People's Democratic Republic of Algeria
Ministry of Higher Education and Scientific Research
Ferhat Abbas University
-Science Faculty -

THESIS

Presented in Mathematics Department

For Obtaining the Degree of 3rd cycle Doctorate in Mathematics

Option : Optimization and Control

Logarithmic barrier and inverse barrier interior point methods in nonlinear programming

---

By

FELLAHI Boutheina

In front of the jury :

BENSALEM Naceurdine	Pr	Univ. Ferhat Abbas Setif 1	President
MERIKHI Bachir	Pr	Univ. Ferhat Abbas Setif 1	Supervisor
MERZOUGUI Abdelkrim	Pr	Univ. Mohamed Boudiaf M'sila	Examiner
BOUAFIA Mousaab	MCA	Univ. 08 mai 1945, Guelma	Examiner

2022/2023

Trust the timing of your life

Dedicate

I dedicate this work:

To my dear mother Nassira Moudjed and my dear father Maamar, who have never stopped believing in me and encouraging me throughout these years. May they find here evidence of my heartfelt gratitude and appreciation.

To my brothers Hamza and Bilel, sisters Samiha and Zineb and everyone in my family who loves and encourages me.

To the beautiful thing that comforted me during this period my nephews Ali, Iyad, Malak, Noursine and Amine.

To everyone who has helped me - directly or indirectly - as well as those who have shared their emotions with me during the creation of this work and who have warmly supported and encouraged me throughout my journey.

To all of my friends who have consistently supported me, I extend my heartfelt gratitude and wish you continued success in all your endeavors.

Finally, I would like to thank my wonderful parents once again.

And for sure I dedicate this work to myself!

Boutheina Fellahi

Acknowledgment

This thesis would not have been possible without the generous support of many people.

First and foremost, I would like to acknowledge and thank my supervisor Professor Bachir MERIKHI for all the interesting discussions, helpful advice, inspiring conversations, and his guidance throughout this project.

I would like to thank Professors Mousaab BOUAAFIA and Abdelkrim MERZOUGUI for accepting to serve on the jury and for considerably boosting the quality of this work.

Many thanks to Professor Naceurdine BENSALEM, for serving as jury president.

I am deeply grateful for the support and guidance of my Professors and colleagues at the University of Constantine.

I would like to express my sincere gratitude to Professor Adnan Yassine at Le Havre University, France, for his warm reception and help during my visit to his esteemed institution.

I would also like to thank all of the members of the " Laboratory of Fundamental and Numerical Mathematics LMFN", Ferhat Abbas University, Setif 1, Setif, Algeria.

Boutheina Fellahi ✍

LIST OF ALGORITHMS

LIST OF TABLES

NOTATIONS AND SYMBOLS

The notations and symbols used throughout this thesis are listed below:

Notations

- MP: Mathematics programming.
- LP: Linear programming.
- NLP: Nonlinear programming.
- QP: quadratic programming.
- SDP: Semi Definite programming.
- IPMs: Interior point methods.
- lsc: Lower semicountinous.
- usc: Upper semicountinous.

Symbols

Vector spaces and sets

- $\mathbb{R}$: the set of real numbers.
- $\overline{\mathbb{R}} = [-\infty, +\infty]$.
- $\mathbb{R}_+$: the set of positive real numbers.
- $\mathbb{R}^n$: n-dimensional Euclidean vector space.
- $\mathbb{R}^n_+$: positive orthant of $\mathbb{R}^n$.

- $[\alpha, \beta]$: the closed interval of real numbers between α and β.
- (α, β): the open interval of real numbers between α and β.
- $Dom(f)$: the effective domain of a function $f : X \to \mathbb{R}$,

$$Dom(f) = \{x \in X;\ f(x) < \infty\}$$

.

- $x \geq 0$: means that all elements x_i: in the vector x are greater than or equal to zero.
- x^k: the k^{th} vector of a sequence of vectors.
- x_i: the i^{th} component of x.
- x^t: is the transpose of the vector x.
- $e_1, ...e_n$: the elements of the canonical basis of $\mathbb{R}^n$.
- e: the vector of $\mathbb{R}^n$ whose all components are equal to 1.
- $A \in \mathbb{R}^{m,n}$ a matrix with m rows and n columns,
- $int(X)$: the interior of a set X.

Functions

- m : is the number of constraints in the nonlinear programming problem, which is equivalent to the number of rows in the constraint matrix A.
- n: is the number of decision variables in the nonlinear programming problem, which is equivalent to the number of columns in the constraint matrix A.
- $\nabla f(x)$: the gradient of f at $x \in Dom(f)$, in which

$$(\nabla f(x))_i = \frac{\partial f(x)}{\partial x_i},\ \forall i = \overline{1, n}.$$

- $\nabla^2 f(x)$: the hessian of f at $x \in Dom(f)$, in which

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

- $\| x \|$: is the euclidean norm of the vector x.
- $\langle ., . \rangle$: usual scalar product of $\mathbb{R}^n$.

CONTENTS

GENERAL INTRODUCTION

Optimization is the process of finding the best possible solution for a given problem, a set of constraints and objectives. In other terms, it is the process of maximizing or minimizing a particular function, subject to certain conditions or constraints.
Optimization problems arise in various fields, including mathematics, engineering, economics, and computer science, among others. The goal of optimization is to find the optimal solution that satisfies the constraints and achieves the objectives.
There are different types of optimization techniques, including:

Linear programming.
Integer Linear Programming.
Nonlinear programming.
Quadratic programming.
Semi definite programming.

These techniques involve using mathematical algorithms and models to find the best solution. Optimization is essential in many practical applications, such as designing efficient systems, allocating resources, and minimizing costs.
In this work, we are interested to solve a nonlinear convex differentiable optimization programming problem. Unfortunately, general optimization problems are difficult to solve, so we considered using different interior point methods.
The first interior point methods and their polynomial complexity are appeared on 1955 by K. R. Frisch to solve a convex programming [19]. On 1967 P.Huard propose a method of centers for solving problems with nonlinear constraints [20]. On 1968, the interior point methods are developed by A. V. Fiacco and G. P. McCormick to solving convex nonlinear programming [18]. On 1970 N.Shor introduce a ellipsoid method for solving a linear programming [37], which developed on 1979 by L.G.Kachian and proved its polynomial complexity [23].
Lots of previous studies solve the linear, quadratic, semi definite and nonlinear programming using the interior point methods give much attention to the logarithmic barrier methods [10, 12] which proposed by Frish [18] and developed by Fiacco and McCormick [19]. In this method, the non negativity constrains $x_i \geq 0$ are replaced by a penalty term $-r\ln(x_i)$. If the barrier parameter r tends to zero, we prove the

convergence.
There are another known barrier methods as the inverse barrier methods which proposed by Carroll [8] where the inverse barrier function is given by $\sum_{i=1}^{n}\frac{1}{x_i}$, and developed by Fiacco and McCormick [19]. Kowalik [28] proposed the quadratic barrier function$\sum_{i=1}^{n}\frac{1}{x_i^2}$, D.En, C.Roos and T.Terklay [15] proposed a new type of inverse barrrier to solving linear programming using the barrier function $\frac{1}{r}\sum_{i=1}^{n}\frac{1}{x_i^r}$ in which $r \to 0$.

The penalization is a simple concept which makes it possible to transform an optimization problem with constraints into a problem or a sequence of problems without constraints. From a theoretical point of view, the penalization approach is sometimes used to study an optimization problem for which certain constraints are difficult to take into account, while the penalized problem has properties that are better understood or easier to demonstrate. If the penalty is well chosen, if we are lucky the desired properties of the original problem can be obtained directly properties of the penalized problem, in this case we speak of exact penalization, otherwise sometimes delicate passages to the limit make it possible to obtain properties of the original problem, we speak of inexact penalization.

From a numerical point of view, this transformation allows to use optimization algorithms, very efficient and better adapted to the structure of the penalized problem, to obtain the solution of problems whose admissible set can have a complex structure. However, it is not a "one size fits all" technique, because it has its own drawbacks: difficulty or impossibility to find an exact penalization, non-accuracy exact penalization, non-accuracy, need to minimize a sequence of functions,...
Before 1984, any linear optimization problem was solved by the simplex method developed by G.Dantzig [14].
The simplex method had not any seriously competition until 1984 when Karmarkar proposed a new polynomial time algorithm to solve a linear programming problem [21]. The polynomial complexity of Karmarkar algorithm made it better than the simplex method since of its exponential complexity. After the development that's Karmarkar made, the researchers improve the complexity of the algorithm [4] [27] [31] [23] [41] [33] and develop it to solve quadratic and nonlinear programming [6][39][16].
As we say, this thesis is devoted of the study of optimization problem and specially the resolution of the nonlinear programming which described bellow:

$$\begin{cases} \min\limits_{x\in\mathbb{R}^n} f(x) \\ Ax = b \\ x \geq 0 \end{cases} \tag{P}$$

Where the objective function f is nonlinear, convex, twice differentiable on

$$Y = \{x \in \mathbb{R}^n : \ Ax = b, \ x \geq 0\}$$

The stricture of this thesis

This thesis is divided into four chapters.

In the first chapter, we give some preliminary notions on convex analysis, optimization and mathematical programming and the results of existence and uniqueness of an optimal solution, and some methods of resolution of mathematical programming which will be used as support for the continuation, which we used it to demonstrate the theoretical results in the following chapters.

In the second chapter, we are interested to solve a nonlinear optimization problem using a barrier logarithmic penalty method, we thought to use a logarithmic barrier function $\sum_{i=1}^{n} r_i \ln r_i - \sum_{i=1}^{n} r_i \ln x_i$ where $r \in \mathbb{R}_+^n$ to perturb the (NLP). Then, we prove the existence and the uniqueness of the optimal solution of the perturbed problem, in addition to studying the convergence of the perturbed problem to the original problem. We calculate the Newton descent direction and we compute the step size using a technique of majorant function related to a secant technique which make it possible to calculate it in less time and with fewer number of iteration. We prove the effectiveness of our study by numerical simulations very encouraging.

In the third chapter, we thought to use the barrier function in [15] to solving a constrained convex nonlinear programming which had the same properties as the logarithmic barrier function and conserve all the properties of the objective function. The inverse barrier function using in this chapter is given by $r \sum_{i=1}^{n} \frac{1}{x_i^r}$ where $r > 0$ which ensure the convexity of the perturbed function. We define the perturbed problem of the original problem using the inverse barrier function which kept the properties of the first objective function, we studied the existence and the uniqueness of the optimal solution of the perturbed problem and we prove its convergence to the optimal solution of the original problem. Then, we are interested into the resolution of the perturbed problem. Firstly, we calculated the Newton descent direction, secondly, we calculate the step size along this direction using a tangent technique. In the last section, we present some numerical tests which prove the effectiveness of our approach when we compared it with the classical line search method.

In the fourth chapter, we are interested in the resolution of a nonlinear programming using the reduction of Karmarkar, we develop a new approach based on the technique of majorant function [12, 2] which facilitates the calculate of the step size instead of the potential function used by Karmarkar [40, 38, 25] . The main difficult in our study is how to linearize the objective function and ensure that the new function is a majorant linear approximation of the objective function, the linearization by a mojorant approximation help us to majorate the potential function and calculate the step size explicitly. The technique of the majorant function along a direction of descent is a very reliable alternative that will be confirmed as the technique of choice as well as for nonlinear programming and for other classes of optimization problems, and their practical performances seems quite promising for the field of continuous optimization.

At the end we give a general conclusion resume our work and outline potential future directions.

CHAPTER 1

PRELIMINARY NOTIONS

In this chapter, we will cover some fundamental concepts of convex analysis and general mathematical programming.
These concepts will be useful in demonstrating the theoretical results in the subsequent chapters. (See [1][26][34][11][5][36] [7]).

1.1 Convex analysis

The convexity is an important mathematics concept in the theoretical and numerical study of the optimization programming problems.

1.1.1 Affine sets and applications

Definition 1. *(Affine set).*
A set $E \subseteq \mathbb{R}^n$ is said an affine set if:

$$\forall x_1, x_2 \in E,\ \lambda x_1 + (1-\lambda)x_2 \in E,\ \forall \lambda \in \mathbb{R}.$$

This definition means that a set E is affine if the line which connects any two points x_1 and x_2 is contained in E.

Definition 2. *(Affine application).*
We say that an application f is affine if:

$$f : \mathbb{R}^n \to \mathbb{R}$$

$$\forall x_1, x_2 \in \mathbb{R}^n,\ \ \forall \lambda \in \mathbb{R}; f(\lambda x_1 + (1-\lambda)x_2) = \lambda f(x_1) + (1-\lambda) f(x_2)$$

1.1.2 Convex sets and applications

Definition 3. *(Convex set).*
A set $E \subseteq \mathbb{R}^n$ is said convex if:

$$\forall x_1, x_2 \in E,\ \lambda x_1 + (1-\lambda)x_2 \in E.\ \forall \lambda \in [0,1].$$

This definition means that a set E is convex if the segment which connects any two points x_1 and x_2 is contained in E.
ie:

$$[x,y] = \{\lambda x + (1-\lambda)y\ : 0 \leq \lambda \leq 1\} \subseteq E$$

.

Definition 4. *(Convex function).*
A real function f defined in a convex $E \subseteq \mathbb{R}^n$ is said convex if:

$$\forall x_1, x_2 \in E, \forall \lambda,\ 0 \leq \lambda \leq 1; f(\lambda x_1 + (1-\lambda)x_2) \leq \lambda f(x_1) + (1-\lambda) f(x_2)$$

Remark 1.
f is convex if and only if:

- *$\nabla^2 f(x)$ is a semi-definite positive matrix.*

$$y^t \nabla^2 f(x) y \geq 0,\ \forall x, y \in D. and y \neq 0$$

- *The mean values of $\nabla^2 f(x)$ are all positives.*

Definition 5. *(Strictly convex function).*
A real function f defined in a convex $E \subseteq \mathbb{R}^n$ is said strictly convex if:

$$\forall x_1, x_2 \in E : x_1 \neq x_2, \forall \lambda \in [0,1]; f(\lambda x_1 + (1-\lambda)x_2) < \lambda f(x_1) + (1-\lambda) f(x_2)$$

Remark 2.
f is strictly convex if and only if:

- *$\nabla^2 f(x)$ is a definite positive matrix ($y^t \nabla^2 f(x) y > 0,\ \forall x, y \in D$ and $y \neq 0$).*
- *The mean values of $\nabla^2 f(x)$ are all strictly positives.*

Definition 6. *(Strongly convex function).*
A real function f is said strongly convex function with coefficient $\delta > 0$ if and only if:

$$f(\lambda x_1 + (1-\lambda)x_2) \leq \lambda f(x_1) + (1-\lambda) f(x_2) - \frac{1}{2}\delta\lambda \parallel x_2 - x_1 \parallel^2$$

For all $x_1, x_2 \in E$ and $\lambda \in [0,1]$.

Definition 7. *(Convex combination).*
Either the set of elements $\{x_1, \ldots, x_m\}$ *of* $\mathbb{R}^n$, *we say that* x *is a convex combination of these points if there exists the reals* $\{\alpha_1, \ldots, \alpha_m\}$ *checking:*

- $\alpha_i \geq 0, \ \forall\, i = \overline{1, m}$.
- $\sum_{i=1}^{m} \alpha_i = 1$.
- $x = \sum_{i=1}^{m} \alpha_i x_i$.

Definition 8. *(Convex polyhedron)*
A subset D *is called a convex polyhedron if it is the intersection of a finite number of half-spaces of* $\mathbb{R}^n$.
i.e.

$$D = \bigcap_{i=1}^{m} \{x \in \mathbb{R}^n; \langle a_i, x\rangle \leq b_i, \quad a_i \in \mathbb{R}^n, b_i \in \mathbb{R}\}.$$

Definition 9. *(Convex envelope).*
Either the set $K \subseteq \mathbb{R}^n$, *we call a convex envelope of* K, *and we denote by* $conv(K)$, *the smallest convex set containing* K.

