



*République Algérienne Démocratique et Populaire*  
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique  
Université Larbi Tebessi –TEBESSA-  
Faculté des Sciences Exactes et Sciences de la Nature et de la Vie  
Département de mathématique et informatique



## *Thèse*

Présentée en vue de l'obtention du diplôme de Doctorat  
En : MATHEMATIQUES  
Option : MATHEMATIQUES APPLIQUÉES

Par : **CHERGUI Ahmed**

**Intitulée :**

***Développement Récente Des méthodes  
d'optimisation sans contraintes***

**Devant le jury :**

<b>ZARAI Abderrahmane</b>	Professeur	Université de Tébéssa	Président
<b>BOUALI Tahar</b>	MCA	Université de Tébéssa	Rapporteur
<b>BELLOUFI Mohammed</b>	MCA	Université de Souk Ahras	Examineur
<b>RAFIK Guefaifia</b>	MCA	Université de Tébéssa	Examineur
<b>SELLAMI Badreddine</b>	MCA	Université de Souk Ahras	Examineur

**Date de soutenance : .. / .. / ....**

# Table des matières

<b>1</b>	<b>Optimisation sans contraintes</b>	<b>13</b>
1.1	Rappels de pré-requis Mathématiques . . . . .	13
1.1.1	Espace normé . . . . .	13
1.1.2	Normes matricielles . . . . .	14
1.1.3	Valeurs propres et vecteurs propres . . . . .	15
1.1.4	Matrices symétriques . . . . .	15
1.2	Rappel de quelques définitions . . . . .	17
1.2.1	Convergence des algorithmes. . . . .	17
1.2.2	Vitesse de convergence . . . . .	18
1.2.3	Fonctions convexes . . . . .	18
1.2.4	Fonctions convexes différentiable . . . . .	20
1.3	Conditions d'optimalité des problèmes de minimisation sans contraintes : . . . . .	22
1.3.1	Minima locaux et globaux . . . . .	22
1.4	Direction de descente . . . . .	23
1.4.1	Schéma général des algorithmes d'optimisation sans contraintes . . . . .	26
1.4.2	Conditions nécessaires d'optimalité . . . . .	27
1.4.3	Conditions suffisantes d'optimalité . . . . .	28
<b>2</b>	<b>Recherche linéaire</b>	<b>32</b>
2.0.4	Introduction . . . . .	32
2.0.5	Objectifs de la recherche linéaire . . . . .	33
2.0.6	But de la recherche linéaire . . . . .	34
2.0.7	Intervalle de sécurité . . . . .	34
2.0.8	Algorithme de base . . . . .	35
2.1	Recherches linéaires exactes . . . . .	35
2.2	Recherches linéaires inexactes . . . . .	38
2.2.1	Caractérisation de l'intervalle de sécurité . . . . .	39
2.3	Condition de Zoutendijk . . . . .	45

<b>3</b>	<b>La méthode du gradient conjugué</b>	<b>48</b>
3.1	La méthode du gradient conjugué linéaire . . . . .	48
3.2	Méthodes itératives d'optimisation sans contraintes . . . . .	48
3.2.1	Principe des méthodes de descente . . . . .	49
3.2.2	Principe des méthodes de gradient(méthodes de la plus forte descente) . . . . .	50
3.2.3	Les méthodes utilisant des directions conjuguées . . . . .	56
3.3	La methode du gradient conjugué non linéaire . . . . .	57
3.4	la convergence de quelques méthodes du gradient conjugué non linéaire avec la recherche linéaire inexacte . . . . .	59
3.4.1	méthode de Fletcher-Reeves . . . . .	62
3.4.2	. La propriété de descente de la mètode de FR . . . . .	63
3.4.3	Méthode de Polak-Ribière-Polyak . . . . .	68
3.4.4	Méthode de la descente conjugué . . . . .	71
3.4.5	Mèthode de Dai-Yuan . . . . .	74
<b>4</b>	<b>Une nouvelle méthode de gradient conjugué non linéaire avec la recherche linéaire de Wolf-Powell fort</b>	<b>80</b>
4.1	1Introduction . . . . .	80
4.2	nouvelle formule de GC . . . . .	82
4.3	analyse convergente . . . . .	82
4.3.1	Analyse convergente basée sur une recherche linear exacte	82
4.3.2	Convergence globale par la recherche linear exacte . . . . .	83
4.3.3	Analyse convergente basée sur une recherche linéaire inexacte . . . . .	85
4.3.4	Analyse convergence globale . . . . .	88
4.4	.Résultats numériques et discussions . . . . .	89
4.5	conclusion . . . . .	96

## Remerciements

*Je remercie particulièrement, mon encadreur **Taher Bouali**, Docteur à l'Université l'arbie Tebesse Tebessa , pour m'avoir proposer le sujet de ce 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 monsieur **Zarai abderrahmane**, professeur à l'Université l'arbie Tebesse Tebessa, pour avoir bien voulu me faire l'honneur d'accepter de présider le jury.*

*De même je remercie monsieur **belloufi Mohammed**, maître de conférence classe A, à l'Université souk ahras , et monsieur **Salami Badred-dine**, maître de conférence classe A à l'Université souk ahras, et aussi monsieur **Rafik guefaifia**, maître de conférence classe A, à l'Université l'arbie Tebesse Tebessa. pour l'honneur qu' ils m'ont fait de bien vouloir accepter de faire partie du jury*

*Je remercie également tout ceux qui ont contribué de près ou de loin à l'élaboration de ce travail, en particulier, ma femme pour ses encouragements.*

*Enfin, Merci à toute ma famille, en particulier, ma **mère**, mon **père**, mes enfants , et à tous mes amis et tous mes collègues sans exception.*

## Résumé

Le but de cette thèse est de développer la méthode de gradient conjugué pour résoudre les problèmes d'optimisation sans contraintes. Tout en trouvant une nouvelle formule qui assure la convergence globale et vérifie la condition suffisante de descente sous la recherche linéaire exacte et la recherche linéaire inexacte de Wolfe-Powell forte, cette nouvelle formule donne d'excellents résultats numériques en les comparant à certaines méthodes traditionnelles.

**Mots clés :**

optimisation sans contraintes, gradient conjugué, convergence globale, Wolfe-Powell forte,

# Abstract

The purpose of this thesis is to develop the conjugate gradient method for solving large-scale unconstrained problems. while finding a new formula that ensured global convergence and verifies the sufficient descent condition, under the exact linear search and the inexact linear search of Strong Wolfe-Powell. this new formula gives excellent numerical results by comparing them to some traditional methods.

**Mots clés :**

unconstrained optimization, conjugate gradient , global convergence, Strong Wolf Line Search,

ملخص

الهدف من هذه الأطروحة هو تطوير طريقة التدرج المقارن لحل مسائل الاستمثال غير المقيدة من خلال ايجاد طريقة جديدة تضمن شرط التقارب الكلي وتحقق الشرط الكافي للنزول باستعمال البحث الخطي غير الدقيق لـوولف فوروتعطي نتائج عددية ممتازة من خلال مقارنتها مع بعض الطرق التقليدية

# Introduction

Le problème d'optimisation trouve des applications dans plusieurs domaines, tels que la physique classique du continuum, la physique théorique, la physique mathématique et numérique, la physique des particules et nucléaire, la physico-chimie, les mathématiques pures, la physique mathématique, la dynamique des fluides, la science actuarielle, l'économie de l'information appliquée, l'astrostatistique, la biostatistique, statistiques des entreprises, routage du trafic dans les systèmes de télécommunication (64.), sécurité cybophysique (65.), systèmes de transport intelligents (66.) et réseaux intelligents (67.).

La résolution de problèmes d'optimisation est devenue un sujet central en recherche opérationnelle, le nombre de problèmes d'aide à la décision pouvant être formalisés sous la forme d'un problème d'optimisation étant en forte croissance [72]. Les problèmes d'apprentissage de réseaux de neurones, de planification des tâches ou encore d'identification sont, par exemple, des problèmes communs d'optimisation. En outre, l'Homme cherche à améliorer sa vie quotidienne, l'Homme aime la perfection ! Et sans qu'il se rende compte, il essaye à minimiser ses charges son loyer ou la consommation de sa voiture... Il tente toujours à vrai dire à optimiser que se soit minimiser ses dépenses et maximiser ses biens. Le mathématicien vient pour concrétiser le vœux de l'Homme en modélisant les problèmes de la vie sous des fonctions coûts en utilisant les différents types d'optimisation. De nos jours l'optimisation est devenue un domaine indispensable pour résoudre plusieurs problèmes que se soit dans l'industrie ou d'autres secteurs [73]. En effet nous avons assisté ces dernières années à une croissance très rapide des travaux utilisant les méthodes d'optimisation. Cette tendance peut être observée dans tous les domaines de la science.

**Optimiser** : rendre optimal, donner à quelque chose les meilleures conditions d'utilisation, de fonctionnement ou de rendement au regard de certaines circonstances. (Déf. du Larousse).

## **Les gens optimisent :**

Des compagnies aériennes établissent les horaires des équipages et des avions avant de minimiser le coût. Des investisseurs cherchent à constituer des portefeuilles boursiers qui évitent des risques excessifs tout en donnant un bon rendement. Des fabricants visent le maximum d'efficacité dans la conception et l'opération de leurs processus de production.

## **La nature optimise :**

Des systèmes physiques tendent vers un état d'énergie minimale. Des molécules dans un système chimique réagissent entre eux jusqu'à ce que l'énergie potentielle totale de leurs électrons soit minimisée. Des rayons de lumière

suivent des trajectoires qui minimisent leur temps de trajet. L'optimisation est un outil important dans la théorie des décisions et dans l'analyse des systèmes physiques. Avant de l'utiliser, nous devons d'abord identifier un objectif. Ceci est une mesure quantitative de la performance du système. Cet objectif pourrait être : bénéfique, temps, énergie potentielle, ou toute autre quantité ou combinaison des quantités qui peut être représentée par un seul nombre. L'objectif dépend de certaines caractéristiques du système, appelées des variables ou des inconnues. Notre but est de trouver des valeurs de variables qui optimisent l'objectif. Souvent les variables sont restreintes, ou contraintes, d'une certaine manière. Par exemple, la densité ou l'énergie ou le taux d'intérêt ne peuvent pas être négatifs. Le processus de l'identification de l'objectif, variables et contraintes pour un problème donné est appelé la modélisation. La construction d'un modèle approprié est la première étape. Parfois c'est l'étape la plus importante du processus d'optimisation. Si le modèle est simpliste, il ne donnera pas d'informations utiles concernant le problème pratique. Par contre, si le modèle est trop complexe, il pourra devenir trop difficile à résoudre. Une fois que le modèle a été formulé, un algorithme d'optimisation peut être utilisé avant de trouver sa solution. Ceci est fait, normalement, par un ordinateur. Il n'existe pas un algorithme universel d'optimisation. Plutôt, il existe de nombreux algorithmes, chacun étant adapté à un type particulier de problème d'optimisation. Il est souvent de la responsabilité de l'utilisateur/programmeur/modéleur de choisir un algorithme approprié pour l'application spécifique. Ce choix est très important puisqu'il déterminera si le problème est résolu rapidement ou pas et effectivement, même si la solution est déjà trouvée. Après l'application d'un algorithme d'optimisation au problème, on doit être capable de reconnaître si l'on a réussi à trouver une solution. Dans certains cas, il existe des expressions mathématiques, les conditions d'optimalité, pour vérifier si l'ensemble des variables obtenu est bien la solution du problème. Si les conditions d'optimalité ne sont pas vérifiées, elles peuvent donner des indications sur la manière d'améliorer notre estimation actuelle. Finalement, le modèle peut être amélioré par l'application de techniques comme l'analyse de sensibilité qui révèle la sensibilité de la solution aux changements dans le modèle et dans les données. En résumé, voici les étapes principales :

- 1- identifier un objectif et des variables.
- construire un modèle approprié ;
- choisir un algorithme d'optimisation ;
- vérifier les conditions d'optimalité ;
- faire éventuellement une analyse de sensibilité..

Nous divisons les problèmes d'optimisation en plusieurs familles et catégories (voir la Figure 0.0.1)

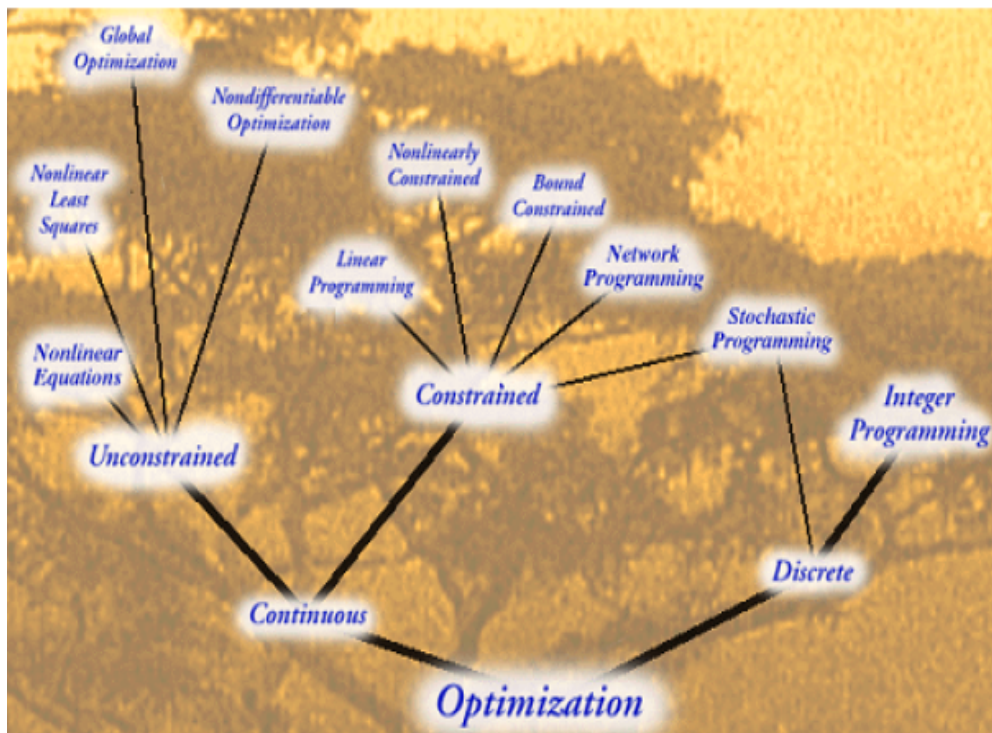


FIGURE 0.0.1. L'arbre d'optimisation

- 1- optimisation continue.
- optimisation discrète ;
- optimisation stochastique.

Parmi les problèmes pratiques en optimisation continue, nous pouvons citer les suivants :

- trouver le trajectoire optimal pour un avion, une navette spatiale ou le bras d'un robot ;

- identifier les propriétés sismiques de l'écorce terrestre en ajustant un modèle à un ensemble d'enregistrements pris sur le terrain (en fait, un problème inverse) ; composer une porte-feuille d'investissements afin de maximiser le retour tout en gardant un niveau de risque acceptable ;

- contrôler un processus chimique ou un dispositif mécanique afin d'optimiser la performance ;

- calculer la forme optimale d'une voiture, un voilier ou un avion

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 18<sup>ième</sup> siècle dans les travaux de :

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

- Cauchy fut le premier à mettre en oeuvre 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.

Parmi les plus anciennes méthodes utilisées pour résoudre cet problème du type ,  $(P)$

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

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

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 , 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 (*méthode de Fletcher-Reeves*) puis en 1969 par Polak, Ribière et Ployak (*méthode de Polak-Ribière-Ployak*). Une autre variante a été étudiée en 1987

par Fletcher (*Méthode de la descente conjuguée*). Une nouvelle variante a été proposée en 1991 par Liu et Storey (*Méthode de Liu et Storey*). Une Autre variante a été étudiée en 1999 par Dai et Yuan (*Méthode de Dai et Yuan*) Récemment, Wei et al. (2006) ont proposé une nouvelle formule.

Le but de cette thèse est donc d'améliorer le travail de Wei et al. (2006), pour cela nous proposons une nouvelle formule de méthode du gradient conjugué pour l'optimisation sans contraintes dans laquelle

$$B_k^{WYLM} = \frac{g_k^T \left( g_k - \frac{\|g_k\|}{\|g_{k-1}\|} d_{k-1} \right)}{g_{k-1}^T g_{k-1}}. \quad (0.2)$$

nous présentons La preuve de la condition suffisante de descente de la méthode de gradient conjugué de (WYLM) sous la recherche linéaire exacte et inexacte de Wolfe-Powell forte et la convergence globale

### Plan de thèse

Cette thèse comporte une introduction, quatre chapitres, des simulations numériques et une conclusion

#### Chapitre 1

Ce chapitre contient des remarques préliminaires et des notions essentielles nécessaires pour notre travail. On insiste surtout sur les notions de direction de descente, la notion de convexité ainsi que les notions de conditions d'optimalité des problèmes de minimisation sans contraintes.

#### Chapitre 2

Ce chapitre comporte les grandes lignes des méthodes d'optimisation sans contraintes basées sur les directions de descente et les recherches linéaires. Ces méthodes génèrent des suites  $\{x_k\}_{k \in \mathbb{N}}$  de la façon suivante :  $x_{k+1} = x_k + \alpha_k d_k$  On suppose connaître la direction de descente  $d_k$  au point  $x_k$ . La recherche linéaire consiste à trouver  $\alpha_k$  de façon à diminuer la fonction  $f$  suffisamment le long de cette direction

#### Chapitre 3

Dans ce chapitre, on étudie tout d'abord la contribution de la recherche linéaire à la convergence des algorithmes à direction de descente. Ce n'est qu'une contribution. parce que la recherche linéaire seule ne pourra pas assurer la convergence des itère. il est claire que le choix de la direction de descente joue aussi un rôle. Cela se traduit par une condition dite de *Zoutendijk* que assure la convergence des méthodes du gradient conjugué. Puis on donne les principes des méthodes de descente et du gradient ainsi qu'une description des méthodes utilisant des directions conjugués notamment la méthode du gradient conjugué. puis on étudie la convergence de quelques méthodes de

gradient conjugué avec la recherche linéaire inexacte de Wolf. On s'intéressera spécialement sur les méthodes de *FR. WYL. PRP. DY*.

**Chapitre 4.** est la partie la plus importante de ce travail, car il contient une nouvelle famille de coefficient  $B_k$  de la méthode de gradient conjugué pour résoudre les problèmes d'optimisation sans contrainte à grand taille et démontrons que ce  $B_k$  assurée la condition suffisent de descente et la problèmes de convergence sous la recherche linéair exacte et inexacts de Wolf fort, et on termine par des résultats numérique avec quelque fonctions test et on compare le nouveau  $B_k$  avec les méthodes de *WYL. PRP. DY, FR*

Les résultats numériques montrent que cette méthode est efficace pour les problèmes des tests données

Enfin, le manuscrit se termine par une conclusion générale

# Chapitre 1

## Optimisation sans contraintes

### 1.1 Rappels de pré-requis Mathématiques

Nous allons ici rappeler brièvement les définitions de quelques notions importantes auxquelles nous ferons appel par la suite.

#### 1.1.1 Espace normé

##### L'espace euclidien $\mathbb{R}^n$

Soit  $\mathbb{R}^n$  L'espace vectoriel réel de dimension  $n \in \mathbb{N}$ . On notera sa base canonique  $e_1, \dots, e_n$ . On munit  $\mathbb{R}^n$  du produit scalaire usuel  $\langle \cdot, \cdot \rangle$  et de la norme associée  $\|\cdot\|_2$ . C'est 'a dire que si dans la base canonique de  $\mathbb{R}^n$  les vecteurs  $U; V \in \mathbb{R}^n$  s'écrivent  $U = (u_1, u_2, \dots, u_n)$  et  $V = (v_1, v_2, \dots, v_n)$ , alors :

$$\langle U, V \rangle = U^T V = u_1 v_1 + u_2 v_2 + \dots + u_n v_n$$

$$\|\cdot\| = \langle U, U \rangle^{1/2} = \sqrt{u_1^2 + u_2^2 + \dots + u_n^2}$$

On parle alors de l'espace euclidien  $(\mathbb{R}^n; \langle \cdot, \cdot \rangle)$  de dimension  $n$ . On y vérifie l'inégalité de Cauchy-Schwartz :

$$|\langle U, V \rangle| \leq \|U\| \|V\|$$

##### Normes de $\mathbb{R}^n$

Une norme  $\|\cdot\|$  sur  $\mathbb{R}^n$  est une application de  $\mathbb{R}^n$  dans  $\mathbb{R}^+$  vérifiant :

$\forall X, Y \in \mathbb{R}^n \forall \lambda \in \mathbb{R}$  :

(separation)  $\|X\| = 0 \implies X = 0.$

(homogénéité)  $\|\lambda X\| = |\lambda| \|X\|.$

(sous-additivité)  $\|X + Y\| \leq \|X\| + \|Y\|.$

Lorsque l'on munit  $\mathbb{R}^n$  d'une norme  $\|\cdot\|$ , on parle de l'espace normé  $(\mathbb{R}^n, \|\cdot\|)$ . Voici plusieurs exemples de normes sur  $\mathbb{R}^n$  :

la norme 1 :  $\|X\|_1 = \sum_{i=1}^n |x_i|$ .

la norme 2 :  $\|X\|_2 = \left( \sum_{i=1}^n x_i^2 \right)^{\frac{1}{2}}$ .

la norme p :  $\|X\|_p = \left( \sum_{i=1}^n x_i^p \right)^{\frac{1}{p}}$ .

la norme p :  $\|X\|_\infty = \max(|x_1|, \dots, |x_n|)$

Sur  $\mathbb{R}^n$  toutes les normes sont équivalentes, c'est à dire si  $\|\cdot\|_a$  et  $\|\cdot\|_b$  désignent deux normes de  $\mathbb{R}^n$ , il existe  $c_1, c_2 > 0$  tels que  $\forall X \in \mathbb{R}^n$  :  $\|\cdot\|_a \leq c_1 \|\cdot\|_b$  et  $\|\cdot\|_a \leq c_2 \|\cdot\|_b$

L'espace normé  $(\mathbb{R}^n, \|\cdot\|)$  est un espace complet : toute suite de Cauchy est convergente.

