University of Windsor
Scholarship at UWindsor

Electronic Theses and Dissertations

Theses, Dissertations, and Major Papers

2010

On Murty's gravitational interior point method for quadratic programming

Pooyan Shirvani Ghomi
University of Windsor

On Murty's Gravitational Interior Point Method for Quadratic Programming

by

Pooyan Shirvani Ghomi

A Thesis

Submitted to the Faculty of Graduate Studies

through the Department of Mathematics and Statistics

in Partial Fulfillment for the Requirements of

the Degree of Master of Science at the

University of Windsor

Windsor, Ontario, Canada

2010

Your file Votre référence
ISBN: 978-0-494-62731-0
Our file Notre référence
ISBN: 978-0-494-62731-0

Author's Declaration of Originality

I hereby certify that I am the sole author of this thesis and that no part of this thesis has been published or submitted for publication.

I certify that, to the best of my knowledge, my thesis does not infringe upon anyone's copyright nor violate any proprietary rights and that any ideas, techniques, quotations, or any other material from the work of other people included in my thesis, published or otherwise, are fully acknowledged in accordance with the standard referencing practices. Furthermore, to the extent that I have included copyrighted material that surpasses the bounds of fair dealing within the meaning of the Canada Copyright Act, I certify that I have obtained a written permission from the copyright owner(s) to include such materials in my thesis and have included copies of such copyright clearances to my appendix.

I declare that this is a true copy of my thesis, including any final revisions, as approved by my committee and the Graduate Studies office, and that this thesis has not been submitted for a higher degree to any other University or Institution.

Abstract

This thesis presents a modification of the gravitational interior point method for quadratic programming [**7**]. Murty presented the algorithm as a generalization of his gravitational method for linear programming [**8**]. Murty claims that this method is matrix inverse free unlike other interior point methods, however convergence of his algorithm is not guaranteed. This thesis introduces modifications in the centering step of the algorithm and, using a MatlabR2009a implementation, demonstrates the centering step.

Dedication

This thesis is dedicated to all Iranian students in jail, who have devoted their lives to fighting for justice and freedom. Words alone cannot express the regret I have that I'm not beside them. I hope to see all of them free and happy in a "Free Iran".

Acknowledgements

First and foremost I would like to express my most sincere thanks to my supervisor, Dr. Richard J. Caron, for his guidance throughout the course of this study. I am grateful to Dr. Caron for his invaluable help which made this thesis possible and all of his supports for in helping earn my Master's degree. I would like to acknowledge my internal reader Dr. Tim Traynor and my external reader Dr. Yash P. Aneja.

I would like to give thanks to all the nice people I have met in the past two years at Department of Mathematics and Statistics, University of Windsor.

I would like to acknowledge my sisters Pardis and Parmis. Last, but not least I would like to thank my parents, Hossain and Azadeh, for their constant love, support and encouragement.

Contents

List of Figures

List of Tables

CHAPTER 1

Introduction

1.1. Overview and Outline of Thesis

In this thesis we introduce a new inverse free interior point algorithm for solving the convex quadratic programming problem (QP)

$$\begin{array}{llll} \textit{minimize} & Q(x) = c^\top x + \frac{1}{2}x^\top \mathrm{H}x & & \\ \textit{subject to} & \quad a_i^\top x \ \leq \ b_i & \quad \textit{for } i = 1, \ldots, m, \end{array} \tag{1.1}$$

where $x = (x_1, x_2, ..., x_n)^\top$ is the column vector of variables, H is a symmetric n by n positive definite matrix, $c, a_1, \ldots, a_m$ are vectors of order n, and $b_1, \ldots, b_m$ are scalars. The vector a_i is the gradient of the i^{th} constraint function $a_i^\top x$. Without loss of generality, we assume that for all i the a_i have been normalized so that $\|a_i\| = 1$, where $\|\cdot\|$ is the Euclidean norm. We use $\mathcal{I} = \{1, 2, \ldots, m\}$ to index the constraint set for (1.1).

The *feasible region* for the QP is denoted by $\mathcal{R}$ and is the set of all points x in $\mathbb{R}^n$ that satisfy the constraints $a_i^\top x \leq b_i$ for all $i \in \mathcal{I}$. For simplicity we assume that $\mathcal{R}$ is bounded.

The point $\hat{x}$ is feasible for (1.1) if $\hat{x} \in \mathcal{R}$ and infeasible otherwise. We say constraint i is *active*, *inactive* or *violated* if $a_i^\top \hat{x} = b_i$, $a_i^\top \hat{x} \leq b_i$ or $a_i^\top \hat{x} > b_i$, respectively. For any $\hat{x} \in \mathcal{R}$, we define the set

$$\mathcal{J}(\hat{x}) = \{i : a_i^\top \hat{x} = b_i, i \in \mathcal{I}\};$$

that indexes the set of all constraints active at $\hat{x}$. We assume that the feasible region $\mathcal{R}$ has a non-empty interior $\mathcal{R}_0 = \{\, x \in \mathbb{R}^n \mid a_i^\top x < b_i, \ \forall i \in \mathcal{I} \,\}$. That is, we assume that there is a $\hat{x} \in \mathcal{R}$ with $\mathcal{J}(\hat{x}) = \emptyset$.

