وزارة التعليم العالي والبحث العلمي

Université Badji Mokhtar Annaba

Badji Mokhtar University - Annaba

جامعة باجي مختار

عنابة

Faculté des Sciences
Département de Mathématiques

THESE

Présentée en vue de l'obtention du diplôme de
Doctorat en Mathématiques
Option : Analyse Numérique et Optimisation

Développements récents de la méthode du gradient conjugué

Par :
M0HAMMED Belloufi

Sous la direction de
Pr. RACHID Benzine

Devant le jury

PRESIDENT :	Laskri YAMINA	Pr.	Université Badji Mokhtar d'Annaba.
EXAMINATRICE :	Rebbani FAOUZIA	Pr.	Université Badji Mokhtar d'Annaba.
EXAMINATEUR :	Benterki DJAMEL	Pr.	Université Sétif 1.
EXAMINATEUR :	Ellaggoun FATEH	M.C.A	Université 8 mai 1945 de Guelma.

Année : 2013/2014

الإهداء

الحمد لله رب العالمين والصلاة والسلام على سيدنا محمد صلى الله عليه وعلى آله وصحبة ومن سار على نهجه واقتفى أثره إلى يوم الدين، وبعد:

فأحمد الله عز وجل على ما من به عليّ من إتمام هذه الرسالة،

كما أتقدم بالثناء والتقدير

إلى ..... من تحت قدمها تكمن الجنة، إلى أمي الحنون.

إلى ..... من جعل مشواري العلمي ممكنا، إلى أبي الرحيم.

إلى ..... من ساندني وآزرني في دربي، إلى زوجتي الصابرة.

إلى ..... من لأجله سرت في الدرب، إلى أبني العزيز علي.

إلى ..... إخواني وأخواتي الأعزاء

إلى ..... الأهل والأقارب

إليهم جميعا أهدي جهدي المتواضع هذا راجيا الله الإطالة بأعمارهم ليروا ثمرة جهدهم.

انطلاقا من العرفان بالجميل، فإنه ليسرني وليثلج صدري أن أتقدم بالشكر والامتنان إلى أستاذي، ومشرفي الأستاذ الدكتور بن زين رشيد الذي مدني من منابع علمه بالكثير، والذي ما توانى يوما عن مد يد المساعدة لي وفي جميع المجالات، وحمدا لله بأن يسره في دربي ويسر به أمري وعسى أن يطيل عمره ليبقى نبراسا متلألئا في نور العلم والعلماء.

كما أتتقدم بجزيل الشكر إلى أساتذتي أعضاء لجنة النقاش الموقرين على ما تكبدوه من عناء في قراءة رسالتي المتواضعة وإغنائها بمقترحاتهم القيمة.

وفي النهاية يسرني أن أتقدم بجزيل الشكر إلى كل من مد لي يد العون في مسيرتي العلمية.

Remerciement

Je tiens à adresser mes remerciements les plus chaleureux et ma profonde gratitude à mon encadreur Monsieur. R. Benzine, professeur à l'université d'Annaba pour m'avoir proposé le sujet de cette thèse. C'est grâce à sa grande disponibilité, ses conseils, ses orientations, et ses encouragements que j'ai pu mener à bien ce travail.

Mes remerciements vont également à...madame Y.Laskri, professeur à l'université d'Annaba, pour avoir bien voulu me faire l'honneur d'accepter de présider le jury.

De même je remercie Mme Rebbani Faouzia. Pr Université Badji Mokhtar Annaba, Mr Benterki Djamel. Pr. Université Sétif 1 et Mr Ellagoun Fateh, Maitre de Conférences classe A Université 8 Mai 45 Guelma, pour l'honneur qu'ils m'ont fait de bien vouloir accepter de faire partie du jury.

Enfin, je n'oublie pas de remercier toutes les personnes qui m'ont facilité la tâche et tous ceux que j'ai connu à l'institut de mathématiques qui ont rendu mes séjours au département agréables.

ملخص

لتكن $f:\mathbb{R}^n \to \mathbb{R}$. المطلوب هو حل مسالة الأمثلة بدون قيود التالية: $\min\{f(x): x\in\mathbb{R}^n\}.....(P)$

تعتبر خوارزمية طريقة التدرج السلمي المشتق **إستيان استيفال** من أهم الطرق المستخدمة في الأمثليات بدون قيود لما تتميز به من خصائص عددية جيدة، إلاّ أنه لا توجد أي نتائج تثبت تقارب هذه الخوارزمية المرفقة بالبحث الخطي المعروف.

في هذه الأطروحة تم الاعتماد على البحث الذي قدمه **شي و شان (2007)** لاقتراح تعديل جديد في البحث غير الدقيق **أرميجوا** بطريقة تضمن تقارب خوارزمية التدرج السلمي المشتق **إستيان استيفال.** وقد تم عرض نتائج التجارب العددية لإثبات مدى تميز الخوارزمية الجديدة.

الكلمات المفتاحية

التدرج السلمي المشتق، خوارزمية، التقارب العام، البحث الخطي غير الدقيق، معيار ارميجو، معيار فولف، طريقة إستبان استيفال، طريقة فلتشر ريفز، طريقة بولاك ربيار بولاياك، طريقة الهبوط المشتق، طريقة داي يوان، مزج طرق التدرج السلمي المشتق.

Résumé

Soit $f:\mathbb{R}^n \to \mathbb{R}$. On cherche à résoudre le problème de minimisation sans contraintes suivant: $\min\{f(x): x\in\mathbb{R}^n\}.....(P)$

L'algorithme de la méthode du Gradient conjugué Hestenes-Stiefel (HS) est un outil de l'optimisation numérique sans contraintes qui a une bonne performance numérique, mais aucun résultat de convergence globale avec la recherche linéaire inexacte n'a été prouvé. Dans cette thèse, en s'inspirant du travail de **Shi, Z.J et Shen, J. (2007)**, on propose une nouvelle modification de la recherche linéaire inexacte d'Armijo qui nous assure la convergence globale de la méthode du gradient conjugué, version Hestenes et Steifel: $\beta_k^{HS} = \frac{g_k^T y_k}{d_{k-1}^T y_{k-1}}$. Des tests numériques sont donnés pour montrer la performance du nouveau algorithme.

Mots clés: Gradient conjugué, Algorithme, Convergence globale,Recheche linéaire inexacte, Règle d'Armijo, Règle de Wolfe (forte et faible), Méthode de Hestenes-Stiefel, Méthode de Fletcher-Reeves, Méthode de Polak-Ribière-Polyak, Méthode de la descente conjuguée, Méthode de Dai-Yuan,Méthode hybride du Gradient conjugué .

Abstract

Consider the following unconstrained optimization problem, $\min\{f(x): x\in\mathbb{R}^n\}.....(P)$ where $f:\mathbb{R}^n \to \mathbb{R}$

The Hestenes-Stiefel (HS) conjugate gradient algorithm is a useful tool of unconstrained numerical optimization. which has good numerical performance but no global convergence result under traditional line searches. In this thesis, we proposes a line search technique that guarante the global convergence of the Hestenes-Stiefel (HS) conjugate gradient method, in which β_k is defined by $\beta_k^{HS} = \frac{g_k^T y_k}{d_{k-1}^T y_{k-1}}$. Numerical tests are presented to validate the different approaches.

Key words: Conjugate gradient, Algorithm, Global convergence, Inexact line search, Armijo line search, Strong Wolfe line search, Weak Wolfe line search, Hestenes-Stiefel Method, Fletcher-Reeves Method, Polak-Ribière-Polyak Method, Conjugate descent Method, Dai-Yuan Method, Hybrid Conjugate Gradient Method.

Table des matières

Table des matières

Introduction générale

Dans la vie courante, nous sommes fréquemment confrontés à des problèmes "d'optimisation" plus ou moins complexes. Cela peut commencer au moment où l'on tente de ranger son bureau, de placer son mobilier, et aller jusqu'à un processus industriel, par exemple pour la planification des différentes tâches. Ces problèmes peuvent être exprimés sous la forme générale d'un "problème d'optimisation".

L'optimisation peut être définie comme la science qui détermine la meilleure solution à certains problèmes mathématiquement définis, qui sont souvent des modèles de physique réel. C'est une technique qui permet de "quantifier" les compromis entre des critères parfois non commensurables ([03]).

L'optimisation recouvre l'étude des critères d'optimalité pour les différents problèmes, la détermination des méthodes algorithmiques de solution, l'étude de la structure de telles méthodes et l'expérimentation de l'ordinateur avec ces méthodes avec vrais problèmes de la vie ([34]).

D'un point de vue mathématique, l'optimisation consiste à rechercher le *minimum* ou le *maximum* d'une fonction avec ou sans contraintes.

L'optimisation possède ses racines au 18ième siècle dans les travaux de :

-Taylor, Newton , Lagrange, qui ont élaboré les bases des développements limités.

- Cauchy ([14]) fut le premier à mettre en œuvre une méthode d'optimisation, méthode du pas de descente, pour la résolution de problèmes sans contraintes.

Il faut attendre le milieu du vingtième siècle, avec l'émergence des calculateurs et surtout la fin de la seconde guerre mondiale pour voir apparaître des avancées spectaculaires en termes de techniques d'optimisation. A noter, ces avancées ont été essentiellement obtenues en Grande Bretagne.

De nombreuses contributions apparaissent ensuite dans les années soixante. G. Zoutendijk ([81]), C. W. Carroll ([11]), P. Wolfe ([73]), R. Fletcher et M. J. D. Powell ([36]), C. Reeves ([35]), A. A. Goldstein ([44]) et A. V. Fiacco et G. P. McCormick ([37]) pour la programmation non linéaire ainsi que E. Polak et G. Ribière([56]), B.T. Polyak ([57]) et J.F. Price ([43]).

Soit $f : \mathbb{R}^n \to \mathbb{R}$. On cherche à résoudre le problème de minimisation sans contraintes suivant :

$$(P) : \ \min \{f(x) : x \in \mathbb{R}^n\} \tag{0.1}$$

Parmi les plus anciennes méthodes utilisées pour résoudre les problèmes du type (P), on peut citer la méthode du Gradient conjugué. Cette méthode est surtout utilisée pour les problèmes de grande taille. Cette méthode a été découverte en 1952 par Hestenes et Steifel ([48]), pour la minimisation de fonctions quadratiques strictement convexes.

Plusieurs mathématiciens ont étendu cette méthode pour le cas non linéaire. Ceci a été réalisé pour la première fois, en 1964 par Fletcher et Reeves ([35]) (*méthode de Fletcher-Reeves*) puis en 1969 par Polak, Ribière ([56]) et Polyak ([57]) (*méthode de Polak-Ribière-Polyak*). Une autre variante a été étudiée en 1987 par Fletcher ([34]) (*Méthode de la descente conjuguée*).

Toutes ces méthodes génèrent une suite $\{x_k\}_{k\in\mathbb{N}}$ de la façon suivante :

$$x_{k+1} = x_k + \alpha_k d_k \tag{0.2}$$

Le pas $\alpha_k \in \mathbb{R}$ est déterminé par une optimisation unidimensionnelle ou recherche linéaire exacte ou inexacte.

Les directions d_k sont calculées de façon récurrente par les formules suivantes :

$$d_k = \begin{cases} -g_k \quad \text{si } k = 1 \\ -g_k + \beta_k d_{k-1} \quad \text{si } k \geq 2 \end{cases} \tag{0.3}$$

$g_k = \nabla f(x_k)$ et $\beta_k \in \mathbb{R}$.

Les différentes valeurs attribuées à β_k définissent les différentes formes du gradient conjugué.

Si on note $y_{k-1} = g_k - g_{k-1}, s_k = x_{k+1} - x_k$ on obtient les variantes suivantes :

1- Gradient conjugué variante Hestenes - Stiefel(HS)[48]

$$\beta_k^{HS} = \frac{g_{k+1}^T y_k}{d_k^T y_k}$$

2- Gradient conjugué variante Fletcher Reeves(FR)[35]

$$\beta_k^{FR} = \frac{\|g_k\|^2}{\|g_{k-1}\|^2}$$

3- Gradient conjugué variante Daniel (D)[32]

$$\beta_k^{D} = \frac{g_{k+1}^T \nabla^2 f(x_k)\, d_k}{d_k^T \nabla^2 f(x_k)\, d_k}$$

4- Gradient conjugué variante Polak-Ribière-Polyak(PRP)[56,57]

$$\beta_k^{PRP} = \frac{g_k^T y_k}{\|g_{k-1}\|^2}$$

5- Gradient conjugué variante descente – Fletcher (CD)[34]

$$\beta_k^{CD} = -\frac{\|g_k\|^2}{d_{k-1}^T g_{k-1}}$$

6- Gradient conjugué variante Liu - Storey(LS)[52]

$$\beta_k^{LS} = -\frac{g_{k+1}^T y_k}{d_k^T g_k}$$

7- Gradient conjugué variante de *Dai-Yuan*(DY)[24]

$$\beta_k^{DY} = \frac{\|g_k\|^2}{d_{k-1}^T y_{k-1}}$$

8- Gradient conjugué variante de Dai–Liao (DL)[19]

$$\beta_k^{DL} = \frac{g_{k+1}^T (y_k - t s_k)}{y_k^T s_k}$$

9- Gradient conjugué variante Hager-Zhang(HZ)[45]

$$\beta_k^{HZ} = \left(y_k - 2 d_k \frac{\|y_k\|^2}{d_k^T y_k} \right)^T \frac{g_{k+1}}{d_k^T y_k}$$

10- Gradient conjugué variante de Z. Wei [74]

$$\beta_k^* = \frac{g_k^T \left(g_k - \frac{\|g_k\|}{\|g_{k-1}\|} g_{k-1} \right)}{\|g_{k-1}\|^2}$$

11- Gradient conjugué variante de Hao Fan, Zhibin Zhu et Anwa Zhou [38]

$$\beta_k^{MN} = \frac{\|g_k\|^2 - \frac{\|g_k\|}{\|g_{k-1}\|} g_k^T g_{k-1}}{\mu \left| g_k^T d_{k-1} \right| + \|g_{k-1}\|^2}$$

12- Gradient conjugué variante Rivaie-Mustafa-Ismail-Leong(RMIL)[60]

$$\beta_k^{RMIL} = \frac{g_k^T \left(g_k - g_{k-1} \right)}{\|d_{k-1}\|^2}$$

Rappelons que les premiers résultats de convergence de la méthode du gradient conjugué ont d'abord été établis avec des recherches linéaires inexactes de Wolfe fortes, c'est à dire que les pas α_k dans (0.2), doivent vérifier aussi les deux relations suivantes :

$$f(x + \alpha_k d_k) \leq f(x_k) + \rho \alpha_k \nabla f(x_k)^T d_k \tag{0.4}$$

$$\left| \nabla f(x + \alpha_k d_k)^T . d_k \right| \leq -\sigma \nabla f(x_k)^T d_k \tag{0.5}$$

avec

$$0 < \rho < \sigma < 1$$

Le premier théorème démontrant la descente d'une méthode du gradient conjugué non linéaire fut établi par Albaali ([1]). Albaali a utilisé $\beta_k^{FR} = \frac{\|g_k\|^2}{\|g_{k-1}\|^2}$ et la recherche linéaire inexacte de Wolfe forte (0.4), (0.5) avec $\sigma \leq \frac{1}{2}$. Gilbert et Nocedal ([39]) ont généralisé ce résultat pour tout algorithme du gradient conjugué dont

$$|\beta_k| \leq \beta_k^{FR}$$

Ces résultats furent par la suite généralisés en considérant des recherches linéaires inexactes du type Wolfe faibles, c'est à dire que les pas α_k dans (0.2), doivent vérifier les deux relations plus faibles que (0.4) et (0.5), suivantes :

$$f(x + \alpha_k d_k) \leq f(x_k) + \rho \alpha_k \nabla f(x_k)^T d_k \tag{0.6}$$

$$\nabla f(x + \alpha_k d_k)^T . d_k \geq \sigma \nabla f(x_k)^T d_k \tag{0.7}$$

Dans ce cas, le principal résultat obtenu est celui de Dai et Yuan ([24]).

Si on considère les variantes Polak-Ribière-Polyak (PRP) et Hestenes et Steifel (HS) non linéaire du gradient conjugué, peu de résultats de convergence complets ont été établis. Grippo et Lucidi ([41]) et P. Armand (2005) ont suggéré des modifications dans le choix de α_k afin d'établir le résultat de la convergence.

La question naturelle qui se pose est la suivante :

Peut on obtenir des résultats de convergence de la variante du Gradient Conjugué de Hestenes et Steifel(HS) en utilisant une modification sur la recherche linéaire inexacte d'armijo ?

Une première étude a été établie par **Shi, Z.J et Shen, J. (2007, [68])** dont le but a été d'étudier la convergence de la méthode de Polak-Ribière-Polyak **(PRP). Ils ont** modifié la recherche linéaire d'Armijo (α_k vérifiant (0.6)) pour assurer la convergence de la méthode **PRP.**

Dans cette thèse, en s'inspirant du travail de **Shi, Z.J et Shen, J. (2007), on** propose une nouvelle modification de la recherche linéaire inexacte d'Armijo qui nous assure la convergence globale de la méthode du gradient conjugué, version Hestenes et Steifel : $\beta_k^{HS} = \frac{g_k^T y_k}{d_{k-1}^T y_{k-1}}$.

Cette thèse comporte une introduction, cinq chapitres. et une conclusion avec des perspectives. On introduit dans Le premier chapitre les notions préléminaires et de base. On aborde dans ce chapitre quelques notions sur les problèmes de minimisation sans contraintes et leurs algorithmes.

Le deuxième chapitre introduit une classe importante d'algorithmes de résolution des problèmes d'optimisation sans contraintes. Le concept central est celui de direction de descente. On expose aussi dans ce chapitre les principales règles de recherches linéaires.

Le troisième chapitre est consacré à l'étude de la *contribution* de la recherche linéaire inexacte sur 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.

Le chapitre 4 est une synthèse des différents résultats de convergence des différentes variantes de la méthode du gradient conjugué avec la recherche linéaire inexacte de Wolfe forte et Wolfe faible.

Le dernier chapitre contient le résultat original de cette thèse. On introduit une nouvelle

modification de la recherche linéaire inexacte d'Armijo et on l'applique à la méthode du gradient conjugué version Hestenes et Steifel : $\beta_k^{HS} = \frac{g_k^\intercal y_k}{d_{k-1}^\intercal y_{k-1}}$. Avec cette modification la propriété de descente est assurée à chaque itération et certains résultats généraux de convergence sont montrés. Les résultats numériques montrent que cette méthode est plus performante que d'autres méthodes de gradient conjugué assez connues. Le manuscrit se termine par une conclusion générale et quelques perspectives.

Chapitre I

Notions de base et résultats préliminaires

Dans ce chapitre, nous allons introduire certaines notions et résultats de calcul matriciel ainsi que des notions de base de l'analyse convexe et la programmation mathématique qui seront utile par la suite. On terminera ce chapitre par quelques généralités sur les problèmes de minimisation sans contraintes et leurs algorithmes. Des propriétés concernant les matrices dans ce chapitre, sont classiques. On pourra consulter l'ouvrage de Horn et Johnson [50].

1 Calcul matriciel

Définition 1.1 • *Soit A une matrice carrée d'ordre n. S'il existe une matrice carrée B d'ordre n telle que $AB = BA = I$, alors B est appelée la matrice inverse de A et est notée A^{-1}.*

• *Une matrice carrée A est dite symétrique si $A = A^t$.*

• *Le rang d'une matrice A est donné par le nombre de lignes ou de colonnes linéairement indépendantes de A.*

• *Soit A une matrice carrée, le polynôme caractéristique de A est donné par :*

$p(\lambda) = |A - \lambda I|$, *$p(\lambda)$ est un polynôme de degré n donc possèdant au plus n racines réelles distinctes.*

• *Une matrice A est singuliére si et seulement si $det\,(A) = 0$, est dite A non singulière si $det\,(A) \neq 0$.*

• *Les valeurs propres de la matrice A sont les racines de $p(.)$; i,e., les valeurs λ_i telles que $p(\lambda_i) = 0$.*

• *Une matrice Q est dite orthogonale si $Q^{-1} = Q^t$.*

1.1 Indépendance linéaire

Soit $\left\{v^{(1)}, v^{(2)}, ..., v^{(k)}\right\}$ un ensemble de vecteurs dans $\mathbb{R}^n$. On dit que cet ensemble est linéairement indépendants si : $\alpha_1 v^{(1)} + \alpha_2 v^{(2)} + ... + \alpha_k v^{(k)} = 0 \Rightarrow \alpha_1 = \alpha_2 = ... = \alpha_k = 0$.

1.2 Produit scalaire et normes vectorielles

Le produit scalaire de deux vecteurs x et y de $\mathbb{R}^n$ est

$$x^t y = \sum_{i=1}^{i=n} x_i y_i$$

Définition 1.1.2

Une norme d'un espace vectoriel E est une application $\| \ . \ \|$ de E dans $\mathbb{R}^+$ qui vérifie les propriétés suivantes :

$$\forall x \in E, \| \ x \ \| \geq 0$$

$$\|x\| = 0 \Leftrightarrow x = 0$$

$$\forall \lambda \in \mathbb{R}, x \in E, \|\lambda x\| = \mid \lambda \mid \| \ x \ \|$$

$$\forall x, y \in E, \| \ x + y \ \| \leq \| \ x \ \| + \| \ y \ \|$$

Dans $\mathbb{R}^n$, les trois normes les plus courantes sont la norme infinie, la norme 1 et la norme euclidienne.

- *norme infinie* :

$$\|x\|_\infty = Max_{1 \leq i \leq n} \mid x_i \mid$$

- *norme 1 :*

$$\| \ x \ \|_1 = \sum_{i=1}^{i=n} \mid x_i \mid$$

- *norme 2 ou norme euclidienne* **:**

$$\| \ x \ \|_2 = \sqrt{x^t x} = (\sum_{i=1}^{i=n} x_i^2)^{\frac{1}{2}}$$

ou $x = (x_1, ..., x_n)^t$.

La norme euclidienne est donc définie par le produit scalaire $x^t y$.

1.3 Normes matricielles

Définition 1.2 *Soit $\| \ . \ \|$ une norme quelconque sur $\mathbb{R}^n$, on appelle norme matricielle subordonnée de la matrice carrée A d'ordre n*

$$\| A\| = \max_{x\in\mathbb{R}^n, \|x\|\neq 0} \frac{\| Ax \|}{\| x\|} = \max_{x\in\mathbb{R}^n, \|x\|=1} \| Ax \|$$

1.4 Orthogonalité dans $\mathbb{R}^n$

Définition 1.3

$$x \perp y \Longleftrightarrow x^t y = 0$$

Théorème 1.1 *Toute matrice symétrique a des valeurs propres réelles et des vecteurs propres réels. Elle est diagonalisable. De plus, les vecteurs propres forment une base orthonormée.*

Théorème 1.2 *Si A est une matrice définie positive d'ordre n, alors :*

1) A est non singulière ; i.e, A^{-1} existe

2) $A_{ii} > 0$, pour $i = 1, ..., n$.

Théorème 1.3 *Si A est une matrice symétrique et D est une matrice diagonale contenant les valeurs propres de A ; alors il existe une matrice orthogonale Q telle que $D = Q^{-1}AQ = Q^tAQ$.*

2 Analyse convexe

2.1 Ensembles et fonctions convexes

Définition 2.1 *Soient E un espace vectoriel de dimension finie sur $\mathbb{R}$ et $x, y \in E$. On appelle segment de E, un ensemble noté et défini comme suit :*

$$[x; y] := \{(1 - t)x + ty : t \in [0; 1]\}$$

on dit qu'une partie C de E est convexe si pour tout $x, y \in C$, le segment $[x; y]$ est contenu dans C.

Proposition 2.1 *1) Si C_1,...,C_n sont des espaces convexes alors l'intersection $\cap C_i$ est convexe.*

2) Pour $i = 1, ..., k$ soient $C_i \in \mathbb{R}^{ni}$ des ensembles convexes alors $C = C_1 \times ... \times C_k$ est un ensemble convexe de $\mathbb{R}^{n_1\times...\times n_k}$.

3) Soit $C \subset \mathbb{R}^n$ un ensemble convexe et $f : \mathbb{R}^n \to \mathbb{R}^m$ une application affine, alors $f(C) = \{f(x) / x \in C\}$ est convexe.

Définition 2.2

$$f : \Omega \to \mathbb{R}, \Omega \subset \mathbb{R}^n$$

est convexe sur Ω *si son épigraphe est un ensemble convexe.*

Définition 2.3 *Soit* $\Omega \subset \mathbb{R}^n$ *un ensemble convexe non vide.Une fonction* $f : \Omega \subset \mathbb{R}^n \to \mathbb{R}$ *est une fonction convexe si et seulement si pour tout* $x, y \in \Omega$*, on a :*

$$f(\alpha x + (1-\alpha)y) \leq \alpha f(x) + (1-\alpha)f(y), \text{ pour tout } \alpha \in [0,1]$$

$\diamondsuit$ *On dit que* $f : \Omega \subset \mathbb{R}^n \to \mathbb{R}$ *est strictement convexe si pour tout* $x_1,\ x_2 \in \Omega$ *avec* $x_1 \neq x_2$*, on a :*

$$f[tx_1 + (1-t)x_2] < tf(x_1) + (1-t)f(x_2), \text{ pour tout } t \in]0,1[$$

$\diamondsuit$ *On dit que* $f : \Omega \subset \mathbb{R}^n \to \mathbb{R}$ *est fortement convexe ou uniformément convexe de module* $\delta > 0$ *(*δ*-convexe), si pour tout* $x,\ y \in \Omega$*, on a :*

$$f[(1-t)x + ty] \leq (1-t)f(x) + tf(y) - \frac{\delta}{2}t(1-t)\left\|x-y\right\|^2, \text{ pour tout } t \in]0,1[$$

$\diamondsuit$ *On dit que* $f : \Omega \subset \mathbb{R}^n \to \mathbb{R}$ *est quasi-convexe, si pour tout* $x, y \in \Omega$*, on a :*

$$f[(1-t)x + ty] \leq \max[f(x), f(y)], \text{ pour tout } t \in]0,1[$$

$\diamondsuit$ *On dit que* $f : \Omega \subset \mathbb{R}^n \to \mathbb{R}$ *est strictement quasi-convexe, si pour tout* $x, y \in \Omega$*, avec* $f(x) \neq f(y)$*, on a :*

$$f[(1-t)x + ty] < \max[f(x), f(y)], \text{ pour tout } t \in]0,1[$$

3 Optimisation sans contraintes

Dans ce paragraphe, on définit et on introduit les outils fonctionnels de base nécessaires pour l'optimisation sans contraintes.

3.1 Différentiabilité

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

Soit Ω un ouvert de $\mathbb{R}^n$.

3.1.1 Gradient, Hessien

Définition 3.1 • *On note par*

$$(\nabla f(x))^T = \left(\frac{\partial f}{\partial x_1}, ..., \frac{\partial f}{\partial x_n}\right)(x)$$

le gradient de f au point $x = (x_1, .., x_n)$.

• *On appelle Hessien de f la matrice symétrique de $M_n(\mathbb{R})$*

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

Définition 3.2 • *On dit que x^* est un point stationnaire de f si $\nabla f(x^*) = 0$.*

• *Soit $\mathcal{M}_n$ l'anneau des matrices carrées $n \times n$ dans $\mathbb{R}$. Une matrice $A \in \mathcal{M}_n$ est dite semi-définie positive si*

$$\forall x \in \mathbb{R}^n : x^\top Ax \geq 0$$

elle est dite définie positive si

$$\forall x \in \mathbb{R}^n_* : x^\top Ax > 0$$

3.1.2 Dérivée directionnelle

Définition 3.3 ([7,p.8]) *On appelle dérivée directionnelle de f dans la direction d au point x, notée $\delta f(x, d)$, la limite (éventuellement $\pm\infty$) du rapport :*

$$\frac{f(x + hd) - f(x)}{h} \quad \textit{lorsque } h \textit{ tend vers } 0.$$

Autrement dit :

$$\delta f(x, d) = \lim_{h \to 0} \frac{f(x + hd) - f(x)}{h} = \nabla^T f(x) d$$

• *Si $\|d\| = 1$: la dérivée directionnelle est le taux d'accroissement de f dans la direction d au point x.*

Remarque 3.1 • *Le taux d'accroissement est maximal dans la direction du gradient*

• *Le gradient indique la direction de la plus grande pente.*

Définition 3.4 *Soit $f : \mathbb{R}^n \to \mathbb{R}.y = f(x)$ définit une surface dans $\mathbb{R}^{n+1}$. $f(x) = r$, avec r constant, définissent des courbes de niveau ou des contours de cette surface.*

Un contour de niveau r est l'ensemble des points $R(r) = \{x/\ f(x) = r\}$

Propriété 1.1.1

Soit $f : \mathbb{R}^n \to \mathbb{R}$ et ∇f son gradient, R le contour, alors le gradient ∇f est orthogonal au contour.

Démonstration :

Soit le contour R de niveau $f(x_0)$ passant par x_0 : $R = \{x/\ f(x) = f(x_0)\}$

Soit une courbure dans R paramétrée par $v(t) : \mathbb{R}^n \to \mathbb{R}$ avec $v(t_0) = x_0$ et $v(t) = x \in R$.

La tangente à cette courbe en x_0 est de direction

$$\frac{dv}{dt/_{t=t_0}}$$

Nous avons :

$$\begin{aligned} f(v(t)) &= f(v(t_0)) = f(x_0) \\ &\Rightarrow \frac{d}{dt} f(v(t))/_{t=t_0} = 0 \\ &\iff \frac{\partial f}{\partial x_1}(v(t_0))\frac{dv_1}{dt}/_{t=t_0} + ... + \frac{\partial f}{\partial x_n}(v(t_0))\frac{dv_n}{dt}/_{t=t_0} = 0 \\ &\iff \nabla^T f(x_0)\frac{dv}{dt}/_{t=t_0} = 0 \end{aligned}$$

Ce qui veut dire que $\nabla f(x_0)$ est orthogonal à R en x_0, $\forall x_0$ $\square$

3.2 Conditions d'optimalité des problèmes d'optimisation sans contraintes

Définition 3.5 *Considérons le problème de minimisation sans contraintes (P).*

a) $x_* \in \mathbb{R}^n$ s'appelle minimum global du problème (P) si

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

b) x_* est un minimum local de (P) s'il existe un voisinage $V_\varepsilon(x_*)$ de x_* tel que

$$f(x_*) \leq f(x),\ \forall x \in V_\varepsilon(x_*).$$

c) x_* est minimum local strict s'il existe un voisinage $V_\varepsilon(x_*)$ de x_* tel que

$$f(x_*) < f(x),\ \forall x \in V_\varepsilon(x_*) \quad et \quad x \neq x_*. \quad \square$$

3.2.1 Conditions nécessaires d'optimalité

Théorème 3.1 ([7]) *Soit $f : \mathbb{R}^n \to \mathbb{R}$ telle que f soit différentiable au point $\overline{x} \in \mathbb{R}^n$. Soit $d \in \mathbb{R}^n$ telle que $\nabla f(\overline{x})^t d < 0$. Alors il existe $\delta > 0$ tel que $f(\overline{x}+\alpha d) < f(\overline{x})$ pour tout $\alpha \in]0, \delta[$. La direction d s'appelle dans ce cas direction de descente.*

Preuve. Comme f est différentiable en $\overline{x}$ alors

$$f(\overline{x} + \alpha d) = f(\overline{x}) + \alpha \nabla f(\overline{x})^t d + \alpha \|d\| \lambda(\overline{x}; \alpha d)$$

où $\lambda(\overline{x}; \alpha d) \to 0$ pour $\alpha \to 0$. Ceci implique :

$$\frac{f(\overline{x} + \alpha d) - f(\overline{x})}{\alpha} = \nabla f(\overline{x})^t d + \|d\| \lambda(\overline{x}; \alpha d), \ \alpha \neq 0$$

et comme $\nabla f(\overline{x})^t d < 0$ et $\lambda(\overline{x}; \alpha d) \to 0$ pour $\alpha \to 0$, il existe $\delta > 0$ tel que

$$\nabla f(\overline{x})^t d + \|d\| \lambda(\overline{x}; \alpha d) < 0 \text{ pour tout } \alpha \in]0, \delta[$$

et par conséquent on obtient :

$$f(\overline{x} + \alpha d) < f(\overline{x}) \text{ pour tout } \alpha \in]0, \delta[. \quad \square$$

■

Corollaire 3.1 *Soit $f : \mathbb{R}^n \to \mathbb{R}$ différentiable en $\overline{x}$, si $\overline{x}$ est un minimum local alors $\nabla f(\overline{x}) = 0$.*

Preuve. Par contre, on suppose que $\nabla f(\overline{x}) \neq 0$. Si on pose $d = -\nabla f(\overline{x})$, on obtient :

$$\nabla f(\overline{x})^t d = - \|\nabla f(\overline{x})\|^2 < 0$$

et par le théorème précédent, il existe $\delta > 0$ tel que :

$$f(\overline{x} + \alpha d) < f(\overline{x}) \text{ pour tout } \alpha \in]0, \delta[$$

mais ceci est contradictoire avec le fait que $\overline{x}$ est un minimum local, d'où $\nabla f(\overline{x}) = 0$.