## 1.1.2 Normes matricielles

**Définition 1.1** *Définition 1.2* : Soit  $\|\cdot\|$  une norme quelconque sur  $\mathbb{R}^n$ , on appelle norme matricielle subordonnée de la matrice carrée  $A$  d'ordre  $n$

$$\|A\| = \max_{\|X\| \neq 0} \frac{\|AX\|}{\|X\|} = \max_{\|X\| \neq 0} \|AX\| \text{ si } \|X\| = 1$$

**Proposition 1.1** (a) Une norme matricielle subordonnée vérifie les propriétés suivantes

$$\begin{aligned} \|A\| = 0 &\iff A = 0 \text{ pour tout } A, \\ \|\lambda A\| &= |\lambda| \|A\| \text{ pour tout } A, \lambda \in \mathbb{R}, \\ \|A + B\| &\leq \|A\| + \|B\| \text{ pour tout } A, B, \\ \|AB\| &\leq \|A\| \|B\| \text{ pour tout } A, B, \end{aligned}$$

(b) Soit  $A = (a_{ij})$  une matrice carrée réelle, pour les normes vectorielles usuelles on a les normes matricielles subordonnées :  $\|X\|_\infty = \max_{j=1}^{j=n} |a_{ij}|$

et  $1 \leq i \leq n$ ,

$$\|X\|_1 = \max_{i=1}^{i=n} \sum_{j=1}^{j=n} |a_{ij}| \text{ et } 1 \leq j \leq n,$$

$$\|X\|_2 = \sqrt{\rho(A^T A)} = \|A^T\|_2$$

ou  $\rho(A^T A)$  est le rayon spectral de la matrice symétrique  $A^T A$

### 1.1.3 Valeurs propres et vecteurs propres

**Définition 1.3** *Le nombre complexe  $\lambda$  est valeur propre de la matrice carrée  $A$  si la matrice  $(A - \lambda I)$  est singulière, autrement dit s'il existe un vecteur  $X$  non nul tel que :  $AX = \lambda X$*

**Proposition 1.2** *Toute matrice carrée  $A$  d'ordre  $n$  a  $n$  valeurs propres complexes, qui sont les racines du polynôme caractéristique  $\det(A - \lambda I)$ . On a :*

$$\det(A) = \lambda_1 \cdot \lambda_2 \cdot \lambda_3 \dots \lambda_n$$

et

$$\text{tr}(A) = \lambda_1 + \lambda_2 + \lambda_3 \dots + \lambda_n$$

### 1.1.4 Matrices symétriques

**Définition 1.4** *La transposée d'une matrice  $A \in M_{m,n}(\mathbb{k})$  est la matrice notée  $A^T \in M_{n,m}(\mathbb{k})$  obtenue en inter changeant les lignes et les colonnes. Soit  $A = (a_{ij})$  et  $B = A^T = (b_{ij})$ , on a donc  $b_{ij} = a_{ji}$ .*

**Définition 1.5** *Une matrice carrée  $A$  est symétrique si :  $A = A^T$*

**Proposition 1.3** *On suppose ici que  $|$  est un anneau commutatif. L'application « transposition » est linéaire :*

$$(A + B)^T = A^T + B^T.$$

$$(\alpha \cdot A)^T = \alpha \cdot A^T.$$

$$(AB)^T = B^T A^T.$$

Si une matrice carrée  $A$  est inversible, alors sa transposée l'est aussi, et la transposée de l'inverse de  $A$  est égale à l'inverse de sa transposée :

$$(A^{-1})^T = (A^T)^{-1}$$

Si  $A$  désigne une matrice carrée de taille  $n$  et  $B$  sa transposée, alors  $A$  et  $B$  ont la même diagonale principale (et par conséquent la même trace) :

$$\forall i \in \{1, \dots, n\}, \quad b_{i,i} = a_{i,i}.$$

En particulier, toute matrice diagonale est symétrique, c'est-à-dire égale à sa transposée. Plus généralement, deux matrices carrées transposées l'une de l'autre ont même polynôme caractéristique donc mêmes valeurs propres, comptées avec leurs multiplicités (en particulier, non seulement même trace mais aussi même déterminant), et même polynôme minimal. Mieux : sur un corps, elles sont semblables. Cela se montre en remarquant qu'elles ont les mêmes invariants de similitude.

**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*

### Définie positivité

En algèbre linéaire, la notion de matrice définie positive est analogue à celle de nombre réel strictement positif : une matrice définie positive est une matrice positive inversible.

**Définition 1.6** (*Matrice symétrique réelle définie positive*) *Soit  $M$  une matrice symétrique réelle d'ordre  $n$ . Elle est dite définie positive si elle est positive et inversible, autrement dit si elle vérifie l'une des quatre propriétés équivalentes suivantes :*

1. Pour toute matrice colonne non nulle  $x$  à  $n$  éléments réels, on a  $x^T M x > 0$  (autrement dit, la forme quadratique définie par  $M$  est strictement positive pour  $x \neq 0$ ) :

2. Toutes les valeurs propres de  $M$  (qui sont nécessairement réelles) sont strictement positives.

3. La forme bilinéaire symétrique  $\mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$ ,  $(x; y) \rightarrow x^T M y$  est un produit scalaire sur  $\mathbb{R}^n$ .

4. Il existe une matrice  $N \in M_n(\mathbb{R})$  inversible telle que  $M = N^T N$  (autrement dit :  $M$  est congruente à la matrice identité).

### Intérêt des matrices définies positives

Les problèmes de résolution de systèmes linéaires les plus faciles à traiter numériquement sont ceux dont les matrices sont symétriques définies positives.

Nous allons exposer dans ce premier chapitre quelques concepts clés concernant l'optimisation, l'effort définitionnel portera surtout sur la méthode du gradient conjugué, où les questions suivantes seront prises en considération les problèmes proposés comportent-ils des contraintes ?

- les fonctions en jeu sont-elles quadratiques ? sont-elles convexes ?

- les domaines de définition des fonctions sont-ils continus ou discrets ?

Chaque problème possède sa structure spécifique et demande donc d'être traité de manière particulière. Le présent projet ne se focalisera que sur les problèmes d'optimisation sans contraintes et non linéaires.

Ce problème sera formulé comme suit :

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

$$\text{ou } f : \mathbb{R}^n \longrightarrow \mathbb{R}$$

La majorité des méthodes de résolution d'optimisation qu'on étudie par la suite sont de nature itérative, c'est à dire qu'à partir d'un point initial  $x_1$ , elles engendrent une suite infinie  $x_1, x_2, \dots, x_k, \dots$  dont on espère qu'elle converge vers la solution optimale.

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

$$x_{k+1} = x_k + \alpha_k d_k, \quad (1.2)$$

où  $d_k$  détermine la direction de déplacement à partir du point  $x_k$  et  $t_k$  est solution optimale du problème d'optimisation unidimensionnel suivant  $\min_{t>0} f(x_k + \alpha d_k)$ , c'est à dire que  $\alpha_k$  vérifié :

$$f(x_k + \alpha_k d_k) \leq f(x_k + \alpha d_k), \forall \alpha > 0 \quad (1.3)$$

## 1.2 Rappel de quelques définitions

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

### 1.2.1 Convergence des algorithmes.

**Définition 1.7** *Un algorithme de résolution est un procédé qui permet, à partir d'une point initial  $x_1$ , d'engendrer la suite  $x_1, x_2, \dots, x_k, \dots$*

Un algorithme est parfaitement défini par la donnée de l'application  $A$  qui associée à  $x_k$  le point  $x_{k+1} \in A(x_k)$ . Ceci permettra de confondre un algorithme et l'application  $A$  qui lui est associée.

**Définition 1.8** *Nous dirons qu'un algorithme décrit par une application multivoque  $A$ , est globalement convergent (ou encore : possède la propriété de convergence globale) si, quelque 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 (ou solution optimale).*

## 1.2.2 Vitesse de convergence

Il est bien entendu très important de garantir la convergence d'un algorithme sous certaines hypothèses, mais la vitesse de convergence et la complexité sont également des facteurs à prendre en compte lors de la conception ou de l'utilisation d'un algorithme ; en effet, on a tout intérêt à ce que la méthode choisie soit à la fois rapide, précise et stable. Pour cela, on introduit les notions de vitesse (ou taux) de convergence qui mesurent l'évolution de l'erreur commise  $\|x_k - x_*\|$ .

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

◆ Si

$$\limsup \frac{\|x_{k+1} - x_*\|}{\|x_k - x_*\|} = \tau < 1,$$

On dit que  $\{x_k\}_{k \in \mathbb{N}}$  converge vers  $x_*$  linéairement avec le taux  $\tau$ .

◆ Si

$$\frac{\|x_{k+1} - x_*\|}{\|x_k - x_*\|} \longrightarrow 0 \text{ quand } k \longrightarrow \infty,$$

on dit que la convergence est super linéaire.

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

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

on dit que la convergence est super linéaire d'ordre  $p$ .

En particulier si

$$\limsup_{k \rightarrow \infty} \frac{\|x^{k+1} - x_*\|}{\|x^k - x_*\|^2} < +\infty,$$

on dit que la convergence est quadratique super linéaire d'ordre 2

## 1.2.3 Fonctions convexes

**Ensemble convexe** .

**Définition 1.10** Soit  $L$  l'ensemble  $C \subseteq \mathbb{R}^n$ .  $C$  est convexe si seulement si  $\forall x, y \in C^2, \forall \lambda \in [0, 1], \lambda x + (1 - \lambda)y \in C$

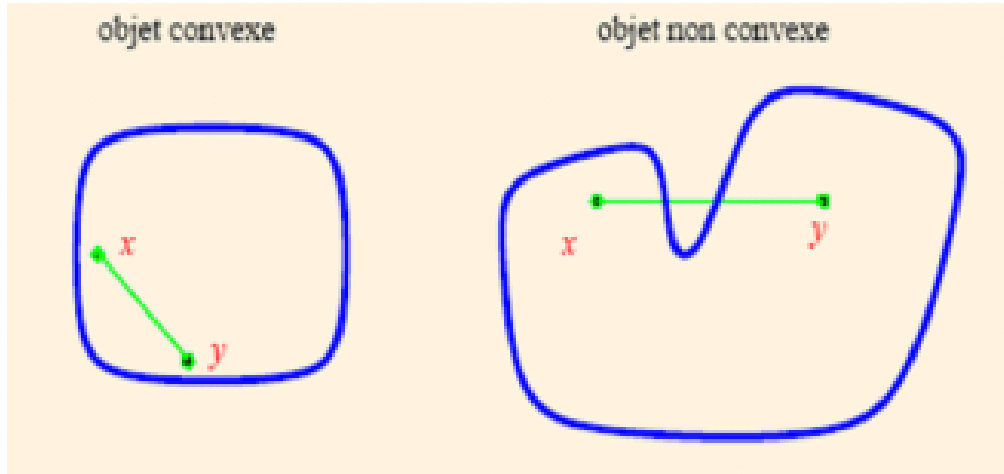


FIG. 1.1 – Objet convexe et non convexe

Objet convexe et non convexe

### Fonction convexe

**Définition 1.11** Soit  $C \subseteq \mathbb{R}^n$  un ensemble convexe non vide. Une fonction  $f : C \rightarrow \mathbb{R}$  est convexe si seulement si

$$\forall x, y \in C^2, \forall \lambda \in [0, 1], \quad f(\lambda x + (1 - \lambda)y) \leq \lambda f(x) + (1 - \lambda)f(y)$$

Une fonction  $f$  est concave si  $-f$  est convexe. On dira que  $f$  est strictement convexe dans  $C$  si seulement si

$$\forall x, y \in C^2, x \neq y, \forall \lambda \in [0, 1], \quad f(\lambda x + (1 - \lambda)y) < \lambda f(x) + (1 - \lambda)f(y)$$

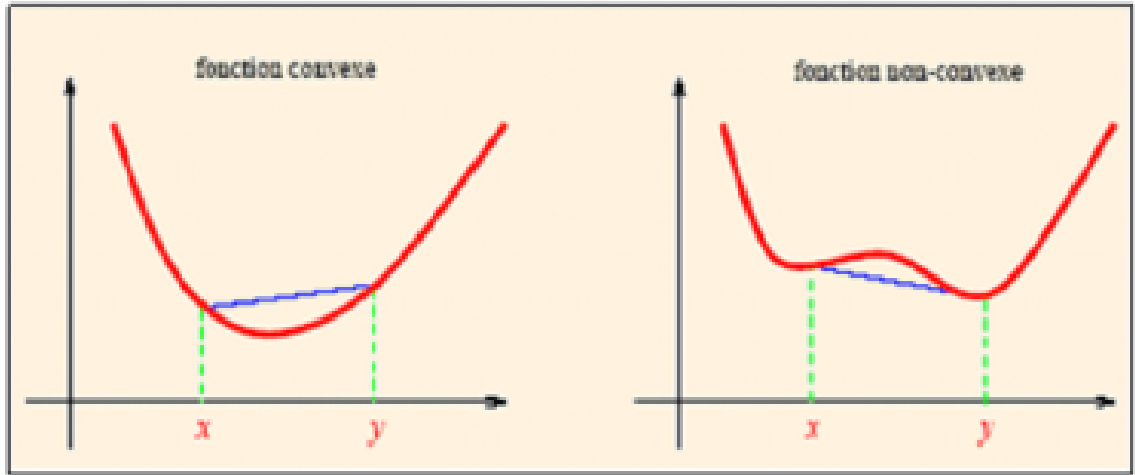


FIG. 1.2 – Fonction convexe et non convexe

#### 1.2.4 Fonctions convexes différentiable

**Définition 1.12 (Fonction convexe différentiable),**

Soit  $C \subseteq \mathbb{R}^n$  non vide et  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  et  $\hat{x} \in \text{int}(C)$   $f$  est dite différentiable au point  $\hat{x}$  s'il existe un vecteur  $A \in \mathbb{R}^n$  et une fonction  $t : \mathbb{R}^n \rightarrow \mathbb{R}$  telles que

$$f(x) = f(\hat{x}) + A(x - \hat{x}) + \|x - \hat{x}\| t(\hat{x}, x - \hat{x}),$$

ou  $t(\hat{x}, x - \hat{x}) \xrightarrow{x \rightarrow \hat{x}} 0$  On peut noter le vecteur  $A$  comme suit

$$A = \nabla f(\hat{x})^T = \left( \frac{\partial f(\hat{x})}{\partial x_1}, \dots, \frac{\partial f(\hat{x})}{\partial x_n} \right)_{(x)} \quad (1.4)$$

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

**Exemple 1.2**  $f(\hat{x}) = x_1^4 + x_2^4 - 4x_1x_2$

Gradient :  $\nabla f(\hat{x})^T = (4x_1^3 - 4x_2, -4x_1 + 4x_2^3)$

Valeur du gradient en  $\hat{x} = (1, -1)$  .  $\nabla f(\hat{x})^T = (8, -8)$

**Définition 1.13 (Fonction convexe deux fois différentiable).**

Soit  $C \subseteq \mathbb{R}^n$  non vide et  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  est dite deux fois différentiable au point  $\hat{x} \in \text{int}(C)$  s'il existe un vecteur  $\nabla f(\hat{x})$  et une matrice symétrique

$H(\hat{x})$  d'ordre  $(n, n)$  appelée matrice hessienne, et une fonction  $\alpha : \mathbb{R}^n \rightarrow \mathbb{R}$  telles que

$$\forall x \in C : f(x) = f(\hat{x}) + \nabla f(\hat{x})^T(x - \hat{x}) + \frac{1}{2}(x - \hat{x})^T H(\hat{x})(x - \hat{x}) + \|x - \hat{x}\|^2 t(\hat{x}, x - \hat{x}), \quad (1.5)$$

ou  $t(\hat{x}, x - \hat{x}) \xrightarrow{x \rightarrow \hat{x}} 0$ . On peut écrire la matrice hessienne  $H(\hat{x})$  comme suit :

$$H(\hat{x}) = \begin{bmatrix} \frac{\partial^2 f(\hat{x})}{\partial x_1 \partial x_1} & \frac{\partial^2 f(\hat{x})}{\partial x_1 \partial x_2} & \dots & \dots & \frac{\partial^2 f(\hat{x})}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f(\hat{x})}{\partial x_2 \partial x_1} & \frac{\partial^2 f(\hat{x})}{\partial x_2 \partial x_2} & \dots & \dots & \frac{\partial^2 f(\hat{x})}{\partial x_2 \partial x_n} \\ \dots & \dots & \dots & \dots & \dots \\ \frac{\partial^2 f(\hat{x})}{\partial x_n \partial x_1} & \frac{\partial^2 f(\hat{x})}{\partial x_n \partial x_2} & \dots & \dots & \frac{\partial^2 f(\hat{x})}{\partial x_n \partial x_n} \end{bmatrix}$$

**Exemple 1.3** Fonction  $f(\hat{x}) = x_1^4 + x_2^4 - 4x_1x_2$

Gradient :  $\nabla f(\hat{x})^T = (4x_1^3 - 4x_2, -4x_1 + 4x_2^3)$

Matrice Hessienne  $H(\hat{x}) = \begin{pmatrix} 12x_1^2 & -4 \\ -4 & 12x_2^2 \end{pmatrix}$

**Définition 1.14** - on dit que  $\hat{x}$  est un point stationnaire de  $f$  si  $\nabla f(\hat{x}) = 0$   
- soit  $M_n$  l'anneau des matrices carrées  $n \times n$  dans  $\mathbb{R}$  Une matrice  $H \in M_n$

◆ La matrice  $H(\hat{x})$  est dite semi-définie positive si

$$\forall x \in \mathbb{R}^n : x^T H(\hat{x})x \geq 0$$

◆ La matrice  $H(\hat{x})$  est dite définie positive si

$$\forall x \in \mathbb{R}^n, x \neq 0 : x^T H(\hat{x})x > 0$$

**Théorème 1.4** soit  $H(\hat{x})$  la matrice hessienne, on note par  $\{\sigma_i\}_{i=1 \dots n}$  ses valeurs propres (réelles) on a les équivalences suivantes :

- ◆  $H(\hat{x}) \geq 0 \iff \sigma_i \geq 0$
- ◆  $H(\hat{x}) > 0 \iff \sigma_i > 0$

## 1.3 Conditions d'optimalité des problèmes de minimisation sans contraintes :

Considérons le problème d'optimisation sans contraintes (P)

$$(P) \quad \min_{x \in \mathbb{R}^n} f(x)$$

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

### 1.3.1 Minima locaux et globaux

**Définition 1.15** 1)  $\hat{x} \in \mathbb{R}^n$  est un minimum local de (P) si et seulement si il existe un voisinage  $V_\varepsilon(\hat{x})$  tel que

$$f(\hat{x}) \leq f(x) : \forall x \in V_\varepsilon(\hat{x}) \tag{1.6}$$

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

$$f(\hat{x}) < f(x) : \forall x \in V_\varepsilon(\hat{x}), x \neq \hat{x} \tag{1.7}$$

3)  $\hat{x} \in \mathbb{R}^n$  est un minimum global de (P) si et seulement si

$$f(\hat{x}) \leq f(x) : \forall x \in \mathbb{R}^n \tag{1.8}$$

**Remarque 1.1** Dans le cas d'une fonction objectif convexe, il n'y a pas de distinction entre minimum local et global : tout minimum local est également global, comme l'établit le théorème suivant.

**Théorème 1.5** . Soit  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  une fonction convexe définie sur l'ensemble convexe  $X$ . Alors, tout minimum local de  $f$  est également un minimum global. Si  $f$  est strictement convexe, alors il existe au plus un minimum global de  $f$

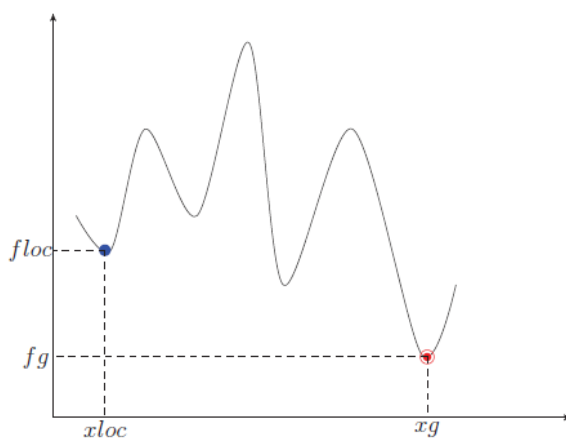


FIG. 1.1 – Optimum global et optimum local

## 1.4 Direction de descente

**Principes généraux :** Considérons le problème d'optimisation sans contraintes  $(P) : \min_{x \in \mathbb{R}^n} f(x)$ ,

ou  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  est supposé régulière.

On note aussi  $\nabla f(x)$  et  $\nabla^2 f(x)$  le gradient et le hessien de  $f$  en  $x$  pour le produit scalaire euclidien sur  $\mathbb{R}^n$ .

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

**Définition 1.16** Soit  $d$  un vecteur non nul de  $\mathbb{R}^n$ . On dit que  $d$  est une direction de descente de  $f$  en  $x \in \mathbb{R}^n$  si

$$\nabla f(x)^T d < 0. \quad (1.9)$$

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

$$\theta = \arccos \frac{-\nabla f(x)^T d}{\|\nabla f(x)\| \|d\|} \in \left[0, \frac{\pi}{2}\right]. \quad (1.10)$$

L'ensemble des directions de descente de  $f$  en  $x$  est le demi espace suivant  $\{d \in \mathbb{R}^n : \nabla f(x)^T d < 0\}$

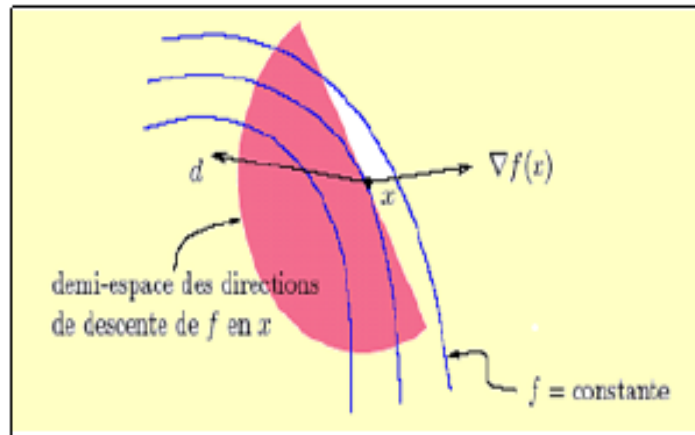
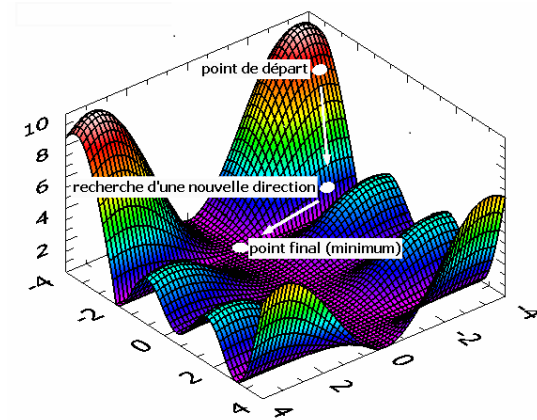


FIG. 1.3 – Demi-espace des directions de descente  $d$  de  $f$  en  $x$ .

On voit que si  $d$  est une direction de descente,  $f(x + \alpha d) < f(x)$ , pour tout  $\alpha > 0$  suffisamment petit, et donc que  $f$  décroît strictement dans la direction  $d$ . De telles directions sont intéressantes en optimisation car, pour faire décroître  $f$ , il suffit de faire un déplacement le long de  $d$ . Les méthodes à directions de descentes utilisent cette idée pour minimiser une fonction. Elles construisent la suite des itérés  $\{x_k\}_{k \geq 1}$ , approchant une solution  $x_k$  du problème (1.1) par la récurrence :

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



Où  $\alpha_k$  est appelé le pas et  $d_k$  la direction de descente de  $f$  en  $x_k$ . Pour définir une direction de descente il faut donc spécifier deux choses :

- ◆ Dire comment la direction  $d_k$  est calculée, la manière de procéder donne le nom à l'algorithme ;
- ◆ Dire comment on détermine le pas  $\alpha_k$ , c'est ce que l'on appelle la recherche linéaire.

**Remarque 1.2** Par la suite l'angle  $\theta_k$  entre la direction  $d_k$  et  $-g_k$  jouera un rôle important dans le processus de la convergence. Pour  $\theta_k$  on a la relation classique suivante :  $-g_k^T d_k = \|g_k\| \|d_k\| \cos \theta_k$

**Définition 1.17** Soit  $f : x \subseteq \mathbb{R}^n \rightarrow \mathbb{R}$ ,  $\hat{x} \in \mathbb{R}^n$ ,  $d \in \mathbb{R}^n$  est une direction de descente au point  $\hat{x}$  si et seulement si il existe  $\delta > 0$  tel que

$$f(\hat{x} + \alpha d) < f(\hat{x}) : \quad \forall \alpha \in ]0, \delta[, \quad (1.11)$$

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

**Théorème 1.6** Soit  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  différentiable au point  $\hat{x} \in \mathbb{R}^n$  et  $d \in \mathbb{R}^n$  une direction vérifiant la condition suivante :

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

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

**Proof.**  $f$  est différentiable au point  $\hat{x}$ . Donc

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

avec  $\alpha(\hat{x}, d) \xrightarrow{\alpha \rightarrow 0} 0$  ceci implique que

$$f'(\hat{x}, d) = \lim_{\alpha \rightarrow 0} \frac{f(\hat{x} + \alpha d) - f(\hat{x})}{\alpha} = \nabla f(\hat{x})^T \cdot d < 0. \quad (1.12)$$

La limite étant strictement négative, alors il existe un voisinage de zéro  $V(0)$  tel que

$$\frac{f(\hat{x} + \alpha d) - f(\hat{x})}{\alpha} < 0, \quad \forall \alpha \in V(0). \quad (1.13)$$

La relation (1.11) est particulièrement vraie pour tout  $\alpha \in ]-\delta, +\delta[$ . On obtient le résultat cherché en multipliant la relation (1.11) par  $\alpha > 0$  ■

### 1.4.1 Schéma général des algorithmes d'optimisation sans contraintes

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 :

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

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

$$f(x_{k+1}) = f(x_k + \alpha_k d_k) < f(x_k).$$

Un bon choix de  $d_k$  et  $\alpha_k$  permet ainsi de construire une multitude d'algorithme d'optimisation.