Each iteration of our algorithm contains two steps. The first is a centering step and it is followed by a descent step. In the centering step, we determine a feasible solution that has the largest minimum distance to all the constraint boundaries and that has an objective value less than or equal to that of the current interior feasible solution. Let $B(x,r) = \{y \in \mathbb{R}^n | (y-x)^\top (x-y) \leq r^2\}$ be the ball with center x and radius r; and suppose that $\hat{x}$ is the current interior feasible solution. In the centering step we are looking for a point $\tilde{x}$ that will maximize r with $B(\tilde{x}, r) \in \mathcal{R}$ and $Q(\tilde{x}) \leq Q(\hat{x})$. Let $\tilde{r}$ be the radius of the ball corresponding to the centre $\tilde{x}$.

In the descent step we look for the minimum of the quadratic function subject to the ball constraint. That is, we solve min $\{\, Q(x) \,|\, x \in B(\tilde{x}, \tilde{r})\}$.

The concept for this kind of algorithm was first introduced by Murty in 2006 [**8**] for linear programming and then he extended it for convex quadratic programming [**7**]. The algorithm presented in this thesis overcomes some of the difficulties in Murty's algorithm, yet it maintains the positive aspects, such as the avoidance of matrix inverse operations, the major cost factor for standard interior point methods.

In chapter 2, we present Murty's algorithm for linear programming. Chapter 3 is devoted to Murty's QP algorithm and it is here that we demonstrate its weaknesses and suggest improvements. In chapter 4 we introduce our procedure for the centering step and present our experimental results. First, we present background information on quadratic programming.

1.2. The Quadratic Programming Problem (QP)

In this section we present material related to the QP (1.1). The gradient of $Q(x)$ at the point x is the vector of order n given by

$$\nabla Q(x) = \begin{bmatrix} \frac{\partial Q(x)}{\partial x_1} \\ \frac{\partial Q(x)}{\partial x_2} \\ \vdots \\ \frac{\partial Q(x)}{\partial x_n} \end{bmatrix} = c + Hx.$$

The *Hessian matrix* of $Q(x)$ is the $n \times n$ matrix whose $(i,j)^{th}$ entry is

$$\frac{\partial^2 Q(x)}{\partial x_i x_j}.$$

It follows that the *Hessian matrix* of $Q(x)$ is H. In this thesis, we assume that H is positive definite.

DEFINITION 1.2.1 (Positive Definite and Positive Semi-Definite). *Let* $\mathrm{H} \in \mathbb{R}^{n\times n}$ *be symmetric.* H *is said to be positive definite if* $x^\top \mathrm{H} x > 0$ *for all* $x \in \mathbb{R}^n \setminus \{0\}$. *Similarly,* H *is said to be positive semi-definite if* $x^\top \mathrm{H} x \geq 0$ *for all* $x \in \mathbb{R}^n$.

DEFINITION 1.2.2 (Convex, Strictly Convex and Concave Functions [1]).

i) *The function f is convex if and only if for any two points x, y in the domain of f and $\lambda \in [0,1]$*

$$f(\lambda x + (1-\lambda)\, y) \leq \lambda f(x) + (1-\lambda)\, f(y).$$

[1]Some books refer to the inequalities as Jensen's inequalities

ii) *The function f is strictly convex if and only if for any two distinct points x, y in the domain of f and $\lambda \in [0, 1]$*

$$f(\lambda x + (1 - \lambda)\, y) < \lambda f(x) + (1 - \lambda)\, f(y).$$

iii) *The function f is concave if and only if for any two points x, y in the domain of f and $\lambda \in [0, 1]$*

$$f(\lambda x + (1 - \lambda)\, y) \geq \lambda f(x) + (1 - \lambda)\, f(y).$$

LEMMA 1.2.3. *If the function $f(x)$ is convex and differentiable, then $f(x) \geq f(x_0) + \nabla f(x_0)(x - x_0)$*

PROOF. Since $f(x)$ is convex for any two points $x \neq y$ and $\lambda \in (0, 1)$

$$f(\lambda x + (1 - \lambda)\, y) - f(y) \leq \lambda f(x) - \lambda f(y).$$

Divide both sides by λ to get

$$\frac{f(\lambda x + (1 - \lambda)\, y) - f(y)}{\lambda} \leq f(x) - f(y)$$

Now let $\lambda \to 0$ then the definition left hand side is the gradient of $f(x)$ in the direction $(x - y)$ at the point y and we have

$$\nabla f(y)^{\top}(x - y) + f(y) \leq f(x). \tag{1.2}$$

$\square$

The next theorem provide conditions that help us determine whether or not a quadratic function is convex.

