

وزارة التعليم العالي والبحث العلمي
Ministère de l'enseignement supérieur
et la recherche scientifique

Université Badji Mokhtar
– Annaba –

Badji Mokhtar University
– Annaba –



جامعة باجي مختار
– عنابة –

Faculté des Sciences
Département de Mathématiques

THÈSE

Présentée en vue de l'obtention du diplôme de
Doctorat En Mathématiques

MÉTHODES À RÉGION DE CONFIANCE ET GRADIENT CONJUGUÉ

Option

Mathématiques Appliquées

Présentée par

HACHELFI DALAL

DIRECTEUR DE THESE: LASKRI YAMINA PROF, U.B.M. ANNABA
CO-DIRECTEUR DE THESE: BENZINE RACHID PROF, E.P.S.T. ANNABA

Devant le jury

PRESIDENTE : REBBANI FAOUZIA PROF, U.B.M. ANNABA
EXAMINATEUR DJEBBAR BACHIR PROF, U.S.T. Oran
EXAMINATEUR ELLAGGOUNE FATEH MC(A), U. Guelma
EXAMINATEUR NISSE LAMINE MC(A), U.B.M. ANNABA
INVITÉ SAHARI MED LAMINE MC(B), U.B.M. ANNABA

Année: 2014 /2015

Table des matières

Remerciements	iv
I Introduction	1
1 Position du problème	2
1.1 Formulation Mathématique.	4
1.2 Conditions d'optimalité.	5
1.3 Taux de convergence.	7
1.4 Objet et apport de ce travail.	8
2 Méthodes de globalisation en optimisation	11
2.1 Principe général des Méthodes à directions de descente	11
2.1.1 Direction de descente	12
2.1.2 Description des méthodes à directions de descentes . .	13
2.1.3 Exemples de méthodes à directions de descente	14
2.1.4 Exemples de choix du directions de descente	14
2.1.5 Exemples du choix de pas α_k	14
2.2 Recherche linéaire	14
2.2.1 Objectifs à atteindre	15
2.2.2 Recherches linéaires exactes et inexactes	16
2.2.3 Intervalle de sécurité	18
2.2.4 Algorithme de base	18
2.2.5 Recherches linéaires inexactes	19
2.2.6 La règle d'Armijo	20
2.2.7 La règle de Goldstein&Price.	23
2.2.8 La règle de Wolfe	27

2.2.9	Convergence des méthodes à directions de descente . . .	32
2.3	Méthode du gradient conjugué.	35
2.4	Méthode de Dai-Yuan	36
2.4.1	Description de la méthode	36
2.4.2	La propriété de descente de la méthode de Dai-Yuan . .	39
2.4.3	Convergence de la méthode de Dai-Yuan	40
II Méthodes à région de confiance et gradient conjugué		42
3	Méthodes des pénalités	43
3.1	La convergence des méthodes de pénalité	44
4	Méthodes à régions de confiance	47
4.1	Méthodes à régions de confiance pour le cas quadratique . . .	47
4.1.1	Optimisation de fonctions modèles quadratiques	47
4.1.2	Fonctions modèles utilisant le gradient exact de la fonction objectif	48
4.1.3	Description d'un sous-problème standard à région de confiance	49
4.1.4	Algorithme 1(Région de confiance pour des fonctions modèle quadratique)	50
4.1.5	Méthode du point de Cauchy	52
4.1.6	Calcul pratique du point de Cauchy	53
4.1.7	Condition de décroissance suffisante	54
4.1.8	Résultats de convergence	55
4.1.9	Fonctions modèles utilisant un gradient approché de la fonction objectif	59
4.2	Optimisation de fonctions modèles générales	60
5	Méthode à région de confiance et gradient conjugué	64
5.1	Algorithme 5.1.	65
5.2	Résultats de convergence	67
5.2.1	Le point de Cauchy	67
5.2.2	Condition de décroissance suffisante	68
5.2.3	Convergence des méthodes de pénalité	69
5.2.4	Résultats de convergence	72

TABLE DES MATIÈRES

5.3 Tests numériques 75

Remerciements

Je remercie Dieu tout-puissant, qui m'a donné la force et la patience pour accomplir cette thèse.

Il n'est un secret pour personne que la rédaction d'une thèse est un exercice qui met à contribution bien plus de personnes que celles dont le nom figure sur la couverture de l'ouvrage. De nombreuses personnes ont par leur participation parfois inconsciente contribué au bon déroulement de ce projet.

Je voudrais en tout premier lieu remercier Madame Laskri Yamina, Professeur à l'Université Badji Mokhtar Annaba, à qui je dois cette aventure. Je voudrais d'abord la remercier pour sa confiance, son soutien, son attention, son intérêt pour l'évolution de mes recherches, ses bons conseils, ses qualités humaines et sa grande compréhension face à mes choix de vie qui ne furent pas exactement de ceux qui simplifient la vie d'un promoteur de thèse.

Je remercie également l'Université Badji Mokhtar-Annaba, la faculté des Sciences et le département de Mathématiques pour m'avoir permis de mener mes recherches dans un environnement stimulant. À ce titre, je tiens tout spécialement à remercier les enseignants du département de Mathématique pour toutes les discussions scientifiques approfondies et les orientations qui m'ont guidées tout au long de l'acheminement de ce travail .

Monsieur le professeur Rachid Benzine qui est à la base de l'idée d'introduire cet axe de recherche dans le groupe d'optimisation.

Je remercie Madame Faouzia Rebbani, Professeur à l'Université Badji Mokhtar Annaba m'a fait le grand honneur de présider le jury, Je l'en remercie très vivement,

Monsieur Djebbar Bachir, Professeur à l'Université des sciences et technologies d'Oran, Monsieur Ellagoune Fateh, Maître de Conférences, classe A à l'Université de Guelma, et Monsieur Lamine NISSE, Maître de Conférences classe A à l'Université Badji Mokhtar Annaba de m'avoir fait l'honneur d'examiner cette thèse.

Je n'oublie pas de remercier infiniment Monsieur Sahari Mohamed Lamine, Maître de Conférences classe B à l'Université Badji Mokhtar Annaba pour l'intérêt qu'il a porté à ce travail et pour l'aide et les idées qu'il m'a donné pour la bonne évolution de mes recherches aussi bien en théorie qu'en pratique.

Monsieur Jérôme Pousin, Professeur à l'Institut National des Sciences Appliquées de Lyon en France à eu l'amabilité de m'accueillir dans son laboratoire au cours de mon stage, je le remercie pour tous ses conseils et son aide surtout concernant les simulations numériques.

Enfin, j'adresse mes plus sincères remerciements à mes proches, mes parents, mes soeurs et mes amies pour le soutien constant tout au long des ces années de thèse.

Résumé

Pour résoudre les problèmes d'optimisation sans contraintes du type

$$\left\{ \min_{x \in \mathbb{R}^n} f(x) \right.$$

On utilise en général les méthodes à directions de descente et les méthodes à région de confiance.

Dans ce travail, on s'intéresse aux résultats de convergence qui utilisent les méthodes à région de confiance et qui emploient comme sous programme le gradient conjugué à chaque itération. De plus, on va pénaliser les problèmes quadratiques avec contraintes et les convertir en une série de problèmes sans contraintes.

Abstract

For solving the unconstrained optimization problem (p)

$$(p) \left\{ \min_{x \in \mathbb{R}^n} f(x) \right.$$

We use generally, descent directions methods and trust region methods.

In this paper, we are interested to convergence results that use trust region methods and which employ such as subprogram the conjugate gradient for each iteration and in addition, will be penalized the quadratic problems with constraints and convert them a series of unconstrained problems.

تلخيص

لحل مسائل الامتثال دون قيود من نوع

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

نستخدم على العموم طرق التوجه التنازلي و طرق مجالات الثقة.

في هذا العمل، نهتم بنتائج التقارب التي تستعمل طرق مجالات الثقة وفق طرق التدرج المقارن كبرنامج ثانوي في كل تكرار. بالإضافة الى ذلك فإنه سيتم استثناء المسائل التربيعية بقيود الى سلسلة من المسائل دون قيود.

Première partie

Introduction

Chapitre 1

Position du problème

Depuis la nuit des temps, l'homme optimise : que ce soit pour accélérer une technique de fabrication, pour minimiser le coût d'une construction ou pour ne pas payer plus d'impôt qu'il n'en faut. Il doit cependant tenir compte des contraintes externes pour parvenir à ses fins : accélérer une fabrication doit laisser inchangée la qualité du produit, minimiser le coût d'une construction ne doit pas pousser à l'inaction (la construction la moins chère étant celle qui n'existe pas) et le calcul de l'impôt doit répondre aux prescrits légaux et fiscaux.

Il convient d'abord de choisir sur base de quel(s) critère(s) un objet peut être considéré comme meilleur qu'un autre. La meilleure voiture est-elle la moins chère sur le marché ? Auquel cas l'épreuve de sélection est relativement simple : il suffit de comparer les prix des différents modèles. Mais la meilleure peut aussi être la plus sûre et, dans ce cas, une série d'expériences est nécessaire. Des dizaines d'autres critères sont possibles : espace disponible, vitesse, consommation de carburant, etc. Mais la définition de l'optimum peut aussi résulter d'une combinaison de ces différents critères : le rapport qualité/prix est la plus célèbre de ces combinaisons.

Les techniques d'optimisation sont des procédés systématiques permettant d'approcher, voire de trouver la technique de fabrication la plus rapide, le coût le plus bas ou l'impôt le plus juste. La technique d'optimisation la plus simple est sans conteste celle de l'essai-erreur. Il suffit de faire varier quelques paramètres et de conserver le candidat dès que celui-ci, tout en respectant les contraintes du problème, s'avère meilleur que le modèle le plus performant actuellement disponible. Avec un peu d'habitude, d'intuition ou de chance, une amélioration peut être obtenue. Mais s'agit-il vraiment d'un

optimum ?

Partie intégrante des mathématiques appliquées, l'optimisation se veut de résoudre des problèmes scientifiques et industriels. L'optimisation est une méthode (ensemble de méthodes) qui permet d'obtenir le meilleur résultat approché d'un problème de recherche de minimum d'effort à fournir pour une machine (par exemple) ou le besoin d'obtenir un bénéfice maximal dans une gestion de production ou dans la gestion d'un portefeuille. " optimiser c'est donner les meilleures conditions de fonctionnement, de rendement " (le Petit Robert).

D'un point de vue mathématique, l'optimisation consiste à rechercher le minimum ou le maximum d'une fonction avec ou sans contraintes, Ces dites fonctions peuvent être linéaires ou non linéaires.

Nous devons, en grande partie, l'existence de l'optimisation à Taylor (1685-1731), Newton (1643-1727), Lagrange (1736-1813) et Cauchy (1789-1857) qui ont élaboré les bases des développements limités.

Cauchy ([7, 1847]) fût le premier à mettre en oeuvre une méthode d'optimisation, méthode du pas de descente, pour la résolution de problèmes sans contrainte. Depuis ces considérables contributions, il y a eu peu d'avancées. Il faut dire qu'à l'époque les mathématiciens et les ingénieurs calculaient à la main.

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

De nombreuses contributions apparaissent ensuite dans les années soixante : G.Zontendijk ([54, 1960]), C.W.Carroll ([8, 1961]), P.Wolfe ([46, 1961]), R. Fletcher et M.J.D.Powell ([20, 1963]), C. Reeves ([22, 1964]), A.A.Goldstein ([26, 1965]) et A.V.Fiacco et G.P.McCormick ([23, 1968]) pour la programmation non linéaire ainsi que E. Polak et G. Ribière ([32, 1969]), B.T.Polyak ([33, 1969]) et J.F.Price ([29, 1969]).

Les méthodes à région de confiance concurrence les méthodes de descente. Le principe des méthodes à région de confiance est de remplacer le problème d'optimisation initial par une suite de sous-problèmes d'optimisation, plus simple à résoudre. Dans chaque sous-problème, la fonction objectif f est remplacée par une fonction modèle Ψ , à un itéré courant x . La région de confiance est définie comme la région à l'intérieur de laquelle la confiance est donnée à la fonction modèle quant à sa qualité à fournir une bonne approximation de la fonction objectif. Si la décroissance de la fonction f ,

évaluée entre le point courant x et le point \hat{S} qui minimise la fonction modèle Ψ , n'est pas jugée suffisante, le rayon de la région de confiance est diminué et en général, la direction de recherche est modifiée.

Une description complète des méthodes à région de confiance peut être trouvée dans Conn et al.[9]et dans Nocedal et Wright [40] lesquels ont proposé un cadre général de procédure d'optimisation couplant réductions de modèle et méthodes à régions de confiance.

Le but de ce travail est la résolution du problème d'optimisation non-linéaire suivant :

$$(P) \quad \left\{ \begin{array}{l} \min_{x \in \mathbb{R}^n} f(x) \end{array} \right.$$

où la fonction objectif $f : \mathbb{R}^n \mapsto \mathbb{R}$ est supposée être suffisamment régulière.

Tous les théorèmes de convergence des algorithmes pour résoudre les problèmes de type (P) imposent des hypothèses sur le gradient et le hessien de la fonction objectif f .

Les techniques plus sophistiquées, dont celles que présentent ce travail, ont recours à une formulation mathématique.

1.1 Formulation Mathématique.

La formulation mathématique d'un problème d'optimisation doit en identifier l'objectif, les variables et les contraintes. Cette formulation est la plus importante et souvent la plus difficile des étapes. De cette formulation, le présent travail ne parle pas : nous partirons d'une formulation mathématique générale.

La première étape consiste à identifier les variables d'un problème. Les variables sont les éléments sur lesquels nous avons la possibilité d'agir directement pour en modifier les valeurs. Les variables sont généralement amalgamées dans un vecteur x .

L'étape suivante consiste à définir la fonction objectif $f(x)$. La fonction objectif mesure la quantité à minimiser ou à maximiser. Sauf mention contraire, dans le reste de ce travail, nous nous concentrerons sur les problèmes de minimisation de $f(x)$, un problème de maximisation pouvant être facilement converti en cherchant à minimiser l'opposé de la fonction $f(x)$.

Il convient enfin d'identifier les fonctions de contraintes $c_j(x)$. Les contraintes

peuvent être de deux natures : nous avons, d'une part, les contraintes d'égalité et d'autre part les contraintes d'inégalité.

Plus formellement, le problème général que nous traitons s'écrit

$$\begin{cases} \min f(x) \\ \text{s.c. } c_j(x) = 0 \text{ pour } j \in E, \\ c_j(x) \leq 0 \text{ pour } j \in I \end{cases} \quad (1.1)$$

où E et I sont, respectivement, les ensembles disjoints des indices des contraintes d'égalité et d'inégalité. Les fonctions f et c_j sont supposées continûment dérivables.

On désigne par Ω , l'ensemble admissible, le sous-ensemble de \mathbb{R}^n où les contraintes sont satisfaites, i.e.

$$\Omega = \{x \in \mathbb{R}^n \mid c_j(x) = 0 \text{ pour } j \in E \text{ et } c_j(x) \leq 0 \text{ pour } j \in I\} \quad (1.2)$$

Pour certains problèmes, les variables n'ont un sens qu'à la condition d'être choisies dans un ensemble de valeurs discrètes. Il s'agit là d'optimisation discrète, à opposer à l'optimisation continue. Le présent travail ne traitera pas de l'optimisation discrète.

1.2 Conditions d'optimalité.

Les solutions les plus intéressantes d'un problème d'optimisation sont les minima globaux, c'est à dire l'ensemble des arguments x^* pour lequel la fonction f atteint sa plus petite valeur dans l'ensemble admissible, i.e.

$$f(x^*) \leq f(x) \forall x \in \Omega \quad (1.3)$$

Le minimum global est généralement difficile à trouver car les valeurs des fonctions $f(x)$ et $c_j(x)$ ne sont généralement pas connues en tous les points de l'ensemble admissible. La plupart des algorithmes se contentent donc de trouver un minimum local.

On dit que la fonction objectif présente un minimum local en un point x^* si la fonction $f(x^*)$ est la plus petite valeur dans un voisinage de ce point (voir figure 1.1), i.e. x^* est un minimum local s'il existe un voisinage $V(x^*)$ tel que

$$f(x^*) \leq f(x) \forall x \in \Omega \cap V(x^*). \quad (1.4)$$

où $V(x)$ est un voisinage de x (est un ensemble contenant un ouvert contenant x).

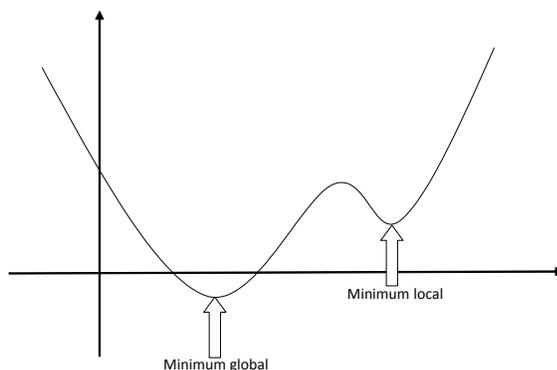


Figure 1.1

Le minimum local est dit strict si l'inégalité peut être remplacée par une inégalité stricte.

Heureusement, cette définition ne constitue pas la seule manière de déterminer si un point x^* est oui ou non un minimum local. Il serait en effet impossible d'explorer tout les points du voisinage de x^* pour être sûr que la fonction objectif n'est pas supérieure à $f(x^*)$. Lorsque le problème est suffisamment régulier, des moyens beaucoup plus efficaces existent. Par exemple, lorsque la fonction f est deux fois continûment dérivable, les critères d'optimalité sont bien connus (voir, par exemple, Fletcher [22]). Dans un problème non-contraint, il suffit que le gradient de f au point x^* soit nul

$$\nabla_x f(x^*) = 0 \tag{1.5}$$

et que son Hessien (aussi appelé matrice hessienne) $H(x^*)$ soit définie positive pour que x^* soit un minimum local de f .