In finite dimension, it is still the set of convex combinations of the elements of K*:*

$$conv(K) = \{x \ \in \ \mathbb{R}^n \ : \ x = \sum_{i=1}^{m} \alpha_i x_i \ , \ x_i \in K \ , \ \sum_{i=1}^{m} \alpha_i = 1, \ \alpha_i \geq 0, \ m \in \mathbb{N}\}.$$

Definition 10. *(Proper function).*
A real function f *is defined on* $E \subseteq \mathbb{R}^n$ *is said a proper function if:*

- $f(x) > -\infty, \ \forall x \in E$.
- $dom(f) \neq \Phi$.

Definition 11.
Let X *be a convex and closed part of* $\mathbb{R}^n$.
A point x *of* X *is said an extreme point if the equality*

$$x = ty + (1-t)z; \quad y, z \in X, \ t \in]0, 1[$$

has no solution other than $x = y = z$.
Geometrically this means that there is no line segment not reduced to a point, contained in A *and containing the point* x *inside it (i.e. in such a way that* x *is not one of the extremities of this segment).*

1.1.3 Implicit function theorem

Theorem 1. *(Version $\mathbb{R}^2$).*
Either D is an open of $\mathbb{R}^2$, and $F : D \to \mathbb{R}$ an application of class C^k, with $k \geq 1$. Either $(x_0, y_0) \in \mathbb{R}^n$ such that:

$$F(a, b) = 0 \text{ et } \frac{\partial F}{\partial y}(x_0, y_0) \neq 0.$$

Then, there exists a neighborhoods U and V of x_0 and y_0 in $\mathbb{R}$, and an application $\varphi : U \to V$ of class C^k such that:

$$\forall x \in U,\ y \in V : F(x, y) = 0 \Leftrightarrow y = \varphi(x).$$

We can also choose U and V in which the partial derivative $\partial_y F \neq 0$ in $U \times V$, and we have:

$$\forall x \in U,\ \varphi^{'}(x) = -\frac{\frac{\partial F}{\partial x}(x, \varphi(x))}{\frac{\partial F}{\partial y}(x, \varphi(x))}$$

1.1.4 Lower and upper semicountinous functions

Definition 12. *Lower semicountinous function (lsc)*
Either $f : \mathbb{R}^n \to \overline{\mathbb{R}}$ is a lower semicountinous function on $x^0 \in \mathbb{R}^n$ if:

$$\forall \epsilon > 0, \exists \alpha > 0 : f(x) \geq f(x^0) - \epsilon; \parallel x - x^0 \parallel \leq \alpha.$$

Definition 13. *Upper semicountinous function (usc)*
f is a upper semicountinous on x^0 if $-f$ is lower semicountinous on x^0.

Remark 3.
f is countinuous on x^0 if f is both lsc and usc on x^0.

1.2 Mathematical programming

A general constrained mathematical programming problem can be stated as an optimization problem with constraints which define as follow:

$$\begin{cases} \min f(x) \\ x \in D \end{cases} \tag{MP}$$

In which

$$D = \begin{cases} x \in \mathbb{R}^n, \\ g_j(x) \leq 0 \quad j = \overline{1, m}, \\ h_i(x) = 0 \quad i = \overline{1, p}. \end{cases}$$

D is called the set of feasible solutions of (MP)
$f, h_i, g_j : \mathbb{R}^n \longrightarrow \mathbb{R}$ are a given functions.
f is called the objective function.

Definition 14.

- *We call a feasible solution of (MP), any point satisfying the constraints, ie: any point belonging to D.*
- *We call a global optimal solution x^* of (MP) the feasible solution which minimizes the objective function on D.*

  We write:
$$x^* = \underset{x \in D}{argmin} f(x)$$
- *We call a local optimal solution $x^* \in D$ of (MP) if there is a neighborhood V of x^* such that:*
$$f(x^*) \leq f(x) \quad \forall x \in D$$

1.2.1 Classification of a mathematical programming

Every mathematical programming is classified according of two fundamental properties: the convexity and the diferentiability of the objective function and the constraints.

- (MP) is called a differentiable problem if f, g_j, h_i are all differentiable.
- (MP) is called a convex problem if f is convex and the set of constraints D is convex.

We can also classify a mathematical programming according to the nature of the objective function and the constrains.

- (MP) is called a linear programming if f is a linear function and g_j, h_i are affine.
- (MP) is called a quadratic (nonlinear) programming if f is a quadratic (nonlinear) function.

1.2.2 Existence and uniqueness of the optimal solution of (MP)

Theorem 2. *Existence (Weistrass)*
If f is a continue function on the set D which is compact (closed and bounded) then (MP) have at least one optimal solution $x^* \in D$.

Corollary 1.
If f is a continue and coercive function on the set D ($\lim_{\|x\|\to+\infty} f(x) = +\infty$) which is closed and non-empty then (MP) have at least one optimal solution.

Theorem 3. *Uniqueness*
If D is a convex non-empty set, and f is a strictly convex set then (MP) admits one optimal solution $x^* \in D$.

Remark 4.
The strict convexity of f does not ensure the existence of the optimal solution but only the uniqueness.

1.2.3 Qualification of constraints

Definition 15.
An inequality constraint $g_j(x) \leq 0$ is called an active constraint on $x^* \in D$ if:

$$g_j(x^*) = 0 \quad \forall j = \overline{1,m}$$

We define the set $I(x^*)$ by:

$$I(x^*) = \{j;\ g_j(x^*) = 0\}.$$

An equality constraint $h_i(x) = 0$ is active for all feasible point $x \in D$.

The constraints are qualified in all feasible point $x \in D$ if:

- Slater 1950: D is convex and $int(D) \neq \phi$.
- Karlin 1959: D is a polyeder convex (h_i, g_j are affines).
- Mangasarian-Fromovitz 1967: If the gradients of all active constraints on $\overline{x} \in D$ are linearly independent, then the constraints are qualified on $\overline{x}$.

1.2.4 Optimality conditions

In the rest of this chapter, we assume that the functions f, g_j and h_i are at least twice continuously differentiable.

We consider the problem (MP) in the following form :

$$\begin{cases} \min f(x) \\ h_i(x) = 0, \quad i = \overline{1,p} \\ g_j(x) \leq 0, \quad j = \overline{1,m} \\ x \in \mathbb{R}^n \end{cases} \tag{MP}$$

Definition 16.
The Lagrangian associated to the problem (MP) is given by:

$$L(x, \lambda, \mu) = f(x) + \sum_{i=1}^{p} \lambda_i h_i + \sum_{j=1}^{m} \mu_j g_j$$

λ_i, μ_j *are called the Lagrange multipliers, in which* $\lambda_i \in \mathbb{R}$ *for all* $i = \overline{1,p}$ *and* $\mu_j \in \mathbb{R}^+$ *for all* $j = \overline{1,m}$.

Theorem 4. *Karush-Kuhn-Tucker*
Either x^* is an optimal local solution of (MP) satisfying the condition of qualifiaction of constraints, then there exists two lagrange multipliers $\lambda^* \in R^p$ and $\mu^* \in R^m_+$ such that:

$$\begin{cases} \nabla_x L(x^*; \lambda^*, \mu^*) = 0, \text{ (Optimality conditions)} \\ \mu_j^* g_j(x^*) = 0, \quad \text{(Complementary condition)} \\ \mu_j^* \geq 0, \ j = \overline{1,m} \\ g_j(x^*) \leq 0, \ j = \overline{1,m} \\ h_i(x^*) = 0, \ i = \overline{1,p} \end{cases}$$

Are define the first necessary conditions of the first order.

Remark 5.

1. *If the constrains are not qualified on x^*, then the conditions of KKT do not apply.*
2. *If (MP) is convex, then the KKT conditions are necessary and sufficient for x^* be a global minimum.*
3. *If (MP) is convex, then all local minimum is a global minimum.*

1.2.5 Method of solving a mathematical programming

We can classify the methods of solving a mathematical programming into three main categories:

1) Gradient methods

a) Conjugate gradient methods

The conjugate gradient methods were proposed by Hestenes on 1952 to solve a linear programming with a positive definite matrix. In 1964 Fletcher and Reeves generalized this method to solve a nonlinear programming problems.

The conjugate gradient methods are known by there efficiency to solve a quadratic problems without constraints.

In the constrained problems, there is a simple change of variable help us to return to the unconstrained problem.

Indeed: $\overline{x}$ verify $A\overline{x} = 0$.

We suppose: $x = \overline{x} + P_A z$ such that $P_A = I - A^t(AA)^{-1}A$ is the operator of the projection on the kernel of the matrix A.

The main purpose of this method is to progressively build up directions $d_0, ..., d_n$ mutually conjugate with the hessian matrix of the objective function f notice with $\nabla^2 f(x)$

$$d_j \nabla^2 f(x) d_k = 0, \quad \forall j, k = \overline{1, n}$$

b) Projected gradient

Projected gradient methods solve general programs of the form :

$$\begin{cases} \min f(x) \\ Ax = b \\ x \geq 0 \end{cases}$$

In which f is a differentiable, not necessarily convex function.

The main purpose of this method is to project the gradient of the objective function onto the boundary of the domain, which gives a path along the boundary in the direction of the strongest relative slope. This method is essentially interesting in the case of linear constraints, because in this case, the projection is easy to calculate.

2) Interior point methods

We can classify the interior point methods into three main categories which described bellow, in all of these methods we replace the constraint of non negativity $x \geq 0$ by an ellipsoid, or a barrier function.

a) Reduction potential methods

In operations research, linear programming models are one of the most commonly utilized models to handle a wide range of real-world problems. The application of linear programming in agriculture, telecommunications, transportation challenges, profit maximization and cost reduction, and so on are widely known in the literature. It is still one of several operational research approaches employed by military forces across the world. Large linear programming problems were unsolvable before to the development of Karmarkar's [21] efficient interior-point approach for solving linear problems. A linear programming problem with millions of variables and equations may now be solved utilizing multiple interior point methods established following Karmarkar's fundamental work [21]. Because linear programming models are utilized in

many different sectors of the economy and industry, finding efficient methods to tackle linear programming problems is essential.
The Karmarkar algorithm described in [21] for solving a linear problem presents the important role of the potential function in the interior point methods.
Karmarkar proves the convergence and the polynomiality of his algorithm by showing that this function is reduced at each iteration by at least one constant. His algorithm is the first interior point algorithm which competes the simplex method at that time.

b) Trajectory centry methods

They were delivered at the same time as the potential reduction methods and developed in the early 90s. They have good properties theoretical: polynomial complexity and superlinear convergence. The algorithms of (TC) restrict the iterates to a neighborhood of the central path, the latter is an arc of perfectly feasible points.

c) Barrier methods

Barrier methods are designed to solve (MP) by instead solving a sequence of specially constructed unconstrained optimization problems. The idea in the barrier penalty method is to choose a penalty function $B(x)$ which define bellow and a constant r so that the optimal solution $x(r)$ of the perturbed problem is also an optimal solution of the original problem (MP).

The presentation of penalty methods has assumed either that the problem has no equality constraints, or that the equality constraints have been converted to inequality constraints. For the latter, the conversion is easy to do, but the conversion usually violates good judgement in that it unnecessarily complicates the problem. Furthermore, it can cause the linear independence condition to be automatically violated for every feasible solution. In a barrier method, we presume that we are given a point x_0 that lies in the interior of the feasible set of (MP)

Definition 17. *Barrier function.*

We say a barier function associated to $X \subseteq \mathbb{R}^n$, all function B define on $int(X)$ such that:

- *B is continue.*
- $B(x) \geq 0, \ \forall x \in int(X)$.
- $\lim B(x) = +\infty; x \to x^* \in Fr(X)$.

The most used barrier functions are the logarithmic and inverse functions defined respectively by :

$$B_l(x) = -\sum_{i=1}^{n} \ln(-g_i(x)),$$

$$B_r(x) = -\sum_{i=1}^{n} (g_i(x))^{-1}$$

CHAPTER 2

A LOGARITHMIC BARRIER APPROACH VIA MAJORANT FUNCTION FOR NONLINEAR PROGRAMMING

2.1 Introduction

In this chapter, we are interested in the theoretical and numerical study of a convex nonlinear optimization problem with linear and affine constraints, our objective is to develop a new approach of logarithmic barrier interior point methods. In the first section, we define the problem under consideration and perturb it using the logarithmic barrier function which has the following form $\sum_{i=1}^{n} r_i \ln r_i - \sum_{i=1}^{n} r_i \ln x_i$ where the penalty term is represented by a vector $r \in \mathbb{R}^n_{++}$. We then begin our work by investigating the existence and uniqueness of the optimal solution to the perturbed problem followed by a convergence study. In the second section, we are investigated in the resolution of the perturbed problem which is based on the classical Newton method to compute the descent direction and a majorant function approach connected to a secant technique to calculate the step size.

Finally, we conclude with a summary of the algorithm and some numerical tests show the effectiveness of this strategy when compared to the classical line search method.

2.1.1 The problem formulation

The following is the problem to be explored in this chapter:

$$\begin{cases} \min\limits_{x\in\mathbb{R}^n} f(x) \\ Ax = b \\ x \geq 0 \end{cases} \tag{P}$$

In which:

$$Y = \{x \in \mathbb{R}^n : \ Ax = b, \ x \geq 0\},$$

$$Y^0 = \{x \in \mathbb{R}^n : \ Ax = b, \ x > 0\},$$

are the sets of feasible and strictly feasible solutions of (P) respectively.

Assumptions:

A1 f is nonlinear, convex, twice continuously differentiable function on Y.

A2 $A \in \mathbb{R}^{m \times n}$ is a full rank matrix, $b \in \mathbb{R}^m$, $(m < n)$.

A3 There exists $x^0 > 0$ such that $Ax^0 = b$.

A4 The set of optimal solutions of $(P1)$ is nonempty and bounded.

For x^* be an optimal solution in the problem (P), there exists two Lagrange multipliers $u^* \in \mathbb{R}^m, v^* \in \mathbb{R}^n_+$ such as:

$$\begin{cases} \nabla f(x^*) + A^t u^* - v^* = 0, \\ Ax^* = b, \\ < v^*, x^* >= 0. \end{cases} \tag{2.1}$$

We are able to write

$$f^* = f(x^*) = \min_{x \in Y^0} f(x).$$

In the following, we replace the nonlinear problem (P) with a perturbed problem using the barrier function $\sum_{i=1}^n r_i \ln r_i - \sum_{i=1}^n r_i \ln x_i$. What is new in our work is that the term of penalty r is taken as a vector $r = (r_1, r_2, ..., r_n)^t \in \mathbb{R}^n$ with r_i are all strictly positive for $i = \overline{1, n}$.

2.1.2 The perturbed problem

In this part, we define the function $\psi : \mathbb{R}^n_{++} \times \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ which is convex, lower semi continuous and proper function.

$$\psi(r, x) = \begin{cases} \sum_{i=1}^n r_i \ln(r_i) - \sum_{i=1}^n r_i \ln(x_i), & \text{if } x, r > 0, \\ 0, & \text{if } \ r = 0, \ x \geq 0, \\ +\infty, & \text{if not.} \end{cases} \tag{2.2}$$

Then consider the function ϕ defined on $\mathbb{R}^n_{++} \times \mathbb{R}^n$ by:

$$\phi_r(x) = \phi(r,x) = \begin{cases} f(x) + \sum_{i=1}^n r_i \ln(r_i) - \sum_{i=1}^n r_i \ln(x_i) & \text{if} \quad Ax = b;\ x, r \geq 0, \\ +\infty \quad \text{if not.} \end{cases} \tag{2.3}$$

Finally, the convex function m is defined as follows:

$$m(r) = \inf_x \{\phi_r(x); \quad x \in \mathbb{R}^n\} \qquad (P_r)$$

Remark 6.
Because of the convexity of ϕ_r, it's clearly that m is convex.