THEOREM 1.2.4. *The Taylor's series for a quadratic function $Q(x)$ about x_0 is*

$$Q(x) = Q(x_0) + \nabla Q(x_0)^{\top}(x - x_0) + \frac{1}{2}(x - x_0)^{\top}\mathrm{H}(x - x_0). \tag{1.3}$$

Furthermore, $Q(x)$ is differentiable and

$$\nabla Q(x) = \nabla Q(x_0) + \mathrm{H}(x - x_0). \tag{1.4}$$

THEOREM 1.2.5. *The function $Q(x) = c^{\top}x + \frac{1}{2}x^{\top}\mathrm{H}x$ is strictly convex if and only if H is positive definite.*

PROOF. ($\longrightarrow$) Let $Q(x)$ be a strictly convex function, then the inequality $Q(\lambda x + (1-\lambda)y) < \lambda Q(x) + (1-\lambda)Q(y)$ holds for all $x \neq y$ and λ with $0 < \lambda < 1$. Let $s \neq 0$ be a vector of order n and x be arbitrary, and λ be such that $0 < \lambda < 1$. Replacing y with $x + s$ in the above inequality gives

$$Q(x + \lambda s) < \lambda Q(x + s) + (1 - \lambda)Q(x).$$

Now, expanding both $Q(x + s)$ and $Q(x + \lambda s)$ using Taylor's series we have

$$Q(x) + \lambda \nabla Q(x)^{\top}s + \frac{1}{2}\lambda^2 s^{\top}\mathrm{H}s < \lambda Q(x) + \lambda \nabla Q(x)^{\top}s + \frac{\lambda}{2}s^{\top}\mathrm{H}s + (1 - \lambda)Q(x).$$

Simplifying, we have

$$(1 - \lambda)\lambda s^{\top}\mathrm{H}s > 0$$

but $(1-\lambda)\lambda$ is always bigger than 0 so that

$$s^\top \mathrm{H} s > 0.$$

Since s was arbitrary, H is positive definite.

($\longleftarrow$) Suppose H is positive definite. Let $z \neq y$ be any two points and λ that $0 < \lambda < 1$. For simplicity, define $\omega = \lambda z + (1-\lambda)y$. Now consider Taylor's series (1.2.4). Since H is positive definite, $(z-\omega)^\top \mathrm{H}(z-\omega)$ is always positive, therefore the following inequality holds

$$Q(z) > Q(\omega) + \nabla Q(\omega)^\top (z-\omega). \tag{1.5}$$

This inequality also holds when z is replaced by y, so we have

$$Q(y) > Q(\omega) + \nabla Q(\omega)^\top (y-\omega). \tag{1.6}$$

Multiplying (1.5) by λ, (1.6) by $(1-\lambda)$ and adding together, gives

$$\lambda Q(z) + (1-\lambda)Q(y) > (\lambda + 1 - \lambda)Q(\omega) + \nabla Q(\omega)^\top (\lambda z + (1-\lambda)y - \omega).$$

After simplifying, we have

$$\lambda Q(z) + (1-\lambda)Q(y) > Q(\lambda z + (1-\lambda)y),$$

as required. $\square$

1.3. Optimality Conditions

We consider the model problem (1.1). A more compact form is

$$\textit{minimize}\,\{Q(x) \mid Ax \leq b\}\,, \tag{1.7}$$

where $A^\top = [a_1, \ldots, a_m]$ and $b = [b_1, \ldots, b_m]^\top$. Then the feasible region, i.e. $\mathcal{R}$, for (1.7) is

$$\mathcal{R} = \{x \mid Ax \leq b\}\,.$$

The point x^* is an optimal solution (simply optimal) if $x^* \in \mathcal{R}$ and $Q(x^*) \leq Q(x)$ for all $x \in \mathcal{R}$. The objective function for (1.7) is unbounded from below if there exists a point x_0 and direction s_0 such that $x_0 - \delta s_0 \in \mathcal{R}$ for all $\delta \geq 0$ and $Q(x_0 - \delta s_0) \to -\infty$ when $\delta \to \infty$.

Definition 1.3.1 (Descent Direction). *The direction s is said to be a descent direction for $Q(x)$ at the point x if $Q(x - \alpha s) < Q(x)$ for all α where $0 < \alpha \leq \epsilon$ for some $\epsilon > 0$.*

Lemma 1.3.2. *The direction $-s$ is a descent direction for $Q(x)$ at the point x if $\nabla Q(x)^\top s > 0$.*

Definition 1.3.3 (Optimal Step Size). *The value of α that minimizes $Q(x - \alpha s)$, $\alpha \in \mathbb{R}$, is called the optimal step size and it is given by*

$$\tilde{\alpha} = \tfrac{\nabla Q(x)^\top s}{s^\top \mathrm{H} s}, \quad s^\top \mathrm{H} s \neq 0. \tag{1.8}$$

If $s^\top \mathrm{H} s = 0$, then $Q(x)$ is unbounded from below.