■

Théorème 3.2 ([28]) *Soit $f : \mathbb{R}^n \to \mathbb{R}$ deux fois différentiable en $\overline{x}$. Supposons que $\overline{x}$ soit minmum local. Alors $\nabla f(\overline{x}) = 0$ et $H(\overline{x})$ est semi définie positive.*

Preuve. Le corollaire ci-dessus montre la pemière proposition, pour la deuxième proposition on a :

$$f(\overline{x} + \alpha d) = f(\overline{x}) + \alpha \nabla f(\overline{x})^t d + \frac{1}{2}\alpha^2 d^t H(\overline{x}) d + \alpha^2 \|d\|^2 \lambda(\overline{x}; \alpha d)$$

où $\lambda(\overline{x}; \alpha d) \to 0$ pour $\alpha \to 0$. Ceci implique :

$$\frac{f(\overline{x} + \alpha d) - f(\overline{x})}{\alpha^2} = \frac{1}{2} d^t H(\overline{x}) d + \|d\|^2 \lambda(\overline{x}; \alpha d), \ \alpha \neq 0.$$

Comme $\overline{x}$ est un minimum local alors $f(\overline{x} + \alpha d) \geq f(\overline{x})$ pour α suffisamment petit, d'où

$$\frac{1}{2} d^t H(\overline{x}) d + \|d\|^2 \lambda(\overline{x}; \alpha d) \ \geq 0 \text{ pour } \alpha \text{ petit.}$$

En passant à la limite qund $\alpha \to 0$, on obtient que $d^t H(\overline{x}) d \geq 0$, d'où $H(\overline{x})$ est semi définie positive. ■

3.2.2 Conditions suffisantes d'optimalité

Théorème 3.3 ([28]) *Soit $f : \mathbb{R}^n \to \mathbb{R}$ deux fois différentiable en $\overline{x}$. Si $\nabla f(\overline{x}) = 0$ et $H(\overline{x})$ est définie positive, alors $\overline{x}$ est un minmum local strict.*

Preuve. f est deux fois différentiable au point $\overline{x}$. Alors $\forall x \in \mathbb{R}^n$, on obtient :

$$f(x) = f(\overline{x}) + \nabla f(\overline{x})^t (x - \overline{x}) + \frac{1}{2}(x - \overline{x})^t H(\overline{x})(x - \overline{x}) + \|x - \overline{x}\|^2 \lambda(\overline{x}; x - \overline{x}) \qquad (1.1)$$

où $\lambda(\overline{x}; x - \overline{x}) \to 0$ pour $x \to \overline{x}$. Supposons que $\overline{x}$ n'est pas un minimum local strict. Donc il existe une suite $\{x_k\}$ convergente vers $\overline{x}$ telle que

$f(x_k) \leq f(\overline{x}),\ x_k \neq \overline{x},\ \forall k$. Posons $d_k = (x_k - \overline{x}) / \|x_k - \overline{x}\|$. Donc $\|d_k\| = 1$ et on obtient à partir de (1.1) :

$$\frac{f(x_k) - f(\overline{x})}{\|x_k - \overline{x}\|^2} = \frac{1}{2} d_k^t H(\overline{x}) d_k + \lambda(\overline{x}; x_k - \overline{x}) \leq 0, \ \forall k \qquad (*)$$

et comme $\|d_k\| = 1,\ \forall k$ alors $\exists \{d_k\}_{k \in N_1 \subset \mathbb{N}}$ telle que $d_k \to d$ pour $k \to \infty$ et $k \in N_1$. On a bein sûr $\|d\| = 1$. Considérons donc $\{d_k\}_{k \in N_1}$ et le fait que $\lambda(\overline{x}; x - \overline{x}) \to 0$ pour $k \to \infty$ et $k \in N_1$. Alors $(*)$ donne : $d^t H(\overline{x}) d \leq 0$, ce qui conterdit le fait que $H(\overline{x})$ est définie positive car $\|d\| = 1$ (donc $d \neq 0$). Donc $\overline{x}$ est un minimum local strict. ■

Cas convexe

Théorème 3.4 ([07]) *Soit $f : \mathbb{R}^n \to \mathbb{R}$ telle que f est convexe et différentiable. Alors x_* est un minimum global de f si et seulement si $\nabla f(x_*) = 0$.*

Théorème 3.5 ([07]) *Soit $f : \mathbb{R}^n \to \mathbb{R}$ telle que f est pseudo convexe. Soit $x_* \in \mathbb{R}^n$ tel que $\nabla f(x_*) = 0$, alors x_* est un minimum global de f.*

Remarque 3.2 *Les théorèmes* 1.4 *et* 1.5 *demeurent vrais si on remplace $\mathbb{R}^n$ par un ouvert S de $\mathbb{R}^n$.*

Remarque 3.3 *Dans le cas où f est convexe, alors tout minimum local est aussi global. De plus si f est strictement convexe, alors tout minimum local devient non seulement global mais aussi unique ([07]).*

3.3 Les problèmes de minimisation sans contraintes et leurs algorithmes

Soit $f : \mathbb{R}^n \to \mathbb{R}$. On appelle problème de minimisation sans contraintes le problème suivant :

$$\text{Minimiser} \{ f(x) : x \in \mathbb{R}^n \}. \qquad (P)$$

L'étude de ces problèmes est importante pour des raisons diverses. Beaucoup de problèmes d'optimisation avec contraintes sont transformés en des suites de problèmes d'optimisation sans contraintes (multiplicateur de Lagrange, méthodes des penalités, ...). L'étude des problèmes d'optimisation sans contraintes trouve aussi des applications dans la résoulution des systèmes non linéaires.

Les méthodes numériques de résolution de divers problèmes de minimisation sans contraintes ont pris ces dernières années un bel essor si bien que la bibliographie correspondante contient des centaines d'ouvrages et d'articles. Cet intérêt n'est nullement fortuit, il reflète le rôle de premier plan que les problèmes d'optimisation jouent dans les applications.

La construction d'algorithmes consacrés à la recherche efficace de minimum d'une fonction sans contraintes est un problème complexe, car il ne suffit pas d'élaborer un algorithme il faut montrer de plus qu'il emporte sur ceux connus.

On compare des algorithmes en se basant sur des plusieurs critères, par exemple, la précision du résultat, le nombre d'évalutions fonctionnelles et celui d'évaluations du gradient, la vitesse de la convergence, le temps de calcul, l'occupation de la mémoire nécessaire, ...

Même en se fixant des critères de comparaison, on ne peut pas classer les algorithmes ni en indiquant lequel est le meilleur ou le pire. Le fait est qu'on obtient les estimations de l'un des critères précédents pour des classes des problèmes, et un algorithme mauvais pour une vaste classe peut s'averer efficace pour une autre, plus restreinte. L'utilisateur doit posseder tout un arsenal d'algorithmes pour être en mesure de faire face à chaque problème posé.

3.3.1 Aspect général des algorithmes

Pour construire des algorithmes de minimisation sans contraintes on fait appel à des procesuss itératifs du type

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

où d_k détermine la direction de déplacement à partir du point x_k et α_k est un facteur numérique dont le grandeur donne la longueur du pas dans la direction d_k.

Le type d'algorithme permettant de résoudre le problème (P) sera déterminé dés qu'on définit les procédes de construction du vecteur d_k et de calcul de α_k à chaque itération.

La façon avec laquelle on construit les vecteurs d_k et les scalaires α_k détermine directement les propriètés du procesuss et spécialement en ce qui concerne la convergence de la suite $\{x_k\}$, la vitesse de la convergence,....

Pour s'approcher de la solution optimale du problème (P) (dans le cas général, c'est un point en lequel ont lieu peut être avec une certaine précision les conditions nécessaires d'optimalité de f), on se déplace naturellement à partir du point x_k dans la direction de la décroissance de la fonction f.

3.3.2 Fonctions multivoques et algorithmes

Une application multivoque est une application A qui à $x \in \mathbb{R}^n$ fait correspondre un sous ensemble $A(x)$ de $\mathbb{R}^n$.

Etant donné un point x_k. En appliquant les instructions d'un certain algorithme, on obtient un nouveau point x_{k+1}. Cette procédure peut être décrite par une application multivoque A applée application algorithmique. Donc étant donné un point $x_1 \in \mathbb{R}^n$, l'application algorithmique génère une suite $x_1, x_2, ..., $ où $x_{k+1} \in \mathbb{R}^n$ pour tout k.

La notion d'application algorithmique fermé est directement liée à la convergence des algorithmes

Définition 3.6 *Soient X et Y deux sous ensembles fermés non vides de $\mathbb{R}^n$ et $\mathbb{R}^m$ respectivement et $A : X \to Y$ une application multivoque. A est dite fermée au point $x \in \mathbb{R}^n$ si*

$$\left.\begin{array}{l} x_k \in X,\ x_k \to x \\ y_k \in Y,\ y_k \to y \end{array}\right\} \textit{implique que } y \in A(x).$$

Donnons un théorème de convergence qui utilise les fonctions multivoques et qui nous sera utile par la suite. Avant celà, notons qu'à cause de la non convexité, de la taille du problème et d'autres difficultés, on peut arrêter le procesuss itératif si on trouve un point appartient à un ensemble spécifique qu'on appelle ensemble des solutions Ω.

Voici ci-dessous quelques exemples typiques de cet ensemble.

$\Omega = \{x_* : \nabla f(x_*) = 0\}$

$\Omega = \{x_* : x_*$ est une solution optimale locale du problème $(P)\}$

$\Omega = \{x_* : f(x_*) < \nu_* + \varepsilon\ \}$ où $\varepsilon > 0$ est une tolérance définie à l'avance et ν_* est la valeur minimale de la fonction objective.

Nous dirons que l'application algorithmique $A : X \to X$ converge dans $Y \subseteq X$ si commençant par n'importe quel point initial $x_1 \in Y$, la limite de toute sous suite convergente, extraite de la suite $x_1, x_2, ...$, générée par l'algorithme, appartient à Ω.

Théorème 3.6 ([7]) *Soient X un ensemble non vide fermé dans $\mathbb{R}^n$ et $\Omega \subseteq X$ un ensemble des solutions non vide. Soit $\alpha : \mathbb{R}^n \to \mathbb{R}$ une fonction continue, on considère que l'application algorithmique $C : X \to X$ vérifie la propriété suivante : "étant donné $x \in X$ alors $\alpha(y) \leq \alpha(x)$ pour $y \in C(x)$". Soit $B : X \to X$ une application algorithmique fermée sur le complémentaire de Ω qui vérifie :*

$\alpha(y) < \alpha(x)$ pour $y \in B(x)$ et $x \notin \Omega$. Maintenant considérons l'application algorithmique composé $A = CB$. Soit $x_1 \in X$, la suite $\{x_k\}$ est générée comme suit :

Si $x_k \in \Omega$, stop ; sinon poser $x_{k+1} \in A(x_k)$, remplacer k par $(k+1)$ et répéter.

Supposons que $\Lambda = \{x \in X,\ \alpha(x) \leq \alpha(x_1)\}$ est compact. Alors, ou bien l'algorithme s'arrête à un nombre fini d'itérations par un point $\in \Omega$, ou tous les points d'accumulation de $\{x_k\}$ appartenants à Ω.

3.4 Convergence globale des algorithmes

Définition 3.7 *Nous dirons qu'un algorithme décrit par une application multivoque A, est globalement convergent (ou encore : possède la propriété de convergence globale) si, quel que soit le point de départ x_1 choisi, la suite $\{x_k\}$ engendrée par $x_{k+1} \in A(x_k)$ (ou une sous suite) converge vers un point satisfaisant les conditions nécessaires d'optimalité.*

3.5 Modes et 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 ε près (ε 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.*

Nous introduisons ici les principales définitions de base qui seront abondamment utilisées par la suite.

Plaçons nous dans $\mathbb{R}^n$, où $\|.\|$ désigne la norme euclidienne et considérons une suite $\{x_k\}$ convergeant vers x^*.

·Si $\limsup \dfrac{\|x_{k+1-}x^*\|}{\|x_k - x^*\|} = \lambda < 1$

On dit que la convergence est *linéaire* et λ est le taux *de convergence* associé.

·Si $\dfrac{\|x_{k+1-}x^*\|}{\|x_k - x^*\|} \longrightarrow 0$ quand $k \longrightarrow \infty$

on dit que la convergence est *superlinéaire.*

Plus précisément si $\exists p > 1$ tel que :

$$\limsup_{k\longrightarrow\infty} \frac{\|x_{k+1} - x^*\|}{\|x_k - x^*\|^p} < +\infty$$

on dit que la convergence est *superlinéaire* d'ordre p.

En particulier si

$$\limsup_{k\longrightarrow\infty} \frac{\left\|x^{k+1} - x^*\right\|}{\|x^k - x^*\|^2} < +\infty$$

on dit que la convergence est *quadratique* (superlinéaire d'ordre 2)

Chapitre II

Optimisation unidimensionnelle ou recherche linéaire

Faire de la recherche linéaire veut dire déterminer un pas α_k le long d'une direction de descente d_k, autrement dit résoudre le problème unidimensionnel :

$$\min f(x_k + \alpha d_k), \qquad \alpha \in \mathbb{R} \tag{P}$$

Notre intérêt pour la recherche linéaire ne vient pas seulement du fait que dans les applications on rencontre, naturellement, des problèmes unidimensionnels, mais plutôt du fait que la recherche linéaire est un composant fondamental de toutes les méthodes traditionnelles d'optimisation multidimensionnelle. D'habitude, nous avons le schéma suivant d'une méthode de minimisation sans contraintes multidimensionnelle :

En regardant le comportement local de l'objectif f sur l'itération courante x_k, la méthode choisit la "direction du mouvement" d_k (qui, normalement, est une direction de descente de l'objectif : $\nabla^T f(x).d < 0$) et exécute un pas dans cette direction :

$$x_k \longmapsto x_{k+1} = x_k + \alpha_k d_k \tag{*}$$

Afin de réaliser un certain progrès en valeur de l'objective, c'est-à-dire, pour assurer que : $f(x_k + \alpha_k d_k) \leq f(x_k)$

Et dans la majorité des méthodes le pas dans la direction d_k est choisi par la minimisation unidimensionnelle de la fonction :

$$h_k(\alpha) = f(x_k + \alpha d_k).$$

Ainsi, la technique de recherche linéaire est une brick de base fondamentale de toute méthode

multidimensionnelle.

1 Objectifs à atteindre

Il s'agit de réaliser deux objectifs

1.1 Le premier objectif

Consiste à *faire décroître f suffisamment*. Cela se traduit le plus souvent par la réalisation d'une inégalité de la forme

$$f(x_k + \alpha_k d_k) \leq f(x_k) + \text{"un terme négatif "} \tag{2.1}$$

Le terme négatif, disons ν_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 (il existe une constante C telle que $f(x_k) \geq C$ pour tout k), alors ce terme négatif tend nécessairement vers zéro : $\nu_k \to 0$. 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 $f(x_k + \alpha_k d_k) < f(x_k)$.

1.2 Le second objectif

Consiste d'empêcher le pas $\alpha_k > 0$ d'être trop petit, trop proche de zéro.

Le premier objectif n'est en effet pas suffisant car l'inégalité (2.1) est en général satisfaite par des pas $\alpha_k > 0$ arbitrairement petit.

Or ceci peut entraîner une "*fausse convergence*", c'est-à-dire la convergence des itérés vers un point non stationnaire, comme le montre l'observation suivante ([40]).

Si on prend

$$0 < \alpha_k \leq \frac{\varepsilon}{2^k \|d_k\|}$$

la suite $\{x_k\}$ générée par (*) est de Cauchy, puisque pour $1 \leq l < k$ on a :

$$\|x_k - x_l\| = \left\| \sum_{i=1}^{i=k-1} \alpha_i d_i \right\| \leq \sum_{i=1}^{i=k-1} \frac{\varepsilon}{2^i} \to 0, \text{ lorsque : } l \to \infty.$$

Donc $\{x_k\}$ converge, disons vers un point $\bar{x}$. En prenant $l = 1$ et $k \to \infty$ dans l'estimation ci-dessus, on voit que $\bar{x} \in \bar{B}(x_1, \varepsilon)$ et donc $\bar{x}$ ne saurait être solution s'il n'y a pas de solution dans $\bar{B}(x_1, \varepsilon)$.

On a donc arbitrairement forcer la convergence de $\{x_k\}$ en prenant des pas très petits.

Pour simplifier les notations, on définit la restriction de f à la droite $\{x_k + \alpha d_k / \alpha \in \mathbb{R}\} \subset \mathbb{R}^n$: comme la fonction : $h_k : \mathbb{R}_+ \to \mathbb{R}$;

$$\alpha \longmapsto h_k(\alpha) = f(x_k + \alpha d_k)$$

2 Recherches linéaires exactes et inexactes

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

2.1 Recherches linéaires exactes

Comme on cherche à minimiser f, il semble naturel de chercher a minimiser le critère le long de d_k et donc de déterminer le pas α_k comme solution du problème

$$\min\ h_k(\alpha) \quad \alpha \geq 0$$

C'est ce que l'on appelle la règle de Cauchy et le pas déterminé par cette règle est appelé pas de Cauchy ou pas optimal.

Dans certains cas, on préférera le plus petit point stationnaire de h_k qui fait décroître cette fonction :

$$\alpha_k = \inf \{\alpha \geq 0 : h_k(\alpha) < h_k(0)\}.$$

On parle alors de règle de Curry et le pas déterminé par cette règle est appelé pas de Curry. De manière un peu imprécise, ces deux règles sont parfois qualifiées de recherche linéaire exacte.

Remarque 2.1 *Ces deux règles ne sont utilisées que dans des cas particuliers, par exemple lorsque h_k est quadratique.*

Le mot exact prend sa signification dans le fait que si f est quadratique la solution de la recherche linéaire s'obtient de façon exacte et dans un nombre fini d'itérations.

2.1.1 Les inconvénients des recherches linéaires exactes

Pour une fonction non linéaire arbitraire,

- il peut ne pas exister de pas de Cauchy ou de Curry,

- la détermination de ces pas demande en général beaucoup de temps de calcul et ne peut de toutes façons pas être faite avec une précision infinie,

- l'efficacité supplémentaire éventuellement apportée à un algorithme par une recherche linéaire exacte ne permet pas, en général, de compenser le temps perdu à déterminer un tel pas,

- les résultats de convergence autorisent d'autres types de règles (recherche linéaire inexacte), moins gourmandes en temps de calcul.

2.1.2 Intervalle de sécurité

Dans la plupart des algorithmes d'optimisation modernes, on ne fait jamais de recherche linéaire exacte, car trouver α_k signifie qu'il va falloir calculer un grand nombre de fois la fonction h_k et cela peut être dissuasif du point de vue du temps de calcul.

En pratique, on recherche plutôt une valeur de α^* qui assure une décroissance suffisante de f.

Cela conduit à la notion d'intervalle de sécurité.

Définition 2.1 *On dit que $[\alpha_g, \alpha_d]$ est un intervalle de sécurité s'il permet de classer les valeurs de α de la façon suivante :*

- *Si $\alpha < \alpha_g$ alors α est considéré trop petit,*
- *Si $\alpha_d \geq \alpha \geq \alpha_g$ alors α est satisfaisant,*
- *Si $\alpha > \alpha_d$ alors est considéré trop grand.*

Le problème est de traduire de façon numérique sur h_k les trois conditions précédentes, ainsi que de trouver un algorithme permettant de déterminer α_g et α_d.

2.1.3 Algorithme de base

Algorithme 2.1 (Algorithme de base)

Etape 0 : (initialisation)

$\alpha_g = \alpha_d = 0$,choisir $\alpha_1 > 0$, poser $k = 1$ et aller à l'étape 1 ;

Etape 1 :

Si α_k convient, poser $\alpha^* = \alpha_k$ et on s'arrête.

Si α_k est trop petit on prend $\alpha_{g,k+1} = \alpha_k$, $\alpha_d = \alpha_d$
et on va à l'étape 2 .
Si α_k est trop grand on prend $\alpha_{d,k+1} = \alpha_k$, $\alpha_g = \alpha_g$
et on va à l'étape 2 .

Etape 2 :

si $\alpha_{d,k+1} = 0$ déterminer $\alpha_{k+1} \in]\alpha_{g,k+1}, +\infty[$
si $\alpha_{d,k+1} \neq 0$ déterminer $\alpha_{k+1} \in]\alpha_{g,k+1}, \alpha_{d,k+1}[$
remplacer k par $k+1$ et aller à l'étape 1. □

Au lieu de demander que α_k minimise , on préfère imposer des conditions moins restrictives, plus facilement vérifiées, qui permettent toute fois de contribuer à la convergence des algorithmes. En particulier, il n'y aura plus un unique pas (ou quelques pas) vérifiant ces conditions mais tout un intervalle de pas (ou plusieurs intervalles), ce qui rendra d'ailleurs leur recherche plus aisée. C'est ce que l'on fait avec les règles d'Armijo, de Goldstein et de Wolfe décrites dans la prochaine section.

2.2 Recherches linéaires inexactes

On considère la situation qui est typique pour l'application de la technique de recherche linéaire à l'intérieur de la méthode principale multidimensionnelle.

Sur une itération k de la dernière méthode nous avons l'itération courante $x_k \in \mathbb{R}^n$ et la direction de recherche $d_k \in \mathbb{R}^n$ qui est direction de descente pour notre objectif : $f : \mathbb{R}^n \to \mathbb{R}$:

$$\nabla^T f(x_k).d_k < 0 \tag{2.2}$$

Le but est de réduire "de façon importante" la valeur de l'objectif par un pas $x_k \longmapsto x_{k+1} = x_k + \alpha_k d_k$ de x_k dans la direction d_k. Pour cela de nombreux mathématiciens (Armijo, Goldstein, Wolfe, Albaali, Lemaréchal, Fletcher...) ont élaboré plusieurs règles (tests).

L'objectif de cette section consiste à présenter les principaux tests.

D'abord présentons le schéma d'une recherche linéaire inexacte.

2.2.1 Schéma des recherches linéaires inexactes

Elles reviennent à déterminer, par tâtonnement un intervalle $[\alpha_g, \alpha_d]$, où $\alpha^* \in [\alpha_g, \alpha_d]$, dans lequel :

$$h_k(\alpha_k) < h_k(0) \quad (f(x_k + \alpha_k d_k) < f(x_k))$$

Le schéma de l'algorithme est donc :

Algorithme 2.2(Schéma général des recherches linéaires inexactes)

Etape 0 : (initialisation)

$\alpha_{g,1} = \alpha_{d,1} = 0$, choisir $\alpha_1 > 0$, poser $k = 1$ et aller à l'étape 1.

Etape 1 :

si α_k est satisfaisant (suivant un certain critère) : STOP($\alpha^* = \alpha_k$).

si α_k est trop petit (suivant un certain critère) : nouvel intervalle : [$\alpha_{g,k+1} = \alpha_k$, $\alpha_{d,k+1} = \alpha_d$]

et aller à l'étape 2.

siα_k est trop grand (suivant un certain critère) : nouvel intervalle : [$\alpha_{g,k+1} = \alpha_g$, $\alpha_{d,k+1} = \alpha_k$]

et aller à l'étape 2.

Etape 2 :

si $\alpha_{d,k+1} = 0$ déterminer $\alpha_{k+1} \in]\ \alpha_{g,k+1}\ , +\infty\ [$

si $\alpha_{d,k+1} \neq 0$ déterminer $\alpha_{k+1} \in]\ \alpha_{g,k+1}\ , \alpha_{d,k+1}[$

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

Il nous reste donc à décider selon quel(s) critère(s) α est trop petit ou trop grand ou satisfaisant.

2.3 La règle d'Armijo

On bute à réduire "de façon importante" la valeur de l'objectif par un pas $x_k \longmapsto x_{k+1} = x_k + \alpha_k d_k$ de x_k dans la direction d_k, tel que $f(x_k + \alpha_k d_k) < f(x_k)$.

Or cette condition de décroissance stricte n'est pas suffisante pour minimiser h_k au moins localement. Par exemple, avec la fonction $h : \mathbb{R} \to \mathbb{R};\ \ x \mapsto h(x) = x^2$ et $x_1 = 2$, les choix $d_k = (-1)^{k+1}$; $\alpha_k = 2 + 3 \times 2^{-(k+1)}$, donnent : $x_k = (-1)^k(1 + 2^{-k})$.

$h(x_k)$ est bien strictement décroissante mais $\{x_k\}_{k\geq 0}$ ne converge pas vers le minimum zéro mais vers 1 (dans cet exemple le pas est trop grand).

La règle d'Armijo [02] impose une contrainte sur le choix de α_k suffisante pour minimiser localement h.

Une condition naturelle est de demander que f décroisse autant qu'une portion $\rho \in]0, 1[$ de ce que ferait le modèle linéaire de f en x_k. Cela conduit à l'inégalité suivante, parfois appelée *condition d'Armijo* ou *condition de décroissance linéaire* :

$$f(x_k + \alpha_k d_k) \leq f(x_k) + \rho\alpha_k \nabla^T f(x_k) d_k \tag{2.3}$$

Elle est de la forme (2.3), car ρ devra être choisi dans$]0,1[$.

On voit bien à la figure 2.3 ce que signifie cette condition.

Il faut qu'en α_k , la fonction h_k prenne une valeur plus petite que celle prise par la fonction $\psi_\rho(\alpha)$:

$$\alpha \longmapsto h_k(0) + \rho h_k'(0)\alpha \quad \text{autremment dit } \alpha \mapsto f(x_k) + \rho\alpha_k \nabla^T f(x_k) d_k$$

2.3.1 Test d'Armijo :

♢ Si

$$\begin{aligned} h_k(\alpha) &\leq h_k(0) + \rho h_k'(0)\alpha \\ \text{autrement dit } f(x_k + \alpha d_k) &\leq f(x_k) + \rho\alpha \nabla^T f(x_k) d_k \end{aligned}$$

alors α convient.

♢ Si

$$\begin{aligned} h_k(\alpha) &> h_k(0) + \rho h_k'(0)\alpha \\ \text{autrement dit } f(x_k + \alpha d_k) &> f(x_k) + \rho\alpha \nabla^T f(x_k) d_k \end{aligned}$$

alors α est trop grand. □

Algorithme 2.3 (Règle d'Armijo)

Etape 0 : (initialisation)

$\alpha_{g,1} = \alpha_{d,1} = 0$, choisir $\alpha_1 > 0$, $\rho \in]0,1[$ poser $k = 1$ et aller à l'étape 1.

Etape 1 :

si $h_k(\alpha_k) \leq h_k(0) + \rho h_k'(0)\alpha_k$: STOP $(\alpha^* = \alpha_k)$.

si $h_k(\alpha_k) > h_k(0) + \rho h_k'(0)\alpha_k$, alors

$\alpha_{d,k+1} = \alpha_d$,$\alpha_{g,k+1} = \alpha_k$ et aller à l'étape 2.

Etape 2 :

si $\alpha_{d,k+1} = 0$ déterminer $\alpha_{k+1} \in]\ \alpha_{g,k+1}\ ,\ +\infty\ [$

si $\alpha_{d,k+1} \neq 0$ déterminer $\alpha_{k+1} \in]\ \alpha_{g,k+1}\ ,\ \alpha_{d,k+1}\ [$

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

Remarque 2.2 *En pratique, la constante ρ est prise très petite, de manière à satisfaire (2.3) le plus facilement possible. Typiquement, $\rho = 10^{-4}$.*

Notons que cette constante ne doit pas être adaptée aux données du problème et donc que l'on ne se trouve pas devant un choix de valeur délicat. □

Remarque 2.3 *Il est clair que l'inégalité (2.3) est toujours vérifiée si $\alpha_k > 0$ est suffisamment petit.*

*Cela se démontre aussi facilement **([40])**.*

En effet dans le cas contraire, on aurait une suite de pas strictement positifs $\{\alpha_{k,i}\}_{i\geq 1}$ convergeant vers 0 lorsque $i \to \infty$ et tels que

(2.3) n'ait pas lieu pour $\alpha_k = \alpha_{k,i}$.

En retranchant $f(x_k)$ dans les deux membres, en divisant par $\alpha_{k,i}$ et en passant à la limite quand $i \to \infty$, on trouverait

$$\nabla^T f(x_k)d_k \geq \rho \nabla^T f(x_k)d_k$$

ce qui contredirait le fait que d_k est une direction de descente ($\rho < 1$). □

Dans le théorème suivant on va assurer l'existence du pas d'Armijo en posant quelques conditions sur la fonction h_k.

Théorème 2.1 ([40]) *Si $h_k : \mathbb{R}_+ \to \mathbb{R}$;définie par $h_k(\alpha) = f(x_k + \alpha d_k)$ est continue et bornée inférieurement, si d_k est une direction de descente en x_k ($h'_k(0) < 0$) et si $\rho \in]0, 1[$, alors l'ensemble des pas vérifiant la règle d'Armijo est non vide.*

Preuve. On a

$$\begin{aligned} h_k(\alpha) &= f(x_k + \alpha d_k) \\ \psi_\rho(\alpha) &= f(x_k) + \rho\alpha_k \nabla^T f(x_k)d_k \end{aligned}$$

Le développement de Taylor-Yong en $\alpha = 0$ de h_k est :

$$h_k(\alpha) = f(x_k + \alpha d_k) = f(x_k) + \rho\alpha\nabla^T f(x_k)d_k + \alpha\xi(\alpha) \text{ où } \xi(\alpha) \to 0, \alpha \to 0.$$

et comme $\rho \in]0, 1[$ et $h'_k(0) = \nabla^T f(x_k)d_k < 0$ on déduit :

$$f(x_k) + \alpha_k \nabla^T f(x_k)d_k < f(x_k) + \rho\alpha_k \nabla^T f(x_k)d_k \text{ pour } \alpha > 0$$

On voit que pour $\alpha > 0$ assez petit on a :

$$h_k(\alpha) < \psi_\rho(\alpha)$$

De ce qui précède et du fait que h_k est bornée inférieurement,

et $\psi_\rho(\alpha) \to -\infty; \alpha \to +\infty$, on déduit que la fonction $\psi_\rho(\alpha) - h_k(\alpha)$ a la propriété :

$$\begin{cases} \psi_\rho(\alpha) - h_k(\alpha) > 0 & \text{pour } \alpha \text{ assez petit} \\ \psi_\rho(\alpha) - h_k(\alpha) < 0 & \text{pour } \alpha \text{ assez grand} \end{cases}$$

donc s'annule au moins une fois pour $\alpha > 0$.