We notice that the two problems (P) and (P_r) are coincided when $\| r \| \to 0$, which means that the optimal solution of the problem (P_r) is an approximated solution of the problem (P), then $\phi_r^* = m(0)$.

Existence and uniqueness of the optimal solution of perturbed problem

Definition 18.
The function $f : \mathbb{R}^m \to \mathbb{R} \cup \{\infty\}$ is called inf-compact if the set

$$C_d(f) = \{d \in \mathbb{R}^n : \ [f]_\infty(d) \leq 0,\ Ad = 0,\ d \geq 0\}$$

is compact which means that its cone of recession is reduced to zero.
$[f]_\infty(d)$ is the asymptotic function of f, defined as follows:

$$[f]_\infty(d) = \lim_{t \to +\infty} \frac{f(x+td) - f(x)}{t}$$

Lemma 1.
The problem (P_r) has a unique optimal solution $x(r)$ if the objective function ϕ_r is inf-compact.

Proof.
To demonstrate that (P_r) admits one unique optimal solution, suffice it to prove that the cone of recession of ϕ_r is reduced to zero.

According to the fourth assumption, (P) admits one unique optimal solution then the cone of recession $C_d(f)$ of f is reduced to zero:

$$C_d(f) = \{0\}$$

We have:

$$\begin{aligned} [\phi_r]_\infty(d) &= \lim_{t \to +\infty} \tfrac{\phi_r(x+td) - \phi_r(x)}{t} \\ &= \lim_{t \to +\infty} \tfrac{f(x+td) - f(x)}{t} - \lim_{t \to +\infty} \tfrac{\sum_{i=1}^n r_i(\ln(x_i+td_i) - \ln(x_i))}{t} \end{aligned}$$

Which give us:

$$[\phi_r]_\infty = \begin{cases} [f]_\infty(d) & \text{if} \quad Ad = 0, \ d \geq 0, \\ \\ +\infty & \text{if not.} \end{cases}$$

Then we can conclude:

$$\{d \in \mathbb{R}^n; \quad [\phi_r]_\infty \leq 0\} = \{0\}$$

Which means that $C_d(\phi) = \{0\}$.

By taking into account that ϕ_r is strictly convex, we come to conclusion that the perturbed problem (P_r) admits one unique optimal solution which is denoted by $x(r) \in Y^0$. □

Convergence of perturbed problem

Lemma 2.
We assume that $x(r)$ is the optimal solution of (P_r) and x^* is the optimal solution of (P).
We have

$$x(r) \to x^* \ \ when \ \ \| r \| \to 0.$$

Proof.

According to the necessary and sufficient optimality conditions, there exists $\lambda(r) \in \mathbb{R}^m$ verify:

$$\begin{cases} \nabla f(x(r)) - X_r^{-1} r + A^t \lambda(r) = 0, \\ \\ Ax(r) - b = 0. \end{cases} \tag{2.4}$$

In which X is the diagonal matrix with diagonal entries $X_{ii} = x_i, \forall i = \overline{1, n}$.

We impose that the couple $(x(r), \lambda(r))$ is a solution of the equation $F\left(x(r), \lambda(r)\right) = 0$ in which

$$F\left(x(r), \lambda(r)\right) = \begin{pmatrix} \nabla f(x(r)) - X_r^{-1} r + A^t \lambda(r) \\ \\ Ax(r) - b \end{pmatrix}$$

The two functions $r \mapsto x(r)$ and $r \mapsto \lambda(r)$ are differentiable on $\mathbb{R}^n_+$, by using the implicit function theorem, we get

$$\begin{pmatrix} \nabla^2 f(x(r)) + R X_r^{-2} & A^t \\ A & 0 \end{pmatrix} \begin{pmatrix} \nabla x(r) \\ \nabla \lambda(r) \end{pmatrix} = \begin{pmatrix} X_r^{-1} \\ 0 \end{pmatrix} \tag{2.5}$$

Where R is the diagonal matrix with diagonal entries $R_{ii} = r_i, \forall i = \overline{1, n}$.
And

$$\nabla x(r) = \begin{pmatrix} \frac{\partial x_1}{\partial r_1} & \frac{\partial x_1}{\partial r_2} & \cdots & \frac{\partial x_1}{\partial r_n} \\ \frac{\partial x_2}{\partial r_1} & \frac{\partial x_2}{\partial r_2} & \cdots & \frac{\partial x_2}{\partial r_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial x_n}{\partial r_1} & \frac{\partial x_n}{\partial r_2} & \cdots & \frac{\partial x_n}{\partial r_n} \end{pmatrix}, \quad \nabla \lambda(r) = \begin{pmatrix} \frac{\partial \lambda_1}{\partial r_1} & \frac{\partial \lambda_1}{\partial r_2} & \cdots & \frac{\partial \lambda_1}{\partial r_n} \\ \frac{\partial \lambda_2}{\partial r_1} & \frac{\partial \lambda_2}{\partial r_2} & \cdots & \frac{\partial \lambda_2}{\partial r_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial \lambda_m}{\partial r_1} & \frac{\partial \lambda_m}{\partial r_2} & \cdots & \frac{\partial \lambda_m}{\partial r_n} \end{pmatrix}.$$

Remember that the function m which is differentiable on $\mathbb{R}^n_+$ is define by:

$$m(r) = f(x(r)) + \sum_{i=1}^{n} r_i \ln(r_i) - \sum_{i=1}^{n} r_i \ln(x_i(r))$$

We have

$$\nabla m(r) = (\nabla x(r))^t \left(\nabla f(x(r)) - X_r^{-1} r\right) + (e + z_1 - z_2)$$

In which

$$e = (1, 1, ..., 1)^t,$$

$$z_1 = (\ln r_1, \ln r_2, ..., \ln r_n)^t,$$

$$z_2 = (\ln x_1, \ln x_2, ..., \ln x_n)^t.$$

According to (2.4) and (2.5), we get

$$\begin{aligned} \nabla m(r) &= -(\nabla x(r))^t A^t \lambda(r) + (e + z_1 - z_2) \\ &= -(A \nabla x(r))^t \lambda(r) + (e + z_1 - z_2) \\ &= e + z_1 - z_2. \end{aligned}$$

For $x(r) \in Y$ and since of the convexity of m, we get:

$$\begin{aligned} m(0) &\geq m(r) - r^t \nabla m(r) \\ &\geq f(x(r)) + \textstyle\sum_{i=1}^n r_i \ln r_i - \sum_{i=1}^n r_i \ln x_i(r) - r^t(e + z_1 - z_2) \end{aligned}$$

And therefore we have

$$\begin{aligned} m(0) &\geq f(x(r)) + \textstyle\sum_{i=1}^n r_i \ln r_i - \sum_{i=1}^n r_i \ln x_i(r) - \sum_{i=1}^n r_i - \sum_{i=1}^n r_i \ln r_i + \sum_{i=1}^n r_i \ln x_i(r) \\ &= f(x(r)) - \textstyle\sum_{i=1}^n r_i \end{aligned}$$

Taking into account that:

$$f^* = m(0) = \min_x f(x^*)$$

Then, we come conclusion

$$f^* \leq f(x(r)) \leq f^* + \sum_{i=1}^{n} r_i.$$

for the rest, we are interesting on the trajectory of $x(r)$ when $\| r \|$ tends to zero.

a) The case in which f is only convex,

This case is a little complicated, we impose that $\|r\|_\infty \leq 1$, and for that we note

$$x(r) \in \{x;\ Ax = b,\ x > 0,\ f(x) \leq n + f^*\}$$

This set is convex, bounded and non empty, its cone of recession is reduced to zero.
It follows that each accumulation point of x_r is an optimal solution of (P) only if $\|r\| \to 0$.

b) The case in which f is strongly convex with coefficient γ strictly positive,

We have

$$\sum_{i=1}^{n} r_i \geq f(x(r)) - f(x^*) \geq \langle \nabla f(x^*), x(r) - x^* \rangle + \frac{\gamma}{2} \| x(r) - x^* \|^2$$

Using (2.1), we obtain

$$\sum_{i=1}^{n} r_i \geq \langle v^*, x(r) \rangle + \frac{\gamma}{2} \| x(r) - x^* \|^2$$

Then

$$\| x(r) - x^* \| \leq \left(\frac{2}{\gamma} \sum_{i=1}^{n} r_i \right)^{\frac{1}{2}}$$

We come to conclusion that the convergence of $x(r)$ to x^* is of order $\frac{1}{2}$.

□

Remark 7.
If the problem (P) or the perturbed problem (P_r) will have an optimal solution and the values of their objective functions are equal and finite, the other problem has an optimal solution.

The general principle of the method

The general prototype of our method is as follows:

i Starting by $(r_0, x_0) \in \mathbb{R}^n_+ \times Y^0$.

ii Find an approximate solution of (P_r) has been noted by x_{k+1} such that:

$$\phi(r_k, x_{k+1}) \leq \phi(r_k, x_k).$$

iii Take: $\|r_{k+1}\|_\infty \leq \|r_k\|_\infty$.

The iterations continue until we obtained the good approximate solution.

2.2 Solving the perturbed problem

Consider the following perturbed problem defined as follows:

$$\begin{cases} \min_x \phi_r(x) \\ x \in \mathbb{R}^n \end{cases}$$

2.2.1 Some useful inequalities

Taking into consideration the statistical serie of n real numbers $\{z_1, ..., z_n\}$, we define their arithmetic mean $\overline{z}$ and their standard deviation σ_z.

These quantities are defined as follows:

$$\overline{z} = \frac{1}{n}\sum_{i=1}^{n} z_i, \quad \sigma_z^2 = \frac{1}{n}\sum_{i=1}^{n} z_i^2 - \overline{z}^2 = \frac{1}{n}\sum_{i=1}^{n}(z_i - \overline{z})^2$$

For the following result see [13][40]

Proposition 1.

$$\overline{z} - \sigma_z\sqrt{n-1} \leq \min_{i=\overline{1,n}} z_i \leq \overline{z} - \frac{\sigma_z}{\sqrt{n-1}}$$

$$\overline{z} + \frac{\sigma_z}{\sqrt{n-1}} \leq \max_{i=\overline{1,n}} z_i \leq \overline{z} + \sigma_z\sqrt{n-1}$$

In the case where z_i are all positifs, we have:

$$\ln\left(\overline{z} - \sigma_z\sqrt{n-1}\right) \leq \sum_{i=1}^{n}\ln(z_i) \leq \ln\left(\overline{z} + \sigma_z\sqrt{n-1}\right)$$

Theorem 5. *[12][33]*

Assume that $z_i > 0$ for all $i = \overline{1,n}$, then:

$$A_1 \leq \sum_{i=1}^{n}\ln(z_i) \leq A_2,$$

with:

$$A_1 = (n-1)\ln\left(\overline{z} + \tfrac{\sigma_z}{\sqrt{n-1}}\right) + \ln\left(\overline{z} - \sigma_z\sqrt{n-1}\right),$$

$$A_2 = \ln\left(\overline{z} + \sigma_z\sqrt{n-1}\right) + (n-1)\ln\left(\overline{z} - \tfrac{\sigma_z}{\sqrt{n-1}}\right).$$

In this part, we are interested in the numerical solution of the problem $(P1)$, we begin our work by calculating the descent direction and step size using a new majorant function approach.

2.2.2 The descent direction and line search function

Generally, a descent direction d may be computed by various methods, in our study we use the Newton's method and so d is given by solving the following quadratic convex minimization problem:

$$\begin{cases} \min\limits_{d} \left(\frac{1}{2}\langle \nabla^2\phi_r(x)d, d\rangle + \langle \nabla\phi_r(x), d\rangle\right) \\ \\ Ad = 0. \end{cases}$$

According to the necessary and sufficient optimality conditions, there exists $\mu \in \mathbb{R}^m$ such that:

$$\begin{cases} \nabla^2\phi_r(x)d + \nabla\phi_r(x) + A^t\mu = 0 \\ \\ Ad = 0. \end{cases}$$

Which is equivalent to

$$\begin{pmatrix} \nabla^2 f(x) + RX^{-2} & A^t \\ \\ A & 0 \end{pmatrix} \begin{pmatrix} d \\ \\ \mu \end{pmatrix} = \begin{pmatrix} X^{-1}r - \nabla f(x) \\ \\ 0 \end{pmatrix}$$

From which we get

$$\begin{pmatrix} d^t & 0 \end{pmatrix} \begin{pmatrix} \nabla^2 f(x) + RX^{-2} & A^t \\ \\ A & 0 \end{pmatrix} \begin{pmatrix} d \\ \\ \mu \end{pmatrix} = \begin{pmatrix} d^t & 0 \end{pmatrix} \begin{pmatrix} X^{-1}r - \nabla f(x) \\ \\ 0 \end{pmatrix}$$

Then

$$\langle \nabla^2 f(x)d, d\rangle + \langle \nabla f(x), d\rangle = \langle r, X^{-1}d\rangle - \langle RX^{-1}d, X^{-1}d\rangle \tag{2.6}$$

This system is equivalent to

$$\Gamma Z = B \tag{2.7}$$

In which

$$\Gamma = \begin{pmatrix} X\nabla^2 f(x)X + R & XA^t \\ AX & 0 \end{pmatrix}, \quad Z = \begin{pmatrix} X^{-1}d \\ \mu \end{pmatrix}, \quad B = \begin{pmatrix} r - X\nabla f(x) \\ 0 \end{pmatrix}$$

The linear system (2.7) can be efficiently solved via the Cholesky decomposition. The direction d being computed, we look for t giving a significant decreasing of the search line function. Then, the next iteration will be taken equal to $x + td$.

2.2.3 Computation of the step size

Generally, the most used methods in the search line are the classical iterative methods as Armijo-Goldstein, Wolfe, Fibonnaci,..., but the computational cost in there becomes high when n is very large.

In this part, we are interested to avoid this difficulty. The method that we use bellow is simple and more effective than the first, it consists on the use of majorant function of the line search function. The choice of the step size $t^* > 0$ must give us a significant decrease of the last function, we have:

$$\begin{aligned} \theta_0(t) &= \phi_r(x+td) - \phi_r(x) \\ &= f(x+td) - f(x) - \textstyle\sum_{i=1}^n r_i ln(1+ty_i), \qquad y = X^{-1}d \end{aligned}$$

According to proposition 2.1, we have: $\rho \leq \min_i r_i \leq r_i \quad \forall i = \overline{1,n}$

In which $\rho = \overline{r} - \sigma_r\sqrt{n-1}$

Then, we obtain

$$\theta(t) = \frac{\theta_0(t)}{\rho} \leq \theta_1(t) = \frac{1}{\rho}\left(f(x+td) - f(x)\right) - \sum_{i=1}^n ln(1+ty_i)$$

We have

$$\theta'(t) = \tfrac{1}{\rho}\left(\langle \nabla f(x+td), d\rangle - \textstyle\sum_{i=1}^n r_i \frac{y_i}{1+ty_i}\right),$$

$$\theta''(t) = \tfrac{1}{\rho}\left(\langle \nabla^2 f(x+td)d, d\rangle + \textstyle\sum_{i=1}^n r_i \frac{y_i^2}{(1+ty_i)^2}\right).$$

And

$$\theta_1'(t) = \tfrac{1}{\rho}\langle \nabla f(x+td), d\rangle - \textstyle\sum_{i=1}^n \frac{y_i}{1+ty_i},$$

$$\theta_1''(t) = \tfrac{1}{\rho}\langle \nabla^2 f(x+td)d, d\rangle + \textstyle\sum_{i=1}^n \frac{y_i^2}{(1+ty_i)^2}.$$

We deduce from (4.11), that

$$\theta'(0) + \theta''(0) = 0$$

We have

$$\theta^{''}(0) \geq 0 \text{ and } \theta^{''}(0) \geq 0$$

Which give us that $\theta^{'}(0) \leq 0$.
Now it must to prove that $\theta_1^{'}(0) \leq 0$, we have:

a If $y_i \geq 0$, it is clearly that $\theta_1^{'}(0) \leq 0$.

b If $y_i < 0$,

We deduce from (4.11), that

$$\theta_1^{'}(0) + \theta_1^{''}(0) \leq 0$$

We have

$$\theta_1^{''}(0) \geq 0$$

We come conclusion that $\theta_1^{'}(0) \leq 0$

What is prove the significant decrease of θ_1.