Definition 1.3.4 (Extreme Point). *For a given convex set S, x is said to be an extreme point of S if it is not possible to represent x as linear combination of any other two distinct points of S.*

Definition 1.3.5 (Maximum feasible step size). *Let $x \in \mathcal{R}$ and $s \in \mathbb{R}^n$, the maximum value of $\alpha \geq 0$ with $x - \alpha s \in \mathcal{R}$ is called the maximum feasible step size, and it denoted by $\hat{\alpha}$.*

Lemma 1.3.6. *If $a_i^\top s \geq 0$ for all $i = 1, \ldots, m$, the maximum feasible step size is taken as $+\infty$, otherwise*

$$\hat{\alpha} = \min \left\{ \frac{a_i^\top x - b_i}{a_i^\top s} \mid \quad i = 1, \ldots, m, \text{ and } a_i^\top s < 0 \right\}.$$

Most linear programming algorithms, except interior point methods, are based on the fact that the feasible region possesses a finite number of extreme points and at least one of these extreme points is an optimal solution. In quadratic programming, the analog to an extreme point is a quasistationary point.

Definition 1.3.7 (Quasistationary point). *The point $x_0 \in \mathcal{R}$ is a quasistationary point for (1.7) if x_0 is an optimal solution for*

$$\text{minimize} \left\{ Q(x) \mid a_i^\top x = b_i, i \in \mathcal{J}(x_0) \right\}. \tag{1.9}$$

It is obvious that every extreme point of $\mathcal{R}$ is a quasistationary point for (1.7) and that any optimal solution for (1.7) is a quasistationary point since it is a strictly convex QP. In general, (1.7) possesses many quasistationary points. For example if there is a solution $x_0^* \in \mathcal{R}$ to

$$minimize\left\{c^\top x + x^\top \mathrm{H}x | a_i^\top x = b_i, i \in \mathcal{I}_0\right\}, \tag{1.10}$$

for some subset $\mathcal{I}_0$ of $\mathcal{I} = \{1, \ldots, m\}$, then x_0^* is a quasistationary point. It is theoretically possible to find all quasistationary point and solve (1.7), but there are 2^m such subsets, so a rather large amount of computation is required. Many algorithms iteratively determine a sequence of quasistationary point $x_1, \ldots, x_{j-1}, x_j$ with

$$Q(x_j) < Q(x_{j-1}) < \ \ldots \ < Q(x_1),$$

and locate an optimal solution for (1.7). The fact that a finite number of quasistationary exists, implies that in a finite number of iterations, an optimal solution will be found. As we know, Beale was the first one that used this argument to show finite termination. Best in [**4**] shows under some assumption there are many QP methods that produce the same sequence of quasistationary points.

A quadratic programming problem (1.1) is bounded from below if there is a number γ, such that for all $x \in \mathcal{R}$,$Q(x) \geq \gamma$. In the other words there is not a direction s with $Q(x - \alpha s) \to -\infty$ when $\alpha \to \infty$.

LEMMA 1.3.8. *Suppose that $Q(x)$ is convex and it is bounded from below on $\mathcal{R}$. Let x_0 be an arbitrary point in $\mathcal{R}$. Then there is a quasistationary point $\tilde{x}$ with $Q(\tilde{x}) \leq Q(x_0)$.*

THEOREM 1.3.9 (Existence of an optimal solution for quadratic programming problem). *If $Q(x)$ be bounded from below on $\mathcal{R}$, then there exist an optimal solution*

$\hat{x}$ *for*

$$\textit{minimize}\{\ Q(x) \mid Ax \leq b\}$$

and $\hat{x}$ *is a quasistationary point.*

PROOF. There are finitely many quasistationary points. Each is associated with a subset of $\{1, \ldots, m\}$. Suppose that Let y is a quasistationary point. Associated with this y is the quasistationary set

$$S(y) := \{x \mid a_i^\top x = b_i, \textit{for all } i \in \mathcal{J}(y), Q(x) = Q(y)\}.$$

Since there are finitely many quasistationary points, there are a finite number, say p, of quasistationary sets. Let x_i be a quasistationary point from i^{th} quasistationary the set. We choose $\hat{x}$ so that

$$Q(\hat{x}) = \min\ \{Q(x_i) \mid i = 1, \ldots, p\}.$$

Suppose $x \in \mathcal{R}$. From Lemma (1.3.8) we know that there exists a quasistationary point $\tilde{x}$ with $Q(\tilde{x}) \leq Q(x)$. But $\tilde{x} \in S(x_i)$ for some i where $1 \leq i \leq p$. So $Q(x_i) = Q(\tilde{x})$.But

$$Q(\hat{x}) \leq Q(x_i) = Q(\tilde{x}) \leq Q(x).$$

So $Q(\hat{x}) \leq Q(x)$ for all $x \in \mathcal{R}$ and $\hat{x}$ is an optimal solution. □

THEOREM 1.3.10 (Optimality condition for quadratic programming problem). *The point*x_0 *is an optimal solution for (1.1) if and only if there exist scalar* $u_1, \ldots, u_m$ *which together with* x_0 *satisfy*

(1) $a_i^\top x_o \leq b_i$ *for all* $i = 1, \ldots, m$

(2) $-\nabla Q(x_0) = u_1 a_1 + \ldots + u_m a_m, \quad u_i \geq 0, i = 1, \ldots, m,$

(3) $u_i(a_i^\top x_0 - b_i) = 0, i = 1, \ldots, m.$

PROOF. See [**6**] page 68-69. □

These are the KKT conditions for quadratic programming.

