{\|\lambda_{k_j} g_{k_j}\|^2} \\ &= \|g_{k_j}\|^2 \lambda_{k_j} \frac{s_{k_j}^\top M_{k_j} s_{k_j}}{\|M_{k_j} s_{k_j}\|^2} \end{aligned}$$

car :  $M_{k_j} s_{k_j} = -\lambda_{k_j} g_{k_j}$

alors :

$$\sum_{j=1}^{\infty} \|g_{k_j}\|^2 \lambda_{k_j} \frac{s_{k_j}^\top M_{k_j} s_{k_j}}{\|M_{k_j} s_{k_j}\|^2} = -\sum_{j=0}^{\infty} g_{k_j}^\top s_{k_j} < \infty \quad (4.21)$$

si (4.3) pas satisfait alors , il existe une constante  $\delta > 0$  tel que  $\|g_k\| \geq \delta$  pour tout  $k$ .

$$\sum_{j=1}^{\infty} \lambda_{k_j} \frac{s_{k_j}^\top M_{k_j} s_{k_j}}{\|M_{k_j} s_{k_j}\|^2} < \infty.$$

Par conséquent, pour n'importe quel  $\xi > 0$ , il existe un entier  $j_0 > 0$ , tel que pour tout entier positif  $q$ ,

on a

$$\left( \prod_{j=j_0+1}^{j_0+q} \lambda_{k_j} \frac{s_{k_j}^\top M_{k_j} s_{k_j}}{\|M_{k_j} s_{k_j}\|^2} \right)^{\frac{1}{q}} \leq \frac{1}{q} \sum_{j=j_0+1}^{j_0+q} \lambda_{k_j} \frac{s_{k_j}^\top M_{k_j} s_{k_j}}{\|M_{k_j} s_{k_j}\|^2} \leq \frac{\xi}{q}$$

ceci implique que

$$\begin{aligned} \left( \prod_{j=j_0+1}^{j_0+q} \lambda_{k_j} \frac{s_{k_j}^\top M_{k_j} s_{k_j}}{\|M_{k_j} s_{k_j}\|^2} \right)^{\frac{1}{q}} &\leq \frac{\xi}{q} \\ \left( \prod_{j=j_0+1}^{j_0+q} \lambda_{k_j} \right)^{\frac{1}{q}} &\leq \frac{\xi}{q} \left( \prod_{j=j_0+1}^{j_0+q} \frac{\|M_{k_j} s_{k_j}\|^2}{s_{k_j}^\top M_{k_j} s_{k_j}} \right)^{\frac{1}{q}} \\ &\leq \frac{\xi}{q^2} \sum_{j=j_0+1}^{j_0+q} \frac{\|M_{k_j} s_{k_j}\|^2}{s_{k_j}^\top M_{k_j} s_{k_j}} \\ &\leq \frac{\xi}{q^2} \sum_{j=0}^{j_0+q} \frac{\|M_{k_j} s_{k_j}\|^2}{s_{k_j}^\top M_{k_j} s_{k_j}} \end{aligned}$$

et on a par (4.19)

$$\sum_{j=0}^{j_0+q} \frac{\|M_{k_j} s_{k_j}\|^2}{s_{k_j}^\top M_{k_j} s_{k_j}} \leq c_2 i_k$$

où  $i_k = (j_0 + q + 1)$

donc

$$\sum_{j=0}^{j_0+q} \frac{\|M_{k_j} s_{k_j}\|^2}{s_{k_j}^\top M_{k_j} s_{k_j}} \leq (j_0 + q + 1) c_2$$

alors

$$\left( \prod_{j=j_0+1}^{j_0+q} \lambda_{k_j} \right)^{\frac{1}{q}} \leq \frac{\xi (j_0 + q + 1)}{q^2} c_2$$

Si  $q \rightarrow \infty$ , alors on aboutit à une contradiction, car le lemme (3) vérifie que le terme à gauche de l'inégalité ci-dessus est plus grand qu'une constante positive. ■

**Remarque.** Pour montrer la convergence globale de la méthode BFGS et la recherche linéaire inexacte de Wolfe, il est suffisant de montrer implicitement l'existence de la condition  $\frac{y_k^T s_k}{\|s_k\|^2} \geq \varepsilon \|g_k\|^\alpha$ . C'est-à-dire que le BFGSA est consacré à la méthode BFGS.