2.2.4 The first majorant function

The choice of t^* in which $\theta^{'}(t^*) = \theta^{'}(t_{opt}) = 0$ consists of some numerical complications, so generally we can't obtain t^* directly. To solve this problem, we propose to find an approximation function of θ.

This method is based on the use of a majorant function θ_2 of the function θ.

Lemma 3.

The first majorant function of θ is given by:

$$\theta_2(t) = \frac{1}{\rho}(f(x+td) - f(x)) - (n-1)ln(1+t\alpha) - ln(1+t\beta)$$

In which

$$\begin{cases} \alpha = \overline{y} + \frac{\sigma_y}{\sqrt{n-1}} \\ \beta = \overline{y} - \sigma_y\sqrt{n-1} \end{cases}$$

Proof.

In the following, we take:

$$x_i = 1 + ty_i, \ \overline{x} = 1 + t\overline{y} \ \text{ and } \ \sigma_x = t\sigma_y$$

Applying the inequality $\sum_{i=1}^{n} ln(x_i) \geq A_1$ (theorem 2.2), we get that

$$\theta_1(t) \leq \theta_2(t)$$

Such that

$$\theta_2(t) = \frac{1}{\rho}\left(f(x+td) - f(x)\right) - (n-1)ln(1+t\alpha) - ln(1+t\beta)$$

Where

$$\alpha = \overline{y} + \tfrac{\sigma_y}{\sqrt{n-1}},$$

$$\beta = \overline{y} - \sigma_y\sqrt{n-1}$$

We have

$$\theta_2'(t) = \tfrac{1}{\rho}\langle \nabla f(x+td), d\rangle - (n-1)\tfrac{\alpha}{1+t\alpha} - \tfrac{\beta}{1+t\beta},$$

$$\theta_2''(t) = \tfrac{1}{\rho}\langle \nabla^2 f(x+td)d, d\rangle + (n-1)\tfrac{\alpha^2}{(1+t\alpha)^2} + \tfrac{\beta^2}{(1+t\beta)^2}.$$

The domains of θ_2 is $H_2 =]0, T[$ in which $T = max\{t :\ 1 + t\beta > 0\}$.
This domain is content in the domain of the line search function θ.
We notice that :

$$\theta(0) = \theta_1(0) = \theta_2(0) = 0,$$

$$\theta_1''(0) = \theta_2''(0) > 0,$$

$$\theta_1'(0) = \theta_2'(0) < 0.$$

We prove that the strictly convex function θ_2 is a good approximation of θ_1 in a neighborhood of 0, hence the unique minimum t^* of θ_2 guarantee a significant decrease of the function θ_1, and we have the follows inequalities:

$$\theta(t) \leq \theta_1(t) \leq \theta_2(t) < 0$$

□

Remark 8.
θ_2 is a strictly convex function.

Proposition 2.
The auxiliary function ω is given by

$$\omega(t) = n\eta t - (n-1)ln(1+t\alpha) - ln(1+t\beta)$$

In which

$$\begin{cases} \alpha = \overline{y} + \frac{\sigma_y}{\sqrt{n-1}} \\ \\ \beta = \overline{y} - \sigma_y\sqrt{n-1} \end{cases}$$

We have

$$\omega^{'}(t) = n\eta - (n-1)\tfrac{\alpha}{1+t\alpha} - \tfrac{\beta}{1+t\beta},$$

$$\omega^{''}(t) = (n-1)\tfrac{\alpha^2}{(1+t\alpha)^2} + \tfrac{\beta^2}{(1+t\beta)^2}.$$

Then:

$$\begin{cases} \omega(0) = 0, \\ \omega^{'}(0) = n(\eta - \overline{y}), \\ \omega^{''}(0) = n(\overline{y}^2 + \sigma_y^2) = \| y \|^2 . \end{cases}$$

We impose that $\omega^{'}(0) \leq 0$ and $\omega^{''}(0) \geq 0$.

Case when f is linear

We impose that f is a linear function

$$f(x) = c^t x, \quad x, c \in \mathbb{R}^n$$

The auxiliary function ω is given by:

$$\omega(t) = \frac{c^t d}{\rho} t - (n-1) ln(1 + t\alpha) - ln(1 + t\beta)$$

Remark 9.

- *ω have the same properties as θ_2.*
- *The unique root of the equation $\omega^{'}(t) = 0$ rated by t^* is the minimum of θ_2.*
- *t^* that we have guarantee a significant decrease of the function ϕ_r along the Newton descent direction d.*

Case when f is only convex

In this case, the equation $\theta_2^{'}(t) = 0$ is no longer reduces to an equation of second degree, we thought to look at another function greater than θ_2, for this we use the secant technique.

Given $\overline{t} \in]0, T[$, for all $t \in]0, \overline{t}]$, we have

$$\frac{f(x + td) - f(x)}{\rho} \leq \frac{f(x + \overline{t}d) - f(x)}{\rho \overline{t}} t.$$

Then the auxiliary function ω is define as follows:

$$\omega(t) = n\eta t - (n-1) ln(1 + t\alpha) - ln(1 + t\beta)$$

Such as, we take

$$\eta = \frac{f(x+\bar{t}d) - f(x)}{n\rho\bar{t}}.$$

and we calculate t^* the root of the equation $\omega'(t) = 0$.

1. If $\bar{t} = 1$ and $T > 1$ then $\bar{t}$ is the optimal solution.

2. If $\bar{t} \neq 1$, then

   a If $t^* \leq \bar{t}$, in this case we have $\theta(t^*) \leq \theta_1(t^*) \leq \theta_2(t^*) \leq \omega(t^*)$, which means that we assure a significant decrease of the function ϕ_r along the direction d.

   b If $t^* > \bar{t}$, we must to choose another $\bar{t} \in]t^*, T[$ and calculate t^* for the new auxiliary function and repeat this until we have that $t^* \leq \bar{t}$, for example we choose

$$\bar{t} = t^* + \zeta(T - t^*); \quad \zeta \in [0,1].$$

Minimization of the auxiliary function ω

We have

$$\omega(t) = n\eta t - (n-1)ln(1+t\alpha) - ln(1+t\beta)$$

And

$$\omega'(t) = n\eta - (n-1)\tfrac{\alpha}{1+t\alpha} - \tfrac{\beta}{1+t\beta}$$

$$\omega''(t) = (n-1)\tfrac{\alpha^2}{(1+t\alpha)^2} + \tfrac{\beta^2}{(1+t\beta)^2}$$

For getting t^*, we need to calculate the root of the equation:

$$\omega'(t) = 0$$

Equivalent to

$$\eta\alpha\beta t^2 + (\eta(\alpha+\beta) - \alpha\beta)t + \eta - \overline{y} = 0$$

1) If $\eta = 0$, $t^* = \frac{-\overline{y}}{\alpha\beta}$.

2) If $\alpha = 0$, $t^* = \frac{\overline{y}-\eta}{\eta\beta}$.

3) If $\beta = 0$, $t^* = \frac{\overline{y}-\eta}{\eta\alpha}$.

4) If $\eta\alpha\beta \neq 0$, in this case we have two roots of the equation of the second degree but there is just only root t^* which belongs to the domain of definition of ω, both roots are:

$$t_1^* = \frac{1}{2}\left(\frac{1}{\eta} - \frac{1}{\alpha} - \frac{1}{\beta} - \sqrt{\Delta}\right), \quad t_2^* = \frac{1}{2}\left(\frac{1}{\eta} - \frac{1}{\alpha} - \frac{1}{\beta} + \sqrt{\Delta}\right).$$

In which

$$\Delta = \frac{1}{\eta^2} + \frac{1}{\alpha^2} + \frac{1}{\beta^2} - \frac{2}{\alpha\beta} - \left(\frac{2n-4}{n\eta}\right)\left(\frac{1}{\alpha} - \frac{1}{\beta}\right).$$

2.2.5 The second majorant function

Here, we considered finding another approximation of θ_1 simpler than θ_2 and has one logarithm.

Remember that:

$$\theta_1(t) = \frac{1}{\rho}\left(f(x+td) - f(x)\right) - \sum_{i=1}^{n} ln(1+ty_i);\ \rho = \overline{r} - \sigma_r\sqrt{n-1}$$

Using the inequality:

$$\sum_{i=1}^{n} \ln(1+ty_i) \geq (\| y \| + n\overline{y})t + \ln(1 - t \| y \|)$$

Then, we get a second majorant function of θ noting by θ_3 such that:

$$\theta_3(t) = \frac{1}{\rho}\left(f(x+td) - f(x)\right) - (\| y \| + n\overline{y})t - \ln(1 - t \| y \|)$$

We have

$$\theta_3'(t) = \tfrac{1}{\rho}\langle \nabla f(x+td), d\rangle - \| y \| - n\overline{y} + \tfrac{\|y\|}{1-t\|y\|}$$

$$\theta_3''(t) = \tfrac{1}{\rho}\langle \nabla^2 f(x+td)d, d\rangle + \tfrac{\|y\|^2}{(1-t\|y\|)^2} > 0$$

The domains of θ_3 is $H_3 = [0, T_3[$, with $T_3 = \max\{t; 1 - t \| y \| > 0\}$.

Lemma 4.
θ_3 is strictly convex function.

Proposition 3.
For all $t \in [0, T_3[$, we have

a) $\theta_3(0) = \theta_1(0) = 0$.

b) $\theta_3'(0) = \theta_1'(0) < 0$

c) $\theta_3''(0) = \theta_1''(0) > 0$.

d) $\theta_3(t) \leq \theta_1(t)$.

Proof.

a) We have from definition that

$$\theta_3(0) = \theta_1(0) = 0.$$

b) We have

$$\theta_3'(0) = \frac{1}{\rho}\langle \nabla f(x), d\rangle - n\overline{y} = \theta_1'(0) < 0$$

c) We have

$$\theta_3''(0) = \frac{1}{\rho}\langle \nabla^2 f(x)d, d\rangle + \| y \|^2 = \theta_1''(0) > 0.$$

d) Taking in consideration the function

$$\begin{aligned}\phi(t) &= \theta_1(t) - \theta_3(t)\\ &= -\textstyle\sum_{i=1}^n \ln(1+ty_i) + (\| y \| + n\overline{y})t + \ln(1 - t \| y \|)\end{aligned}$$

We have

$$\begin{aligned}\phi'(t) &= \theta_1'(t) - \theta_3'(t)\\ &= -\textstyle\sum_{i=1}^n \frac{y_i}{1+ty_i} + \| y \| + n\overline{y} - \frac{\|y\|}{1-t\|y\|}\\ &= \left(n\overline{y} - \textstyle\sum_{i=1}^n \frac{y_i}{1+ty_i}\right) + \left(\| y \| - \frac{\|y\|}{1-t\|y\|}\right)\\ &= \textstyle\sum_{i=1}^n \left(y_i - \frac{y_i}{1+ty_i}\right) + \left(\| y \| - \frac{\|y\|}{1-t\|y\|}\right)\\ &= t\textstyle\sum_{i=1}^n \left(\frac{y_i^2}{1+ty_i}\right) - t\left(\frac{\|y\|^2}{1-t\|y\|}\right)\\ &= t\textstyle\sum_{i=1}^n \left(\frac{y_i^2}{1+ty_i}\right) - t\left(\frac{\sum_{i=1}^n y_i^2}{1-t\|y\|}\right)\\ &= t\textstyle\sum_{i=1}^n y_i^2\left(\frac{1}{1+ty_i} - \frac{1}{1-t\|y\|}\right)\\ &\le 0\end{aligned}$$

Since we have $-\| y \| \le y_i \le \| y \|,\ \forall i = \overline{1,n}$ then $1 - t\| y \| \le 1 + ty_i\ \forall i = \overline{1,n}$. We prove that ϕ is decreasing for all $t \in [0, T_3[$ and $\phi(0) = 0$ so we come to conclusion that $\phi(t) \le 0$ so $\theta_1(t) \le \theta_3(t)$ for all $t \in [0, T_3[$.

□

θ_3 is a good approximation of θ_1 in a neighbourhood of 0, the unique minimum t^* of θ_3 guarantee a significant decreasing of the function θ_1, and we have:

$$\theta_1(t^*) \le \theta_2(t^*) \le \theta_3(t^*)$$

Minimization of an auxiliary function

Let us define the convex function ω_2, where it's minimum is reached at t^*.

$$\omega_2(t) = n\eta t - (\| y \| + n\overline{y})t - ln(1 - t \| y \|)$$

Taking into consideration that

$$\eta = \begin{cases} \frac{c^t d}{n\rho}, & \text{if } f \text{ is linear,} \\ \\ \frac{f(x+\bar{t}d)-f(x)}{n\rho\bar{t}}, & \text{if } f \text{ is convex.} \end{cases}$$

It is easy to calculate

$$\omega_2'(t) = n\eta - \| y \| - n\overline{y} + \frac{\|y\|}{1-t\|y\|}$$

$$\omega_2''(t) = \frac{\|y\|^2}{(1-t\|y\|)^2}$$

Then:

$$\omega_2(0) = 0,$$
$$\omega_2'(0) = n(\eta - \overline{y}),$$
$$\omega_2''(0) = \| y \|^2 .$$

We impose that $\omega_2'(0) \leq 0$ and $\omega_2''(0) \geq 0$.
For getting t^*, we need to calculate the root of the equation $\omega_2'(t) = 0$.

$$t^* = \frac{n(\overline{y} - \eta)}{\| y \|^2}.$$

2.3 Description of the algorithm

In this part, we present the algorithm which resume our study to obtain the optimal solution x^* of the problem (P).

2.4 Numerical tests

In the tables bellow

- Iter represents the number of iterations to obtain x^*.
- Min represents the minimum and T(s) represents the time in seconds.
- Method 1 corresponds to the method of majorant function introduced in this work.
- Method 2 corresponds to the method of majorant function introduced in [10].
- Method 3 corresponds to the classical line search method (Wolf).

Algorithm 1 The general algorithm using majorant function technique

1. **Input** $\epsilon > 0$, $r_s > 0$, $x_0 \in K$, X with $X(i,i) = x_0(i)$, $r \in \mathbb{R}^n_+$, $\delta \in [0,1]^n$.

2. **Iteration**

   * Calculate d and $y = X^{-1}d$

   (a) If $\| y \| > \epsilon$ do

   i Calculate η, α, β and solve the equation $\omega^{'}(t) = 0$ for obtain t^* .

   ii Calculate $x = x + t^* d$, and return to $(*)$

   (b) If $\| y \| \leq \epsilon$, we obtained a good approximation of $m(r)$.

   i If $\|r\| \geq r_s$, $r = \delta \times r$ and return to $(*)$.
   With $\delta \times r = (\delta_1 \times r_1, ..., \delta_n \times r_n)$.

   ii If $\|r\| \leq r_s$, Stop. We have a good approximation of the optimal solution .

We indicate by O.F. and O.S. the objective function and the optimal solution respectively.

Remark 10.
In the numerical examples, we use the first majorant function to calculate the step size.

2.4.1 Nonlinear convex objective

Example 1.
The two following examples are written as follows:

$$\begin{cases} \min\limits_{x \in \mathbb{R}^n} f(x) \\ \\ Ax = b \\ \\ x \geq 0 \end{cases}$$

And $f^* = f(x^*)$.