La littérature fait aussi grand usage des conditions nécessaires d'optimalité. La condition du premier ordre est tout simplement

$$\nabla_x f(x^*) = 0 \tag{1.6}$$

Les points répondant à cette propriété sont appelés points stationnaires ou points critiques du premier ordre. La condition nécessaire du second ordre se formule

$$H(x^*) \text{ semi-définie positive} \quad (1.7)$$

Pour les problèmes contraints, l'établissement d'une condition nécessaire demande de faire quelques hypothèses sur la régularité des contraintes. Ces conditions de régularité sont généralement connues sous le nom de qualification des contraintes. Il existe plusieurs hypothèses de qualification des contraintes ; par exemple la plus simple et la plus connue est celle dû à Nocedal et Wright [40] : il s'agit de la condition d'indépendance linéaire de qualification des contraintes. Pour satisfaire cette condition, il suffit que les gradients $\nabla_x c_j(x^*)$ des contraintes actives (Une contrainte $c_j(x)$ est dite active au point x^* si $c_j(x^*) = 0$). au point x^* soient linéairement indépendants. Dans ce cas, nous avons les célèbres conditions nécessaires du premier ordre de Karush–Kuhn–Tucker [31]

$$\left\{ \begin{array}{l} \nabla_x f(x^*) + \sum_{j \in E \cup I} \lambda_j^* \nabla_x c_j(x^*) = 0 \\ c_j(x^*) = 0 \text{ pour } j \in E, \\ c_j(x^*) \leq 0 \text{ pour } j \in I, \\ \lambda_j^* \geq 0 \text{ pour } j \in I, \\ \lambda_j^* c_j(x^*) = 0 \text{ pour } j \in E \cup I. \end{array} \right. \quad (1.8)$$

Les paramètres λ_j^* sont les multiplicateurs de Lagrange du problème.

En règle générale, il n'est pas possible de déterminer si un minimum local est le minimum global du problème.

1.3 Taux de convergence.

Les différentes techniques d'optimisation se distinguent bien entendu par leurs performances. Une des mesures de la performance d'un algorithme est le taux de convergence ; il s'agit d'une mesure de la vitesse de convergence. Les deux taux de convergence les plus utilisés sont définis comme suit (voir, par exemple, Nocedal et Wright [40]).

Soit une suite d'itérés $\{x^{(k)}\}$ de R^n qui converge vers x^* . On dit que le taux de convergence est linéaire s'il existe une constante $r \in]0, 1[$ telle que

$$\frac{\|x^{(k+1)} - x^*\|}{\|x^{(k)} - x^*\|} \leq r \text{ pour } k \text{ suffisamment grand.} \quad (1.9)$$

Cela signifie que l'écart entre l'itéré et la solution décroît au moins, à chaque itération, d'un facteur constant r . La convergence est superlinéaire si

$$\lim_{k \rightarrow \infty} \frac{\|x^{(k+1)} - x^*\|}{\|x^{(k)} - x^*\|} = 0. \quad (1.10)$$

Une convergence quadratique se définit quant à elle par

$$\frac{\|x^{(k+1)} - x^*\|}{\|x^{(k)} - x^*\|^2} \leq M \text{ pour } k \text{ suffisamment grand,} \quad (1.11)$$

où M est une constante positive. Cela signifie que l'écart avec la solution décroît quadratiquement au fil des itérations.

1.4 Objet et apport de ce travail.

Ce travail porte sur une classe particulière de techniques d'optimisation : il s'intéresse aux problèmes non-linéaires, contraints et non-contraints. Les méthodes envisagées sont itératives : elles construisent, au fur et à mesure des itérations, une suite $\{x^{(k)}\}$ qui du moins nous l'espérons convergera vers un minimum local x^* . Un point de départ $x^{(0)}$ pour les paramètres à optimiser est arbitrairement choisi et la fonction objectif $f(x^{(0)})$ ainsi que les fonctions de contraintes $c_j(x^{(0)})$ sont évaluées. Pour chaque itéré $x^{(k)}$ ces mêmes fonctions seront également évaluées. Notre travail s'inscrit dans le paradigme où le temps nécessaire pour évaluer celles-ci s'avèrent extrêmement long par rapport aux calculs nécessaires à l'algorithme lui-même. Plus formellement, cette hypothèse peut s'exprimer comme suit.

Hypothèse 1.1. *Le temps nécessaire au calcul de l'itéré $x^{(k+1)}$ à partir de la valeur de la fonction objectif $f(x^{(k)})$, des fonctions de contraintes $c_j(x^{(k)})$ et/ou de leurs dérivées au point $x^{(k)}$ est négligeable par rapport au temps nécessaire à l'évaluation de ces dernières.*

Ceci signifie que notre priorité, en terme de performance, est à la diminution du nombre total d'itérations engendrées par l'algorithme. Ce paradigme présente l'avantage que la mesure de performance ne dépend ni du problème envisagé ni de la puissance de calcul utilisée. Cette hypothèse est réaliste et utile : pensons par exemple aux modèles numériques complexes utilisés en météorologie ou en aéronautique. Comme toutes les hypothèses, celle-ci à ses limites : en particulier lorsque nous avons affaire à des problèmes de grande

taille puisque le nombre d'opérations de calcul causé par l'algorithme lui-même croît généralement exponentiellement avec le nombre de paramètres de la fonction à optimiser.

La première partie du travail, incluant le présent chapitre, fait office d'introduction générale aux méthodes par régions de confiance et à l'utilisation d'approximations quadratiques. Le deuxième chapitre porte sur différentes techniques de globalisation celles qui utilisent les recherches linéaire et les méthodes du gradient conjugué.

Dans la deuxième partie, on fait une introduction générale aux méthodes par régions de confiance et à l'utilisation d'approximations quadratiques. En quelques mots, elle consiste à considérer que les fonctions $f(x)$ et $cj(x)$ peuvent être remplacées par une approximation locale, plus facile à utiliser, dans une certaine zone de confiance autour de l'itéré courant $x^{(k)}$. De manière similaire aux méthodes de recherches linéaires, la détermination de pas optimaux n'est pas une condition nécessaire pour obtenir une convergence globale. Sous certaines conditions, une bonne approximation de ces pas peut être acceptable. Il suffit de déterminer une solution approchée, à l'intérieur de la région de confiance, qui produise une réduction suffisante de la fonction modèle. Cette réduction peut être obtenue par la méthode du point de Cauchy. Comme nous venons de le voir, le point de Cauchy permet de valider ou non une solution approchée du sous problème à région de confiance. La convergence des méthodes à région de confiance est prouvée lorsque le pas produit une décroissance suffisante de la fonction modèle. A chaque itération, la réduction de la fonction modèle en un point doit au moins être égale à un multiple de la décroissance obtenue au point de Cauchy. La condition de la fraction de décroissance de Cauchy (Fraction of Cauchy Decrease, fcd) doit alors être vérifiée ainsi que des différents théorèmes de convergence globale découlent de la condition de décroissance suffisante sont abordés au quatrième chapitre.

Le cinquième chapitre comporte le travail original, présente un algorithme complet et efficace de résolution d'un problème non linéaire approximé par un problème pénalisé. Cette méthode originale se base sur une fonction de pénalité laquelle est présentée dans le troisième chapitre. Le principe du nouvel algorithme est de construire une stratégie de recherche d'incrément d_k du point courant x_k à l'intérieur de la région de confiance qui produit une réduction suffisante de la fonction modèle, cette réduction est obtenue par la méthode du point de cauchy. De plus dans ce chapitre, on va tester quelques fonctions non linéaires et on donne le temps nécessaire de la convergence.

CHAPITRE 1. POSITION DU PROBLÈME

Enfin, on termine par une conclusion et on propose un problème ouvert.

Chapitre 2

Méthodes de globalisation en optimisation

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

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

Après avoir décrit comment fonctionne un algorithme à directions de descente, nous donnons quelques exemples d'algorithmes de ce type. Nous décrivons ensuite les principales règles de recherche linéaire.

2.1 Principe général des Méthodes à directions de descente

Considérons le problème d'optimisation sans contrainte

$$\min_{x \in \mathbb{R}^n} f(x) \tag{2.1}$$

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

On note aussi $\nabla f(x)$ et $\nabla^2 f(x)$ le gradient et le hessien de f en x pour ce produit scalaire. On s'intéresse ici à une classe d'algorithmes qui sont fondés sur la notion de direction de descente.

2.1.1 Direction de descente

Définition 2.1.1 Soit $f : \mathbb{R}^n \rightarrow \mathbb{R}$, d un vecteur non nul de \mathbb{R}^n est dit une direction de descente, s'il existe un réel $\delta > 0$ tel que pour tout λ dans l'intervalle $]0, \delta[$, on a

$$f(x + \lambda d) < f(x) \quad (2.2)$$

Théorème 2.1.1 Si f est différentiable en un point $x \in \mathbb{R}^n$ et $d \in \mathbb{R}^n$ telle que

$$\nabla^T f(x) d < 0 \quad (2.3)$$

Alors d est une direction de descente en x

Remarque 2.1.1 d fait avec l'opposé du gradient $-\nabla f(x)$ un angle θ strictement plus petit que 90° :

$$\theta := \arccos \frac{-\nabla^T f(x) \cdot d}{\|\nabla f(x)\| \|d\|} \in \left[0, \frac{\pi}{2}\right]. \quad (2.4)$$

Remarque 2.1.2 L'ensemble des directions de descente de f en x ,

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

Forme un demi-espace ouvert de \mathbb{R}^n (illustration à la figure 2.1).

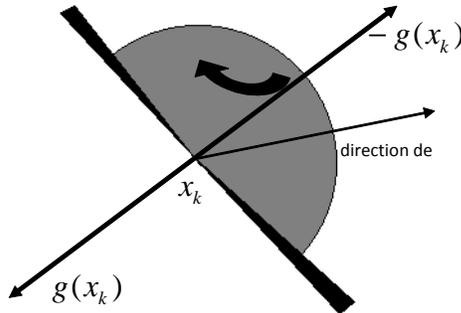


Figure 2.1

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

2.1.2 Description des méthodes à directions de descentes

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

Elles construisent la suite des itérés $\{x_k\}_{k \geq 1}$, approchant une solution x_k de (2.1) par la récurrence :

$$x_{k+1} = x_k + \alpha_k d_k, \text{ pour } k \geq 1 \quad (2.5)$$

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

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

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

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

Décrivons cette classe d'algorithmes de manière précise.

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

Etape 0 : (initialisation)

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

Etape1 :

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

Etape2 :

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

Etape3 :

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

Etape4 :

Si la recherche linéaire réussit : $x_{k+1} = x_k + \alpha_k d_k$;
remplacer k par $k + 1$ et aller à l'étape 1. □

2.1.3 Exemples de méthodes à directions de descente

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

$$x_{k+1} = x_k + \alpha_k d_k, \quad \alpha_k \in]0, +\delta[. \quad (2.6)$$

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

$$f(x_{k+1}) = f(x_k + \alpha_k d_k) < f(x_k) \quad (2.7)$$

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

2.1.4 Exemples de choix du directions de descente

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

Bien sur $d_k = -\nabla f(x_k)$ est une direction de descente puisque

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

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

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

2.1.5 Exemples du choix de pas α_k

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

$$f(x_k + \alpha_k d_k) \leq f(x_k + \alpha d_k) : \forall \alpha \in [0, +\infty[. \quad (2.8)$$

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

2.2 Recherche linéaire

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

$$\min f(x_k + \alpha d_k), \quad \alpha \in \mathbb{R} \quad (2.9)$$

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

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

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

afin de réaliser un certain progrès en valeur de l'objective, c'est-à-dire, pour assurer que :

$$f(x_k + \alpha_k d_k) \leq f(x_k)$$

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

$$h_k(\alpha) = f(x_k + \alpha d_k). \quad (2.10)$$

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

2.2.1 Objectifs à atteindre

Il s'agit de réaliser deux objectifs

Le premier objectif

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

$$f(x_k + \alpha_k d_k) \leq f(x_k) + \text{"un terme négatif"} \quad (2.11)$$

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

L'argument est le suivant :

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

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

Le second objectif

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

Le premier objectif n'est en effet pas suffisant car l'inégalité (2.11) 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 [25].

Si on prend

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

la suite $\{x_k\}$ générée par (2.2) est de Cauchy, puisque pour $1 \leq l < k$ on a :

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

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 $\bar{x} \in \bar{B}(x_1, \varepsilon)$ et donc \bar{x} ne saurait être solution s'il n'y a pas de solution dans $\bar{B}(x_1, \varepsilon)$.

On a donc arbitrairement 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 $\{x_k + \alpha d_k / \alpha \in \mathbb{R}\} \subset \mathbb{R}^n$: comme la fonction :

$$h_k : \mathbb{R}_+ \rightarrow \mathbb{R}; \quad \alpha \longmapsto h_k(\alpha) = f(x_k + \alpha d_k) \quad (2.12)$$

2.2.2 Recherches linéaires exactes et inexactes

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

Recherches linéaires exactes

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 α_k comme solution du problème

$$\min h_k(\alpha) \quad \alpha \geq 0 \quad (2.13)$$

C'est ce que l'on appelle la règle de Cauchy et le pas déterminé par cette règle est appelé pas de Cauchy ou pas optimal. Dans certains cas, on préférera le plus petit point stationnaire de h_k qui fait décroître cette fonction :

$$\alpha_k = \inf \{ \alpha \geq 0 : h'_k(\alpha) < h'_k(0) \}. \quad (2.14)$$

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

Remarque 2.2.1 *Ces deux règles ne sont utilisées que dans des cas particuliers, par exemple lorsque h_k est quadratique.*

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

Les inconvénients des recherches linéaires exactes Pour une fonction non linéaire arbitraire,

- il peut ne pas exister de pas de Cauchy ou de Curry,
- la détermination de ces pas demande en général beaucoup de temps de calcul et ne peut de toutes façons pas être faite avec une précision infinie,
- l'efficacité supplémentaire éventuellement apportée à un algorithme par une recherche linéaire exacte ne permet pas, en général, de compenser le temps perdu à déterminer un tel pas,
- les résultats de convergence autorisent d'autres types de règles (recherche linéaire inexacte), moins gourmandes en temps de calcul.

2.2.3 Intervalle de sécurité

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

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

Cela conduit à la notion d'intervalle de sécurité.

Définition 2.2.1 *On dit que $[\alpha_g, \alpha_d]$ est un intervalle de sécurité s'il permet de classer les valeurs de α de la façon suivante :*

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

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

2.2.4 Algorithme de base

Algorithme 2.2 (Algorithme de base)

Etape 0 : (initialisation)

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

Etape 1 :

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

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

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

Etape 2 :

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

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

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

Au lieu de demander que α_k minimise, on préfère imposer des conditions moins restrictives, plus facilement vérifiées, qui permettent toute fois

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

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

$$\nabla^T f(x_k) d_k < 0 \quad (2.15)$$

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

L'objectif de cette section consiste à présenter les principaux tests. D'abord présentons le schéma d'une recherche linéaire inexacte.

Schéma des recherches linéaires inexactes Elles reviennent à déterminer, un intervalle $[\alpha_g, \alpha_d]$, où $\alpha^* \in [\alpha_g, \alpha_d]$, dans lequel :

$$h_k(\alpha_k) < h_k(0) \quad (f(x_k + \alpha_k d_k) < f(x_k)) \quad (2.16)$$

Le schéma de l'algorithme est donc :

Algorithme 2.3 (Schéma général des recherches linéaires inexactes)

Étape 0 : (initialisation)

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

Étape 1 :

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

si α_k est trop petit (suivant un certain critère) : nouvel intervalle : [

$\alpha_{g,k+1} = \alpha_k$, $\alpha_{d,k+1} = \alpha_d$]

et aller à l'étape 2.

si α_k est trop grand (suivant un certain critère) : nouvel intervalle :
 $[\alpha_{g,k+1} = \alpha_g, \alpha_{d,k+1} = \alpha_k]$
 et aller à l'étape 2.

Etape 2 :

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

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

2.2.6 La règle d'Armijo

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

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

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

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

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

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

$$f(x_k + \alpha_k d_k) \leq f(x_k) + \rho \alpha_k \nabla^T f(x_k) d_k \tag{2.17}$$

Elle est de la forme (2.11), car ρ devra être choisi dans $]0, 1[$.

On voit bien à la figure 2.2 ce que signifie cette condition.

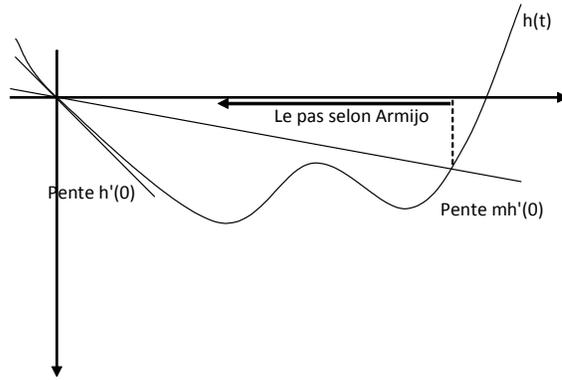


Figure 2.2

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

$$\alpha \mapsto h_k(0) + \rho h'_k(0)\alpha \quad \text{autrement dit} \quad \alpha \mapsto f(x_k) + \rho \alpha \nabla^T f(x_k) d_k$$

Test d'Armijo :

◇ Si

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

autrement dit $f(x_k + \alpha d_k) \leq f(x_k) + \rho \alpha \nabla^T f(x_k) d_k$

alors α convient.

◇ Si

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

autrement dit $f(x_k + \alpha d_k) > f(x_k) + \rho \alpha \nabla^T f(x_k) d_k$

alors α est trop grand. \square

Algorithme 2.4 (Règle d'Armijo)

Etape 0 : (initialisation)

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

1.

Etape 1 :

si $h_k(\alpha_k) \leq h_k(0) + \rho h'_k(0)\alpha_k$: STOP ($\alpha^* = \alpha_k$).
 si $h_k(\alpha_k) > h_k(0) + \rho h'_k(0)\alpha_k$, alors
 $\alpha_{d,k+1} = \alpha_d, \alpha_{g,k+1} = \alpha_k$ et aller à l'étape 2.

Etape 2 :

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

Remarque 2.2.2 *En pratique, la constante ρ est prise très petite, de manière à satisfaire (2.17) le plus facilement possible. Dans la pratique $\rho = 10^{-4}$.*

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

Remarque 2.2.3 *Il est clair que l'inégalité (2.17) est toujours vérifiée si $\alpha_k > 0$ est suffisamment petit.*

En effet dans le cas contraire, on aurait une suite de pas strictement positifs $\{\alpha_{k,i}\}_{i \geq 1}$ convergeant vers 0 lorsque $i \rightarrow \infty$ et tels que (2.17) n'ait pas lieu pour $\alpha_k = \alpha_{k,i}$.

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

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

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

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

Théorème 2.2.1 *Si $h_k : \mathbb{R}_+ \rightarrow \mathbb{R}$; définie par $h_k(\alpha) = f(x_k + \alpha d_k)$ est continue et bornée inférieurement, si d_k est une direction de descente en x_k ($h'_k(0) < 0$) et si $\rho \in]0, 1[$, alors l'ensemble des pas vérifiant la règle d'Armijo est non vide.*

Démonstration.