En choisissant le plus petit de ces zéros on voit qu'il existe $\bar{\alpha} > 0$ tel que

$$h_k(\bar{\alpha}) = \psi_\rho(\bar{\alpha}) \text{ et } h_k(\alpha) < \psi_\rho(\alpha) \text{ pour } 0 < \alpha < \bar{\alpha}$$

Ce qui achève la démonstration. ■

Décrivons maintenat la règle de Goldstein&Price.

2.4 La règle de Goldstein&Price.

Dans la règle d'Armijo on assure la décroissance de la fonction objectif à chaque pas, mais c'est ne pas suffisant ; reprenant l'exemple de la fonction $h : \mathbb{R} \to \mathbb{R}; x \mapsto h(x) = x^2$ et $x_1 = 2$, et cette fois $d_k = -1$ le choix $\alpha_k = 2^{-(k+1)}$ donnent : $x_k = (1 + 2^{-k})$

h_k est bien strictement décroissante mais $\{x_k\}_{k\geq 0}$ ne converge pas vers le minimum zéro mais vers 1. Alors que la condition d'Armijo est satisfaite.

Les conditions de Goldstein & Price ([43]) suivant sont, comme on va le prouver,suffisante pour assurer la convergence sous certaines conditions et indépendamment de l'algorithme qui calcule le paramètre .

Etant données deux réels ρ et σ tels que $0 < \rho < \delta < 1$; ces conditions sont :

$$f(x_k + \alpha d_k) \leq f(x_k) + \rho\alpha\nabla^T f(x_k)d_k \tag{2.4}$$

$$f(x_k + \alpha d_k) \geq f(x_k) + \delta\alpha\nabla^T f(x_k)d_k \tag{2.5}$$

autrement dit :

$$h_k(\alpha) \leq h_k(0) + \rho h_k'(0)\alpha$$

$$h_k(\alpha) \geq h_k(0) + \delta h_k'(0)\alpha$$

2.4.1 Test de Goldstein&Price :

♢ Si

$$\begin{aligned} h_k(0)+\delta h_k'(0)\alpha &\leq h_k(\alpha)\leq h_k(0)+\rho h_k'(0)\alpha \\ \text{autrement dit } f(x_k)+\delta\alpha\nabla^T f(x_k)d_k &\leq f(x_k+\alpha d_k)\leq f(x_k)+\rho\alpha\nabla^T f(x_k)d_k \end{aligned}$$

alors α convient.
♢ Si

$$\begin{aligned} h_k(\alpha) &> h_k(0)+\rho h_k'(0)\alpha \\ \text{autrement dit } f(x_k+\alpha d_k) &> f(x_k)+\rho\alpha\nabla^T f(x_k)d_k \end{aligned}$$

alors α est trop grand.
♢ Si

$$\begin{aligned} h_k(\alpha) &< h_k(0)+\delta h_k'(0)\alpha \\ \text{autrement dit } f(x_k+\alpha d_k) &< f(x_k)+\delta\alpha\nabla^T f(x_k)d_k \end{aligned}$$

alors α est trop petit. □
On voit bien à la figure 2.5 ce que signifie cette condition.

Algorithme 2.4 (Règle de Goldstein&Price)

Etape 0 : (initialisation)
$\alpha_{g,1}=\alpha_{d,1}=0$, choisir $\alpha_1>0$, $\rho\in\,]0,1[\,,\delta\in\,]\rho,1[\,$, poser$k=1$ et aller à l'étape 1.
Etape 1 :
si $h_k(0)+\delta h_k'(0)\alpha\leq h_k(\alpha_k)\leq h_k(0)+\rho h_k'(0)\alpha_k$: STOP($\alpha^*=\alpha_k$).
si $h_k(\alpha_k)>h_k(0)+\rho h_k'(0)\alpha_k$, alors
$\alpha_{d,k+1}=\alpha_k$,$\alpha_{g,k+1}=\alpha_{g,k}$ et aller à l'étape 2.
si $h_k(\alpha_k)<h_k(0)+\delta h_k'(0)\alpha_k$, alors
$\alpha_{d,k+1}=\alpha_{d,k}$,$\alpha_{g,k+1}=\alpha_k$ et aller à l'étape 2.
Etape 2 :
si $\alpha_{d,k+1}=0$ déterminer $\alpha_{k+1}\in]\alpha_{g,k+1},+\infty[$
si $\alpha_{d,k+1}\neq 0$ déterminer $\alpha_{k+1}\in]\alpha_{g,k+1},\alpha_{d,k+1}[$ □

Dans le théorème suivant on va assurer l'existence du pas de Goldstein&price en posant quelques conditions sur la fonction h_k.

Théorème 2.2 ([40]) *Si $h_k : \mathbb{R}_+ \to \mathbb{R}$;définie par $h_k(\alpha) = f(x_k + \alpha d_k)$ est continue et bornée inférieurement, si d_k est une direction de descente en x_k et si $\rho \in]0, 1[\,, \delta \in]\rho, 1[\,$, alors l'ensemble des pas vérifiant la règle de Goldstein&Price (2.4)-(2.5) est non vide.*

Preuve. On a

$$\begin{aligned} h_k(\alpha) &= f(x_k + \alpha d_k) \\ \psi_\rho(\alpha) &= f(x_k) + \rho\alpha\nabla^T f(x_k)d_k \\ \psi_\delta(\alpha) &= f(x_k) + \delta\alpha\nabla^T f(x_k)d_k \end{aligned}$$

Le développement de Taylor-Yong en $\alpha = 0$ de h_k est :

$$h_k(\alpha) = f(x_k + \alpha d_k) = f(x_k) + \rho\alpha\nabla^T f(x_k)d_k + \alpha\xi(\alpha) \text{ où } \xi(\alpha) \to 0, \alpha \to 0.$$

et comme $\rho \in]0, 1[$ et $h'_k(0) = \nabla^T f(x_k)d_k < 0$ on déduit :

$$f(x_k) + \alpha\nabla^T f(x_k)d_k < f(x_k) + \delta\alpha\nabla^T f(x_k)d_k < f(x_k) + \rho\alpha\nabla^T f(x_k)d_k \text{ pour } \alpha > 0$$

On voit que pour $\alpha > 0$ assez petit on a :

$$h_k(\alpha) < \psi_\delta(\alpha) < \psi_\rho(\alpha)$$

De ce qui précède et du fait que h_k est bornée inférieurement,
et $\psi_\rho(\alpha) \to -\infty; \alpha \to +\infty$, on déduit que la fonction $\psi_\rho(\alpha) - h_k(\alpha)$ a la propriété :

$$\begin{cases} \psi_\rho(\alpha) - h_k(\alpha) > 0 & \text{pour } \alpha \text{ assez petit} \\ \psi_\rho(\alpha) - h_k(\alpha) < 0 & \text{pour } \alpha \text{ assez grand} \end{cases}$$

donc s'annule au moins une fois pour $\alpha > 0$.
En choisissant le plus petit de ces zéros on voit qu'il existe $\bar{\alpha} > 0$ tel que

$$h_k(\bar{\alpha}) = \psi_\rho(\bar{\alpha}) \text{ et } h_k(\alpha) < \psi_\rho(\alpha) \text{ pour } 0 < \alpha < \bar{\alpha}$$

De la meme manière, il existe $\tilde{\alpha} > 0$ tel que :

$$h_k(\tilde{\alpha}) = \psi_\delta(\tilde{\alpha}) \text{ et } h_k(\alpha) < \psi_\delta(\alpha) \text{ pour } 0 < \alpha < \tilde{\alpha}$$

et comme $\psi_\delta(\alpha) < \psi_\rho(\alpha)$ pour $\alpha > 0$, forcément $\tilde{\alpha} < \bar{\alpha}$ et $\alpha = \bar{\alpha}$ satisfait (2.4)-(2.5)

$$\left(\begin{array}{c} \psi_\delta(\tilde{\alpha}) = \ h_k(\tilde{\alpha}) < \psi_\rho(\alpha) \text{ n'est autre que} \\ f(x_k) + \delta\tilde{\alpha}\nabla^T f(x_k)d_k = f(x_k + \tilde{\alpha}d_k) < f(x_k) + \rho\tilde{\alpha}\nabla^T f(x_k)d_k \end{array} \right)$$

Ce qu'il fallait démontrer. ■

2.5 La règle de Wolfe

La conditions (2.4)-(2.5) "règle de Goldstein&Price" peuvent exclure un minimum ce qui est peut être un inconvénient.

Les conditions de wolfe ([71]) n'ont pas cet inconvénient.

Etant donnés deux réels ρ et σ tel que $0 < \rho < \sigma < 1$, ces conditions sont :

$$f(x_k + \alpha d_k) \leq f(x_k) + \rho\alpha\nabla^T f(x_k)d_k \tag{2.6}$$

$$\nabla^T f(x_k + \alpha d_k)d_k \geq \sigma\nabla^T f(x_k)d_k \tag{2.7}$$

autrement dit :

$$h_k(\alpha) \leq h_k(0) + \rho h'_k(0)\alpha$$

$$h'_k(\alpha) \geq \sigma h'_k(0)$$

2.5.1 Test de Wolfe :

♢ Si

$$h_k(\alpha) \leq h_k(0) + \rho h'_k(0)\alpha \text{ et } h'_k(\alpha) \geq \sigma h'_k(0)$$

alors α convient.

♢ Si

$$h_k(\alpha) > h_k(0) + \rho h'_k(0)\alpha$$

alors α est trop grand.

♢ Si

$$h'_k(\alpha) < \sigma h'_k(0)$$

alors α est trop petit. □

Algorithme 2.5 (Règle de Wolfe)

Etape 0 : (initialisation)

$\alpha_{g,1} = \alpha_{d,1} = 0$, choisir $\alpha_1 > 0$, $\rho \in]0, 1[$, $\sigma \in]\rho, 1[$, poser$k = 1$ et aller à l'étape 1.

Etape 1 :

si $h_k(\alpha_k) \leq h_k(0) + \rho h'_k(0)\alpha_k$ et $h'_k(\alpha) \geq \sigma h'_k(0)$: STOP$(\alpha^* = \alpha_k)$.

si $h_k(\alpha_k) > h_k(0) + \rho h'_k(0)\alpha_k$, alors

$\alpha_{d,k+1} = \alpha_k$,$\alpha_{g,k+1} = \alpha_{g,k}$ et aller à l'étape 2.

si $h'_k(\alpha) < \sigma h'_k(0)$, alors

$\alpha_{d,k+1} = \alpha_{d,k}$,$\alpha_{g,k+1} = \alpha_k$ et aller à l'étape 2.

Etape 2 :

si $\alpha_{d,k+1} = 0$ déterminer $\alpha_{k+1} \in]\alpha_{g,k+1}, +\infty[$

si $\alpha_{d,k+1} \neq 0$ déterminer $\alpha_{k+1} \in]\alpha_{g,k+1}, \alpha_{d,k+1}[$ □

Remarque 2.4 *La règle de Wolfe fait appel au calcul de h'_k, elle est donc en théorie plus couteuse que la règle de Goldstein&Price. Cependant dans de nombreuses applications, le calcul du gradient $\nabla f(x)$ représente un faible cout additionnel en comparaison du cout d'évaluation de $f(x)$, c'est pourquoi cette règle est très utilisée.*

2.5.2 Conditions de Wolfe fortes

Pour certains algorithmes (par exemple le gradient conjugué non linéaire, voir chapitre 3) il est parfois nécéssaire d'avoir une condition plus restrictive que (2.7).

Pour cela la deuxième condition (2.7) est remplacée par

$$\left|\nabla^T f(x_k + \alpha d_k) d_k\right| \leq \sigma \left|\nabla^T f(x_k) d_k\right| = -\nabla^T f(x_k) d_k.$$

On aurra donc les conditions de Wolfe fortes ([72]) :

$$f(x_k + \alpha d_k) \leq f(x_k) + \rho\alpha \nabla^T f(x_k) d_k \tag{2.8}$$

$$\left|\nabla^T f(x_k + \alpha d_k) d_k\right| \leq -\sigma \nabla^T f(x_k) d_k \tag{2.9}$$

autrement dit :

$$h_k(\alpha) \leq h_k(0) + \rho h'_k(0)\alpha$$

$$|h'_k(\alpha)| \leq -\sigma h'_k(0)$$

où $0 < \rho < \sigma < 1$.

Remarque 2.5 *La seconde condition (2.7) ou condition de courbure interdit le choix de pas trop petit pouvant entrainer une convergence lente ou prématurée.*

Remarque 2.6 *On voit bien que les conditions de Wolfe fortes impliquent les conditions de Wolfe faibles. Effectivement (2.8) est équivalente à (2.6), tandis que (2.9)*

$$\begin{aligned}\left|\nabla^T f(x_k+\alpha_k d_k)d_k\right| &\leq -\sigma\nabla^T f(x_k)d_k \\ &\Leftrightarrow \sigma\nabla^T f(x_k)d_k \leq \nabla^T f(x_k+\alpha_k d_k)d_k \leq -\sigma\nabla^T f(x_k)d_k \\ &\Rightarrow \sigma\nabla^T f(x_k)d_k \leq \nabla^T f(x_k+\alpha_k d_k)d_k \qquad \square\end{aligned}$$

2.5.3 La règle de Wolfe relaxée

Proposée par Dai et Yuan ([25]), cette règle consiste à choisir le pas α satisfaisant aux conditions :

$$f(x_k+\alpha d_k) \leq f(x_k) + \rho\alpha\nabla^T f(x_k)d_k \tag{2.10}$$

$$\sigma_1\nabla^T f(x_k)d_k \leq \nabla^T f(x_k+\alpha_k d_k)d_k \leq -\sigma_2\nabla^T f(x_k)d_k \tag{2.11}$$

autrement dit :

$$h_k(\alpha) \leq h_k(0) + \rho h_k'(0)\alpha$$

$$\sigma_1 h_k'(0) \leq h_k'(\alpha) \leq -\sigma_2 h_k'(0)$$

où $0 < \rho < \sigma_1 < 1$ et $\sigma_2 > 0$.

Remarque 2.7 *On voit bien que les conditions de Wolfe relaxée impliquent les conditions de Wolfe fortes. Effectivement (2.8) est équivalente à (2.10), tandis que pour le cas particulier $\sigma_1 = \sigma_2 = \sigma$, (2.9) est équivalente à (2.11).*

En effet :

$$\begin{aligned}\sigma_1\nabla^T f(x_k)d_k &\leq \nabla^T f(x_k+\alpha_k d_k)d_k \leq -\sigma_2\nabla^T f(x_k)d_k \\ &\Rightarrow \sigma\nabla^T f(x_k)d_k \leq \nabla^T f(x_k+\alpha_k d_k)d_k \leq -\sigma\nabla^T f(x_k)d_k \\ &\Rightarrow \left|\nabla^T f(x_k+\alpha_k d_k)d_k\right| \leq -\sigma\nabla^T f(x_k)d_k\end{aligned}$$

Remarque 2.8 *Les conditions de Wolfe relaxée impliquent les conditions de Wolfe faibles. Effectivement (2.6) est équivalente à (2.14), tandis que pour le cas particulier $\sigma_1 = \sigma$ et $\sigma_2 = +\infty$, (2.7) est équivalente à (2.11).*

En effet :

$$\begin{aligned} \sigma_1 \nabla^T f(x_k) d_k &\leq \nabla^T f(x_k + \alpha_k d_k) d_k \leq -\sigma_2 \nabla^T f(x_k) d_k \\ &\Rightarrow \quad \sigma \nabla^T f(x_k) d_k \leq \nabla^T f(x_k + \alpha_k d_k) d_k \end{aligned}$$

Dans le théorème suivant on va assurer l'existence du pas de Wolfe en posant quelques conditions sur la fonction h_k.

Théorème 2.3 ([18]) *Si $h_k : \mathbb{R}_+ \to \mathbb{R}$;définie par $h_k(\alpha) = f(x_k + \alpha d_k)$ est dérivable et bornée inférieurement, si d_k est une direction de descente en x_k et si $\rho \in]0, 1[\,, \sigma \in]\rho, 1[\,$, alors l'ensemble des pas vérifiant la règle de Wolfe (faible) (2.6)-(2.7) est non vide.*

Preuve. On a

$$\begin{aligned} h_k(\alpha) &= f(x_k + \alpha d_k) \\ \psi_\rho(\alpha) &= f(x_k) + \rho \alpha \nabla^T f(x_k) d_k \end{aligned}$$

Le développement de Taylor-Yong en $\alpha = 0$ de h_k est :

$$h_k(\alpha) = f(x_k + \alpha d_k) = f(x_k) + \rho \alpha \nabla^T f(x_k) d_k + \alpha \xi(\alpha) \text{ où } \xi(\alpha) \to 0, \alpha \to 0.$$

et comme $\rho \in]0, 1[$ et $h'_k(0) = \nabla^T f(x_k) d_k < 0$ on déduit :

$$f(x_k) + \alpha \nabla^T f(x_k) d_k < f(x_k) + \rho \alpha \nabla^T f(x_k) d_k \text{ pour } \alpha > 0$$

On voit que pour $\alpha > 0$ assez petit on a :

$$h_k(\alpha) < \psi_\rho(\alpha)$$

De ce qui précède et du fait que h_k est bornée inférieurement, et $\psi_\rho(\alpha) \to -\infty; \alpha \to +\infty$, on déduit que la fonction $\psi_\rho(\alpha) - h_k(\alpha)$ a la propriété :

$$\begin{cases} \psi_\rho(\alpha) - h_k(\alpha) > 0 & \text{pour } \alpha \text{ assez petit} \\ \psi_\rho(\alpha) - h_k(\alpha) < 0 & \text{pour } \alpha \text{ assez grand} \end{cases}$$

donc s'annule au moins une fois pour $\alpha > 0$.

En choisissant le plus petit de ces zéros on voit qu'il existe $\bar{\alpha} > 0$ tel que

$$h_k(\bar{\alpha}) = \psi_\rho(\bar{\alpha}) \text{ et } h_k(\alpha) < \psi_\rho(\alpha) \text{ pour } 0 < \alpha < \bar{\alpha} \qquad (*)$$

La formule des accroissements finis fournit alors un nombre $\hat{\alpha}, 0 < \hat{\alpha} < \bar{\alpha}$ tel que

$$\begin{aligned} h_k(\bar{\alpha}) - h_k(0) &= \bar{\alpha} h_k'(\hat{\alpha}) = \bar{\alpha}\nabla^T f(x_k + \hat{\alpha} d_k) d_k \\ &\Rightarrow \rho\bar{\alpha}\nabla^T f(x_k) d_k = \bar{\alpha}\nabla^T f(x_k + \hat{\alpha} d_k) d_k \\ &\Rightarrow \nabla^T f(x_k + \hat{\alpha} d_k) d_k = \rho\nabla^T f(x_k) d_k \geq \sigma\nabla^T f(x_k) d_k \end{aligned}$$

car $0 < \rho < \sigma < 1$ et $\nabla^T f(x_k) d_k < 0$.
Donc $\hat{\alpha}$ satisfait (2.7)
D'autre part, $\alpha = \hat{\alpha}$ satisfait (2.6), en effet,
$\hat{\alpha}$ satisfait (*) $h_k(\hat{\alpha}) < \psi_\rho(\hat{\alpha})$ n'est autre que :

$$f(x_k + \hat{\alpha} d_k) < f(x_k) + \rho\hat{\alpha}\nabla^T f(x_k) d_k$$

Ce qu'il fallait démontrer. ■

En pratique, on utilise des algorithmes spécifiques pour trouver un pas de Wolfe.

On va présenter un algorithme simple, appelé de Fletcher-Lemaréchal, dont on peut montrer qu'il trouve un pas de Wolfe en un nombre fini d'étapes.

Algorithme 2.6 (Règle de Fletcher-Lemaréchal)

Etape0 : (initialisation)

$\alpha_{g,1} = 0$, $\alpha_{d,1} = +\infty$, $\rho \in]0, 1[$, $\sigma \in]\rho, 1[$, $\upsilon \in \left]0, \frac{1}{2}\right[$, $\tau > 1$,choisir $\alpha_1 > 0$, poser $k = 1$ et aller à l'étape 1.

Etape 1 :

1.1 si $h_k(\alpha_k) > h_k(0) + \rho h_k'(0)\alpha_k$, alors

$\alpha_{d,k+1} = \alpha_k$,$\alpha_{g,k+1} = \alpha_{g,k}$ et aller à l'étape 2.

1.3 Si non $(h_k(\alpha_k) \leq h_k(0) + \rho h_k'(0)\alpha_k)$, alors

1.3.1 si $h_k'(\alpha) \geq \sigma h_k'(0)$: STOP($\alpha^* = \alpha_k$).

1.3.2 si non $(h_k'(\alpha) < \sigma h_k'(0))$

$\alpha_{d,k+1} = \alpha_{d,k}$,$\alpha_{g,k+1} = \alpha_k$ et aller à l'étape 2.

Etape 2 :

si $\alpha_{d,k+1} = +\infty$ déterminer $\alpha_{k+1} \in]\tau\alpha_{g,k+1}, +\infty[$

si non déterminer $\alpha_{k+1} \in](1-\upsilon)\alpha_{g,k+1} + \upsilon\alpha_{d,k+1}, \upsilon\alpha_{g,k+1} + (1-\upsilon)\alpha_{d,k+1}[$
poser $k = k+1$ et aller à l'étape 1. $\square$

Théorème 2.4 ([40]) *Si $h_k : \mathbb{R}_+ \to \mathbb{R}$;définie par $h_k(\alpha) = f(x_k + \alpha d_k)$ est dérivable et bornée inférieurement, si d_k est une direction de descente en x_k et si $\rho \in]0,1[\,, \sigma \in]\rho, 1[\,,$ alors l'alogorithme de Fletcher-Lemaréchal trouve le pas de Wolfe(2.6)-(2.7) en un nombre fini d'étapes.*

Chapitre III

Méthodes a directions de descentes et méthodes à directions conjuguées

Ce chapitre introduit une classe importante d'algorithmes de résolution des problèmes d'optimisation sans contraintes.

Le concept central est celui de direction de descente. On le retrouvera dans des contextes variés, également pour résoudre des problèmes avec contraintes. Tous les algorithmes d'optimisation n'entrent pas dans ce cadre. Une autre classe importante de méthodes se fonde sur la notion de région de confiance.

Après avoir décrit comment fonctionne un algorithme a directions de descente, nous donnons quelques exemples d'algorithmes de ce type .

1 Principe général des Méthodes à directions de descentes

Considérons le problème d'optimisation sans contraintes

$$\min_{x \in \mathbb{R}^n} f(x) \tag{3.1}$$

où $f : \mathbb{R}^n \to \mathbb{R}$ est supposée régulière.

On note aussi $\nabla f(x)$ et $\nabla^2 f(x)$ le gradient et le hessien de f en x.

On s'intéresse ici à une classe d'algorithmes qui sont fondés sur la notion de direction de descente.

1.1 Direction de descente

Définition 1.1 *Soit $f : \mathbb{R}^n \to \mathbb{R}$, d un vecteur non nul de $\mathbb{R}^n$ est dit une direction de descente., s'il existe un réel $\delta > 0$ tel que pour tout λ dans l'intervalle$]0, \delta[$, on a*

$$f(x + \lambda d) < f(x) \tag{3.2}$$

Théorème 1.1 *Si f est différentiable en un point $x \in \mathbb{R}^n$ et $d \in \mathbb{R}^n$ telle que*

$$\nabla^T f(x)\, d < 0 \tag{3.3}$$

Alors d est une direction de descente en x

Remarque 1.1 *d fait avec l'opposé du gradient $-\nabla f(x)$ un angle θ strictement plus petit que $90°$:*

$$\theta := \arccos \frac{-\nabla^T f(x).d}{\|\nabla f(x)\| \, \|d\|} \in \left[0, \frac{\pi}{2}\right[. \tag{3.4}$$

L'ensemble des directions de descente de f en x, $\left\{d \in \mathbb{R}^n : \nabla^T f(x).d < 0\right\}$ forme un demi-espace ouvert de $\mathbb{R}^n$.

De telles directions sont intéressantes en optimisation car, pour faire décroître f, il suffit de faire un déplacement le long de d.

1.2 Description des méthodes à directions de descentes

Les méthodes à directions de descentes utilisent l'idée suivante pour minimiser une fonction [14].

Elles construisent la suite des itérés $\{x_k\}_{k\geq 1}$, approchant une solution x_k de (P) par la récurrence :

$$x_{k+1} = x_k + \alpha_k d_k, \text{ pour } k \geq 1 \tag{3.5}$$

Où α_k est appelé le pas et d_k la direction de descente de f en x_k.

Pour définir une méthode à directions de descente il faut donc spécifier deux choses :

$\diamondsuit$ dire comment la direction d_k est calculée ; la manière de procéder donne le nom à l'algorithme ;

$\diamondsuit$ dire comment on détermine le pas α_k; c'est ce que l'on appelle : la recherche linéaire.

Décrivons cette classe d'algorithmes de manière précise.

Algorithme 2.1(méthode à directions de descente- une itération)

Etape 0 : (initialisation)

On suppose qu'au début de l'itération k, on dispose d'un itéré $x_k \in \mathbb{R}^n$

Etape1 :

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

Etape2 :

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

Etape3 :

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

Etape4 :

Si la recherche linéaire réussit : $x_{k+1} = x_k + \alpha_k d_k$;

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

2 Exemples de méthodes à directions de descente

Supposons que d_k soit une direction de descente au point x_k . Ceci nous permet de considérer le point x_{k+1}, successeur de x_k de la manière suivante :

$$x_{k+1} = x_k + \alpha_k d_k, \qquad \alpha_k \in]0, +\delta[\,. \tag{3.6}$$

Vu la définition de direction de descente, on est assuré que

$$f\left(x_{k+1}\right) = f\left(x_k + \alpha_k d_k\right) < f\left(x_k\right) \tag{3.7}$$

Un bon choix de d_k et de α_k permet ainsi de construire une multitude d'algorithmes d'optimisation

2.1 Exemples de choix de directions de descente

par exemple si on choisit $d_k = -\nabla f(x_k)$ et si $\nabla f(x_k) \neq 0$, on obtient la méthode du gradient.

Bien sur $d_k = -\nabla f(x_k)$ est une direction de descente puisque $\nabla f(x_k)^T.d_k = -\nabla f(x_k)^T \nabla f(x_k) = -\left\|\nabla f(x_k)\right\|^2 < 0$

La méthode de Newton correspond à $d_k = -\left(H\left(x_k\right)\right)^{-1}.\nabla f(x_k)$.

$d_k$direction de descente si la matrice hessienne $H\left(x_k\right)$ est définie positive

2.2 Exemples de choix de pas α_k

On choisit en général$_k$ de façon optimale, c'est à dire que α_k doit vérifier

$$f(x_k + \alpha_k d_k) \leq f(x_k + \alpha d_k) : \ \forall \alpha \in [0, +\infty[\,. \tag{3.8}$$

En d'autres termes on est ramené à étudier à chaque itération un problème de minimisation d'une variable réelle .C'est ce qu'on appelle recherche linéaire.

3 Convergence des méthodes à directions de descente

3.1 Condition de Zoutendijk

Dans cette section, on va étudier la *contribution* de la recherche linéaire inexacte à 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 satisfait la condition de Zoutendijk s'il existe une constante $C > 0$ telle que pour tout indice $k \geq 1$ on ait

$$f(x_{k+1}) \leq f(x_k) - C \left\|\nabla f(x_k)\right\|^2 \cos^2 \theta_k \tag{3.9}$$

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

$$\cos \theta_k = \frac{-\nabla^T f(x_k) d_k}{\|\nabla f(x)\| \, \|d_k\|}$$

Voici comment on se sert de la condition de Zoutendijk.

Proposition 3.1 ([40]) *Si la suite $\{x_k\}$ générée par un algorithme d'optimisation vérifie la condition de Zoutendijk (3.9) et si la suite $\{f(x_k)\}$ est minorée, alors*

$$\sum_{k \geq 1} \left\|\nabla f(x_k)\right\|^2 \cos^2 \theta_k < \infty \tag{3.10}$$

Preuve. En sommant les inégalités (3.9), on a

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

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

Les deux propositions suivantes précisent les circonstances dans les quelles la condition de Zoutendijk (3.9) est vérifié avec les règles d'Armijo et de Wolfe.

Proposition 3.2 ([40]) *Soit $f : \mathbb{R}^n \to \mathbb{R}$ une fonction continuement différentiable dans un voisinage de $\Gamma = \{x \in \mathbb{R}^n : f(x) \leq f(x_1)\}$.*

On considère un algorithme à directions de descente d_k, qui génère une suite $\{x_k\}$ en utilisant la recherche linéaire d'Armijo(2.6) avec $\alpha_1 > 0$.

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

$$f(x_{k+1}) \leq f(x_k) - C\nabla^T f(x_k) d_k \tag{3.11}$$

ou

$$f(x_{k+1}) \leq f(x_k) - C\|\nabla f(x_k)\|^2 \cos^2\theta_k \tag{3.12}$$

est vérifiée.

Preuve. Si le pas $\alpha_k = \alpha_1$ est accepté, on a (3.9), car α_1 est uniformément positif.