1. ***O.F***

$$f(x) = x_1^2 + x_2^2 - 3x_1 - 5x_2$$
$$A = \begin{pmatrix} 1 & 2 & 1 & 0 \\ 3 & 1 & 0 & 1 \end{pmatrix}, \quad b = \begin{pmatrix} 4 \\ 7 \end{pmatrix}$$

O.S

$$x^* = (1, \frac{3}{2}, 0, \frac{5}{2}), \ and \ f^* = -7,25.$$

2. ***O.F***

$$f(x) = x_1^2 + x_2^2 - 2x_1 - x_2$$

$$A = \begin{pmatrix} 1 & 4 & 1 & 0 \\ 2 & 3 & 0 & 1 \end{pmatrix}, \quad b = \begin{pmatrix} 5 \\ 6 \end{pmatrix}$$

O.S

$$x^* = (\frac{13}{7}, \frac{18}{17}, 0, \frac{22}{17}), \ and \ f^* = -4,059.$$

2.4.2 Example with variable size

The objective function is linear

Example 2.
We consider a linear optimization problem

$$f^* = \min\{a^t x : \ Ax = b, \ x \geq 0\}.$$

In which A is the $m * 2m$ matrix defined by:

$$A[i,j] = \begin{cases} 1 \ if \ i = j \ or \ j = 1 + m, \\ 0 \ if \ not. \end{cases}$$

$a \in \mathbb{R}^{2m}$, *and* $b \in \mathbb{R}^m$ *are defined by:*

$$a[i] = -1, \ a[i+m] = 0, \ b[i] = 2 \ \forall i = \overline{1, m}.$$

O.S
$f^* = -2m.$

The objective function is nonlinear

Example 3. *Quadratic case [35]*

We consider the following quadratic problem with $n = m + 2$

$$f^* = \min\{f(x) : Ax = b, x \geq 0\}.$$

In which:

$$f(x) = \frac{1}{2}\langle x, Qx \rangle.$$

With

$$Q[i,j] = \begin{cases} 2 & if\ i = j = 1\ or\ i = j = m, \\ 4 & if\ i = j\ and\ i \neq \{1, m\}, \\ 2 & if\ i = j - 1\ or\ i = j + 1, \\ 0 & otherwise. \end{cases}$$

$$A[i,j] = \begin{cases} 1 & if\ i = j, \\ 2 & if\ i = j - 1, \\ 3 & if\ i = j - 2, \\ 0 & otherwise. \end{cases}$$

$f_i = 1,\ \forall i = \overline{1, n},\ \forall j = \overline{1, m}.$
We test this example for different value of n.

Example 4. *Erikson's problem*

Consider the following convex problem:

$$f^* = \min[f(x) :\ Ax = b,\ x \geq 0].$$

Where $f(x) = \sum_{i=1}^{n} x_i ln\left(\frac{x_i}{a_i}\right)$, $a_i, b_i \in \mathbb{R}$ *are fixed, and*

$$A[i,j] = \begin{cases} 1\ if\ i = j\ or\ j = i + m, \\ 0\ if\ not. \end{cases}$$

We test this example for different values of n, a_i *and* b_i.

2.5 Tables

Table 2.1: Numerical simulations for example 1

	Method 1			Method 2			Method 3		
Example	Min	Iter	T(s)	Min	Iter	T(s)	Min	Iter	T(s)
1	-7.17	3	0.0009	-7.17	4	0.0010	-7.18	9	0.0100
2	-4.04	4	0.0012	-4.05	7	0.0020	-4.05	10	0.0100

Table 2.2: Numerical simulations for example 2

	Method 1			Method 2			Method 3		
n	Min	Iter	T(s)	Min	Iter	T(s)	Min	Iter	T(s)
5	-9.91	7	0.0007	-9.96	9	0.001	-9.79	9	0.018
25	-50.01	8	0.026	-49.8	9	0.03	-49.67	11	0.049
50	-99.86	8	0.14	-99.6	9	0.17	-99.35	11	0.22
100	-199.53	9	1.25	-199.21	9	1.21	-199.60	13	1.79
200	-399.01	8	10.9200	-398.43	9	11.9650	-399.21	13	18.4203
500	-994.24	12	22.4312	-992.34	14	25.9720	-993.34	19	29.2319

Table 2.3: Numerical simulations for example 3

	Method 1			Method 2			Method 3		
n	Min	Iter	T(s)	Min	Iter	T(s)	Min	Iter	T(s)
4	0.285	8	0.0061	0.285	9	0.007	0.285	14	0.019
50	5.37	6	0.019	5.372	7	0.021	5.374	14	0.053
100	10.924	6	0.065	10.927	7	0.08	10.93	14	0.188
500	55.3722	8	14.5	55.372	7	13.8	55.374	14	29.815

2.6 Comments

The objective of this work is solving a nonlinear problem (P) with a logarithmic barrier interior point method based on a majorant function technique which utilized to calculate the step size instead of the line search method.
The numerical tests prove the efficiency of our approach.
In table (2.1), we evaluate the effectiveness of our approach using two quadratic examples with fixed size $n = 4$, we remark at the first that the number of iteration when we compared method 1 to 3 is about the half and in less execution time. We also notice a small difference between method 1 and 2 in the number of iteration or in the time.

Table 2.4: The case where $a_i = 1$ and $b_i = 6, \forall i = \overline{1.n}$ (Example 4)

	Method 1			Method 2			Method 3		
n	Min	Iter	T(s)	Min	Iter	T(s)	Min	Iter	T(s)
10	32.94	3	0.0038	32.95	3	0.004	32.95	4	0.0180
50	164.79	4	0.02600	164.79	4	0.0250	164.79	6	0.0820
100	329.57	5	0.1011	329.58	5	0.0760	329.58	6	0.2300
500	1647.34	9	0.2300	1646.04	10	0.2420	1647.01	12	0.4320
1000	3295.54	12	0.4500	3297.54	13	0.5100	3294.42	15	0.6800

Table 2.5: The case where $a_i = 2$ and $b_i = 5, \forall i = \overline{1.n}$ (Example 4)

	Method 1			Method 2			Method 3		
n	Min	Iter	T(s)	Min	Iter	T(s)	Min	Iter	T(s)
10	0.62×10^{-8}	2	0.0021	0.66×10^{-7}	2	0.002	9.09×10^{-6}	3	0.01
50	0.1×10^{-7}	3	0.0028	0.11×10^{-8}	3	0.003	1.86×10^{-4}	4	0.07
500	0.75×10^{-7}	3	0.0045	0.76×10^{-7}	3	0.04	1.0003×10^{-4}	5	0.22

In table (2.2), we study a quadratic example with variable size, we change n at each time. As in table 1 Method 1 and 2 are close with a small advantage to the first. On the other hand method 1 takes less time with fewer number of iterations than method 3 which makes our approach more effective than the line search.
In table (2.3), we evaluate the effectiveness of our approach using a linear program with variable size. We studied it with different value of n. We notice at the start that the number in method 1 is less than the number of iterations method 3,

Table (2.4) and (2.5) present a nonlinear program which is evaluated with various values of n, a_i and b_i, as shown in the tables above. Our approach takes less time and requires fewer iterations than method 3 and competes with method 2.

These tests clearly demonstrate the influence of our approach compared to the classical line search method, as shown in the tables by a reduction in the number of iterations and a fewer the calculation time. However, it competes with the method introduced in [10] where $r \in \mathbb{R}$.

CHAPTER 3

AN INVERSE BARRIER PENALTY METHOD FOR NONLINEAR PROGRAMMING

3.1 Introduction

In this chapter, we thought to use a new inverse barrier approach to solve a nonlinear programming problem.
The following is how this chapter is organized: In the second section, we define the perturbed problem of (P) using the inverse barrier function in [15] that conserves the properties of the first objective function, investigate the existence and uniqueness of the perturbed problem's optimal solution, and demonstrate its convergence to the optimal solution of the original problem.
The resolution of the perturbed problem is the focus of the third section. First, we calculated the Newton descent direction, and then we used a tangent technique to calculate the step size along this direction.

The numerical experiments presented in the final section demonstrate the effectiveness of our approach.

3.1.1 The problem formulation

The problem under consideration in this chapter is given by:

$$f^* = \min \{f(x);\ x \in Y\} \qquad (P)$$

Under the following assumptions:

a) f is a non linear, convex, twice continuously differentiable function on Y,

$$Y = \{x \in R^n;\ Ax = b,\ x \geq 0\}$$

is called the set of feasible solution of (P).

And

$$Y^0 = \{x \in R^n;\ Ax = b,\ x > 0\}$$

is called the set of strictly feasible solution of (P).

b) $A \in R^{m\times n}$ is a full rank matrix, $b \in R^m, (m < n)$.

c) There exists $x^0 > 0$ such that $Ax^0 = b$.

d) The set of optimal solutions of (P) is non empty and bounded.

x^* is an optimal solution of (P), if there exist two Lagrange multipliers. $\lambda^* \in R^m, \mu^* \in R^n_+$ such as:

$$\begin{cases} \nabla f(x^*) + A^t\lambda^* - \mu^* = 0, \\ Ax^* = b, \\ \langle \mu^*, x^* \rangle = 0. \end{cases} \tag{3.1}$$

3.2 The penalization

The penalty method consists in replacing the constrained programming (P) with a simpler constrained problem or an unconstrained one, hence the non-negativity constraints $x_i > 0$ are replaced by an inverse barrier function $r\sum_{i=1}^n \frac{1}{x_i^r}$, in which $r > 0$.

The following perturbed problem (P_r) is associated to the original problem (P):

$$(P_r) \quad \begin{cases} \min f_r(x) \\ x \in R^n \end{cases}$$

In which

$$f_r(x) = \begin{cases} f(x) + r\sum_{i=1}^n \frac{1}{x_i^r}, & \text{if } Ax = b, x \geq 0, \\ \infty & \text{if not} \end{cases}$$

$r > 0$ is called barrier term.

The first and the second order derivatives of f_r are given by

$$\nabla f_r(x) = \nabla f(x) - r^2 X^{-r-1}e$$

$$\nabla^2 f_r(x) = \nabla^2 f(x) + r^2(r+1)X^{-r-2}$$

In which X is the diagonal matrix with diagonal entries $X_{ii} = x_i, \forall i = \overline{1, n}$, and $e = (1, 1, ..., 1)^t \in R^n$.

f_r is a strictly convex, proper and lower semi continuous function.

3.2.1 Existence and uniqueness of optimal solution of perturbed problem

Lemma 5.
For (P_r) admits a unique optimal solution, let f_r be an inf compact and strictly convex function.

Proof.
According to assumption d, (P) admits a unique optimal solution which means that the cone of recession C_f of f is reduced to zero, we have:

$$C_f = \{d \in R^n;\ [f]_\infty(d) \leq 0,\ Ad = 0,\ d \geq 0\} = \{0\}$$

$[f]_\infty(d)$ is the asymptotic function of f, which defined by:

$$[f]_\infty(d) = \lim_{\alpha \to +\infty} \frac{f(x_0 + \alpha d) - f(x_0)}{\alpha}$$

To demonstrate that (P_r) admits a unique optimal solution, suffice it to demonstrate that the cone of recession of f_r is reduced to zero.

We notice that the two asymptotic function of f and f_r respectively are related by

$$[f_r]_\infty = \begin{cases} [f]_\infty(d) & \text{if} \quad Ad = 0,\ d \geq 0, \\ \\ \infty \text{ if not.} \end{cases}$$

Then we deduce that: $\{d \in R^n;\ [f_r]_\infty \leq 0\} = \{0\}$, which means that $C_{f_r} = \{0\}$.

Furthermore, because f_r is a strictly convex function, (P_r) admits only an optimal solution $x(r)$. □

In the rest we denote $x(r)$ by x_r.

3.2.2 The convergence of (P_r)

Lemma 6.
For $r > 0$, we assume that x^* and x_r are the optimal solutions of the original problem (P) and the perturbed problem (P_r) respectively, then we have

$$\lim_{r \to 0} x_r = x^*$$

Proof.
Let us begin by defining the differentiable convex function on R_+^* by:

$$m(r) = f(x_r) + r \sum_{i=1}^{n} \frac{1}{x_i^r(r)}$$

When $r \to 0$, the two problems (P) and (P_r) coincide, i.e.

$$m(0) = f^* = \min f(x)$$

We have

$$m'(r) \quad = \langle \nabla f_r(x_r), x'_r \rangle + \sum_{i=1}^n \frac{1}{x_i^r(r)} - r \sum_{i=1}^n \frac{\ln(x_i)}{x_i^r(r)}$$

As we suppose that x_r is the optimal solution of (P_r), which implies that $\nabla f_r(x_r) = 0$, then

$$m'(r) = \sum_{i=1}^n \frac{1}{x_i^r(r)} - \sum_{i=1}^n \frac{\ln(x_i^r(r))}{x_i^r(r)}$$

Since m is convex, we have

$$m(0) \geq m(r) - rm'(r) = f(x_r) - r \sum_{i=1}^n \frac{\ln\left(\frac{1}{x_i^r(r)}\right)}{x_i^r(r)}$$

We suppose that $z_i = \frac{1}{x_i^r(r)}$.
Then

$$m(0) \geq f(x_r) - r \sum_{i=1}^n z_i \ln z_i$$

According to proposition 2, we have

$$z_i \ \leq \ \gamma, \ \ \gamma = \overline{z} + \sigma_z \sqrt{n-1} \Rightarrow \ -z_i \ \geq \ -\gamma$$

According to theorem 7, we have

$$\sum_{i=1}^n \ln(z_i) \leq A_2$$

Then $m(0) \geq f(x_r) - r\gamma A_2$
Then $f(x_r) - nr\gamma A_2 \leq f^* = m(0) \leq f(x_r)$
Now, we are interested on the trajectory when $r \to 0$. We have the two following cases:

a) The case in which f is strongly convex with coefficient δ strictly positive.

We have

$$\begin{aligned} r\gamma A_2 \quad &\geq f(x_r) - f(x^*) \\ &\geq \langle \nabla f(x^*), x_r - x^* \rangle + \tfrac{\delta}{2} \parallel x_r - x^* \parallel^2 \\ &\geq \tfrac{\delta}{2} \parallel x_r - x^* \parallel^2 \end{aligned}$$

Then $\parallel x(r) - x^* \parallel \leq \ \sqrt{2r\gamma A_2}$

b) The case in which b is only convex.

This case is a little complicated, we impose that $r \leq 1$, and for that we note

$$x(r) \in \{x; \ Ax = b, \ x > 0, \ f(x) \ \leq \ n\gamma A_2 + f^*\}$$

This set is convex, bounded, and non-empty, and its cone of recession is reduced to zero.
As a result, each accumulation point of x_r is an optimal solution of (P) if $r \to 0$.

□

3.3 Solving the perturbed problem

In this part, we are interested in the numerical solution of the perturbed problem (P_r).