On a

$$\begin{aligned} h_k(\alpha) &= f(x_k + \alpha d_k) \\ \psi_\rho(\alpha) &= f(x_k) + \rho \alpha h'_k(0) \end{aligned}$$

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

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

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

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

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

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

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

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

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

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

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

Ce qui achève la démonstration. \square

Décrivons maintenant la règle de Goldstein&Price.

2.2.7 La règle de Goldstein&Price.

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

h_k est bien strictement décroissante mais $\{x_k\}_{k \geq 0}$ ne converge pas vers le minimum zéro mais vers 1 (voir figure 2.3).

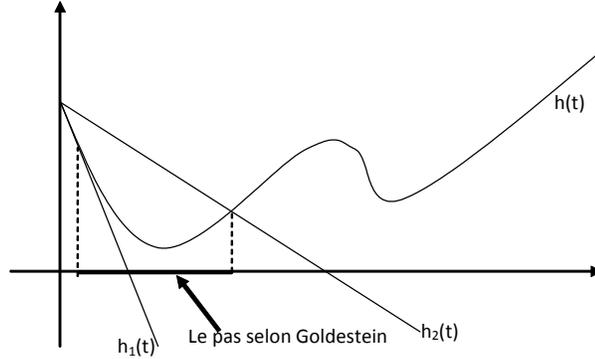


Figure 2.3

Alors que la condition d'Armijo est satisfaite quelque soit vérifiant.

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

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

$$f(x_k + \alpha d_k) \leq f(x_k) + \rho \alpha \nabla^T f(x_k) d_k \quad (2.18)$$

$$f(x_k + \alpha d_k) \geq f(x_k) + \delta \alpha \nabla^T f(x_k) d_k \quad (2.19)$$

autrement dit :

$$h_k(\alpha) \leq h_k(0) + \rho h'_k(0) \alpha \quad (2.20)$$

$$h_k(\alpha) \geq h_k(0) + \delta h'_k(0) \alpha \quad (2.21)$$

Test de Goldstein&Price :

◇ Si

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

$$\text{autrement dit } f(x_k) + \delta \alpha \nabla^T f(x_k) d_k \leq f(x_k + \alpha d_k) \leq f(x_k) + \rho \alpha \nabla^T f(x_k) d_k$$

alors α convient.

◇ Si

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

autrement dit $f(x_k + \alpha d_k) > f(x_k) + \rho \alpha \nabla^T f(x_k) d_k$

alors α est trop grand.

◇ Si

$$h_k(\alpha) < h_k(0) + \delta h'_k(0)\alpha$$

autrement dit $f(x_k + \alpha d_k) < f(x_k) + \delta \alpha \nabla^T f(x_k) d_k$

alors α est trop petit. \square

Algorithme 2.5 (Règle de Goldstein&Price)

Etape 0 : (initialisation)

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

Etape 1 :

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

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

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

si $h_k(\alpha_k) < h_k(0) + \delta h'_k(0)\alpha_k$, alors

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

Etape 2 :

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

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

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

Théorème 2.2.2 Si $h_k : \mathbb{R}_+ \rightarrow \mathbb{R}$; définie par (2.10) $h_k(\alpha) = f(x_k + \alpha d_k)$ est continue et bornée inférieurement, si d_k est une direction de descente en x_k et si $\rho \in]0, 1[$, $\delta \in]\rho, 1[$, alors l'ensemble des pas vérifiant la règle de Goldstein&Price (2.18)-(2.19) est non vide.

Démonstration.[25]

On a

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

et comme $\psi_\delta(\alpha) < \psi_\rho(\alpha)$ pour $\alpha > 0$, forcément $\tilde{\alpha} < \bar{\alpha}$ et $\alpha = \bar{\alpha}$ satisfait (2.18)-(2.19)

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

Ce qu'il fallait démontrer. \square

2.2.8 La règle de Wolfe

Les conditions (2.18)-(2.19) "règle de Goldstein&Price" peuvent exclure un minimum ce qui est peut être un inconvénient.

Les conditions de Wolfe n'ont pas cet inconvénient.

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

$$f(x_k + \alpha d_k) \leq f(x_k) + \rho \alpha \nabla^T f(x_k) d_k \quad (2.22)$$

$$\nabla^T f(x_k + \alpha d_k) d_k \geq \sigma \nabla^T f(x_k) d_k \quad (2.23)$$

autrement dit :

$$h_k(\alpha) \leq h_k(0) + \rho h'_k(0) \alpha \quad (2.24)$$

$$h'_k(\alpha) \geq \sigma h'_k(0) \quad (2.25)$$

Test de Wolfe :

◇ Si

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

alors α convient.

◇ Si

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

alors α est trop grand.

◇ Si

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

alors α est trop petit. \square

On voit bien à la figure 2.4 ce que signifie cette condition.

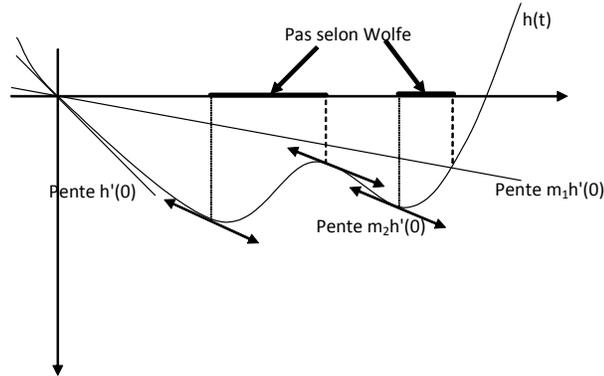


Figure 2.4

Algorithme 2.6 (Règle de Wolfe)

Etape 0 : (initialisation)

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

Etape 1 :

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

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

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

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

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

Etape 2 :

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

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

Remarque 2.2.4 La règle de Wolfe fait appel au calcul de h'_k , elle est donc en théorie plus coûteuse que la règle de Goldstein & Price.

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.

Conditions de Wolfe fortes

Pour certains algorithmes (par exemple le gradient conjugué non linéaire) il est parfois nécessaire d'avoir une condition plus restrictive que (2.23).

Pour cela la deuxième condition (2.23) est remplacée par

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

On aura donc les conditions de Wolfe fortes :

$$f(x_k + \alpha d_k) \leq f(x_k) + \rho \alpha \nabla^T f(x_k) d_k \quad (2.26)$$

$$|\nabla^T f(x_k + \alpha d_k) d_k| \leq -\sigma \nabla^T f(x_k) d_k \quad (2.27)$$

autrement dit :

$$h_k(\alpha) \leq h_k(0) + \rho h'_k(0) \alpha \quad (2.26')$$

$$|h'_k(\alpha)| \leq -\sigma h'_k(0) \quad (2.27')$$

où $0 < \rho < \sigma < 1$.

Remarque 2.2.5 *La seconde condition (2.23) ou condition de courbure interdit le choix de pas trop petit pouvant entraîner une convergence lente ou prématurée.*

Remarque 2.2.6 *On voit bien que les conditions de Wolfe fortes impliquent les conditions de Wolfe faibles. Effectivement (2.26) est équivalente à (2.22), tandis que (2.27)*

$$\begin{aligned} |\nabla^T f(x_k + \alpha_k d_k) d_k| &\leq -\sigma \nabla^T f(x_k) d_k \\ \Leftrightarrow \sigma \nabla^T f(x_k) d_k &\leq \nabla^T f(x_k + \alpha_k d_k) d_k \leq -\sigma \nabla^T f(x_k) d_k \\ \Rightarrow \sigma \nabla^T f(x_k) d_k &\leq \nabla^T f(x_k + \alpha_k d_k) d_k \quad (2.23) \quad \square \end{aligned}$$

La règle de Wolfe relaxée

Cette règle consiste à choisir le pas satisfaisant aux conditions :

$$f(x_k + \alpha d_k) \leq f(x_k) + \rho \alpha \nabla^T f(x_k) d_k \quad (2.28)$$

$$\sigma_1 \nabla^T f(x_k) d_k \leq \nabla^T f(x_k + \alpha_k d_k) d_k \leq -\sigma_2 \nabla^T f(x_k) d_k \quad (2.29)$$

autrement dit :

$$h_k(\alpha) \leq h_k(0) + \rho h'_k(0) \alpha \quad (2.28')$$

$$\sigma_1 h'_k(0) \leq h'_k(\alpha) \leq -\sigma_2 h'_k(0) \quad (2.29')$$

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

Remarque 2.2.7 On voit bien que les conditions de Wolfe relaxée impliquent les conditions de Wolfe fortes. Effectivement (2.28) est équivalente à (2.22), tandis que pour le cas particulier $\sigma_1 = \sigma_2 = \sigma$, (2.29) est équivalente à (2.23). En effet :

$$\begin{aligned} \sigma_1 \nabla^T f(x_k) d_k &\leq \nabla^T f(x_k + \alpha_k d_k) d_k \leq -\sigma_2 \nabla^T f(x_k) d_k \\ \Rightarrow \sigma \nabla^T f(x_k) d_k &\leq \nabla^T f(x_k + \alpha_k d_k) d_k \leq -\sigma \nabla^T f(x_k) d_k \\ \Rightarrow |\nabla^T f(x_k + \alpha_k d_k) d_k| &\leq -\sigma \nabla^T f(x_k) d_k \quad (2.27) \quad \square \end{aligned}$$

Remarque 2.2.8 Les conditions de Wolfe relaxée impliquent les conditions de Wolfe faibles. Effectivement (2.28) est équivalente à (2.22), tandis que pour le cas particulier $\sigma_1 = \sigma$ et $\sigma_2 = +\infty$, (2.29) est équivalente à (2.23). En effet :

$$\begin{aligned} \sigma_1 \nabla^T f(x_k) d_k &\leq \nabla^T f(x_k + \alpha_k d_k) d_k \leq -\sigma_2 \nabla^T f(x_k) d_k \\ \Rightarrow \sigma \nabla^T f(x_k) d_k &\leq \nabla^T f(x_k + \alpha_k d_k) d_k \quad (2.23) \quad \square \end{aligned}$$

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

Théorème 2.2.3 Si $h_k : \mathbb{R}_+ \rightarrow \mathbb{R}$; définie par (2.10) $h_k(\alpha) = f(x_k + \alpha d_k)$ est dérivable et bornée inférieurement, si d_k est une direction de descente en x_k et si $\rho \in]0, 1[$, $\sigma \in]\rho, 1[$, alors l'ensemble des pas vérifiant la règle de Wolfe (faible) (2.22)-(2.23) est non vide.

Démonstration.

On a

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

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

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

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

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

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

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

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

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

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

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

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

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

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

car $0 < \rho < \sigma < 1$ et $\nabla^T f(x_k) d_k < 0$.

Donc $\hat{\alpha}$ satisfait (2.23).

D'autre part, $\alpha = \hat{\alpha}$ satisfait (2.22), en effet, $\hat{\alpha}$ satisfait (*) $h_k(\hat{\alpha}) < \psi_\rho(\hat{\alpha})$ n'est autre que :

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

Ce qu'il fallait démontrer.

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

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

Algorithme 2.7 (Règle de Fletcher-Lemaréchal)

Etape0 : (initialisation)

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

Etape 1 :

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

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

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

1.3.1 si $h'_k(\alpha) \geq \sigma h'_k(0) : \text{STOP}(\alpha^* = \alpha_k)$.

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

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

Etape 2 :

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

si non déterminer $\alpha_{k+1} \in](1 - v)\alpha_{g,k+1} + v\alpha_{d,k+1}, v\alpha_{g,k+1} + (1 - v)\alpha_{d,k+1}[$

poser $k = k + 1$ et aller à l'étape 1. \square

Théorème 2.2.4 Si $h_k : \mathbb{R}_+ \rightarrow \mathbb{R}$; définie par (2.10) $h_k(\alpha) = f(x_k + \alpha d_k)$ est dérivable et bornée inférieurement, si d_k est une direction de descente en x_k et si $\rho \in]0, 1[, \sigma \in]\rho, 1[$, alors l'algorithme de Fletcher-Lemaréchal trouve le pas de Wolfe(2.22)-(2.23) en un nombre fini d'étapes. \square

2.2.9 Convergence des méthodes à directions de descente

Condition de Zoutendijk

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

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

$$f(x_{k+1}) \leq f(x_k) - C \|\nabla f(x_k)\|^2 \cos^2 \theta_k \quad (2.30)$$

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

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

Voici comment on se sert de la condition de Zoutendijk.

Proposition 2.2.1 *Si la suite $\{x_k\}$ générée par un algorithme d'optimisation vérifie la condition de Zoutendijk (2.30) et si la suite $\{f(x_k)\}$ est minorée, alors*

$$\sum_{k \geq 1} \|\nabla f(x_k)\|^2 \cos^2 \theta_k < \infty \quad (2.31)$$

Démonstration. [25]

En sommant les inégalités (2.30), on a

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

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

Les deux propositions suivantes précisent les circonstances dans lesquelles la condition de Zoutendijk (2.30) est vérifiée avec les règles d'Armijo et de Wolfe.

Proposition 2.2.2 *Soit $f : \mathbb{R}^n \rightarrow \mathbb{R}$ une fonction contiuement différentiable dans un voisinage de $\Gamma = \{x \in \mathbb{R}^n : f(x) \leq f(x_1)\}$.*

On considère un algorithme à directions de descente d_k , qui génère une suite $\{x_k\}$ en utilisant la recherche linéaire d'Armijo(2.17) avec $\alpha_1 > 0$.

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

$$f(x_{k+1}) \leq f(x_k) - C \nabla^T f(x_k) d_k \quad (2.32)$$

ou

$$f(x_{k+1}) \leq f(x_k) - C \|\nabla f(x_k)\|^2 \cos^2 \theta_k \quad (2.33)$$

est vérifiée.

Démonstration.[25]

Si le pas $\alpha_k = \alpha_1$ est accepté, on a (2.30), car α_1 est uniformément positif.

Dans le cas contraire, (2.15) n'est pas vérifiée avec un pas $\alpha'_k \leq \frac{\alpha_k}{\tau}$, c'est-à-dire $f(x_k + \alpha'_k d_k) > f(x_k) + \rho \alpha'_k \nabla^T f(x_k) d_k$

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

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

CHAPITRE 2. MÉTHODES DE GLOBALISATION EN OPTIMISATION

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

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

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

or

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

d'où

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

ce qui permet de minorer $\alpha' \|d_k\|$ et donc aussi $\alpha \|d_k\|$ par une constante fois $\|\nabla^T f(x_k)\| \|d_k\|$. Cette minoration et l'expression suivante de (2.15)

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

conduit à(2.30). \square

Proposition 2.2.3 *Soit $f : \mathbb{R}^n \rightarrow \mathbb{R}$ une fonction contiuement différentiable dans un voisinage de $\Gamma = \{x \in \mathbb{R}^n : f(x) \leq f(x_1)\}$.*

On considère un algorithme à directions de descente d_k , qui génère une suite $\{x_k\}$ en utilisant la recherche linéaire de Wolfe (2.22)-(2.23).

Alors il existe une constante $C > 0$ telle que, pour tout $k \geq 1$, la condition de Zoutendijk (2.30) est vérifiée.

Démonstration. [25]

D'après (2.22)

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

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

et du fait que f est continuellement différentiable :

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

en utilisant (2.22), on aura :

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

On en déduit (2.30). \square

2.3 Méthode du gradient conjugué.

Les principaux résultats de convergence pour les différentes variantes du gradient conjugué ont été obtenus en utilisant des recherches linéaires inexactes de Wolfe fortes. La question naturelle qui se pose est la suivante :

Peut-on obtenir des résultats de convergence avec ces variantes du Gradient Conjugué ou d'autres en utilisant les méthodes à régions de confiance ?

La réponse a été donnée positivement dans des travaux récents de Y.H.DAI et Y. YUAN ([14], [16]), qui ont proposé des modifications du gradient conjugué de la forme (2.8),(2.12),(2.17),(2.19)

CHAPITRE 2. MÉTHODES DE GLOBALISATION EN OPTIMISATION

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

$$\min \{f(x) : x \in \mathbb{R}^n\} \quad (2.34)$$

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 :

$$x_{k+1} = x_k + \alpha_k d_k \quad (2.35)$$

où les directions d_k vérifient :

$$d_k = \begin{cases} -g_0 & \text{si } k = 0 \\ -g_k + \beta_k d_{k-1} & \text{si } k \geq 1 \end{cases} \quad (2.36)$$

Le coefficient β_k détermine la méthode du gradient conjugué, le pas $\alpha_k \in \mathbb{R}^+$ étant déterminé par une recherche linéaire inexacte.

Dans ce travail on s'intéresse à la méthode du gradient conjugué version Dai-Yuan dont la description est donnée dans le paragraphe suivant.

2.4 Méthode de Dai-Yuan

Cette méthode a été découverte par Dai et Yuan en 1999,

2.4.1 Description de la méthode

On suppose que la direction recherche d_k est une direction de descente, c'est à dire

$$d_k^T g_k < 0$$

Maintenant nous avons besoin de trouver un coefficient β_k qui définit une direction de descente d_{k+1} . Cela exige que :

$$-\|g_{k+1}\|^2 + \beta_k g_{k+1}^T d_k < 0$$

Supposons que : $\beta_k > 0$ et

$$r_k = \|g_{k+1}\|^2 / \beta_k \iff r_k > g_{k+1}^T d_k,$$

puisque

$$r_k = d_k^T y_k$$

on a notre nouvelle formule

$$\beta_k = \frac{\|g_{k+1}\|^2}{d_k^T y_k} \quad (2.37)$$

De (2.36) et (2.37), on a

$$g_{k+1}^T d_{k+1} = \frac{\|g_{k+1}\|^2}{d_k^T y_k} g_k^T d_k = \beta_k g_k^T d_k. \quad (2.38)$$

Ce qui donne

$$\beta_k^{DY} = \frac{g_{k+1}^T d_{k+1}}{g_k^T d_k} \quad (2.39)$$

Un des derniers résultats de convergence de la méthode du gradient conjugué de Dai-Yuan a été prouvé avec la recherche linéaire de wolfe faible.

Proposition 2.4.1 (i) La fonction f est bornée inférieurement dans \mathbb{R}^n

(ii) Dans un voisinage N de $L = \{x \in \mathbb{R}^n; f(x) \leq f(x_1)\}$, la fonction objectif f est continûment différentiable et son gradient est lipchitzien i.e

$$\exists L > 0 \text{ telque } \|g(x) - g(\tilde{x})\| \leq L \|x - \tilde{x}\|, \forall x, \tilde{x} \in N. \quad (2.40)$$

Théorème 2.4.1 Considérons une méthode du type (2.35) et (2.36) dans laquelle d_k est une direction de descente et le pas α_k est déterminé par de Wolfe faible (2.22) et (2.23) avec $\sigma \in]0, \frac{1}{2}[$. Considérons aussi que la proposition 2.1 soit satisfaite. Alors pour une telle méthode la condition de Zoutendijk suivante :

$$\sum_{K \geq 1} \cos^2 \theta \|g_k\|^2 < \infty \quad (2.41)$$

est vérifiée.