**Exemple de choix de direction 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. Dans le cas  $d_k = -(H(x_k))^{-1} \cdot \nabla f(x_k)$ , on obtient la méthode de Newton, où la matrice hessienne  $H(x_k)$  est définie positive.

**Exemple de choix de pas  $\alpha_k$  :**

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

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

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

### 1.4.2 Conditions nécessaires d'optimalité

Etant donné un vecteur  $\hat{x}$ , nous souhaiterions être capables de déterminer si ce vecteur est un minimum local ou global de  $(P)$ . La propriété de différentiabilité de  $f$  fournit une première manière de caractériser une solution optimale. Énonçons tout d'abord un théorème, pour établir une première condition nécessaire d'optimalité.

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

**Théorème 1.7** *Soit  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  différentiable au point  $\hat{x} \in \mathbb{R}^n$ . Si  $\hat{x}$  est un minimum local de  $(P)$ , alors  $\nabla f(\hat{x}) = 0$ .*

**Proof.** C'est une conséquence directe du théorème (1.2). En effet, supposons que  $\nabla f(\hat{x}) \neq 0$ . Puisque la direction  $d = -\nabla f(\hat{x})$  est une direction de descente, alors il existe  $\delta > 0$  tel que :

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

Ceci est contradiction avec le fait que  $\hat{x}$  est une solution optimale locale de  $(P)$ . ■

**Remarque 1.3** *si  $f$  est convexe, la condition nécessaire du premier ordre est également suffisante pour que  $\hat{x}$  soit un minimum global. Dans le cas où  $f$  est deux fois différentiable, une autre condition nécessaire est donnée par le théorème (1.4). Elle est appelée condition nécessaire du second ordre car elle fait intervenir la matrice hessienne de  $f$  (que nous noterons  $\nabla^2 f(x)$ , dont les éléments sont ses secondes dérivées partielles).*

**Exemple 1.8**  $f(\hat{x}) = x_1^4 + x_2^4 - 4x_1x_2$

Gradient :  $\nabla f(\hat{x})^T = (4x_1^3 - 4x_2, -4x_1 + 4x_2^3)$  Points stationnaires vérifient  $\nabla f(\hat{x}) = 0$  Trois solutions  $x^{(1)} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$ ,  $x^{(2)} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$ ,  $x^{(3)} = \begin{pmatrix} -1 \\ -1 \end{pmatrix}$

**Remarque 1.4**  $x^{(2)}$  et  $x^{(3)}$  sont des minimum alors que  $x^{(1)}$  ne l'est pas .

**Remarque 1.5** *tout point stationnaire n'est pas forcément un minimum*

*Nécessite d'une autre condition comment s'assurer qu'un point stationnaire est un minimum*

### Conditions nécessaires d'optimalité du second ordre

**Théorème 1.9** Soit  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  deux fois différentiable au point  $\hat{x} \in \mathbb{R}^n$ . Si  $\hat{x}$  est un minimum local de  $(P)$ , alors  $\nabla f(\hat{x}) = 0$  et  $H(\hat{x})$  est semi définie positive.

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

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

Ceci implique

$$\frac{f(\hat{x} + \alpha x) - f(\hat{x})}{\alpha^2} = \frac{1}{2}x^T H(\hat{x})x + \|x\|^2 \alpha(\hat{x}, \alpha x). \quad (1.14)$$

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

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

Si on prend en considération (1.13) et on passe à la limite quand  $\alpha \rightarrow 0$ ,  $\alpha \neq 0$ , on obtient :

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

■

### 1.4.3 Conditions suffisantes d'optimalité

Les conditions données précédemment sont nécessaires (si  $f$  n'est pas convexe), c'est-à-dire qu'elles doivent être satisfaites pour tout minimum local ; cependant, tout vecteur vérifiant ces conditions n'est pas nécessairement un minimum local. Le théorème (1.5) établit une condition suffisante pour qu'un vecteur soit un minimum local, si  $f$  est deux fois différentiable.

#### Condition suffisante d'optimalité du second ordre

**Théorème 1.10** Soient  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  deux fois différentiable au point  $\hat{x} \in \mathbb{R}^n$ . Si  $\nabla f(\hat{x}) = 0$  et  $H(\hat{x})$  est définie positive, alors  $\hat{x}$  est un minimum local strict de  $(P)$ .

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

$$f(x) = f(\hat{x}) + \frac{1}{2}(x-\hat{x})^t H(\hat{x})(x-\hat{x}) + \|x - \hat{x}\|^2 t(\hat{x}, x-\hat{x}), \quad \alpha(\hat{x}, (x-\hat{x})) \rightarrow 0, \quad (\nabla f(\hat{x}) = 0) \quad (1.15)$$

Supposons que  $\hat{x}$  n'est pas un optimum local strict. Alors il existe une suite  $\{x_k\}_{k \in \mathbb{N}^*}$  telle que  $x_k \neq \hat{x} : \forall k$  et

$$x_k \neq \hat{x} : \forall k, \quad x_k \xrightarrow[k \rightarrow \infty]{} \hat{x} \quad \text{et} \quad f(x_k) \leq f(\hat{x}). \quad (1.16)$$

Dans (1.14) prenons  $x = x_k$  divisons le tout par  $\|x_k - \hat{x}\|^2$  et notons  $d_k = \frac{(x_k - \hat{x})}{\|x_k - \hat{x}\|}$ , on obtient

$$\frac{f(x_k) - f(\hat{x})}{\|x_k - \hat{x}\|^2} = \frac{1}{2} d_k^T H(\hat{x}) d_k + \alpha(\hat{x}, (x_k - \hat{x})), \quad \alpha(\hat{x}, (x_k - \hat{x})) \xrightarrow[k \rightarrow \infty]{} 0. \quad (1.17)$$

(1.15) et (1.16) impliquent

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

D'autre part la suite  $\{d_k\}_{k \in \mathbb{N}^*}$  est bornée ( $\|d_k\| = 1, \forall n$ ). Donc il existe une sous suite  $\{d_k\}_{k \in \mathbb{N}_1 \subset \mathbb{N}}$  telle que  $d_k \xrightarrow[k \rightarrow \infty]{} d$ . Finalement lorsque  $k \rightarrow \infty$  on obtient

$$\frac{1}{2} \tilde{d}_k^T H(\hat{x}) \tilde{d}_k \leq 0.$$

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

**Remarque 1.6** *La matrice  $H$  est définie positive ssi toutes ses valeurs propres sont positives*

**Remarque 1.7** *Dans le cas scalaire,  $\hat{x} \in \mathbb{R}$ , cette condition revient à dire qu'à un minimum la dérivée de la fonction est nulle et sa dérivée second est positive*

**Remarque 1.8** *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.*

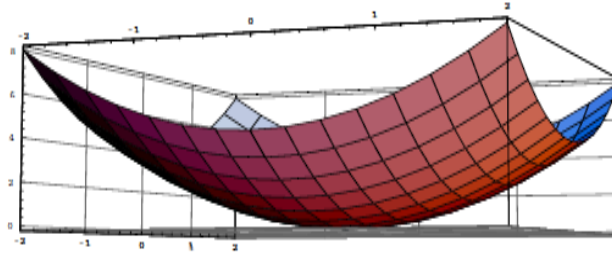


FIG. 1.2 – Graphe de  $z = f(x, y) = x^2 + y^2$  .

**Exemple 1.11** Soit la fonction  $f(x, y) = x^2 + y^2$  .

Les candidats au extrema s'obtiennent en résolvant les système d'équations

$$\frac{\partial f}{\partial x} = 0, \text{ et } \frac{\partial f}{\partial y} = 0$$

$$\frac{\partial f}{\partial x} = 2x = 0 \Rightarrow x = 0$$

$$\frac{\partial f}{\partial y} = 2y = 0 \Rightarrow y = 0$$

ce point est forcément un minimum puisque  $f(x, y) = x^2 + y^2 > 0 \quad \forall x \neq 0, \forall y \neq 0$  comme la figure 3.6

$$\frac{\partial^2 f}{\partial x^2} = 2, \quad \frac{\partial^2 f}{\partial y^2} = 2, \quad \frac{\partial^2 f}{\partial x \partial y} = 0$$

La matrice hessienne est donc définie par :  $H = \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix}$

Ici,  $\Delta_1 = 2$  et  $\Delta_2 = 4$  Par conséquent,  $f$  possède un minimum en  $(0; 0)$



## Chapitre 2

# Recherche linéaire

### 2.0.4 Introduction

Nous venons de présenter les algorithmes les plus fréquemment utilisés pour estimer une direction de déplacement ( $d_k$ ) dans l'espace de la paramétrisation locale du modèle. Mais ces algorithmes ne s'attachent pas à calculer la norme (ou l'amplitude) du déplacement à réaliser dans la direction ( $d_k$ ). C'est ce que proposent de faire les techniques de recherche linéaire (line search en anglais) en tentant de déterminer l'amplitude optimale du déplacement, qui minimise le plus efficacement possible la fonction objectif du problème. Cela permet à l'algorithme de minimisation de converger vers la solution du problème en moins d'itérations. Une illustration d'une étape de la minimisation par recherche linéaire est présentée en figure 2.1. Pour résoudre ce problème, il existe deux types de recherche linéaire : les méthodes exactes et inexactes.

Considérons le problème d'optimisation sans contraintes  $P$ .

$$(P) : \min_{x \in \mathbb{R}^n} f(x), \quad (2.1)$$

$$\text{où } f : \mathbb{R}^n \longrightarrow \mathbb{R}.$$

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

$$x_{k+1} = x_k + \alpha_k d_k \quad (2.2)$$

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

$$\min_{t > 0} f(x_k + \alpha d_k),$$

c'est à dire que  $\alpha_k$  vérifie

$$f(x_k + \alpha_k d_k) \leq f(x_k + \alpha d_k), \forall \alpha > 0,$$

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

$$\alpha \rightarrow \varphi(\alpha) = f(x_k + \alpha d_k). \quad (2.3)$$

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

Dans ce chapitre nous allons décrire les différentes manières de déterminer un pas  $\alpha_k > 0$  le long d'une direction de descente  $d_k$ . C'est ce que l'on appelle faire de la recherche linéaire. Il existe deux grandes classes de méthodes qui s'intéressent à l'optimisation unidimensionnelle :

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

### 2.0.5 Objectifs de la recherche linéaire

Il s'agit de réaliser deux objectifs :

Le premier objectif, faire décroître  $f$  suffisamment, et pour cela on cherche à vérifier l'inégalité :

$$f(x_k + \alpha_k d_k) \leq f(x_k) + \text{“un terme négatif”} \quad (2.4)$$

Le terme négatif joue un rôle-clé dans la convergence de l'algorithme.

Le second objectif, on choisit le pas  $\alpha_k > 0$  d'être trop petit, pour assurer la convergence d'algorithme au point stationnaire .

Le premier objectif n'est en effet pas suffisant car l'inégalité (2.4) 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

Si on prend

$$0 < \alpha_i \leq \frac{\varepsilon}{2^i \|d_k\|},$$

la suite  $\{x_k\}$  générée par (2.4) est de Cauchy, puisque pour  $1 \leq l < k$  on a

$$\|x_k - x_l\| = \left\| \sum_{i=l}^{k-1} \alpha_i d_i \right\| \leq \sum_{i=l}^{k-1} \frac{\varepsilon}{2^i} \rightarrow 0,$$

lorsque  $l \rightarrow \infty$ .

Donc  $\{x_k\}$  converge, disons vers un point  $\bar{x}$ . En prenant  $l = 1$  et  $k \rightarrow \infty$  dans l'estimation ci-dessus, on voit que  $\bar{x} \in \overline{B}(x_1, \epsilon)$  et donc  $\bar{x}$  ne saurait être solution s'il n'y a pas de solution dans  $\overline{B}(x_1, \epsilon)$ . On a donc arbitrairement forcé la convergence de  $\{x_k\}$  en prenant des pas très petits.

Pour simplifier les notations, on définit la restriction de  $f$  à la droite  $\{x_k + \alpha d_k : \alpha \in \mathbb{R}\}$  comme la fonction

$$\phi : \mathbb{R}_+ \rightarrow \mathbb{R}; \quad \alpha \mapsto \phi(\alpha) = f(x_k + \alpha d_k).$$

### 2.0.6 But de la recherche linéaire

Dans le cas non-quadratique les méthodes de descente (2.2), nécessitent la recherche d'une valeur de  $\alpha_k > 0$  optimale ou non, vérifiant :

$$f(x_k + \alpha_k d_k) \leq f(x_k). \tag{2.5}$$

Rappelons que si  $f$  est différentiable, le pas optimal  $\alpha^*$  peut être caractérisé par

$$\phi'(\alpha^*) = 0,$$

$$\phi(\alpha^*) \leq \phi(\alpha), \text{ pour } 0 < \alpha \leq \alpha^*,$$

autrement dit,  $\alpha^*$  est un minimum local de  $\phi$  qui assure de plus la décroissance de  $f$ . En fait, dans la plupart des algorithmes d'optimisation modernes, on ne fait jamais de recherche linéaire exacte, car trouver  $\alpha^*$  signifie qu'il va falloir calculer un grand nombre de fois la fonction  $\phi$ , et cela peut être dissuasif du point de vue du temps de calcul. En pratique, on recherche plutôt une valeur de  $\alpha$  qui assure une décroissance suffisante de  $f$ . Cela conduit à la notion d'intervalle de sécurité.

### 2.0.7 Intervalle de sécurité

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

- ◆ Si  $\alpha < a$  alors  $\alpha$  est considéré trop petit.
- ◆ Si  $b \geq \alpha \geq a$  alors  $\alpha$  est satisfaisant.
- ◆ Si  $\alpha > b$  alors  $\alpha$  est considéré trop grand.

Le problème est de traduire de façon numérique sur  $\varphi$  les trois conditions précédentes, ainsi que de trouver un algorithme permettant de déterminer  $a$  et  $b$ . L'idée est de partir d'un intervalle suffisamment grand pour contenir  $[a, b]$ , et d'appliquer une bonne stratégie pour itérativement réduire cet intervalle.

### 2.0.8 Algorithme de base

**Etape 0 : (initialisation)**

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

**Etape 1 :**

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

Si  $\alpha_k$  est trop petit on prend  $a_{k+1} = \alpha_k$ ,  $b_{k+1} = b$ ,  
et on va à l'étape 2 .

Si  $\alpha_k$  est trop grand on prend  $b_{k+1} = \alpha_k$ ,  $a_{k+1} = a$ ,  
et on va à l'étape 2 .

**Etape 2 :**

Si  $b_{k+1} = 0$  déterminer  $\alpha_{k+1} \in ]a_{k+1}, +\infty[$ .

Si  $b_{k+1} \neq 0$  déterminer  $\alpha_{k+1} \in ]a_{k+1}, b_{k+1}[$ ,  
remplacer  $k$  par  $k + 1$  et aller à l'étape 1.

Il faut maintenant préciser quelles sont les relations sur  $\varphi$  qui va nous permettre de caractériser les valeurs de  $\alpha$  convenables, ainsi que les techniques utilisées pour réduire l'intervalle.

## 2.1 Recherches linéaires exactes

Les techniques de recherche linéaire exacte tentent de résoudre à chaque itération de la minimisation l'équation suivante

$$\alpha_k = \arg \min f(x_k + \alpha_k d_k), \quad \alpha_k > 0 \quad (*)$$

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

$$\min_{\alpha > 0} \varphi(\alpha)$$

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  $\varphi$  qui fait décroître par cette fonction :

$$\alpha_k = \inf \{ \alpha > 0 : \varphi'(\alpha) = 0, \varphi(\alpha) < \varphi(0) \}.$$

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. Elles ne sont utilisées que

dans des cas particuliers, par exemple lorsque  $\varphi$  est quadratique, la solution de la recherche linéaire s'obtient de façon exacte et dans un nombre fini d'itérations.

Lorsque la fonction objectif est quadratique :

$$f(x) = \frac{1}{2}x^T H(x)x - b^T x$$

la solution exacte de ce problème est la suivante [*Nocedal1999, Frandsen1999, Higham2001*] :

$$\alpha_k = -\frac{\nabla f(x_k)^T d_k}{d_k^T H d_k}$$

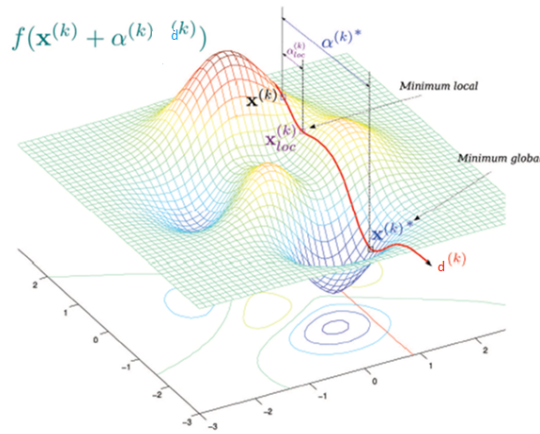


FIG. 2.1 – Recherche linéaire : estimation de la norme  $\blacksquare$  du déplacement . Exemple sur une fonction objectif à deux variables. Deux longueurs sont proposées ( $\blacksquare_{loc}$  et  $\blacksquare^*$ ), qui correspondent à deux solutions :  $x_{loc}$  est un minimum local et  $x^*$  est le minimum global dans la direction du déplacement

**Inconvénients** : Pour une fonction non linéaire arbitraire,

- ▶ il ne peut 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, moins gourmandes en temps de calcul.

Au lieu de demander que  $\alpha_k$  minimise  $\varphi$ , on préfère imposer des conditions moins restrictives, plus facilement vérifiées, qui permettent toutefois 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 facile.

C'est ce que l'on fait avec les règles d'Armijo, de Goldstein et de Wolfe décrites ci-dessous.

Malheureusement la majorité des problèmes de vision par ordinateur possède des fonctions objectives  $F(x)$  non-linéaires et l'équation (\*) devient alors difficile à résoudre. Dans ces conditions, il est beaucoup plus intéressant de

faire appel aux algorithmes de recherche linéaire inexacte.

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

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

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

### Schéma des recherches linéaires inexactes

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

$$f(x_k + \alpha_k d_k) < f(x_k).$$

Le schéma de l'algorithme est donc :

#### Algorithme (Schéma général des recherches linéaires inexactes)

**Etape 0 :** (initialisation)

$a_1 = b_1 = 0$ , choisir  $\alpha_1 > 0$ , poser  $k = 1$  et aller à l'étape 1 :

**Etape 1 :**

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

Si  $\alpha_k$  est trop petit (suivant un certain critère) : nouvel intervalle :

$$[a_{k+1} = \alpha_k, b_{k+1} = b],$$

et aller à l'étape 2.

**Etape 2 :**

Si  $b_{k+1} = 0$  déterminer  $\alpha_{k+1} \in ]a_{k+1}, +\infty[$ .

Si  $a_{k+1} \neq 0$  déterminer  $\alpha_{k+1} \in ]a_{k+1}, b_{k+1}[$ ,

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

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

### 2.2.1 Caractérisation de l'intervalle de sécurité

#### la règle d'Armijo

Une condition naturelle est de demander que  $f$  décroisse autant qu'une portion  $\delta \in ]0, 1[$  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 :

$$f(x_k + \alpha_k d_k) \leq f(x_k) + \delta \alpha_k \nabla f(x_k)^T d_k. \quad (2.6)$$

Elle est de la forme (2.4), car  $\delta$  devra être choisi dans  $]0, 1[$ . On voit bien à la figure (2.1) ce que signifie cette condition. Il faut qu'en  $\alpha_k$ , la fonction prenne une valeur plus petite que celle prise par la fonction affine

$$\alpha \rightarrow f(x_k) + \delta \alpha \nabla^T f(x_k) d_k. \quad (2.8)$$

#### Règle d'Armijo

- ◆ Si  $\varphi(\alpha) \leq \varphi(0) + \delta \varphi'(0)\alpha$ , alors  $\alpha$  convient.
  - ◆ Si  $\varphi(\alpha) > \varphi(0) + \delta \varphi'(0)\alpha$ , alors  $\alpha$  est trop grand.
- On peut noter que l'on a

$$\varphi(\alpha) = f(x_k + \alpha d_k)$$

$$\varphi(0) = f(x_k),$$

$$\varphi'(0) = \nabla f(x_k)^T d_k.$$

#### **Algorithme 2.1 (Règle d'Armijo)**

##### **Etape 0 : (initialisation)**

$a_1 = b_1 = 0$ , choisir  $\alpha_1 > 0$ ,  $\delta \in ]0, 1[$  poser  $k = 1$  et aller à l'étape 1.

##### **Etape 1 :**

Si  $\varphi(\alpha_k) \leq \varphi(0) + \delta \varphi'(0)\alpha_k$  : STOP ( $\alpha^* = \alpha_k$ ).

Si  $\varphi(\alpha_k) > \varphi(0) + \delta \varphi'(0)\alpha_k$ , alors

$b_{k+1} = b$ ,  $a_{k+1} = \alpha_k$  et aller à l'étape 2.

##### **Etape 2 :**

Si  $b_{k+1} = 0$  déterminer  $\alpha_{k+1} \in ] a_{k+1}, +\infty [$ .

Si  $b_{k+1} \neq 0$  déterminer  $\alpha_{k+1} \in ] a_{k+1}, b_{k+1}[$ ,

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

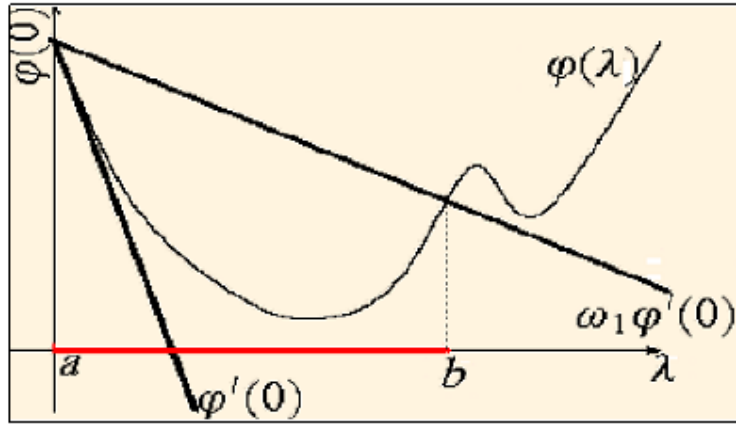


FIG. 2.1 – La règle d'Armijo

### Remarque 2.1

1) En pratique, la constante  $\delta$  est prise très petite, de manière à satisfaire (2.6) le plus facilement possible. Typiquement,  $\delta = 10^{-4}$ . 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.

2) Dans certains algorithmes, il est important de prendre  $\delta < \frac{1}{2}$  pour que le pas  $\alpha_k$  soit accepté lorsque  $x_k$  est proche d'une solution.

3) Il est clair d'après la figure (2.1) que l'inégalité (2.6) est toujours vérifiée si  $\alpha_k > 0$  est suffisamment petit.