✂ **premier étape :** pour tous  $k \geq 1$

$$\frac{y_k^T s_k}{\|s_k\|^2} \geq (1 - \sigma_2) c_1 \lambda_k^{-1} \quad (4.22)$$

en effet, en utilisant (2.7) nous avons

$$\begin{aligned}
 y_k^\top s_k &= (g^\top(\mathcal{X}_{k+1}) - g^\top(\mathcal{X}_k)) s_k \\
 &= g^\top(\mathcal{X}_{k+1}) s_k - g^\top(\mathcal{X}_k) s_k \\
 &\geq \sigma_2 g_k^\top s_k - g_k^\top s_k \\
 &\geq -(1 - \sigma_2) g_k^\top s_k
 \end{aligned}$$

où  $s_k = \lambda_k d_k$

$$d_k^\top = \lambda_k^{-1} s_k^\top \text{ et } g_k^\top = -d_k^\top M_k^{-1} \quad (4.23)$$

alors

$$\begin{aligned}
 y_k^\top s_k &\geq -(1 - \sigma_2) g_k^\top s_k \\
 y_k^\top s_k &\geq (1 - \sigma_2) \lambda_k^{-1} s_k^\top M_k^{-1} s_k
 \end{aligned}$$

et si la recherche linéaire inexacte utilisée est de wolfe, alors nous avons la condition  $y_k^\top s_k \geq 0$  c'est-à-dire que le caractère définitif positif de  $M_k$  est préservé, donc, il existe des constantes positives  $c_1 \leq C_1$

$$c_1 \|z\|^2 \leq z^\top M z \leq C_1 \|z\|^2,$$

$$c_1 \|z\|^2 \leq z^\top M^{-1} z \leq C_1 \|z\|^2 \quad \forall z \in \mathbb{R}^n$$

par (4.23) on a

$$y_k^\top s_k \geq (1 - \sigma_2) \lambda_k^{-1} c_1 \|s_k\|^2$$

implique

$$\begin{aligned}
 \frac{y_k^\top s_k}{\|s_k\|^2} &\geq (1 - \sigma_2) c_1 \lambda_k^{-1} \\
 \frac{y_k^\top s_k}{\|s_k\|^2} &\geq (1 - \sigma_2) c_1 \lambda_k^{-\frac{1}{k}}
 \end{aligned}$$

✠ *Deuxième étape* : pour tous  $k \geq 1$

$$\left( \prod_{i=1}^k \lambda_i \|g_i\|_2^2 \right)^{\frac{1}{k}} \leq \frac{c}{k}, \quad c > 0 \quad (4.24)$$

en effet, on a

$$\sum_{i=1}^k \frac{\|M_i s_i\|_2^2}{s_i^\top M_i s_i} = \sum_{i=1}^k \frac{\lambda_i \|g_i\|_2^2}{-g_i^\top s_i} \leq c_1.$$

alors, On utilise ensuite deux fois l'inégalité des moyennes et la première condition de Wolfe (2.7) donc

$$\lambda_i \|g_i\|_2^2 \leq c_1 (-g_i^\top s_i)$$

implique

$$\begin{aligned} \prod_{i=1}^k \lambda_i \|g_i\|_2^2 &\leq c_1^k \prod_{i=1}^k (-g_i^\top s_i) \\ &\leq \left( \frac{c_1}{k} \sum_{i=1}^k (-g_i^\top s_i) \right)^k \end{aligned}$$

et on a

$$f(x_i + \lambda_i d_i) \leq f(x_i) + \sigma_1 \lambda_i g(x_i)^\top d_i$$

et  $s_i = \lambda_i d_i$

donc

$$-\sigma_1 g(x_i)^\top s_i \leq f(x_i) - f(x_{i+1})$$

$$-g(x_i)^\top s_i \leq \frac{f(x_i) - f(x_{i+1})}{\sigma_1}$$

alors

$$\begin{aligned} \prod_{i=1}^k \lambda_i \|g_i\|_2^2 &\leq \left( \frac{c_1}{\sigma_1 k} \sum_{i=1}^k (f(x_i) - f(x_{i+1})) \right)^k \\ \left( \prod_{i=1}^k \lambda_i \|g_i\|_2^2 \right)^{\frac{1}{k}} &\leq \left( \frac{c_1}{\sigma_1 k} (f(x_1) - f(x_{k+1})) \right) \end{aligned}$$