Dans le cas contraire, (2.6) n'est pas vérifiée avec un pas $\alpha'_k \leq \frac{\alpha_k}{\tau}$, c'est-à-dire $f(x_k+\alpha'_k d_k) > f(x_k) + \rho\alpha'_k \nabla^T f(x_k) d_k$

Comme f est continuement différentiable, on a pour tout $\alpha_k > 0$:

$$\begin{aligned} f(x_k + \alpha_k d_k) &= f(x_k) + \alpha_k \nabla^T f(x_k) d_k + \int_0^1 \left[\nabla f(x_k + t\alpha_k d_k) - \nabla f(x_k)\right]^T \alpha_k d_k dt \\ &\Rightarrow \quad f(x_k + \alpha d_k) \leq f(x_k) + \alpha_k \nabla^T f(x_k) d_k + C\alpha_k^2 \|d_k\|^2 \end{aligned}$$

où C > 0 est une constante. Avec l'inégalité précédente, et le fait que , on obtient :

$$\begin{cases} f(x_k + \alpha'_k d_k) - f(x_k) > \rho\alpha'_k \nabla^T f(x_k) d_k \\ f(x_k + \alpha'_k d_k) - f(x_k) \leq \alpha'_k \nabla^T f(x_k) d_k + C\alpha'^2_k \|d_k\|^2 \end{cases}$$

$$\begin{aligned} &\Rightarrow \quad \rho\alpha'_k\nabla^T f(x_k)d_k \leq \alpha_k\nabla^T f(x_k)d_k + C\alpha_k^2 \left\|d_k\right\|^2 \\ &\Rightarrow \quad -C\alpha_k'^2 \left\|d_k\right\|^2 \leq (1-\rho)\alpha'_k\nabla^T f(x_k)d_k \end{aligned}$$

or

$$\rho < 1 \Rightarrow 0 < 1-\rho < 1 \Rightarrow \frac{1}{1-\rho} > 1$$

d'où

$$\begin{aligned} \nabla^T f(x_k)d_k \quad &\geq \quad \frac{-C}{1-\rho}\alpha'_k \left\|d_k\right\|^2 \Rightarrow -\nabla^T f(x_k)d_k \leq \frac{C}{1-\rho}\alpha'_k \left\|d_k\right\|^2 \\ &\Rightarrow \quad \left|\nabla^T f(x_k)d_k\right| = \left\|\nabla^T f(x_k)\right\| \left\|d_k\right\| \cos\theta_k =\leq \frac{C}{1-\rho}\alpha'_k \left\|d_k\right\|^2 \end{aligned}$$

ce qui permet de minorer $\alpha' \left\|d_k\right\|$ et donc aussi $\alpha \left\|d_k\right\|$ par une constante fois $\left\|\nabla^T f(x_k)\right\| \left\|d_k\right\|$. Cette minoration et l'expression suivante de (2.6)

$$f(x_k + \alpha_k d_k) \leq f(x_k) - \rho\alpha_k \left\|\nabla^T f(x_k)\right\| \left\|d_k\right\| \cos\theta_k$$

conduit à(3.9). ■

Proposition 3.3 ([40]) *Soit $f : \mathbb{R}^n \to \mathbb{R}$ une fonction continuement différentiable dans un voisinage de $\Gamma = \{x \in \mathbb{R}^n : f(x) \leq f(x_1)\}$.*

On considère un algorithme à directions de descente d_k, qui génère une suite $\{x_k\}$ en utilisant la recherche linéaire de Wolfe (2.10)-(2.11).

Alors il existe une constante $C > 0$ telle que, pour tout $k \geq 1$, la condition de Zoutendijk (3.9) est vérifiée.

Preuve. D'après (2.10)

$$\nabla^T f(x_k + \alpha_k d_k)d_k \geq \sigma\nabla^T f(x_k)d_k$$

$$\begin{aligned} &\Rightarrow \quad \left(\nabla f(x_k + \alpha_k d_k) - \nabla f(x_k)\right)^T d_k \geq (\sigma - 1)\nabla^T f(x_k)d_k \\ &= \quad -(1-\sigma)\nabla^T f(x_k)d_k = (1-\sigma)\left|\nabla^T f(x_k)d_k\right| \\ &\Leftrightarrow \quad (1-\sigma)\left|\nabla^T f(x_k)d_k\right| \leq \left(\nabla f(x_k + \alpha_k d_k) - \nabla f(x_k)\right)^T d_k \end{aligned}$$

et du fait que f est contiuement différentiable :

$$\begin{aligned}(1-\sigma)\left|\nabla^T f(x_k)d_k\right| &= (1-\sigma)\left\|\nabla^T f(x_k)\right\|\|d_k\|\cos\theta_k \\ &\leq \|\nabla f(x_k+\alpha_k d_k)-\nabla f(x_k)\|\,\|d_k\| \\ &\Rightarrow (1-\sigma)\left\|\nabla^T f(x_k)\right\|\cos\theta_k \leq L\alpha_k\|d_k\| \\ &\Rightarrow \alpha_k\|d_k\| \leq \frac{(1-\sigma)}{L}\left\|\nabla^T f(x_k)\right\|\cos\theta_k\end{aligned}$$

en utilisant (2.10), on aura :

$$\begin{aligned}f(x_k+\alpha_k d_k) &\leq f(x_k)+\rho\alpha_k\nabla^T f(x_k)d_k \\ &\Rightarrow f(x_k+\alpha d_k)\leq f(x_k)+\rho\alpha\nabla^T f(x_k)d_k \leq f(x_k)+\left|\rho\alpha\nabla^T f(x_k)d_k\right| \\ &\Rightarrow f(x_k+\alpha d_k)\leq f(x_k)+\rho\alpha\left|\nabla^T f(x_k)d_k\right| \leq f(x_k)-\rho\alpha\nabla^T f(x_k)d_k \\ &\Rightarrow f(x_k+\alpha d_k)\leq f(x_k)-\rho\alpha\left\|\nabla^T f(x_k)\right\|\|d_k\|\cos\theta_k \\ &\Rightarrow f(x_k+\alpha d_k)\leq f(x_k)-\frac{\rho(1-\sigma)}{L}\left\|\nabla^T f(x_k)\right\|^2\cos^2\theta_k\end{aligned}$$

On en déduit (3.9). ■

4 Méthodes du gradient conjugué

Les méthodes du gradient conjugué sont utilisées pour résoudre les problèmes d'optimisation non linéaires sans contraintes spécialement les problèmes de grandes tailles. On l'utilise aussi pour résoudre les grands systèmes linéaires.

Elles reposent sur le concept des directions conjuguées parce que les gradients successifs sont orthogonaux entre eux et aux directions précédentes.

L'idée initiale était de trouver une suite de directions de descente permettant de résoudre le problème

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

Où f est régulière (continûment différentiable)

Dans ce chapitre on va décrire toutes ces méthodes, mais avant d'accéder à ces dernières, on va d'abord donner le principe général d'une méthode à directions conjuguées

5 Le principe général d'une méthode à directions conjuguées

Donnons la définition de "*conjugaison*" :

Définition 5.1 *Soit A une matrice symétrique $n \times n$, définie positive. On dit que deux vecteurs x et y de $\mathbb{R}^n$ sont $A-$conjugués (ou conjugués par rapport à A) s'ils vérifient*

$$x^T A y = 0 \tag{3.13}$$

5.1 Description de la méthode

Soit $\{d_0, d_1, ..., d_n\}$ une famille de vecteurs A-conjugués. On appelle alors méthode de directions conjuguées toute méthode itérative appliquée à une fonction quadratique strictement convexe de n variables : $q(x) = \frac{1}{2}x^T Ax + b^T x + c$; avec $x \in \mathbb{R}^n$ et $A \in \mathcal{M}_{n\times n}$ est symétrique et définie positive, $b \in \mathbb{R}^n$ et $c \in \mathbb{R}$, conduisant à l'*optimum* en n étapes au plus. Cette méthode est de la forme suivante :

$$x_0 \text{ donné}$$

$$x_{k+1} = x_k + \alpha_k d_k \tag{3.14}$$

où α_k est optimal et $d_1, d_2, .., d_n$ possédant la propriété d'être *mutuellement conjuguées* par rapport à la fonction quadratique

Si l'on note $g_k = \nabla q\,(x_k)$, la méthode se construit comme suit :

Calcul de α_k

Comme α_k minimise q dans la direction d_k, on a, $\forall k$:

$$q'\,(\alpha_k) = d_k^T \nabla q(x_{k+1}) = 0$$

$$d_k^T \nabla q(x_{k+1}) = d_k^T (Ax_{k+1} + b) = 0.$$

Soit :

$$d_k^T A(x_k + \alpha_k d_k) + d_k^T b = 0$$

d'où l'on tire :

$$\alpha_k = \frac{-d_k^T\,(Ax_k + b)}{d_k^T A d_k} \tag{3.15}$$

Comment construire les directions A-conjuguées ?

Des directions A-conjuguées $d_0, ..., d_k$ peuvent être générées à partir d'un ensemble de vecteurs linéairement indépendants $\xi_0, ..., \xi_k$ en utilisant la procédure dite de Gram-Schmidt, de telle sorte que pour tout i entre 0 et k, le sous-espace généré par $d_0, ..., d_i$ soit égale au sous-espace généré par $\xi_0, ..., \xi_i$.

Alors d_{i+1} est construite comme suit :

$$d_{i+1} = \xi_{i+1} + \sum_{m=0}^{i} \varphi_{(i+1)m} d_m$$

Nous pouvons noter que si d_{i+1} est construite d'une telle manière, elle est effectivement linéairement indépendante avec $d_0, ..., d_i$.

En effet, le sous-espace généré par les directions $d_0, ..., d_i$ est le même que le sous-espace généré par les directions $\xi_0, ..., \xi_i$, et ξ_{i+1} est linéairement indépendant de $\xi_0, ..., \xi_i$.

ξ_{i+1} ne fait donc pas partie du sous-espace généré par les combinaisons linéaires de la forme$^i_{m=0}\varphi_{(i+1)m} d_m$, de sorte que d_{i+1} n'en fait

pas partie non plus et est donc linéairement indépendante des $d_0, ..., d_i$.

Les coefficients $\varphi_{(i+1)m}$, eux sont choisis de manière à assurer la A-conjugaison des $d_0, ..., d_{i+1}$.

6 Méthode de gradient conjugué dans le cas quadratique

La méthode du gradient conjugué quadratique est obtenue en appliquant la procédure de Gram-Schmidt aux gradients $\nabla q(x_0), ..., \nabla q(x_{n-1})$, c'est-à-dire en posant $\xi_0 = -\nabla q(x_0), ..., \xi_{n-1} = -\nabla q(x_{n-1})$.

En outre, nous avons :

$$\begin{aligned} \nabla q(x) &= Ax + b \\ \text{et } \nabla^2 q(x) &= A. \end{aligned}$$

Notons que la méthode se termine si $\nabla q(x_k) = 0$.

La particularité intéressante de la méthode du gradient conjugué est que le membre de droite de l'équation donnant la valeur de d_{k+1} dans la procédure de Gram-Schmidt peut être grandement simplifié.

Notons que la méthode du gradient conjugué est inspirée de celle du gradient (plus profonde pente).

6.1 Algorithme de La méthode du gradient conjugué pour les fonctions quadratiques

On suppose ici que la fonction à minimiser est quadratique sous la forme : $q(x) = \frac{1}{2}x^T Ax + b^T x + c$

Si l'on note $g_k = \nabla f(x_k)$, l'algorithme prend la forme suivante

Cet algorithme consiste à générer une suite d'itérés $\{x_k\}$ sous la forme :

$$x_{k+1} = x_k + \alpha_k d_k \tag{3.16}$$

L'idée de la méthode est :

1- construire itérativement des directions $d_0, ..., d_k$ mutuellement conjuguées :

A chaque étape k la direction d_k est obtenue comme combinaison linéaire du gradient en x_k et de la direction précédente d_{k-1}

c'est-à-dire

$$d_{k+1} = -\nabla q(x_{k+1}) + \beta_{k+1} d_k \tag{3.17}$$

les coefficients β_{k+1}étant choisis de telle manière que d_k soit conjuguée avec toutes les directions

précédentes. autrement dit :

$$d_{k+1}^T A d_k = 0,$$

on en déduit :

$$\begin{aligned} d_{k+1}^T A d_k &= 0 \Rightarrow \left(-\nabla q(x_{k+1}) + \beta_{k+1} d_k\right)^T A d_k = 0 \\ &\Rightarrow -\nabla^T q(x_{k+1}) A d_k + \beta_{k+1} d_k^T A d_k = 0 \end{aligned}$$

$$\Rightarrow \beta_{k+1} = \frac{\nabla^T q(x_{k+1}) A d_k}{d_k^T A d_k} = \frac{g_{k+1}^T A d_k}{d_k^T A d_k} \tag{3.18}$$

2-déterminer le pas α_k :

En particulier, une façon de choisir α_k consiste à résoudre le problème d'optimisation unidimensionnelle suivant :

$$\alpha_k = \min f(x_k + \alpha d_k) \quad , \quad \alpha > 0 \tag{3.19}$$

on en déduit :

$$\alpha_k = \frac{-d_k^T g_k}{d_k^T A d_k} = -\frac{1}{Ad_k} g_k \frac{d_k^T}{d_k^T} = \frac{-d_k^T g_k}{d_k^T A d_k} \tag{3.20}$$

Le pas α_k obtenu ainsi s'appelle le pas optimal.

Algorithme 3.1 (Algorithme du gradient conjugué "quadratique"

Etape 0 : (initialisation)

Soit x_0 le point de départ, $g_0 = \nabla q(x_0) = Ax_0 + b$, poser $d_0 = -g_0$

poser $k = 0$ et aller à l'étape 1.

Etape 1 :

si $g_k = 0$: STOP ($x^* = x_k$)."Test d'arrêt"

si non aller à l'étape 2.

Etape 2 :

Prendre $x_{k+1} = x_k + \alpha_k d_k$ avec :

$$\alpha_k = \frac{-d_k^T g_k}{d_k^T A d_k}$$

$$d_{k+1} = -g_{k+1} + \beta_{k+1} d_k$$

$$\beta_{k+1} = \frac{g_{k+1}^T A d_k}{d_k^T A d_k}$$

Poser $k = k + 1$ et aller à l'étape 1. □

6.2 La validité de l'algorithme du gradient conjugué linéaire

On va maintenant montrer que l'algorithme ci-dessus définit bien une méthode de directions conjuguées.

Théorème 6.1 ([60]) *A une itération k quelconque de l'algorithme où l'optimum de $q(x)$ n'est pas encore atteint (c'est-à-dire $g_i \neq 0$, $i = 0, 1, ..., k$) on a :*

a)

$$\alpha_k = \frac{g_k^T g_k}{d_k^T A d_k} \neq 0 \tag{3.21}$$

b)

$$\beta_{k+1} = \frac{g_{k+1}^T [g_{k+1} - g_k]}{g_k^T g_k} \tag{3.22}$$

$$= \frac{g_{k+1}^T g_{k+1}}{g_k^T g_k} \tag{3.23}$$

c) Les directions $d_0, d_1, ..., d_{k+1}$ engendrées par l'algorithme sont mutuellement conjuguées.

Preuve. On raisonne par récurrence sur k en supposant que $d_0, d_1, ..., d_k$ sont mutuellement conjuguées.

a) Montrons d'abord l'équivalence de (3.21) et de (3.22)

On a : $d_k = -g_k + \beta_k d_{k-1}$.

Donc (3.8) s'écrit :

$$\begin{aligned} \alpha_k &= \frac{-d_k^T g_k}{d_k^T A d_k} \\ &= \frac{-\left[-g_k + \beta_k d_{k-1}\right]^T g_k}{d_k^T A d_k} \\ &= \frac{g_k^T g_k}{d_k^T A d_k} - \beta_k \frac{d_{k-1}^T g_k}{d_k^T A d_k} \end{aligned}$$

Comme $(d_0, d_1, ..., d_{k-1})$ sont mutuellement conjuguées, x_k est l'optimum de $q(x)$ sur la variété ν^k passant par x_0 et engendrée par $(d_0, d_1, ..., d_{k-1})$.

Donc $d_{k-1}^T g_k = 0$ d'où l'on déduit (3.9).

b) Pour démontrer (3.10) remarquons que :

$$\begin{aligned} g_{k+1} - g_k &= A(x_{k+1} - x_k) = \alpha_k A d_k \\ &\Rightarrow \quad A d_k = \frac{1}{\alpha_k}\left[g_{k+1} - g_k\right] \end{aligned}$$

On a alors :

$$g_{k+1}^T A d_k = \frac{1}{\alpha_k} g_{k+1}^T \left[g_{k+1} - g_k\right]$$

et en utilisant (3.9)

$$\alpha_k = \frac{g_k^T g_k}{d_k^T A d_k}$$

il vient

$$\begin{aligned} g_{k+1}^T A d_k &= \frac{d_k^T A d_k}{g_k^T g_k} . g_{k+1}^T \left[g_{k+1} - g_k\right] \\ &\Rightarrow \quad \frac{g_{k+1}^T A d_k}{d_k^T A d_k} = \frac{g_{k+1}^T \left[g_{k+1} - g_k\right]}{g_k^T g_k} \end{aligned}$$

Or de (3.6) on aura :

$$\beta_{k+1} = \frac{g_{k+1}^T A d_k}{d_k^T A d_k} = \frac{g_{k+1}^T \left[g_{k+1} - g_k\right]}{g_k^T g_k}$$

ce qui démontre (3.10).

(3.11) découle alors du fait que :

$$g_{k+1}^T g_k = 0$$

car

$$g_k = d_k - \beta_k d_{k-1}$$

Appartient au sous-espace engendré par $(d_0, d_1, ..., d_k)$ et que g_{k+1} est orthogonal à ce sous-espace .

c) Montrons enfin que d_{k+1} est conjuguée par rapport à $(d_0, d_1, ..., d_k)$.

On a bien $d_{k+1}^T A d_k$ car, en utilisant $d_{k+1} = -g_{k+1} + \beta_k d_k$ on aura :

$$\begin{aligned} d_{k+1}^T A d_k &= \left(-g_{k+1} + \beta_{k+1} d_k\right)^T A d_k \\ &= -g_{k+1}^T A d_k + \beta_{k+1} d_k^T A d_k \\ &= -g_{k+1}^T A d_k + \frac{g_{k+1}^T A d_k}{d_k^T A d_k} d_k^T A d_k \\ &= 0 \end{aligned}$$

Vérifions maintenant que :

$$d_{k+1}^T A d_i = 0 \quad \text{pour } i = 0, 1, ..., k-1.$$

On a :

$$d_{k+1}^T A d_i = -g_{k+1}^T A d_i + \beta_{k+1} d_k^T A d_i$$

Le seconde terme est nul par l'hypothèse de récurrence ($(d_0, d_1, ..., d_k$)sont mutuellement conjuguées).

Montrons qu'il en est de même du premier terme. Puisque $x_{i+1} = x_i + \alpha_i d_i$ et que $\alpha_i \neq 0$ on a :

$$\begin{aligned} A d_i &= \frac{1}{\alpha_i} \left(A x_{i+1} - A x_i\right) \\ &= \frac{1}{\alpha_i} \left(g_{i+1} - g_i\right) \end{aligned}$$

En écrivant :

$$\begin{aligned} g_{i+1} &= d_{i+1} - \beta_i d_i \\ g_i &= d_i - \beta_{i-1} d_{i-1} \end{aligned}$$

on voit que Ad_i est combinaison linéaire de d_{i+1}, d_i et de d_{i-1} seulement).

Mais puisque $(d_0, d_1, ..., d_k)$ sont mutuellement conjuguées, on sait que le point x_{k+1} est l'optimum de $q(x)$ sur la variété ν^{k+1}, engendrée par $(d_0, d_1, ..., d_k)$.

Donc g_{k+1} est orthogonal au sous-espace engendré par $(d_0, d_1, ..., d_k)$ et comme Ad_i appartient à ce sous-espace pour $i = 0, 1, ..., k-1$, on en déduit $g_{k+1}^T Ad_i = 0$ ce qui achève la démonstration. ■

Remarque 6.1 *dans ce cas d_k est une direction de descente puisque*

$$\begin{aligned} d_k^T \nabla q(x_k) &= \left(-\nabla q(x_k) + \beta_{k+1} d_{k-1}\right)^T \nabla q(x_k) \\ &= -\nabla q(x_k)^T \nabla q(x_k) + \beta_{k+1} d_{k-1}^T \nabla q(x_k) \\ &= -\left\|\nabla q(x_k)\right\|^2 \quad (car\ d_{k-1}^T \nabla q(x_k) = 0\ ) \end{aligned}$$

$$d_k^T \nabla q(x_k) = -\left\|\nabla q(x_k)\right\|^2 < 0$$

6.2.1 Les avantages de la méthode du gradient conjugué quadratique

1- la consommation mémoire de l'algorithme est minime : on doit stocker les quatre vecteurs x_k, g_k, d_k, Ad_k

(bien sur x_{k+1} prend la place de x_k au niveau de son calcul avec des remarques analogues pour g_{k+1}, d_{k+1}, Ad_{k+1}) et les scalaires α_k, β_{k+1}.

2- L'algorithme du gradient conjugué linéaire est surtout utile pour résoudre des grands systèmes creux, en effet il suffit de savoir appliquer la matrice A à un vecteur.

3- La convergence peut être assez rapide : si A admet seulement r $(r < n)$ valeurs propres distincts la convergence a lieu en au plus r itération.

6.3 Différentes formules de β_{k+1} dans le cas quadratique

Formule de Hestenes-Stiefel Cette méthode été découverte en 1952 par Hestenes et Stiefels ([48]),

$$\beta_{k+1}^{HS} = \frac{g_{k+1}^T \left[g_{k+1} - g_k\right]}{d_k^T \left[g_{k+1} - g_k\right]} \tag{3.24}$$

Formule de Polak-Ribière-Polyak Cette méthode été découverte en 1969 par Polak, Ribière ([56]) et Ployak ([57]),

$$\beta_{k+1}^{PRP} = \frac{g_{k+1}^T \left[g_{k+1} - g_k\right]}{\left\|g_k\right\|^2} \tag{3.25}$$

Formule de Fletcher-Reeves Cette méthode été découverte en 1964 par Fletcher et Reeves ([35]),

$$\beta_{k+1}^{FR} = \frac{\|g_{k+1}\|^2}{\|g_k\|^2} \tag{3.26}$$

Formule de la descente conjuguée Cette méthode été découverte en 1987 par Fletcher ([34]),

$$\beta_{k+1}^{CD} = \frac{\|g_{k+1}\|^2}{-d_k^T g_k} \tag{3.27}$$

Remarque 6.2 *Dans le cas quadratique on a vu que :*

$$\beta_{k+1}^{HS} = \beta_{k+1}^{PRP} = \beta_{k+1}^{FR} = \beta_{k+1}^{CD} = \beta_{k+1}^{DY}.$$

Dans le cas non quadratique, ces quantités ont en général des valeurs différentes.

7 Méthode du gradient conjugué dans le cas non quadratique

On s'intéresse dans cette section à la minimisation d'une fonction $f : \mathbb{R}^n \to \mathbb{R}$, non nécessairement quadratique :

$$\min f(x); \quad x \in \mathbb{R}^n \tag{3.28}$$

Les méthodes du gradient conjugué génèrent des suites $\{x_k\}_{k=0,1,2...}$de la forme suivante :

$$x_{k+1} = x_k + \alpha_k d_k \tag{3.29}$$

Le pas $\alpha_k \in \mathbb{R}$ étant déterminé par une recherche linéaire. la direction d_k est définie par la formule de récurrence suivante($\beta_k \in \mathbb{R}$)

$$d_k = \begin{cases} -g_k & \text{si } k = 1 \\ -g_k + \beta_k d_{k-1} & \text{si } k \geq 2 \end{cases} \tag{3.30}$$

Ces méthodes sont des extensions de la méthode du gradient conjugué linéaire du cas quadratique, si β_k prend l'une des valeurs

$$\beta_k^{PRP} = \frac{g_k^T y_k}{\|g_{k-1}\|^2}$$

$$\beta_k^{FR} = \frac{\|g_k\|^2}{\|g_{k-1}\|^2}$$

$$\beta_k^{CD} = \frac{\|g_k\|^2}{-d_{k-1}^T g_{k-1}}$$

où $y_{k-1} = g_k - g_{k-1}$.

7.1 Algorithme de La méthode du gradient conjugué pour les fonctions quelconques

Etape0 : (initialisation)

Soit x_0 le point de départ, $g_0 = \nabla f(x_0)$, poser $d_0 = -g_0$

Poser $k = 0$ et aller à l'étape 1.

Etape 1 :

Si $g_k = 0$: STOP ($x^* = x_k$)."Test d'arrêt"

Si non aller à l'étape 2.

Etape 2 :

Définir $x_{k+1} = x_k + \alpha_k d_k$ avec :

$$\alpha_k : \text{calculer par la recherche linéaire}$$

$$d_{k+1} = -g_{k+1} + \beta_{k+1} d_k$$

où

$$\beta_{k+1} : \text{défini selon la méthode}$$

Poser $k = k + 1$ et aller à l'étape 1. □

Remarque 7.1 *Dans le cas quadratique avec recherche linéaire exacte, on a vu que* $\beta_k^{PRP} = \beta_k^{FR} = \beta_k^{CD} = \beta_k^{DY}$.

Chapitre IV

Synthèse des résultats de convergence des méthodes du gradient conjugué

Dans ce chapitre on va essayer de présenter une synthèse sur les différents résultats de convergence des méthodes du gradient conjugué pour la minimisation des fonctions sans contraintes. Ces méthodes seront utilisées avec une recherche linéaire inexacte. L'analyse couvre quatre classes de méthodes qui sont globalement convergentes pour des fonctions régulières non nécessairement convexes. Dans la première famille, ce sont certaines propriétés de la méthode de Fletcher-Reeves qui jouent un rôle crucial, tandis que la seconde famille c'est la méthode de Polak-Ribière-Polyak qui a une propriété importante. La troisième concerne la méthode de la descente conjuguée et la dans dernière famille on va présenter quelques propriétés de la nouvelle méthode du gradient conjugué non linéaire dite de Dai-Yuan. On terminera ce chapitre par quelques développements récents de la méthode du Gradient conjugué.

1 Hypothèses C1 et C2 et théorème de Zoutendijk

1.1 Hypothèses C1 et C2 (de Lipschitz et de bornetude)

Rappelons dans ce qui suit les deux conditions suivantes que toutes les méthodes du gradient conjugué doivent vérifier (ou au moins l'une d'entre elles) pour prouver les résultats de convergence.

Condition $\mathcal{C}1$

Soit $f : \mathbb{R}^n \to \mathbb{R}$. On dit que f vérifie la condition $\mathcal{C}1$ si f est *contiuement différentiable dans un voisinage* $V(\Gamma)$ *de* $\Gamma = \{x \in \mathbb{R}^n : f(x) \leq f(x_1) : x_1 \in \mathbb{R}^n \text{ point initial}\}$ et si $\nabla f(x)$ vérifie la condition de Lipschitz dans $V(\Gamma)$, c'est à dire, il existe une constante $L > 0$ telle que

$$\|\nabla f(x) - \nabla f(y)\| \leq L \|x - y\| : \quad \text{pour tout } x, y \ \in V(\Gamma)$$

Condition $\mathcal{C}2$

L'ensemble $\Gamma = \{x \in \mathbb{R}^n : f(x) \leq f(x_1)\}$ est borné, i.e., il existe une constante $B < \infty$ telle que

$$\|x\| \leq B : \ \forall x \in \Gamma$$

1.2 Théorème de Zoutendijk

L'outil de base utilisé par les différentes variantes du gradient conjugué avec une recherche linéaire inexacte est le théorème suivant du à Zoutendijk

Théorème 1.1 ([40]) *Considérons une méthode itérative quelconque générant une suite* $\{x_k\}$ *de la forme :*

$$x_{k+1} = x_k + \alpha_k d_k,$$

d_k *étant une direction de descente et* α_k *est une recherche linéaire inéxacte de Wolfe. Supposons aussi que* f *vérifie la Condition* $C1$*. Alors on a*

$$\sum_{k=0}^{\infty} \frac{\left(g_k^T d_k\right)^2}{\|d_k\|^2} < \infty \tag{4.1}$$

1.3 Utilisation du théorème de Zoutendijk pour démontrer la convergence globale

La condition $\sum_{k=0}^{\infty} \frac{\left(g_k^T d_k\right)^2}{\|d_k\|^2} < \infty$ est équivalente à $\sum_{k=1}^{\infty} \cos^2(\theta_k) \|g_k\|^2 < \infty$ où θ_k est l'angle que fait d_k avec $-g_k$. Il est évident de voir que si

$$\cos(\theta_k) \geq \delta > 0$$

pour tout k, alors on a

$$\lim_{k\to\infty} \|g_k\| = 0.$$

Dans les différentes méthodes du gradient conjugué on n'arrive pas à démontrer le résultat précédent i.e., $\lim_{k\to\infty} \|g_k\| = 0$, mais seulement

$$\liminf_{k\to\infty} \|g_k\| = 0. \tag{4.2}$$

La condition (4.2) implique qu'il existe une sous suite de $\{\|g_k\|\}$ qui tend vers zéro.

Conditions suffisantes associés au théorème de Zoutendijk pour démontrer la relation(4.2)

Pour démontrer (4.2) on associe au théorème de Zoutendijk les 2 hypothèses suivantes :

Hypothèse1

La decente suffisante est assurée, i.e., il existe une constante c indépendante de k telle que

$$g_k^T d_k \leq -c \|g_k\|^2$$

Hypothèse 2

Il existe une constante β telle que

$$\|d_k\|^2 \leq \beta k$$

On a donc

$$\left\{ \begin{array}{c} \textbf{relation de Zoutendijk } (\sum_{k=0}^{\infty} \frac{\left(g_k^T d_k\right)^2}{\|d_k\|^2} < \infty) \\ \textbf{Hypothèse1} \\ \textbf{Hypothèse2} \end{array} \right\} \Longrightarrow \left\{ \liminf_{k\to\infty} \|g_k\| = 0 \right\}$$

Remarque importante

La condition **Hypothèse1** est plus forte que la descente. En effet

$$g_k^T d_k \leq -c \|g_k\|^2 \Longrightarrow g_k^T d_k < 0.$$

La condition **Hypothèse1** n'est pas nécessaire pour avoir $\liminf_{k\to\infty} \|g_k\| = 0$. Dai et Yuan ([31]) ont montré la convergence globale seulement avec la condition de descente : $g_k^T d_k < 0$.

D'autre part la condition **Hypothèse1,** i.e. $g_k^T d_k \leq -c \|g_k\|^2$ n'est pas suffisante toute seule, pour avoir la convergence globale.

Définition 1.1 *Nous dirons qu'une méthode de gradient conjugué converge globalement si l'une des deux conditions suivantes est vérifiée*

a) $g_{k_0} = 0$, *pour un certain* $k_0 \in \mathbb{N}$

b) $\liminf_{k\to\infty} \|g_k\| = 0$.

Une autre version du théorème de Zoutendijk se trouve dans [40]. Ce sera l'objet du théorème suivant

Théorème 1.2 ([40]) *Considérons une méthode itérative quelconque générant une suite* $\{x_k\}$ *de la forme :*

$$x_{k+1} = x_k + \alpha_k d_k,$$

d_k *étant une direction de descente et* α_k *est une recherche linéaire inéxacte de Wolfe forte. Supposons aussi que* f *vérifie la Condition* $C1$*. Alors ou bien*

$$\liminf_{k\to\infty} \|g_k\| = 0.$$

ou

$$\sum_{k=0}^{\infty} \frac{\|g_k\|^4}{\|d_k\|^2} < \infty$$

2 Role joué par la direction de recherche initiale

Commençons d'abord par donner un petit aperçu sur le role joué par la direction de recherche initiale.

Il est essentiel de prendre $d_0 = -g_0$ dans un algorithme de CG. En 1972, Crowder et Wolfe [12] ont donné un exemple en 3 dimensions qui montre que le taux de convergence est linéaire si la direction de la recherche initiale n'est pas la direction de la plus grande pente, même pour une fonction quadratique fortement convexe. En 1976, Powell [58] a obtenu un résultat encore plus fort, il a montré que si la fonction objectif est une fonction quadratique convexe et si la direction de recherche initiale est une direction de descente arbitraire, alors, soit la condition d'optimalité est obtenue dans $n+1$ itérations, ou la vitesse de convergence est seulement linéaire. En outre, en analysant la relation entre x_0 et d_0, il s'ensuit que la convergence linéaire est plus fréquente que la convergence finie.

Afin de parvenir à une convergence finie pour une direction de recherche initiale arbitraire, Nazareth [53] a proposé un algorithme CG basée sur une récurrence à trois termes :

$$d_{k+1} = -y_k + \frac{y_k^T y_k}{d_k^T y_k} d_k + \frac{y_{k-1}^T y_k}{d_{k-1}^T y_{k-1}} d_{k-1} \tag{4.3}$$

avec $d_{-1} = 0$ et d_0 une direction de descente arbitraire. Si f est une fonction quadratique convexe, alors pour tout α_k, les directions de recherche générés par (4.3) sont conjugués par rapport au Hessian de f. Toutefois, cette innovation intéressante n'a pas vu une utilisation

importante dans la pratique.