3.3.1 The Newton descent direction

Several methods exist for calculating the descent direction d. In this work, we choose Newton's method, so d is obtained by solving the quadratic convex minimization problem:

$$\begin{cases} \min_d \frac{1}{2} d^t \nabla^2 f_r(x) d + \nabla^t f_r(x) d \\ \\ Ad = 0. \end{cases}$$

Using the necessary and sufficient optimality conditions, there exists $\lambda \in R^m$ such that:

$$\begin{cases} \nabla^2 f_r(x) d + \nabla f_r(x) + A^t \lambda = 0 \\ \\ Ad = 0. \end{cases}$$

In which

$$\begin{cases} H(x) = \nabla^2 f_r(x) = \nabla^2 f(x) + r^2(r+1) X^{-r-2} \\ \\ h(x) = \nabla f_r(x) = \nabla f(x) - r^2 X^{-r-1} e \\ \\ C = \begin{pmatrix} H(x) & A^t \\ \\ A & 0 \end{pmatrix} \end{cases}$$

Then, we get

$$C \begin{pmatrix} d \\ \\ \lambda \end{pmatrix} = \begin{pmatrix} -h(x) \\ \\ 0 \end{pmatrix} \quad (*)$$

By multiplying $(*)$ by the vector $(d^t \quad 0)$, we get the following system

$$(d^t \quad 0)\, C \begin{pmatrix} d \\ \\ \lambda \end{pmatrix} = (d^t \quad 0) \begin{pmatrix} -h(x) \\ \\ 0 \end{pmatrix} \quad (**)$$

The system $(**)$ is equivalent to

$$\langle \nabla^2 f(x)d, d\rangle + \langle \nabla f(x), d\rangle = r^2(\langle X^{-r-1}d, e\rangle - (r+1) \parallel X^{-\frac{r}{2}-1}d \parallel^2)(3.2)$$

3.3.2 The step size

In this part, we will calculate the step size using a tangent technique rather than the classical line search methods, which require minimizing a uni dimensional search line function $\theta : R^*_+ \to R$ to find the optimal solution α^* which given a significant decrease of the following convex function:

$$\begin{aligned} \theta(\alpha) &= f_r(x+\alpha d) \\ &= f(x+\alpha d) + r\sum_{i=1}^{n} \frac{1}{(x_i+\alpha d_i)^r} \end{aligned}$$

We have

$$\theta^{'}(\alpha) = \langle \nabla f(x+\alpha d), d\rangle - r^2 \sum_{i=1}^{n} \frac{d_i}{(x_i+\alpha d_i)^{r+1}}$$

$$\theta^{''}(\alpha) = \langle \nabla^2 f(x+\alpha d), d\rangle + r^2(r+1) \sum_{i=1}^{n} \frac{d_i^2}{(x_i+\alpha d_i)^{r+2}}$$

Remark 11.

1. *To ensure that the search line function θ is well defined, it is necessary that the point $x+\alpha d \in Y$, i.e. θ is a convex function.*
2. *According to* (2)*, we have $\theta^{'}(0) + \theta^{''}(0) = 0$ and such as $\theta^{''}(\alpha) \geq 0$ ie $\theta^{''}(0) \geq 0$, which give us that $\theta^{'}(0) \leq 0$.*

Because of the complex form in which the function θ is defined, solving the equation $\theta^{'}(\alpha) = 0$ explicitly to obtain the optimal step size $alpha*$ is difficult. To avoid this difficulty, we proposed another method, which is described below.

Tangent technique

Using the tangent method allowed us to calculate the step size α^* in a more straightforward manner than the classical methods. Our algorithm terminated when the condition $\theta^{'}(\alpha^*) < \epsilon$ was satisfied.

Description of the method

This technique is divided into two phases, which are described below:

Phase 1:

Obtain the domain $I_k =]\alpha_k, \alpha_{k+1}[$ such that its bounds satisfy the condition:

$$\theta'(\alpha_k).\theta'(\alpha_{k+1}) < 0$$

Which means that the optimal step size $\alpha^* \in I_k$.

Phase 2:

a) Draw the tangents T_k and T_{k+1} at the points α_k and α_{k+1}.

$$T_k : \theta(\alpha) = \theta(\alpha_k) + (\alpha - \alpha_k)\theta^{'}(\alpha_k)$$

$$T_{k+1} : \theta(\alpha) = \theta(\alpha_{k+1}) + (\alpha - \alpha_{k+1})\theta^{'}(\alpha_{k+1})$$

b) Determine the point $\alpha_k^{'}$ corresponding to the intersection of the two tangents T_k and T_{k+1}.

$$\alpha_k^{'} = \frac{\theta(\alpha_{k+1}) - \theta(\alpha_k) + \alpha_k\theta^{'}(\alpha_k) - \alpha_{k+1}\theta'(\alpha_{k+1})}{\theta'(\alpha_k) - \theta'(\alpha_{k+1})}$$

We have two cases:

* If $\theta^{'}(\alpha_k^{'}) < \epsilon$ stop.
* If $\theta^{'}(\alpha_k^{'}) > \epsilon$, we have two cases:
  - If $\theta'(\alpha_k).\theta'(\alpha_k^{'}) < 0$, put

$$\alpha_k = \alpha_k$$

$$\alpha_k^{'} = \alpha_{k+1}$$

    And return to **a**.
  - If $\theta'(\alpha_k^{'}).\theta'(\alpha_{k+1}) < 0$, put

$$\alpha_k^{'} = \alpha_k$$

$$\alpha_{k+1} = \alpha_{k+1}$$

    And return to **a**.

3.4 Description of the algorithm

In this section, we present the algorithm that will allow us to resume our investigation in order to find the optimal solution to the original problem (P).

Algorithm 2 The general algorithm using the tangent technique (Inverse barrier method)

1. **Input** $\epsilon > 0$, $\overline{r} > 0$, $x_0 \in Y^0$, X with $X(i,i) = x_0(i)$, $\rho \in]0,1[$.
2. **Iteration**
   (a) Calculate d and $h = X^{-1-r}d$.
   (b) If $\| h \| > \epsilon$
      - Calculate α^* according to the tangent technique.
      - Calculate $x = x + \alpha^* d$.

   (c) If $\| h \| < \epsilon$
      - If $r > \overline{r}$: $r = \rho r$, go to 1.
      - If $r < \overline{r}$: Stop, we have a good approximation of the optimal solution.

3.4.1 Numerical experiments

The two following examples are written as follows:

$$f^* = \min \{ f(x) : \ Ax = b, \ x \geq 0 \} .$$

Example 5.
$f(x) = x_1^2 + x_2^2 - 3x_1 - 5x_2$

$$A = \begin{pmatrix} 1 & 2 & 1 & 0 \\ 3 & 1 & 0 & 1 \end{pmatrix}, \quad b = \begin{pmatrix} 4 \\ 7 \end{pmatrix}$$

The optimal solution is

$$x^* = (1, \frac{3}{2}, 0, \frac{5}{2}), \ \textit{and} \ f^* = -7,25.$$

Example 6.
$f(x) = x_1^2 + x_2^2 - 2x_1 - x_2$

$$A = \begin{pmatrix} 1 & 4 & 1 & 0 \\ 2 & 3 & 0 & 1 \end{pmatrix}, \quad b = \begin{pmatrix} 5 \\ 6 \end{pmatrix}$$

The optimal solution is

$$x^* = (\frac{13}{7}, \frac{18}{17}, 0, \frac{22}{17}), \ \textit{and} \ f^* = -4,059.$$

Example 7. *Erikson's problem.*
Consider the following convex problem:

$$f^* = \min[f(x) : \ Ax = b, \ x \geq 0].$$

Where $f(x) = \sum_{i=1}^{n} x_i ln\left(\frac{x_i}{a_i}\right)$, $a_i, b_i \in R$ *are fixed, and*

$$A[i,j] = \begin{cases} 1 & if\ \ i = j\ or\ j = i+m, \\ 0 & if\ not\ . \end{cases}$$

We test this example for different values of n, a_i *and* b_i.

Tables:

In the following tables:

Iter represents the number of iterations to obtain x^*. Min represents the minimum and T(s) represents the time in seconds. Method 1 corresponds to the tangent technique introduced in this work.

Method 2 corresponds to the classical line search method (Wolf).

Table 3.1: Numerical simulations for example 6 and 7

	Method 1			Method 2		
Example	Min	Iter	T(s)	Min	Iter	T(s))
1	-7.21	4	0.025	-7.13	6	0.062
2	-4.04	24	0.061	-4.05	36	0.075

Table 3.2: The case where $a_i = 2$ and $b_i = 4, \forall i = \overline{1.n}$ (Example 8)

	Method 1			Method 2		
n	Min	Iter	T(s)	Min	Iter	T(s)
10	$0.75\ \times 10^{-6}$	3	0.19	$0.86\ \times 10^{-4}$	4	0.21
50	$0.2\ \times 10^{-6}$	3	0.16	$0.275\ \times 10^{-4}$	5	0.23
200	$0.72\ \times 10^{-7}$	4	0.22	$0.2001\ \times 10^{-3}$	6	0.32

Table 3.3: The case where $a_i = 1$ and $b_i = 6, \forall i = \overline{1.n}$ (Example 8)

	Method 1			Method 2		
n	Min	Iter	T(s)	Min	Iter	T(s)
10	32.95	3	0.07	32.95	5	0.19
50	164.77	3	0.18	164.76	6	0.25
200	659.07	4	0.24	659.04	7	0.38

3.5 Comments

The goal of this work is to solve a nonlinear programming problem using a new interior point method based on an inverse barrier function which utilized for the first time in this work to solve the nonlinear optimization problems. The numerical tests described above demonstrate the effectiveness of our approach, as we observe fewer iterations in less times when compared to the classical line search method (Wolf).

In table (3.1), we evaluate the effectiveness of our approach using two quadratic programs which are a type of nonlinear programming (Example 6 and 7). We notice at the start that the number of iterations in our technique "the inverse barrier method" (method 1) is less than the number of iterations in the classical line searches (method 2), and on the other hand method 1 takes less time than method 2 which makes our approach more effective than the line searches.

Table (3.2) and (3.3) present a nonlinear model which is evaluated with various values of n, a_i and b_i, as shown in the tables above. Like the previous quadratic problems, our approach takes less time and requires fewer iterations than method 2. From all the aforementioned, one can conclude the effectiveness of our approach when compared with the classical line search method.

CHAPTER 4

A PROJECTIVE INTERIOR POINT METHOD FOR NONLINEAR PROGRAMMING

4.1 Introduction

In this chapter, we proposed an algorithm for solving a convex problem, having an idea to extend the Karmarkar algorithm to the non linear problems. calculation of feasible solution in each iteration and the fact that the objective function value must decrease in each iteration with the fixed desired tolerance.First, we briefly recall the algorithm proposed by Karmarkar for solving the linear problems. In addition, we present our approach to solve the nonlinear problem which is separated into two parts: linearizing the objective function and calculating the step size using a majorant function technique. Finally, we present some numerical test which prove the importance of our algorithm.

4.1.1 Karmarkar method

We consider the following linear optimization programming:

$$(LP)\begin{cases} \min c^t x = z^* \\ Ax = 0 \\ x \in S_n \end{cases}$$

In which:

$$S_n = \{x \in \mathbb{R}^n : \ e_n^t x = 1, \ x \geq 0\}$$

is the simplex of dimension n and of center a such that:

$$a_i = \frac{1}{n}, \ \forall i = \overline{1, n}$$

.

Hypotheses

H1) The optimal value $z^* = 0$.

H2) x^0 is a strictly realizable point such that: $Ax^0 = 0, x^0 \in S_n$, we can take $x^0 = \frac{1}{n}e_n$.

H3) $A \in \mathbb{R}^{p \times n}$ is a full rank matrix.

Remark 12.

1. *If $z^* \neq 0$, using the equality $e_n^t x = 1$, we get that*

$$c^t x = z^* e_n^t x \Rightarrow (c - z^* e_n)^t x = 0 \Rightarrow (c^{'})^t x = 0$$

The condition (z^* is known) is restrictive in practice. For this reason, the researchers proposed different approximations of z^* as the upper and lower bounds, or both at the same time.
The most used variant is the Ye-Lustig approximation, which proposed a majoration of z^* [41] such that:

$$z^k = c^t x^k > z^*$$

.

2. *If $b \neq 0$ using the equality $e_n^t x = 1$ we get that*

$$Ax = be_n^t x \Rightarrow A^{'} x = (A - be_n^t)x = 0$$

Description of the algorithm

starting from the initial point x^0, the algorithm constructs a series of interior points which converges to the optimal solution of the problem in polynomial time. In order to reduce the objective function to zero, we minimized locally on a sphere inscribed in the feasible region.
At each iteration k, the iterated x^k is brought back to the center of the simplex S using a projective transformation T_k define by:

$$T_k : S_n \quad \rightarrow S_n$$

$$x \quad \mapsto y = T_k(x) = \frac{X_k^{-1}x}{e_n^t X_k^{-1} x}$$

We can write

$$x = T_k^{-1}(y) = \frac{X_k y}{e_n^t X_k y}$$

In which: X_k is the diagonal matrix with diagonal entries $(X_k)_{ii} = (x^k)_i, \forall i = \overline{1,n}$.
The problem transformed The linear transformed programming of (LP) is given by:

$$\begin{cases} \min \frac{c^t X_k y}{e_n^t X_k y} = 0 \\ A \frac{X_k y}{e_n^t X_k y} = 0 \\ e_n^t y = 1 \\ y \geq 0 \end{cases}$$

Which equivalent to

$$\begin{cases} \min c^t X_k y = 0 \\ A X_k y = 0 \\ e_n^t y = 1 \\ y \geq 0 \end{cases} \tag{4.1}$$

Lemma 7. *[25]*
If we know a feasible solution y^0 for a linear programming, in which y_i^0 are all positive fori $= \overline{1,n}$, then the ellipsoid

$$E = \left\{ y \in \mathbb{R}^n; \sum_{i=1}^{n} \left(\frac{y_i - y_i^0}{y_i^0} \right)^2 \leq \beta^2, \ 0 < \beta < 1 \right\}$$

is in the interior of the positive orthant of $\mathbb{R}^n$.

The problem (4.1) can be written under the form:

$$\begin{cases} \min c^t X_k y = 0 \\ A X_k y = 0 \\ e_n^t y = 1 \\ \| y - a \|^2 \leq (tr)^2 \end{cases} \tag{4.2}$$

In which, $0 < t < 1$ and $r = \frac{1}{\sqrt{n(n-1)}}$.

Theorem 6.

The optimal solution of (4.2) *is given explicitly by:* $y^k = a - trd^k$, *in which*

$$\begin{cases} d^k = \frac{p^k}{\|p^k\|}, \\ p^k = (I - B_k^t(B_kB_k^t)^{-1}B_k)cX_k, \\ B_k = \begin{bmatrix} A^k \\ e_n^t \end{bmatrix}. \end{cases}$$

Algorithm 3 The algorithm proposed by Karmarkar for linear problem

Input

$k = 0 : \epsilon > 0, x^0 = a = \frac{1}{n}e_n,\ r = \frac{1}{\sqrt{n(n-1)}}$.

Iteration

1. If $c^tx^k > 0$ do
   $(X_k)_{ii} = (x^k)_i,\ A_k = AX_k,\ B_k = \begin{bmatrix} A_k \\ e_n^t \end{bmatrix}$.
   Calculate $p_k = (I - B_k^t(B_kB_k^t)^{-1}B_k)cX_k$,
   $d^k = \frac{p^k}{\|p^k\|}$,
   $y^k = a - trd^k,\ 0 < t < 1$.
2. $x^{k+1} = T_k^{-1}(y^k) = \frac{X_ky^k}{e_n^tX_ky^k}$,
   $k = k+1$ and return to 1.

Stop.

Convergence

To control the convergence karmarkar introduces the potential function defined by:

$$P(x) = \sum_{i=1}^{n} \ln\left(\frac{c^tx}{x_i}\right)$$

Theorem 7. *[27]*
The point y^k *verified:*

$$\frac{c^tX_ky^k}{c^tX_ka} \leq 1 - \frac{t}{n-1}.$$

Lemma 8. *[27]*
For x^k *is the* k^{th} *iteration of the algorithm, we have*

$$\frac{c^tx^k}{c^tx^0} \leq \left(\exp(P(x^k) - P(x^0))\right)^{\frac{1}{n}}$$

Which means that if $P(x^k)$ tends to $-\infty$, then $z^k = c^t x^k$ tends to zero.

Theorem 8. *[27]*
For $0 < t \leq \frac{1}{4}$, then we start from $x^0 = a$, the algorithm find the feasible point x after $O(nq + n \ln n)$ iterations such that:

a) $c^t x = 0$.