On en déduit (4.24), avec  $c = c_1 (f(x_1) - f_{\min}) / \sigma_1$  où  $f_{\min} \in \mathbb{R}$  est un minorant de  $\{f(x_k)\}_{k \in \mathbb{N}}$ .

alors

$$\left( \prod_{i=1}^k \lambda_i \|g_i\|_2^2 \right)^{\frac{1}{k}} \leq \frac{c}{k}$$

✠ troisième étape :

On conclut que

$$\frac{y_k^T s_k}{\|s_k\|^2} \geq \varepsilon \|g_k\|^\alpha$$

en effet, par (4, 22) on obtient pour tout  $k \geq 1$

$$(\lambda_k \|g_k\|^2)^{\frac{1}{k}} \leq \frac{c}{k}$$

implique pour tout  $k \geq 1$

$$\frac{k}{c} (\lambda_k \|g_k\|^2)^{\frac{1}{k}} \leq 1$$

$$\frac{k}{c} (\|g_k\|^2)^{\frac{1}{k}} \leq (\lambda_k)^{-\frac{1}{k}}$$

$$\frac{k}{c} (\|g_k\|)^{\frac{2}{k}} \leq (\lambda_k)^{-\frac{1}{k}} \quad (4.25)$$

par (4.22) implique

$$\frac{y_k^T s_k}{\|s_k\|^2} \geq (1 - \sigma_2) c_1 \lambda_k^{-\frac{1}{k}}$$

et on a par (4.25)

$$(\lambda_k)^{-\frac{1}{k}} \geq \frac{k}{c} (\|g_k\|)^{\frac{2}{k}}$$

alors

$$\frac{y_k^T s_k}{\|s_k\|^2} \geq (1 - \sigma_2) c_1 \frac{k}{c} (\|g_k\|)^{\frac{2}{k}}$$

et si on met  $\alpha = \frac{2}{k}$  et  $\varepsilon = (1 - \sigma_2) c_1 \frac{k}{c}$ , on obtient

$$\frac{y_k^T s_k}{\|s_k\|^2} \geq \varepsilon \|g_k\|^\alpha$$

## 4.4 Tests numériques

Cette section contient une étude numérique faite avec l'**algorithme de Méthode BFGS avec mise à jour appropriée**. Nous avons testé l'**algorithme de Méthode BFGS avec mise à jour appropriée** sur les trois problèmes suivants pris de ([20]).

**Problème 1 Extension de la fonction singulière de Powell**

$$f(\mathcal{x}) = \sum_{i=1}^{n/4} \{ (\mathcal{x}_{4i-3} + 10\mathcal{x}_{4i-2})^2 + 5(\mathcal{x}_{4i-1} - \mathcal{x}_{4i})^2 \\ + (\mathcal{x}_{4i-2} - 2\mathcal{x}_{4i-1})^4 + 10(\mathcal{x}_{4i-1} - \mathcal{x}_{4i})^4 \},$$

$$\mathcal{x}^* = (0, 0, \dots, 0)^\top.$$

**Problème 2 Fonction de Rosenbrock (Extension)**

$$f(\mathcal{x}) = \sum_{i=1}^{n/2} \{ 100(\mathcal{x}_{2i} - \mathcal{x}_{2i-1}^2) + (1 - \mathcal{x}_{2i-1})^2 \},$$

$$\mathcal{x}^* = (1, 1, \dots, 1)^\top.$$

**Problème 3 Fonction de Wood (Extension)**

$$f(\mathcal{x}) = \sum_{i=1}^{n/4} \{ 100(\mathcal{x}_{4i-2} - \mathcal{x}_{4i-3}^2)^2 + (1 - \mathcal{x}_{4i-3})^2 + 90(\mathcal{x}_{4i} - \mathcal{x}_{4i-1}^2)^2 \\ + (1 - \mathcal{x}_{4i-1})^2 + 10(\mathcal{x}_{4i-2} + \mathcal{x}_{4i} - 2)^2 + 0.1(\mathcal{x}_{4i-2} - \mathcal{x}_{4i})^2 \},$$