1.4. Concluding Remarks

We established basic theorems of quadratic programming. Now we can discuss algorithms. The next chapter is an overview of spherical method that Murty proposed for linear programming. It introduces the concepts behind this new method.

CHAPTER 2

Spherical Method for LP

2.1. Introduction

About 20 years ago, Chang and Murty in [**12**] developed new methods for Linear Programming(LP), but in [**13**] Morin, Parbhu and Zhang showed that this algorithm has worst case exponential growth as dose the simplex method. Murty, in [**8**], developed the new Spherical method, which it based on a "gravitational model". This new method can be classified as an Interior Point Method (IPM). In the next section we explain the concept behind this method and we try to clarify steps of this method to gain insight about the challenges this method faces when it is adopted for QP.

2.2. Spherical Method concept

Consider the LP in the following form

$$\begin{aligned} &\mathit{maximize} \quad c^\top x \\ &\mathit{subject\ to} \quad a_i^\top x \leq b_i \quad \mathit{for}\, i \in \{1, \dots, m\}. \end{aligned} \tag{2.1}$$

Suppose that we are given an $x_0 \in \mathcal{R}_0$, so that there exists a ball with x_0 as center and radius r_0 that is completely contained in $\mathcal{R}$. The gravitational method, traces the path of the center as the ball drops under the gravitational force pulling it in the direction $-c^\top$. After some initial descent, the ball will be blocked by a facet of $\mathcal{R}$. After that the ball starts to move along the facets of $\mathcal{R}$. So the center of ball will stay close ,within r_0, the boundary and it is expected the gravitational method will behave like boundary or active set methods. One way to improve the efficiency of gravitational methods is to keep the center of ball far away from boundary of feasible region. Therefore, we must try to maximize the radius of the ball. A benefit of this strategy is that you can move inside the ball without any concern about violating constraints or of getting stuck in corners. This improvement leads to the spherical method,

Before we start a description of the algorithm we need some preliminary definitions. Suppose point $x \in \mathcal{R}_0$, since $\|a_i\| = 1$, $b_i - a_i^\top x$ is Euclidian distance of the point x to the boundary of constraint i. Now, let $\delta(x) = \min \{b_i - a_i^\top x \mid i \in \mathcal{I}\}$, then the biggest ball with x as center that can be inscribed inside the feasible region has radius $\delta(x)$. This ball is denoted by $B(x, \delta(x))$ and is defined as

$$B(x, \delta(x)) := \{y \mid (y - x)^\top (y - x) \leq \delta(x)^2\}.$$

Some of the constraint boundaries of the feasible region are tangent to $B(x)$. The set of such constraints is

$$T(x) := \{i \mid \delta(x) = b_i - a_i^\top x, \quad \text{for } i \in \mathcal{I}\}.$$

To determine the biggest ball that can be inscribed within the feasible region we should maximize $\delta(x)$, or, in other words, *maximize* $\{\min\{b_i - a_i^\top x \mid \forall\ i \in \mathcal{I}\}\ \}$. This is a min-max problems and we can rewrite it as

$$\begin{array}{lrcl} \textit{maximize} & \delta & & \\ \textit{subject to} & \delta + a_i^\top x & \leq & b_i \quad \textit{for } i \in \mathcal{I}, \end{array} \tag{2.2}$$

where (x, δ) is the vector of variables. But we want the objective function value not increase. Suppose y is provided as the initial point for iteration j of the algorithm, then all points to be considered for center of the ball must satisfy $c^\top x \leq c^\top y$. So, in each iteration we want to the determine biggest ball that can be inscribed inside the feasible and that has a center with an improved objective function value. We can achieve this by solving

$$\begin{aligned} &\textit{maximize} && \delta \\ &\textit{subject to} && \delta + a_i^\top x \;\leq\; b_i \quad \textit{for } i \in \mathcal{I}, \\ & && c^\top x \leq c^\top y. \end{aligned} \tag{2.3}$$

The only difference between (2.2) and (2.3) is the constraint $c^\top x \leq c^\top y$. In figure (2.1) we can see the biggest ball that can be inscribed in feasible region of the problem (2.3).

Each iteration of the spherical method then consists of two main steps, the centering step and the descent step. We begin with an interior point. The centering step is to solve (2.3). The descent step moves from the center to a point with smaller objective value.