Or

b) $\frac{c^t x}{c^t x^0} \leq \epsilon = 2^{-q}$, *in wich q is a fixed precision.*

4.1.2 Generalization of the algorithm

Either the following linear programming:

$$\begin{cases} \min c^t x = z^* \\ Ax = b \\ x \geq 0 \end{cases} \tag{4.3}$$

Hypotheses

H1) The optimal value $z^* = 0$.

H2) x^0 is a strictly realizable point such that: $Ax^0 = b, x^0 > 0$.

H3) $A \in \mathbb{R}^{p \times n}$ is a full rank matrix.

The projective transformation is defined by $T_k : \mathbb{R}^n_+ \to S^{n+1}$ such that:

$$T_k(x) = \begin{cases} y_i = \frac{\frac{x_i}{x_i^k}}{1+\sum_{i=1}^n \frac{x_i}{x_i^k}}; \qquad i = \overline{1, n} \\ \\ y_{n+1} = 1 - \sum_{i=1}^n y_i \end{cases}$$

It's simple to see that:

$$x = T_k^{-1}(x) = \frac{X_k y[n]}{y_{n+1}}$$

In which: $X_k = diag(x^k)$ and $y[n] = (y_1, ..., y_n)^t$.
By applying the projective transformation, we get that the following problem

$$\begin{cases} \min c^t \frac{X_k y[n]}{y_{n+1}} = z^* \\ \\ A \frac{X_k y[n]}{y_{n+1}} = b \\ \\ \sum_{i=1}^n y_i = 1 \\ \\ y[n] \geq 0, y_{n+1} \geq 0 \end{cases}$$

Which equivalent to

$$\begin{cases} \min (c^k)^t y \\ A^k y = 0 \\ \sum_{i=1}^{n+1} y_i = 1 \\ y \geq 0 \end{cases} \tag{4.4}$$

In which: $c^k = (c^t X_k \quad - z^*)$, $A^k = (AD_k \quad - b)$ and $y = \begin{pmatrix} y[n] \\ y_{n+1} \end{pmatrix}$

Taking into consideration that we can replace the condition $y \geq 0$ by the sphere $S(a, \alpha r)$, since if we have $y \in S(a, \alpha r)$ then $y \geq 0$ (here we cite karmarkar), In which $r = \frac{1}{\sqrt{n(n-1)}}$ and $0 < \alpha < 1$.

Now, we can write (4.4) in the following form:

$$\begin{cases} \min (c^k)^t y \\ B_k y = 0 \\ \| y - a \| \leq \alpha r \end{cases} \tag{4.5}$$

Where $a = \frac{1}{n+1} e_{n+1}$, $B_k = \begin{pmatrix} A^k \\ e^t_{n+1} \end{pmatrix}$ and $y = \begin{pmatrix} y[n] \\ y_{n+1} \end{pmatrix}$

Lemma 9.
The optimal solution of (4.5) is given by

$$y^k = a - \alpha r d_k$$

In which $d_k = \frac{p_k}{\|p_k\|}$, $p_k = (I - B_k^t (B_k B_k^t)^{-1} B_k) c^k$

Remark 13.
We notice that each feasible solution of (4.3) can be transformed by T_k into a feasible solution of (4.5) and reciprocally each feasible solution of (4.5) with $y_{n+1} \geq 0$ can be transformed by T_k^{-1} into a feasible solution of (4.3).

4.2 Extension of the algorithm

Here, we are interested in the resolution of the nonlinear problem (P) using the reduction of Karmarkar, we develop a new approach based on the technique of majorant function [12][2] which facilitates the calculate of the step size instead of the potential function used by Karmarkar [21]. The main difficult in our study is the linearization of the objective function. The step size is calculated using a majorant function technique.

4.2.1 The problem formulation

The problem to be studied in this paper is as follows:

$$\begin{cases} \min_x f(x) = z^* \\ Ax = b \\ x \geq 0 \end{cases} \tag{4.6}$$

In which: f is a nonlinear, convex and differentiable function.

Hypotheses:

H1) $A \in \mathbb{R}^{m\times n}$ is a full rank matrix.

H2) There exists $x^0 > 0$ such that: $Ax^0 = b$.

H3) The optimal value $z^* = f(x^*)$ is known from the beginning.

The problem transformed:

The Karmarkar's projective transformation T_k is define as follows:

$$T_k : \mathbb{R}^n \to S_{n+1}$$

$$x \mapsto y = T_k(x)$$

Where

$$\begin{cases} y_i = \frac{\frac{x_i}{x_i^k}}{1+\sum_{j=1}^n \frac{x_j}{x_j^k}}, & i = \overline{1,n} \\ y_{n+1} = 1 - \sum_{i=1}^n y_i \end{cases}$$

In which

$$S_{n+1} = \{y \in \mathbb{R}^{n+1}; \sum_{i=1}^{n+1} y_i = 1,\ y \geq 0\}$$

is the simplex of dimension n and of center $a_i = \frac{1}{n+1}$ for all $i = \overline{1, n+1}$.
It's clearly that we can write

$$x = T_k^{-1}(y) = \frac{X^k y[n]}{y_{n+1}}$$

such that:

$$y[n] = y_{n+1}\left((X_k)^{-1}x\right) = (y_i)_{i=1}^n,$$

Where X^k is the diagonal matrix with diagonal entries $X_{ii}^k = x_i^k, \forall i = \overline{1,n}$, and x^k is the strictly feasible point of (4.6).
The problem (4.6) is transformed as follows:

$$\begin{cases} \min\limits_{y} h(y) = y_{n+1}(f(T_k^{-1}(y)) - z^*) \\ AT_k^{-1}(y) = b \\ \sum_{i=1}^{n+1} y_i = 1 \\ y_i \geq \ 0, \ \forall i = \overline{1,n} \end{cases} \tag{4.7}$$

Which equivalent to

$$\begin{cases} \min\limits_{y} h(y) = y_{n+1}(f(T_k^{-1}(y)) - z^*) \\ A^k y = 0 \\ y \in S \end{cases} \tag{4.8}$$

In which:

$$A^k = (AX^k \quad - b) \quad \text{and} \quad y = \begin{pmatrix} y[n] \\ y_{n+1} \end{pmatrix}$$

$h : \mathbb{R}^{n+1} \to \mathbb{R}$ is a nonlinear, convex [25] and differentiable function on the set

$$X = \{y \in \mathbb{R}^{n+1};\ A^k y = 0,\ y \in S\}$$

,

in which the optimal objective value of h tends to zero.

4.2.2 The linearization

For solving the problem (4.8) using the Karmarkar's projective method, it must to linearize the convex function h.
By applying the linearization technique to the function h at the neighborhood of the point a which is the center of the simplex S_{n+1}

$$h(y) = h(a) + \nabla^t h(a)(y - a) \tag{4.9}$$

for all $y \in Y$
Where

$$Y = \{y \in \mathbb{R}^{n+1}; ||y - a|| \leq \alpha r\}$$

We have: $0 < \alpha < 1$ and $r = \frac{1}{\sqrt{n(n+1)}}$ is the radius of the inscribed sphere.
Now, we introduce the following problem:

$$\begin{cases} \min_y \nabla^t h(a) y \\ A^k y = 0 \\ \sum_{i=1}^{n} y_i = 1,\ y \geq 0 \end{cases} \tag{4.10}$$

Taking into consideration that we can replace the condition $y \geq 0$ by the sphere $S(a, \alpha r)$, since if we have $y \in S(a, \alpha r)$ then $y \geq 0$ [27].

We can write (4.10) with the following form:

$$\begin{cases} \min_y \nabla^t h(a) y \\ A^k y = 0 \\ \sum_{i=1}^{n+1} y_i = 1 \\ \|y - a\|^2 \leq (\alpha r)^2 \end{cases} \tag{4.11}$$

The optimal solution of (4.11)

Lemma 10.
The optimal solution of the problem (4.11) *is explicitly given by:* $y^k = a - \alpha r d^k$ *and:*

$$x_i^{k+1} = x_i^k \frac{1 - (n+1)\alpha r d_i^k}{1 - (n+1)\alpha r d_{n+1}^k}.$$

In which: $d^k = \frac{s^k}{\|s^k\|}$ *and* $s^k = [I - B_k^t (B_k B_k^t)^{-1} B_k] \nabla h(a)$.

Proof.

We put $z = y - a$, then we can write the problem (4.11) as follows:

$$\begin{cases} \min_x \nabla^t h(a) z \\ B_k z = 0 \\ \|z\|^2 \leq (\alpha r)^2 \end{cases} \tag{4.12}$$

In which: $B_k = \begin{pmatrix} A_k \\ e_{n+1}^t \end{pmatrix}$.

For z^* be an optimal solution of (4.12), there exists $\lambda \in \mathbb{R}^{m+1}$ and $\mu \geq 0$ such that:

$$\nabla h(a) + B_k^t \lambda + \mu z^* = 0 \tag{A}$$

By multiplying the both members of (A) by B_k we obtain

$$B_k \nabla h(a) + B_k B_k^t \lambda + \mu B_k z^* = 0$$

Equivalent to

$$B_k \nabla h(a) + B_k B_k^t \lambda = 0$$

Then

$$\lambda = -(B_k B_k^t)^{-1} B_k \nabla h(a)$$

By substituting in (A):

$$z^* = -\frac{1}{\mu}[I - B_k^t (B_k B_k^t)^{-1} B_k] \nabla h(a) = -\frac{1}{\mu} s^k$$

And as:

$$\| z^* \| = \alpha r$$

We directly deduce that:

$$z^* = -\alpha r d^k \quad \text{where} \quad d^k = \frac{s^k}{\| s^k \|}$$

And

$$y^k = a + z^* = a - \alpha r d^k$$

,

We have $x^{k+1}(\alpha) = T_k^{-1}(y^k) = \frac{X^k y[n]}{y_{n+1}}$
Then

$$\begin{aligned} x_i^{k+1}(\alpha) &= \frac{x_i^k y_i^k}{y_{n+1}^k} \\ &= \frac{x_i^k (a_i - \alpha r d_i^k)}{a_{n+1} - \alpha r d_{n+1}^k} \\ &= \frac{\frac{1}{n+1} x_i^k (1 - (n+1)\alpha r d_i^k)}{\frac{1}{n+1}(1 - (n+1)\alpha r d_{n+1}^k)} \\ &= x_i^k \frac{1 - (n+1)\alpha r d_i^k}{1 - (n+1)\alpha r d_{n+1}^k} \end{aligned}$$

□

4.2.3 The principle of the method

Starting by initial point x^0, the algorithm builds a sequence of interior points that converges to the optimal solution of the problem. In order to reduce the objective value to to zero, we minimize it locally on a sphere inscribed in the feasible orthant. In every iteration k, x^k is reduced to the simplex center by the projective transformation T_k.
In every iteration, we calculate the step size and the descent direction.
We apply the inverse projective transformation T_k^{-1} to return to the initial variable, it is then enough to control the reduction of the objective function f for verifying the convergence of the algorithm proposed.

In the following, we describe how we calculate the optimal step size.

4.2.4 The step size

The line search function is given by:

$$\theta(\alpha) = P\left(x^{k+1}(\alpha)\right) - P(x^k)$$

Where P is the potential function associated to the problem (4.6) which is define by:

$$P(x) = (n+1)\ln\left(f(x) - z^*\right) - \sum_{i=1}^{n+1} \ln(x_i)$$

Lemma 11.
The search line function is given by:

$$\theta(\alpha) = (n+1)\ln\left(\frac{h(a - \alpha r d^k)}{h(a)}\right) - \sum_{i=1}^{n+1} \ln\left(1 - (n+1)\alpha r d_i^k\right)$$

Proof.
We have

$$\theta(\alpha) = (n+1)\ln\left(\frac{f(x^{k+1}) - z^*}{f(x^k) - z^*}\right) - \sum_{i=1}^{n} \ln\left(\frac{x_i^{k+1}}{x_i^k}\right)$$

Then

$$\begin{aligned}\theta(\alpha) &= (n+1)\ln\left(\tfrac{h(y^k)}{h(a)}\right) - \textstyle\sum_{i=1}^{n+1} \ln(y_i^k) \\ &= (n+1)\ln\left(\tfrac{h(a-\alpha r d^k)}{h(a)}\right) - \textstyle\sum_{i=1}^{n+1} \ln(y_i^k)\end{aligned}$$

□

The technique of majorant function

Our approach consist to obtaining the optimal value $\overline{\alpha} > 0$ which gives a significant decreasing of the potential function P associated to the problem (4.6). Since the equation $\theta'(\alpha) = 0$ cannot solved explicitly, to avoid this problem we exploit the idea suggested by J.P. Crouzeix and B. Merikhi [12] and we thought about minimizing the uni-dimensional majorant function of θ which the minimum is calculated explicitly.

In order to calculate the step size we use the technique of majorant function described in chapter 2.

Lemma 12.
The majorant function θ_1 is given by:

$$\theta_1(\alpha) = (n+1)\ln\left(\tfrac{\nabla^t h(a)(a-b)+h(b)-\alpha r\nabla^t h(a)d^k}{h(a)}\right) - n\ln\left(1 + \sqrt{\tfrac{n+1}{n}}\alpha r||d||\right) -$$
$$\ln\left(1 - \sqrt{n(n+1)}\alpha r||d||\right)$$

Proof.

For calculate the majorant function of the line search, we must to majorate the two parts of the line search function.
For the first part, we propose a new approximation of h in neighborhood of a defined by:

$$h(y) = h(a) + \nabla^t h(a)(y - a) + \beta$$

The choice of the parameter β guarantees that the tangent of h at the point a becomes a secant of h at the point $b \in X$.
We obtain β by solving the equation:

$$h(a) + \nabla^t h(a)(b - a) + \beta = h(b)$$

It's clear that:

$$\beta = \nabla^t h(a)(a - b) - h(a) + h(b)$$

Then

$$h(y) = \nabla^t h(a)(y - b) + h(b)$$

Let $y^k = a - \alpha r d^k$ then

$$\ln\left(\frac{h(a - \alpha r d^k)}{h(a)}\right) \leq \ln\left(\frac{\nabla^t h(a)(a - b) + h(b) - \alpha r \nabla^t h(a) d^k}{h(a)}\right)$$

For the second parts, As $y_i = 1 - (n+1)\alpha r d_i^k$.

By using theorem 5 and proposition 1 in chapter two, we can prove that

$$\sum_{i=1}^{n+1} \ln(1 - (n+1)\alpha r d_i^k) \geq n \ln\left(1 + \sqrt{\frac{n+1}{n}}\alpha r ||d||\right) + \ln\left(1 - \sqrt{n(n+1)}\alpha r ||d||\right)$$

□

Lemma 13.

The optimal step size is given by:

$$\overline{\alpha} = \frac{\nabla^t h(a) d^k}{r||d||\left(\sqrt{n(n+1)} - \sqrt{\frac{n+1}{n}}\right)\nabla^t h(a) d^k + (n+1) r ||d||^2 \left(\nabla^t h(a)(a - b) + h(b)\right)}.$$

4.2.5 Convergence

Generally to establish the convergence of the algorithm we use the potential function P which defined previously associated to the problem (4.6).

Lemma 14. *[24]*
We assume that y^k is the optimal solution of the problem (4.11)*, then we have $h(y^k) \leq h(a)$.*

Lemma 15. *[24]*
We assume that y^k is the optimal solution of the problem (4.11)*, then we have $h(y^k) \leq (1 - \frac{\alpha}{n+1})h(a)$.*

Theorem 9. *[24]*
In every iteration of the algorithm the potential function is reduced by a constant value δ such that $P(x^{k+1}) - P(x^k) \leq -\delta$, in which $\delta = -\alpha + \frac{-\alpha^2}{2(1-\alpha)^2}$

4.2.6 Description of the algorithm

Algorithm 4 General algorithm of our approach

$a-$ Initialization:
$\epsilon > 0$, x^0 is a strictly realizable point.