Nous avons appliqué l'**algorithme de Méthode BFGS avec mise à jour appropriée** qu'on a appelé la méthode BFGSA (A : abréviation de appropriée), avec la recherche linéaire de Wolfe à ces problèmes et nous l'avons comparé avec la méthode ordinaire de BFGS. Nous avons utilisé la condition  $\max \{ \|g(\mathcal{x}_k)\|, \|\mathcal{x}_k - \mathcal{x}^*\| \} \leq 10^{-5}$  comme critère d'arrêt, et pour chaque problème, nous avons choisi différents points de départ (points initiaux) mais avec la même matrice initiale  $M_0 = I$  (matrice d'unité). Pour chaque problème, les paramètres communs aux deux méthodes ont été placés identiquement. Spécialement nous avons choisi les paramètres comme suit  $\sigma_1 = 0.1$  et  $\sigma_2 = 0.49$  dans la recherche linéaire de Wolfe (2.6)-(2.7) et  $\alpha = \begin{cases} 0.01, & \text{si } \|g_k\| \geq 1 \\ 3, & \text{si } \|g_k\| < 1 \end{cases}$ ,  $\varepsilon = 0.1$ , dans la mise à jour appropriée (4.4). Il n'est pas difficile de voir que le théorème de la convergence dans la section 4.2 est vrai si le choix de  $\alpha$  est comme ci-dessus ; en général, si  $\alpha \in [\mu_1, \mu_2]$  avec  $\mu_1 > 0$ , tous les théorèmes dans la section 4.2 sont vrais. Les résultats sont donnés dans les tableaux, où P1, P2 et P3 représentent les problèmes 1, 2 et 3, respectivement, et  $k$  représente le nombre d'itérations. Pour la méthode de BFGSA, "off" dénote le nombre de fois où la condition dans la mise à jour prudente n'a pas été réunie, c'est-à-dire, le nombre des  $k$  où  $\frac{y_k^\top s_k}{\|s_k\|^2} < \varepsilon \|g_k\|^\alpha$ . Pour la méthode de BFGS, "off" dénote le nombre des  $k$  où  $y_k^\top s_k < 10^{-17}$ , et pour

la méthode de BFGS avec la recherche linéaire de Wolfe, "off" est égale à zéro comme  $y_k^\top s_k$  est toujours positive à cause de l'inégalité de (2.7).

Dans les tableaux "init" présente le point initial. Les points initiaux sont  $x_0 = (0, 0, \dots, 0)^\top$ ,  $x_1 = (1, 1, \dots, 1)^\top$ ,  $x_2 = (10, 10, \dots, 10)^\top$ ,  
 $x_3 = (100, 100, \dots, 100)^\top$ ,  $x_4 = -x_2$ ,  $x_5 = -x_3$ ,  
 $x_6 = (0, 100, 0, 100, \dots)^\top$  et  $x_7 = -x_6$ .

Dans la suite, pour voir l'influence de  $\alpha$  et  $\varepsilon$  dans la méthode de BFGSA, on fait la résolution du problème 1 avec des valeurs différent de  $\alpha$  et  $\varepsilon$  partant du point initial  $x_2 = (10, 10, \dots, 10)^\top$ .

Les résultats sont dans le tableau 3 où

$$\alpha_1 = \begin{cases} 0.0001, & \text{si } \|g_k\| \geq 1 \\ 4, & \text{si } \|g_k\| < 1 \end{cases}, \alpha_2 = \begin{cases} 0.001, & \text{si } \|g_k\| \geq 1 \\ 4, & \text{si } \|g_k\| < 1 \end{cases}, \alpha_3 = \begin{cases} 0.01, & \text{si } \|g_k\| \geq 1 \\ 4, & \text{si } \|g_k\| < 1 \end{cases},$$

$$\alpha_4 = \begin{cases} 0.1, & \text{si } \|g_k\| \geq 1 \\ 4, & \text{si } \|g_k\| < 1 \end{cases},$$

$$\alpha_5 = \begin{cases} 1, & \text{si } \|g_k\| \geq 1 \\ 4, & \text{si } \|g_k\| < 1 \end{cases}, \alpha_6 = 1, \alpha_7 = 2.$$

Nous avons constaté que la condition de la mise à jour appropriée a été habituellement satisfaite. Cependant, quand elle est violée, la méthode de BFGSA est aussi mauvaise que la méthode BFGS. Les résultats prouvent aussi que le choix des paramètres  $\alpha$  et  $\varepsilon$  sont numériquement importants. Si nous choisissons  $\alpha$  et  $\varepsilon$  convenablement, alors la condition de la mise à jour appropriée est presque toujours satisfaite et la méthode de BFGSA se réduit à la méthode de BFGS classique.