On a vu qu'il était dangereux d'accepter des pas trop petits, cela pouvait conduire à une fausse convergence. Il faut donc un mécanisme supplémentaire qui empêche le pas d'être trop petit. On utilise souvent la technique de rebroussement due à *Armijo* [60, 1966] ou celle de *Goldstein*.

### La règle de Goldstein et price

La règle de *Goldstein* remédie à cet inconvénient (le pas  $\alpha_k$  doit être trop petit). Dans celle-ci, en ajoutant une deuxième inégalité à la règle d'Armijo on obtient la règle de *Goldstein*.

$$f(x_k) + \delta \alpha_k \nabla^T f(x_k) d_k \geq f(x_k + \alpha_k d_k) \geq f(x_k) + \sigma \alpha_k \nabla^T f(x_k) d_k, \quad (2.8)$$

où  $\delta$  et  $\sigma$  sont deux constantes vérifiant  $0 < \delta < \sigma < 1$ , cette inégalité qui empêche le pas d'être trop petit.

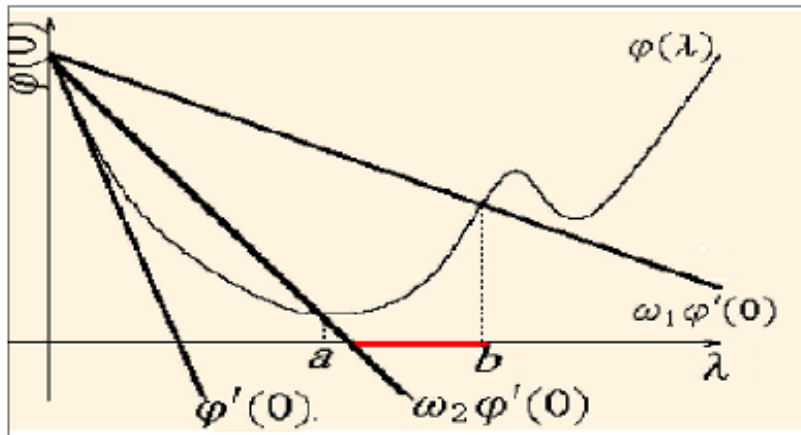


FIG. 2.2 – La règle de Goldstein

### Règle de Goldstien

Si  $\varphi(\alpha) \leq \varphi(0) + \sigma\varphi'(0)\alpha$  : alors  $\alpha$  est trop petit

Si  $\varphi(\alpha) > \varphi(0) + \delta\varphi'(0)\alpha$  , alors  $\alpha$  est trop grand

Si  $\varphi(0) + \delta\varphi'(0)\alpha \geq \varphi(\alpha) \geq \varphi(0) + \sigma\varphi'(0)\alpha$  , alors convient

On peut noter que l'on a

$$\varphi(\alpha) = f(x_k + \alpha d_k)$$

$$\varphi(0) = f(x_k),$$

### Algorithme 2.2 (Règle de Goldstein et price)

#### Etape 0 : (initialisation)

$a_1 = b_1 = 0$ , choisir  $\alpha_1 > 0$ ,  $\delta \in ]0, 1[$ ,  $\sigma \in ]\delta, 1[$  poser  $k = 1$  et aller à l'étape 1.

#### Etape 1 :

Si  $\varphi(0) + \sigma\varphi'(0)\alpha \leq \varphi(\alpha_k) \leq \varphi(0) + \delta\varphi'(0)\alpha_k$  : STOP ( $\alpha^* = \alpha_k$ ).

Si  $\varphi(\alpha_k) > \varphi(0) + \delta\varphi'(0)\alpha_k$  , alors  $b_{k+1} = \alpha_k$ ,  $a_{k+1} = a_k$  et aller à l'étape 2.

Si  $\varphi(\alpha_k) < \varphi(0) + \sigma\varphi'(0)\alpha_k$ , alors  $b_{k+1} = b_k$ ,  $a_{k+1} = \alpha_k$  et aller à l'étape 2.

#### Etape 2 :

Si  $b_{k+1} = 0$  déterminer  $\alpha_{k+1} \in ]a_{k+1}, +\infty[$ .

Si  $b_{k+1} \neq 0$  déterminer  $\alpha_{k+1} \in ]a_{k+1}, b_{k+1}[$ ,

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

**la règle de Wolf :**

les conditions de la règle de Goldstein et price peuvent executer un minimum ce qui peut être un inconvient. La règle de *Wolfe* fait appel au calcul de  $\varphi'(\alpha)$ , elle est donc en théorie plus coûteuse que la règle de *Goldstein*. Cependant dans de nombreuses applications, le calcul du gradient  $\nabla f(x)$  représente un faible coût additionnel en comparaison du coût d'évaluation de  $f(x)$ , c'est pourquoi cette règle est très utilisée.

Nous allons présenter les conditions de *Wolfe faibles* sur  $\alpha > 0$  :

**Règle de wolf faible**

$$f(x_k + \alpha d_k) \leq f(x_k) + \delta \alpha \nabla f(x_k)^T \cdot d_k. \quad (w1)$$

$$\nabla f(x_k + \alpha d_k)^T \cdot d_k \geq \sigma \nabla f(x_k)^T \cdot d_k. \quad (W2)$$

avec  $0 < \delta < \sigma < 1$  .

Pour expliquer ces conditions posons

$$l(\alpha) = \phi(0) + (\sigma \phi'(0))\alpha.$$

**Règle de Wolfe faible**

- ◆ Si  $\phi(\alpha) \leq \phi(0) + \delta \varphi'(0)\alpha$  et  $\varphi'(\alpha) \geq \sigma \varphi'(0)$ , alors  $\alpha$  convient.
- ◆ Si  $\phi(\alpha) > \phi(0) + \delta \varphi'(0)\alpha$ , alors  $\alpha$  est trop grand.
- ◆ Si  $\varphi'(\alpha) < \sigma \varphi'(0)$ , alors  $\alpha$  est trop petit.

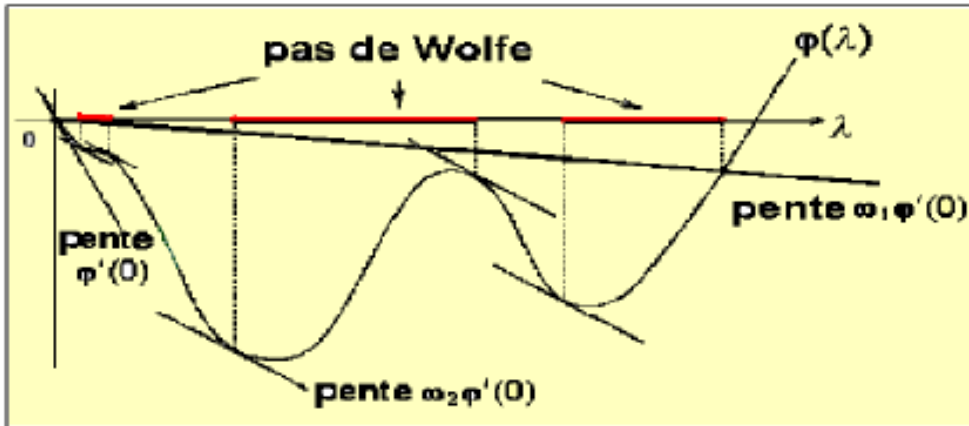


FIG. 2.3 – La règle de Wolfe

**Algorithme 2.3 (Règle de Wolfe)**

**Etape 0 : (initialisation)**

$a_1 = b_1 = 0$ , choisir  $\alpha_1 > 0$ ,  $\delta \in ]0, 1[$ ,  $\sigma \in ]\delta, 1[$ , poser  $k = 1$  et aller à l'étape 1.

**Etape 1 :**

Si  $\phi(\alpha_k) \leq \phi(0) + \delta\phi'(0)\alpha_k$  et  $\phi'(\alpha) \geq \sigma\phi'(0)$  : STOP ( $\alpha^* = \alpha_k$ ).

Si  $\phi(\alpha_k) > \phi(0) + \delta\phi'(0)\alpha_k$ , alors

$b_{k+1} = \alpha_k$ ,  $a_{k+1} = a_k$ , et aller à l'étape 2.

Si  $\phi'(\alpha) < \sigma\phi'(0)$ , alors

$b_{k+1} = b_k$ ,  $a_{k+1} = \alpha_k$  et aller à l'étape 2.

**Etape 2 :**

Si  $b_{k+1} = 0$  déterminer  $\alpha_{k+1} \in ]a_{k+1}, +\infty[$ .

Si  $b_{k+1} \neq 0$  déterminer  $\alpha_{k+1} \in ]a_{k+1}, b_{k+1}[$ .

**Règle de Wolfe Forte** On obtient des contraintes plus fortes si l'on remplace (W2) par

$$|\nabla f(x_k + \alpha d_k)^T \cdot d_k| \leq -\sigma \nabla f(x_k)^T \cdot d_k. \quad (W3)$$

Les (W1) et (W3) sont les conditions de *Wolfe fortes*. La contrainte (W3) entraîne que  $\sigma\phi'(0) \leq \phi'(\alpha) \leq -\sigma\phi'(0)$  c-à-d.  $\phi'(\alpha)$  n'est pas "trop" positif.

1) On voit bien que les conditions de *Wolfe fortes* impliquent les conditions de *Wolfe faibles*, en effet :

$$\begin{aligned} |\nabla^T f(x_k + \alpha_k d_k) d_k| &\leq -\sigma \nabla^T f(x_k) d_k \\ \Leftrightarrow \sigma \nabla^T f(x_k) d_k &\leq \nabla^T f(x_k + \alpha_k d_k) d_k \leq -\sigma \nabla^T f(x_k) d_k \\ \Rightarrow \sigma \nabla^T f(x_k) d_k &\leq \nabla^T f(x_k + \alpha_k d_k) d_k. \end{aligned} \quad (W2)$$

2) L'existence de valeurs  $\alpha$  de vérifiant les conditions de *Wolfe faibles* et *fortes*. est donnée par la Proposition suivante.

**Proposition 2.1** [20]. *Soit  $f \in C^1$  et  $d$  une direction de descente de  $f$  en  $x$ , on suppose que  $f$  est minorée. Alors, si  $0 < \delta < \sigma < 1$ , il existe des intervalles dans  $\mathbb{R}_+$  qui vérifient les conditions de *Wolfe faibles* et *fortes*.*

**La règle de Wolfe relaxée** Proposée par *Dai* et *Yuan* [1996], cette règle consiste à choisir le pas satisfaisant aux conditions :

$$f(x_k + \alpha d_k) \leq f(x_k) + \delta \alpha \nabla^T f(x_k) d_k, \quad (w4)$$

$$\sigma \nabla^T f(x_k) d_k \leq \nabla^T f(x_k + \alpha_k d_k) d_k \leq -\bar{\sigma} \nabla^T f(x_k) d_k, \quad (w5)$$

où  $0 < \delta < \sigma < 1$  et  $\bar{\sigma} > 0$ .

**Remarque 2.2** .

1) On voit bien que les conditions de *Wolfe relaxée* impliquent les conditions de *Wolfe fortes*. Effectivement (w4) est équivalente à (w1), tandis que pour le cas particulier  $\sigma = \bar{\sigma} = \sigma$ , (w5) est équivalente à (w3). En effet :

$$\begin{aligned} \sigma \nabla^T f(x_k) d_k &\leq \nabla^T f(x_k + \alpha_k d_k) d_k \leq -\bar{\sigma} \nabla^T f(x_k) d_k \\ \Rightarrow \sigma \nabla^T f(x_k) d_k &\leq \nabla^T f(x_k + \alpha_k d_k) d_k \leq -\sigma \nabla^T f(x_k) d_k \\ \Rightarrow |\nabla^T f(x_k + \alpha_k d_k) d_k| &\leq -\sigma \nabla^T f(x_k) d_k. \end{aligned} \quad (w3)$$

2) Les conditions de *Wolfe relaxée* impliquent les conditions de *Wolfe faibles*. Effectivement (w4) est équivalente à (w1), tandis que pour le cas particulier  $\sigma = \sigma$  et  $\bar{\sigma} = +\infty$ , (w5) est équivalente à (w2). En effet :

$$\begin{aligned} \sigma \nabla^T f(x_k) d_k &\leq \nabla^T f(x_k + \alpha_k d_k) d_k \leq -\bar{\sigma} \nabla^T f(x_k) d_k \\ \Rightarrow \sigma \nabla^T f(x_k) d_k &\leq \nabla^T f(x_k + \alpha_k d_k) d_k. \end{aligned} \quad (w2)$$

## 2.3 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. 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

$$f(x_{k+1}) \leq f(x_k) - C \|\nabla f(x_k)\|^2 \cos^2 \theta_k, \quad (2.9)$$

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

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

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

**Proposition 2.2** . *Si la suite  $\{x_k\}$  générée par un algorithme d'optimisation vérifie la condition de Zoutendijk (2.1) et si la suite  $\{f(x_k)\}$  est minorée, alors*

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

**Proof.** En sommant les inégalités (2.1), on a

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

La série est donc convergente puisqu'il existe une constante  $C'$  telle que pour tout  $k$ ,  $f(x_k) \geq C'$ . Il ya des propositions précisent les circonstances dans lesquelles la condition de *Zoutendijk* (3.1) est vérifiée avec les règles de la recherche linéaire exacte (*Cauchy, Curry*) et aussi les règles de la recherche linéaire inexacte (*Armijo, Wolfe*).

qui concernant la proposition qui est vérifiée la condition de *Zoutendijk* avec la règle de *Wolfe*. ■

**Proposition 2.3** *Soit  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  une fonction continument différentiable dans un voisinage de  $\Omega = \{x \in \mathbb{R}^n : f(x) \leq f(x_1)\}$ . On considère un algorithme à directions de descente  $d_k$ , qui génère une suite  $\{x_k\}$  en utilisant la recherche linéaire de Wolfe (w1)-(w2).*

Alors il existe une constante  $C > 0$  telle que, pour tout  $k \geq 1$ , la condition de Zoutendijk (3.1) est vérifiée.

**Proof.** Noton  $g_k = \nabla f(x_k)$  et  $g_{k+1} = \nabla f(x_k + \alpha_k d_k)$ . D'après (W2)

$$g_{k+1}^T d_k \geq \sigma g_k^T d_k$$

$$\begin{aligned} \Rightarrow & (g_{k+1} - g_k)^T d_k \geq (\sigma - 1) g_k^T d_k \\ = & -(1 - \sigma) g_k^T d_k = (1 - \sigma) |g_k^T d_k| \\ \Leftrightarrow & (1 - \sigma) |g_k^T d_k| \leq (g_{k+1} - g_k)^T d_k, \end{aligned}$$

et du fait que  $f$  est continuellement différentiable :

$$\begin{aligned} (1 - \sigma) |g_k^T d_k| &= (1 - \sigma) \|g_k\| \|d_k\| \cos \theta_k \\ &\leq \|g_{k+1} - g_k\| \|d_k\| \\ \Rightarrow & (1 - \sigma) \|g_k\| \cos \theta_k \leq L \alpha_k \|d_k\| \\ \Rightarrow & \alpha_k \|d_k\| \leq \frac{(1 - \sigma)}{L} \|g_k\| \cos \theta_k, \end{aligned}$$

en utilisant (w1), on aura :

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

On en déduit (3.1). ■



## Chapitre 3

# La méthode du gradient conjugué

### 3.1 La méthode du gradient conjugué linéaire

En analyse numérique, la méthode du gradient conjugué est un algorithme pour résoudre des systèmes d'équations linéaires dont la matrice est symétrique définie positive. Cette méthode, imaginée en 1950 simultanément par Cornelius Lanczos et Magnus Hestenes, est une méthode itérative qui converge en un nombre fini d'itérations (au plus égal à la dimension du système linéaire)[14]. Toutefois, son grand intérêt pratique du point de vue du temps de calcul vient de ce qu'une initialisation astucieuse (dite « pré conditionnement ») permet d'aboutir en seulement quelques passages à une estimation très proche de la solution exacte, c'est pourquoi en pratique on se borne à un nombre d'itérations bien inférieur au nombre d'inconnues.

### 3.2 Méthodes itératives d'optimisation sans contraintes

Au travers du présent chapitre et du suivant, on s'attachera dorénavant à la description plus spécifique des algorithmes itératifs (ou méthodes itératives) qui ont été implémentés et qui permettent la résolution des problèmes d'optimisation non linéaire.

Il convient de souligner que la plupart des algorithmes d'optimisation sans ou avec contrainte fonctionnent selon un schéma général consistant, à chaque itération, à se rapprocher du minimum par la résolution d'un sous-problème

de minimisation. Nous considérons ici les méthodes permettant de résoudre un problème d'optimisation sans contraintes, soit le problème  $(P)$

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

pour lequel nous allons commencer par décrire les méthodes suivantes :

- ◆ Les méthodes de gradient
- ◆ Les méthodes utilisant des directions conjuguées
- ◆ Les méthodes de Newton et quasi-Newton.

Ces méthodes utilisent des dérivées (et donc la propriété de différentiabilité de  $f$ ) à l'exception des méthodes de directions conjuguées (sauf dans le cas particulier de la méthode du gradient conjugué) basée elle, sur des propriétés plus géométriques.

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.

Après les avoir décrites, nous donnons la définition d'une forme quadratique et *Principe* des méthodes de descente et de gradient.

**Définition 3.1** . Soit  $H$  une matrice symétrique  $n \times n$  et  $b \in \mathbb{R}^n$  On appelle forme quadratique la fonction  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  définie par

$$f(x) = \frac{1}{2}x^T H(x)x - b^T x. \quad (3.1)$$

Lorsque la matrice  $H$  est définie positive (resp. semi-définie positive), on dira que  $f(x)$  est une forme quadratique définie positive (resp. semi-définie positive).

### 3.2.1 Principe des méthodes de descente

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

$$x_{k+1} = x_k + \alpha_k d_k \quad \alpha_k > 0, \text{ tout en assurant la propriété}$$

$$f(x_{k+1}) < f(x_k). \quad (3.2)$$

Le vecteur  $d_k$  est la direction de descente en  $x_k$ . Le scalaire  $\alpha_k$  est appelé le pas de la méthode à l'itération  $k$ . On peut caractériser les directions de descente en  $x_k$  à l'aide du gradient.

### 3.2.2 Principe des méthodes de gradient (méthodes de la plus forte descente)

On cherche à déterminer la direction de descente qui fait décroître  $\varphi(\alpha) = f(x_k + \alpha_k d_k)$  le plus vite possible (au moins localement). Pour cela on va essayer de minimiser la dérivée de  $\varphi(\alpha)$  en 0. On a  $\varphi'(0) = \nabla f(x)^T d$ , et on cherche  $d$  solution du problème

$$\min_{d \in \mathbb{R}^n, \|d\|=1} \phi'(0).$$

La solution est bien sûr

$$d = -\frac{\nabla f(x)}{\|\nabla f(x)\|}.$$

En vertu de l'inégalité de Schwartz. Il y a ensuite de nombreuses façon d'utiliser cette direction de descente. On peut par exemple utiliser un pas fixé a priori  $\alpha_k = \alpha > 0, \forall k$ . On obtient alors la méthode du gradient simple :

$$\begin{cases} d_k = -\nabla f(x_k), \\ x_{k+1} = x_k + \alpha d_k. \end{cases} \quad (3.3)$$

Sous certaines hypothèses de régularité ( $f$  deux fois différentiable) cette méthode converge si  $\alpha$  est choisi assez petit. Ou bien consiste à faire les itérations suivantes

$$\begin{cases} d_k = -\nabla f(x_k), \\ x_{k+1} = x_k + \alpha_k d_k. \end{cases} \quad (3.4)$$

Où  $\alpha_k$  est choisi de manière à ce que

$$f(x_k + \alpha_k d_k) \leq f(x_k + \alpha d_k), \forall \alpha > 0. \quad (3.5)$$

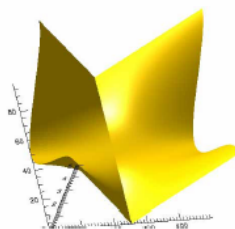
On obtient alors la méthode du gradient à pas optimal, cette méthode possède une propriété intéressante :

## CHAPITRE 3. LA MÉTHODE DU GRADIENT CONJUGUÉ

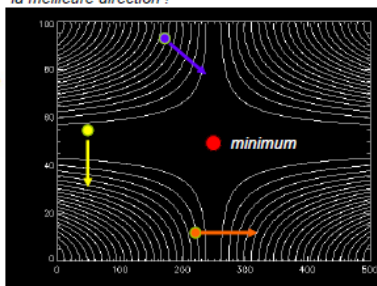
---

Soit à minimiser  $F(x,y)$  (dessin ci-dessous)

Imaginons que nous sommes dans une « vallée » étroite.



La flèche est la direction de - gradient.  
Pour les points jaune et orange le gradient n'est pas  
la meilleure direction !



=> gradient conjugué

**Proposition 3.1** Soit  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  une fonction différentiable. Les directions de descente  $d_k$  générées par la méthode (2.15) et (2.16) vérifient

$$d_{k+1}^T d_k = 0. \quad (3.6)$$

**Proof.** Si on introduit la fonction

$$\varphi(\alpha) = f(x_k + \alpha d_k),$$

on a

$$\varphi'(\alpha) = \nabla f(x_k + \alpha d_k)^T d_k,$$

et puisque  $\varphi$  est dérivable on a nécessairement

$$\varphi'(\alpha) = 0,$$

donc

$$\nabla f(x_k + \alpha_k d_k)^T d_k = \nabla f(x_{k+1})^T d_k = -d_{k+1}^T d_k = 0.$$

■

► Calcul du pas optimal dans le cas quadratique :

On a  $f(x) = \frac{1}{2}x^T Hx - b^T x$  avec  $H > 0$  et on note  $\varphi(\alpha) = f(x_k + \alpha d_k)$ .

Le pas optimal  $\alpha_k$  est caractérisé par

$$\varphi'(\alpha) = 0,$$

on a donc

$$\nabla f(x_k + \alpha_k d_k)^T d_k = (H(x_k + \alpha_k d_k) - b)^T d_k = 0,$$

soit

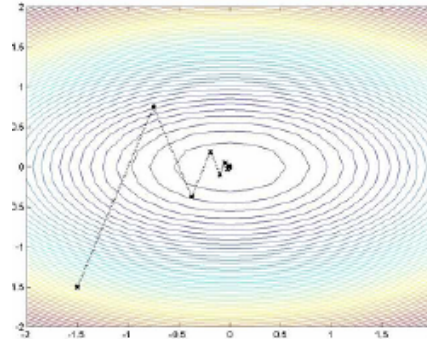
$$(\nabla f(x_k) + \alpha_k H d_k)^T d_k = 0,$$

on obtient donc

$$\alpha_k = -\frac{\nabla f(x_k)^T d_k}{d_k^T H d_k}, \quad (3.7)$$

qui est bien positif car  $d_k$  est une direction de descente et

$$d_k^T H d_k > 0 \text{ (car } H > 0 \text{)}.$$



La méthode du gradient à pas optimal peut donc s'écrire (dans le cas quadratique)

$$\begin{aligned}d_k &= b - Hx_k, \\ \alpha_k &= \frac{d_k^T d_k}{d_k^T H d_k}, \\ x_{k+1} &= x_k + \alpha_k d_k.\end{aligned}$$

### Exemple 3.1

► Méthode du gradient simple dans le cas quadratique

Dans le cas où  $f(x) = \frac{1}{2}x^T H(x)x - b^T x$  la méthode du gradient simple peut s'écrire

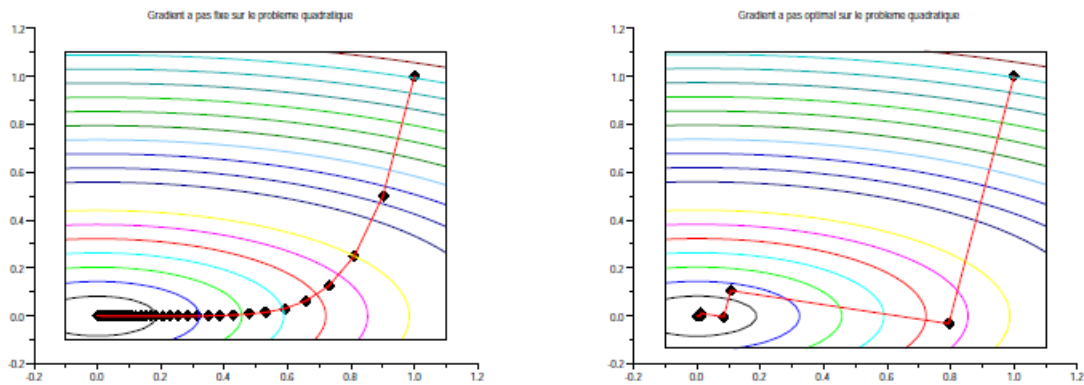
$$\begin{aligned}d_k &= b - Hx_k, \\ x_{k+1} &= x_k + \alpha d_k,\end{aligned}$$

où  $\alpha > 0$  est fixé a priori. Il existe bien sûr des conditions sur pour que la méthode converge.