$b-$ Iteration:

- If $f(x^k) - z^* > \epsilon$ do

$$X^k = diag(x^k),$$

$$A_k = (AX^k \quad -b)$$

$$B_k = \begin{pmatrix} A_k \\ e^t_{n+1} \end{pmatrix}$$

$$s^k = (I - B^t_k(B_kB^t_k)^{-1}B_k)\nabla h(a)$$

$$d^k = \frac{s^k}{||s^k||}$$

- Calculate $\overline{\alpha}$.
- $y^k = a - \overline{\alpha} r d^k$.
- $x^{k+1} = T^{-1}_k(y^k) = \frac{X^k y[n]}{y_{n+1}}$.
  Back to b.

End of the algorithm.

4.3 Numerical experiments

In the tables below, we represent some numerical results of two algorithms:

- Method 1: the algorithm define in our approach (the technique of majorant function to calculate the step size).
- Method 2: The classical line search.

Iter represents the number of iterations to obtain x^*, Min represents the minimum and T(s) represents the time in seconds.

Examples

Example with fixed size The two following examples are written as follows:

$$\begin{cases} \min\limits_{x\in\mathbb{R}^n} f(x) \\ Ax = b \\ x \geq 0 \end{cases}$$

And $f^* = f(x^*)$.

Example 8.

$$f(x) = x_1^2 + x_2^2 - 3x_1 - 5x_2$$
$$A = \begin{pmatrix} 1 & 2 & 1 & 0 \\ 3 & 1 & 0 & 1 \end{pmatrix}, \quad b = \begin{pmatrix} 4 \\ 7 \end{pmatrix}$$
$$x^0 = (\frac{9}{5}, \frac{3}{5}, 1, 1)^t$$

$$f(x) = x_1^2 + x_2^2 - 2x_1 - x_2$$
$$A = \begin{pmatrix} 1 & 4 & 1 & 0 \\ 2 & 3 & 0 & 1 \end{pmatrix}, \quad b = \begin{pmatrix} 5 \\ 6 \end{pmatrix}$$
$$x^0 = (1, 1, 0, 1)^t$$

Example with variable size

Example 9.
We consider the following problem:

$$\begin{cases} \min f(x) = \sum_{i=1}^n x_i \ln \frac{x_i}{a_i} \\ x_i + x_{i+m} = b_i, \; i = \overline{1, n}, n = 2m \\ x \geq 0 \end{cases}$$

In which $a_i \in \mathbb{R}^n_+$ and $b_i \in \mathbb{R}$ are fixed, we have tested this examples with different value of n, a_i and b_i.

4.3.1 Tables

Table 4.1: Numerical simulations for example 1 and 2

	Method 1			Method 2		
Example	Min	Iter	T(s)	Min	Iter	T(s)
1)	-7.3528	14	0.0537	-7.1213	32	0.0721
2)	-4.0658	4	0.0101	-4.0509	27	0.0564

Table 4.2: Numerical simulations for example 10 (The case when $a_i = 1, b_i = 6$)

	Method 1			Method 2		
Example	Min	Iter	T(s)	Min	Iter	T(s)
n=50	159.1254	5	0.0302	160.2314	7	0.0412
n=100	390.2658	20	0.1204	393.2514	31	0.2013

Table 4.3: Numerical simulations for example 10 (The case when $a_i = 2, b_i = 4$)

	Method 1			Method 2		
Example	Min	Iter	T(s)	Min	Iter	T(s)
n=50	0.000040	20	2.1	0.000041	33	2.9
n=100	0.000071	22	0.236	0.000081	31	0.4313

4.3.2 Comments

The goal of this study is to solve the nonlinear problem (P) using the potential reduction of Karmarkar based on a majorant function technique to compute the step size rather than the line search method. The numerical tests demonstrate the effectiveness of our technique.
These tests clearly demonstrate the influence of our adjustment on the algorithm's numerical behavior, as shown by a reduction in the number of iterations and a fewer the calculation time.
In table 4.1: We study two quadratic examples with fixed variable, we notice that the time of calculation in method 1 is nearly to the half when compared to method 2, in addition there is a difference between them in the number of iteration.
In table 4.2 and table 4.3: We study a nonlinear example with variable size, we have tested this examples with different size and different value of a_i and b_i, as in table 4.1 we notice that the number of iterations in method 1 is better than 2 and in less execution time.

CONCLUSION

In this thesis, we have been able to develop three variants of interior point methods to solve a nonlinear optimization problem (P). We provided theoretical, algorithmic, and numerical advances in this work, as well as addressed unresolved topics about the solution of nonlinear problem.

- The choice of the penalty term as a vector made it possible to accelerate the convergence.
- The introduction of the majorant function technique associated with the secant to contribute to the minimization of the computational cost of the algorithm.
- Extend the inverse barrier penalty method to the nonlinear case.
- The adaptation of the tangent technique for the step calculation was determined for the robustness of our inverse barrier variant.
- Extend the application of Karmarkar's algorithm to the nonlinear case, using the linearization and translation of the objective.

In all of these works, we calculate the descent direction with the classical Newton method, our study is supported by numerical tests very important.

PERSPECTIVES

The results obtained are very encouraging which motivates us to generalize our work to several optimization problems such as semi definite programming problems, linear complementarity problems,...etc.
Here, we present some perspectives:

- Look for new majorant function in the case where r is taken as a vector.
- In the inverse barrier method using a majorant function technique to calculate the step size.
- Using the minorant function technique to calculate the step size.

BIBLIOGRAPHY

[1] A. Antoniou, Wu-Sheng lu, "PRACTICAL OPTIMIZATION Algorithms and Engineering Applications", Department of Electrical and Computer Engineering, University of Victoria, Canada.

[2] K.Baiche, "Adaptation de la téchnique des fonctions majorantes à la programmation linéaire". Thèse de magistère, Département de mathématiques, Faculté des Sciences, Université Sétif 1, (2012).

[3] M. S. Bazzara, J. Jarvis, and H. D. Sherali, "Linear Programming and Network Flows", John Wiley Sons, New York, NY, USA, (1984).

[4] D. Benterki, M. Bouafia, "Improving complexity of Karmarkar's approach for linear programming", Journal of Numerical Analysis and Approximation Theory, 43(2),(2015), 159-167.

[5] M. Bierlaire, "Introduction à l'optimisation différentiable", press polytechniques et universitaires romandes, (2006).

[6] M. Bouafia, "Étude asymptotique des méthodes de points intérieurs pour la programmation linéaire", Thèse de Doctorat, Université du Havre, France, (2016).

[7] S. Boyd and L. Vandenberghe, "Convex Optimization", Cambridge University Press, (2004).

[8] C. W. Carroll, "The Created Response Surface Technique for Optimizing Nonlinear Restrainecl Systems", Opérations Research, 9, (1961), 169-184.

[9] S. Chaghoub, D. Benterki, "A Logarithmic Barrier Method Based On a New Majorant Function for Convex Quadratic Programming", IAENG International Journal of Applied Mathematics, 51, 2021.

[10] L.B. Cherif, B. Merikhi, "A penalty method for nonlinear programming", RAIRO Oper Res, 53, (2019), 29-38.

[11] J.P. Crouzeix, A. Keraghel. W.S. Sandoval. "Programmation Mathématique Différentiable".

[12] J.P. Crouzeix, B. Merikhi, "A logarithm barrier method for semidefinite programming", RAIRO Oper Res, 42, (2008), 123-139.

[13] J.P. Crouzeix and A. Seegerm, "New bounds for the extreme values of a finite sample of real numbers", J. Math. Anal. Appl, 197, (1996), 411-426.

[14] G. B. Dantzig, "Maximization of a linear function of variables subject to linear inequalities", In Tj.C. Koopmans, éditeur, Activity Analysis of Production and Allocation, 339-347, Wiley, NewYork, (1951), 554-567.

[15] D. En, C. Roos and T. Terklay, "Inverse barrier methods for linear programming", Oper Res, 28 , 2, (1994), 135-163.

[16] P. Fei and Y. Wang, "A primal infeasible interior point algorithm for linearly constrained convex programming", Control and Cybernetics, 38, (2009), 687–704.

[17] B. Fellahi, B. Merikhi, A Logarithmic Barrier Approach Via Majorant Function for Nonlinear Programming, J. Sib. Fed. Univ. Math. Phys., 2023, 16(4), 528–539.

[18] A. V. Fiacco, G. P. Mccormick, "Nonlinear Programming : Sequential Unconstrained Minimization Techniques", Wiley, New York, (1968).

[19] K.R. Frisch, "The logarithmic potential method of convex programming", Technical report, University Institute of Economics, Oslo, Norway, (1955).

[20] P. Huard, "Resolution of mathematical programming with nonlinear constraints by the method of centers", In J. Abadie, editor, Nonlinear Programming, North Holland, Amsterdam, The Netherlands, (1967), 207-219.

[21] N.K. Karmarkar, "A new polynomial-time algorithm for linear programming", Proc. of the 16th Annual ACM Symposium on Theory of Computing, 4, (1984), 373–395.

[22] R.M.R. Karp, "George Dantzig's impact on the theory of computation", Discrete Optimization. 5(2), (2008), 174–185.

[23] L. G. Khachiyan, "A polynomial algorithm in linear programming", Soviet Mathematics Doklady, 20, (1979), 191-194.

[24] Z. Kebiche, "Étude et extensions d'algorithmes de points intérieurs pour la programmation non linéaire", Université Ferhat Abbas, Sétif (2007).

[25] Z. Kebbiche, A. Keraghel and A. Yassine, "Extension of a projective interior point method for linearly constrained convex programming", Applied mathematics and computation, 193(2), (2007), 553-559.

[26] A. Keraghel, "Analyse convexe : théorie fondamentale et exercices", Editions Dar el'Houda, Ain M'lila, Algérie, (1999).

[27] A. Keraghel, " Etude adaptative et comparative des principales variantes dans l'algorithme de Karmarkar", Thèse de Doctorat, Université Joseph Fourier Grenoble I, France, (1989).

[28] J. Kowalik, "Nonlinear Programming Procedures and Design Optimization", Acta Polyntech, Scand, Trondheim, (1966), 13.

[29] S. Lamri, "Méthode de Point Intérieur de Type Barrière Logarithmique pour la Résolution d'un Problème d'optimisation convexe", Thèse de Doctorat, Institut de Mathématiques, Université Ferhat Abbas, Sétif, Algérie, (2021).

[30] A. Leulmi , B. Merikhi , D. Benterki, "Study of a logarithmic barrier approach for linear semi definite programming", Journal of the Siberian Federal University, Mathematics and Physics, 11, no 3, (2018), 300-312, .

[31] I.J. Lustig, "A pratical approach to Karmarkar's algorithm", Technical report sol 85-5 Systems optimizatioin laboratory ; dep of operations res. STANFORD, univ Stanford California, 9, (1985).

[32] L. Menniche, D. Benterki, "A logarithmic barrier approach for linear programming", Journal of Computational and Applied Mathematics, Elsevier, 312, (2017), 267-275.

[33] B. Merikhi, "Extension de quelques méthodes de points intérieurs pour la programmation semi-définie", Thèse de Doctorat, Institut de Mathématiques, Université Ferhat Abbas, Sétif (2006).

[34] J. Nocedal, S.J. Wright. "Numerical Optimization", Springer Series in Operations Research and Financial Engineering.

[35] M. Ouriemchi, "Résolution de problèmes non linéaires par les méthodes de points intèrieurs, Théorie et algorithmes", Thèse de doctorat, Univerité du Havre, France, (2006).

[36] S.S. Rao. "Engineering Optimization Theory and Practice", JOHN WILEY and SONS, INC.

[37] N. Z. Shor, "Utilization of the operation of space dilatation in the minimization of convex functions", Kibernetika, 1, (1970), 6-12.

[38] S. Tahmasebzadeh, H.Navidi and A.Malek, "Novel interior point algoritms for solving nonlinear convex optimization problems", Hindawi Publishing Corporation, Advances in Operaions Research, (2005), 1-7.

[39] E. Tse and Y. Ye, "An extension of Karmarkar's projective algorithm for convex quadratic programming", Mathematical Programming, 44, (1989), 157–179.

[40] H. Wolkowicz and G.P.H. Styan, "Bounds for eigenvalues using traces", Linear Algebra Appl, 29, (1980), 471-506.

[41] Y. Ye, E. Tse, "An extension of Karmarkar's projective algorithm for convex quadratic programming", Mathematical Programming, 4 (1989), 157-179.

Title

Logarithmic barrier and inverse barrier interior point methods in nonlinear programming

Abstract

In this thesis, we are interested in the theoretical and numerical study of some interior point methods in convex optimization problems.

In the first, we propose a logarithmic barrier approach in which the penalty term is taken as a vector, followed by a convergence study where the step size is determined using a majorant function technique.

In addition, we extend an inverse barrier in the nonlinear case, the step size is determined with a tangent technique.

We finish this work by an extension of Karmarkar's algorithm in nonlinear case, and this by using the linearization and translation of objective function.
In all of these work, the descent direction is calculated with the classical Newton method.
This study is supported by numerical tests which show the effectiveness of these approaches.

Key words

Convex programming, Interior point method, Logarithmic barrier, Inverse barrier, Potential function, Line search, Majorant function, Tangent technique.

Mathematics Subject Classification

90C25, 90C30, 90C51, 49N15, 49M15, 65K05, 49M37.

Titre

Méthodes de point intérieur de type barrière logarithmique et barrière inverse en programmation non linéaire

Résumé

Dans cette thèse, nous nous intéressons à l'étude théorique et numérique de quelque méthodes des points intérieurs en optimisation convexe.

En premier, nous proposons une approche barrière logarithmique dans laquelle le terme de pénalité est pris comme un vecteur, suivie d'une étude de convergence où le pas de déplacement est déterminé par une technique de fonction majorante.

En outre, on étend l'approche barrière inverse au cas non linéaire, le pas de déplacement de cette dernière est déterminé à l'aide d'une technique de la tangente.

On termine ce travail par une extension de l'algorithme de Karmarkar au cas non linéaire, et ce, en utilisant la linéarisation et la translation de l'objectif.

Dans tous ces travaux, la direction de descente est calculée avec la méthode de Newton.

Cette étude est soutenue par des tests numériques qui montrent l'efficacité de ces approches.

Mots clés

Programmation convexe, Méthode de point intérieur, Barrière logarithmique, Barrière inverse, Fonction potentiel, Recherche linéaire, Fonctions majorantes, technique de la tangente.

Mathematics Subject Classification

90C25, 90C30, 90C51, 49N15, 49M15, 65K05, 49M37.

العنوان

طرق النقاط الداخلية للحاجز اللوغاريتمي والحاجز العكسي في البرمجة غير الخطية

ملخص

في هذه الأطروحة، نهتم بالدراسة النظرية والعددية لبعض طرق النقاط الداخلية في التحسين المحدب.

أولاً ، نقترح نهج الحاجز اللوغاريتمي الذي يتم فيه تمثيل عامل الحاجز كشعاع، متبوعاً بدراسة التقارب حيث يتم فيها تحديد خطوة الإنتقال باستخدام تقنية دالة الحد العلوي.
بالإضافة إلى ذلك ، نقترح تمديد نهج الحاجز العكسي إلى الحالة غير الخطية ، ويتم تحديد خطوة الإنتقال للأخير باستخدام تقنية المماس.
ننهي هذا العمل بدراسة امتداد خوارزمية كارماركار للحالة غير الخطية.
في كل هذه الأعمال ، يتم حساب اتجاه الهبوط بطريقة نيوتن.

هذه الدراسة مدعومة باختبارات عددية تظهر فعالية هذه الأساليب.

الكلمات المفتاحية

البرمجة المحدبة ، طريقة النقطة الداخلية، الحاجز اللوغاريتمي، الحاجز العكسي، دالة الكمون، البحث الخطي ، دالة الحد العلوي، تقنية المماس.

تصنيف مواضيع الرياضيات

90C25 , 90C30 , 90C51 , 49N15 ,49M15 , 65K05 , 49M37.