Tableau 1

tests numériques pour la méthode de BFGSA (recherche de Wolfe)

Problem	$n$	Init	$k$	$\ \nabla f_k\ $	$\ x_k - x^*\ $	off
P1	4	$x^1$	31	1.0E-7	1.3E-3	0
P1	4	$x^2$	35	1.0E-6	3.6E-3	0
P1	4	$x^3$	39	1.0E-6	4.6E-3	0
P1	4	$x^4$	35	1.0E-6	3.6E-3	0
P1	4	$x^5$	39	1.0E-6	4.6E-3	0
P1	4	$x^6$	43	1.0E-6	1.7E-3	0
P1	4	$x^7$	43	1.0E-6	1.7E-3	0
P2	2	$x^0$	17	1.1E-4	1.0E-5	0
P2	2	$x^2$	62	2.0E-6	1.2E-5	1
P2	2	$x^4$	75	1.0E-7	2.0E-6	1
P2	2	$x^6$	44	1.1E-5	1.0E-6	0
P2	2	$x^7$	25	1.0E-7	4.3E-5	0
P2	10	$x^0$	37	1.7E-4	8.0E-6	0
P2	10	$x^2$	134	1.0E-6	1.5E-5	0
P2	10	$x^4$	140	2.9E-5	6.0E-6	1
P2	10	$x^6$	99	7.7E-5	6.0E-6	0
P2	10	$x^7$	70	3.0E-6	1.5E-5	0
P2	100	$x^0$	93	4.7E-5	6.0E-6	0
P2	100	$x^2$	354	8.0E-6	1.3E-5	0
P2	100	$x^6$	308	5.1E-5	7.0E-6	0
P2	100	$x^7$	277	3.6E-5	9.0E-6	0
P3	4	$x^0$	19	3.4E-5	4.0E-6	0
P3	4	$x^2$	29	4.9E-5	1.0E-7	0
P3	4	$x^3$	45	2.4E-4	6.0E-6	0
P3	4	$x^6$	31	4.0E-4	1.0E-6	0
P3	4	$x^7$	38	7.2E-5	1.0E-7	0
P3	40	$x^0$	51	6.9E-5	4.0E-6	0
P3	40	$x^2$	185	7.0E-4	8.0E-6	0
P3	40	$x^3$	499	1.3E-4	6.0E-6	0
P3	40	$x^4$	530	7.4E-5	3.0E-6	0
P3	40	$x^5$	524	8.0E-6	1.0E-6	0
P3	40	$x^6$	168	1.8E-4	5.0E-6	0
P3	40	$x^7$	148	2.0E-5	1.0E-5	0
P3	100	$x^0$	75	1.8E-4	6.0E-6	0
P3	100	$x^2$	352	1.0E-4	8.0E-6	0
P3	100	$x^3$	758	2.1E-5	5.0E-6	0
P3	100	$x^4$	875	1.6E-4	5.0E-6	0
P3	100	$x^5$	1239	1.1E-4	7.0E-6	0
P3	100	$x^6$	289	1.0E-4	5.0E-6	0
P3	100	$x^7$	233	3.4E-4	7.0E-6	0

Tableau 2

tests numériques pour la méthode de BFGSA (recherche de Wolfe)