► Méthode du gradient à pas optimal dans le cas quadratique

Dans le cas où  $f(x) = \frac{1}{2}x^T H(x)x - b^T x$  la méthode du gradient à pas optimal peut s'écrire :

### CHAPITRE 3. LA MÉTHODE DU GRADIENT CONJUGUÉ



*Algorithmes de gradient à pas fixe et à pas optimal.*

$$\begin{aligned}d_k &= b - Hx_k, \\ \alpha_k &= \frac{d_k^T d_k}{d_k^T H d_k}, \\ x_{k+1} &= x_k + \alpha_k d_k.\end{aligned}$$

### Algorithme de la méthode de la plus forte pente

**Etape initiale :**

Choisir un  $\varepsilon > 0$ .

Choisir un point initial  $x_1$ .

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

**Etape principale :**

Si  $\|\nabla f(x)\| < \varepsilon$  stop.

Si non poser  $d_k = -\nabla f(x_k)$  et soit  $\alpha_k$  la solution optimale de la recherche linéaire

$$\min \{f(x_k + \alpha d_k); \alpha > 0\}.$$

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

Remplacer  $k$  par  $k + 1$  et répéter l'étape principale.

### Inconvénients de la méthode de la plus forte pente

► Lenteur de la méthode au voisinage des points stationnaires.

► Cette méthode travaille de façon performante dans les premières étapes de l'algorithme. Malheureusement, dès qu'on s'approche du point stationnaire, La méthode devient très lente. On peut expliquer intuitivement ce phénomène par les considérations suivantes

$$f(x_k + \alpha d_k) = f(x_k) + \alpha \nabla f(x_k)^T d + \alpha \|d\| \alpha(x_k; \alpha d),$$

où  $\alpha(x_k; \alpha d) \rightarrow 0$  quand  $\alpha d \rightarrow 0$ .

Si  $d_k = -\nabla f(x_k)$ , on obtient :  $x_{k+1} = x_k - \alpha \nabla f(x_k)$  et par conséquent

$$f(x_{k+1}) - f(x_k) = \alpha [-\|\nabla f(x_k)\|^2 + \|\nabla f(x_k)\| \alpha(x_k; \alpha \nabla f(x_k))].$$

D'après l'expression précédente, on voit que lorsque  $x_k$  s'approche d'un point stationnaire, et si  $f$  est continument différentiable, alors  $\|\nabla f(x_k)\|$  est proche de zéro. Donc le terme à droite s'approche de zéro, indépendamment de  $\alpha$ , et par conséquent  $f(x_{k+1})$  ne s'éloigne pas beaucoup de  $f(x_k)$  quand on passe du point  $x_k$  au point  $x_{k+1}$ .

### 3.2.3 Les méthodes utilisant des directions conjuguées

#### Description de la méthode

Ces méthodes sont basées sur l'important concept de la conjugaison et ont été développées afin de résoudre le problème quadratique

$$\min_{x \in \mathbb{R}^n} f(x) = \frac{1}{2} x^T A x + b^T x + c$$

Où  $A \in \mathbb{R}^{n \times n}$  est symétrique et définie positive,  $b \in \mathbb{R}^n$  et  $c \in \mathbb{R}$ . Les méthodes de direction conjuguées peuvent résoudre les problèmes de cette forme en au plus  $n$  itérations et, contrairement aux méthodes présentées jusqu'à présent, elles n'utilisent pas de dérivées, sauf dans le cas particulier de la méthode du gradient conjugué. Donnons la définition de la notion de "conjugaison" :

**Définition 3.2** Soient  $A$  une matrice  $n \times n$  symétrique et définie positive et un ensemble de directions non nulles  $\{d_1, d_2, \dots, d_k\}$ . Ces directions sont dites  $A$ -conjuguées si

$$d_i^T A d_j = 0, \quad \forall i, j \text{ tels que } i \neq j. \quad (3.8)$$

on en déduit :

$$\begin{aligned} d_{k+1}^T A d_k = 0 &\implies (-\nabla f(x_{k+1}) + B_{k+1} d_k)^T A d_k = 0 \\ &\implies (-\nabla^T f(x_{k+1}) A d_k + B_{k+1} d_k^T A d_k) = 0 \\ &\implies B_{k+1} = \frac{\nabla^T f(x_{k+1}) A d_k}{d_k^T A d_k} = \frac{g_{k+1}^T A d_k}{d_k^T A d_k} \end{aligned}$$

**Propriété 2.1.** Si  $d_1, \dots, d_k$  sont  $A$ -conjuguées, alors elles sont linéairement indépendantes.

**Propriété 2.2.** Comme des directions  $A$ -conjuguées sont linéairement indépendantes, alors l'espace vectoriel engendré par un ensemble de  $n$  directions  $A$ -conjuguées est de dimension  $n$ .

Étant donné un ensemble de  $n$  directions  $A$ -conjuguées  $d_0, d_1, \dots, d_{n-1}$ , la méthode de directions conjuguées correspondante est donnée par

$$x_{k+1} = x_k + \alpha_k d_k, \quad k = 0, \dots, n-1,$$

où  $x_0$  est un vecteur de départ choisi arbitrairement et où les  $\alpha_k$  sont obtenus par minimisation monodimensionnelle le long de  $d_k$ .

Le principal résultat concernant les méthodes utilisant des directions conjuguées est qu'à chaque itération  $k$ , la méthode minimise  $f$  sur le sous-espace généré par les  $k$  premières directions  $A$ -conjuguées utilisées par l'algorithme. A la  $n$ ème itération au plus tard, ce sous-espace inclura alors le minimum global de  $f$ , grâce à la propriété d'indépendance linéaire des directions  $A$ -conjuguées qui assure qu'à l'itération  $n$ , l'espace vectorielle générée par les  $n$  directions  $A$ -conjuguées ne sera autre que  $\mathbb{R}^n$ .

Remarque 2.5. la notion de conjugaison n'a pas de sens dans le cas non quadratique.

### 3.3 La methode du gradient conjugué non linéaire

Afin de minimiser une fonction non linéaire, la méthode de gradient conjugué doit être modifiée. D'abord nous faisons une recherche linéaire afin de calculer le pas  $\alpha_k$  qui minimise la fonction non-linéaire  $f$  le long de  $d_k$ . Deuxièmement, le résidu  $r$  doit être remplacé par le gradient de  $f$ . Nous obtenons ainsi un algorithme très efficace pour l'optimisation non-linéaire [74]. Si  $f$  est strictement convexe et si  $\alpha_k$  est le minimiseur exacte, alors cet algorithme est exactement le gradient conjugué linéaire [75]. Le paramètre  $\alpha_k$  doit vérifier certaines conditions afin d'assurer que la direction  $d_{k+1}$  soit une direction de descente. Si la recherche linéaire est exacte, nous voyons que :  $\nabla f(x_k)^T \cdot d_k < 0$

Mais si la recherche n'est pas exacte (parce que la recherche exacte est trop coûteuse), il faut imposer d'autres conditions .

L'idée de la méthode est de construire itérativement des directions  $d_0, \dots, d_k$  mutuellement conjuguées. A chaque étape  $x_k$  la direction  $d_k$  est obtenue comme combinaison linéaire du gradient en  $x_k$  et de la direction précédente  $d_{k-1}$ , les coefficients étant choisis de telle manière que  $d_k$  soit conjuguée avec toutes les directions précédentes. On s'intéresse ici à la minimisation d'une fonction  $f : \mathbb{R}^n \rightarrow \mathbb{R}$ , non nécessairement quadratique :

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

et on cherche à étendre la méthode du gradient conjugué à ce problème. Il y a plusieurs manières de le faire et peu de critères permettant de dire laquelle est la meilleure. Une extension possible consiste simplement a reprendre les

formules utilisées dans le cas quadratique. On se propose donc d'étudier les méthodes où la direction  $d_k$  est définie par la formule de récurrence suivante ( $\beta_k \in \mathbb{R}$ )

$$d_k = \begin{cases} -g_1 & \text{si } k = 1, \\ -g_k + \beta_k d_{k-1} \dots\dots\dots & \text{si } k \geq 2, \end{cases} \quad (3.9)$$

où  $g_k = \nabla f(x_k)$ ,  $\beta_k \in \mathbb{R}$  et  $\{x_k\}$  est générée par la formule :

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

le pas  $\alpha_k \in \mathbb{R}$  étant déterminé par une recherche linéaire.

Ces méthodes sont des extensions de la méthode du gradient conjuguée si  $\beta_k$  prend l'une des valeurs

$$\beta_k^{HS} = \frac{g_k^T (g_k - g_{k-1})}{d_{k-1}^T (g_k - g_{k-1})}, \quad \text{Hestenes-Stiefel [14], 1952} \quad (3.10)$$

$$\beta_k^{FR} = \frac{g_k^T g_k}{g_{k-1}^T g_{k-1}}, \quad \text{Fletcher Reeves [11], 1964} \quad (3.11)$$

$$\beta_k^{PRP} = \frac{g_k^T (g_k - g_{k-1})}{\|g_{k-1}\|^2} \quad \text{Polak-Ribière-Polyak [18, 19], 1969} \quad (3.12)$$

$$\beta_k^{CD} = -\frac{g_k^T g_k}{d_{k-1}^T g_{k-1}}, \quad \text{descente conjugué. [10], 1987} \quad (3.13)$$

$$\beta_k^{LS} = -\frac{g_k^T (g_k - g_{k-1})}{d_{k-1}^T g_{k-1}}, \quad \text{Liu-Storey [15], 1991} \quad (3.14)$$

$$\beta_k^{DY} = \frac{g_k^T g_k}{(g_k - g_{k-1})^T d_{k-1}}, \quad \text{Dai-Yuan [7], 1999} \quad (3.15)$$

où  $y_{k-1} = g_k - g_{k-1}$ .

Pour faciliter la présentation nous appelons les méthodes qui correspondent à (3.12) - (3.17) la méthode HS, la méthode FR, la méthode PRP, la méthode CD, et la méthode LS, et la méthode DY, respectivement.

Dans le cas où  $f$  est une fonction quadratique strictement convexe avec une recherche linéaire exacte toutes ces variantes de  $\beta_k$  ont la même valeur :

$$\beta_k^{PRP} = \beta_k^{FR} = \beta_k^{CD} = \beta_k^{DY} = \beta_k^{LS} = \beta_k^{HS}.$$

Si  $f$  est quelconque, il n'en est plus de même et on parle respectivement de *méthode de Polak-Ribière-Ployak*  $n$  ([35, 1969])-( [36, 1969] ) *méthode de Fletcher-Reeves* ([18, 1964 ]), *méthode de la descente conjuguée* ([55, 1987]) ou de *méthode de Dai-Yuan* ([54, 1999]) selon que l'on utilise  $\beta_k^{PRP}$ ,  $\beta_k^{FR}$ ,  $\beta_k^{CD}$  ou  $\beta_k^{DY}$  à la place de  $\beta_k$  dans (3.11).

Pour que les méthodes ainsi définies soient utilisables, il faut répondre aux deux questions suivantes.

► Les directions  $d_k$  définies par (3.11) sont-elles des directions de descente de  $f$  ?

► Les méthodes ainsi définies sont-elles convergentes ?

En ce qui concerne la première question remarquons que, quel que soit  $\beta_k \in \mathbb{R}$ ,  $d_k$  est une direction de descente si on fait de la recherche linéaire exacte, c'est-à-dire si le pas  $\alpha_{k-1}$  est un point stationnaire de  $\alpha \rightarrow f(x_{k-1} + \alpha d_{k-1})$ . En effet, dans ce cas  $g_k^T d_{k-1} = 0$  et on trouve lorsque  $g_k \neq 0$  :

$$\begin{aligned} d_k^T g_k &= (-g_k + \beta_k d_{k-1})^T g_k \\ &= -\|g_k\|^2 + \beta_k d_{k-1}^T g_k = -\|g_k\|^2 < 0 \end{aligned}$$

Cependant, il est fortement déconseillé de faire de la recherche linéaire exacte lorsque  $f$  n'est pas quadratique : le coût de détermination de  $k$  est excessif.

L'efficacité de la méthode du gradient conjugué repose essentiellement sur deux points :

- La recherche linéaire détermination du pas optimal) doit être exacte,
- Les relations de conjugaison doivent être précises.

La recherche du pas optimal doit être réalisée à l'aide d'un algorithme spécifique, puisque  $f$  est quelconque. Par contre la notion de conjugaison n'a pas de sens dans le cas non quadratique.

L'étude des propriétés de convergence de quelques méthodes du gradient conjugué non linéaire est l'objectif du troisième chapitre.

### 3.4 la convergence de quelques méthodes du gradient conjugué non linéaire avec la recherche linéaire inexacte

Notre problème consiste de minimiser une fonction  $f$  de  $n$  variables de valeurs réelles

$$(P) \quad \text{minimiser } f(x) \quad x \in \mathbb{R}^n \quad (3.16)$$

où  $f$  est régulière (continûment différentiable) et  $g$  est son gradient. Notons par  $g_k$  le gradient de  $f$  au point  $x_k$ .

Rappelons que les différentes méthodes du gradient conjugué génèrent des suites  $\{x_k\}$  de la forme suivante :

$$x_{k+1} = x_k + \alpha_k d_k \quad \alpha_k > 0, \quad (3.17)$$

où la direction recherchée est définie par la formule de récurrence suivante :

$$d_k = \begin{cases} -g_1 & \text{si } k = 1, \\ -g_k + \beta_k d_{k-1} \dots & \text{si } k \geq 2, \end{cases} \quad (3.18)$$

Le coefficient  $\beta_k$  détermine la méthode du gradient conjugué en question (Fletcher Reeves, Polak Ribière Polyak,...) Le pas  $\alpha_k \in \mathbb{R}^+$  étant déterminé par une recherche linéaire.

### Supposition 3.1.

(i) L'ensemble  $\Omega := \{x \in \mathbb{R}^n; f(x) \leq f(x_1)\}$  est borné ; où  $x_1 \in \mathbb{R}^n$  est le point initial.

(ii) Sur un voisinage  $N$  de  $\Omega$ , la fonction objectif  $f$  est continûment différentiable et son gradient est lipschitzien i.e

$$\exists L > 0 \text{ tel que } \|g(x) - g(\tilde{x})\| \leq L \|x - \tilde{x}\|, \forall x, \tilde{x} \in \mathcal{N} \quad (3.19)$$

Ces suppositions impliquent qu'il existe  $\gamma > 0$  tel que

$$\|g(x)\| \leq \gamma, \forall x \in \mathcal{L} \quad (3.20)$$

### Définition 3.3 .([40, 1992])

On dit que  $d_k$  est une direction de descente suffisante si

$$g_k^T d_k < -c \|g_k\|^2. \quad (3.21)$$

Rappelons les conditions de Wolfe faibles :

$$f(x_k + \alpha d_k) \leq f(x_k) + \delta \alpha \nabla f(x_k)^T \cdot d_k. \quad (3.22)$$

$$\nabla f(x_k + \alpha d_k)^T \cdot d_k \geq \sigma \nabla f(x_k)^T \cdot d_k. \quad (3.23)$$

avec  $0 < \delta < \sigma < 1$  :

Les conditions de Wolfe fortes :

$$|\nabla f(x_k + \alpha d_k)^T \cdot d_k| \leq -\sigma \nabla f(x_k)^T \cdot d_k \quad (3.24)$$

$$|\nabla f(x_k + t d_k)^T \cdot d_k| \leq -\sigma \nabla f(x_k)^T \cdot d_k \quad (3.25)$$

Les conditions de Wolfe relaxée :

$$f(x_k + \alpha d_k) \leq f(x_k) + \delta \alpha \nabla^T f(x_k) d_k \quad (3.26)$$

$$\acute{\sigma} \nabla^T f(x_k) d_k \leq \nabla^T f(x_k + \alpha_k d_k) d_k \leq -\acute{\sigma} \nabla^T f(x_k) d_k, \quad (3.27)$$

où  $0 < \delta < \acute{\sigma} < 1$  et  $\acute{\sigma} > 0$

Présentons maintenant un théorème fondamental qui assure la condition de Zoutendijk pour toute méthode du type (3.17)-(3.18), dans laquelle le pas  $\alpha_k$  est déterminé par la règle de Wolfe faible (3.22)-(3.23). Ce théorème était démontré par Zoutendijk ([52, 1970]) et Powell ([53, 1971]).

Considérons une méthode du type (3.17) et (3.18), dans laquelle  $d_k$  est une direction de descente et le pas  $\alpha_k$  est déterminé par la règle de Wolfe faible (3.22)-(3.23) avec  $0 < \delta < 1/2$  : Considérons aussi que la supposition 4.1 soit satisfaite. Alors pour une telle méthode la condition de Zoutendijk suivante :

$$\sum_{k \geq 1} \cos^2 \theta_k \|g_k\|^2 \leq \infty \quad (3.28)$$

**Algorithm 3.2** (*Méthode du gradient conjugué*)

si on fixe  $x_0$ , puis on calcule  $f_0 = f(x_0)$  et  $\nabla f_0 = \nabla f(x_0)$ .

l'algorithme est initialisé par une étape du gradient simple.  $d_0 = -\nabla f_0$ .

Tant qu'un critère de convergence n'est pas vérifié :

1. Détermination d'un pas  $\alpha_k$  par la méthode de recherche linéaire de son choix. Calcule d'un nouvel itéré  $x_{k+1} = x_k + \alpha_k d_k$  :

2. Evaluation d'un nouveau gradient  $\nabla f_{k+1}$  :
3. Calcul du parametre  $B_{k+1}$  par la méthode de son choix
4. Construction d'une nouvelle direction  $d_{k+1} = -g_{k+1} + \beta_{k+1}d_k$
5. Incréméntation :  $k = k + 1$

### 3.4.1 méthode de Fletcher-Reeves

La méthode de Fletcher-Reeves ([11, 1964]) est une extension directe de la méthode du gradient conjugué linéaire au cas des fonctions quelconques. Appliquée à une fonction quadratique, elle est identique au gradient conjugué linéaire.

$$\beta_k^{FR} = \frac{\|g_k\|^2}{\|g_{k-1}\|^2} \quad \text{Gradient conjugué variante Fletcher-Reeves} \quad (3.29)$$

Lorsque la fonction à minimiser est quadratique, le coefficient  $B_K$  assure la conjugaison entre  $d_k$  et  $d_{k-1}$  et la méthode est assurée de converger en au plus  $n$  itérations, où  $n$  est la dimension du vecteur d'inconnues. Dans le cas non quadratique, il existe de nombreuses variantes à la méthode de Fletcher et Reeves, différant par le choix du paramètre  $B_K$

#### Algorithme de la méthode de Fletcher-Reeves

Cette méthode est très intéressante, d'une part parce qu'elle nécessite le stockage de très peu d'informations (essentiellement trois vecteurs de dimension  $n$ ); d'autre part, par sa vitesse de convergence très supérieure à celle des algorithmes du gradient classique

##### Algorithme 3.1 de la méthode de Fletcher-Reeves

###### Etape 0 : (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.

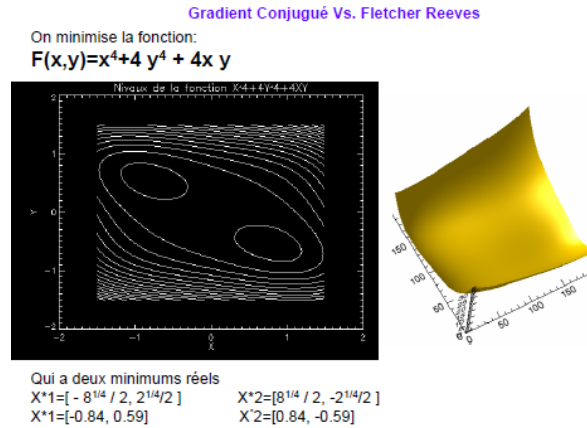
###### Etape 1 :

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 :  $\alpha_k = \arg \min_{\lambda > 0} f(x_k + \lambda d_k)$   
 $d_{k+1} = -g_{k+1} + \beta_{k+1}^{FR} d_k$ .



où

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

### 3.4.2 . La propriété de descente de la méthode de FR

Powell ([42, 1984]) a démontré la satisfaction de la propriété de descente de la fonction objective pour la méthode de Fletcher-Reeves avec recherche linéaire exacte.

Al-Baali ([50, 1985]) a démontré la satisfaisse de la propriété de descente de la fonction objectif pour la méthode de Fletcher-Reeves avec la recherche linéaire inexacte de Wolfe forte.