3 Les méthodes dans lesquelles le terme $\|g_{k+1}\|^2$ figure dans le numérateur de β_k

Les méthodes **FR**, DY et CD ont toutes comme numérateur commun le terme $\|g_{k+1}\|^2$. Une différence fondamentale qui caractérise ces méthodes par rapport aux autres méthodes du gradient conjugué où β_k est calculé différement, est que les théorèmes de convergence globale nécessitent seulement la condition de Lipschitz : **Condition** $\mathcal{C}1$ et n'ont pas besoin de la condition de bornétude **Condition** $\mathcal{C}2$.

3.1 Méthode de Fletcher-Reeves

Cette méthode été découverte en 1964 par Fletcher et Reeves ([35]), β_k est égale à :

$$\beta_{k+1}^{FR} = \frac{\|g_{k+1}\|^2}{\|g_k\|^2} \tag{4.4}$$

3.1.1 Synthèse des principaux résultats de convergence pour la méthode de Fletcher-Reeves

Le premier résultat de convergence globale de la méthode de **FR** a été donnée par Zoutendijk [80] en 1970. Il a prouvé que la méthode Fletcher-Reeves converge globalement quand α_k est une recherche linéaire exacte , en d'autres mots. En 1977, Powell [59] a fait remarquer que la méthode de Fletcher-Reeves, avec une recherche linéaire exacte, était sensible numériquement. Autrement dit, l'algorithme pourrait prendre de nombreuses mesures courtes sans faire d'importants progrès au minimum. La mauvaise performance de la méthode de FR dans les applications a été souvent attribuée à ce phénomène de brouillage .

Le premier résultat de convergence globale de la méthode de **FR** pour une recherche linéaire inexacte a été donné par Al- Baali [1] en 1985. En utilisant les conditions de Wolfe forte avec $\sigma < \frac{1}{2}$, il a prouvé que la méthode **FR** génère des directions de descente suffisantes. Plus précisément , il a prouvé que :

$$\frac{1-2\sigma+\sigma^{k+1}}{1-\sigma} \leq \frac{-g_k^T d_k}{\|g_k\|^2} \leq \frac{1-\sigma^{k+1}}{1-\sigma},$$

pour tout $k \geq 0$. Comme conséquence, la convergence globale a été démontrée en utilisant

la condition Zoutendijk. Pour $\sigma = 1/2$, d_k est une direction de descente, toutefois, l'analyse n'a pas établi la descente suffisante.

Touati Ahmed et Story ([70]) ont généralisé ce résultat pour

$$0 \leq \beta_k \leq \beta_k^{FR} \tag{4.5}$$

Gilbert et Nocedal ([39]) ont généralisé ce résultat pour

$$|\beta_k| \leq \beta_k^{FR} \tag{4.6}$$

Dans Liu et al. [51], la preuve de la convergence d'Al-Baali est étendue au cas $\sigma = 1/2$. Dai et Yuan [21] ont montré que dans les versions **FR** consécutives, au moins une itération satisfait la propriété de descente suffisante. En d'autres termes,

$$\max\left\{\frac{-g_k^T d_k}{\|g_k\|^2}, \frac{-g_{k-1}^T d_{k-1}}{\|g_{k-1}\|^2}\right\} \geq \frac{1}{2}$$

Le théorème de Zoutendijk peut également être utilisé pour obtenir un résultat de convergence globale pour les méthodes**FR** mis en œuvre avec une recherche linéaire inexacte de Wolfe forte et $\sigma \leq 1/2$, puisque les directions de recherche sont toujours des directions de descente.

Dans [28], Dai et Yuan montrent qu'avec la méthode de **FR**, les conditions Wolfe fortes n'engendrent pas en général des directions de descente quand $\sigma > 1/2$, même pour la fonction $f(x) = \lambda \|x\|^2$, où $\lambda > 0$ est une constante. Par conséquent, la contrainte $\sigma \leq 1/2$ doit être imposée pour assurer la descente. Dans les implémentations typiques des conditions Wolfe, il est souvent plus efficace de choisir σ proche de 1. Par conséquent, la contrainte $\sigma \leq 1/2$, nécessaire pour assurer la descente, représente une restriction importante dans le choix des paramètres de la recherche linéaire. D'autre part, Dai et Yuan montrent dans [25] que, lorsque $\sigma > 1/2$ et $g_k^T d_k > 0$, $-d_k$ peut être utilisé pour une direction de recherche, et si $g_k^T d_k = 0$, alors la recherche linéaire peut être ignorée par le choix $x_{k+1} = x_k$. S'il existe une constante γ telle que $\|g_k\| \leq \gamma$, sous la condition de Lipschitz, la méthode de **FR**, avec une recherche linéaire de Wolfe et ces ajustements spéciaux quand $g_k^T d_k \geq 0$, est globalement convergente.

Dans [21], la recherche linéaire de Wolfe forte est étendue à une recherche linéaire de Wolfe. La convergence globale est obtenue lorsque $\sigma_1 + \sigma_2 \leq 1$. Pour une recherche linéaire de Wolfe forte, $\sigma_1 = \sigma_2 = \sigma$, dans ce cas, la contrainte $\sigma_1 + \sigma_2 \leq 1$ implique que $\sigma \leq 1/2$. Par conséquent, la condition $\sigma_1 + \sigma_2 \leq 1$ est plus faible que la contrainte de Wolfe forte $\sigma \leq 1/2$. Et il est possible de prendre σ_1 proche de 1, en prenant σ_2 près de 0.

3.1.2 Résultats d'El Baali

Comme on l'a remarqué auparavant le premier et le plus important résultat de convergence de la méthode Fletcher Reeves (FR), avec une recherche linéaire inexacte est celui d'El Baali ([39]). Nous exposons dans ce qui suit en détail ses résultats.

Théorème 3.1 ([1]) *Supposons que L'hypothèse* ***Condition*** $\mathcal{C}1$ *soit satisfaite. Considérons une méthode du type (3.29) et (3.30) avec* $\beta_k = \beta_k^{FR} = \frac{\|g_k\|^2}{\|g_{k-1}\|^2}$ *et le pas* α_k *satisfaisant à la règle de Wolfe forte (2.8)-(2.9) où* $\sigma \in \left]0, \frac{1}{2}\right[$. *Alors on a*

$$\frac{-1}{1-\sigma} \leq \frac{d_k^T g_k}{\|g_k\|^2} \leq \frac{2\sigma - 1}{1-\sigma}; \quad k = 1, ... \tag{4.7}$$

Preuve. La démonstration se fait par récurrence

1) Pour $k = 1$:

$$\frac{d_1^T g_1}{\|g_1\|^2} = \frac{-\|g_1\|^2}{\|g_1\|^2} = -1$$

D'autre part :

$$0 < \sigma < \frac{1}{2} \Rightarrow \begin{cases} \frac{-1}{1-\sigma} \leq -1 \\ \frac{2\sigma-1}{1-\sigma} \geq -1 \end{cases}$$

2) Supposons que (4.7) est satisfaite pour $k > 1$ et démontrons qu'elle le sera pour $k+1$:
Supposons que :

$$\frac{-1}{1-\sigma} \leq \frac{d_k^T g_k}{\|g_k\|^2} \leq \frac{2\sigma - 1}{1-\sigma}; k = 1, ... \tag{4.8}$$

On a :

$$\frac{d_{k+1}^T g_{k+1}}{\|g_{k+1}\|^2} = \frac{(-g_{k+1} + \beta_{k+1} d_k)^T g_{k+1}}{\|g_{k+1}\|^2} = -1 + \beta_{k+1} d_k^T g_{k+1}$$

D'autre part de (3.26) on aura :

$$\beta_k^{FR} = \frac{\|g_k\|^2}{\|g_{k-1}\|^2} \Rightarrow \frac{1}{\beta_{k+1}^{FR}} = \frac{\|g_k\|^2}{\|g_k + 1\|^2}$$

d'où :

$$\frac{d_{k+1}^T g_{k+1}}{\|g_{k+1}\|^2} = -1 + \frac{\beta_{k+1}}{\beta_k^{FR}} \frac{d_k^T g_{k+1}}{\|g_k\|^2} \tag{4.9}$$

En utilisant la condition de la recherche linéaire (2.9) on aura :

$$\left| d_k^T g_{k+1} \right| \leq -\sigma d_k^T g_k \Rightarrow \sigma \left| \beta_{k+1} \right| d_k^T g_k \leq \beta_{k+1} d_{k+1}^T g_{k+1} \leq -\sigma \left| \beta_{k+1} \right| d_k^T g_k$$

soit en remplaçant ceci dans (4.9), on obtient :

$$-1+\sigma\frac{|\beta_{k+1}|\, d_k^T g_k}{\beta_{k+1}^{FR}\left\|g_k\right\|^2} \leq \frac{d_{k+1}^T g_{k+1}}{\left\|g_{k+1}\right\|^2} \leq -1-\sigma\frac{|\beta_{k+1}|\, d_k^T g_k}{\beta_{k+1}^{FR}\left\|g_k\right\|^2}$$

De (4.8) on aura :

$$-1-\frac{|\beta_{k+1}|\,\sigma}{\beta_{k+1}^{FR}\,(1-\sigma)} \leq \frac{d_{k+1}^T g_{k+1}}{\left\|g_{k+1}\right\|^2} \leq -1+\frac{|\beta_{k+1}|\,\sigma}{\beta_{k+1}^{FR}\,(1-\sigma)}$$

et de (4.9) :

$$\frac{-\sigma}{1-\sigma} < \frac{|\beta_{k+1}|}{\beta_{k+1}^{FR}} < 1$$

Par conséquent on aura :

$$\frac{-1}{1-\sigma} \leq \frac{d_{k+1}^T g_{k+1}}{\left\|g_{k+1}\right\|^2} \leq \frac{2\sigma-1}{1-\sigma}$$

Ce qui achève la démonstration. ■

Corollaire 3.1 *Sous les mêmes hypothèses et conditions du Théorème précédent, on a pour tout $k \geq 1$, d_k est une direction de descente suffisante i.e., il existe une constante $C > 0$ telle que*

$$g_k^T d_k \leq -C\left\|g_k\right\|^2$$

Preuve du Corollaire

On a d'après (4.7)

$$\begin{aligned}\frac{-1}{1-\sigma} &\leq \frac{d_k^T g_k}{\left\|g_k\right\|^2} \leq \frac{2\sigma-1}{1-\sigma} \\ &\Rightarrow \quad d_k^T g_k \leq \frac{2\sigma-1}{1-\sigma}\left\|g_k\right\|^2 = -\frac{1-2\sigma}{1-\sigma}\left\|g_k\right\|^2 = -C\left\|g_k\right\|^2,\end{aligned}$$

avec $C = \frac{1-2\sigma}{1-\sigma}$ □

Théorème 3.2 ([1]) *Supposons que L'hypothèse* ***Condition*** $\mathcal{C}1$ *soit satisfaite. Considérons une méthode du type (3.29) et (3.30) avec $\beta_k = \beta_k^{FR} = \frac{\|g_k\|^2}{\|g_{k-1}\|^2}$ et le pas α_k satisfaisant à la règle de Wolfe forte (2.8)-(2.9) où $\sigma \in \left]0, \frac{1}{2}\right[$. Alors cette méthode est globalement convergente, dans le sens suivant :*

$$\lim_{k\to\infty} \inf \left\|g_k\right\| = 0 \tag{4.10}$$

Preuve. Puisque les conditions du théorème 4.2 sont satisfaites alors on a :

$$\frac{-1}{1-\sigma} \leq \frac{d_k^T g_k}{\|g_k\|^2} \leq \frac{2\sigma-1}{1-\sigma} \Rightarrow -\sigma d_{k-1}^T g_{k-1} \leq \frac{\sigma}{1-\sigma}\|g_{k-1}\|^2$$

D'autre part de (2.9)

$$\left|d_k^T g_{k+1}\right| \leq -\sigma d_k^T g_k \Rightarrow \left|d_{k-1}^T g_k\right| \leq -\sigma d_{k-1}^T g_{k-1}$$

d'où

$$\left|d_{k-1}^T g_k\right| \leq -\sigma d_{k-1}^T g_{k-1} \leq \frac{\sigma}{1-\sigma}\|g_{k-1}\|^2 \tag{4.11}$$

De (4.7), (3.26) et (4.11) :

$$\begin{aligned}
\|d_k\|^2 &= \left|\|g_k\|^2 - 2\beta_k d_{k-1}^T g_k + \beta_k^2 \|d_{k-1}\|^2\right| \\
&\leq \|g_k\|^2 + \left|2\beta_k d_{k-1}^T g_k\right| + \beta_k^2 \|d_{k-1}\|^2 \\
&\leq \left(\frac{1+\sigma}{1-\sigma}\right)\|g_k\|^2 + \left(\beta_k^{FR}\right)^2 \|d_{k-1}\|^2
\end{aligned}$$

Posons $\hat{\sigma} = \frac{1+\sigma}{1-\sigma} \geq 1$, on aura :

$$\begin{aligned}
\|d_k\|^2 &\leq \hat{\sigma}\|g_k\|^2 + \left(\beta_k^{FR}\right)^2 \|d_{k-1}\|^2 \\
&\leq \hat{\sigma}\|g_k\|^2 + \left(\beta_k^{FR}\right)^2 \left[\leq \hat{\sigma}\|g_{k-1}\|^2 + \left(\beta_k^{FR}\right)^2 \|d_{k-2}\|^2\right]
\end{aligned}$$

$$\leq \hat{\sigma}\|g_k\|^4 \sum_{j=2}^{k} \|g_j\|^{-2} + \hat{\sigma}\|g_k\|^4 \|g_1\|^{-2} = \hat{\sigma}\|g_k\|^4 \sum_{j=1}^{k} \|g_j\|^{-2} \tag{4.12}$$

Supposons que g_k est borné en dehors du zéro ($\lim_{k\to\infty} \inf \|g_k\| \neq 0$), c'est-à-dire :

$$\|g_k\| \geq \omega > 0; \forall k \Rightarrow \|g_k\|^{-2} \leq \omega^{-2}$$

de (4.12) on a :

$$\begin{aligned}
\|d_k\|^2 &\leq \hat{\sigma}\|g_k\|^4 \sum_{j=1}^{k} \|g_j\|^{-2} \leq \hat{\sigma}\frac{\lambda^4}{\omega^2} \sum_{j=1}^{k} 1 \\
&\Rightarrow \|d_k\|^2 \leq \hat{\sigma}\frac{\lambda^4}{\omega^2} k
\end{aligned}$$

d'où

$$\sum_{k\geq 1}\frac{1}{\|d_k\|^2}\geq\frac{\omega^2}{\hat{\sigma}\lambda^4}\sum_{k\geq 1}\frac{1}{k}>\infty \tag{4.13}$$

Ce qui veut dire que $\sum_{k\geq 1}\frac{1}{\|d_k\|^2}$ est divergente.

D'autre part, puisque les conditions du Théorème 4.1, et du théorème 4.2 sont satisfaites, alors on a :

$$\sum_{k\geq 1}\cos^2\theta_k\,\|g_k\|^2<\infty$$

et

$$c_1\frac{\|g_k\|}{\|d_k\|}\leq\cos\theta_k\leq c_2\frac{\|g_k\|}{\|d_k\|}$$

d'où

$$\begin{aligned}\sum_{k\geq 1}c_1^2\frac{\|g_k\|^2}{\|d_k\|^2}\|g_k\|^2 &\leq \sum_{k\geq 1}\cos^2\theta_k\,\|g_k\|^2<\infty\\ &\Rightarrow \sum_{k\geq 1}\frac{\|g_k\|^4}{\|d_k\|^2}<\infty\\ &\Rightarrow \sum_{k\geq 1}\frac{\omega^4}{\|d_k\|^2}<\infty\\ &\Rightarrow \sum_{k\geq 1}\frac{1}{\|d_k\|^2}<\infty\end{aligned}$$

Ce qui contredit (4.13), d'où le résultat. Donc

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

■

Remarque 3.1 *La méthode de Fletcher-Reeves avec une recherche linéaire exacte génère des directions de descente.*

En effet, à chaque itération $k\geq 1$, on a :

$$\begin{aligned}d_{k+1}^T g_{k+1} &= \left(-g_{k+1}+\beta_{k+1}^{FR}d_k\right)^T g_{k+1}\\ &= -g_{k+1}^T g_{k+1}+\beta_{k+1}^{FR}d_k^T g_{k+1}\\ &= -\|g_{k+1}\|^2\end{aligned}$$

Puisque

$$\alpha_k = \min_{\alpha>0} f(x_k + \alpha d_k) = \min_{\alpha>0} h_k(\alpha)$$

Donc α_k vérifie la condition nécessaire d'optimalité : $h'_k(\alpha_k) = d_k^T \nabla f(x_k + \alpha_k d_k) = d_k^T g_{k+1} = 0, \quad \forall k \geq 1$

3.2 Méthode de descente conjuguée

Cette méthode été découverte en 1987 par Fletcher ([34]), β_k est égale à :

$$\beta_{k+1}^{CD} = \frac{\|g_{k+1}\|^2}{-d_k^T g_k} \tag{4.14}$$

La méthode de descente conjuguée **CD** proposée par Fletcher [34] est étroitement liée à la méthode Fletcher Reeves FR. Avec une recherche linéaire exacte, $\beta_k^{FR} = \beta_k^{CD}$. Une différence importante entre FR et **CD** est que, avec **CD**, la descente suffisante est assurée pour une recherche linéaire de Wolfe forte et la contrainte $\sigma \leq 1/2$ avec FR, n'est pas nécessaire pour la méthode **CD**. En outre, pour une recherche linéaire qui satisfait aux conditions de Wolfe relaxée (2.10-2.11) avec $\sigma_1 \leq 1$ et $\sigma_2 = 0$, il peut être démontré que :

$0 \leq \beta_k^{CD} \leq \beta_k^{FR}$.

Par conséquent, à partir de l'analyse [1] ou le théorème de Zoutendijk bis, la convergence globale est assurée. D'autre part, si $\sigma_1 \geq 1$ ou $\sigma_2 > 0$, Dai et Yuan [22] construisent des exemples où $\|d_k\|^2$ augmente de façon exponentielle et le procédé de CD converge vers un point où le gradient ne s'annule pas. En particulier, la méthode **CD** peut ne pas converger vers un point stationnaire pour une recherche linéaire de Wolfe forte .

Théorème 3.3 ([22]) *Supposons que l'hypothèse **Condition** $\mathcal{C}1$ soit satisfaite. Considérons une méthode du type (3.29) et (3.30) avec $\beta_k = \beta_k^{CD} = \frac{\|g_{k+1}\|^2}{-d_k^T g_k}$ et le pas α_k satisfait aux conditions de Wolfe relaxée (2.10)-(2.11) :*

$$\begin{aligned} f(x_k + \alpha_k d_k) &\leq f(x_k) + \rho d_k^T g_k \\ et \;\; \sigma_1 d_k^T g_k &\leq d_k^T g_{k+1} \leq -\sigma_2 d_k^T g_k \end{aligned}$$

avec $0 < \rho < \sigma_1 < 1$

et $0 \leq \sigma_2 \leq 1$. *Alors la méthode génère des directions de descente suffisante à chaque itération* $k \geq 1$.

Preuve. On a

$$\begin{aligned} -d_k^T g_k &= -\left(-g_k + \beta_k^{CD} d_{k-1}\right)^T g_k \\ &= \|g_k\|^2 \left[1 + \frac{d_{k-1}^T g_k}{d_{k-1}^T g_{k-1}}\right] \\ \Rightarrow \quad \frac{-d_k^T g_k}{\|g_k\|^2} &= 1 + \frac{d_{k-1}^T g_k}{d_{k-1}^T g_{k-1}} \end{aligned}$$

D'autre part de (2.11), on a

$$\begin{aligned} \sigma_1 d_k^T g_k &\leq d_k^T g_{k+1} \leq -\sigma_2 d_k^T g_k \\ \Rightarrow \quad 1 - \sigma_2 &\leq 1 + \frac{d_{k-1}^T g_k}{d_{k-1}^T g_{k-1}} \leq 1 + \sigma_1 \end{aligned}$$

d'où

$$1 - \sigma_2 \leq \frac{-d_k^T g_k}{\|g_k\|^2} \leq 1 + \sigma_1$$

Donc si $\|g_k\| \neq 0$, on a :

$$d_k^T g_k \leq -C \|g_k\|^2 \quad \text{où } C = 1 - \sigma_2 > 0$$

et donc d_k est une direction de descente suffisante. ∎

Théorème 3.4 ([22]) *Supposons que L'hypothèse* ***Condition*** $\mathcal{C}1$ *soit satisfaite. Considérons une méthode du type (3.29) et (3.30) avec* $\beta_k = \beta_k^{CD} = \frac{\|g_{k+1}\|^2}{-d_k^T g_k}$ *et le pas* α_k *satisfait aux conditions de Wolfe relaxée (2.10)-(2.11) :*

$$\begin{aligned} f(x_k + \alpha_k d_k) &\leq f(x_k) + \rho d_k^T g_k \\ et \quad \sigma_1 d_k^T g_k &\leq d_k^T g_{k+1} \leq -\sigma_2 d_k^T g_k \end{aligned}$$

avec $0 < \rho < \sigma_1 < 1$

et $\sigma_2 = 0$; *est globalement convergente, dans le sens suivant :*

$$\lim_{k \to \infty} \inf \|g_k\| = 0$$

Preuve. Du théorème précédent on a :

$$\begin{aligned}
1-\sigma_2 &\leq \frac{-d_k^T g_k}{\|g_k\|^2} \leq 1+\sigma_1 \\
&\Rightarrow 1 \leq \frac{-d_k^T g_k}{\|g_k\|^2} \leq 1+\sigma_1 \\
&\Rightarrow (1+\sigma_1)^{-1} \leq \frac{\|g_k\|^2}{-d_k^T g_k} \leq 1 \\
&\Rightarrow (1+\sigma_1)^{-1} \leq \frac{\|g_{k+1}\|^2 \|g_k\|^2}{-d_k^T g_k \|g_{k+1}\|^2} \leq 1 \\
&\Rightarrow (1+\sigma_1)^{-1} \leq \frac{\beta_{k+1}^{CD}}{\beta_{k+1}^{FR}} \leq 1 \\
&\Rightarrow \beta_{k+1}^{CD} \leq \beta_{k+1}^{FR}
\end{aligned}$$

Donc β_{k+1}^{CD} vérifie l'inégalité (4.6).

D'après le théorème 4.2 on a :

$$\lim_{k\to\infty} \inf \|g_k\| = 0$$

■

3.3 Méthode de Dai-Yuan

3.3.1 Introduction

Cette méthode été découverte par Y. H. Dai et Y. Yuan ([24]) en 1999, β_k est égale à :

$$\beta_k^{DY} = \frac{\|g_{k+1}\|^2}{d_k^T y_k}; \quad y_k = g_{k+1} - g_k \tag{4.15}$$

Cette méthode est fondamentalement différente de la méthode Fletcher Reeves et aussi de la méthode CD. Avec une recherche linéaire de Wolfe standard (2.6, 2.7), la méthode de **DY** génère toujours des directions de descente. En plus elle est globalement convergente aves des hypothèses très générales sur la fonction objectif f. On exige seulement que f soit continueument différentiable et que le gradient soit Lipschitzien, c'est à dire que f vérifie l'hypothèse **Condition** $\mathcal{C}1$.

3.3.2 Théorèmes de convergence de Dai et Yuan exigeant seulement la condition de Wolfe et la condition de Lipschitz $C1$

Théorème 3.5 ([31]) *Supposons que la condition de Lipschitz $C1$ soit satisfaite. Considérons des méthodes du type (3.29) et (3.30) où β_k satisfait à $\beta_k^{DY} = \frac{\|g_{k+1}\|^2}{d_k^T y_k}$; $y_k = g_{k+1} - g_k$ et le pas α_k satisfait aux conditions de Wolfe faible :*

$$\begin{aligned} f(x_k + \alpha_k d_k) &\leq f(x_k) + \rho d_k^T g_k \\ d_k^T g_{k+1} &\geq \sigma d_k^T g_k \end{aligned}$$

avec $0 < \rho < \sigma < 1$. Alors toutes les directions générées par ces méthodes sont de descente, autrement dit :

$$d_k^T g_k < 0; \quad \forall k \geq 1 \tag{4.16}$$

Preuve. La démonstration se fait par récurrence.

1) Pour $k = 1$:

$$d_1^T g_1 = -\|g_1\|^2 < 0$$

2) Supposons que (4.16) est satisfaite pour $k > 1$ et démontrons qu'elle le sera pour $k + 1$:

Supposons que :

$$d_k^T g_k < 0; \quad k > 1$$

En utilisant (2.7), on aura :

$$d_k^T y_k = d_k^T (g_{k+1} - g_k) > d_k^T (g_{k+1} - g_k) = (\sigma - 1) d_k^T g_k = -(1 - \sigma) d_k^T g_k > 0$$

D'autre part :

$$\begin{aligned} d_{k+1}^T g_{k+1} &= \left(-g_{k+1} + \beta_{k+1}^{DY} d_k\right)^T g_{k+1} \\ &= -\|g_{k+1}\|^2 + \beta_{k+1}^{DY} d_k^T g_{k+1} \\ &= -\|g_{k+1}\|^2 + \frac{\|g_{k+1}\|^2}{d_k^T y_k} d_k^T g_{k+1} \\ &= -\|g_{k+1}\|^2 + \frac{\|g_{k+1}\|^2}{d_k^T y_k} d_k^T (y_k + g_k) \\ &= -\|g_{k+1}\|^2 + \|g_{k+1}\|^2 + \frac{\|g_{k+1}\|^2}{d_k^T y_k} d_k^T g_k \\ &= \frac{\|g_{k+1}\|^2}{d_k^T y_k} d_k^T g_k \end{aligned}$$

or puisque :$d_k^T g_k < 0$; $d_k^T y_k > 0$; il en résulte :

$$d_{k+1}^T g_{k+1} < 0$$

Ce qui achève la démonstration. ■

Théorème 3.6 ([24]) *Supposons que la condition de Lipschitz $C1$ soit satisfaite. Considérons des méthodes du type (3.29) et (3.30) où β_k satisfait à $\beta_k^{DY} = \frac{\|g_{k+1}\|^2}{d_k^T y_k}$; $y_k = g_{k+1} - g_k$ et le pas α_k satisfait aux conditions de Wolfe faible :*

$$\begin{aligned} f(x_k + \alpha_k d_k) &\leq f(x_k) + \rho d_k^T g_k \\ d_k^T g_{k+1} &\geq \sigma d_k^T g_k \end{aligned}$$

avec $0 < \rho < \sigma < 1$.. La suite $\{x_k\}$ générée par l'algorithme de Dai-Yuan converge globalement i.e.

$$\lim_{k\to\infty} \inf \|g_k\| = 0 \tag{4.17}$$

Preuve. En utilisant le théorème précédent, on aura :

$$\sum_{k\geq 1} \cos^2\theta \|g_k\|^2 < \infty \tag{4.18}$$

d'autre part on a :

$$\begin{aligned} \|d_{k+1}+g_{k+1}\|^2 &= \left\|\beta_{k+1}^{DY} d_k\right\|^2 \\ &\Rightarrow \|d_{k+1}\|^2 = \left(\beta_{k+1}^{DY}\right)^2 \|d_k\|^2 - 2d_{k+1}^T g_{k+1} - \|g_{k+1}\|^2 \; 4.19 \end{aligned} \qquad \text{(IV.1)}$$

De (3.30) :

$$\begin{aligned} d_{k+1}^T g_{k+1} &= \left(-g_{k+1}+\beta_{k+1}^{DY} d_k\right)^T g_{k+1} \\ &= \frac{\|g_{k+1}\|^2}{d_k^T y_k} d_k^T g_k = \beta_{k+1}^{DY} d_k^T g_k \\ &\Rightarrow \beta_{k+1}^{DY} = \frac{d_{k+1}^T g_{k+1}}{d_k^T g_k} \end{aligned}$$

remplaçant ceci dans (4.19), on aura :

$$\begin{aligned} \frac{\|d_{k+1}\|^2}{\left(d_{k+1}^T g_{k+1}\right)^2} &= \frac{\left(\beta_{k+1}^{DY}\right)^2 \|d_k\|^2}{\left(d_{k+1}^T g_{k+1}\right)^2} - \frac{2d_{k+1}^T g_{k+1}}{\left(d_{k+1}^T g_{k+1}\right)^2} - \frac{\|g_{k+1}\|^2}{\left(d_{k+1}^T g_{k+1}\right)^2} \\ &= \frac{\|d_k\|^2}{(d_k^T g_k)} - \left[\frac{1}{\|g_{k+1}\|^2} + 2\frac{1}{d_{k+1}^T g_{k+1}} - \frac{\|g_{k+1}\|^2}{\left(d_{k+1}^T g_{k+1}\right)^2}\right] + \frac{1}{\|g_{k+1}\|^2} \\ &= \frac{\|d_k\|^2}{(d_k^T g_k)} - \left[\frac{1}{\|g_{k+1}\|} + \frac{\|g_{k+1}\|}{d_{k+1}^T g_{k+1}}\right]^2 + \frac{1}{\|g_{k+1}\|^2} \\ &\leq \frac{\|d_k\|^2}{(d_k^T g_k)} + \frac{1}{\|g_{k+1}\|^2} \end{aligned}$$

d'où

$$\begin{aligned} \frac{\|d_k\|^2}{(d_k^T g_k)^2} &\leq \frac{1}{\|g_k\|^2} + \frac{\|d_{k-1}\|^2}{\left(d_{k-1}^T g_{k-1}\right)} \\ &\leq \frac{1}{\|g_k\|^2} + \frac{1}{\|g_{k-1}\|^2} + \frac{\|d_{k-2}\|^2}{\left(d_{k-2}^T g_{k-2}\right)} \\ &\cdot \\ &\leq \sum_{i=1}^{k} \frac{1}{\|g_i\|^2} \end{aligned}$$

Supposons maintenant que (4.18) n'est pas satisfaite, autrement dit :

$$\exists \omega > 0 \ \text{ tel que } \ \|g_k\| > \omega; \forall k$$

On aura :

$$\begin{aligned} \frac{\|d_k\|^2}{\left(d_k^T g_k\right)^2} &\leq \frac{1}{\|g_i\|^2} \leq \frac{1}{\omega^2} \sum_{i=1}^{k} 1 = \frac{1}{\omega^2} k \\ &\Rightarrow \sum_{k\geq 1} \frac{\left(d_k^T g_k\right)^2}{\|d_k\|^2} \geq \omega^2 \sum_{k\geq 1} \frac{1}{k} \end{aligned}$$

d'où

$$\sum_{k\geq 1} \frac{\left(d_k^T g_k\right)^2}{\|d_k\|^2} = \infty$$

ce qui contredit (4.18).Ceci achève la démonstration. ■

3.3.3 Résultat de Dai et Yuan sur la descente suffisante

Dans [16], Dai a analysé la méthode de **DY** de façon plus approfondie. Le résultat suivant donne des conditions suffisantes pour obtenir des directions de descente suffisantes.