Preuve. De (2.23) on a

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

D'autre part

CHAPITRE 2. MÉTHODES DE GLOBALISATION EN OPTIMISATION

$$\begin{aligned} d_k^T(g_{k+1} - g_k) &\leq \|g_{k+1} - g_k\| \|d_k\| \\ &\leq \alpha_k L \|d_k\|^2 \end{aligned}$$

d'où

$$\alpha_k \geq \left(\frac{\sigma-1}{L}\right) \frac{d_k^T g_k}{\|d_k\|^2}$$

En remplaçant ceci dans (2.22) on aura :

$$f(x_k + \alpha d_k) \leq f(x_k) + \mu \cos^2 \theta_k \|g_k\|^2$$

$$\text{où } \mu = \frac{\rho(\sigma-1)}{L}$$

Or puisque f est bornée sur N on a :

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

Ce qui achève la démonstration. ■

Algorithme 2.8 Méthode de Dai-Yuan avec la règle de Wolfe faible

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 :

α_k vérifie les conditions de **Wolfe faible**

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

où

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

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

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

Dai et Yuan ([13, 1998]) ont démontré qu'à chaque itération $k \geq 1$; la direction d_k définie par les relations (2.36) et (2.37) et utilisant une recherche linéaire de Wolfe faible (2.22)-(2.23), est de descente, si la fonction objectif f est strictement convexe.

Dai et Yuan ([14, 1999]) ont généralisé ce résultat pour toute fonction régulière.

Théorème 2.4.2 *Supposons que L'hypothèse 2.2.1 soit satisfaite. Pour toute méthode du type (2.35) et (2.36) dont β_k satisfait à (2.37) et le pas α_k satisfait aux conditions de Wolfe faible :*

$$\begin{aligned} f(x_k + \alpha d_k) &\leq f(x_k) + \rho \nabla^T f(x_k) d_k \\ \nabla^T f(x_k) d_k &\geq \sigma \nabla^T f(x_k + \alpha_k d_k) d_k \end{aligned}$$

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

$$d_k^T g_k < 0; \forall k \geq 1. \quad (2.42)$$

Preuve. La démonstration se fait par récurrence.

1) Pour $k = 1$:

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

2) Supposons que (2.42) est satisfaite pour $k > 1$ et démontrons qu'elle le sera pour $k + 1$:

Supposons que :

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

En utilisant (2.22), 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 :

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

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

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

$$d_k^T g_k < 0$$

Ce qui achève la démonstration. ■

2.4.3 Convergence de la méthode de Dai-Yuan

Dai-Yuan ont démontré la convergence de la méthode de Dai-Yuan si le pas α_k est déterminé par la règle de *wolfe faible* et la fonction objective satisfait les conditions citées dans la proposition 2.2.1

Théorème 2.4.3 [14] *Supposons que la proposition 2.2.1 est satisfaite. La suite $\{x_k\}$ générée par l'algorithme 2.8 converge dans le sens*

$$\liminf_{k \rightarrow \infty} \|g_k\| = 0 \quad (2.43)$$

Preuve. En utilisant le théorème 2.2.1, on aura :

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

D'autre part on a :

$$\|d_k + g_{k+1}\|^2 = \|\beta_{k+1} d_k\|^2$$

$$\Rightarrow \|d_{k+1}\|^2 = \|d_k\|^2 (\beta_{k+1})^2 - 2d_{k+1}^T g_{k+1} - \|g_{k+1}\|^2 \quad (2.44)$$

De (2.36) :

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

remplaçant ceci dans (2.44), on aura :

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

$$\begin{aligned}
 \text{d'où} \\
 \frac{\|d_k\|^2}{(d_k^T g_k)^2} &\leq \frac{\|d_{k-1}\|^2}{d_{k-1}^T g_{k-1}} + \frac{1}{\|g_k\|^2} \\
 &\leq \frac{\|d_{k-2}\|^2}{d_{k-2}^T g_{k-2}} + \frac{1}{\|g_k\|^2} + \frac{1}{\|g_{k-1}\|^2} \\
 &\leq \sum_{i=1}^k \frac{1}{\|g_i\|^2}
 \end{aligned}$$

Supposons maintenant que (2.43) n'est pas satisfaite, autrement dit :

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

on aura :

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

d'où

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

Ce qui contredit (2.41)

Ce qui achève la démonstration. ■

Deuxième partie

Méthodes à région de confiance et gradient conjugué

Chapitre 3

Méthodes des pénalités

Dans la suite, on va utiliser une méthode de pénalité, on introduit les contraintes dans l'objectif sous forme de pénalités, on part d'une solution non admissible et on essaye d'imposer à l'optimum non contraint d'arriver dans l'ensemble admissible.

Considérons le problème de programmation mathématique suivant

$$(P) \left\{ \min_{x \in \tilde{\mathbb{R}}} f(x) \right. \quad (3.1)$$

où $\tilde{\mathbb{R}} \subset \mathbb{R}^n$ et la fonction $f : \mathbb{R}^n \rightarrow \mathbb{R}$ est continue sur \mathbb{R}^n . Le plus souvent l'ensemble $\tilde{\mathbb{R}}$ est spécifié à l'aide d'un ensemble de contraintes d'égalité et/ou d'inégalité.

Dans cette méthode, la résolution du problème (P) est remplacée par celle d'une suite de problèmes sans contraintes où les contraintes du problème original (P) sont intégrées dans la fonction objectif des problèmes de la suite.

Dans la méthode des pénalités, nous cherchons à déterminer une solution optimale du problème

$$(P) \left\{ \min_{x \in \tilde{\mathbb{R}}} f(x) \right. \quad (3.2)$$

En résolvant une suite de problèmes de la forme :

$$(P_1) \left\{ \min_{x \in \mathbb{R}^n} \{f(x) + \mu_k p(x)\} = \min_{x \in \mathbb{R}^n} q(\mu_k, x) \right. \quad (3.3)$$

La suite des valeurs $\{\mu_k\}$ a les propriétés suivantes :

- i) $\mu_k \geq 0, \forall k$
- ii) La suite est croissante (c'est-à-dire $\mu_{k+1} > \mu_k$)
- iii) la suite diverge (c'est-à-dire $\mu_k \rightarrow \infty$ quand $k \rightarrow \infty$)

La fonction de pénalité est une fonction à valeur réelle $p : \mathbb{R}^n \rightarrow \mathbb{R}$ ayant les propriétés suivantes :

- i) p est continue sur \mathbb{R}^n .
- ii) $p(x) \geq 0 \forall x \in \mathbb{R}^n$.
- iii) $p(x) = 0 \Leftrightarrow x \in \tilde{\mathbb{R}}$.

Lorsque $\tilde{\mathbb{R}} = \{x \in \mathbb{R}^n : f_i(x) \leq 0, i = 1, \dots, m\}$, une fonction de pénalité souvent utilisée est la suivante :

$$p(x) = \sum_{i=1}^m (\max \{0, f_i(x)\})^2 \quad (3.4)$$

Exemple

$\tilde{\mathbb{R}} \subset \mathbb{R}^1$ où $f_1(x) = x - b \leq 0$ (i.e : $x \leq b$) et $f_2(x) = a - x \leq 0$ (i.e : $x \geq a$)

$$p(x) = (\max \{0, a - x\})^2 + (\max \{0, x - b\})^2 \quad (3.5)$$

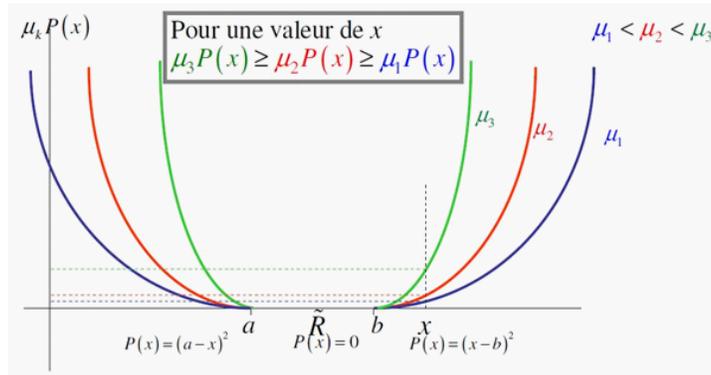


Figure 2.5

3.1 La convergence des méthodes de pénalité

Les 2 lemmes suivants servent à démontrer la convergence de la méthode. Dénotons par x_k la solution de $\min q(\mu_k, x)$.

3.1. LA CONVERGENCE DES MÉTHODES DE PÉNALITÉ

Lemme 3.1.1 *pour toute valeur de k ,*

- i) $q(\mu_k, x_k) \leq q(\mu_{k+1}, x_{k+1})$
- ii) $p(x_k) \geq p(x_{k+1})$
- iii) $f(x_k) \leq f(x_{k+1})$

Preuve

$$\begin{aligned}
 \text{i) } q(\mu_{k+1}, x_{k+1}) &= f(x_{k+1}) + \mu_{k+1}p(x_{k+1}) \\
 &\geq f(x_{k+1}) + \mu_k p(x_{k+1}) && \text{(puisque } \mu_{k+1} > \mu_k \text{ et } \\
 p(x_{k+1}) \geq 0) \\
 &\geq f(x_k) + \mu_k p(x_k) && \text{(puisque } x_k \text{ minimise } \\
 q(\mu_k, x)
 \end{aligned}$$

Donc,

$$q(\mu_{k+1}, x_{k+1}) \geq q(\mu_k, x_k) \quad (3.6)$$

$$\begin{aligned}
 \text{ii) } f(x_k) + \mu_k p(x_k) &\leq f(x_{k+1}) + \mu_k p(x_{k+1}) && \text{(puisque } x_k \text{ minimise } q(\mu_k, x)) \\
 \text{et} \\
 f(x_{k+1}) + \mu_{k+1} p(x_{k+1}) &\leq f(x_k) + \mu_{k+1} p(x_k) && \text{(puisque } x_{k+1} \text{ minimise } \\
 q(x, \mu_{k+1})) \\
 \text{additionnant les deux inégalités,}
 \end{aligned}$$

$$(\mu_{k+1} - \mu_k)p(x_{k+1}) \leq (\mu_{k+1} - \mu_k)p(x_k) \quad (3.7)$$

et comme $\mu_{k+1} > \mu_k$, on obtient :

$$p(x_{k+1}) \leq p(x_k) \quad (3.8)$$

$$\begin{aligned}
 \text{iii) } f(x_{k+1}) + \mu_k p(x_{k+1}) &\geq f(x_k) + \mu_k p(x_k) && \text{(puisque } x_k \text{ minimise } \\
 q(\mu_k, x)) \\
 \text{en utilisant ii) et le fait que } \mu_k \geq 0, &\text{ on obtient :}
 \end{aligned}$$

$$f(x_{k+1}) + \mu_k p(x_{k+1}) \geq f(x_k) + \mu_k p(x_{k+1}) \quad (3.9)$$

donc, $f(x_{k+1}) \geq f(x_k)$.

Lemme 3.1.2 *Soit x^* une solution optimale de (p_1) . Alors pour chaque k ,*

$$f(x^*) \geq q(\mu_k, x_k) \geq f(x_k) \quad (3.10)$$

Preuve

$$\begin{aligned}
 f(x^*) &= f(x^*) + \mu_k p(x^*) \text{ pour tout } k \text{ puisque } p(x^*) = 0 \\
 &\geq f(x_k) + \mu_k p(x_k) = q(\mu_k, x_k) \forall k \text{ (puisque } x_k \text{ minimise } q(\mu_k, x)) \\
 &\geq f(x_k) \forall k \text{ (puisque } \mu_k \geq 0 \text{ et } p(x_k) \geq 0). \\
 \text{Donc, } f(x^*) &\geq q(\mu_k, x_k) \geq f(x_k) \forall k.
 \end{aligned}$$

Théorème 3.1.1 *On suppose que x_k est un minimum global exact de (p_1) et $\{\mu_k\}$ une suite divergente. Alors tout point limite x^* de la suite $\{x_k\}$ est une solution du problème (p) .*

Preuve

Soit \bar{x} une solution de (p_1) , telle que

$$f(\bar{x}) \leq f(x) \text{ pour tout } \|x\| \leq \Delta$$

puisque x_k minimise $q(\mu_k, x)$ pour tout k , on a alors,

$$q(\mu_k, x_k) \leq q(\mu_k, \bar{x}) \tag{3.11}$$

donc

$$f(x_k) + \mu_k p(x_k) \leq f(\bar{x}) + \mu_k p(\bar{x}) = f(\bar{x}) \text{ (puisque } p(\bar{x}) = 0)$$

alors,

$$p(x_k) \leq \frac{1}{\mu_k} [f(\bar{x}) - f(x_k)] \tag{3.12}$$

On suppose que x^* est le point limite de $\{x_k\}$. Alors elle est une infinité de sous-suite K tel que

$$\lim_{k \in K} x_k = x^* \tag{3.13}$$

En prenant la limite de l'inégalité (3.12), on obtient :

$$p(x^*) = \lim_{k \in K} p(x_k) \leq \lim_{k \in K} \frac{1}{\mu_k} [f(\bar{x}) - f(x_k)] = 0 \text{ (puisque } \{\mu_k\} \text{ est une suite divergente)}$$

Donc, on a $p(x^*) = 0$, et par conséquent $\|x^*\| \leq \Delta$.

L'optimalité de x^* découle directement du lemme.2. En effet la relation $f(x_k) \leq f(x^*)$ étant vérifiée pour tout k , il s'ensuit que

$$f(x^*) = \lim_{k \in K} f(x_k) \leq f(\bar{x}) \tag{3.14}$$

Chapitre 4

Méthodes à régions de confiance

4.1 Méthodes à régions de confiance pour le cas quadratique

Dans cette partie, on s'intéresse aux méthodes à régions de confiance et plus particulièrement au cas utilisant des fonctions modèles quadratiques.

4.1.1 Optimisation de fonctions modèles quadratiques

Les méthodes à régions de confiance, dites classiques, consistent à résoudre une suite de sous problèmes d'optimisation de fonctions quadratiques d'un problème d'optimisation d'une fonction objectif f plus complexe du type (P)

$$(P) \quad \left\{ \min_{x \in \mathbb{R}^n} f(x) \right. . \quad (4.1)$$

Les méthodes à région de confiance sont fondées sur le principe suivant, qui est très général et peut donc s'appliquer à d'autres contextes. La fonction objectif f est donc remplacée par une fonction modèle quadratique Ψ , dans une certaine région autour d'un point x donné. Les deux premiers termes intervenant dans la fonction quadratique peuvent s'identifier aux deux premiers termes de la série de Taylor de f autour de l'itéré courant x . La fonction modèle est alors basée sur le gradient de la fonction objectif. Dans des situa-

tions pour lesquelles il est difficile d'évaluer le gradient de la fonction objectif, une approximation de ce dernier peut être utilisée.

4.1.2 Fonctions modèles utilisant le gradient exact de la fonction objectif

Lorsque le gradient de la fonction objectif est connu, on peut facilement évaluer la fonction modèle Ψ de la façon suivante. Au point courant $x_k \in \mathbb{R}^n$, on suppose donné un modèle de la variation de f pour un incrément s de x_k . En optimisation différentiable, il est raisonnable de considérer un modèle quadratique de la forme

$$\Psi(x_k + s) = f(x_k) + \nabla f(x_k)^T s + \frac{1}{2} s^T H_k s \quad (4.2)$$

où $\nabla f(x_k)$ est le gradient de f en x_k et H_k est une matrice symétrique qui peut être $\nabla^2 f(x_k)$, le hessien de f en x_k , ou une approximation de celui-ci. Gradient et hessien sont supposés être calculés pour le produit scalaire euclidien $\langle u, v \rangle = \sum_i u_i v_i$.

on n'impose pas à la matrice H_k d'être définie positive (en particulier, elle peut être nulle).

si $H_k = \nabla^2 f(x_k)$, on a le développement limité à l'ordre 2 :

$$f(x_k + s) \simeq \Psi(x_k + s). \quad (4.3)$$

On pose $\Psi_k(s) = \Psi(x_k + s)$, dans ce cas, le modèle quadratique est adéquat pour s voisin de 0. Dans les méthodes à régions de confiance, on considère que Ψ_k est un modèle de la variation de f qui est acceptable dans un voisinage de la forme :

$$\overline{B}(0, \Delta_k) = \{x \in \mathbb{R}^n : \|x\| \leq \Delta_k\}, \quad (4.4)$$

où $\Delta_k > 0$ et $\|\cdot\|$ est la norme euclidienne. Ce domaine est appelé **la région de confiance du modèle** Ψ_k et Δ_k est appelé le **rayon de confiance**.

Pour trouver l'accroissement s_k à donner à x_k , on minimise le modèle quadratique Ψ_k sur la région de confiance. On doit donc résoudre le sous-problème quadratique :

$$(RC_k) \begin{cases} \min \Psi_k(s) \\ \|s\| \leq \Delta_k \end{cases} \quad (4.5)$$

4.1. MÉTHODES À RÉGIONS DE CONFIANCE POUR LE CAS QUADRATIQUE

Les méthodes à région de confiance sont illustrées dans la figure ci-dessous

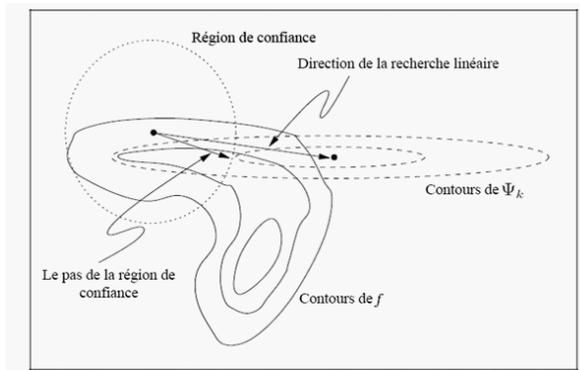


Figure 2.6

Soit s_k une solution (en général approchée) de sous-problème quadratique (RC_k). Si $f(x_k + s)$ est suffisamment plus petit que $f(x_k)$, on accepte le pas s_k et on passe à l'itération suivante avec $x_{k+1} = x_k + s_k$. Dans le cas contraire, on résout à nouveau (RC_k) avec un rayon de confiance plus petit, car on considère que le modèle courant n'est pas fiable sur la boule $\overline{B}(0, \Delta_k)$. Nous verrons que, comme f et Ψ_k coïncident au premier ordre, l'algorithme finit par trouver un rayon de confiance $\Delta_k > 0$ pour lequel la solution de (RC_k) est acceptée.