J. C. Gilbert et Nocedal([40, 1992]) ont généralisé ce résultat pour toute méthode du type (3.17)-(3.18) dont

$$|\beta_k| \leq \beta_k^{FR}. \quad (3.30)$$

**Théorème 3.3** *On considère que la supposition (3.1) est satisfaite. Considérons une méthode du type(3.17) et (3.18) avec  $\beta_k$  satisfaisant à (3.29) et le pas  $t_k$  vérifiant la règle de Wolfe forte (3.25) où  $\sigma \in ]0, \frac{1}{2}[$*

Alors cette méthode génère des directions de descente. De plus on a :

$$\frac{-1}{1-\sigma} \leq \frac{d_k^T g_k}{\|g_k\|^2} \leq \frac{2\sigma-1}{1-\sigma}; \quad k = 1, \dots \quad (3.31)$$

**Proof.** . ([40, 1992]) 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 (3.31) est satisfaite pour  $k$  et démontrons qu'elle le sera pour  $k+1$  : Supposons que :

$$\frac{-1}{1-\sigma} \leq \frac{d_k^T g_k}{\|g_k\|^2} \leq \frac{2\sigma-1}{1-\sigma}; \quad k = 1, \dots \quad (*)$$

On a :

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

D'autre part de (3.29) on aura :

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

d'où :

$$\frac{d_{k+1}^T g_{k+1}}{\|g_{k+1}\|^2} = -1 + \frac{\beta_{k+1}}{\beta_k^{FR}} \frac{d_k^T g_{k+1}}{\|g_k\|^2}. \quad (**)$$

En utilisant la condition de recherche linéaire (3.25) on aura :

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

Remplaçant ceci dans (\*\*):

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

De (\*) on aura :

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

et de (3.30)

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

On aura :

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

Ce qui achève la démonstration. ■

D'après (\*\*\*) on déduit que pour tout  $k \geq 1$  :  $d_k$  est une direction de descente avec  $C = \frac{2\sigma-1}{1-\sigma}$

En effet :

$$\frac{-1}{1-\sigma} \leq \frac{d_k^T g_k}{\|g_k\|^2} \leq \frac{2\sigma-1}{1-\sigma} \Rightarrow d_k^T g_k \leq -C \|g_k\|^2 . \text{ ou } C = \frac{1-2\sigma}{1-\sigma}$$

**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 :

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

Puisque

$$\alpha_k = \arg \min_{t>0} f(x_k + \alpha_k d_k) = \arg \min_{t>0} \varphi_k(t)$$

Donc  $\alpha_k$  vérifie la condition nécessaire d'optimalité :

$$\dot{\varphi}_k(\alpha_k) = \nabla f(x_k + \alpha_k d_k)^T . d_k = g_{k+1}^T d_k = 0 , \quad \forall k \geq 1.$$

### Convergence de la méthode de Fletcher-Reeves

Le premier résultat de convergence de la méthode du gradient conjugué non linéaire (version Fletcher- Reeves) avec des recherches linéaires inexactes (recherche linéaire inexacte de Wolfe forte (3.24)-(3.25))

où  $\sigma < \frac{1}{2}$ ) était démontré par Al-Baali ([50, 1985]). Touati Ahmed et Story ([51,1990]) ont généralisé ce résultat pour

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

Gilbert et Nocedal ([48, 1992]) ont généralisé ce résultat pour

$$|\beta_k| \leq \beta_k^{FR} \quad (3.32)$$

**Théorème 3.4** *Supposons que l'hypothèse 4.1 soit satisfaite. Considérons une méthode du type (3.17) et (3.18) dont  $\beta_k$  satisfait à (3.30) et le pas  $\alpha_k$  satisfait aux conditions de Wolfe fortes (3.24)-(3.25) où  $\sigma \in ]0, \frac{1}{2}[$  : Alors cette méthode est globalement convergente, dans le sens suivant :*

$$\liminf_{k \rightarrow \infty} \|g_k\| = 0. \quad (3.33)$$

**Proof.** . ([49, 1992])

Puisque les conditions du théorème 4.1 sont satisfaites alors on a :

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

D'autre part de (3.10)

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

d'où

$$|d_{k-1}^T g_k| \leq -\sigma d_{k-1}^T g_{k-1} \leq \frac{\sigma}{1-\sigma} \|g_{k-1}\|^2. \quad (3.34)$$

De (3.9), (3.32) et (3.34) :

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

Posons

$\hat{\omega} = \frac{1+\sigma}{1-\sigma}$  on aura :

$$\begin{aligned}
 \|d_k\|^2 &\leq \hat{\omega} \|g_k\|^2 + (\beta_k^{FR})^2 \|d_{k-1}\|^2 \\
 &\leq \hat{\omega} \|g_k\|^2 + (\beta_k^{FR})^2 \left[ \hat{\omega} \|g_{k-1}\|^2 + (\beta_{k-1}^{FR})^2 \|d_{k-2}\|^2 \right] \\
 &= \hat{\omega} \|g_k\|^4 \sum_{j=2}^k \|g_j\|^{-2} + \hat{\omega} \|g_k\|^4 \|g_1\|^{-2} = \hat{\omega} \|g_k\|^4 \sum_{j=1}^k \|g_j\|^{-2}.
 \end{aligned}$$

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

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

de (3.20) on a :

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

d'où

$$\sum_{k \geq 1} \frac{1}{\|d_k\|^2} \geq \frac{\omega^2}{\hat{\omega} \gamma^4} \sum_{k \geq 1} \frac{1}{k} > \infty. \quad (3.35)$$

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

D'autre part, 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 (3.35), d'où le résultat :

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

■

### 3.4.3 Méthode de Polak-Ribière-Polyak

Cette méthode fut découverte par Polak, Ribière ([18, 1969]) et Polyak ([19, 1969]). Rappelons que pour la méthode de Polak-Ribière-Polyak la variante  $\beta_k$  est :

$$\beta_k^{PRP} = \frac{g_k^T y_{k-1}}{\|g_{k-1}\|^2} \quad (3.36)$$

**Algorithme 3.2 de la méthode de Polak-Ribière-Polyak**

**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 :

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

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

où

$$\beta_{k+1}^{PRP} = \frac{g_{k+1}^T (g_{k+1} - g_k)}{\|g_k\|^2} = \frac{g_{k+1}^T y_k}{\|g_k\|^2}$$

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

**Convergence de la méthode de PRP**

La convergence de cette méthode est assurée pour une fonction fortement convexe avec recherche linéaire, mais si  $f$  n'est pas convexe elle ne converge pas.

**Théorème 3.5** *Si  $f$  est fortement convexe, continûment différentiable avec un gradient lipschitzien, alors la méthode de Polak Ribière avec recherche linéaire exacte génère une suite  $\{x_k\}$  convergeant vers l'unique point  $x^*$  réalisant le minimum de  $f$ .*

**Proof.** ([33]) Montrons dans un premier temps que

$$\cos \theta_k = \frac{-d_k^T g_k}{\|g_k\| \|d_k\|},$$

est uniformément positif.

Grâce à la recherche linéaire exacte, on a

$$\begin{aligned}
 d_{k-1}^T y_{k-1} &= d_{k-1}^T (g_k - g_{k-1}) \\
 &= -d_{k-1}^T g_{k-1} = -(-g_{k-1} + \beta_{k-1}^{PRP} d_{k-2})^T g_{k-1} \\
 &= \|g_{k-1}\|^2.
 \end{aligned}$$

La forte convexité de  $f$  implique que

$$d_{k-1}^T y_{k-1} = \frac{1}{t_{k-1}} (x_k - x_{k-1})^T y_{k-1} \geq \frac{\eta}{t_{k-1}} \|x_k - x_{k-1}\|^2,$$

où  $\eta > 0$  est le module de forte convexité de  $f$ .

On en déduit, en utilisant la constante de lipschitz  $L$  de  $g$  :

$$|\beta_k^{PRP}| = \frac{|g_k^T y_{k-1}|}{\|g_{k-1}\|^2} = \frac{|g_k^T y_{k-1}|}{d_{k-1}^T y_{k-1}} \leq \frac{t_{k-1} L \|g_k\| \|x_k - x_{k-1}\|}{\eta \|x_k - x_{k-1}\|^2} = \frac{L}{\eta} \frac{\|g_k\|}{\|d_{k-1}\|}.$$

On peut alors borner  $\|d_k\|$  par :

$$\begin{aligned}
 \|d_k\| &\leq \|g_k\| + |\beta_k^{PRP}| \|d_{k-1}\| \\
 &\leq \|g_k\| + \frac{L}{\eta} \frac{\|g_k\|}{\|d_{k-1}\|} \|d_{k-1}\| \\
 &\leq \left(1 + \frac{L}{\eta}\right) \|g_k\|.
 \end{aligned}$$

Ensuite

$$d_k^T g_k = -\|g_k\|^2 \leq -\left(1 + \frac{L}{\eta}\right)^{-1} \|g_k\| \|d_k\|,$$

ou encore

$$\cos \theta_k = \frac{-d_k^T g_k}{\|g_k\| \|d_k\|} \geq \left(1 + \frac{L}{\eta}\right)^{-1}.$$

La condition de Zoutendijk est vérifiée. Donc  $\{f(x_k)\}$  est bornée inférieurement (car  $f$  est fortement convexe).

On en déduit que  $g_k \rightarrow 0$ .

D'autre part,  $\{x_k\}$  est bornée ( $f$  est fortement convexe) et possède donc des sous suites convergentes.

La limite de celles-ci ne peut être que l'unique minimum  $x^*$  de  $f$  (car  $g_k \rightarrow 0$ ).

Donc toute la suite  $\{x_k\}$  converge vers  $x^*$ . ■

### 3.4.4 Méthode de la descente conjugué

Cette méthode fût proposée en 1987 par Fletcher et Reeves([37, 1987]), Rappelons que pour la méthode de la descente conjuguée la variante  $\beta_k$  est :

$$\beta_k^{CD} = \frac{\|g_k\|^2}{-d_{k-1}^T g_{k-1}} \quad (3.37)$$

#### Algorithme 3.2 de la méthode de descente conjugué

**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 :

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

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

où

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

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

#### La propriété de descente de la méthode de la descente conjuguée

Fletcher ([37, 1987]) a démontré que la méthode de la descente conjuguée est une méthode de descente si le pas  $\alpha_k$  est déterminé par la règle forte de Wolfe(3.24) et (3.25) avec  $\sigma \leq \frac{1}{2}$ . Dai et Yuan ([35, 1996]) ont démontré que cette méthode avec la règle de Wolfe relaxé ou  $0 < \delta < \sigma < 1$  et  $0 \leq \sigma \leq 1$  génère des directions de descente a chaque itération  $k \geq 1$

#### Théorème 3.6

Supposons que l'hypothèse 3.1 est satisfaite.

Pour toute méthode du type (3.17.2) et (3.18) dont  $\beta_k$  satisfait à (3.37) et le pas  $\alpha_k$  satisfait aux conditions de Wolfe relaxée :

$$\begin{aligned} f(x_k + \alpha_k d_k) &\leq f(x_k) + \delta \alpha_k d_k^T g_k \\ \text{et } \sigma d_k^T g_k &\leq d_k^T g_{k+1} \leq -\sigma d_k^T g_k \end{aligned}$$

où  $0 < \delta < \sigma < 1$  et  $0 \leq \sigma \leq 1$

Alors notre méthode génère des directions de descente suffisante à chaque itération  $k \geq 1$

**Proof.** ([35, 1996]) On a

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

D'autre part de (4.12)

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

d'où

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

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

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

et donc  $d_k$  est une direction de descente suffisante ■

### Convergence de la méthode de la descente conjuguée

Yuan ([36, 1993]) a démontré la convergence au sens (3.18) de cette méthode avec un pas satisfaisant aux conditions (3.11)-(3.12) si  $\sigma < \frac{1}{2}$  et  $\acute{\sigma} = 0$ . Dai et Yuan ([35,1996]) ont démontré ce résultat pour  $\sigma < 1$  et  $\acute{\sigma} = 0$ ,

**Théorème 3.7** *Supposons que la l'hypothèse 3.1 est satisfaite. Toute méthode du type (3.17) et (3.18) dans laquelle  $\beta_k$  vérifie (3.19) et le pas  $\alpha_k$  est déterminé par la règle de Wolfe relaxée (3.11)-(3.12) où  $0 < \delta < \sigma < 1$  et  $\acute{\sigma} = 0$ ; est de descente convergente, dans le sens où*

$$\liminf_{k \rightarrow \infty} \|g_k\| = 0$$

**Proof.** ([35,1996]) Du théorème 3.4 on a :

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

Donc  $\beta_{k+1}^{CD}$  vérifie l'inégalité (3.15).D'après le théorème 3.2 on a :

$$\liminf_{k \rightarrow \infty} \|g_k\| = 0$$

■

■

### 3.4.5 Méthode de Dai-Yuan

Cette méthode était découverte par Dai et Yuan ([33, 1999]),

$$\beta_k^{DY} = \frac{\|g_k\|^2}{d_{k-1}^T y_{k-1}} \quad (3.38)$$

Remarquons que cette variante on a les mêmes numérateur et dénominateur des variantes de Fletcher-Reeves et Hestenes-Stiefel respectivement . Cette méthode possède plusieurs propriétés, par exemple elle possède la propriété de descente à chaque itération, la convergence au sens (3.32) si le pas est déterminé par la règle de Wolfe faible

**Algorithme 3.3 de la Méthode de Dai-Yuan avec la règle de Wolfe faible**

**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 :

$\alpha_k$  vérifie les conditions (3.25)

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

où

$$\beta_{k+1}^{DY} = \frac{\|g_{k+1}\|^2}{d_k^T [g_{k+1} - g_k]} = \frac{\|g_{k+1}\|^2}{d_k^T y_k}$$

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

**La propriété de descente de la méthode de Dai-Yuan**

Dai et Yuan ([34, 1998]) ont démontré que à chaque itération  $k \geq 1$  la direction recherchée par cette méthode avec la recherche de Wolfe faible (3.22)-(3.23), est de descente si la fonction objectif  $f$  est strictement convexe.

**Théorème 3.8** *Supposons que L'hypothèse "1 soit satisfaite. Pour toute méthode du type (3.17) et (3.18) dont  $\beta_k$  satisfait à (3.38) et le pas  $\alpha_k$  satisfait aux conditions de Wolfe faible :*

$$\begin{aligned} f(x_k + \alpha d_k) &\leq f(x_k) + \delta \alpha \nabla f(x_k)^T \cdot d_k \\ \text{et } d_k^T g_{k+1} &\geq \sigma d_k^T g_k \end{aligned}$$

où  $0 < \delta < \sigma < 1$

toutes les directions générées sont de descente, autrement dit  $:d_k^T g_k < 0$  ;  $\forall k \geq 1$

**Proof.** ([33, 1999]) La démonstration se fait par récurrence.1) Pour  $k = 1$  :

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

2) Supposons que (3.21) 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 (3.7), 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 :

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

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. ■

### Convergence de la méthode de Dai-Yuan

Dai-Yuan ([33, 1999]) ont démontré la convergence de la méthode de Dai-Yuan au sens (3.38) si le pas  $\alpha_k$  est déterminé par la règle de Wolfe faible, et la fonction objective satisfait les conditions qui sont dans la proposition suivantes.

**Théorème 3.9** *Supposons que la proposition 3.1 est satisfaite. La suite  $\{x_k\}$  générée par l'algorithme 3.3 converge dans le sens*

$$\liminf_{k \rightarrow \infty} \|g_k\| = 0$$

**Proof.** ([33, 1999]) En utilisant le théorème 3.1, on aura :

$$\sum_{k \geq 1} \cos^2 \theta \|g_k\|^2 < \infty \quad (*)$$

d'autre part on a :

$$\begin{aligned} \|d_{k+1} + g_{k+1}\|^2 &= \|\beta_{k+1}^{DY} d_k\|^2 \\ \Rightarrow \|d_{k+1}\|^2 &= (\beta_{k+1}^{DY})^2 \|d_k\|^2 - 2d_{k+1}^T g_{k+1} - \|g_{k+1}\|^2 (**)$$

De (3.35) :

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

remplaçant ceci dans (\*\*), on aura :

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

d'où

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

Supposons maintenant que (3.33) n'est pas satisfaite, autrement dit :

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

On aura :

$$\begin{aligned} \frac{\|d_k\|^2}{(d_k^T g_k)^2} &\leq \frac{1}{\|g_i\|^2} \leq \frac{1}{\omega^2} \sum_{i=1}^k 1 = \frac{1}{\omega^2} k \\ \Rightarrow \sum_{k \geq 1} \frac{(d_k^T g_k)^2}{\|d_k\|^2} &\geq \omega^2 \sum_{k \geq 1} \frac{1}{k} \end{aligned}$$

d'où

$$\sum_{k \geq 1} \frac{(d_k^T g_k)^2}{\|d_k\|^2} = \infty$$

ce qui contredit (\*) Ce qui achève la démonstration. ■

**Exemple 3.10** Minimiser  $f(x) = \frac{1}{2}x^T \begin{pmatrix} 2 & 0 \\ 0 & 8 \end{pmatrix} x + \begin{pmatrix} -4 \\ -8 \end{pmatrix} x$

par la méthode de Fletcher-Reeves à partir de  $x^0 = (0 \ 0)^T$ .

Vérifier que :

- les directions de déplacement sont mutuellement conjuguées
- les gradients sont orthogonaux
- $d_k^T g_k = -g_k^T g_k < 0$  (Condition de descente)

-----  
 $f(x)$  est quadratique

$\Rightarrow$  F.R.  $\equiv$  algo des gradients conjugués

$$\nabla f(x) = \begin{pmatrix} 2 & 0 \\ 0 & 8 \end{pmatrix} x + \begin{pmatrix} -4 \\ -8 \end{pmatrix}$$

Première itération :

$$x^0 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad d_0 = -\nabla f(x^0) = \begin{pmatrix} 4 \\ 8 \end{pmatrix} = -g_0$$

$$\alpha_k ? \quad \alpha_k = \frac{g_k^T g_k}{g_k^T A g_k}$$

$$\Rightarrow \alpha_0 = \frac{(4 \ 8) \begin{pmatrix} 4 \\ 8 \end{pmatrix}}{(4 \ 8) \begin{pmatrix} 2 & 0 \\ 0 & 8 \end{pmatrix} \begin{pmatrix} 4 \\ 8 \end{pmatrix}} = \frac{80}{544} = \frac{5}{34}$$

$$x^1 = x^0 + \alpha_0 d_0 = \frac{5}{34} \begin{pmatrix} 4 \\ 8 \end{pmatrix} = \begin{pmatrix} \frac{20}{34} \\ \frac{40}{34} \end{pmatrix}$$

$$g_1 = \nabla f(x^1) = \begin{pmatrix} \frac{-48}{17} \\ \frac{24}{17} \end{pmatrix}$$

$$\Rightarrow B_0 = \frac{g_1^T g_1}{g_0^T g_0} = \frac{36}{289}$$

$$d_1 = -g_1 + B_0 d_0 = \frac{120}{289} \begin{pmatrix} 8 \\ -1 \end{pmatrix}$$

Deuxième itération :

$$\alpha_1 = \frac{\frac{-24 \ 120}{17 \ 289} (-2 \ 1) \begin{pmatrix} 8 \\ -1 \end{pmatrix}}{\frac{120 \ 120}{289 \ 289} (8 \ -1) \begin{pmatrix} 2 & 0 \\ 0 & 8 \end{pmatrix} \begin{pmatrix} 8 \\ -1 \end{pmatrix}} = \dots = \frac{17}{40}$$

$$x^2 = x^1 + \alpha_1 d_1 = \begin{pmatrix} \frac{20}{34} \\ \frac{40}{34} \end{pmatrix} + \frac{17 \ 120}{40 \ 289} \begin{pmatrix} 8 \\ -1 \end{pmatrix} = \begin{pmatrix} 2 \\ 1 \end{pmatrix}$$

$$g_2 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

$$\Rightarrow B_1 = 0 \Rightarrow \alpha_2 = 0; \quad x^2 = x^3 = \dots$$

$\Rightarrow$  optimum atteint (pt stationnaire!)

La méthode converge bien en n (=2) étapes , au plus pour une fct quadratique.

Le min est en  $x^* = (2 \ 1)^T$ ;  $f(x^*) = -8$

C'est un min global (cf. répet précédente)

Vérifications :

• Directions mutuellement conjuguées ?

$$d_1^T A d_0 = \frac{120}{289} (8 \ -1) \begin{pmatrix} 2 & 0 \\ 0 & 8 \end{pmatrix} \begin{pmatrix} 4 \\ 8 \end{pmatrix} = 0$$

$\Rightarrow d_0, d_1$  conjuguées par rapport à A

• Gradients orthogonaux ?

$$\nabla f(x^0) = g_0 = \begin{pmatrix} -4 \\ -8 \end{pmatrix}$$

$$\nabla f(x^1) = g_1 = -\frac{24}{17} \begin{pmatrix} 2 \\ -1 \end{pmatrix}$$

$$\implies g_0^T g_1 = 0 \text{ ok}$$

•Condition de descente ?

$$k=0 \quad d_0^T g_0 = -g_0^T g_0 < 0 \quad \text{vérifie car } d_0 = -g_0$$

$$k=1 \quad d_1^T g_1 = \frac{120}{289} \begin{pmatrix} 8 & -1 \end{pmatrix} \begin{pmatrix} 2 \\ -1 \end{pmatrix} \frac{24}{17} = \frac{-120 \cdot 24}{289}$$

$$-g_1^T g_1 = -\left(\frac{24}{17}\right)^2 \begin{pmatrix} -2 & 1 \end{pmatrix} \begin{pmatrix} 2 \\ -1 \end{pmatrix} = \frac{-120 \cdot 24}{289}$$

$\implies d_0$  et  $d_1$  sont bien des directions de descente

## Chapitre 4

# Une nouvelle méthode de gradient conjugué non linéaire avec la recherche linéaire de Wolf-Powell fort

### 4.1 Introduction

Dans ce chapitre

Nous avons constaté que les résultats de convergence pour les différentes variantes du gradient conjugué ont été obtenus en utilisant des recherches linéaires inexactes. La question naturelle qui se pose est la suivante :

Peut-on obtenir des résultats de convergence avec des nouvelles variantes du Gradient Conjugué ou d'autres en utilisant la recherche linéaire non-monotone ?

Durant les dernières années des modifications et des améliorations de la méthode du gradient conjugué ont été proposées par G. H. LIU, L. L. JING, L. X. HAN, AND D. HAN ([63]), BONNANS, J. F., PANIER, E., TITS, A., and ZHOU, J. L ([62]), Y. H. Dai ([5]), E. Polak and G. Ribière ([18]). Wei et al.(29) On se basant sur ces études nous proposons de construire des nouvelles méthodes avec de bons résultats numériques

Récemment, Wei et al. [29] ont proposé une nouvelle formule.  $WYL$

La formule  $WYL$  a non seulement de bons résultats numérique mais aussi assure la condition suffisante de descente et les propriétés de convergence globale sous la recherche linéaire Wolfe-Powell forte[29,30].

. Dans ce context on proposer une nouvelle famille de coefficient  $B_k^{WYLM}$  qui est une extention du  $B_k^{wyl}$  de la méthode de gradient conjugué pour

**CHAPITRE 4. UNE NOUVELLE MÉTHODE DE GRADIENT  
CONJUGUÉ NON LINÉAIRE AVEC LA RECHERCHE  
LINÉAIRE DE WOLF-POWELL FORT**

---

résoudre les problèmes d'optimisation sans contrainte à grand taille et démontrons que ce  $B_k$  assurée la condition suffisent de descente et la problèmes de convergence sous la recherche linear exacte et inexact de Wolf fort,

On considère le problème d'optimisation sans contraintes suivants :

$$(p) : \min \left\{ f(x) : x \in \tilde{\mathbb{R}}^n \right\} \quad (4.1)$$

où  $f : \tilde{\mathbb{R}}^n \rightarrow \mathbb{R}$  est une fonction non linéaire, lisse dont le gradient est  $g$ . La formule itérative de la méthode du gradient conjugué est donnée par :

$$x_{k+1} = x_k + \alpha_k d_k, \alpha_k > 0, k = 0, 1, 2, 3, \dots \quad (4.2)$$

où  $\alpha_k$  est une longueur de pas obtenu par une recherche linéaire unidimensionnelle et  $d_k$  est le direction de la recherche définie par

$$d_k = \begin{cases} -g_1 & \text{si } k = 1, \\ -g_k + \beta_k d_{k-1} \dots \dots \dots \text{si } k \geq 2, \end{cases} \quad (4.3)$$

où  $g_k = \nabla f(x_k)$  et  $\beta_k$  est un scalaire, Il y'a quatre variantes pour  $\beta_k$ , dans ce recherche :

$$\beta_k^{DY} = \frac{g_k^T g_k}{(g_k - g_{k-1})^T d_{k-1}} \quad \text{Dai-Yuan [7], 1999} \quad (4.4)$$

$$\beta_k^{PRP} = \frac{g_k^T (g_k - g_{k-1})}{\|g_{k-1}\|^2} \quad \text{Polak-Ribière-Polyak [18, 19], 1969} \quad (4.5)$$

$$B_k^{wyl} = \frac{g_k^T \left( g_k - \frac{\|g_k\|}{\|g_{k-1}\|} g_{k-1} \right)}{g_{k-1}^T g_{k-1}} \quad \text{Wei et al [29], 2006} \quad (4.6)$$

$$\beta_k^{FR} = \frac{g_k^T g_k}{g_{k-1}^T g_{k-1}}, \quad \text{Fletcher Reeves [11], 1964} \quad (4.7)$$

pour trouver le pas  $\alpha_k$  il y a beaucoup de recherches linéaire souvent utilisé comme

-Recherche linéaire exact,

$$f(x_k + \alpha_k d_k) = \min f(x_k + \alpha d_k), \alpha > 0. \quad (4.8)$$

-Recherche linéaire de Wolf fort

$$f(x_k + \alpha_k d_k) - f(x_k) \leq \delta \alpha_k \nabla f(x_k)^T \cdot d_k \dots \quad (4.9)$$

$$|\nabla f(x_k + \alpha_k d_k)^T \cdot d_k| \leq -\sigma \nabla f(x_k)^T \cdot d_k. \quad (4.10)$$

**CHAPITRE 4. UNE NOUVELLE MÉTHODE DE GRADIENT  
CONJUGUÉ NON LINÉAIRE AVEC LA RECHERCHE  
LINÉAIRE DE WOLF-POWELL FORT**

---

ou  $\delta \in (0, \frac{1}{2})$  et  $\sigma \in (0, 1)$

D'un point de vue bibliographique, les méthodes du gradient conjugué non linéaire peuvent être améliorées en utilisant les nouvelles techniques proposées dans (68.), (69.), (70.), (71) ".

dans ce chapitre nous présenterons la nouvelle formule dans la section 2. En outre, la condition suffisante de descente et la convergence globale de la nouvelle méthode sous la recherche linéaire exacte et la recherche linéaire inexacte (4.9), (4.10) sera présentée à la section 3. Enfin, nous discuterons des résultats numériques et des conclusions à les sections 4 et 5 respectivement.