2.2.1. The Centering Step. It is a good question to ask how we should determine the solution of (2.3). Since (2.3) is an LP, we could use methods like simplex method or interior point methods. This is quite counter productive since (2.3) must be solved several times in order to determine the optimal solution of (2.1). The spherical method will be practical only if there is a computationally inexpensive procedure that can carry out the centering step, i.e. solve (2.3). Murty approach is to solve

$$
\begin{aligned}
\textit{maximize} \quad & \delta \\
\textit{subject to} \quad & \delta + a_i^\top x \le b_i \quad \textit{for}\, i \in \mathcal{I}, \\
& c^\top x = c^\top y.
\end{aligned}
\tag{2.4}
$$

The difference is that $c^\top x \le c^\top y$ is replaced with $c^\top x = c^\top y$. Murty in [**8**] proposed a procedure to get an approximation of the optimal solution of (2.4) and he claimed that it was able to determine a good enough approximation of centering step. However, his proof of convergence depended on of finding the exact solution. Now we describe

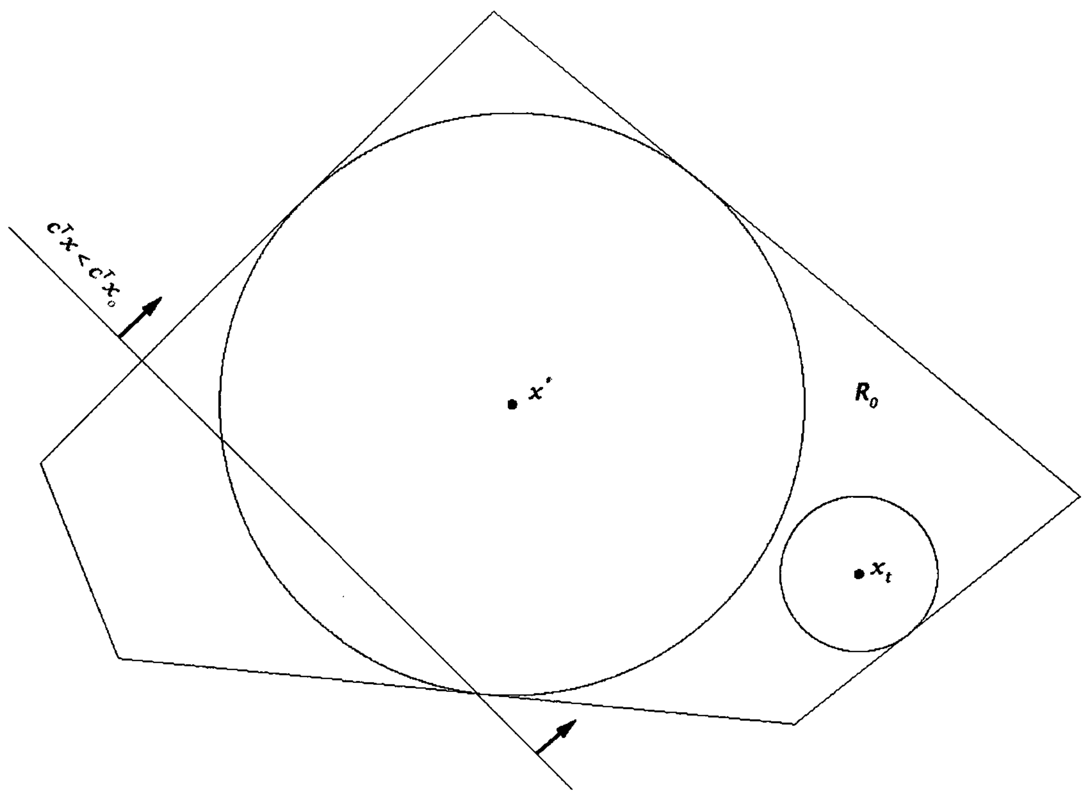

FIGURE 2.1. The centering step for LP.

Murty's procedure. Suppose an interior initial point x_0 is provided. the direction s is what Murty calls a *profitable direction* at x if there exists $\alpha > 0$ with $\delta(x - \alpha s) > \delta(x)$. If x is an interior point, then $T(x)$ is non-empty and $|T(x)| \geq 1$. In figure (2.2) we can see the optimal solution to (2.4).