On ajuste ensuite le modèle à l'itéré suivant x_{k+1} c'est-à-dire le calcul de $\nabla f_{k+1} = \nabla f(x_{k+1})$, de H_{k+1} et de Δ_{k+1} . L'ajustement du rayon de confiance est basé sur la comparaison entre la décroissance réelle de f en passant de x_k à x_{k+1} et la décroissance prédite par le modèle, c'est-à-dire $\Psi_k(x_k) - \Psi_k(x_k + s_k)$. On peut alors passer à l'itération suivante.

4.1.3 Description d'un sous-problème standard à région de confiance

Pour un point x_k et un rayon Δ_k donnés, on résout le sous-problème quadratique (RC_k). La résolution de ce problème peut s'effectuer de manière exacte. Cependant, pour alléger les calculs numériques, elle s'effectue souvent de manière approchée en utilisant la méthode du point de Cauchy que l'on décrira dans le paragraphe (4.1.4). Néanmoins, il n'existe à priori aucune preuve que le nouveau point $x_k + s_k$, construit à l'aide de la solution s_k du

sous-problème (RC_k) , soit acceptable pour le problème de départ (P) . En effet rien ne prouve que l'on ait $f(x_k + s_k) < f(x_k)$, c'est-à-dire que la fonction objectif f décroît. Il est alors nécessaire d'évaluer la valeur de la fonction f au nouveau point $x_k + s_k$, et de la comparer à la valeur de la fonction f au point x_k . Pour cela on évalue la réduction réelle de la fonction f de la façon suivante

$$f(x_k) - f(x_k + s_k). \quad (4.6)$$

Cette réduction est ensuite comparée à la réduction prédite par la fonction modèle Ψ_k , laquelle est donnée par

$$\Psi(x_k) - \Psi(x_k + s_k). \quad (4.7)$$

L'efficacité de la fonction modèle peut être déterminée par le rapport suivant

$$\rho_k = \frac{f(x_k) - f(x_k + s_k)}{\Psi(x_k) - \Psi(x_k + s_k)}. \quad (4.8)$$

Ce rapport est alors utilisé comme critère d'actualisation du rayon de la région de confiance Δ_k . En effet,

(a) Si $\rho_k < 0$, le pas est rejeté, le rayon de confiance est diminué, et une autre minimisation de la fonction modèle est effectuée à l'intérieur de la nouvelle région de confiance

(b) Si $\rho_k > 0$, le pas est accepté et le rayon de confiance est actualisé selon les valeurs prises par ρ_k :

- Si ρ_k est proche de 1, les fonctions modèle et objectif sont en bon accord à l'itération k et le rayon de la région de confiance est augmenté afin d'obtenir des pas plus performants.

- Si ρ_k est proche de 0, la fonction modèle ne représente pas correctement la fonction objectif et le rayon de confiance est diminué.

- Si ρ_k se situe entre 0 et 1, sans être proche de ses extrémités, la fonction modèle est supposée être acceptable et le rayon de confiance n'est pas modifié.

4.1.4 Algorithme 1 (Région de confiance pour des fonctions modèle quadratique)

Décrivons de manière précise une itération de la méthode à régions de confiance, celle passant de l'itéré x_k à x_{k+1} . Nous commenterons ses différentes étapes par la suite.

4.1. MÉTHODES À RÉGIONS DE CONFIANCE POUR LE CAS QUADRATIQUE

Etape d'initialisation

Choix des constantes η_1, η_2 (Seuils de succes), γ_1, γ_2 et γ_3 (Facteurs de mise à jour) telles que

$$0 < \eta_1 < \eta_2 < 1 \text{ et } 0 < \gamma_1 < \gamma_2 < 1 \leq \gamma_3.$$

Choix d'un rayon de confiance Δ_0 et d'un itéré x_0 . Calcul de $f(x_0)$. Poser $k = 0$ et aller à l'étape 1.

Etape 1

Construction de la fonction modèle quadratique $\Psi(x_k + s_k)$, conformément à la relation (3.2), et détermination d'une solution approchée s_k du problème (RC_k)

$$(RC_k) \begin{cases} \min \Psi_k(s) \\ \|s\| \leq \Delta_k . \end{cases}$$

Etape 2

Calcul de $f(x_k + s_k)$ et évaluation du critère de performance

$$\rho_k = \frac{f(x_k) - f(x_k + s_k)}{\Psi(x_k) - \Psi(x_k + s_k)}.$$

Etape 3

Actualisation du rayon de confiance :

(a) Si $\rho_k \geq \eta_1$:

L'étape est un succès et le pas est accepté. $x_{k+1} = x_k + s_k$ et choix de $\Delta_k \in [\Delta_k, \gamma_3 \Delta_k]$.

(b) Si $\eta_1 \leq \rho_k \leq \eta_2$:

L'étape est un succès et le pas est accepté. $x_{k+1} = x_k + s_k$ et choix de $\Delta_k \in [\gamma_2 \Delta_k, \Delta_k]$.

(c) Si $\rho_k < \eta_1$:

L'étape est un échec et le pas est rejeté. $x_{k+1} = x_k$ et choix de $\Delta_k \in [\gamma_1 \Delta_k, \gamma_2 \Delta_k]$.

Etape 4

Si un critère de convergence est vérifié, l'algorithme est arrêté. Sinon, on pose $k = k + 1$ et on retourne à l'étape 1.

On choisit habituellement $\eta_1 = 0,25$ et $\eta_2 = 0,75$ (Critères de performance), et $\gamma_1 = 0,25$, $\gamma_2 = 0,75$ et $\gamma_3 = 2$ (Critères d'actualisation).

L'évaluation de l'étape 1 et notamment la résolution du sous-problème (RC_k) est un point délicat de l'algorithme précédent.

soit (s_k, Δ_k) la solution approchée et le rayon de confiance obtenus à la sortie de l'étape 2. On ne prend pas les valeurs intermédiaires prises par ce couple lorsque la condition $\rho_k \geq \eta_1$ n'est pas vérifiée.

Ces méthodes peuvent être comparées aux méthodes de recherches linéaires, dites inexactes, pour lesquelles on cherche un pas (non nécessairement optimal). Dans ce cas, il est essentiel d'avoir H_k semi-définie positive, sinon le sous-problème quadratique n'a pas de solution. De plus, en prenant H_k définie positive pour réaliser la décroissance de f .

Lorsque l'on passe à l'étape 4, on a $\rho_k \geq \gamma_1$, ce qui s'écrit encore

$$f(x_k + s_k) \leq f(x_k) + \gamma_1 \Psi_k(s_k). \quad (4.9)$$

Cette condition rappelle la condition d'Armijo pour les méthodes à directions de descente et recherche linéaire, mais le terme contrôlant la décroissance de f , ici $\gamma_1 \Psi_k(s_k) \leq 0$, n'a pas le caractère linéaire qu'il a dans la règle d'Armijo (technique de rebroussement), on force la réalisation de cette condition en faisant décroître Δ_k .

L'étape 2 qui détermine Δ_k correspond à l'étape de calcul du pas λ_k en recherche linéaire. La différence essentielle est que la valeur de Δ_k a une incidence sur la direction $s_k \in \mathbb{R}^n$, alors que λ_k n'affecte pas la direction $d_k \in \mathbb{R}^n$ suivant laquelle on se déplace en recherche linéaire.

La règle de mise à jour du rayon de confiance à l'étape 3 est simple. On sait alors que le critère de performance $\rho_k \geq \gamma_1 > 0$. Si la performance est moyenne ($\gamma_1 \leq \rho_k \leq \gamma_2$) on diminue Δ_k , si elle est bonne ($\rho_k > \gamma_2$) on augmente Δ_k . Remarquons que, contrairement aux méthodes à directions de descente, il n'y a aucun mécanisme permettant d'accroître Δ_k au cours de la même itération. L'accroissement éventuel de Δ_k se fait ici d'une itération à l'autre. Ceci est acceptable car il est raisonnable de faire l'hypothèse que la région de confiance du modèle varie peu d'une itération à l'autre.

4.1.5 Méthode du point de Cauchy

De manière similaire aux méthodes de recherches linéaires, la détermination de pas optimaux n'est pas une condition nécessaire pour obtenir une

4.1. MÉTHODES À RÉGIONS DE CONFIANCE POUR LE CAS QUADRATIQUE

convergence globale. Sous certaines conditions, une bonne approximation de ces pas peut être acceptable. Il suffit de déterminer une solution approchée, s_k , à l'intérieur de la région de confiance, qui produise une réduction suffisante de la fonction modèle. Cette réduction peut être obtenue par la méthode du point de Cauchy s_k^c (voir Figure 4.2). Nous décrivons cette méthode dans l'algorithme ci dessous :

Algorithme (Point de Cauchy)

- Déterminer un vecteur s_k^s , solution du problème (RC_k) linéarisé, soit

$$s_k^s = \arg \min_{s \in \mathbb{R}} (f_k + \nabla f_k^T s), \text{ sous les contraintes } \|s\| \leq \Delta_k. \quad (4.10)$$

- Déterminer le scalaire $\lambda_k > 0$ minimisant $\Psi_k(\lambda s_k^s)$, soit

$$\lambda_k = \arg \min_{\lambda > 0} \Psi_k(\lambda s_k^s), \text{ sous les contraintes } \|\lambda s_k^s\| \leq \Delta_k. \quad (4.11)$$

- Finalement, $s_k^c = \lambda_k s_k^s$.

4.1.6 Calcul pratique du point de Cauchy

La résolution du problème (4.10) donne aisément la solution suivante :

$$s_k^s = -\frac{\Delta_k}{\|\nabla f_k\|} \nabla f_k.$$

Pour trouver le pas λ_k qui minimise

$$\Psi_k(\lambda s_k^s) = f(x_k) + \nabla f(x_k)^T \lambda s_k^s + \frac{1}{2} \lambda^2 s_k^{sT} H_k s_k^s,$$

on s'annule la dérivée par rapport au λ . Donc :

$$\nabla f(x_k)^T s_k^s + \lambda s_k^{sT} H_k s_k^s = 0$$

$$-\nabla f(x_k)^T \frac{\Delta_k}{\|\nabla f_k\|} \nabla f_k + \lambda \frac{\Delta_k}{\|\nabla f_k\|} \nabla f_k^T H_k \frac{\Delta_k}{\|\nabla f_k\|} \nabla f_k = 0$$

Alors

$$\lambda_k = \frac{\|\nabla f_k\|^3}{\Delta_k \nabla f_k^T H_k \nabla f_k}$$

La détermination du pas λ_k dépend du signe de $\nabla f_k^T H_k \nabla f_k$:

- Si $\nabla f_k^T H_k \nabla f_k \leq 0$, la fonction $\Psi_k(\lambda s_k^s)$ décroît de façon monotone avec λ , et ce même si $\nabla f_k \neq 0$. Le pas λ_k est donc la plus grande valeur qui satisfasse le rayon de confiance et donc $\lambda_k = 1$.

- Si $\nabla f_k^T H_k \nabla f_k > 0$, la fonction $\Psi_k(\lambda s_k^s)$ est convexe et quadratique en λ . Le pas λ_k est alors soit la valeur qui minimise la fonction quadratique $\frac{\|\nabla f_k\|^3}{\Delta_k \nabla f_k^T H_k \nabla f_k}$, soit la valeur limite 1 déterminée précédemment.

En résumé, on a :

$$s_k^c = -\lambda_k \frac{\Delta_k}{\|\nabla f_k\|} \nabla f_k,$$

avec

$$\lambda_k = \begin{cases} 1 & \text{si } \nabla f_k^T H_k \nabla f_k \leq 0, \\ \min \left\{ \frac{\|\nabla f_k\|^3}{\Delta_k \nabla f_k^T H_k \nabla f_k}, 1 \right\} & \text{sinon.} \end{cases} \quad (4.12)$$

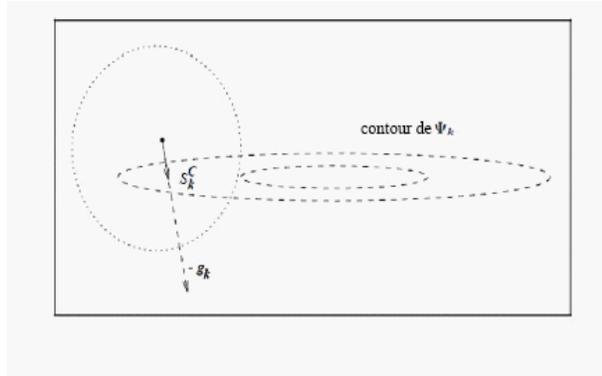


Figure 2.7

4.1.7 Condition de décroissance suffisante

Comme nous venons de le voir, le point de Cauchy s_k^c permet de valider ou non une solution approchée du sous problème à région de confiance. La convergence des méthodes à région de confiance est prouvée lorsque le pas s_k produit une décroissance suffisante de la fonction modèle Ψ_k . A chaque itération, la réduction de la fonction modèle en un point doit au moins être égale à un multiple de la décroissance obtenue au point de Cauchy. La condition de la fraction de décroissance de Cauchy (Fraction of Cauchy Decrease,

4.1. MÉTHODES À RÉGIONS DE CONFIANCE POUR LE CAS QUADRATIQUE

fc) doit alors être vérifiée. Elle s'écrit de la façon suivante :

$$\Psi_k(x_k) - \Psi_k(x_k + s_k) \geq c_{cfd} [\Psi_k(x_k) - \Psi_k(x_k + s_k^c)], c_{cfd} > 0. \quad (4.13)$$

Or, il a été montré par Powell [7] que pour la décroissance de Cauchy, on a :

$$\Psi_k(x_k) - \Psi_k(x_k + s_k^c) \geq \frac{1}{2} \|\nabla f_k\| \min \left\{ \frac{\|\nabla f_k\|}{\|H_k\|}, \Delta_k \right\}. \quad (4.14)$$

En combinant (4.13) et (4.14), et en supposant que la recherche du point s_k , effectuée à l'algorithme (région de confiance modèle quadratique) est un succès, c'est-à-dire si $f(x_k) - f(x_k + s_k) \geq \eta_1 [\Psi_k(x_k) - \Psi_k(x_k + s_k)]$, on aboutit à la condition de suffisance :

$$f(x_k) - f(x_k + s_k) \geq \frac{1}{2} \eta_1 c_{cfd} \|\nabla f_k\| \min \left\{ \frac{\|\nabla f_k\|}{\|H_k\|}, \Delta_k \right\}. \quad (4.15)$$

4.1.8 Résultats de convergence

Différents théorèmes de convergence globale découlent de la condition de décroissance suffisante (4.15).

Théorème 4.1.1 (Convergence globale faible) [40]

Soit $f : \mathbb{R}^n \rightarrow \mathbb{R}$ une fonction continûment dérivable sur

$$X = \{x \in \mathbb{R}^n : f(x) < f(x_0)\},$$

et soit $\{x_k\}$ une suite générée par l'algorithme (région de confiance pour des modèles fonctions quadratiques), avec H_k uniformément bornée. Si de plus le pas s_k vérifie la condition (4.13), on a alors :

$$\liminf_{k \rightarrow \infty} \|\nabla f_k\| = 0. \quad (4.16)$$

Preuve

En utilisant l'expression du rapport ρ_k , formule (4.8), on peut obtenir

$$\begin{aligned} |\rho_k - 1| &= \left| \frac{(f(x_k) - f(x_k + s_k)) - (\Psi_k(0) - \Psi_k(s_k))}{\Psi_k(0) - \Psi_k(s_k)} \right| \\ &= \left| \frac{\Psi_k(s_k) - f(x_k + s_k)}{\Psi_k(0) - \Psi_k(s_k)} \right|. \end{aligned} \quad (4.17)$$

D'après le théorème de Taylor on a :

$$f(x_k + s_k) = f(x_k) + \nabla f(x_k)^T s_k + \int_0^1 [\nabla f(x_k + ts_k) - \nabla f(x_k)]^T s_k dt,$$

pour tout $t \in]0, 1[$, il suit de la définition de Ψ_k que

$$\begin{aligned} |\Psi_k(s_k) - f(x_k + s_k)| &= \left| \frac{1}{2} s_k^T B_k s_k - \int_0^1 [\nabla f(x_k + ts_k) - \nabla f(x_k)]^T s_k dt \right| \\ &\leq \left(\frac{\beta}{2} \right) \|s_k\|^2 + \beta_1 \|s_k\|^2, \end{aligned} \quad (4.18)$$

où β_1 est la constante de Lipchitz de ∇f sur l'ensemble $\overline{B}(0, \Delta_k)$.

On suppose que $\|s_k\|$ ce qui assure que x_k et $x_k + ts_k$ reste à l'intérieur de $\overline{B}(0, \Delta_k)$.

Raisonnons par l'absurde, donc il existe $\varepsilon > 0$ et un indice positive K tel que

$$\|\nabla f(x_k)\| > \varepsilon, \text{ pour tout } k \geq K. \quad (4.19)$$

D'après (4.14), on a pour tout $k \geq K$

$$\Psi_k(0) - \Psi_k(s_k) \geq c_1 \|\nabla f_k\| \min \left\{ \frac{\|\nabla f_k\|}{\|H_k\|}, \Delta_k \right\} \geq c_1 \varepsilon \min \left\{ \frac{\varepsilon}{\beta}, \Delta_k \right\}. \quad (4.20)$$

En utilisant (4.18), (4.20) et le fait que

$$\|s_k\| \leq \gamma \Delta_k, \text{ pour toute constante } \gamma \geq 1,$$

on a

$$|\rho_k - 1| \leq \frac{\gamma^2 \Delta_k^2 (\beta/2 + \beta_1)}{c_1 \varepsilon \min \left\{ \frac{\varepsilon}{\beta}, \Delta_k \right\}}. \quad (4.21)$$

4.1. MÉTHODES À RÉGIONS DE CONFIANCE POUR LE CAS
QUADRATIQUE

En choisissant $\bar{\Delta}$ tel que $\Delta_k \leq \bar{\Delta}$ et défini par

$$\bar{\Delta} = \min \left(\frac{1}{2} \frac{c_1 \varepsilon}{\gamma^2 (\beta/2 + \beta_1)}, \frac{\Delta_k}{\gamma} \right), \quad (4.22)$$

$$\Delta_k \geq \min (\Delta_K, \bar{\Delta}/4), \text{ pour tout } k \geq K. \quad (4.1)$$

Supposons que pour un sous ensemble K' tel que pour tout $k \in K'$ on a $\rho_k \geq \frac{1}{4}$, alors

$$|\rho_k - 1| \leq \frac{1}{2}, \text{ pour tout } k \in K'$$

et par suite que

$$\begin{aligned} f(x_k) - f(x_{k+1}) &= f(x_k) - f(x_k + s_k) \\ &\geq \frac{1}{4} [\Psi_k(0) - \Psi_k(s_k)] \\ &\geq \frac{1}{4} c_1 \varepsilon \min \left\{ \frac{\varepsilon}{\beta}, \Delta_k \right\}. \end{aligned}$$

Puisque f est bornée on a

$$\lim_{k \rightarrow \infty} \Delta_k = 0,$$

contradiction avec (4.22). Donc

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

■

Une convergence forte peut par ailleurs être énoncée si l'on impose des contraintes supplémentaires sur la fonction objectif.