Problem	$n$	Init	$k$	$\ \nabla f_k\ $	$\ x_k - x^*\ $	off
P1	4	$x^1$	31	1.0E-7	1.3E-3	0
P1	4	$x^2$	35	1.0E-6	3.6E-3	0
P1	4	$x^3$	39	1.0E-6	4.6E-3	0
P1	4	$x^4$	35	1.0E-6	3.6E-3	0
P1	4	$x^5$	39	1.0E-6	4.6E-3	0
P1	4	$x^6$	43	1.0E-6	1.7E-3	0
P1	4	$x^7$	43	1.0E-6	1.7E-3	0
P2	2	$x^0$	17	1.1E-4	1.0E-5	0
P2	2	$x^2$	61	1.8E-4	4.0E-6	0
P2	2	$x^4$	70	1.1E-4	7.0E-6	0
P2	2	$x^6$	44	1.1E-5	1.0E-6	0
P2	2	$x^7$	25	1.0E-7	4.3E-5	0
P2	10	$x^0$	37	1.7E-4	8.0E-6	0
P2	10	$x^2$	134	1.0E-6	1.5E-5	0
P2	10	$x^4$	144	1.4E-5	2.0E-6	0
P2	10	$x^6$	99	7.7E-5	6.0E-6	0
P2	10	$x^7$	70	3.0E-6	1.5E-5	0
P2	100	$x^0$	93	4.7E-5	6.0E-6	0
P2	100	$x^2$	354	8.0E-6	1.3E-5	0
P2	100	$x^6$	308	5.1E-5	7.0E-6	0
P2	100	$x^7$	277	3.6E-5	9.0E-6	0
P3	4	$x^0$	19	3.4E-5	4.0E-6	0
P3	4	$x^2$	29	4.9E-5	1.0E-7	0
P3	4	$x^3$	45	2.4E-4	6.0E-6	0
P3	4	$x^6$	31	4.0E-4	1.0E-6	0
P3	4	$x^7$	38	7.2E-5	1.0E-7	0
P3	40	$x^0$	51	6.9E-5	4.0E-6	0
P3	40	$x^2$	185	7.0E-4	8.0E-6	0
P3	40	$x^3$	499	1.3E-4	6.0E-6	0
P3	40	$x^4$	530	7.4E-5	3.0E-6	0
P3	40	$x^5$	524	8.0E-6	1.0E-6	0
P3	40	$x^6$	168	1.8E-4	5.0E-6	0
P3	40	$x^7$	148	2.0E-5	1.0E-5	0
P3	100	$x^0$	75	1.8E-4	6.0E-6	0
P3	100	$x^2$	352	1.0E-4	8.0E-6	0
P3	100	$x^3$	758	2.1E-5	5.0E-6	0
P3	100	$x^4$	875	1.6E-4	5.0E-6	0
P3	100	$x^5$	1239	1.1E-4	7.0E-6	0
P3	100	$x^6$	289	1.0E-4	5.0E-6	0
P3	100	$x^7$	233	3.4E-4	7.0E-6	0

Tableau 5 (recherche de Wolfe) pour la fonction de Powell  
 Différentes choix des paramètres dans la méthode de BFGSA