## 4.2 nouvelle formule de GC

Dans cette section, nous proposons le nouveau  $B_k$  qui est une extension du  $B_k^{wyl}$  (4.6) que nous avons nommé  $B_k^{WYLM}$  (la méthode de wyl modifiée)

$$B_k^{WYLM} = \frac{g_k^T \left( g_k - \frac{\|g_k\|}{\|g_{k-1}\|} d_{k-1} \right)}{g_{k-1}^T g_{k-1}}. \quad (4.11)$$

## 4.3 analyse convergente

Dans cette section, nous allons montrer les propriétés convergentes de  $B_k^{WYLM}$  en utilisant à la fois la recherche linéaire exacte et inexacte

### 4.3.1 Analyse convergente basée sur une recherche linéaire exacte

Dans cette section, nous allons montrer l'analyse convergente basée sur la recherche linéaire exacte. Nous montrerons aussi que ce coefficient GC possèdera la condition suffisante de descente et la convergence globale

#### la conditions suffisante de descente

pour la condition suffisante de descente doit,

$$g_k^T d_k \leq -\|g_k\|^2, k \geq 0 \quad (4.12)$$

**Théorème 4.1** *Considérons une méthode du CG de la forme (4.3) et  $B_k^{WYLM}$  donné comme dans (4.11) le (4.12) et vérifie*

**CHAPITRE 4. UNE NOUVELLE MÉTHODE DE GRADIENT  
CONJUGUÉ NON LINÉAIRE AVEC LA RECHERCHE  
LINÉAIRE DE WOLF-POWELL FORT**

---

**Proof.** la preuve est par induction

si  $k = 0$ , donc  $g_k^T d_k = -\|g_k\|^2$ . pour  $k \geq 1$ , multiplier (4.3) par  $g_{k+1}^T$ , donc

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

pour une recherche linear exact il est facile à connaitre  $g_{k+1}^T d_k = 0$ .

$$\text{Donc } g_{k+1}^T d_{k+1} = -\|g_{k+1}\|^2$$

Cela impliquera que  $d_{k+1}$  est une direction suffisant de descent.

d'ou,  $g_k^T d_k \leq -\|g_k\|^2$   $k \geq 0$ . est vrai et la preuve est terminée ■

### Propriétés de convergence globale

Ensuit, nous montrerons que la méthode GC avec  $B_{k+1}$  Convergerons globalement, D'abord nous devons simplifier notre nouveau  $B_{k+1}$  afin que notre preuve de convergence sera plus facile. De (4.12) nous savons que

$$B_k^{WYLM} = \frac{\|g_k\|^2 - \frac{\|g_k\|}{\|g_{k-1}\|} |g^T d_{k-1}|}{\|g_{k-1}\|^2} \leq \frac{\|g_k\|^2}{\|g_{k-1}\|^2}$$

En utilisant l'négaité de Cauchy-Schwartz :

$$B_k^{WYLM} = \frac{\|g_k\|^2 - \frac{\|g_k\|}{\|g_{k-1}\|} |g^T d_{k-1}|}{\|g_{k-1}\|^2} \geq \frac{\|g_k\|^2 - \frac{\|g_k\|}{\|g_{k-1}\|} (\|d_{k-1}\| \|g_k\|)}{\|g_{k-1}\|^2} \geq 0$$

ainsi nous obtenons

$$B_k^{WYLM} \geq 0 \tag{4.13}$$

$$B_k^{WYLM} \leq \frac{\|g_k\|^2}{\|g_{k-1}\|^2} \tag{4.14}$$

### 4.3.2 Convergence globale par la recherche linear exacte

Les hypothèses de base suivantes sont toujours nécessaires dans l'analyse des propriétés de convergence des méthodes GC

**Supposition1 :**

On suppose que

1) l'ensemble de niveau  $\Omega = \{x \in \mathbb{R}^n; f(x) \leq f(x_0)\}$  est borné. où  $x_0$  est le point de départ

2) Dans certains voisinages  $N$  de  $\Omega$ ,  $f(x)$  est continûment différentiable et son gradient est lipschitz continu, c'est-à-dire, pour tout  $x, y \in N$ , il existe une constante  $L \geq 0$  telle que

$$\|g(x) - g(y)\| \leq L \|x - y\| \tag{4.15}$$

**CHAPITRE 4. UNE NOUVELLE MÉTHODE DE GRADIENT  
CONJUGUÉ NON LINÉAIRE AVEC LA RECHERCHE  
LINÉAIRE DE WOLF-POWELL FORT**

---

Cette hypothèse donne le lemme suivant (Lemme 1), qui a été prouvé par Zoutendijk [22].

**Lemme 4.1** *Supposons que la supposition 1 soit vraie. Considérons toute méthode GC de la forme (4.3), où  $d_k$  est une direction de descente et  $\alpha_k$  vérifie la règle de minimisation exacte. La condition suivante, connue sous le nom de condition Zoutendijk,*

$$\sum_{k=0}^{\infty} \frac{(g_k^T d_k)^2}{\|g_k\|^2} < \infty \quad (4.16)$$

Substituer (4.14) à (4.16), il est équivalent à ce qui suit

$$\sum_{k=0}^{\infty} \frac{\|g_k\|^4}{\|d_k\|^2} < \infty \quad (4.17)$$

**Théorème 4.2** *Supposons que les hypothèses (1) et (2) ont vérifiées. considérons la méthode de la forme (4.2) et (4.3), où  $\alpha_k$  est obtenu par recherche linéar exact. Supposons en outre que l'hypothèse (1) et la condition suffisante descente soient vraies. alors*

$$\begin{aligned} \lim_{k \rightarrow \infty} \|g_k\| = 0 \text{ ou } \sum_{k=0}^{\infty} \frac{(g_k^T d_k)^2}{\|g_k\|^2} < \infty \\ \lim_{k \rightarrow \infty} \|g_k\| = 0 \end{aligned} \quad (4.18)$$

**Proof.** Pour Démontrer. le Théorème 2, nous utilisons un argument de contradiction. Par conséquent, si le Théorème 2 n'est pas vrai, alors il existe  $\xi > 0$ , tel que

$$\|g_k\| > \xi \quad (4.19)$$

Réécriture (4.3) en tant que  $d_k + g_k = B_k^{WYLM} d_{k-1}$ , et les deux côtés de l'équation, nous obtenons

$$\|d_k\|^2 = (B_k^{WYLM})^2 \|d_{k-1}\|^2 - 2g_k^T d_k - \|g_k\|^2 \quad (4.20)$$

Divisant les deux côtés de (4.20) par  $(g_k^T d_k)^2$ ; alors nous obtenons  $\frac{\|d_k\|^2}{(g_k^T d_k)^2} =$

$$\begin{aligned} \frac{(B_k^{WYLM})^2 \|d_{k-1}\|^2}{(g_k^T d_k)^2} - \frac{2g_k^T d_k}{(g_k^T d_k)^2} - \frac{\|g_k\|^2}{(g_k^T d_k)^2} \\ = \frac{(B_k^{WYLM})^2 \|d_{k-1}\|^2}{(g_k^T d_k)^2} - \left( \frac{1}{\|g_k\|} + \frac{\|g_k\|}{(g_k^T d_k)} \right)^2 + \frac{1}{\|g_k\|^2} \end{aligned}$$

**CHAPITRE 4. UNE NOUVELLE MÉTHODE DE GRADIENT  
CONJUGUÉ NON LINÉAIRE AVEC LA RECHERCHE  
LINÉAIRE DE WOLF-POWELL FORT**

---

$$\frac{\|d_k\|^2}{(g_k^T d_k)^2} \leq \frac{(B_k^{WYLM})^2 \|d_{k-1}\|^2}{(g_k^T d_k)^2} + \frac{1}{\|g_k\|^2} \quad (4.21)$$

En appliquant (4.13) et (4.14) nous avons

$$\frac{\|d_k\|^2}{(g_k^T d_k)^2} \leq \frac{1}{\|g_k\|^2}$$

D'où,

$$\frac{\|d_k\|^2}{(g_k^T d_k)^2} \leq \sum_{i=0}^k \frac{1}{\|g_i\|^2}$$

$$\frac{(g_k^T d_k)^2}{\|d_k\|^2} \geq \frac{\xi^2}{k} \quad (4.22)$$

donc, de (4.22) et (4.19), cela montre que

$$\sum_{k=0}^{\infty} \frac{(g_k^T d_k)^2}{\|d_k\|^2} = \infty$$

Cela contredit la condition Zoutendijk dans Lemma1 Par conséquent, la preuve est terminée. ■

### 4.3.3 Analyse convergente basée sur une recherche linéaire inexacte

Dans cette section, nous allons montrer la convergence de la méthode sous la recherche linéaire inexact de wolf fort. Nous montrerons Aussi Que ces coefficients GC possèdent la condition suffisante de descente et des propriétés de convergence globales. sous la recherche linéaire inexact (4.9) et (4.10)

Dans le théorème suivant, nous discutons de la condition suffisante de décente,

$$g_k^T d_k \leq -C \|g_k\|^2, C > 0 \quad (4.23)$$

où  $k \geq 0$  et  $c \in (0, 1)$  sous recherche linéaire (SWP).

#### la conditions suffisante de descente

Pour la condition suffisante de descente, nous présentons leThéorem suivant

**Théorème 4.3** *Si les séquences  $g_k$  et  $d_k$  sont générées par les méthodes (4.2), (4.3) et (4.11) avec la longueur de pas  $\alpha_k$  est déterminée par(4.9) et (4.10) si ■  $\in (0, 1/4)$ , alors la condition suffisante de décente est vérifie.*

**CHAPITRE 4. UNE NOUVELLE MÉTHODE DE GRADIENT  
CONJUGUÉ NON LINÉAIRE AVEC LA RECHERCHE  
LINÉAIRE DE WOLF-POWELL FORT**

---

**Proof.** Nous utilisons la preuve par induction de (4.3). Nous savons que pour  $k = 0$  il est en attente. Supposons que c'est.

$$\begin{aligned}
 g_k^T d_k &= -\|g_k\|^2 + B_k^{WYLM} g_k^T d_{k-1} \\
 \text{Diviser par } \|g_k\|^2 &\text{ a indiqué que ;} \\
 \frac{g_k^T d_k}{\|g_k\|^2} &= -1 + B_k^{WYLM} \frac{g_k^T d_{k-1}}{\|g_k\|^2} \\
 &= -1 + \left( \frac{g_k^T \left( g_k - \frac{\|g_k\|}{\|g_{k-1}\|} d_{k-1} \right)}{g_{k-1}^T g_{k-1}} \right) \frac{g_k^T d_{k-1}}{\|g_k\|^2} \\
 &= -1 + \frac{1}{\|g_k\|^2} \left( g_k^T g_k - \frac{\|g_k\| g_k^T d_{k-1}}{\|g_{k-1}\|} \right) \frac{g_k^T d_{k-1}}{g_{k-1}^T g_{k-1}} \\
 \frac{g_k^T d_k}{\|g_k\|^2} &= -1 + \frac{g_k^T d_{k-1}}{g_{k-1}^T g_{k-1}} \left( 1 - \frac{g_k^T d_{k-1}}{\|g_k\| \|g_{k-1}\|} \right) \tag{4.24}
 \end{aligned}$$

en utilisant (4.10), nous avons

$$-1 + \frac{\sigma g_{k-1}^T d_{k-1}}{\|g_{k-1}\|^2} \left( 1 - \frac{g_k^T d_{k-1}}{\|g_k\| \|g_{k-1}\|} \right) \leq \frac{g_k^T d_k}{\|g_k\|^2} \leq -1 + \frac{-\sigma g_{k-1}^T d_{k-1}}{\|g_{k-1}\|^2} \left( 1 - \frac{g_k^T d_{k-1}}{\|g_k\| \|g_{k-1}\|} \right) \tag{4.25}$$

et en appliquant l'inégalité de Cauchy-Schwartz nous obtenons

$$0 \leq 1 - \frac{g_k^T d_{k-1}}{\|g_k\| \|g_{k-1}\|} \leq 2, \tag{4.26}$$

Cela implique que

$$-1 + 2\sigma \frac{g_{k-1}^T d_{k-1}}{\|g_{k-1}\|^2} \leq \frac{g_k^T d_k}{\|g_k\|^2} \leq -1 - 2\sigma \frac{g_{k-1}^T d_{k-1}}{\|g_{k-1}\|^2} \tag{4.27}$$

En répétant ce processus et le fait  $g_1^T d_1 = -\|g_1\|^2$ , nous avons

$$-\sum_{j=0}^{k-1} (2\sigma)^j \leq \frac{g_k^T d_k}{\|g_k\|^2} \leq -2 + \sum_{j=0}^{k-1} (2\sigma)^j \tag{4.28}$$

comme

$$-\frac{1}{1-2\sigma} \leq \frac{g_k^T d_k}{\|g_k\|^2} \leq -2 + \frac{1}{1-2\sigma}, \tag{4.29}$$

En faisant la restriction  $\sigma \in (0, \frac{1}{4})$  nous avons  $g_k^T d_k < 0$ . Donc, par induction,  $\forall k \in N$   $g_k^T d_k < 0$  vérifie. Maintenant, nous démontrons la condition suffisante de descente  $d_k$ . si  $\sigma \in (0, \frac{1}{4})$

**CHAPITRE 4. UNE NOUVELLE MÉTHODE DE GRADIENT  
CONJUGUÉ NON LINÉAIRE AVEC LA RECHERCHE  
LINÉAIRE DE WOLF-POWELL FORT**

---

on pose  $\lambda = 2 - \frac{1}{1-2\sigma}$ , donc  $0 < \lambda < 1$ , et (4.29) se révèle être

$$\lambda - 2 \leq \frac{g_k^T d_k}{\|g_k\|^2} \leq -\lambda, \quad (4.30)$$

Ainsi nous obtenons  $g_k^T d_k \leq -\gamma \|g_k\|^2$  ou  $\gamma = -2 + \frac{1}{1-2\sigma}$ , où  $\lambda \in (0, 1)$ .  
La preuve est terminée. ■

### 4.3.4 Analyse convergence globale

L'hypothèse suivante est nécessaire pour procéder à la démonstration des propriétés de convergence globale.

**Supposition 1 :**

(i) L'ensemble  $\Omega := \{x \in \mathbb{R}^n; f(x) \leq f(x_1)\}$  est borné. où  $x_1 \in \mathbb{R}^n$  est le point initial.

(ii) Dans certains voisinages  $N$  de  $\Omega$ ,  $f(x)$  est continûment différentiable et son gradient est lipschitz continu, c'est-à-dire, pour tout  $x, y \in N$ , il existe une constante  $L \geq 0$  telle que

$$\|g(x) - g(y)\| \leq L \|x - y\|$$

**Théorème 4.4** *Supposons que la supposition 1 est vérifiées. considérons la méthode de la forme (4.2) et (4.3), où  $d_k$  satisfait  $g_k^T d_k < 0$ , pour tout  $k$ , et  $t_k$  est obtenu par SWP (4.9) et (4.10) alors,  $\lim_{k \rightarrow \infty} \|g_k\| = 0$*

**Proof.** nous avons  $g_k^T d_k < 0$  pour tout  $k \geq 1$  Nous avons aussi de (4.9) et utiliser la condition de Lipschitz nous avons que

$$-(1 - \sigma)g_k^T d_k \leq (g_{k+1} - g_k)^T d_k \leq Lt_k \|d_k\|^2 \quad (4.31)$$

d'où

$$t_k \geq -\frac{(1 - \sigma) g_k^T d_k}{L \|d_k\|^2} \quad (4.32)$$

avec (4.9) on obtient :

$$f(x_k) - f(x_k + t_k d_k) \geq -\delta t_k g_k^T d_k \geq -\delta \frac{(1 - \sigma) (g_k^T d_k)^2}{L \|d_k\|^2} \quad (4.33)$$

En outre, de l'hypothèse (i) nous avons que  $\{f(x_k)\}$  est une suite décroissante et a une limite dans  $\Omega$ , ce qui montre que  $\lim_{k \rightarrow \infty} f(x_{k+1}) < +\infty$ ; et d'après (4.33) on a :

$$+\infty > f(x_1) - \lim_{k \rightarrow \infty} f(x_{k+1}) = \sum |f(x_k) - f(x_{k+1})| \geq \delta \frac{(1 - \sigma)}{L} \sum \frac{(g_k^T d_k)^2}{\|d_k\|^2} \quad (4.34)$$

alors  $\sum \frac{(g_k^T d_k)^2}{\|d_k\|^2} < +\infty$ . par conséquent  $\lim_{k \rightarrow \infty} \|g_k\| = 0$

La preuve est terminée ■

## 4.4 .Résultats numériques et discussions

Dans cette section, nous présentons les résultats numériques de notre méthode proposée *WYLM* et on comparaison avec les méthodes *CG* de *DY*, *WYL*, *FR* et *PRP*. Nous utiliserons certains problèmes de test considérés dans Andrei [61] pour analyser l'efficacité de  $B_k^{WYLM}$ . Nous avons considéré  $\epsilon = 10^{-6}$ ,  $\sigma = 0.1$ ,  $\delta = 0.01$  et la valeur du gradient comme critère d'arrêt. la tolérance  $\epsilon = 10^{-6}$  est choisie pour tous les algorithmes pour étudier la rapidité d'itération de ces algorithmes vers la solution optimale, Nous avons utilisé Matlab R2010 . les résultats de performance sont présentés dans les figures 1, 2, 3,4 et 5

nous utilisons l'algorithme suivant

**Algorithm 4.5 étape 1 :** *Étant donné un point initial  $x_0 \in R^n$  posons  $k = 0$ .  $\epsilon \in (0, 1)$ , et  $d_0 = -g_0$*

*étape 2 :* Calculer  $B_k$  par (4.4).(4.5), (4.6), (4.7) ou (4.10)

*étape 3 :* Calculer  $d_k$  par(4.3), si  $\|g_k\| = 0$ , alors stop.

*étape 4 :* Calculer  $\alpha_k$  basé sur la (Recherche linéaire de Wolf fort)

*étape 5 :* Mise à jour du nouveau point basé sur (4.2).

*étape 6 :* Test convergent et critères d'arrêt, si  $f(x_k) \leq f(x_{k+1})$  et  $\|g_k\| < \epsilon$ , alors stop, sinon, passez à l'étape 1 avec  $k = k + 1$ .

Tableau 1. Problèmes de test utilisés avec la condition de Wolf fort

$N_0$	Fonction	N	point initial
1	Booth	2	(10,10),(14 ,14),(-12 ,-12)
2	Rosenbrock	2	(5, 5),(25, 25),(17, 17)
3	Goldstein-Price	2	(2,2),(-11 ,-11),(-13 ,-13)
4	Extended Powell	4	(7, 7,7,7) (15, 15,15,15)
5	Tridiagonal1	2	(2, 2),(100, 100),(17, 17)
6	Strait	2	(50, 50),(100, 100)
7	Generalized Quartic	2	(10, 10),(200, 200),(17, 17)

Les figures (4.1) à (4.3) présentent les performances des méthodes ci-dessus par rapport au nombre d'itérations, au nombre d'évaluations de gradient et les temps CPU, respectivement.

CHAPITRE 4. UNE NOUVELLE MÉTHODE DE GRADIENT  
CONJUGUÉ NON LINÉAIRE AVEC LA RECHERCHE  
LINÉAIRE DE WOLF-POWELL FORT

---

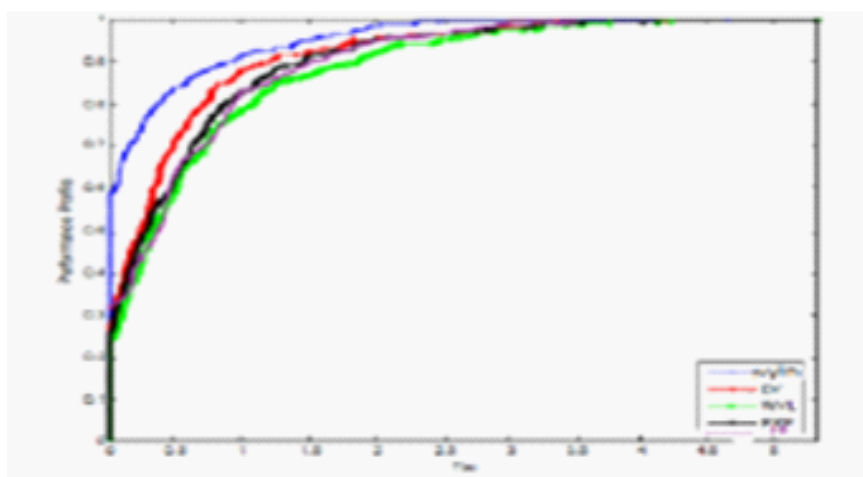


FIG. 4.1 – Performance based en the number of iterations

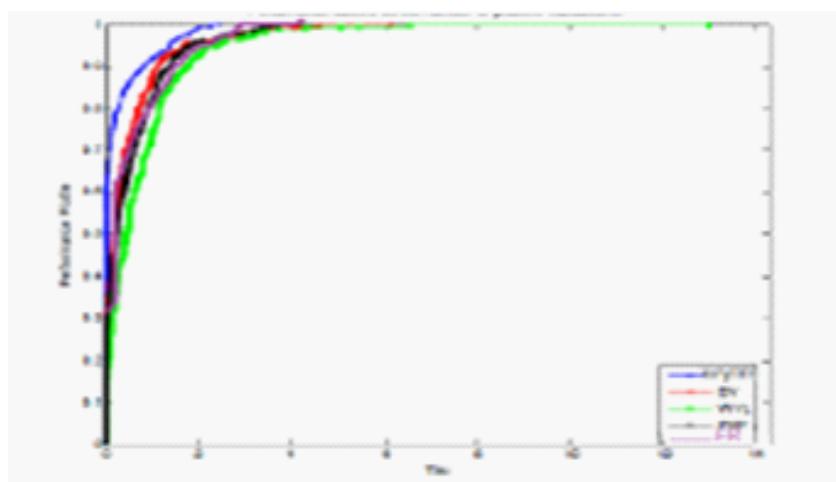


FIG. 4.2 – Performance based on the number of gradient evaluations

**CHAPITRE 4. UNE NOUVELLE MÉTHODE DE GRADIENT  
CONJUGUÉ NON LINÉAIRE AVEC LA RECHERCHE  
LINÉAIRE DE WOLF-POWELL FORT**

---

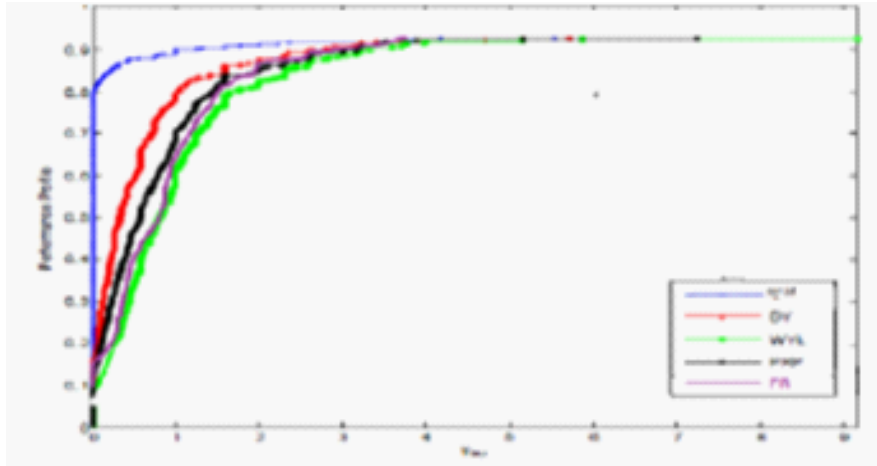


FIG. 4.3 – Performance based CPU time

Remarque 1 : Les figures 1, 2 et 3 montrent que la méthode «WYLM» présente les meilleures performances car elle résout environ 99% des problèmes de test.