Théorème 4.1.2 (*Convergence globale forte*) [40]

Soit $f : \mathbb{R}^n \rightarrow \mathbb{R}$ une fonction dérivable, avec un gradient continu au sens de Lipchitz, inférieurement borné sur $X = \{x \in \mathbb{R}^n : f(x) < f(x_0)\}$, et soit $\{x_k\}$ une suite d'itérés générée par l'algorithme (région de confiance pour

des modèles fonctions quadratiques), avec H_k uniformément bornée. Si de plus le pas s_k vérifie la condition (4.13), on a alors :

$$\lim_{k \rightarrow \infty} \|\nabla f_k\| = 0. \quad (4.23)$$

Preuve

On considère pour un indice particulier m tel que $\nabla f_m \neq 0$ et pour β_1 la constante de Lipchitz de ∇f sur $\overline{B}(0, \Delta_k)$ la relation suivante :

$$\|\nabla f - \nabla f_m\| \leq \beta_1 \|x - x_m\|,$$

pour tout $x \in \overline{B}(0, \Delta_k)$. On définit à présent les scalaires ε et R telles que

$$\varepsilon = \frac{1}{2} \|\nabla f_m\|, \quad R = \min\left(\frac{\varepsilon}{\beta_1}, \Delta_k\right).$$

Notons que la boule

$$B(x_m, R) = \{x : \|x - x_m\| \leq R\}$$

est contenu dans $\overline{B}(0, \Delta_k)$ et comme ∇f est continu au sens de Lipchitz on a

$$x \in B(x_m, R) \Rightarrow \|\nabla f(x)\| \geq \|\nabla f_m\| - \|\nabla f(x) - \nabla f_m\| \geq \frac{1}{2} \|\nabla f_m\| = \varepsilon.$$

On extrait une sous-suite $\{x_k\}_{k \geq m} \subset B(x_m, R)$, pour laquelle on

$$\|\nabla f_k\| \geq \varepsilon, \quad \text{pour tout } k \geq m.$$

Soit $l \geq m$ tel que x_{k+1} soit le premier itéré après x_m à l'extérieur de $B(x_m, R)$. Puisque $\|\nabla f_k\| \geq \varepsilon$ pour tout $k = m, m+1, \dots, l$, on peut écrire que

$$\begin{aligned} f(x_m) - f(x_{l+1}) &= \sum_{k=m}^l f(x_k) - f(x_{k+1}) \\ &\geq \sum_{k=m}^l \eta c_1 \varepsilon \min\left(\Delta_k, \frac{\varepsilon}{\beta}\right) \\ &\geq \eta c_1 \varepsilon \frac{\varepsilon}{\beta}. \end{aligned}$$

Comme la suite $\{f(x_k)\}_{k=0}^l$ est décroissante et bornée alors elle converge vers f^* de plus $(f(x_m) - f^*) \rightarrow 0$, ce qui donne $\nabla f_m \rightarrow 0$. ■

4.1.9 Fonctions modèles utilisant un gradient approché de la fonction objectif

Dans la section précédente, on vient de donner une classe d'algorithmes à région de confiance, pour des fonctions modèles quadratiques construites à l'aide du gradient exacte de la fonction objectif. Généralement, le gradient de la fonction objectif est très difficile à obtenir, et seule une approximation numérique (Par discrétisation) de celui ci est accessible.

A présent g_k représentera une approximation du gradient exact ∇f_k , et M_k une approximation du hessien exacte, Une fonction quadratique basée sur un gradient approché peut s'écrire de la manière suivante :

$$\Psi(s) = f(x_k) + g_k^T s + \frac{1}{2} s^T M_k s. \quad (4.24)$$

Les fonctions (4.24) et (4.2) sont supposées avoir les mêmes propriétés. Afin de prouver la convergence de telles méthodes, il est nécessaire de prendre en considération la différence entre les gradients exact et approché, $\nabla f_k - g_k$.

Différentes méthodes existent pour prendre en compte cette erreur [35].

$$\frac{\|\nabla f_k - g_k\|}{\|g_k\|} \leq \xi, \text{ avec } \xi \in [0, 1]. \quad (4.25)$$

Comme nous l'avons précisé précédemment, il suffit de déterminer une condition de décroissance suffisante afin d'être en mesure de prouver la convergence de méthodes à région de confiance. Pour les méthodes à région de confiance qui utilisent le gradient approché, On propose une condition de décroissance suffisante proposée par Moré (1983) similaire à celles utilisées pour des fonctions modèles qui utilisent le gradient exact. Il remplace simplement $\|\nabla f_k\|$ par $\|g_k\|$:

$$f(x_k) - f(x_k + s_k) \geq \frac{1}{2} \eta_1 c_{cfd} \|g_k\| \min \left\{ \frac{\|g_k\|}{\|H_k\|}, \Delta_k \right\}.$$

Cette condition de décroissance, couplée à la condition (4.25) permet de prouver la convergence globale de méthodes à région de confiance basées sur des gradients approchés [35]. On a donc $\lim_{k \rightarrow \infty} \|g_k\| = 0$.

La condition (4.25) donne finalement un résultat de convergence globale forte :

$$\lim_{k \rightarrow \infty} \|\nabla f_k\| = 0.$$

4.2 Optimisation de fonctions modèles générales

Les principes de base d'un algorithme standard à région de confiance, qui utilise des modèles quadratiques, viennent d'être présentés. Nous allons maintenant nous attacher à décrire les principes d'un algorithme qui utilise des fonctions modèles générales, Ψ^{nonlin} , où aucune hypothèse de non-linéarité n'est effectuée.

On considère une fonction objectif $f : \mathbb{R}^n \mapsto \mathbb{R}$. Pour un point x_0 donné, on définit par X le sous espace ouvert convexe contenant le sous espace $X_0 = \{x \in \mathbb{R}^n \mid f(x) < f(x_0)\}$. Les seules restrictions sur la fonction objectif f sont les suivantes :

- (F1) f est continûment dérivable dans X ,
- (F2) f est bornée inférieurement, et
- (F3) ∇f est L-Lipschitzienne dans X .

De plus, bien que la fonction modèle soit supposée être quelconque, elle doit tout de même vérifier certaines conditions qui sont assez peu restrictives :

- (M1) Ψ_k^{nonlin} est dérivable dans un ouvert convexe contenant le rayon de la région de confiance $\{x \in \mathbb{R}^n \mid \|x - x_k\| \leq \Delta_k\}$,
- (M2) $\Psi_k^{nonlin}(x_k) = f(x_k)$, et
- (M3) pour $\varepsilon \in [0, 1]$, $g_k = \nabla \Psi_k^{nonlin}(x_k)$ est une approximation de $\nabla f(x_k)$ qui vérifie la condition

$$\frac{\|\nabla f(x_k) - g_k\|}{\|g_k\|} \leq \varepsilon.$$

Un algorithme pour les méthodes à région de confiance qui utilise des fonctions modèles quelconques possède la même structure que l'algorithme 1. Le problème (RC) est simplement remplacé par le problème suivant :

$$\min \Psi_k^{nonlin}(x_k + s), \text{ sous les contraintes } \|s\| \leq \Delta_k. \quad (4.26)$$

Il a été montré précédemment que la convergence d'algorithmes à région de confiance qui utilisent des fonctions modèles quadratiques est liée à une condition de décroissance suffisante (fraction de Cauchy) construite pour une étape à succès ($\rho \geq \eta_1$). On rappelle que le point de Cauchy est l'unique point minimisant la fonction modèle le long de la plus grande pente, $-\nabla f_k$, à l'intérieur de la région de confiance. A priori, lorsque la fonction modèle n'est pas

quadratique, rien ne prouve que le point de Cauchy produise une décroissance suffisante de la fonction modèle. Il semble alors intéressant de généraliser cette condition de décroissance suffisante aux fonctions quelconques, non nécessairement quadratiques. Pour ce faire, la décroissance prédite à un itéré k , $\Psi_k^{nonlin}(x_k) - \Psi_k^{nonlin}(x_k + s)$, doit être comparée à la décroissance prédite le long de la plus grande pente, $\Psi_k^{nonlin}(x_k) - \Psi_k^{nonlin}(x_k + \lambda_k g_k)$, pour un pas λ_k positif.

L'algorithme 2, permet de déterminer un pas produit une décroissance suffisante pour une fonction modèle générale.

Algorithme 2(Détermination du pas)

Choix de constantes $\alpha, \beta, v_1, v_2, v_3$ et μ telles que :

$$0 < \alpha < \beta < 1, \quad 0 < v_1 < 1, \quad 0 < v_2, \quad 0 < v_3 \text{ et } 0 < \mu \leq 1. \quad (4.27)$$

1. Détermination de λ_K^A . Deux méthodes peuvent être utilisées.

(a) Méthode 1 :

$$\Psi_k^{nonlin}(x_k - \lambda_K^A g_k) \leq \Psi_k^{nonlin}(x_k) - \alpha \lambda_K^A \|g_k\|^2 \quad (4.28.a)$$

$$\|\lambda_K^A g_k\| \leq \Delta_k \quad (4.28.b)$$

$$\lambda_K^A \geq \min \left\{ v_1 \frac{\Delta_k}{\|g_k\|}, v_2 \right\} \quad (4.28.c)$$

(b) Méthode 2 :

$$\Psi_k^{nonlin}(x_k - \lambda_K^A g_k) \leq \Psi_k^{nonlin}(x_k) - \alpha \lambda_K^A \|g_k\|^2 \quad (4.29.a)$$

$$\|\lambda_K^A g_k\| \leq \Delta_k \quad (4.29.b)$$

$$\lambda_K^A \geq \lambda_K^B > 0 \quad (4.29.c)$$

$$\Psi_k^{nonlin}(x_k - \lambda_K^B g_k) \leq \Psi_k^{nonlin}(x_k) - \beta \lambda_K^B \|g_k\|^2 \quad (4.29.d)$$

2. Si $\Delta_k > v_3$: choix de s_k tel que

$$\Psi_k^{nonlin}(x_k) - \Psi_k^{nonlin}(x_k + s_k) \geq \mu \left[\Psi_k^{nonlin}(x_k) - \Psi_k^{nonlin}(x_k - \lambda_K^A g_k) \right] \quad (4.30.a)$$

$$\|s_k\| \leq \Delta_k. \quad (4.30.b)$$

L'étape 1 de l'algorithme 2 s'apparente à une méthode de recherche inexacte linéaire (méthode d'Armijo), à l'intérieure de la région de confiance, le long de la plus grande pente (Nocedal et Wright,1999). Cette étape fournit un pas de Cauchy généralisé, $s_k^{GC} = \lambda_K^A g_k$.

L'étape 2 précise que si le rayon de la région de confiance est trop grand, il est possible de quitter la direction de plus grande descente. Afin d'éviter ce cas défavorable, il est alors envisagé de déterminer un pas s_k qui vérifie une certaine condition de décroissance. Il s'agit de la condition sur la fraction de décroissance de Cauchy généralisée(théorème3).

Quelques remarque concernant l'algorithme 2 peuvent être formulées :

1.Existence d'une solution de l'algorithme 2

L'algorithme 2 est bien posé. En effet, si les constantes (4.27) sont fixées, il existe toujours un pas s_k qui satisfasse les conditions (4.28) ou (4.29), ainsi que les conditions (4.30). La démonstration de ce lemme peut être trouvée dans Toint (1981).

2.Comportement asymptotique de la solution de l'algorithme 2

Un résultat sur le comportement asymptotique des directions des pas $\{s_k\}$, déterminés par l'algorithme 2, dans la résolution du problème (4.26), peut être énoncé :

Soient $\{\Delta_k\}$, $\{g_k\}$ et $\{s_k\}$ les séquence respectives de rayons de confiance, de gradients de la fonction modèle et de pas déterminés par l'algorithme 2.

Si $\liminf_{k \rightarrow \infty} \|g_k\| > 0$ et $\lim_{k \rightarrow \infty} \Delta_k = 0$, on a : $\lim_{k \rightarrow \infty} \frac{-s_k^T g_k}{\|s_k\| \|g_k\|} = 1$

Le lecteur intéressé par une démonstration de ce lemme pourra se référer à Conn et al(2000).

Il reste maintenant à déterminer une condition de décroissance suffisante, afin de prouver la convergence d'algorithme 2 à région de confiance qui utilisent des fonctions modèles générales. Pour un itéré k , l'approximation du hessien, H_k donne des informations sur la courbure de la fonction objectif, qui serviront à énoncer une condition de décroissance suffisante.

En effet, en définissant une mesure de courbure $\omega(f, x, s)$:

$$\omega(f, x, s) = \frac{2}{\|s\|^2} \left(f(x+s) - f(x) - \nabla f(x)^T s \right), \quad (4.31)$$

Pour une fonction f vérifiant la condition **(F1)**, et pour un pas $s \neq 0$ déterminé par l'algorithme 2 il est possible de prouver que la décroissance de la fonction Ψ_k^{nonlin} est inférieurement bornée.

Un théorème basé sur les travaux de Toint (1981) peut être énoncé :

Théorème 4.2.1 (Décroissance de Cauchy pour des fonction générales)

Soit $g_k \neq 0$ une approximation du gradient exact de la fonction objectif . En combinant l'algorithme 2 et la condition (4.31), on obtient :

$$\omega_k = \begin{cases} 0 & \text{si la méthode 1 de l'algorithme 2 est utilisée,} \\ \omega(\Psi_k^{nonlin}, x_k - \lambda_K^B g_k) & \text{si la méthode 2 de l'algorithme 2 est utilisée.} \end{cases}$$

i.e

$$\omega_k \geq 0,$$

et il existe une constante $c_s > 0$ telle que l'algorithme 2 produise un pas s_k qui vérifie.

$$\Psi_k^{nonlin}(x_k) - \Psi_k^{nonlin}(x_k + s_k) \geq c_s \|g_k\|^2 \min \left\{ \frac{\|g_k\|}{1 + \omega_k}, \Delta_k \right\} \quad (4.32)$$

Pour une étape à succès de l'algorithme 1, une condition de décroissance suffisante peut être obtenue :

$$f(x_k) - f(x_k + s_k) \geq \eta_1 c_s \|g_k\| \min \left\{ \frac{\|g_k\|}{\|1 + \omega_k\|}, \Delta_k \right\}. \quad (4.33)$$

les condition(4.33) est identique à la condition(4.15) déterminée pour des fonctions modèles quadratiques.

Les théorèmes 1 et 2 , associés à la condition (4.33), prouvent la convergence des méthodes à région de confiance pour des fonctions modèles quelconques.

Chapitre 5

Méthode à région de confiance et gradient conjugué

Dans ce travail on cherche à résoudre le problème d'optimisation sans contraintes (P) suivant :

$$(P) : \left\{ \min_{x \in \mathbb{R}^n} f(x) \right. \quad (5.1)$$

où la fonction f est supposée différentiable, non linéaire.

Au point courant $x_k \in \mathbb{R}^n$, on suppose donné un modèle de la variation de f pour un incrément d de x_k . En optimisation différentiable, il est raisonnable de considérer un modèle quadratique de la forme

$$f(x_k + d) \simeq \Psi_k(d) \simeq f(x_k) + g_k^T d + \frac{1}{2} d^T H_k d \quad (5.2)$$

on note par $g_k = \nabla f(x_k)$ le gradient de f en x_k et par $H_k = \nabla^2 f(x_k)$ le hessien de f en x_k . g_k et H_k sont supposés être calculés par le produit scalaire euclidien $\langle u, v \rangle = \sum_i u_i v_i$.

On considère que Ψ_k est un modèle de la variation de f qui est acceptable dans un voisinage de la forme :

$$\overline{B}(0, \Delta_k) = \{x \in \mathbb{R}^n : \|x\| \leq \Delta_k\},$$

où $\Delta_k > 0$ et $\|\cdot\|$ est la norme euclidienne. Ce domaine est appelé **la région de confiance du modèle** Ψ_k et Δ_k est appelé le **rayon de confiance**.

Pour trouver l'accroissement d_k à donner à x_k , on minimise le modèle quadratique Ψ_k sur la région de confiance. On doit donc résoudre le sous-problème quadratique suivant :

$$(P_1) \begin{cases} \min \Psi_k(d) \\ \|d\| \leq \Delta_k . \end{cases} \quad (5.3)$$

Une approche possible pour résoudre des problèmes avec contraintes, est la **méthode de pénalité**, on introduit les contraintes dans l'objectif sous forme de pénalités, on part d'une solution non admissible et on essaye d'imposer à l'optimum non contraint d'arriver dans l'ensemble admissible.

En résolvant une suite de problème de la forme :

$$(P_2) \begin{cases} \min_{x \in \mathbb{R}^n} \Phi(d, \mu_k) = \min_{x \in \mathbb{R}^n} \{ \Psi(d) + \mu_k p(d) \} \end{cases} \quad (5.4)$$

La suite des valeurs $\{\mu_k\}$ a les propriétés suivantes :

- i) $\mu_k > 0, \forall k$
 - ii) La suite est croissante (c'est-à-dire $\mu_{k+1} > \mu_k$)
 - iii) la suite diverge (c'est-à-dire $\mu_k \rightarrow \infty$ quand $k \rightarrow \infty$)
- et les p_i sont des fonctions de pénalité telles que :

$$p_i(d) = \begin{cases} = 0 & \text{si } \|d\| \leq \Delta \\ > 0 & \text{sinon} \end{cases} \quad (5.5)$$

On choisit :

$$p_i(d) = (\max\{0, \|d\| - \Delta\})^2$$

5.1 Algorithme 5.1.

Etape d'initialisation

Choix des constantes $\eta_1, \eta_2, \gamma_1, \gamma_2$ et γ_3 telles que

$$0 < \eta_1 < \eta_2 < 1 \text{ et } 0 < \gamma_1 < \gamma_2 < 1 \leq \gamma_3. \quad (5.6)$$

Choix d'un rayon de confiance Δ_0 et d'un itéré x_0 . Calcul de $f(x_0)$. Poser $k = 0$ et aller à l'étape 1.