$n$	$\alpha$	$\epsilon$	$k$	$\ \nabla f_k\ $	$\ x_k - x^*\ $	off
4	$\alpha_1$	1	36	8.0E-6	3.9E-3	11
4	$\alpha_1$	$10^{-1}$	32	9.0E-6	7.2E-3	2
4	$\alpha_1$	$10^{-3}$	32	9.0E-6	8.3E-3	0
4	$\alpha_1$	$10^{-5}$	32	9.0E-6	8.3E-3	0
4	$\alpha_2$	1	36	8.0E-6	3.9E-3	11
4	$\alpha_2$	$10^{-1}$	32	9.0E-6	7.2E-3	2
4	$\alpha_2$	$10^{-3}$	32	9.0E-6	8.3E-3	0
4	$\alpha_2$	$10^{-5}$	32	9.0E-6	8.3E-3	0
4	$\alpha_3$	1	36	8.0E-6	3.9E-3	11
4	$\alpha_3$	$10^{-1}$	32	9.0E-6	7.2E-3	2
4	$\alpha_3$	$10^{-3}$	32	9.0E-6	8.3E-3	0
4	$\alpha_3$	$10^{-5}$	32	9.0E-6	8.3E-3	0
4	$\alpha_4$	1	36	8.0E-6	3.9E-3	11
4	$\alpha_4$	$10^{-1}$	32	9.0E-6	7.2E-3	2
4	$\alpha_4$	$10^{-3}$	32	9.0E-6	8.3E-3	0
4	$\alpha_4$	$10^{-5}$	32	9.0E-6	8.3E-3	0
4	$\alpha_5$	$10^{-1}$	36	4.0E-6	5.4E-3	5
4	$\alpha_5$	$10^{-3}$	32	9.0E-6	8.3E-3	0
4	$\alpha_5$	$10^{-5}$	32	9.0E-6	8.3E-3	0
4	$\alpha_6$	$10^{-1}$	1056	1.0E-5	1.1E-2	1043
4	$\alpha_6$	$10^{-3}$	187	6.0E-6	4.8E-3	164
4	$\alpha_6$	$10^{-5}$	32	9.0E-6	8.3E-3	0
4	$\alpha_7$	$10^{-3}$	13	8.0E-6	2.4E-3	0
4	$\alpha_7$	$10^{-5}$	2	1.0E-5	2.4E-3	0
40	$\alpha_1$	$10^{-1}$	243	9.0E-6	1.7E-2	9
40	$\alpha_1$	$10^{-3}$	238	9.0E-6	1.6E-2	0
40	$\alpha_1$	$10^{-5}$	238	9.0E-6	1.6E-2	0
40	$\alpha_2$	$10^{-1}$	243	9.0E-6	1.7E-2	9
40	$\alpha_2$	$10^{-3}$	238	9.0E-6	1.6E-2	0
40	$\alpha_2$	$10^{-5}$	238	9.0E-6	1.6E-2	0
40	$\alpha_3$	$10^{-1}$	243	9.0E-6	1.7E-2	9
40	$\alpha_3$	$10^{-3}$	238	9.0E-6	1.6E-2	0
40	$\alpha_3$	$10^{-5}$	238	9.0E-6	1.6E-2	0
40	$\alpha_4$	$10^{-1}$	243	9.0E-6	1.7E-2	9
40	$\alpha_4$	$10^{-3}$	238	9.0E-6	1.6E-2	0
40	$\alpha_4$	$10^{-5}$	238	9.0E-6	1.6E-2	0
40	$\alpha_5$	$10^{-3}$	238	9.0E-6	1.6E-2	0
40	$\alpha_5$	$10^{-5}$	238	9.0E-6	1.6E-2	0
40	$\alpha_7$	$10^{-3}$	82	3.0E-6	3.1E-3	61
40	$\alpha_7$	$10^{-5}$	1	7.0E-6	3.1E-3	0

## 4.5 Conclusion

Ce travail propose une nouvelle preuve de la convergence globale de la méthode BFGS pour le problème de minimisation sans contrainte non convexe, il est prouvé que la condition de la méthode appropriée est satisfaite implicitement à une recherche linéaire inexacte de Wolfe, en suite il vérifié directement la convergence de la méthode BFGS avec la recherche linéaire inexacte de Wolfe.

# Bibliographie

- [1] A. Antoniou, W-S. Lu, Practical Optimization. Algorithms and Engineering Applications, Springer Science Business Media, LLC, 2007.
- [2] A. Griewank, The global convergence of partitioned BFGS on problems with convex decompositions and Lipschitzian gradients, Math. Program., 50 (1991), 141{175}.
- [3] A. V. Fiacco and G. P. McCormick, (1968) "Nonlinear Programming", (Wiley, New York).
- [4] B. A. Murtagh and R. W. H. Sargent (1969), "A constrained minimization method with quadratic convergence", (Optimization, ed. R. Fletcher, pp. 215-246, Academic Press, London).
- [5] Benzine Rachid 27 février 2016 Optimisation sans Contraintes tome1 : Conditions d'optimalite. Recherches lineaires. Methode du gradient . Methode de newton.
- [6] C. G. Broyden, (1965) "Quasi-Newton methods and their application to function minimization", (Maths. Comput., vol. 21, pp. 368–381).
- [7] D.H. Li and M. Fukushima, "A globally and superlinearly convergent Gauss-Newton based BFGS method for symmetric nonlinear equations", (to appear in SIAM Journal on Numerical Analysis).
- [8] D.H. Li and M. Fukushima, ""A modified BFGS method and its global convergence in nonconvex minimization", (Technical Report 98003, Department of Applied Mathematics and Physics, Kyoto University, January 1998).