Théorème 3.7 ([22]) *Considérons une méthode du type (3.29) et (3.30) avec $\beta_k = \beta_k^{DY}$. Si la méthode de DY est mise en œuvre avec une recherche linéaire quelconque garantissant que les directions de recherche soient des directions de descente, et s' il existe des constantes γ_1 et γ_2 telles que $\gamma_1 \leq \|g_k\| \leq \gamma_2$ pour tous $k \geq 0$, alors pour tout $p \in]0,1[$, il existe une constante $c > 0$ telle que la condition de descente suffisante suivante*

$$g_i^T d_i \leq -c \|g_i\|^2$$

soit assurée pour au moins $[pk]$ indices, $i \in [0,k]$, $[r]$ désigne le plus grand entier inférieur ou égal à r.

3.4 Méthode de Dai-Yuan généralisée

Dans le procédé de l'analyse de la méthode de **DY**, Dai et Yuan ont généralisé leurs résultats pour toutes les méthodes du gradient conjugué où β_k peut se mettre sous la forme suivante :

$$\beta_k = \beta_k^{DYG} = \frac{\Phi_{k+1}}{\Phi_k} \tag{4.20}$$

Le procédé de FR correspond au choix $\Phi_k = \|g_k\|^2$. En utilisant (1.3), β_k^{DY} peut être réécrit sous la forme :

$$\beta_k^{DY} = \frac{g_{k+1}^T d_{k+1}}{g_k^T d_k}$$

Par conséquent, le procédé de DY à la forme (DYG) avec $\Phi_k = g_k^T d_k$. Le résultat suivant a été établi par Dai et Yuan dans [29, 31] :

Théorème 3.8 ([36]) *Considérons une méthode du type (3.29) et (3.30) avec* $\beta_k = \beta_k^{DYG} = \frac{\Phi_{k+1}}{\Phi_k}$, *$d_k$ est une direction de descente pour tout k. Supposons aussi que l'hypothèse Condition $C1$ soit satisfaite. Si la relation de Zoutendijk* ($\sum_{k=0}^{\infty} \frac{\left(g_k^T d_k\right)^2}{\|d_k\|^2} < \infty$) *est vérifiée et si l'une des trois relations suivantes*

$$\sum_{k=0}^{\infty} \frac{\left(g_k^T d_k\right)^2}{\Phi_k^2} = \infty \quad ou \quad \sum_{k=0}^{\infty} \frac{\|g_k\|^2}{\Phi_k^2} = \infty \quad ou \quad \sum_{k=1}^{\infty} \prod_{i=1}^{k} \beta_i^{-2} = \infty$$

a lieu, alors la suite générée est globalement convergente, i.e.,

$$\lim_{k\to\infty} \inf \|g_k\| = 0.$$

3.4.1 Applications du théorème à la méthode de Dai Yuan

Comme corollaire de ce résultat, la méthode de DY est globalement convergente, si elle est mise en œuvre avec une recherche linéaire inexacte de Wolfe. En effet, on a $\Phi_k = g_k^T d_k$. Par conséquent

$$\sum_{k=0}^{N} \frac{\left(g_k^T d_k\right)^2}{\Phi_k^2} = N+1 \Longrightarrow \sum_{k=0}^{\infty} \frac{\left(g_k^T d_k\right)^2}{\Phi_k^2} = \lim_{n\to\infty} \sum_{k=0}^{N} \frac{\left(g_k^T d_k\right)^2}{\Phi_k^2} = \infty$$

3.4.2 Applications du théorème à la méthode de Fletcher Reeves

la méthode de FR est globalement convergente, si elle est mise en œuvre avec une recherche linéaire inexacte de Wolfe forte avec $\sigma \leq 1/2$. En effet, on a $\Phi_k = \|g_k\|^2$. Par conséquent

$$\sum_{k=0}^{N} \frac{\|g_k\|^2}{\Phi_k^2} = N+1 \Longrightarrow \sum_{k=0}^{\infty} \frac{\|g_k\|^2}{\Phi_k^2} = \lim_{n\to\infty} \sum_{k=0}^{N} \frac{\|g_k\|^2}{\Phi_k^2} = \infty$$

Notez qu'une méthode générale de gradient conjugué peut être exprimée sous la forme (4.13) en prenant $\Phi_0 = 1$, et

$$\Phi_k = \prod_{j=1}^{k} \beta_j \ , \ \text{ pour } k > 0$$

4 Les méthodes où $g_{k+1}^T y_k$ figure dans le numérateur de β_k

4.1 Méthode de Polak-Ribière-Polyak

4.1.1 Introduction

Cette méthode a été proposée en 1969 par Polak et Ribière ([56]) et Polyak ([57]), β_k est égale à :

$$\beta_k^{PRP} = \frac{g_{k+1}^T y_k}{\|g_k\|^2}; \ \ y_k = g_{k+1} - g_k \tag{4.21}$$

Dans [55] la convergence globale de la méthode **PRP** fut établie lorsque f est fortement convexe et la recherche linéaire est exacte. Pour une fonction non linéaire, Powell [59] a montré que la méthode **PRP** est globalement convergente si

(a) la taille du pas $s_k = x_{k+1} - x_k$ tend vers zéro,

(b) la recherche en ligne est exacte

(c) la condition de Lipshitz $C1$ est vérifiée

D'autre part, Powell a montré plus tard [60], en utilisant un contre exemple à 3 dimensions, que, avec une recherche linéaire exacte, la méthode **PRP** pourrait avoir un cycle infini, sans converger vers un point stationnaire. Par conséquent, il est nécessaire d'imposer la condition (a) pour avoir la convergence.

Dans le cas où la direction de recherche est une direction de descente, Yuan [76] a établi la convergence globale de la méthode **PRP** pour les fonctions fortement convexes associées à une

recherche linéaire inexacte de Wolfe standard (2.6, 2.7). Cependant, pour une recherche linéaire inexacte de *Wolfe forte (2.8)-(2.9)*, Dai [15] a donné un contre exemple qui montre que, même lorsque la fonction objectif est fortement convexe et $\sigma \in (0,1)$ est susamment petit, la méthode **PRP** peut encore générer des directions de recherche d_k ascendantes i.e., $g_k^T d_k > 0$.

En résumé, la convergence de la méthode de **PRP** pour une fonction non linéaire est incertaine ; l'exemple de Powell montre que lorsque la fonction n'est pas fortement convexe, la méthode **PRP** peut ne pas converger, même avec une recherche linéaire exacte. L'exemple de Dai montre que même pour une fonction fortement convexe, la méthode **PRP** peut ne pas générer une direction de descente avec une recherche linéaire inexacte.

4.1.2 Modification de la méthode PRP en agissant sur le coéfficient β_k

Sur la base des connaissances tirées de l'exemple de Powell, Dai a suggéré [60] la modification suivante dans le paramètre de mise à jour pour la méthode **PRP** :

$$\beta_k^{PRP+} = \max\left\{\beta_k^{PRP}, 0\right\}. \tag{4.22}$$

Dans [39] Gilbert et Nocedal ont prouvé la convergence de la méthode **PRP +**.

L'analyse de Gilbert et Nocedal s'applique à une classe d'algorithmes du gradient conjugué qui possèdent la propriété suivante :

Considérons une méthode itérative de la forme *(3.29) et (3.30)*, et supposons la suite des gradients $\{g_k\}$ vérifie la condition suivante :

il existe deux constantes positives γ et $\overline{\gamma}$ telles ques : $\forall k \in \mathbb{N} : 0 < \gamma \leq \|g_k\| \leq \overline{\gamma}$

Définition de la propriété (★)

On dit que la propriété (★) est vérifiée s'il existe des constantes $b > 1$ et $\lambda > 0$ telle que :

$$\forall k \in \mathbb{N} : |\beta_k| \leq b \quad \text{et} \quad \|s_k\| \leq \lambda \quad \text{implique} \quad |\beta_k| \leq \frac{1}{2b}$$

Le résultat suivant est prouvé dans [39]

Théorème 4.1 ([39]) *Considérons une méthode itérative du gradient conjugué de la forme (3.29) - (3.30), qui satisfait aux conditions suivantes :*

(a) $\beta_k \geq 0$

(b) Les directions de recherche sont des directions de descente suffisante i.e., il existe une

constante $C > 0$ telle que

$$g_k^T d_k \leq -C \left\| g_k \right\|^2 : k = 1, 2, ... \tag{4.23}$$

(c) La condition de Zoutendijk $(\sum_{k=0}^{\infty} \frac{\left(g_k^T d_k\right)^2}{\left\| d_k \right\|^2} < \infty)$ *est vérifiée*

(d) La propriété (★) est vérifiée.

(e) La condition de Lipchitz $C1$ et la condition de bornetude $C2$ sont assurées.

Alors la suite générée par cet algorithme est globalement convergente i.e.

$$\liminf_{k \to \infty} \left\| g_k \right\| = 0$$

Comme corollaire de ce résultat, Si on associe à la méthode PRP+, des directions de recherche satisfaisant la condition de descente suffisante DS et des recherche linéaires de Wolfe faible, alors la méthode PRP + est globalement convergente.

Dans [23], Dai et Yuan ont donné des exemples pour montrer que la *la condition de bornetude* $C2$ est nécessaire pour obtenir la convergence globale i.e., $\liminf_{k \to \infty} \left\| g_k \right\| = 0$. D'autre part, la contrainte $\beta_k \geq 0$ ne peut pas être affaiblie par $\max \left\{ \beta_k^{PRP}, -\varepsilon \right\}$ pour tout choix de $\varepsilon > 0$.

Dans [18], il est montré que la condition de descente suffisante dans le théorème précédent, peut être affaiblie à une condition de descente simple du type $g_k^T d_k < 0$, si on utilise une recherche linéaire inexacte de Wolfe forte *(2.8)-(2.9)*.

4.1.3 Modification de la méthode PRP en agissant sur le pas de la recherche linéaire α_k

La méthode PRP+ a été introduite pour remédier à la défaillance de la convergence de la méthode PRP lorsque elle est mise en œuvre avec une recherche linéaire inexacte de Wolfe. Une autre approche pour corriger l'échec de convergence, est de conserver la formule de mise à jour PRP, mais modifier la recherche linéaire elle même. Suivant cette voie, Grippo et Lucidi [40] ont proposé une nouvelle recherche linéaire de type Armijo de la forme suivante :

$$\alpha_k = \max \left\{ \lambda^j \frac{\tau \left| g_k^T d_k \right|}{\left\| d_k \right\|^2} \right\}$$

où $j \geq 0$ est le plus petit entier possedant la propriété suivante

$$f(x_{k+1}) \leq f(x_k) - \delta \alpha_k^2 \left\| d_k \right\|^2, \tag{4.24}$$

et

$$-c_1 \|g_{k+1}\|^2 \leq g_{k+1}^T d_{k+1} \leq c_2 \|g_{k+1}\|^2 \tag{4.25}$$

où $0 < c_2 < 1 < c_1$, $0 < \lambda < 1$ et $\tau > 0$ sont des constantes. Avec cette nouvelle recherche linéaire, ils prouvent [41] la convergence globale de la méthode **PRP**. Dans un document plus récent [42], ils combinent la recherche linéaire précédente avec une technique de «région de confiance".

Dans une autre voie de recherche, il est montré dans [28] que la méthode **PRP** est globalement convergent lorsque la recherche linéaire utilise un pas de progression constant : $\alpha_k = \eta < 1/4L$, où L est une constante de Lipschitz associée à f. Dans [64] Sun et Zhang donnent un résultat de convergence globale avec $\alpha_k = -\delta \frac{g_k^T d_k}{d_k^T Q_k d_k}$, où Q_k est une matrice définie positive dont la plus petite valeur propre $\nu_{\min}$ vérifie $\nu_{\min} > 0,\ \delta \in (0, \nu_{\min}/L)$, L est la constante de Lipschitz associée à ∇f.

Pour ces choix de pas, les directions de recherche ne sont plus conjugués lorsque f est quadratique. Par conséquent, ces méthodes doivent être considérées comme des méthodes de plus grande pente, plutôt que des méthodes de gradient conjugué.

4.2 Méthode de Hestenes et Stiefel HS

4.2.1 Introduction

Cette méthode a été proposée en 1952, dans sa version linéaire par Hestenes et Steifel ([48]). C'est d'ailleurs le premier article publié qui parle de la méthode du gradient conjugué. Les auteurs utlisent cette méthode itérative pour résoudre des systèmes linéaires, β_k est égale à :

$$\beta_k^{HS} = \frac{g_{k+1}^T y_k}{d_k^T y_k};\quad y_k = g_{k+1} - g_k \tag{4.26}$$

La méthode **HS** possède la propriété de conjugaison suivante

$$d_{k+1}^T y_k = 0 \tag{4.27}$$

indépendamment de la recherche linéaire.

4.2.2 Convergence de la méthode HS

Pour une recherche linéaire exacte, $\beta_k^{HS} = \beta_k^{PRP}$. Par conséquent, les propriétés de convergence de la méthode de **HS** doivent être similaires aux propriétés de convergence de la méthode

PRP. En particulier, si on prend en considération l'exemple de Powell [60], la méthode **HS** associée à une recherche linéaire exacte, peut ne pas converger pour une fonction non linéaire. Il est facile de vérifier que si les directions de recherche satisfont à la condition de descente suffisante et si on utilise une recherche linéaire Wolfe, alors la méthode de HS satisfait la propriété (★).

Comme la méthode PRP +, si nous notons

$$\beta_k^{HS+} = \max\left\{\beta_k^{HS}, 0\right\}, \tag{4.28}$$

il résulte du théorème 5.1, que la méthode HS+ est globalement convergente.

4.3 Méthode de Liu et Storey LS

La méthode Liu et Storey LS est également identique à la méthode PRP pour une recherche de ligne exacte. Bien que peu de recherches ont été faites sur ce choix pour le paramètre de mise à jour, à l'exception de l'article [52], nous nous attendons à ce que les techniques développées pour l'analyse de la méthode PRP devraient s'appliquer à la méthode **LS**.

5 Méthodes hybrides et familles paramétriques du gradient conjugué

5.1 Les méthodes hybrides

5.1.1 Méthodes hybrides utilisant FR et PRP

Comme nous l'avons vu, la première série de méthodes **FR**, **DY** et **CD** ont des propriétés de convergence remarquables, mais ils ne sont pas assez performants en pratique à cause du phénomène du brouillage. D'autre part, comme on l'a remarqué, le second ensemble de méthodes **PRP**, **HS**, **LS** peuvent ne pas converger. Cependant numériquement et partiquement, elles sont plus performantes que les méthodes **FR**, **DY** et **CD.** Par conséquent, des combinaisons des deux types de méthodes ont été proposées pour tenter d'exploiter les caractéristiques intéressantes de chaque ensemble. Touati-Ahmed et Storey [70] ont proposé la méthode hybride qui suit :

$$\beta_k = \begin{cases} \beta_k^{PRP} & 0 \leq \beta_k^{PRP} \leq \beta_k^{FR} \\ \beta_k^{FR} & \text{sinon} \end{cases}$$

Ainsi, lorsque les itérations coincent ou bloquent, le paramètre de mise à jour PRP est

utilisé. Par les mêmes motivations, Hu et Storey [49] ont proposé

$$\beta_k = \max\left\{0, \min\left\{\beta_k^{PRP}, \beta_k^{FR}\right\}\right\}$$

Dans [39], il est précisé que β_k^{PRP} peut être négatif, même pour des fonctions fortement convexes. Nocedal et Gilbert [39] ont proposé

$$\beta_k = \max\left\{-\beta_k^{FR}, \min\left\{\beta_k^{PRP}, \beta_k^{FR}\right\}\right\}$$

Avec ce procédé hybride, β_k peut être négatif car β_k^{FR} est toujours positif ou nul. On notera que dans les résultats numériques présentés dans [38], les performances de ce procédé hybride n'étaientt pas meilleure que celles de PRP+.

Les résultats de convergence pour les méthodes du gradient conjugué qui peuvent être bornés en fonction de la méthode FR sont développés dans [39, 47, 49]. En particulier, le résultat suivant est établi dans [47] :

Théorème 5.1 ([47]) *Considérons une méthode itérative du gradient conjugué de coéfficient β_k de la forme (3.29) - (3.30), qui utilise une recherche linéaire inexacte de Wolfe forte (2.8)-(2.9) avec $\sigma \leq 1/2$. Supposons aussi que l'hypothèse de Lipschitz $C1$ soit vérifiée et qu'on a $2\sigma|\beta_k| \leq \beta_k^{FR}$. Alors les directions de recherche engendrées sont toujours les directions de descente et la suite associée à cet algorithme est globalement convergente i.e.*

$$\liminf_{k\to\infty} \|g_k\| = 0$$

Remarque

Dans le théorème précédent, avec la condition $2\sigma|\beta_k| \leq \beta_k^{FR}$, on a obtenu la descente et la convergence globale. Dans [39, 49], on a imposé une condition plus forte, en l'occurence $2\sigma|\beta_k| < \beta_k^{FR}$ et on a obtenu un résultat plus fort, la convergence globale et la descente suffisante.

5.1.2 Méthodes hybrides utilisant DY et HS

Rappelons que la méthode de DY possède des résultats de convergence globale plus performantes que celles de FR. Par conséquent, Dai et Yuan [30] ont étudié la possibilité de combiner DY avec d'autres méthodes CG. Utilisant une recherche linéaire de Wolfe et pour $\beta_k \in \left[-\eta\beta_k^{DY}, \beta_k^{DY}\right]$, où $\eta = (1-\sigma)(1+\sigma)$, ils démontrent la convergence globale lorsque la condition de Lipschitz $C1$ est assurée.

Les deux méthodes hybrides suivantes ont été proposés dans [30] :

$$\beta_k = \max\left\{-\left(\frac{1-\sigma}{1+\sigma}\right)\beta_k^{DY}, \min\left\{\beta_k^{HS}, \beta_k^{DY}\right\}\right\}$$

et

$$\beta_k = \max\left\{0, \min\left\{\beta_k^{HS}, \beta_k^{DY}\right\}\right\}$$

Les expériences numériques dans [20] indiquent que la seconde méthode hybride a donné des résultats plus performants que la méthode PRP +.

Une autre méthode hybride a été proposée par Dai dans [17]. Elle emploie soit le modèle de DY ou celui de CD :

$$\beta_k = \frac{\|g_{k+1}\|^2}{\max\left\{d_k^T y_k, -g_k^T d_k\right\}}$$

Il montre que ce système hybride génère des directions de descente, indépendamment de la recherche linéaire. Cette propriété de descente est plus forte que celle du modèle de DY lui-même, où la descente est valable pour une recherche linéaire de Wolfe. Dai montre que pour ce système hybride, $\beta_k \in \left[0, \beta_k^{DY}\right]$. Cette propriété, ainsi que celle de la descente du modèle hybride, implique la convergence des méthodes pour des recherche linéaires typiques.

5.2 Les méthodes unifiées

Les méthodes quasi-Newtonniennes ont été regrouppées en familles dites de Broyden [9] et ont été étudiées ensemble. De la même manière, Dai et Yuan [26, 31], ont unifié plusieurs méthodes du gradient conjugué et ont proposé une famille à un paramètre des méthodes du gradient conjugué en posant

$$\beta_k^{DY-unif} = \frac{\|g_{k+1}\|^2}{\lambda_k \|g_k\|^2 + (1-\lambda_k) d_k^T y_k} \tag{4.29}$$

où $\lambda_k \in [0,1]$ est un paramètre. Les méthodes unifiées sont en fait une combinaison convexe des méthodes FR et DY. Le procédé de FR correspond à $\lambda_k = 1$, tandis que la méthode de DY correspond à $\lambda_k = 0$. Dans [27], cette famille est étudiée en considérant $\lambda_k \in]-\infty, +\infty[$. Si la condition de Lipschitz $C1$ est vérifiée et si on utilise une recherche linéaire de Wolfe avec

$$\sigma_1 - 1 \leq (\sigma_1 + \sigma_2)\lambda_k \leq 1$$

alors on a la convergence globale pour chaque membre de la famille.

En considérant des combinaisons convexes des numérateurs et des dénominateurs de β_k^{FR} et β_k^{HS}, Nazareth [54] propose et de façon indépendante, une famille à deux paramètres des méthodes du gradient conjugué avec :

$$\beta_k^{N-unif} = \frac{\mu_k \left\|g_{k+1}\right\|^2 + (1-\mu_k)\, g_{k+1}^T y_k}{\lambda_k \left\|g_k\right\|^2 + (1-\lambda_k)\, d_k^T y_k} \tag{4.30}$$

où $\lambda_k, \mu_k \in [0,1]$. Si on attribue à λ_k et μ_k les valeurs extremes 0 ou 1, on obtient les méthodes du gradient conjugué FR, DY, PRP, et SH. Par exemple

$$\begin{aligned}
\lambda_k &= 1, \mu_k = 0 \Longrightarrow \beta_k^{N-unif} = \frac{g_{k+1}^T y_k}{\left\|g_k\right\|^2} = \beta_k^{PRP} \\
\lambda_k &= 0, \mu_k = 1 \Longrightarrow \beta_k^{N-unif} = \frac{\left\|g_{k+1}\right\|^2}{d_k^T y_k} = \beta_k^{DY} \\
\lambda_k &= 1, \mu_k = 1 \Longrightarrow \beta_k^{N-unif} = \frac{\left\|g_{k+1}\right\|^2}{\left\|g_k\right\|^2} = \beta_k^{FR}
\end{aligned}$$

Constatant que les six méthodes du gradient conjugué qu'on a vu précédement ont deux numérateurs et trois dénominateurs, Dai et Yuan [29] ont introduit une nouvelle famille de gradients conjugués encore plus large à trois paramètres, ils ont choisi β_k comme suit :

$$\beta_k = \beta_k^{DY3} = \frac{\mu_k \left\|g_{k+1}\right\|^2 + (1-\mu_k)\, g_{k+1}^T y_k}{(1-\lambda_k-\omega_k)\left\|g_k\right\|^2 + \lambda_k d_k^T y_k - \omega_k d_k^T g_k} \tag{4.31}$$

où $\mu_k,\ \lambda_k,\ \omega_k \in [0,1]$.

Cette famille à trois paramètres comprend les six méthodes standard du gradient conjugué, les familles à 1 ou 2 paramètres vues ci dessus et de nombreuses méthodes hybrides comme cas particuliers. Afin de veiller à ce que les directions de recherche générés par cette famille soient des directions de descente, le critère de redémarrage de Powell [58] est utilisé. Posons $d_k = -g_k$, si

$$\left|g_k^T g_{k-1}\right| > \xi \left\|g_k\right\|^2,$$

où $\xi > 0$ est une certaine constante fixe. Si on utilise une recherche linéaire inexacte de Wolfe forte (2.8)-(2.9) avec

$$(1+\xi)\,\sigma \leq \frac{1}{2}$$

Dai et Yuan [29] ont montré que les directions de recherche sont des directions de descente.

Les résultats de convergence globale sont également établis.

Dans [19] Dai et Liao modifient le numérateur du paramètre de mise à jour pour obtenir HS

$$\beta_k^{DL} = \frac{g_{k+1}^T (y_k - t s_k)}{d_k^T y_k} = \beta_k^{HS} - t \frac{g_{k+1}^T s_k}{d_k^T y_k} \ , \quad s_k = x_{k+1} - x_k = \alpha_k d_k. \ \ y_k = g_{k+1} - g_k \tag{4.32}$$

où $t > 0$ est une constante. Pour une recherche de ligne exacte, g_{k+1} est orthogonal à $s_k = x_{k+1} - x_k = \alpha_k d_k$. Ainsi, pour une recherche linéaire exacte, $\beta_k^{DL} = \beta_k^{HS}$. Par conséquent la méthode du gradient conjugué DL se réduit à la méthode HS et la méthode PRP. Encore une fois, en raison de l'exemple de Powell, la méthode DL peut ne pas converger pour une recherche linéaire exacte. Semblable à la méthode PRP + , Dai et Liao ont également modifié leur formule de la façon suivante pour assurer la convergence :

$$\beta_k^{DL+} = \max \left\{ \frac{g_{k+1}^T y_k}{d_k^T y_k}, 0 \right\} - t \frac{g_{k+1}^T s_k}{d_k^T y_k} \tag{4.33}$$

Si les hypothèses de Lipschitz $C1$ et de bornétude $C2$ sont vérifiées et si d_k satisfait à la condition de descente suffisante (2.3), il est montré dans [19] que la méthode DL+, mise en œuvre une recherche linéaire inexacte de Wolfe forte (2.8)-(2.9), est globalement convergente.

Très récemment, dans un nouveau développement de cette stratégie de mise à jour, Yabe et Takano [75] proposent le choix suivant pour le paramètre de mise à jour, basée sur une condition sécante modifiée donnée par Zhang et al. [78, 79] :

$$\beta_k^{YT} = \frac{g_{k+1}^T (z_k - t s_k)}{d_k^T z_k} \tag{4.34}$$

où

$$z_k = y_k + \left(\frac{\rho \theta_k}{s_k^T u_k} \right) u_k$$

$$\theta_k = 6 (f_k - f_{k+1}) + 3 (g_k + g_{k+1})^T s_k$$

$\rho \geq 0$ est une constante et $u_k \in \mathbb{R}^n$ satisfait $s_k^T u_k \neq 0$; par exemple, $u_k = .d_k$. Encore une fois, et suivant la démarche appliquée à la méthode PRP+, Yabe et Takano ont modifié leur formule de la façon suivante pour assurer la convergence :

$$\beta_k^{YT+} = \max \left\{ \frac{g_{k+1}^T z_k}{d_k^T z_k}, 0 \right\} - t \frac{g_{k+1}^T s_k}{d_k^T z_k} \tag{4.35}$$

Ils montrent que la méthode YT+ est globalement convergente si les hypothèses de Lipschitz $C1$ et de bornétude $C2$ sont vérifiées et d_k satisfait à la condition de descente suffisante, et si une recherche linéaire inexacte de Wolfe forte (2.8)-(2.9) est employée avec

$$0 \leq \rho < \frac{1-\sigma}{3\,(1+\sigma-2\delta)}$$

Sellami, Laskri et Benzine ([69]) ont proposé une nouvelle famille à deux paramètres des méthodes du gradient conjugué pour l'optimisation sans contraintes et ont étudié ses propriétés et sa convergence. β_k^* est de la forme générale suivante :

$$\beta_k^* = \frac{\phi_k}{\phi'_{k-1}}, \tag{0.10}$$

où ϕ'_{k-1} satisfait

$$\phi'_{k-1} = (1-\lambda_k-\mu_k)\left\|g_{k-1}\right\|^2 + (\lambda_k+\mu_k)(-g_{k-1}^T d_{k-1}). \tag{0.11}$$

et

$$\phi_k = \lambda\left\|g_k\right\|^2 + (1-\lambda)(-g_k^T d_k), \qquad \lambda \in [0,1].$$

Avec ces données, nous obtenons

$$\beta_k^* = \frac{\left\|g_k\right\|^2}{(1-\lambda_k-\mu_k)\left\|g_{k-1}\right\|^2 + \mu_k(-d_{k-1}^T g_{k-1}) + \lambda_k(y_{k-1}^T d_{k-1})}$$

Chapitre V

Modification de la recherche linéaire d'Armijo pour satisfaire les propriétés de convergence globale de la méthode du gradient conjugué de Hestenes-Stiefel

Considerons le problème d'optimisation sans contraintes suivant :

$$\min f(x), \quad x \in \mathbb{R}^n, \tag{5.1}$$

où f est continuement differentiable. Notons par $g(x) = \nabla f(x)$ le gradient de f. Pour résoudre le problème (5.1) on utilise des méthodes itératives qui générent des suites $\{x_k\}$ définies de la façon suivante :

$$x_{k+1} = x_k + \alpha_k d_k \tag{5.2}$$

où $x_k \in \mathbb{R}^n$ est la *kième* approximation de la solution, α_k est le pas obtenu en effectuant une recherche linéaire exacte ou inexacte et d_k est la direction de recherche.

Les méthodes du gradient conjugué sont parmi les méthodes les plus efficaces pour résoudre le problème (5.1), spécialement lorsqu'il s'agit de problèmes de grande taille. La direction de recherche d_k est définie par

$$d_k = \begin{cases} -g_k & \text{si } k = 0 \\ -g_k + \beta_k d_{k-1} & \text{si } k \geq 1 \end{cases} \tag{5.3}$$

où β_k est un scalaire et $g_k = g(x_k)$. La version originale du gradient conjugué proposé par

Hestenes et Stiefel (HS conjugate gradient method) [48], dans laquelle β_k est défini par

$$\beta_k^{HS} = \frac{g_k^T(g_k - g_{k-1})}{d_{k-1}^T(g_k - g_{k-1})} \tag{5.4}$$

Les autres formes de β_k, sont définies comme suit :

$$\beta_k^{FR} = \frac{g_k^T g_k}{g_{k-1}^T g_{k-1}} \qquad Fletcher - Reeves[35], \tag{5.5}$$

$$\beta_k^{PRP} = \frac{g_k^T(g_k - g_{k-1})}{g_{k-1}^T g_{k-1}} \qquad Polak - Ribière - Polyak[55,56], \tag{5.6}$$

$$\beta_k^{CD} = -\frac{g_k^T g_k}{d_{k-1}^T g_{k-1}} \qquad Conjugate - Descent[34], \tag{5.7}$$

$$\beta_k^{LS} = -\frac{g_k^T(g_k - g_{k-1})}{d_{k-1}^T g_{k-1}} \qquad Liu - Storey[51], \tag{5.8}$$

$$\beta_k^{DY} = \frac{g_k^T g_k}{d_{k-1}^T(g_k - g_{k-1})} \qquad Dai - Yuan[24], \tag{5.9}$$

La convergence globale des différentes versions du gradient conjugué joue un rôle important dans la théorie du gradient conjugué. Zoutendijk [80] a prouvé que la méthode du gradient conjugué, version **FR** avec une recherche linéaire exacte est globalement convergente. Al-Baali [1] généralisa ce résultat en utilisant la recherche linéaire inéxacte de Wolfe forte. Powell [60] démontra que la suite des normes du gradient $\|g_k\|$ vérifie la condition $\|g_k\| \geq C$, C constante strictement positive, seulement lorsqu'on a

$$\sum_{k\geq 0} \frac{1}{\|d_k\|} < +\infty \tag{5.10}$$

Par conséquent on peut démontrer la convergence globale de la méthode FR en utilisant la relation (5.10). Contrairement aux bons résultats de convergence de la méthode FR, on n'a malheureusement pas réussi à prouver un résultat de convergence pour la méthode PRP en utilisant une recherche linéaire inéxacte de Wolfe forte.

Quelques résultats de convergence ont été prouvés en utilisant des recherches linéaires inexactes assez compliquées ou en restreignant l'ensemble des paramètres β_k à prendre que des valeurs positives [41-66-67-68]. En utilisant la méthode CD on a réussi à prouver la conver-

gence globale en utilisant une recherche linéaire inéxacte de Wolfe forte et une très grande restriction sur les paramètres β_k [21]. Dans la méthode DY on a obtenu un résultat de convergence globale en utilisant seulement des recherches linéaires inexactes de Wolfe faibles [24]. Une impressionnante litterature sur les méthodes du gradient conjugué se trouvent dans [13-18-22-33-49-52-63-65-76].

D'après nos connaissances, il n'existe pas de résultats de convergence globale pour les méthodes LS et HS avec des recherches linéaires de Wolfe. Dans ce travail on introduit une nouvelle recherche linéaire inexacte et avec laquelle nous prouverons la convergence globale de la méthode HS.

Partant de la condition de descente suffisante suivante

$$g_k^T d_k < -c \left\| g_k \right\|^2 , \ c = cte \in]0,1[\tag{5.11}$$

d_k étant détérminée à l'itération k, nous essayons de calculer le pas α_k. Ceci nous permettra à finir l'itération k et passer à litération suivante.