CHAPITRE 5. MÉTHODE À RÉGION DE CONFIANCE ET
GRADIENT CONJUGUÉ

Etape 1

Construction de la fonction modèle $\Phi(d, \mu_k)$, conformément à la relation (4), et détermination d'une solution approchée d_k du problème (P_2)

$$(P_2) \left\{ \min_{d \in \mathbb{R}^n} \Phi(d, \mu_k) \right.$$

Etape 1.0

Former un problème pénalisé $\phi_k(d, \mu_k)$ et choisir $\mu_0 > 0, \beta > 1, s_0$ le point de départ, $M_0 = \nabla \phi(s_0)$, poser $s_0 = -M_0$. poser $k = 0$ et aller à l'étape 1.1.

Etape 1.1

Si $M_k = 0$: stop ($d^* = d_k$). "Test d'arrêt".

Sinon aller à l'étape 1.2.

Etape 1.2

poser $\mu_{k+1} = \beta \mu_k$

Définir $d_{k+1} = d_k + \alpha_k s_k$ avec :

α_k vérifie les conditions de Wolfe faibles

$$s_k = -M_{k+1} + B_{k+1} s_k$$

$$B_{k+1}^{DY} = \frac{\|M_{k+1}\|^2}{s_k^T [M_{k+1} - M_k]} = \frac{\|M_{k+1}\|^2}{s_k^T y_k}$$

poser $k = k + 1$ et aller à l'étape 1.1.

Etape 2

Calcul de $f(x_k + d_k)$ et évaluation du critère de performance

$$\rho_k = \frac{f(x_k) - f(x_k + d_k)}{\Psi(x_k) - \Psi(x_k + d_k)}.$$

Etape 3

Actualisation du rayon de confiance :

(a) Si $\rho_k \geq \eta_2$:

L'étape est un succès et le pas est accepté. $x_{k+1} = x_k + d_k$ et choix de $\Delta_k \in [\Delta_k, \gamma_3 \Delta_k]$.

(b) Si $\eta_1 \leq \rho_k \leq \eta_2$:

L'étape est un succès et le pas est accepté. $x_{k+1} = x_k + d_k$ et choix de $\Delta_k \in [\gamma_2 \Delta_k, \Delta_k]$.

(c) Si $\rho_k < \eta_1$:

L'étape est un échec et le pas est rejeté. $x_{k+1} = x_k$ et choix de $\Delta_k \in [\gamma_1 \Delta_k, \gamma_2 \Delta_k]$.

Etape 4

Si un critère de convergence est vérifié, l'algorithme est arrêté. Sinon, on pose $k = k + 1$ et on retourne à l'étape 1.

On choisit habituellement $\eta_1 = 0,25$, $\eta_2 = 0,75$, $\gamma_1 = 0,25$, $\gamma_2 = 0,75$ et $\gamma_3 = 2$.

5.2 Résultats de convergence

De manière similaire aux méthodes de recherches linéaires, la détermination de pas optimaux n'est pas une condition nécessaire pour obtenir une convergence globale. Sous certaines conditions, une bonne approximation de ces pas peut être acceptable. Il suffit de déterminer une solution approchée, d_k , à l'intérieur de la région de confiance, qui produise une réduction suffisante de la fonction modèle. Cette réduction peut être obtenue par la méthode du point de Cauchy d_k^c

5.2.1 Le point de Cauchy

Définition 5.2.1 On appelle point de Cauchy du sous-problème quadratique (RC_k), le point noté d_k^c solution de :

$$\begin{cases} \min \Psi_k(d) \\ \|d\| \leq \Delta_k . \\ d = -\alpha g_k, \alpha \in \mathbb{R} \end{cases}$$

c'est donc le point minimisant Ψ_k dans la région de confiance, c'est-à-dire le long de la droite de la plus forte pente de Ψ_k .

Proposition 5.2.1 Le point de Cauchy d_k^c est unique, il est donné par :

$$d_k^c = \begin{cases} 0 & \text{si } g_k = 0 \\ -\frac{\Delta_k}{\|g_k\|} g_k & \text{si } g_k \neq 0 \text{ et } g_k^T H_k g_k \leq 0 \\ -\frac{\Delta_k}{\|g_k\|} g_k \min \left(\frac{\|g_k\|^3}{\Delta_k g_k^T H_k g_k}, 1 \right) & \text{sinon} \end{cases} \quad (5.7)$$

Preuve

En effet, le cas où $g_k = 0$ est clair. Supposons à présent que $g_k \neq 0$.

Si $g_k^T H_k g_k \leq 0$, alors $\alpha \mapsto \Psi_k(-\alpha g_k)$ est concave et $\nabla \Psi_k(0)^T(-g_k) = -\|g_k\|^2 < 0$, si bien qu'il faut donc prendre α le plus grand possible tout en gardant $\|-\alpha g_k\| \leq \Delta_k$. Ceci donne $\alpha = \frac{\Delta_k}{\|g_k\|}$.

Il reste à examiner le cas où $g_k^T H_k g_k > 0$. Le résultat se déduit du fait que, dans ce cas, le minimum de $\alpha \mapsto \Psi_k(-\alpha g_k)$ est obtenu en $\alpha = \frac{\|g_k\|^2}{g_k^T H_k g_k}$.

5.2.2 Condition de décroissance suffisante

Comme nous venons de le voir, le point de Cauchy d_k^c permet de valider ou non une solution approchée du sous problème à région de confiance. Donc le point d^{DY} doit vérifier la décroissance de la fonction modèle :

$$\Psi_k(0) - \Psi_k(d_k^{DY}) \geq c_1 (\Psi_k(0) - \Psi_k(d_k^c)). \quad c_1 > 0 \quad (5.8)$$

La proposition suivante est connue sous le nom de **Condition de Powell**.

Proposition 5.2.2 *Le point de Cauchy d_k^c vérifie la condition de Powell avec $C = \frac{1}{2}$, tel que :*

$$\Psi_k(0) - \Psi_k(d_k^c) \geq C \|g_k\| \min \left\{ \frac{\|g_k\|}{\|H_k\|}, \Delta_k \right\}. \quad (5.9)$$

Preuve

On considère deux cas pour la matrice hessienne H_k :

a) $g_k^T H_k g_k \leq 0$:

$$\begin{aligned} \Psi_k(d_k^c) - \Psi_k(0) &= \Psi_k\left(-\frac{\Delta_k}{\|g_k\|} g_k\right) - f_k \\ &= -\frac{\Delta_k}{\|g_k\|} \|g_k\|^2 + \frac{1}{2} \frac{\Delta_k^2}{\|g_k\|^2} g_k^T H_k g_k \\ &\leq -\frac{\Delta_k}{\|g_k\|} \|g_k\|^2 \\ &\leq -\Delta_k \|g_k\| \\ &\leq -\|g_k\| \min \left\{ \frac{\|g_k\|}{\|H_k\|}, \Delta_k \right\} \\ &\leq -\frac{1}{2} \|g_k\| \min \left\{ \frac{\|g_k\|}{\|H_k\|}, \Delta_k \right\} \end{aligned}$$

donc, on a (5.9).

b) $\mathbf{g}_k^T \mathbf{H}_k \mathbf{g}_k > 0$ et $\frac{\|g_k\|^3}{\Delta_k \mathbf{g}_k^T \mathbf{H}_k \mathbf{g}_k} \leq 1$:

$$\begin{aligned} \Psi_k(d_k^c) - \Psi_k(0) &= -\frac{\|g_k\|^4}{\Delta_k \mathbf{g}_k^T \mathbf{H}_k \mathbf{g}_k} + \frac{1}{2} g_k^T H_k g_k \frac{\|g_k\|^4}{(\mathbf{g}_k^T \mathbf{H}_k \mathbf{g}_k)^2} \\ &= -\frac{1}{2} \frac{\|g_k\|^4}{\mathbf{g}_k^T \mathbf{H}_k \mathbf{g}_k} \\ &\leq -\frac{1}{2} \frac{\|g_k\|^4}{\|\mathbf{H}_k\| \|\mathbf{g}_k\|^2} \\ &\leq -\frac{1}{2} \frac{\|g_k\|^2}{\|\mathbf{H}_k\|} \\ &\leq -\frac{1}{2} \|g_k\| \min \left\{ \frac{\|g_k\|}{\|\mathbf{H}_k\|}, \Delta_k \right\} \end{aligned}$$

donc on a (5.9).

c) $\mathbf{g}_k^T \mathbf{H}_k \mathbf{g}_k < \frac{\|g_k\|^3}{\Delta_k}$, $d_k^c = -\frac{\Delta_k}{\|g_k\|} g_k$

on a (5.9).

Théorème 5.2.1 Soit d_k^{DY} un vecteur quelconque tel que $\|d_k^{DY}\| \leq \Delta_k$ et

$$\Psi_k(0) - \Psi_k(d_k^{DY}) \geq c_2 (\Psi_k(0) - \Psi_k(d_k^c)) \quad (5.10)$$

Alors d_k^{DY} satisfait (5.9) avec $c_2 = \frac{c_1}{2}$.

preuve

Comme $\|d_k^{DY}\| \leq \Delta_k$ et d'après (5.8), on a :

$$\Psi_k(0) - \Psi_k(d_k^{DY}) \geq c_1 (\Psi_k(0) - \Psi_k(d_k^c)) \geq \frac{1}{2} c_1 \|g_k\| \min \left\{ \frac{\|g_k\|}{\|\mathbf{H}_k\|}, \Delta_k \right\}$$

En particulier, si d_k^{DY} est la solution exacte de (RC_k) alors, elle satisfie (5.9) avec $c_2 = \frac{1}{2}$.

5.2.3 Convergence des méthodes de pénalité

Les 2 lemmes suivants servent à démontrer la convergence de la méthode. Dénotons par d_k la solution de $\min \Phi(d, \mu_k)$.

Lemme 5.2.1 pour toute valeur de k ,

- i) $\Phi(d_k, \mu_k) \leq \Phi(d_{k+1}, \mu_{k+1})$
- ii) $p(d_k) \geq p(d_{k+1})$
- iii) $\Psi(d_k) \leq \Psi(d_{k+1})$

Preuve

$$\begin{aligned} \text{i) } \Phi(d_{k+1}, \mu_{k+1}) &= \Psi(d_{k+1}) + \mu_{k+1}p(d_{k+1}) \\ &\geq \Psi(d_{k+1}) + \mu_k p(d_{k+1}) && \text{(puisque } \mu_{k+1} > \mu_k \text{ et} \\ p(d_{k+1}) \geq 0) \\ &\geq \Psi(d_k) + \mu_k p(d_k) && \text{(puisque } d_k \text{ minimise} \\ \Phi(d, \mu_k)) \end{aligned}$$

Donc, $\Phi(d_{k+1}, \mu_{k+1}) \geq \Phi(d_k, \mu_k)$.

$$\begin{aligned} \text{ii) } \Psi(d_k) + \mu_k p(d_k) &\leq \Psi(d_{k+1}) + \mu_k p(d_{k+1}) && \text{(puisque } d_k \text{ minimise} \\ \Phi(d, \mu_k)) \\ \text{et} \end{aligned}$$

$$\begin{aligned} \Psi(d_{k+1}) + \mu_{k+1}p(d_{k+1}) &\leq \Psi(d_k) + \mu_{k+1}p(d_k) && \text{(puisque } d_{k+1} \text{ minimise} \\ \Phi(d, \mu_{k+1})) \\ \text{additionnant les deux inégalités,} \end{aligned}$$

$$(\mu_{k+1} - \mu_k)p(d_{k+1}) \leq (\mu_{k+1} - \mu_k)p(d_k)$$

et comme $\mu_{k+1} > \mu_k$, on obtient :

$$p(d_{k+1}) \leq p(d_k)$$

$$\begin{aligned} \text{iii) } \Psi(d_{k+1}) + \mu_k p(d_{k+1}) &\geq \Psi(d_k) + \mu_k p(d_k) && \text{(puisque } d_k \text{ minimise} \\ \Phi(d, \mu_k)) \end{aligned}$$

en utilisant ii) et le fait que $\mu_k \geq 0$, on obtient :

$$\Psi(d_{k+1}) + \mu_k p(d_{k+1}) \geq \Psi(d_k) + \mu_k p(d_{k+1})$$

donc, $\Psi(d_{k+1}) \geq \Psi(d_k)$.

Lemme 5.2.2 *Soit d^* une solution optimale de (p_1) . Alors pour chaque k ,*

$$\Psi(d^*) \geq \Phi(d_k, \mu_k) \geq \Psi(d_k)$$

Preuve

$$\begin{aligned} \Psi(d^*) &= \Psi(d^*) + \mu_k p(d^*) \text{ pour tout } k \text{ puisque } p(d^*) = 0 \\ &\geq \Psi(d_k) + \mu_k p(d_k) = \Phi(d_k, \mu_k) \forall k \text{ (puisque } d_k \text{ minimise } \Phi(d, \mu_k)) \\ &\geq \Psi(d_k) \forall k \text{ (puisque } \mu_k \geq 0 \text{ et } p(d_k) \geq 0. \end{aligned}$$

Donc, $\Psi(d^*) \geq \Phi(d_k, \mu_k) \geq \Psi(d_k) \forall k$.

Théorème 5.2.2 *On suppose que d_k est un minimum global exact de (p_2) et $\{\mu_k\}$ une suite divergente. Alors tout point limite d^* de la suite $\{d_k\}$ est une solution du problème (p_1) .*

Preuve

Soit \bar{d} une solution de (p_1) , telle que

$$\Psi(\bar{d}) \leq \Psi(d) \text{ pour tout } \|d\| \leq \Delta$$

puisque d_k minimise $\Phi(d, \mu_k)$ pour tout k , on a alors,

$$\Phi(d_k, \mu_k) \leq \Phi(\bar{d}, \mu_k)$$

donc

$$\Psi(d_k) + \mu_k p(d_k) \leq \Psi(\bar{d}) + \mu_k p(\bar{d}) = \Psi(\bar{d}) \quad (\text{puisque } p(\bar{d}) = 0)$$

alors,

$$p(d_k) \leq \frac{1}{\mu_k} [\Psi(\bar{d}) - \Psi(d_k)] \quad (5.11)$$

On suppose que d^* est le point limite de $\{d_k\}$. Alors il existe une infinité de sous-suite K tel que

$$\lim_{k \in K} d_k = d^*$$

En prenant la limite de l'inégalité (5.11), on obtient :

$p(d^*) = \lim_{k \in K} p(d_k) \leq \lim_{k \in K} \frac{1}{\mu_k} [\Psi(\bar{d}) - \Psi(d_k)] = 0$ (puisque $\{\mu_k\}$ est une suite divergente)

Donc, on a $p(d^*) = 0$, et par conséquent $\|d^*\| \leq \Delta$.

Le fait que d^* soit optimal découle directement du lemme 5.2. En effet la relation $\Psi(d_k) \leq \Psi(d^*)$ étant vérifiée pour tout k , il s'ensuit que

$$\Psi(d^*) = \lim_{k \in K} \Psi(d_k) \leq \Psi(\bar{d})$$

5.2.4 Résultats de convergence

Théorème 5.2.3 *Soit $f : \mathbb{R}^n \rightarrow \mathbb{R}$ une fonction continûment dérivable sur*

$$X = \{x \in \mathbb{R}^n : f(x) < f(x_0)\},$$

et soit $\{x_k\}$ une suite générée par l'algorithme 5.1, avec H_k uniformément bornée. Si de plus le pas d_k vérifie la condition (5.8), on a alors :

$$\liminf_{k \rightarrow \infty} \|\nabla f_k\| = 0. \quad (5.12)$$

Preuve

En supposant qu'il existe un indice k_1 et une constante $\gamma > 0$ tels que

$$\|\nabla f_k\| \geq \gamma, \forall k \geq k_1 \quad (5.13)$$

Montrons que

$$\Delta_k \rightarrow 0 \quad \text{et} \quad \sum_{k \geq 1} \|d_k\| < \infty \quad (5.14)$$

D'après la condition de Powell (5.9) et du fait que $\{H_k\}$ soit bornée, on a :

$$\begin{aligned} \Psi_k(d_k) &\leq -C \|\nabla f_k\| \min \left\{ \frac{\|\nabla f_k\|}{\|H_k\|}, \Delta_k \right\} \\ &\leq -C \min \{1, \Delta_k\} \end{aligned}$$

où C est une constante strictement positive indépendante de k . En utilisant l'étape 3 de l'algorithme 5.1 ($\rho_k \geq \eta_2$), on obtient :

$$f(x_k + d_k) \leq f(x_k) + \omega_1 \Psi_k(d_k)$$

Ce qui s'écrit encore :

$$f(x_{k+1}) \leq f(x_k) - C \min \{1, \Delta_k\},$$

$$C \min \{1, \Delta_k\} \leq f(x_k) - f(x_{k+1}).$$

Comme f est bornée inférieurement, la suite $\{f(x_k) - f(x_{k+1})\}$ est convergente et on déduit de l'inégalité ci-dessus que $\sum_{k \geq 1} \Delta_k < \infty$, et comme $\|d_k\| \leq \Delta_k$, les assertions de (5.14) s'en déduisent.

Montrons maintenant que le ratio $\rho_k \rightarrow 1$.

Comme f est bornée et $\|d_k\| \leq \Delta_k$ alors :

$$\frac{\left| f(x_k + d_k) - f(x_k) - \nabla f(x_k)_k^T d \right|}{\|d_k\|} \rightarrow 0 \quad (5.15)$$

d'autre part, comme $\{H_k\}$ est bornée, on a :

$$\left| \Psi_k(d_k) - \nabla f(x_k)_k^T d \right| \leq c \|d_k\|^2 \quad (5.16)$$

Enfin, on peut écrire successivement :

$$\rho_k = \frac{f(x_{k+1}) - f(x_k)}{\Psi_k(d_k)} = \frac{f(x_k + d_k) - f(x_k) - \nabla f(x_k)_k^T d}{\Psi_k(d_k)} + \frac{\nabla f(x_k)_k^T d - \Psi_k(d_k)}{\Psi_k(d_k)} + 1$$

$$|\rho_k - 1| \leq \frac{\left| f(x_k + d_k) - f(x_k) - \nabla f(x_k)_k^T d \right|}{|\Psi_k(d_k)|} + \frac{|\nabla f(x_k)_k^T d - \Psi_k(d_k)|}{|\Psi_k(d_k)|}.$$

D'après (5.15), $\Delta \rightarrow 0$ et $\|s_k\| \leq \Delta_k$, on a :

$$|\Psi_k(d_k)| \geq c \|d_k\|.$$

Alors combinant (5.15) et (5.16), on obtient :

$$|\rho_k - 1| \leq \frac{c \|d_k\|^2}{c \|d_k\|} \leq \|d_k\|$$

Ceci montre que $\rho_k \rightarrow 1$.