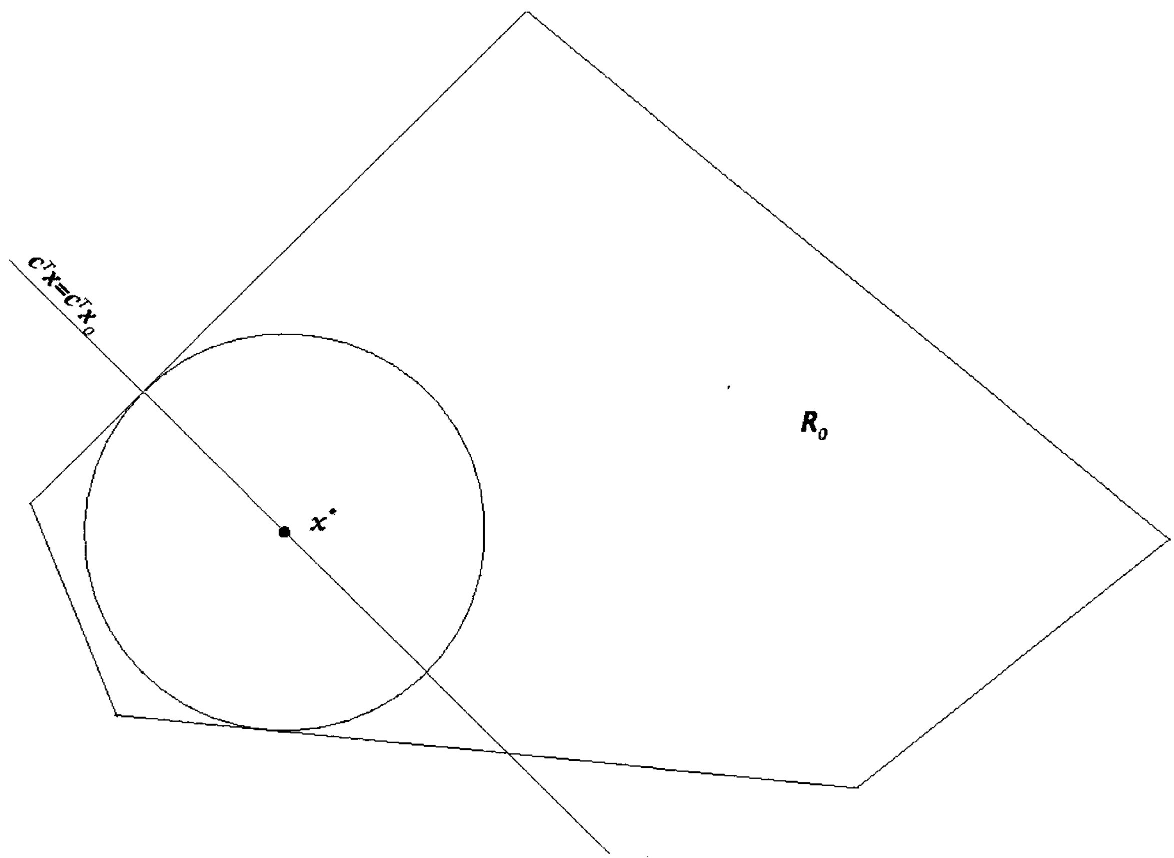

FIGURE 2.2. The modified centering step for LP.

THEOREM 2.2.1. *The direction s is a profitable direction at the point x if and only if* $\min\{a_i^\top s \mid i \in T(x)\} > 0$.

PROOF. ($\longrightarrow$) Let s be a profitable direction. Then for sufficiently small α, $x - \alpha s$ is feasible and $\delta(x - \alpha s) > \delta(x)$. To increase $\delta(x)$ we should move away from the boundaries of the constraints in $T(x)$. Thus, we want $a_i^\top s > 0$ for $i \in T(x)$. Hence, $\min\{a_i^\top s \mid i \in T(x)\} > 0$.

($\longleftarrow$) Suppose $\min\{a_i^\top s \mid i \in T(x)\} > 0$. So $a_i^\top s > 0$ for $i \in T(x)$, since for sufficiently small enough $\alpha > 0$, $b_i - a_i^\top x + \alpha a_i^\top s \in \mathcal{R}_0$ $\forall i \in \mathcal{I} \setminus T(x)$ and increase $\delta(x)$, since $\delta(x) + \min\{a_i^\top s \mid i \in T(x)\}$. Hence s is a profitable direction. □

Murty only considered the normal vectors of the constraints as candidates to be profitable directions. Since each direction must lie on the plane $c^\top x = c^\top x_0$, normal vectors were projected onto the current objective plane. Murty considered the directions $s_i = a_i - c^\top c a_i = (I - c^\top c)a_i \quad i \in \mathcal{I}$. We denote the set of these directions by D. Let x_j be the current point and suppose that s is a profitable direction. We look for the step size α that will maximize

$$\delta(\alpha)_j = \{b_i - a_i^\top x_j + \alpha a_i^\top s \mid i \in \mathcal{I}\}. \tag{2.5}$$

This is a min-max problem and we can rewrite it as the following 2-variable LP

$$\begin{aligned} &\textit{Maximize} && \delta \\ &\textit{Subject to} && \delta + \alpha a_i^\top s \le b_i - a_i^\top x_j, \quad i \in \mathcal{I} \\ & && \delta > 0, \end{aligned} \tag{2.6}$$

where(δ, α) are the variables. Murty used the primal simplex to solve (2.6). In chapter 4 we explain how to solve it using bisection. Here is a short description of Murty's centering procedure

2.2.2. Descent Direction. Once the centering step is done, the spherical method will complete several descent steps from the new center and will take the best point. All of descent steps are computationally inexpensive. Let $\bar{x}_j$ be the approximation of optimal solution of the centering step in iteration j. For each descent directions we