Il existe plusieurs approches pour trouver le pas α_k. La recherche linéaire exacte est l'approche idéale mais son cout est élevé en mémoire et temps ou même impossible à calculer. C'est pour cette raison que les recherches linéaires inexactes ou économiques ont été élaborées. Les plus célèbres sont les recherches linéaires inexactes d'Armijo [1] ou de Goldstein ou de Wolfe [44,71]. La recherche linéaire inexacte d'Armijo est facile à implementer en pratique.

La recherche linéaire inexacte d'Armijo

On se donne une constante $s > 0$, $\rho \in (0,1)$ et $\mu \in (0,1)$. Choisir α_k de sorte qu'elle soit le plus grand α dans $\{s, s\rho, s\rho^2, ..., \}$ et qu'elle vérifie l'inéglité suivante dite d'Armijo

$$f(x_k + \alpha d_k) \leq f(x_k) + \alpha g_k^T d_k. \tag{Armijo}$$

L'inconvénient de la *recherche linéaire inexacte d'Armijo* est comment choisir le point initial s. Si s est trés grand alors l'algorithme a besoin de beaucoup d'évaluations de type (Armijo). Si s est trop petit, alors l'efficacité de l'algorithme sera affectée et diminuée. Ainsi on choisira un point initial s adéquat à chaque itération de sorte qu'on trouvera un pas α_k facilement.

Le travail original de cette thèse consiste à construire une nouvelle recherche linéaire inexacte du type Armijo dans laquelle le pas initial s est choisi de façon appropriée mais varie aussi à chaque itération. La nouvelle recherche linéaire qu'on appelle recherche linéaire d'Armijo modifiée nous permet de trouver le pas α_k facilement et nous garantit par la suite la convergence globale de l'algorithme du gradient conjugué version Hestenes et Steifel HS sous des conditions

assez faibles. On démontrera aussi que le nouveau algorithme admet une vitesse de convergence linéaire. Des tests numériques prouveront que notre nouvelle méthode c'est à dire HS avec recherche linéaire d'Armijo modifiée est plus performande que la méthode classique HS avec recherche linéaire inexacte d'Armijo.

1 Nouvelle recherche linéaire inéxacte d'Armijo modifiée

Avant tout et tout le long de ce chapitre nous supposerons que la fonction objectif f vérifie les deux hypothèses suivantes :

Hypothèse A. La fonction objectif $f \in C^1(\mathbb{R}^n)$ et f est minorée sur $\mathbb{R}^n$

Hypothèse B. Le gradient $g(x) = \nabla f(x)$ de $f(x)$ est Lipschitzien sur un ensemble convexe Γ qui contient l'ensemble de niveau $L(x_0) = \{x \in \mathbb{R}^n \mid f(x) \leq f(x_0)\}$ où x_0 est un point initial donné, i.e., il existe $L > 0$ telle que

$$\|g(x) - g(y)\| \leq L \|x - y\|, \forall x, y \in \Gamma.$$

Nouvelle recherche linéaire inéxacte d'Armijo modifiée

On se donne $\mu \in \left]0, \frac{1}{2}\right[$, $\rho \in]0,1[$ et $c \in \left[\frac{1}{2}, 1\right]$, Poser $s_k = \frac{1-c}{L_k} \frac{d_K^T (g_{k+1} - g_k)}{\|d_k\|^2}$. Calculer α_k comme étant le plus grand α dans l'ensemble $\{s, s\rho, s\rho^2, ...,\}$ et vérifiant les deux inégalités suivantes :

$$f(x_k) - f(x_k + \alpha d_k) \geq -\alpha \mu g_k^T d_k. \qquad \text{(Armijo-modifie9)}$$

$$g(x_k + \alpha d_k)^T d(x_k + \alpha d_k) \leq -c \|g(x_k + \alpha d_k)\|^2$$

et

$$\|g_{k+1}\| \leq c \|d_k\|$$

avec

$$d(x_k + \alpha d_k) = -g(x_k + \alpha d_k) + \frac{g(x_k + \alpha d_k)^T [g(x_k + \alpha d_k) - g(x_k)]}{d_k^T [g(x_k + \alpha d_k) - g(x_k)]} d_k$$

pour

$$d_k^T g_{k+1} > d_k^T g_k$$

et L_k est une approximation de la constante de Lipshitz L de $g(x)$.

2 Le nouveau Algorithme. Définition et propriétés générales

Nous sommes maintenant en mesure d'introduire notre nouveau Algorithme.

Algorithm 1 ***Etape 0 :*** *On se donne* $x_0 \in \mathbb{R}^n$, *Poser* $d_0 = g_0$, $k := 0$.

Etape 1 : *Si* $\|g_k\| = 0$ *alors stop. sinon aller à Etape 2.*

Etape 2 : *Poser* $x_{k+1} = x_k + \alpha_k d_k$ *où* d_k *est définie par (3),* $\beta_k = \beta_k^{HS}$,

α_k *est définie par la nouvelle recherche linéaire inexacte d'Aarmijo modifiée*

Etape 3 : *Poser* $k := k + 1$ *et aller à Etape1.*

Quelques propriétés du nouveau algorithme se trouvent dans les lemmes suivants :

Lemme 2.1 *Supposons que les hypothèses (A) et (B) soient satisfaites et que le nouveau algorithme HS muni de la nouvelle recherche linéaire d'Armijo modifiée génère une suite infinie* $\{x_k\}$. *Alors on a*

$$\alpha_k \leq \frac{1-c}{L} \frac{d_k^T (g_{k+1} - g_k)}{\|d_k\|^2} \tag{5.12}$$

et

$$g_{k+1}^T d_{k+1} \leq -c \|g_{k+1}\|^2$$

Preuve. La condition **(B)**, l'inégalité de Cauchy–Schwartz et la méthode **HS** impliquent

$$\begin{aligned}
(1-c)d_k^T(g_{k+1}-g_k) &\geq \alpha_k L \|d_k\|^2 \\
&= \frac{\alpha_k L \|g_{k+1}\| \|d_k\|}{\|g_{k+1}\|^2} \|g_{k+1}\| \|d_k\| \\
&\geq \frac{\|g_{k+1}\| \|g_{k+1}-g_k\|}{\|g_{k+1}\|^2} \|g_{k+1}\| \|d_k\| \\
&\geq \frac{\left|g_{k+1}^T(g_{k+1}-g_k)\right|}{\|g_{k+1}\|^2} \left|g_{k+1}^T d_k\right| \\
&\geq \frac{g_{k+1}^T(g_{k+1}-g_k)}{d_k^T(g_{k+1}-g_k)} \frac{d_k^T(g_{k+1}-g_k)}{\|g_{k+1}\|^2} \left(g_{k+1}^T d_k\right) \\
&= \beta_{k+1}^{HS} \frac{d_k^T(g_{k+1}-g_k)}{\|g_{k+1}\|^2} \left(g_{k+1}^T d_k\right)
\end{aligned}$$

Donc

$$(1-c)\, d_k^T(g_{k+1}-g_k) \geq \beta_{k+1}^{HS} \frac{d_k^T(g_{k+1}-g_k)}{\|g_{k+1}\|^2} \left(g_{k+1}^T d_k\right)$$

et ainsi

$$-c\|g_{k+1}\|^2 \geq -\|g_{k+1}\|^2 + \beta_{k+1}^{HS} \left(g_{k+1}^T d_k\right) = g_{k+1}^T d_{k+1}$$

Le lemme est démontré. ■

Lemme 2.2 *Supposons que les hypothèses (A) et (B) soient satisfaites. Alors le nouveau algorithme **HS** muni de la nouvelle recherche linéaire d'Armijo modifiée est bien défini.*

Preuve. Puisque

$$\lim_{\alpha\to 0} \frac{f_k - f(x_k+\alpha d_k)}{\alpha} = -g_k^T d_k > -\mu g_k^T d_k$$

alors il existe $\alpha_k^{'} > 0$ telle que

$$\frac{f_k - f(x_k+\alpha d_k)}{\alpha} \geq -\mu g_k^T d_k, \forall \alpha \in \left[0, \alpha_k^{'}\right].$$

Ainsi et en posant $\alpha_k^{"} = min(s_k, \alpha_k^{'})$, nous obtenons

$$\frac{f_k - f(x_k+\alpha d_k)}{\alpha} \geq -\mu g_k^T d_k, \forall \alpha \in \left[0, \alpha_k^{"}\right]. \tag{5.13}$$

D'autre part et en tenant compte du Lemme 5.1, on a :

$$g\left(x_k+\alpha d_k\right)^T d(x_k+\alpha d_k) \leq -c\left\|g\left(x_k+\alpha d_k\right)\right\|^2.$$

si $\alpha \leq \frac{1-c}{L}\frac{d_k^T(g_{k+1}-g_k)}{\|d_k\|^2}$. Posons

$$\overline{\alpha_k} = \min\left(\alpha_k^{"}, \frac{1-c}{L}\frac{d_k^T(g_{k+1}-g_k)}{\|d_k\|^2}\right)$$

Si on prend $\alpha \in [0, \overline{\alpha_k}]$, alors le nouveau algorithme **HS** muni de la nouvelle recherche linéaire d'Armijo modifiée est bien défini. Le lemme est prouvé. ■

3 Convergence globale du nouveau Algorithme

Lemme 3.1 *Supposons que les hypothèses (A) et (B) soient satisfaites et que le nouveau algorithme **HS** muni de la nouvelle recherche linéaire d'Armijo modifiée génère une suite infinie $\{x_k\}$. Supposons aussi qu'il existe $m_0 > 0$ et $M_0 > 0$ telles que $m_0 \leq L_k \leq M_0$. Alors on a,*

$$\|d_k\| \leq \left(1 + \frac{L(1-c)}{m_0}\right)\|g_k\|, \forall k. \tag{5.14}$$

Preuve. Pour $k = 0$, on a

$$\|d_k\| = \|g_k\| \leq \|g_k\|\left(1 + \frac{L(1-c)}{m_0}\right)$$

Si $k > 0$, alors en utilisant la nouvelle recherche linéaire d'Armijo modifiée, nous avons

$$\alpha_k \leq s_k = \frac{1-c}{L_k}\frac{d_k^T(g_{k+1}-g_k)}{\|d_k\|^2} \leq \frac{1-c}{m_0}\frac{d_k^T(g_{k+1}-g_k)}{\|d_k\|^2}$$

L'inégalité de Cauchy Shwartz, L'inégalité précédente et la formule **HS** donnent :

$$\begin{aligned}
\|d_{k+1}\| &= \left\|-g_{k+1} + \beta_{k+1}^{HS} d_k\right\| \\
&\leq \|g_{k+1}\| + \frac{\left|g_{k+1}^T(g_{k+1}-g_k)\right|}{\left|d_k^T(g_{k+1}-g_k)\right|}\|d_k\| \\
&\leq \|g_{k+1}\|\left[1 + L\alpha_k\frac{\|d_k\|^2}{d_k^T(g_{k+1}-g_k)}\right] \\
&\leq \left(1 + L\frac{(1-c)}{m_0}\right)\|g_{k+1}\|
\end{aligned}$$

La preuve est terminée. ■

Théorème 3.1 *Supposons que les hypothèses (A) et (B) soient satisfaites et que le nouveau algorithme **HS** muni de la nouvelle recherche linéaire d'Armijo modifiée génère une suite infinie* $\{x_k\}$. *Supposons aussi qu'il existe* $m_0 > 0$ *et* $M_0 > 0$ *telles que* $m_0 \leq L_k \leq M_0$. *Alors on a,*

$$\lim_{k\to\infty} \|g_k\| = 0. \tag{5.15}$$

Preuve. Notons $\eta_0 = inf\{\alpha_k : k \in \mathbb{N}\}$. Si $\eta_0 > 0$ alors on a

$$f_k - f_{k+1} \geq -\mu\alpha_k g_k^T d_k \geq \mu\eta_0 c \|g_k\|^2 .$$

D'après **(A)** on a

$$\overset{+\infty}{\underset{k=0}{}} \|g_k\|^2 < +\infty$$

et par conséquent

$$\lim_{k\to\infty} \|g_k\| = 0.$$

Maintenant montrons que $\eta_0 > 0$. Supposons que $\eta_0 = 0$. Alors il existe un sous ensemble infini $K \subseteq \{0, 1, 2, ..., \}$ tel que

$$\lim_{k\in K, k\to\infty} \alpha_k = 0 \tag{5.16}$$

D'après le lemme 5.3 on a

$$s_k = \frac{1-c}{L_k}\frac{d_k^T(g_{k+1}-g_k)}{\|d_k\|^2} \geq \frac{1-c}{m_0}\left(1+\frac{L(1-c)}{m_0}\right)^{-2} > 0$$

Donc il existe $k^{'}$ tel que

$\alpha_k/\rho \leq s_k$, $k \geq k^{'}$et $k \in K$.

Soit $\alpha = \alpha_k/\rho$, au moins l'une des deux inégalités :

$$f_k - f(x_k + \alpha d_k) \geq -\alpha\mu g_k^T d_k \tag{5.17}$$

ou

$$g\left(x_k + \alpha d_k\right)^T d(x_k + \alpha d_k) \leq -c \|g\left(x_k + \alpha d_k\right)\|^2 . \tag{5.18}$$

n'a pas lieu. Si (5.17) n'a pas lieu, alors nous avons :

$$f_k - f(x_k + \alpha d_k) < -\alpha\mu g_k^T d_k.$$

Maintenant utilisons le théorème de la valeur moyenne sur la partie gauche de l'inégalité précédente. Alors il existe $\theta_k \in [0,1]$ telle que

$$-\alpha g\left(x_k+\alpha d_k\right)^T d_k < -\alpha\mu g_k^T d_k$$

Par conséquent

$$g\left(x_k+\alpha d_k\right)^T d_k > \mu g_k^T d_k \tag{5.19}$$

La condition **(A)**, l'inégalité de Cauchy–Schwartz, (5.19) et le lemme 5.1 impliquent :

$$\begin{aligned} L\alpha\left\|d_k\right\|^2 &\geq \left\|g\left(x_k+\alpha\theta_k d_k\right)-g\left(x_k\right)\right\|\left\|d_k\right\| \\ &\geq \left|\left(g\left(x_k+\alpha\theta_k d_k\right)-g\left(x_k\right)\right)^T\ d_k\right| \\ &\geq -(1-\mu)g_k^T d_k \\ &\geq c(1-\mu)\left\|g_k\right\|^2. \end{aligned}$$

D'après le lemme 5.3 on a :

$$\alpha_k \geq \frac{c\rho\left(1-\mu\right)}{L}\frac{\left\|g_k\right\|^2}{\left\|d_k\right\|^2} \geq \frac{c\rho\left(1-\mu\right)}{L}\frac{1}{\left(\frac{1}{1+\frac{L(1-\mu)}{m_0}}\right)^2} > 0, k \geq k^{'}, k \in K.$$

Ceci contredit (5.16).

Si (5.18) n'a pas lieu, alors nous avons :

$$g\left(x_k+\alpha d_k\right)^T d(x_k+\alpha d_k) > -c\left\|g\left(x_k+\alpha d_k\right)\right\|^2.$$

et ainsi,

$$\frac{g\left(x_k+\alpha d_k\right)^T\left[g\left(x_k+\alpha d_k\right)-g_k\right]}{d_k^T\left[g\left(x_k+\alpha d_k\right)-g_k\right]} g\left(x_k+\alpha d_k\right)^T d_k > (1-c)\left\|g\left(x_k+\alpha d_k\right)\right\|^2.$$

Maintenant utilisons l'inégalité de Cauchy–Schwartz sur la partie gauche de l'inégalité précédente, on a :

$$\alpha L\frac{\left\|d_k\right\|^2}{d_k^T\left(g\left(x_k+\alpha d_k\right)-g_k\right)} > (1-c)$$

Avec le lemme 5.3, ceci donne

$$\alpha_k > \frac{\rho\,(1-c)}{L}\frac{d_k^T\left(g\left(x_k+\alpha d_k\right)-g_k\right)}{\left(1+\frac{L(1-\mu)}{m_0}\right)^2\|g_k\|^2} > 0, k \geq k^{'}, k \in K.$$

qui contredit aussi (5.16). Ceci montre bien que $\eta_0 > 0$. La preuve complète du théorème es terminée. ■

4 Etude de la vitesse de convergence

Dans cette partie on va démontrer que le nouveau algorithme **HS** muni de la nouvelle recherche linéaire d'Armijo modifiée a une vitesse de convergence linéaire sous des hypothèses assez faibles.

Nous supposons qu'en plus des hypothèses (A) et (B), la fonction objectif f vérifie l'Hypothèse (C) suivante :

Hypothèse C. On suppose que la suite $\{x_k\}$ générée par le nouveau algorithme **HS** muni de la nouvelle recherche linéaire d'Armijo modifiée converge vers un point x^*. De plus supposons que la matrice Hessienne $\nabla^2 f(x^*)$ est définie positive et que $f(x)$ est deux fois continuement differentiable sur l'ensemble : $N(x^*,\varepsilon_0) = \{x/\,\|x-x^*\| < \varepsilon_0\}$.

Lemme 4.1 *Supposons que l'Hypothèse (C) soit vérifiée. Alors il existe $m^{'}, M^{'}$ et ε_0 avec $0 < m^{'} \leq M^{'}$ et $\varepsilon \leq \varepsilon_0$ tels que :*

$$m^{'}\|y\|^2 \leq y^T\nabla^2 f(x)y \leq M^{'}\|y\|^2,\ \forall x,y \in N(x^*,\varepsilon)5.20 \quad \text{(V.1)}$$

$$\frac{1}{2}m^{'}\|x-x^*\|^2 \leq f(x)-f(x^*) \leq \frac{1}{2}M^{'}\|x-x^*\|^2, \forall x \in N(x^*,\varepsilon)5.21 \quad \text{(V.2)}$$

$$M^{'}\|x-y\|^2 \geq (g(x)-g(x^*))^T(x-y) \geq m^{'}\|x-y\|^2, \forall x,y \in N(x^*,\varepsilon). \quad (5.22)$$

Ainsi on a

$$M^{'}\|x-x^*\|^2 \geq g(x)^T(x-x^*) \geq m^{'}\|x-x^*\|^2, \forall x \in N(x^*,\varepsilon) \quad (5.23)$$

De (5.23) et (5.22) on peut avoir en utilisant l'inégalité de Cauchy–Schwartz

$$M^{'}\|x-x^*\| \geq \|g(x)\| \geq m^{'}\|x-x^*\|, \forall x \in N(x^*,\varepsilon) \quad (5.24)$$

et

$$\|g(x) - g(y)\| \leq m^{'} \|x - y\|, \forall x, y \in N(x^*, \varepsilon) \tag{5.25}$$

Preuve. La preuve de ce théorème est trés connue en littérature, voir (e.g. [77]). ■

Théorème 4.1 *Supposons que les hypothèses (C) soit satisfaite et que le nouveau algorithme* ***HS*** *muni de la nouvelle recherche linéaire d'Armijo modifiée génère une suite infinie* $\{x_k\}$. *Supposons aussi qu'il existe* $m_0' > 0$ *et* $M_0' > 0$ *telles que* $m_0' \leq L_k \leq M_0'$. *Alors* $\{x_k\}$ *converge vers* x^* *au moins* ***R****-linéairement.*

Preuve. Supposons que l'Hypothèse (C) soit vérifiée. Alors il existe $k^{'}$ telle que $x_k \in N(x^*, \varepsilon_0)$, $\forall k \geq k^{'}$. Sans nuire à la généralité, nous pouvons supposer que $x_0 \in N(x^*, \varepsilon_0)$. Alors les hypothèses (A) et (B) sont vérifiées. D'après la preuve du théorème 5.1, on a :

$$f_k - f_{k+1} \geq \eta_0 c\mu \|g_k\|^2. \tag{5.26}$$

Le lemme 1 et l'inégalité de Cauchy–Schwartz inequality donnent

$$\|g_k\| \|d_k\| \geq -g_k^T d_k \geq c \|g_k\|^2.$$

Ce qui implique

$$\|d_k\| \geq c \|g_k\| \tag{5.27}$$

le nouveau algorithme **HS** muni de la nouvelle recherche linéaire d'Armijo modifiée et le lemme 5.4 impliquent

$$m^{'} \leq L_k \tag{5.28}$$

La définition de η_0 dans la preuve du Theorem 1, (5.27) et (5.28), donnent :

$$\begin{aligned}
\eta_0 &\leq \frac{1-c}{L_k} \frac{d_k^T (g_{k+1} - g_k)}{\|d_k\|^2} \\
&\leq \frac{1-c}{L_k} \frac{\|d_k\| \|g_{k+1} - g_k\|}{\|d_k\|^2} \\
&\leq \frac{1-c}{m^{'}} \left(\frac{\|g_{k+1}\|}{\|d_k\|} + \frac{\|g_k\|}{\|d_k\|} \right) \\
&\leq \frac{1-c}{m^{'}} \left(\frac{2}{c} \right)
\end{aligned}$$

Par conséquent

$$\eta_0 \leq \frac{2(1-c)}{cm^{'}} \tag{5.29}$$

Posons $\eta = \eta_0 c\mu$. Le lemme 5.4, la relation (5.26) donnent :

$$\begin{aligned} f_k - f_{k+1} &\geq \eta \left\| g_k \right\|^2 . \\ &\geq \eta m^{'2} \left\| x_k - x^* \right\|^2 \\ &\geq \frac{2\eta m^{'2}}{M'} \left(f_k - f^* \right) \end{aligned}$$

Si on note

$$\theta = m^{'} \sqrt{\frac{2\eta}{M'}}$$

alors on a

$$f_k - f_{k+1} \geq \theta^2 \left(f_k - f^* \right) \tag{5.30}$$

et nous pouvons prouver $\theta < 1$. Maintenant, d'après la définition de η, la relation (5.29) et n'oublions pas de noter que $m^{'} < M^{'} \leq L$, alors nous avons :

$$\begin{aligned} \theta^2 &= \frac{2\eta m^{'2}}{M'} \\ &\leq \frac{2m^{'2}\eta_0 c\mu}{M'} \\ &\leq \frac{2m^{'2} c\mu}{M'} \frac{2\left(1-c\right)}{cm^{'}} \\ &< \left(2\mu\right)\left(2\left(1-c\right)\right) < 1 \end{aligned}$$

Si on note

$$\omega = \sqrt{1-\theta^2}$$

et du fait que $\omega < 1$, on obtient en utilisant (5.30) que :

$$\begin{aligned} f_{k+1} - f^* &\leq \left(1-\theta^2\right)\left(f_k - f^*\right) \\ &= \omega^2 \left(f_k - f^*\right) \\ &\leq \ ... \\ &\leq \omega^{2\left(k-k^{'}\right)} \left(f_{k^{'}+1} - f^*\right) \end{aligned}$$

De la première égalité, on obtient :

$$\frac{f_{k+1} - f^*}{f_k - f^*} \leq 1 - \theta^2 < 1$$

qui montre que la suite $\{f_k\}$ converge vers f^* Q-linéairement.
De plus et en utilisant le lemme 5.4, et les précédentes inégalités on a :

$$\begin{aligned} \|x_{k+1} - x^*\|^2 &\leq \frac{2}{m'}\left(f_{k+1} - f^*\right) \\ &\leq \omega^{2\left(k-k'\right)} \frac{2\left(f_{k'+1} - f^*\right)}{m'} \end{aligned}$$

et par conséquent on obtient :

$$\|x_k - x^*\| \leq \omega^k \sqrt{\frac{2\left(f_{k'+1} - f^*\right)}{m'\omega^{2\left(k'+1\right)}}}$$

et

$$\lim_{k\to\infty} \|x_k - x^*\|^{\frac{1}{k}} \leq \omega < 1,$$

qui montre clairement que la suite $\{x_k\}$ converge vers x^* de façon R-linéaire.
D'autre part l'inégalité $f_{k+1} - f^* \leq (1-\theta^2)\left(f_k - f^*\right)$ et (5.23) impliquent que :

$$\frac{\|x_{k+1} - x^*\|}{\|x_k - x^*\|} \leq \sqrt{\frac{M'}{m'}(1-\theta^2)},$$

qui montre que $\{x_k\}$ converge vers x^* Q-linéairement, à condition que :

$$\sqrt{\frac{M'}{m'}(1-\theta^2)} < 1.$$

Si $\sqrt{\frac{M'}{m'}(1-\theta^2)} \geq 1$, on ne peut pas obtenir que la suite $\{x_k\}$ converge vers x^* Q-linéairement. On obtient seulement la convergence de $\{x_k\}$ vers x^* au moins R-linéairement. ■

5 Tests numériques

Dans cette partie nous allons effectuer des tests numériques pour montrer l'efficacité et la performance du nouveau algorithme HS muni de la nouvelle recherche linéaire d'Armijo modifiée.

La constante de Lipschitz L associée à $g(x)$ n'est pas connue dans les calculs pratiques et a besoin d'etre éstimée. Dans ce qui suit, nous allons aborder ce problème et donner des approches qui permettent d'estimer L. Dans [66], quelques approches pour estimer L ont été

proposées. Si $k \geq 1$ alors nous posons $\delta_{k-1} = x_k - x_{k-1}$ et $y_{k-1} = g_k - g_{k-1}$ et nous obtenons les estimations suivantes :

$$L \simeq \frac{\|y_{k-1}\|}{\|\delta_{k-1}\|}, \tag{5.31}$$

$$L \simeq \frac{\|y_{k-1}\|^2}{\delta_{k-1}^T y_{k-1}}, \tag{5.32}$$

$$L \simeq \frac{\delta_{k-1}^T y_{k-1}}{\|\delta_{k-1}\|^2}. \tag{5.33}$$

De ce fait , si L est la constante de Lipschitz, alors toute autre constante $L^{'}$ plus grande que L est aussi une constante de Lipschitz. Ceci nous permet de trouver de grandes constantes de Lipschitz. Cependant de trés grandes constantes de Lipschitz aboutissent à de très petits pas α_k et rendraient la convergence du nouveau algorithme HS muni de la nouvelle recherche linéaire d'Armijo modifiée trés lente. Ainsi nous chercherons des constantes de Lipshitz aussi petites que possible dan les calculs pratiques.

Dans la $k$ième iteration nous prenons la constante de Lipschitz comme suit

$$L_k = \max\left(L_0, \frac{\|y_{k-1}\|}{\|\delta_{k-1}\|}\right), \tag{5.34}$$

$$L_k = \max\left(L_0, \min\left(\frac{\|y_{k-1}\|^2}{\delta_{k-1}^T y_{k-1}}, M_0^{'}\right)\right), \tag{5.35}$$

$$L_k = \max\left(L_0, \frac{\delta_{k-1}^T y_{k-1}}{\|\delta_{k-1}\|^2}\right), \tag{5.36}$$

avec $L_0 > 0$ et $M_0^{'}$ étant un grand nombre.

Lemme 5.1 *Supposons que les hypothèses (A) et (B) soient satisfaites et que le nouveau algorithme HS muni de la nouvelle recherche linéaire d'Armijo modifiée génère une suite infinie* $\{x_k\}$ *et* L_k *est évaluée par (5.34), (5.35) or (5.36). Alors il existe* $m_0 > 0$ *et* $M_0 > 0$ *telles que :*

$$m_0 = L_k = M_0 \tag{5.37}$$

Preuve. Evidemment, $L_k = L_0$, et on peut prendre $m_0 = L_0$. Pour (5.34) nous avons

$$L_k = \max\left(L_0, \frac{\|y_{k-1}\|}{\|\delta_{k-1}\|}\right) \leq \max\left(L_0, L\right).$$

Pour (5.35), on a

$$L_k = \max\left(L_0, \min\left(\frac{\|y_{k-1}\|^2}{\delta_{k-1}^T y_{k-1}}, M_0^{'}\right)\right) \leq \max\left(L_0, M_0^{'}\right).$$

Pour (5.36), utilisant l'inégalité de Cauchy–Schwartz, on a :

$$L_k = \max\left(L_0, \frac{\delta_{k-1}^T y_{k-1}}{\|\delta_{k-1}\|^2}\right) \leq \max\left(L_0, L\right).$$

Si on pose $M_0^{'} = \max(L_0, L, M_0^{'})$, ceci complète la preuve. ■

Notons par HS1, HS2 et HS3 la méthode HS munie de la nouvelle recherche linéaire d'Armijo modifiée et les estimations (5.34)–(5.36), respectivement. HS est la méthode basique de Hestenes et Steifel munie de la recherche linéaire inexacte de Wolfe forte. On note par PRP+ la méthode PRP avec la condition

$$\beta_k = \max\left(0, \beta_k^{PRP}\right)$$

et la recherche linéaire inexacte de Wolfe forte.

5.1 les profils de performance numérique

les profils de performance numérique de la méthode du gradient conjugué HS1, HS2, HS3, PRP+ et HS basés sur le temps CPU.

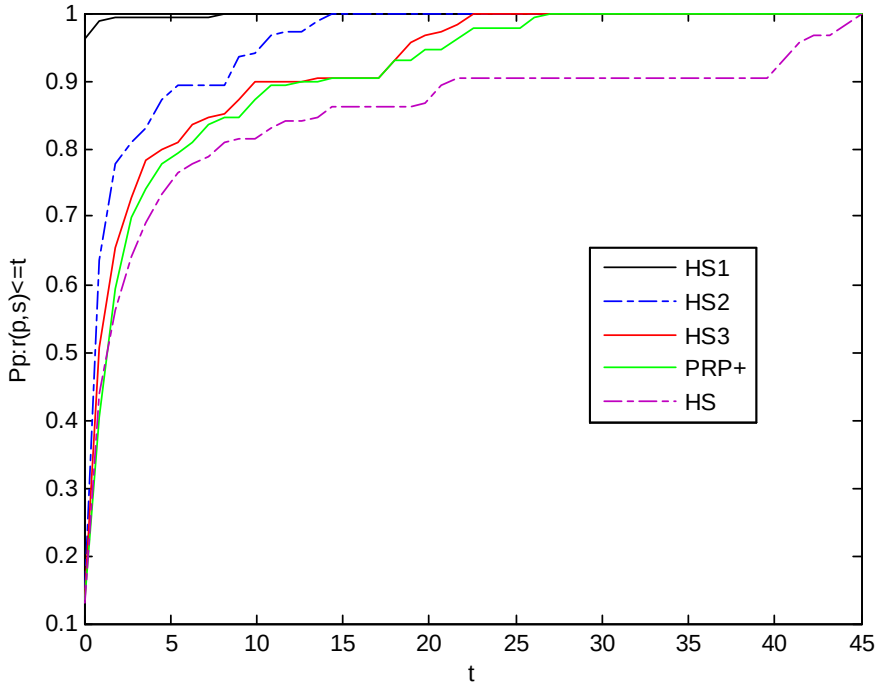

FIG. 5.1 – Performance basée sur le temps CPU.

les profils de performance numérique de la méthode du gradient conjugué HS1, HS2, HS3, PRP+ et HS basés sur le nombre d'itérations.