- 
- [9] D.H. Li, M. Fukushima, On the global convergence of the BFGS method for nonconvex unconstrained optimization problem, *SIAM J. Optim.* 11 (2001), 1054{1064}.
- [10] D. Goldfarb (1970), "A family of variable metric methods derived by variational means", (*Maths. Comput.*, vol. 24, pp. 23–26).
- [11] D. F. Shanno (1970), "Conditioning of quasi-Newton methods for function minimization", (*Maths. Comput.*, vol. 24, pp. 647–656).
- [12] G.Allaire "L'ouvrage complet est disponible auprès des Editions de l'Ecole Polytechnique" paris, le 4 Janvier 2005.
- [13] Hakima Degaichia and Salah Boulaaras "A new proof for the global convergence of the BFGS method for nonconvex unconstrained minimization problems" *Italian Journal of Pure and Applied Mathematics*,N.38, 2017, 469-486.
- [14] J. Nocedal, Theory of algorithms for unconstrained optimization, *Acta Numerica*, 1 (1992), 199{242}.
- [15] J.E. Dennis, J.J. More, Quasi-Newton methods, motivation and theory, *SIAM Review*, 19 (1977), 46{89}.
- [16] J. Nocedal, Theory of algorithms for unconstrained optimization, *Acta Numerica*, 1 (1992), 199{242}.
- [17] Jorge Nocedal et Stephen J.Wright "Numerical Optimization with 85 illustrations" (1999 springer-verlag New York, Inc).
- [18] L.C.W. Dixon, Variable metric algorithms : Necessary and sufficient conditions for identical behavior on nonquadratic functions, *J. Optim. Theory Appl.* 10 (1972), 34{40}.
- [19] L. Armijo (1966), "Minimization of function having lipschitz continuous . . . rst partial derivatives", (*Pacific Journal of Mathematics*, Vol. 16(1), pp.1-3).
- [20] M.J.D. Powell, On the convergence of the variable metric algorithm, (*Journal of the Institute of Mathematics and its Applications*, 7 (1971), 21{36}.

- 
- [21] M.J.D. Powell, Some global convergence properties of a variable metric algorithm for minimization without exact line searches, in *Nonlinear Programming*, SIAM-AMS Proc. IX, R. W. Cottle and C. E. Lemke, eds., AMS, Providence, RI, 1976, 53{72.
- [22] Michel Minoux (1983), "Programmation Mathématique, Théorie et Algorithmes, (tome 1 , Dunod).
- [23] M. S. Bazaraa, H. D. Sherali, and C. M. Shetty (1993), "Nonlinear Programming, Theory and Algorithms", (Wiley-Interscience).
- [24] Ph.L. Toint, Global convergence of the partitioned BFGS algorithm for convex partially separable optimization, *Math. Program.*, 36 (1986), 290{306.
- [25] P.Wolfe (1967), "Methods of nonlinear programming", (*Nonlinear Programming*, ed. J. Abadie, pp. 97–131, Interscience, Wiley, New York).
- [26] P. Wolfe (1969), "Convergence conditions for ascent methods", (*SIAM Review*, 11, pp. 226-235).
- [27] R. Fletcher, *Practical Methods of Optimization*, Second Edition, John Wiley & Sons, Chichester, 1987.
- [28] R. Byrd and J. Nocedal, "A tool for the analysis of quasi-Newton methods with application to unconstrained minimization", (*SIAM Journal on Numerical Analysis*, 26 (1989) 727-739).
- [29] R. Byrd, J. Nocedal and Y.Yuan, "Global convergence of a class of quasi-Newton methods on convex problems", (*SIAM Journal on Numerical Analysis*, 24 (1987) 1171-1189).
- [30] R. Fletcher, An overview of unconstrained optimization, in *Algorithms for Continuous Optimization : The State of the Art*", (E. Spedicato ed.), Kluwer Academic Publishers, Boston, 1994, 109-143.
- [31] R. Fletcher and M. J. D Powell (1963) "A rapidly convergent descent method for minimization". (*Comput. J.* 6 ? 163-168).
- [32] R. Fletcher (1970), "A new approach to variable metric algorithms", (*Computer J.*, vol. 13, pp. 317–322).
- [33] W. C. Davidon (1959), "Variable Metric Methods For Minimization",