**Exemple 4.6** Considérer le problème suivant Rosenbrock function  $f(x, y) = (1 - x)^2 + 100(y - x^2)^2$ ,  $x$  optimal = (1,1)

$$\nabla f(x, y) = \begin{pmatrix} -2 + 2x - 400xy + 400x^2 \\ 200y - 200x^2y \end{pmatrix}$$

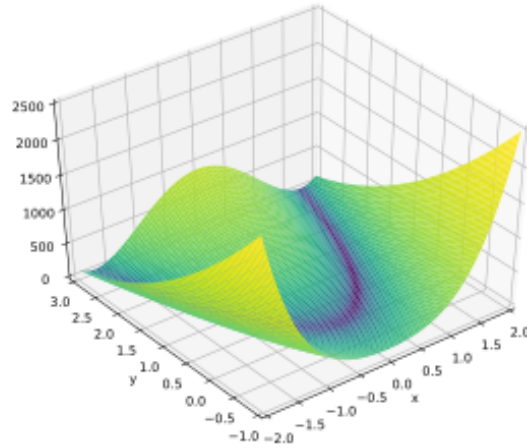
minimiser la fonction  $f(x, y)$

**Table 1** : Fonction Rosenbrock,  $f(x, y) = (x - 1)^2 + 100 (x^2 - y)^2$ ,  $x$  optimal=(1, 1)

Résultats numériques pour WYLM, WYL, DY, FR et PRP en termes de nombre d'itérations  $NI$  et de temps  $cpu$

**CHAPITRE 4. UNE NOUVELLE MÉTHODE DE GRADIENT  
CONJUGUÉ NON LINÉAIRE AVEC LA RECHERCHE  
LINÉAIRE DE WOLF-POWELL FORT**

---



point initiale	WYLM	WYL	PRP	FR	DY
	NI/CPU	NI/CPU	NI/CPU	NI/CPU	NI/CPU
(1000,1000)	290/0.8948	Failed	8377/42.896	Échoué	Failed
(30, 30)	119 / 0.2919	1511/2.61386	8377/42.896	3944	Failed
(-1,-1000)	316/1.2419	2324/ 7.48127	Failed	Failed	Failed
(-1,3)	88/1.99364	14532/ 75.441	466/ 2.60282	420/ 2.6787	170/ 0.2194
(0, 1)	124/0.42538	20000/ 118.46	510/ 3.01014	88.0.215204	100/ 0.3769
(1, 4)	224/0.80902	2000/113.847	456/2.33869	130/ 2.0867	100 /0.3769
(100, 100)	210/ 0.60740	Failed	Failed	5840/ 25.10	Failed
(-1, 7)	201/ 0.75430	20000/113.84	527/ 6.11810	130 / 2.086	219 / 0.4429

**Remarque 2** : dans le tableau 2, la méthode WYLM a réussi dans toutes les tentatives visant à atteindre la solution optimale, alors que les autres méthodes ont échoué

Tableau 3. Résumé des résultats

Méthode	Classement	Le taux de réussite
<i>WYLM</i>	1	100%
<i>PRP</i>	2	75%
<i>FR</i>	3	75%
<i>WYL</i>	4	75%
<i>DY</i>	5	50%

**Remarque 3** : Le tableau 3 montre que «WYLM» offre les meilleures performances puisqu'il résout environ 100% des problèmes de test.

**CHAPITRE 4. UNE NOUVELLE MÉTHODE DE GRADIENT  
CONJUGUÉ NON LINÉAIRE AVEC LA RECHERCHE  
LINÉAIRE DE WOLF-POWELL FORT**

---

Tableau 4 : fonction de Rosenbrock, point initial  $[-3 \ 100]$

	WYLM	WYL	PRP	FR	DY
NI	309	Failed	3505	Failed	2456
CPU(s)	1,0048	Failed	13,2389	Failed	8,5297
X optimal	(1, 1)		(1, 1)		(1, 1)

Les figures (4.5) et (4.6) présentent la comparaison entre la méthode WYLM et les méthodes  $DY, WYL, PRP$  et  $FR$   $x_0 = [-3 \ 100]$

CHAPITRE 4. UNE NOUVELLE MÉTHODE DE GRADIENT  
CONJUGUÉ NON LINÉAIRE AVEC LA RECHERCHE  
LINÉAIRE DE WOLF-POWELL FORT

---

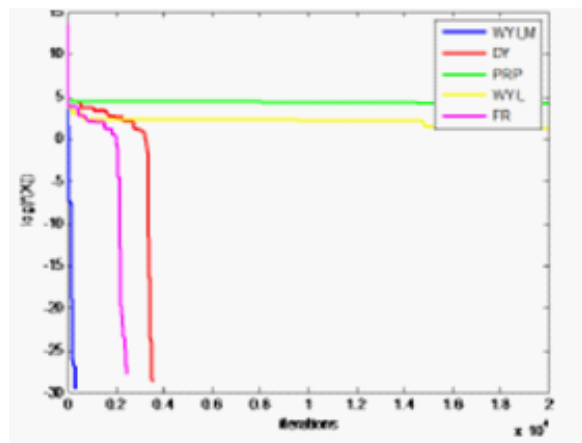


FIG. 4.4 – Profils de performance basés sur le nombre d'évaluations de fonction

**CHAPITRE 4. UNE NOUVELLE MÉTHODE DE GRADIENT  
CONJUGUÉ NON LINÉAIRE AVEC LA RECHERCHE  
LINÉAIRE DE WOLF-POWELL FORT**

---

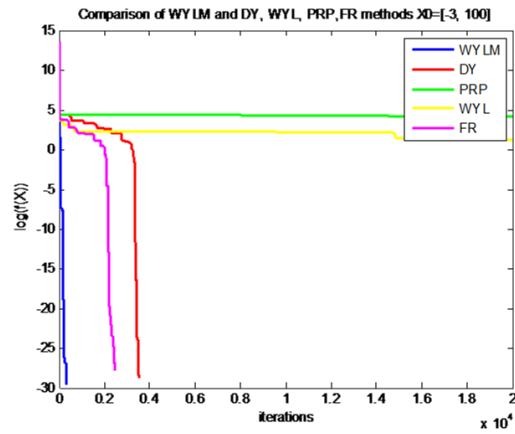


FIG. 4.5 – Profils de performance basés sur le nombre d'évaluations de fonction

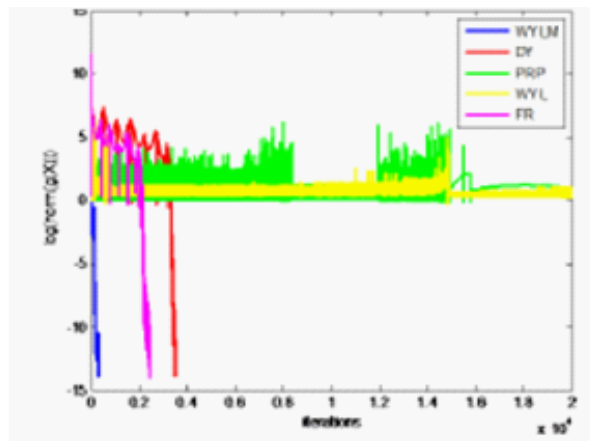


FIG. 4.6 – Profils de performance basés sur le nombre d'évaluations de gradient

**CHAPITRE 4. UNE NOUVELLE MÉTHODE DE GRADIENT  
CONJUGUÉ NON LINÉAIRE AVEC LA RECHERCHE  
LINÉAIRE DE WOLF-POWELL FORT**

---

**Remarque 4 :** Parmi les figures 4 et 5, la méthode *WYLM* fonctionne mieux que d'autres méthodes en sélectionnant un point de départ avec la fonction de Resenbrock  $f(x, y) = (x - 1)^2 + 100(x^2 - y)^2$  Et elle est la meilleure performance en termes de gradients de valeurs et de fonctions et le nombre d'itérat

**commentaires**

(a) A travers les résultats enregistrés dans les tables 1 et 2, nous remarquons une différence significative entre la méthode  $B_k^{WYLM}$  et les autres méthodes

1) La nouvelle méthode converge vers la solution optimale très rapidement

2) La nouvelle méthode prend beaucoup moins de temps d'autres méthodes

3) Lorsque  $x_0$  est trop loin, la nouvelle méthode converge à une vitesse incroyable tandis que d'autres méthodes échouent

4) Le nombre d'itérations inférieur au nombre d'itérations d'autres méthodes

(b) D'après les Figures 1 à 3, il est clairement démontré que la nouvelle méthode *WYLM* est meilleure que les autres méthodes. La meilleure méthode a une courbe qui est en haut et à droite sur le graphique. En outre, le nouveau coefficient résout toutes les fonctions de test ci-dessus où les autres méthodes ne peuvent pas

## 4.5 conclusion

Nous présentons une nouvelle formule  $B_k^{WYLM}$  similaire à la méthode *wyl* et les propriétés de convergence globales sont présentées avec plusieurs recherches linear

Nos résultats numériques ont montré que le nouveau coefficient *WYLM* est efficace lorsqu'il est comparé aux méthodes *PRP*, *DY*, *FR* et *WYL*

# Bibliographie

- [1] 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.  $1.6749286 \times 10^{-27}$  kg
- [2] X. X. Chen, J. Sun, Global convergence of a two-parameter family of conjugate gradient methods without line search *J. Comput Appl.Math.* 146 (2002) 37–
- [3] Y. Dai, Convergence of nonlinear conjugate methods, *J. Comput Math.* 19 (2001) 539–549
- [4] Y. Dai, Convergence of Polak–Ribie‘re–Polyak conjugate gradient method with constant stepsizes, Manuscript, Institute of Computational Mathematics and Scientific/Engineering Computing, Chinese Academy of Sciences, 2001.
- [5] Y. Dai, J. Han, G. Liu, D. Sun, H. Yin, Y. Yan, Convergence properties of nonlinear conjugate methods, *SIAM J. Optim.* 2 (1999) 345–358
- [6] Y. Dai, Y. Yan , Further studies on the Polak–Ribie‘re–Polyak method, Research Report ICM-95-040, Institute of Computational Mathematics and Scientific/ Engineering Computing, Chinese Academy of Sciences, 1995.
- [7] Y. Dai, Y. Yan , A nonlinear conjugate gradient with a strong global convergence properties, *SIAM J. Optim.* 10 (2000) 177–182.
- [8] Y. Dai, Y. Yan , A nonlinear Conjugate Gradient Methods, Science Press of Shanghai, Shanghai, 2000.
- [9] Y. Dai, Y. Yuan, An efficient hybrid conjugate gradient method for unconstrained optimization, *Ann. Oper. Res.* 103 (2001) 33–47
- [10] Practical Method of Optimization, second ed., Unconstrained Optimization, vol. I, Wiley, New York, 1997.
- [11] R. Fletcher, Reeves, Function minimization by conjugate gradients, *Compute. J.* 7 (1964) 149–154

- 
- [12] , J. Nocedal, Global convergence properties of conjugate gradient methods for optimization, *SIAM J. Optim.* 2 (1992) 21–42
- [13] L. Grippo, S. Lucidi, A globally convergent version of the Polak–Ribie‘re gradient method, *Math. Prog.* 78 (1997) 375–391
- [14] M.R. Hestenes, E. Stiefel, Method of conjugate gradient for solving linear equations, *J. Res. Nat. Bur. Stand.* 49 (1952) 409–436..
- [15] Y. Liu, C. Storey, Efficient generalized conjugate gradient algorithms, part 1 : theory, *J. Optim. Theory Appl.* 69 (1992) 129–137.
- [16] J. Nocedal, Conjugate gradient methods and nonlinear optimization, in : L. Adams, J.L. Nazareth (Eds.), *Linear and Nonlinear Conjugate Gradient Related Methods*, SIAM, Philadelphia, 1995, pp. 9–23.
- [17] E. Polak, *Optimization : Algorithms and Consistent Approximations*, Springer, New York, 1997.
- [18] E. Polak, G. Ribie‘re, Note Sur la convergence de directions conjuge‘es, *Rev. Francaise Informat Recherche Operationelle*, 3e Anne‘e 16 (1969) 35–43
- [19] B.T. Polyak, The conjugate gradient method in extreme problems, *USSR Comp. Math. Math. Phys.* 9(1969) 94–112.
- [20] M.J.D. Powell, Nonconvex minimization calculations and the conjugate gradient method, *Lecture Notes in Mathematics*, vol. 1066, Springer, Berlin, 1984, pp. 122–141.
- [21] J. Sun, J. Zhang, Convergence of conjugate gradient methods without line search, *Ann. Oper. Res.* 103(2001) 161–173
- [22] , G. Zoutendijk, Nonlinear programming computational methods, in : J. Abadie (Ed.), *Integer and Nonlinear Programming*, North-Holland, Amsterdam, 1970, pp. 37–86
- [23] Z. Wei et al., The convergence properties of some conjugate gradient methods, *Appl. Math. Nonlinear Programming*, North-Holland, Amsterdam, 1970, pp. 37–86
- [24] E. Polak, G. Ribie‘re, Note Sur la convergence de directions conjuge‘es, *Rev. Francaise Informat Recherche Operationelle*, 3e Anne‘e 16 (1969) 35–43.
- [25] B.T. Polyak, The conjugate gradient method in extreme problems, *USSR Comp. Math. Math. Phys.* 9 (1969) 94–112
- [26] .Powell, Nonconvex minimization calculations and the conjugate gradient method, *Lecture Notes in Mathematics*, vol. 1066, Springer-Verlag, Berlin, 1984, pp. 122–141.

- 
- [27] Zhang, Convergence of conjugate gradient methods without line search, *Annal. Oper. Res.*103 (2001) 161–173.
- [28] Nonlinear programming computational methods, in : J. Abadie (Ed.), *Integer and Nonlinear Programming*, North-Holland, Amsterdam, 1970, pp. 37–86.
- [29] Wei et al., The convergence properties of some conjugate gradient methods, *Appl. Math. Comput.* 183 (2006) 1341–1350.
- [30] The proof of the sufficient descent condition of the Wei–Yao–Liu conjugate gradient method under the strong Wolfe–Powell line search, *Appl. Math. Comput.* doi :doi :10.1016/j.amc.2006.12.006, 2007
- [31] Lin Suihua, Yao Shengwei, A new conjugate gradient method of modified LS formula, Department of Mathematics and Information Science, Guangxi University, 2006.
- [32] Shengwei, Lin Suihua, A new conjugate gradient method combined HS and DY formulas, Department of Mathematics and Information Science, Guangxi University, 2
- [33] Y.H. Dai and Y. Yuan (1999), A non linear conjugate gradient with a strong global convergence property, *SIAM J. Optimization*, Vol. 10(1), pp.177-182.
- [34] Y. H. Dai and Y. Yuan (1998), Some properties of a new conjugate gradient method, in : *Advances in Nonlinear Programming*, ed . Kluwer Academic, Boston, pp. 251-262.
- [35] Y. H. Dai and Y. Yuan (1996), Convergence properties of the conjugate descent method, *Advances in Mathematics*, 6, pp.552-562.Y
- [36] Y.X.Yuan (1993), *Numerical Methods for Non linear Programming*, Shanghai scienti. . . c & Technical Publishers (in Chinese)
- [37] .R. Fletcher (1987), *Practical methods of optimization*, John Wiley&Sons, Chichester.
- [38] P. Armand (2005), Modification of the Wolfe Line Search Rules to Satisfy the Descent Condition in the Polak-Ribière-Polyak Conjugate Gradient Method, *Journal of Optimization Theory and Applications*, Vol. 131, pp. 103-115.
- [39] L. Grippo and S. Lucidi (1997), A Globally Convergent Version of the Polak-Ribière Conjugate Gradient Method, *Mathematical Programming*, Vol. 78, pp. 375–391
- [40] J.C. Gilbert and J. Nocedal (1992), Global convergence properties of conjugate gradient methods for optimization, *SIAM J. Optimization*. Vol. 3, No.1, pp. 21-42.

- 
- [41] M.J.D. Powell (1986), Convergence properties of algorithms for nonlinear optimization, *SIAM rev.*, 28, pp. 487-500
- [42] M.J.D. Powell (1984), Non convex minimization calculation and the conjugate gradient method, in : *Lecture Notes in Mathematics 1066* (Springer, Berlin), pp. 122-14
- [43] M.J.D. Powell (1986), Convergence properties of algorithms for nonlinear optimization, *SIAM rev.*, 28, pp. 487-500.
- [44] A.S. Nemirovsky et D.B. Yudin (1983), *Problem Complexity and Method Efficiency*, New York, Wiley.
- [45] M.J.D. Powell (1977), Restart procedures for the conjugate gradient method, *Math Programming*, 2 , pp 241-254.
- [46] Y. H. Dai and Y. Yuan (1996), Convergence properties of the Fletcher-Reeves method, *IMA J Numer. Anal.*, Vol.16(2), pp. 155-164.
- [47] G.H. Liu and J.Y. Han and H.X. Yin (1995), Global convergence of the Fletcher-Reeves algorithm with an inexact line search, *Appl. math. J. Chinese Univ. Ser. B*, 10, pp. 75-82.
- [48] J.C. Gilbert and J. Nocedal (1992), Global convergence properties of conjugate gradient methods for optimization, *SIAM J. Optimization*. Vol. 3, No.1, pp. 21-42.
- [49] J. C. Gilbert and J. Nocedal, Global convergence properties of conjugate gradient methods for optimization, *SIAM. J. Optimization*. Vol. 2 No. 1 (1992), pp. 21-42
- [50] M. Al-Baali (1985), Descent property and global convergence of the Fletcher-Reeves method with inexacte line search. *IMA J. Num. Anal.*, Vol. 5, pp.121-124.
- [51] D. Touati-Ahmed and C. Storey (1990), Efficient hybrid conjugate gradient techniques, *JOTA*, 64, pp. 379-397.
- [52] G. Zoutendijk (1970), *Nonlinear Programming Computational Methods*, Integer and Nonlinear Programming, North Holland, Amsterdam.
- [53] M.J.D. Powell (1971), On the convergence of the variable metric algorithm ; *J. Inst. Math. Appl*, 7, pp. 21-36
- [54] Y.H. Dai and Y. Yuan (1999), A non linear conjugate gradient with a strong global convergence property, *SIAM J. Optimization*, Vol. 10(1),pp.177-182.
- [55] R. Fletcher (1987), *Practical methods of optimization*, John Wiley&Sons, Chichester

- 
- [56] R. Fletcher and C. Reeves (1964), Function minimization by conjugate gradients. *Comput. J.*, 7, pp.149-154
- [57] B.T. Polyak (1969), The conjugate gradient method in extremem problems, *Comput. Math. Math. Phys.*, 9, pp. 94-112...
- [58] E. Polak and G. Ribière (1969), Note sur la convergence de directions conjuguées, *Rev. Française Informat. Recherche Operationelle*, 3e année, 16, pp. 35-43.
- [59] Y.H. Dai and Y. Yuan (1999), A non linear conjugate gradient with a strong global convergence property, *SIAM J. Optimization*, Vol. 10(1),pp.177-182
- [60] L. Armijo (1966), Minimzation of function having lipschitz continuous...rst partial derivatives, *Paci...c Journal of Mathematics*, Vol. 16(1), pp.1-3.
- [61] N.Andrei,An unconstrained optimization test functions collection,*Adv.Model.Optim*10(2008)147-161
- [62] BONNANS, J. F., PANIER, E., TITS, A., and ZHOU, J. L. : Avoiding the Maratose effect by means of non monotone line search, *SIAM J.Numer.Anal.* 29,no.4(Aug.1992), 1187-1202
- [63] G. H. LIU, L. L. JING, L. X. HAN, AND D. HAN , A class of non-monotone conjugate gradiente methods for unconstrained optimization, *Journal of Optimization Theory and Applications*, 101 (1999), pp. 127-140
- [64] A. Pietrabissa and L. Ricciardi Celsi, "Discrete-Time Selfish Routing Converging to the Wardrop Equilibrium," in *IEEE Transactions on Automatic Control*, vol. 64, no. 3, pp. 1288-1294, March 2019. doi : 10.1109/TAC.2018.2847602
- [65] A. Di Giorgio, A. Giuseppi, F. Liberati, A. Ornatelli, A. Rabezzano and L. R. Celsi, "On the optimization of energy storage system placement for protecting power transmission grids against dynamic load altering attacks," 2017 25th Mediterranean Conference on Control and Automation (MED), Valletta,2017,pp.986-992. doi : 10.1109/MED.2017.7984247
- [66] L. R. Celsi, A. Di Giorgio, R. Gambuti, A. Tortorelli and F. Delli Priscoli, "On the many-to-many carpooling problem in the context of multi-modal trip planning," 2017 25th Mediterranean Conference on Control and Automation (MED) Valletta, 2017,pp. 303-309.doi : 10.1109/MED.2017.7984135
- [67] V. Suraci, L. R. Celsi, A. Giuseppi and A. Di Giorgio, "A distributed wardrop control algorithm for load balancing in smart grids," 2017 25th

- Mediterranean Conference on Control and Automation (MED), Valletta, 2017, pp. 761-767.
- [68] Al-Baali, M., Caliciotti, A., Fasano, G. et al. “Exploiting damped techniques for nonlinear conjugate gradient methods *Math Meth Oper Res* (2017) “, 86 : 501.
- [69] Caliciotti Andrea<sup>1</sup>, Fasano Giovanni, Roma M. “Preconditioning strategies for nonlinear conjugate gradient methods“, based on quasi-Newton updates *AIP Conference Proceedings* 1776, 090007 (2016).
- [70] Al-Baali M., Caliciotti A., Fasano G., Roma M. (2018) “Quasi-Newton Based Preconditioning and Damped Quasi-Newton Schemes for Nonlinear Conjugate Gradient Methods“. In : Al-Baali M., Grandinetti L., Purnama A. (eds) *Numerical Analysis and Optimization. NAO 2017. Springer Proceedings in Mathematics & Statistics*, vol 235. Springer, Cham.
- [71] Caliciotti, Andrea, Fasano, Giovanni ;Roma, Massimo, “Preconditioned Nonlinear Conjugate Gradient methods based on a modified secant equation“, *Applied Mathematics and Computation*, Volume 318, 1 February 2018, Pages 196-214.
- [72] A.R. Conn, K. Scheinberg, and Ph.L. Toint. Recent progress in unconstrained nonlinear optimization without derivatives. *Mathematical Programming*, 79 :397–414, 1997
- [73] G. Cohen. Convexité et optimisation. Ecole Nationale des Ponts et Chaussées et INRIA, <http://cel.archivesouvertes.fr/docs/00/35/66/86/PDF/Ponts-cours-A4-NB.pdf>, 2000.
- [74] WEI, Z. X. , LI, G. Y. , QI, L. Q., New nonlinear conjugate gradient formulas for large-scale unconstrained optimization problems, *Appl. Math. Comput.*179, 407-430 (2006) [16] YU, G. H. , ZHAO, Y. L.
- [75] SHANNO, D.F., Conjugate gradient methods with inexact searches , *Math. Oper. Res.* 3, 244–256 (1978) .