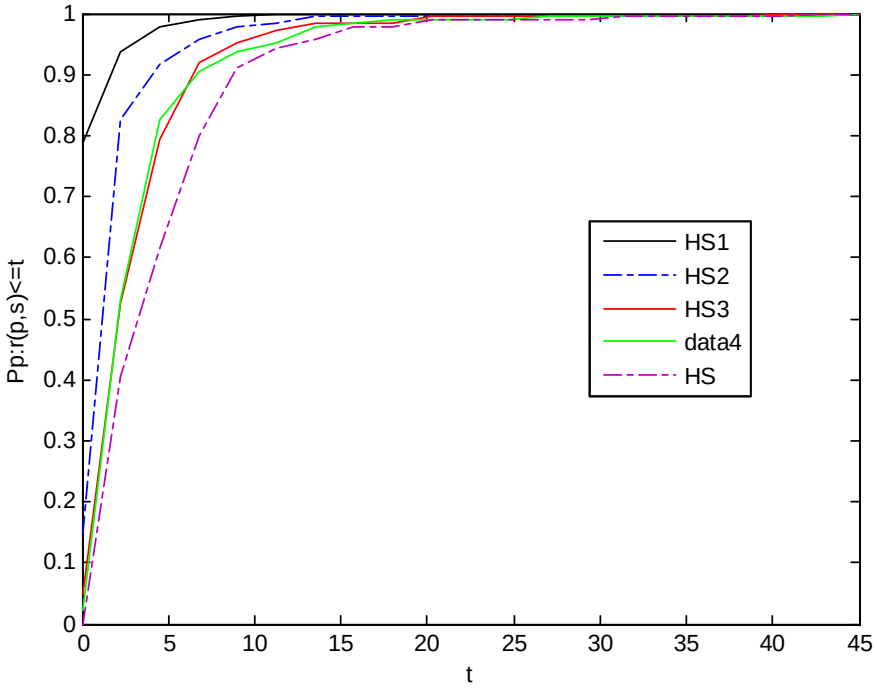

FIG. 5.2 – Performance basée sur le nombre d'itérations.

les profils de performance numérique de la méthode du gradient conjugué HS1, HS2, HS3, PRP+ et HS basés sur le nombre d'évaluations de la fonction.

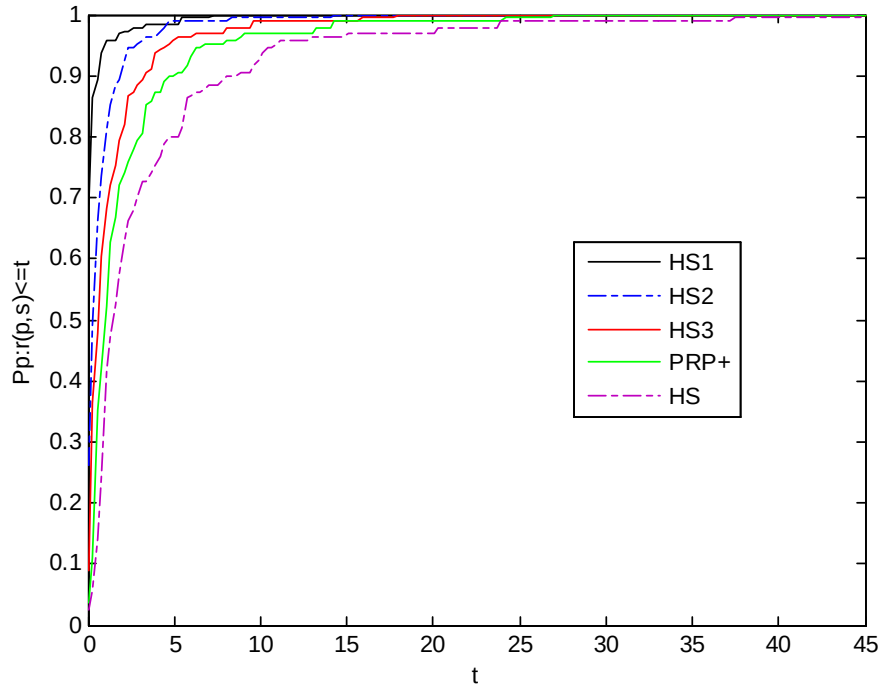

FIG. 5.3 – Performance basée sur le nombre d'évaluations de fonction.

les profils de performance numérique de la méthode du gradient conjugué HS1, HS2, HS3, PRP+ et HS basés sur le nombre d'évaluations de gradient.

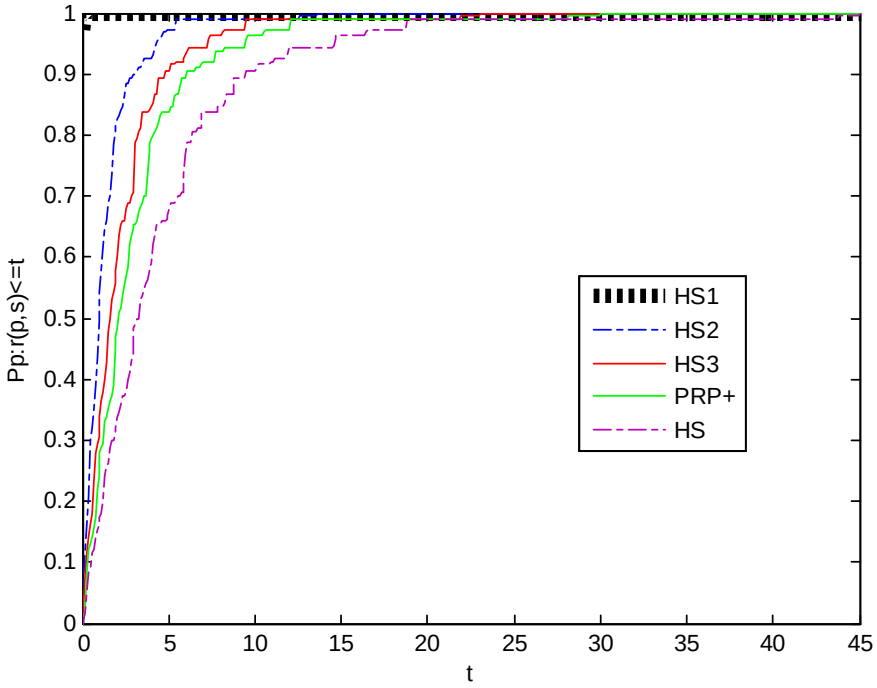

FIG. 5.4 – Performance basée sur le nombre d'évaluations de gradient.

Conclusion

Il est clair d'après les figures. 5.1, 5.2, 5.3 et 5.4, que notre méthode est numériquement plus performante que les autres méthodes du gradient conjugué PRP, PRP+ et HS.

Conclusion générale

Soit $f : \mathbb{R}^n \to \mathbb{R}$ non quadratique et (P) le problème de minimisation sans contraintes suivant :

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

Pour résoudre les problèmes d'optimisation sans contraintes, les méthodes du gradient conjugué génèrent des suites $\{x_k\}_{k=0,1,2...}$ de la forme suivante :

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

Le pas $\alpha_k \in \mathbb{R}$ étant déterminé par une recherche linéaire. La direction d_k est définie par la formule de récurrence suivante($\beta_k \in \mathbb{R}$)

$$d_k = \begin{cases} -g_k & \text{si } k = 1 \\ -g_k + \beta_{k-1} d_{k-1} & \text{si } k \geq 2 \end{cases}$$

Durant les 60 dernières années, beaucoup de méthodes du grdient conjugué ont été élaborées et par conséquent une multitude de résultats de convergence ont été démontrées. On peut se perdre facilement si on cherche à étudier tous ces résultats. On a vu dans cette thèse qu'on peut regrouper les différentes méthodes du gradient conjugué en deux grandes familles qui possèdent les mêmes propriétés de convergence :

a) Les méthodes dans lesquelles le terme $\|g_{k+1}\|^2$ figure dans le numérateur de β_k

Dans cette famille on trouve surtout :

a1) Gradient conjugué variante Fletcher Reeves(FR)

$$\beta_k^{FR} = \frac{\|g_{k+1}\|^2}{\|g_k\|^2}$$

a2) Gradient conjugué variante descente conjuguée (CD)

$$\beta_k^{CD} = -\frac{\|g_{k+1}\|^2}{d_k^T g_k}$$

a3) Gradient conjugué variante de *Dai-Yuan* (DY)

$$\beta_k^{DY} = \frac{\|g_{k+1}\|^2}{d_k^T y_k}$$

b) Les méthodes où $g_{k+1}^T y_k$ figure dans le numérateur de β_k

Dans cette famille on trouve surtout les méthodes suivantes

b1) Gradient conjugué variante Polak-Ribière-Polyak(PRP)

$$\beta_k^{PRP} = \frac{g_{k+1}^T y_k}{\|g_k\|^2}$$

b2) Gradient conjugué variante Hestenes-Stiefel(HS)

$$\beta_k^{HS} = \frac{g_{k+1}^T y_k}{d_k^T y_k}$$

b3) Gradient conjugué variante Liu-Storey(LS)

$$\beta_k^{LS} = -\frac{g_{k+1}^T y_k}{d_k^T g_k}$$

Propriétés caractéristiques des familles a) et b)

Les méthodes appartenant à la classe a) ont des propriétés de convergence semblables à la méthode de Fletcher Reeves.

Les méthodes appartenant à la classe b) ont des propriétés de convergence semblables à la méthode de Polak Ribière Poliak .

Ceci étant, il suffit donc de comparer la méthode de Fletcher Reeves et la méthode de Polak Ribière Poliak et étendre cette comparaison aux deux familles.

On a vu que ces deux familles ont des propriétés de convergence semblables et utilisent presques les mêmes hypothèses.

Les premiers et les plus importants résultats théoriques de convergence ont été obtenus pour la méthode de Fletcher Reeves. Mais numériquent cette méthode présente quelques inconvé-

nients.

Contrairement à la méthode Fletcher Reeves, on n'a pas réussi à établir des résultats théoriques de convergence pour la méthode Polak Ribière Poliak.

Comme on l'a remarqué ci dessus, l'algorithme de la méthode du Gradient conjugué version Hestenes-Stiefel (HS) appartient à la famille b), et par conséquent possède des propriétés semblables à la méthode de Polak Ribière Poliak. Plus précisément la méthode HS a une bonne performance numérique, mais trés peu de résultats de convergence globale avec la recheche linéaire inexacte. On a vu dans cette thèse, qu'en s'inspirant du travail de Shi, Z.J et Shen, J. (2007), et en effectuant une nouvelle modification de la recherche linéaire inexacte d'Armijo, on a pu avoir la convergence globale de la méthode du gradient conjugué, version Hestenes et Steifel. Des tests numériques ont prouvé la performance du nouveau algorithme.

Annexe

1 Programme en fortran 77 d' Armijo

```
c*************************************************************************
                    Programme d' Armijo
c*************************************************************************
      real x(2),xp(2),g(2),gp(2),f0,alph,f,p,s,a,b,pf,t,y,phopt
      integer i,k,test,n
      sig=0.3 ;ro=0.1 ;b=10000000 ;alph=10 ;test=0 ;k=0
      print*,'le dimontion n=' ;read*,n
      do i=1,n print*,'x0(',i,')=' ;read*, xp(i) end do
10    if (test==0.and.k.le.10) then call fonc(f0,gp,xp)
      do i=1,n x(i)=xp(i)-alph*gp(i) end do
      call fonc(f,g,x) ;p=0
      do i=1,n p=p+gp(i)*gp(i) end do
      s=f0-ro*alph*p
     if (f.gt.s) then b=alph else phopt=alph ;test=1 endif
      print*,'alpha(',k,')=', alph ;alph=b/2 ;k=k+1
      go to 10
      end if
      print*,'alpha optimal=', phopt
      end
c*************************************************************************
      subroutine fonc(f,g,x)
      real f,g(2),x(2)
      f=(x(1))**2+(x(2))**4 ;g(1)=2*x(1) ;g(2)=4*(x(2))**3
```

```
      end
c*************************************************************************
c*************************************************************************
```

2 Programme en fortran 77 de Goldschtien

```
c*************************************************************************
c            Programme de Goldschtien
c*************************************************************************
      real x(2),xp(2),g(2),gp(2),f0,alph,f,p,s,a,b,pf,t,y,phopt
      integer i,k,test,n
      ro=0.1 ;delt=0.3 ;b=10000000 ;alph=10 ;test=0 ;k=0
      print*,'le dimontion n=' ;read*,n
      do i=1,n print*,'x0(',i,')=' ;read*, xp(i) end do
10    if (test==0.and.k.le.1000) then call fonc(f0,gp,xp)
      do i=1,n x(i)=xp(i)-alph*gp(i) end do
      call fonc(f,g,x) ;p=0
      do i=1,n p=p+gp(i)*gp(i) end do
      s=f0-ro*alph*p ;y=f0-delt*alph*p
     if (f.gt.s) then b=alph else if (f.lt.y) then a=alph else
      phopt=alph ;test=1 endif
      endif
      print*,'alpha(',k,')=', alph ;alph=(a+b)/2 ;k=k+1
      go to 10
      end if
      print*,'alpha optimal=', phopt
      end
c*************************************************************************
      subroutine fonc(f,g,x)
      real f,g(2),x(2)
      f=(x(1))**2+(x(2))**4 ;g(1)=2*x(1) ;g(2)=4*(x(2))**3
      end
c*************************************************************************
c*************************************************************************
```

3 Programme en fortran 77 de Wolfe

```
c*************************************************************************
          Programme de Wolfe
c*************************************************************************
      real x(2),xp(2),g(2),gp(2),f0,alph,f,p,s,a,b,pf,t,y,phopt
      integer i,k,test,n
      sig=0.3 ;ro=0.1 ;b=10000000 ;alph=10 ;test=0 ;k=0 ;print*,'le dimontion n='
      read*,n
      do i=1,n print*,'x0(',i,')=' ; read*, xp(i) end do
10 if (test==0.and.k.le.10) then call fonc(f0,gp,xp)
      do i=1,n x(i)=xp(i)-alph*gp(i) end do
      call fonc(f,g,x) ;p=0
      do i=1,n p=p+gp(i)*gp(i) end do
      s=f0-ro*alph*p
     if (f.gt.s) then b=alph else pf=0
      do i=1,n pf=pf+g(i)*gp(i) end do
      t=-pf ;y=sig*(-p)
      if (t.lt.y) then a=alph else phopt=alph ;test=1 endif
      endif
      print*,'alpha(',k,')=', alph ;alph=(a+b)/2 ;k=k+1
      go to 10
      end if
      print*,'alpha optimal=', phopt ;x=xp
      end
c*************************************************************************
      subroutine fonc(f,g,x)
      real f,g(2),x(2)
      f=(x(1))**2+(x(2))**4 ;g(1)=2*x(1) ;g(2)=4*(x(2))**3
      end
c*************************************************************************
c*************************************************************************
```

4 Programme en fortran 90 de la méthode du gradient conjugé

```
c*************************************************************************
Programme du gradient conjugé
c*************************************************************************
program CGWP
implicit none
integer, parameter : :n=10000
! test : nombre des direction engendré par la méthode du GC
! et qui ne sont pas des directions de descente
! cont : nombre des évaluations fonctionnelles
integer ik,j,k,contf,test,contg,B
! Y : la suite, g : le gradient, d : la direction de déplacement
! eps : epcilon (critère de convergence)
double precision y(n),g(n),gp(n)
double precision d(n),eps
double precision alphd,alph0
double precision b,w,alphg,f0,f1
integer (2) hour,minut,second,hsedt
integer (2) hour2,minut2,second2,hsedt2
call gettim(hour,minut,second,hsedt)
print*,hour,minut,second,hsedt
print*,'ENTRER Y'
read*, B
do ik=1,n
y(ik)=0.5
enddo
print*,'ENTRER EPS'
eps=1.e-10
b=0.9 ;w=0.1 ;contf=1 ;test=0 ;contg=1
g=df(y) ;f0=f(y) ;d=-g ;j=1 ;k=1
do while(sqrt(dot_product(g,g))>eps)
c*************************************************************************
```

```
! recherche linéaire
c**************************************************************************
alphg=0 ;alphd=100 ;alph0=1
gp=g
do
contf=contf+1
f1=f(y+alph0*d)
if(f1<=f0+w*alph0*dot_product(gp,d))then
contg=contg+1
g=df(y+alph0*d)
if(dot_product(g,d)>=b*dot_product(gp,d))then
exit
else
alphg=alph0 ;alph0=(alphg+alphd)/2.
endif
else
alphd=alph0 ;alph0=(alphg+alphd)/2.
endif
enddo
c**************************************************************************
! le nouveau point
c**************************************************************************
y=y+alph0*d
k=k+1
if(j<n)then
if(B=1)then
d=-g+(dot_product(g,g-gp)/dot_product(d,g-gp))*d
end if
if(B=2)then
d=-g+(dot_product(g,g)/dot_product(gp,gp))*d
end if
if (B=1)then
d=-g+(dot_product(g,g)/dot_product(gp,gp))*d
end if
j=j+1
```

```
if(dot_product(g,d)>=0)then
j=1
d=-g
test=test+1
endif
else
d=-g
j=1
endif
c*********************************************************************************
!test de la décroissance négligeable
c*********************************************************************************
if(f0-f1<1.e-14)then
print*
print*,'THE NORM OF GRADIENT'
print '(e20.2)',sqrt(dot_product(g,g))
g=0
endif
f0=f1
enddo
c*********************************************************************************
! affichage du temps
c*********************************************************************************
call gettim(hour2,minut2,second2,hsedt2)
print*,hour2,minut2,second2,hsedt2
print*,hour2-hour
print*,minut2-minut
print*,second2-second
print*,hsedt2-hsedt
c*********************************************************************************
!affichage des résultats
c*********************************************************************************
print*
print*,'THE NUMBER OF ITERATION'
print*,k
```

```
print*,'THE FUNCTION VALUE'
print '(e20.2)',f(y)
print*,'NOMBRE D EVALUATIONS FONCTIONNELLES',contf
print*,'NOMBRE D EVALUATIONS GRADIENT',contg
print*,'test=',test
! les procédures intérieurs
contains
end
*******************************************************************************
*******************************************************************************
```

Bibliographie

[1] M. Al-Baali, Descent property and global convergence of the Fletcher-Reeves method with inexact line search, IMA J. Numer. Anal., 5 (1985), pp. 121–124.

[2] L. Armijo, Minimization of function having lipschitz continous first partial derivatives, Pacific Journal of Mathematics, Vol. 16 (1966), pp.1-3.

[3] L. Andrieu, Optimisation sous contrainte en probabilité, THESE présentée pour l'obtention du titre de Docteur *de* l'Ecole Nationale des Ponts et Chaussées, décembre (2004)

[4] M. Belloufi, R. Benzine, and Y. Laskri, Modification of the Armijo line search to satisfy the convergence properties of HS method, An International Journal of Optimization and Control : Theories & Applications (IJOCTA), 141(2013), pp. 145-152 .

[5] M. Belloufi and R. Benzine, Global Convergence Properties of the HS Conjugate Gradient Method. Applied Mathematical Sciences, Vol. 7, 142(2013), pp. 7077-7091

[6] E. M. L. Beale, A derivative of conjugate gradients, in Numerical Methods for Nonlinear Optimization, F. A. Lootsma, ed., Academic Press, London, (1972), pp. 39–43.

[7] M. S. Bazaraa and H. D. Sherali, et C. M. Shetty , Nonlinear Programming, Theory and Algorithms, *Wiley-Interscience(1993).*

[8] I. Bongartz, A. R. Conn, N. I. M. Gould, and P. L. Toint : Constrained and unconstrained testing environments, ACM Trans. Math. Software, 21 (1995), pp. 123–160.

[9] C. G. Broyden, The convergence of a class of double-rank minimization algorithms 1. general considerations, J. Inst. Math. Appl., 6 (1970), pp. 76–90.

[10] A. Buckley, Conjugate gradient methods, in Nonlinear Optimization 1981, M. J. D. Powell, ed., Academic Press, London, (1982), pp. 17–22.

[11] C. W. Carroll, The created reponse surface technique for optimizing nonlinear restrained systems, *Operations Res.*, Vol. 9 (1961), pp. 169-84.

[12] H. P. Crowder and P. Wolfe, Linear convergence of the conjugate gradient method, IBM J. Res. Dev., 16 (1969), pp. 431–433.

[13] X.D. Chen and J. Sun, Global convergence of a two-parameter family of conjugate gradient methods without line search, Journal of Computational and Applied Mathematics 146 (2002), pp. 37–45.

[14] A. Cauchy, Analyse mathématique, Méthode générale pour la résolution des systèmes d'équations simultanées, Comptes Rendus de l'Académie des Sciences de Paris, (1847) t-25, pp. 536-538.

[15] Y. H. Dai, Analyses of conjugate gradient methods, Ph.D. thesis, Institute of Computational Mathematics and Scientific/Engineering Computing, Chinese Academy of Sciences, 1997.

[16] Y. H. Dai, New properties of a nonlinear conjugate gradient method, Numer. Math., 89 (2001), pp. 83–98.

[17] Y. H. Dai, A nonmonotone conjugate gradient algorithm for unconstrained optimization, J. Syst. Sci. Complex., 15 (2002), pp. 139–145.

[18] Y. H. Dai, J. Y. Han, G. H. Liu, D. F. Sun, H. X. Yin, and Y. Yuan, Convergence properties of nonlinear conjugate gradient methods, SIAM J. Optim., 10 (1999), pp. 345–358.

[19] Y. H. Dai and L. Z. Liao, New conjugacy conditions and related nonlinear conjugate gradient methods, Appl. Math. Optim., 43(2001), pp. 87–101.

[20] Y. H. Dai and Q. Ni, Testing different conjugate gradient methods for large-scale unconstrained optimization, J. Comput. Math., 21 (2003), pp. 311–320.

[21] Y. H. Dai and Y. Yuan, Convergence properties of the Fletcher-Reeves method, IMA J. Numer. Anal., 16 (1996), pp. 155–164.

[22] Y. H. Dai and Y. Yuan, Convergence properties of the conjugate descent method, Adv. Math. (China), 26 (1996), pp. 552–562.

[23] Y. H. Dai and Y. Yuan, Further studies on the Polak-Ribière-Polyak method, Research report ICM-95-040, Institute of Computational Mathematics and Scientific/Engineering Computing, Chinese Academy of Sciences, 1995.

[24] Y. H. Dai and Y. Yuan, A nonlinear conjugate gradient method with a strong global convergence property, SIAM J. Optim., 10 (1999), pp. 177–182.

[25] Y. H. Dai and Y. Yuan, Convergence of the Fletcher-Reeves method under a generalized Wolfe wearch, J. Comput. Math., 2 (1996), pp. 142–148.

[26] Y. H. Dai and Y. Yuan, A class of globally convergent conjugate gradient methods, Research report ICM-98-030, Institute of Computational Mathematics and Scientific/Engineering Computing, Chinese Academy of Sciences, 1998.

[27] Y. H. Dai and Y. Yuan, Extension of a class of nonlinear conjugate gradient methods, Research report ICM-98-049, Institute of Computational Mathematics and Scientific/Engineering Computing, Chinese Academy of Sciences, 1998.

[28] Y. H. Dai and Y. Yuan, Nonlinear conjugate gradient methods, Shanghai Science and Technology Publisher, Shanghai, 2000.

[29] Y. H. Dai and Y. Yuan, A three-parameter family of hybrid conjugate gradient method, Math. Comp., 70 (2001), pp. 1155–1167.

[30] Y. H. Dai and Y. Yuan, An efficient hybrid conjugate gradient method for unconstrained optimization, Ann. Oper. Res., 103 (2001), pp. 33–47.

[31] Y. H. Dai and Y. Yuan, A class of globally convergent conjugate gradient methods, Sci. China Ser. A, 46 (2003), pp. 251–261.

[32] J. W. Daniel, The conjugate gradient method for linear and nonlinear operator equations, SIAM J. Numer. Anal., 4 (1967), pp. 10–26.

[33] G. Fasano, Planar conjugate gradient algorithm for large-scale unconstrained optimization. Part 1 : Applications, Journal of Optimization Theory and Applications 125(2005), pp. 543–558.

[34] R. Fletcher, Practical Methods of Optimization vol. 1 : Unconstrained Optimization, John Wiley & Sons, New York, 1987.

[35] R. Fletcher and C. Reeves, Function minimization by conjugate gradients, Comput. J., 7(1964), pp. 149–154.

[36] R. Fletcher and M. J. D. Powell, A rapidly convergent descent method for minimization, Computer. J., 6 (1963), pp. 163-168.

[37] A. V. Fiacco and G. P. McCormick, Nonlinear Programming, *John Wiley, New York,* (1968).

[38] H. Fan, Z. Zhu et A. Zhou, A New Descent Nonlinear Conjugate Gradient Method for Unconstrained Optimization, Applied Mathematics, 2011, 2, 1119-1123.

[39] J. C. Gilbert and J. Nocedal, Global convergence properties of conjugate gradient methods for optimization, SIAM J. Optim., 2 (1992), pp. 21–42.

[40] J. C. Gilbert , Eléments d'optimisation différentiable : théorie et algorithmes, Notes de cours, École Nationale Supérieure de Techniques Avancées, Paris, (2007).

[41] L. Grippo and S. Lucidi, A globally convergent version of the Polak-Ribière conjugate gradient method, Math. Prog., 78 (1997), pp. 375–391.

[42] L. Grippo and S. Lucidi, Convergence conditions, line search algorithms and trust region implementations for the Polak-Ribière conjugate gradient method, Technical Report 25-03, Dipartimento d'Informatica et Sistemistica, Università di Roma "La Sapienza", November (2003).

[43] A.A. Goldstein and J.F. Price, An effective algorithm for minimzation, Num.Math., 10 (1969), pp. 184-189.

[44] A. A. Goldstein, On steepest descent, SIAM J. on Control A, Vol. 3, No. 1 (1965), pp. 147-151.

[45] W. W. Hager and H. Zhang, A new conjugate gradient method with guaranteed descent and an efficient line search, November 17, (2003).

[46] W. W. Hager and H. Zhang, CG DESCENT, a conjugate gradient method with guaranteed descent, January 15, (2004) (to appear in ACM TOMS).

[47] J. Y. Han, G. H. Liu, and H. X. Yin, Convergence properties of conjugate gradient methods with strong Wolfe linesearch, Systems Sci. Math. Sci., 11 (1998), pp. 112–116.

[48] M. R. Hestenes and E. L. Stiefel, Methods of conjugate gradients for solving linear systems, J. Research Nat. Bur. Standards, 49 (1952), pp. 409–436.

[49] Y. F. Hu and C. Storey, Global convergence result for conjugate gradient methods, J. Optim. Theory Appl., 71 (1991), pp. 399–405.

[50] R. A. Horn, and C. R. Johnson, Matrix Analysis, Cambridge University Press, (1990)

[51] G. H. Liu, J. Y. Han and H. X. Yin, Global convergence of the Fletcher-Reeves algorithm with an inexact line search, Appl. Math. J. Chinese Univ. Ser. B, 10 (1995), pp. 75–82.

[52] Y. Liu and C. Storey, Efficient generalized conjugate gradient algorithms, Part 1 : Theory, J. Optim. Theory Appl., 69 (1991), pp. 129–137.

[53] J. L. Nazareth, A conjugate direction algorithm without line searches, J. Optim. Theory Appl., 23 (1977), pp. 373–387.

[54] J. L. Nazareth, Conjugate-gradient methods, Encyclopedia of Optimization, C. Floudas and P. Pardalos, eds., Kluwer Academic Publishers, Boston, (1999).

[55] J. M. Perry, A class of conjugate gradient algorithms with a two-step variable-metric memory, Discussion Paper 269, Center for Mathematical Studies in Economics and Management Sciences, Northwestern University, Evanston, Illinois, (1977).

[56] E. Polak and G. Ribière, Note sur la convergence de directions conjugées, Rev. Francaise Informat Recherche Opertionelle, 3e Année 16 (1969), pp. 35–43.

[57] B. T. Polyak, The conjugate gradient method in extreme problems, USSR Comp. Math. Math. Phys., 9 (1969), pp. 94–112.

[58] M. J. D. Powell, Some convergence properties of the conjugate gradient method, Math. Prog., 11 (1976), pp. 42–49.

[59] M. J. D. Powell, Restart procedures of the conjugate gradient method, Math. Prog., 2 (1977), pp. 241–254.

[60] M. J. D. Powell, Nonconvex minimization calculations and the conjugate gradient method, Numerical Analysis, Lecture Notes in Mathematics, Vol. 1066, Springer-Verlag, Berlin, (1984), pp. 122–141.

[61] M. Rivaie, M. Mamat , L. W. June ,and I. Mohd, A new class of nonlinear conjugate gradient coefficients with global convergence properties. Applied Mathematics and Computation 218 (2012) 11323–11332.

[62] D. F. Shanno, On the convergence of a new conjugate gradient algorithm, SIAM J. Numer. Anal., 15 (1978), pp. 1247–1257.

[63] J. Sun, X.Q. Yang, and X.D. Chen, Quadratic cost flow and the conjugate gradient method, European Journal of Opera-tional Research 164 (2005) 104–114.

[64] J. Sun and J. Zhang, Global convergence of conjugate gradient methods without line search, Ann. Oper. Res., 163 (2001), pp. 161–173

[65] Z.J. Shi and J. Shen, Convergence of descent method without line search, Appl. Math. Comput. 167 (2005) 94–107.

[66] Z.J. Shi, Restricted PR conjugate gradient method and its global convergence, Advances in Mathematics 31 (1) (2002) 47–55 (in Chinese).

[67] Z.J.Shi and J.Shen, Convergence of Liu-Storey conjugate gradient method, European Journal of Operational Research,182(2)(2007), 552– 560 .

[68] Z.J. Shi and J. Shen, Convergence of Polak-Rebière-Polyak conjugate gradient method, Nonlinear Analysis, 66 (2007), 1428–1441.

[69] B. Sellami, Y. Laskri et R. Benzine, A new two-parameter family of nonlinear conjugate gradient methods. Optimization, 2013 http ://dx.doi.org/10.1080/02331934.2013.830118.

[70] D. Touati-Ahmed and C. Storey, Efficient hybrid conjugate gradient techniques, J. Optim. Theory Appl., 64 (1990), pp. 379–397.

[71] P. Wolfe, Convergence conditions for ascent methods, SIAM Review, 11 (1969), pp. 226–235.

[72] P. Wolfe, Convergence conditions for ascent methods. II : Some corrections, SIAM Review, 13 (1971), pp. 185–188.

[73] P. Wolfe, A duality theorem for nonlinear programming, *Quart. Appl. Math.*, 19 *(1961),* pp. 239-244.

[74] Z. Wei, S. Yao and L. Liu, The Convergence Properties of Some New Conjugate Gradient Methods, Applied Mathematics and Computation, Vol. 183, No. 2, 2006, pp.1341-1350.

[75] H. Yabe and M. Takano, Global convergence properties of nonlinear conjugate gradient methods with modi ?ed secant condition, Comput. Optim. Appl., 28, (2004), pp. 203–225.

[76] Y. Yuan, Analysis on the conjugate gradient method, Optim. Methods Softw., 2 (1993), pp.19–29.

[77] Y. Yuan, Numerical Methods for Nonlinear Programming, Shanghai Scientific & Technical Publishers, 1993.

[78] J. Z. Zhang, N. Y. Deng, and L. H. Chen, new quasi-Newton equation and related methods for unconstrained optimization, J. Optim. Theory Appl., 102 (1999), pp. 147–167.

[79] J. Z. Zhang and C. X. Xu, Properties and numerical performance of quasi-Newton methods with modified quasi-Newton equations, J. Comput. Appl. Math., 137 (2001), pp. 269–278.

[80] G. Zoutendijk, Nonlinear Programming, Computational Methods, in Integer and Nonlinear Programming, J. Abadie, ed., North-Holland, Amsterdam, (1970), pp. 37–86.

[81] G. Zoutendijk, Methods of Feasible Directions, Elsevier, Amesterdam (1960).