Dès lors $\rho_k > \eta_1$ pour k assez grand. D'après la règle de mise à jour de Δ_k (l'étape 3 de l'algorithme 5.1), cela implique $\Delta_k > \Delta > 0$, ce qui contredit la première assertion de (5.14).

Cette contradiction prouve le théorème.

■

Théorème 5.2.4 *Supposons, en plus des hypothèses et des conditions du théorème 5.2.3, que ∇f_k est continu au sens de Lipchitz et inférieurement borné sur $X = \{x \in \mathbb{R}^n : f(x) < f(x_0)\}$, on a alors :*

$$\lim_{k \rightarrow \infty} \|\nabla f_k\| = 0. \quad (5.17)$$

Preuve

On suppose qu'il existe une constante $\varepsilon > 0$ et une sous-suite $\{x_k\}_{k \geq m} \subset B(x_m, R)$ où $(B(x_m, R))$ est la boule fermée de centre x_m et de rayon R , pour laquelle on a

$$\|\nabla f_k\| \geq \varepsilon, \quad \text{pour tout } k \geq m. \quad (5.18)$$

Soit $l \geq m$ tel que x_{k+1} soit le premier itéré après x_m à l'extérieur de $B(x_m, R)$. Puisque $\|\nabla f_k\| \geq \varepsilon$ pour tout $k = m, m+1, \dots, l$, on peut écrire que

$$f(x_m) - f(x_{l+1}) = \sum_{k=m}^l f(x_k) - f(x_{k+1})$$

En utilisant la condition de Powell (5.9) et le fait que $\{H_k\}$ soit bornée, on a comme dans la démonstration du théorème 5.2.3 :

$$f(x_{k+1}) - f(x_k) \leq -C \min \{ \|\nabla f_k\|, \Delta_k \}.$$

Donc pour tout $k \geq m$, on a :

$$f(x_{l+1}) - f(x_m) \leq -C \min \{1, \Delta_k\}.$$

Comme $f(x_{l+1}) - f(x_m) \rightarrow 0$, $\min \{1, \Delta_k\} = \Delta_k$ pour k assez grand. En utilisant le fait que $\|d_k\| \leq \Delta_k$, on a finalement :

$$\|d_k\| \leq C (f(x_k) - f(x_{l+1})), \quad k = m, m+1, \dots, l.$$

$$\|x_{l+1} - x_m\| \leq \sum_{k=m}^l \|d_k\| \leq C (f(x_m) - f(x_{l+1}))$$

Donc $\|x_{l+1} - x_m\| \rightarrow 0$. Alors par uniforme continuité de ∇f_k , on a :

$$\|f(x_{l+1}) - f(x_m)\| \rightarrow 0.$$

Ceci est en contradiction avec (5.18).
d'où le théorème.

■

5.3 Tests numériques

Nous avons effectué les tests numériques sur l'algorithme des méthodes à régions de confiance pénalisées avec les fonctions suivantes :

Problème 1 : fonction de Rosenbrock généralisée

$$f(x) = \sum_{i=1}^{n-1} 100(x_i^2 - x_{i+1})^2 + (x_i - 1)^2$$

Problème 2 : fonction de Dixon généralisée

$$f(x) = (x_1 - 1)^2 + \sum_{i=2}^n i(2x_i^2 - x_{i-1})^2$$

Problème 3 : fonction de Oren

$$f(x) = \left[\sum_{i=1}^n ix_i^2 \right]^2$$

Problème 4 : la fonction de Powell

$$f(x) = \sum_{i=1}^{n/4} \left[(x_{4i-3} + 10x_{4i-2})^2 + 5(x_{4i-1} - x_{4i})^2 + (x_{4i-2} - 2x_{4i-1})^4 + 10(x_{4i-3} - x_{4i})^4 \right]$$

Dans tout les tests le critère d'arrêt de l'algorithme est $\varepsilon = 10^{-5}$. L'algorithme des régions de confiance est appliqué avec $\Delta_0 = 0.5$, $\eta_1 = 0,25$, $\eta_2 = 0,75$, $\gamma_1 = 0,25$, $\gamma_2 = 0,75$ et $\gamma_3 = 2$. L'algorithme du gradient conjugué est appliqué avec la recherche linéaire de wolfe avec les paramètres : $m_1 = 0.3$, $m_2 = 0.7$. la méthode de pénalité est appliquée avec $\mu_0 = 2$.

Les résultats présentant nombre d'itération et le temps nécessaire pour la convergence, sont illustrés dans les tableaux suivant :

Dimension n	iteration k	$f(x_k)$	$\ g(x_k)\ $	Time of convergence
2	26	$2,59.10^{-12}$	$9,56.10^{-6}$	$4,68.10^{-2}$ s
10	29	$4,82.10^{-11}$	$8,71.10^{-6}$	0,26 s
30	29	$8,16.10^{-13}$	$1,27.10^{-6}$	2,51 s
50	28	$2,72.10^{-11}$	$7,91.10^{-6}$	5,47 s
80	30	3.10^{-11}	$6,46.10^{-6}$	20,57 s
100	28	$2,71.10^{-11}$	$8,59.10^{-6}$	22 s
150	27	$5,67.10^{-12}$	$6,53.10^{-6}$	50,66 s
200	25	$2,11.10^{-13}$	$5,46.10^{-6}$	129,62 s

Tableau 1 : Problème 1

CHAPITRE 5. MÉTHODE À RÉGION DE CONFIANCE ET
GRADIENT CONJUGUÉ

Dimension n	iteration k	$f(x_k)$	$\ g(x_k)\ $	Time of convergence
2	8	$5,62.10^{-12}$	$8,72.10^{-6}$	$1,56.10^{-2}$ s
6	25	$1,90.10^{-12}$	$9,97.10^{-6}$	1,06 s
10	41	0,66	$8,43.10^{-6}$	1,29 s
30	68	0,66	$9,79.10^{-6}$	2,72 s
50	111	0,66	$9,45.10^{-6}$	6,83 s
100	36	0,66	$8,84.10^{-6}$	39,18 s

Tableau 2 : Problème 2

Dimension n	iteration k	$f(x_k)$	$\ g(x_k)\ $	Time of convergence
2	7	$2,68.10^{-8}$	$9,84.10^{-6}$	$1,56.10^{-2}$ s
10	19	$1,10.10^{-8}$	$9,66.10^{-6}$	0,60 s
50	23	$4,20.10^{-9}$	$9,74.10^{-6}$	103,13 s
80	25	$3,13.10^{-8}$	$9,94.10^{-6}$	470 s
100	25	$2,52.10^{-9}$	$9,32.10^{-6}$	967 s

Tableau 3 : Problème 3

Dimension n	iteration k	$f(x_k)$	$\ g(x_k)\ $	Time of convergence
4	20	$1,15.10^{-8}$	$9,90.10^{-6}$	0,15 s
8	53	$1,76.10^{-8}$	$8,89.10^{-6}$	0,24 s
16	79	$9,38.10^{-9}$	$9,28.10^{-6}$	1,04 s
32	18	$1,25.10^{-8}$	$4,40.10^{-6}$	2,72 s
100	19	$2,02.10^{-8}$	$9,30.10^{-6}$	31,90 s
200	20	$3,86.10^{-8}$	$5,92.10^{-6}$	138,37 s

Tableau 4 : Problème 4

Conclusion et perspectives

Dans cette partie nous allons faire un récapitulatif de ce travail en tirant les conclusions et à faire de perspectives.

Dans ce travail nous avons explicité les méthodes à région de confiance comme outil pour résoudre les problèmes d'optimisation sans contraintes. Les résultats obtenus sont des résultats de convergence qui utilisent les méthodes à région de confiance et qui emploient comme sous programme le gradient conjugué non linéaire de Dai Yuan à chaque itération. De plus, on a pénalisé les problèmes quadratiques avec contraintes et on les a converti en une série de problèmes sans contraintes.

L'originalité dans cette thèse est que le gradient conjugué utilisé dans le sous programme est non linéaire alors que d'habitude il est linéaire. Notre choix pour le gradient conjugué de Dai Yuan est que dans ce cas les résultats de convergence sont faibles lorsqu'on utilise les recherches linéaires inexactes de Wolfe.

Problème ouvert

Dans notre travail on a étudié la convergence des méthodes à région de confiance et gradient conjugué dans le cas où la matrice hessienne est définie positive, il est intéressant de voir le cas où la matrice hessienne n'est pas définie positive.

Bibliographie

- [1] Andrew R. Conn , Nicholas I. M. Gould and Philippe L. Toint, Trust-Region Methods, MPS-SIAM Series on Optimization, United States(2000).
- [2] L. Armijo, Minimization of functions having Lipschitz-continuous first partial derivatives, Pacific J. Math.. 16 :1-3(1966).
- [3] F. Bastin, V. Malmedy, M. Mouffe, Ph. L. Toint and D. Tomanos, A Retrospective Trust-Region Method for Unconstrained Optimization, Mathematical Programming, vol.123, issue 2(2007).
- [4] M. Bazaraa, N. Sherali and C. Shetty, Nonlinear Programming, Theory and Applications, John Wiley & Sons Second ed, New York(1993).
- [5] R. H. Byrd, J-C. Gilbert and J. Nocedal, A trust region method based on interior point techniques for nonlinear programming, Mathematical Programming, 89(2000) , pp. 149-185. [dx.doi.org/10.1016/B978-0-12-597050-1.50006-3](https://doi.org/10.1016/B978-0-12-597050-1.50006-3).
- [6] R. H. Byrd, R.B. Schnabel, and G.A.Shultz. Approximate solution of the trust region problem by minimization over two dimensional subspaces. Math. Programming, 40 :247-263, (1988).
- [7] A. Cauchy, Analyse mathématique, Méthode générale pour la résolution des systèmes d'équations simultanées, Comptes Rendus de l'Académie des Sciences de Paris, t-25, pp. 536-538.(1847).
- [8] C. W. Carroll, The created reponse surface technique for optimizing nonlinear restrained systems, Operations Res., Vol. 9(1961), pp. 169-84.
- [9] A. R. Conn, N. I. M. Gould, and P. L. Toint, Trust-Region Methods, MPS-SIAM Series on Optimization, SIAM, (2000).
- [10] David G. Luenberger, Optimization by Vector Space Methods, Wiley Professional Paperback Series, New york(1969).

-
- [11] Y. H. Dai and Y. Yuan, Convergence properties of the Fletcher-Reeves method, *IMA J Numer. Anal.*, Vol.16(2)(1996), pp. 155-164.
- [12] Y. H. Dai and Y. Yuan, Convergence properties of the conjugate descent method, *Advances in Mathematics*, 6(1996), pp.552-562.
- [13] Y. H. Dai and Y. Yuan, Some properties of a new conjugate gradient method, in : *Advances in Nonlinear Programming*, ed . Kluwer Academic, Boston(1998), pp. 251-262.
- [14] Y.H. Dai and Y. Yuan, A non linear conjugate gradient with a strong global convergence property, *SIAM J. Optimization*, Vol. 10(1)(1999) pp.177-182.
- [15] H. Dai and Y. Yuan, New properties of a new conjugate gradient method, *Numer. Math.*, Vol. 89(2)(2001), pp. 83-98.
- [16] Y. H. Dai and Y. Yuan, An Efficient Hybrid Conjugate Gradient Method for Unconstrained Optimization, Chinese Academy of Sciences, P.O. Box 2719, Beijing 100080, P.R. China(2001).
- [17] Y. H. Dai and Y. Yuan, A note on the nonlinear conjugate gradient method *Engineering Computing*, Chinese Academy of Sciences, P. O. Box 2719, Beijing 100080, China(2002).
- [18] J. E. Dennis and R.B. Schnabel. *Numerical Methodes for Unconstrained Optimization and Nonlinear Equations*. Prentice-Hall, Englewood Cliffs, New Jersey, (1983).
- [19] J. E. Dennis and R.B. Schnabel. A view of unconstrained optimization. In G.L. Nemhauser, A.H.G. Rinnooy Kan, and M.J. Todd, editors, *Handbooks in Operations Research and Management Science*. North Holland, Amsterdam, (Vol. 1, Optimization)(1988).
- [20] T. Diallo, Travail pratique de master, Sujet : Etude et Illustration De Méthodes Itératives D'optimisation Non Linéaire, Responsable : Benjamin Leroy-Beaubair, Ecole polytechnique Fédérale de Lausanne(2006).
- [21] Edwin K. P. Chong, Stanislaw H. Zak. *An Introduction to Optimization*, Wiley-Interscience Series in Discrete Mathematics and Optimization second edition, New York(2001).
- [22] R. Fletcher, *Practical methods of optimization*, JohnWiley & Sons, second édition, (1987).
- [23] R. Fletcher and M. J. D. Powell, A rapidly convergent descent method for minimization, *Computer. J.*, 6(1963), pp. 163-168.

-
- [24] D.M.Gay. Computing optimal locally constrained steps. *SIAM J. Sci. Statist. Comput.*, 2(1981),pp.186-197.
- [25] J. C. Gilbert, *Eléments d'Optimisation Différentiable : Théorie et Algorithmes*, Notes de cours, École Nationale Supérieure de Techniques Avancées, Paris(2007).
- [26] J.C. Gilbert and J. Nocedal, Global convergence properties of conjugate gradient methods for optimization, *SIAM J. Optimization*. Vol. 3, No.1(1992), pp. 21-42.
- [27] A. A. Goldstein, On steepest descent, *SIAM J. on Control A*, Vol. 3, No. 1(1965), pp. 147-151.
- [28] A.A. Goldstein, *Constructive Real Analysis*, A Harper International Edition(1967).
- [29] A.A. Goldstein and J.F. Price, An effective algorithm for minimzation, *Num.Math.*, 10(1969), pp. 184-189.
- [30] D. Hachelfi, Y. Laskri and Mohamed Lamine Sahari, Trust region with nonlinear conjugate gradient method, *IJPAM*, Vol. 99, No. 3(2015), pp. 299-313.
- [31] Kuhn, H.W. et Tucker, A.W. Nonlinear programming. Dans J. Neyman, rédacteur, *Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability*, University of California Press,Berkeley, (1951),pages 481–492.
- [32] M. Minoux, *Programmation Mathématique, Théorie et Algorithmes*, tome 1(1983), Dunod.
- [33] J.J.Moré, B.S. Garbow, and K. E. Hillstrom, Testing Unconstrained Optimization Software, *ACM Transactions on Mathematical Software*, Vol 7, No.1(1981), pp.17-41. ACM0098-3500/81/0300-0017.
- [34] J. J. Moré. Recent developments in algorithms and software for trust regions methods. In A. Bachem, M. Grotscchel, and B. Korte, editors, *Mathematical programming. The state of art*, pages 258-287. Springer Verlag, New York,(1983).
- [35] J. J. Moré, and D. C. Sorensen, On the use of directions of nonnegative curvature in a modified Newton method, *Mathematical Programming*,16 (1979), pp. 1-20.
- [36] J. J. Moré, and D. C. Sorensen. Computing a trust region step. *SIAM J. Sci. Statist. Comput.*, 4 :553-572, (1983).

-
- [37] . J. Moré, and D. Thuente. Line search algorithms with guaranteed sufficient decrease. *ACM Trans. Math. Software*, 20 :286-307,(1994).
- [38] S. G. Nash and A. Sofer. *Linear and Nonlinear Programming*. McGraw-Hill, New York(1996).
- [39] J. Nocedal. Theory of algorithms for unconstrained optimization. *Acta Numerica*, pages 199-242(1992).
- [40] J. Nocedal, and S. J. Wright, *Numerical Optimization*, Springer Series in Operations Research and Financial Engineering, second edition, New York(1999).
- [41] J. M. Ortega and W. C. Rheinboldt. *Iterative Solution of Nonlinear Equations in Several Variables*. Academic Press, New York(1970).
- [42] M. J. D. Powell, A FORTRAN subroutine for unconstrained minimization requiring first derivatives of the objective function. Tech. Rep AERE-R. 6469, Mathematics Branch, A. E. R. E. Harwell, berkshire, England(1970).
- [43] M. J. D. Powell, A New Algorithm for Unconstrained Optimization, In : J. B. Rosen, O. L. Mangasarian and K. Ritter, Eds., *Nonlinear Programming*, Academic Press, New York (1970). [dx.doi.org/10.1007/PL00011391](https://doi.org/10.1007/PL00011391).
- [44] M. J. D. Powell, Convergence Properties of a Class of Minimization Algorithms, In : O. L. Mangasarian, R. R. Meyer and S. M. Robinson, Eds., *Nonlinear Programming*, Academic Press, pp. 1-27, New York(1975).
- [45] M. J. D. Powell. On the global convergence of trust region algorithms for unconstrained optimization. *Math. Programming*, 29 :297-303,(1984).
- [46] M.J.D. Powell, Non convex minimization calculation and the conjugate gradient method, in : *Lecture Notes in Mathematics 1066*(Springer, Berlin), pp. 122-141(1984).
- [47] G. A. Shultz, R. B. Schnabel, and R. H. Byrd. A family of trust region based algorithms for unconstrained optimization with strong global convergence properties. *SIAM J. Numer. Anal.*,22 :47-67,(1985).
- [48] D. C. Sorensen. Newton's method with a model trust region modification. *SIAM J. Numer. Anal.*, 19 :409-426,(1982).
- [49] D. C. Sorensen. Minimization of a large scale quadratic function subject to an ellipsoidal constraint. Technical Report TR94-27, Department of computational and Applied Mathematics, Rice University,(1994).

- [50] T. Steihaug. The conjugate gradient method and trust regions in large scale optimization. *SIAM J. Numer. Anal.*, 20 :626-637,(1983).
- [51] S. W. Thomas. Sequential Estimation Techniques for Quasi-Newton Algorithms. PhD thesis, Cornell University, Ithaca, New York,(1975).
- [52] PH. L. Toint. Towards an efficient sparsity exploiting Newton method for minimization. In I. S. Duff, editor, *Sparse Matrices and Their Uses*, pages 57-87. Academic Press, New York, (1981).
- [53] P. Wolfe. Convergence conditions for ascent methods. *SIAM Rev.*, 11 :226-235,(1969).
- [54] P. Wolfe. Convergence conditions for ascent methods. II : Some corrections. *SIAM Rev.*, 13 :185-188,(1971).
- [55] J.Walmag. Optimisation contrainte et non-contrainte par régions de confiance et avec approximations locales quadratiques. Thèse de doctorat, université de Liège, (2010).
- [56] G. Zoutendijk. Nonlinear Programming, Computational Methods. In J. Abadie, editor, *Integer and Nonlinear Programming*, pages 37-86. North Holland, Amsterdam, (1970).