Algorithm 1 Murty's procedure for approximation of centering step

Let x_0 be an initial feasible interior point.
Set $j := 0$ and $x_j = x_0$.
$D_j = \{\pm s_i = \pm (I - c^\top c)a_i \mid i \in T(x_j)\}$
while exist a profitable direction in D_i **do**
  $s \in D_j$ *and is a profitable direction*
  $\bar{\alpha} = arg \ \max \{b_i - a_i^\top x_j + \alpha a_i^\top s \mid \forall\, i \in \mathcal{I}\}$
  $x_{j+1} = x_j - \bar{\alpha} s$
  $j = j + 1$
  update D_j and $T(x_j)$
end while

calculate the maximum feasible step size as follow

$$\lambda = \left\{ \frac{b_i - a_i^\top \bar{x}_j - \epsilon_0}{a_i^\top s} \mid \text{for } i \in \mathcal{I} \text{ and } a_i^\top s < 0 \right\}.$$

Since we want the point $\bar{x}_j - \lambda s$, to be an interior point, we use $\epsilon_0 > 0$ in calculating the maximum feasible step size. We list various descent directions can be used in descent step

1 : $s_1 = -c$. From $\bar{x}_j$ we take $s_1 = -c$

2 : $s_2 = \bar{x}_j - \bar{x}_k$, $1 < k < j - 1$. From $\bar{x}_j$ we $s_2 = \bar{x}_j - \bar{x}_k$ for $1 < k < j - 1$, where $\bar{x}_k$ denotes the ball center at iteration k^{th}.

3 : $s_3 = (I - a_i a_i^\top)c$, $i \in T(\bar{x}_j)$. From $\bar{x}_j$ direction $s_3 = (I - a_i a_i^\top)c$ for $i \in T(\bar{x}_j)$ is a descent direction and they are called gradient projection on touching constraint or shortly GPTC. For more detail refer to [**8**].

4 : $s_4 = \displaystyle\sum_{i \in T(\bar{x}_j)} \frac{(I - a_i a_i^\top)c}{|T(\bar{x}_j)|}$.

For more directions refer to [**8, 9**]. After all these directions are tried, the best result is output as the descent direction. In the next section we will discuss the convergence proof.

2.3. Convergence Proof

In this section we provide the convergence proof for the spherical method under the assumption that the centering step is carried to optimality. The First theorem is about (2.3) and shows that always has feasible solution.

THEOREM 2.3.1. *Consider following parametric formulation of (2.3) with the parameter t replacing $c^\top x_0$.*

$$\begin{aligned} \delta(t) = \quad & \max\ \delta \\ \textit{subject to} \quad & \delta + a_i^\top x \ \leq \ b_i \quad \textit{for}\ i \in \mathcal{I}, \\ & c^\top x \leq t. \end{aligned} \tag{2.7}$$

The function $\delta(t)$ is a concave.

PROOF. Suppose that (x_1, δ_1) and (x_2, δ_2) are optimal solutions of (2.3). Thus, $\delta_1 = \delta(t_1)$ and $\delta_2 = \delta(t_2)$ when $t = t_1$ and $t = t_2$, respectively. Consider $\hat{t} = \lambda t_1 + (1-\lambda)t_2$, $0 \leq \lambda \leq 1$. We will first show that $(\hat{x}, \hat{\delta})$, where $\hat{x} = \lambda x_1 + (1-\lambda)x_2$ and $\hat{\delta} = \lambda\delta_1 + (1-\lambda)\delta_2$, is feasible to (2.7) when $t = \hat{t}$. We have, from feasibility of (x_j, δ_j), for $t = t_j$, that

$$\begin{aligned} \hat{\delta} + a_i^\top \hat{x} &= \lambda\delta_1 + (1-\lambda)\delta_2 + \lambda a_i^\top x_1 + (1-\lambda) a_i^\top x_2 \\ &= \lambda(\delta_1 + a_i^\top x_1) + (1-\lambda)(\delta_2 + a_i^\top x_2) \\ &\leq \lambda b_i + (1-\lambda) b_i \\ &= b_i, \end{aligned}$$

and

$$c^\top \hat{x} = \lambda c^\top x_1 + (1-\lambda) c^\top x_2 \leq \lambda t_1 + (1-\lambda) t_2 = t.$$

Since $(\hat{x}, \hat{\delta})$ is feasible, then

$$\delta(\hat{t}) \geq \hat{\delta} = \lambda\delta_1 + (1-\lambda)\delta_2 = \lambda\delta(t_1) + (1-\lambda)\delta(t_2),$$

which establishes the concavity $\delta(t)$. □

Since $\delta(t)$ is concave the existence of a maximum for $t \in [t_{min}, t_{max}]$ is guaranteed. Therefore there exists biggest ball inside the feasible region.

Let $\mathcal{R}(t)$ denote the feasible region for (2.7), then for $t_1 < t_2$,we have $\mathcal{R}(t_1) \subseteq \mathcal{R}(t_2)$. Since $\delta(t)$ is monotonically decreasing as t decreases, moving in a descent direction leads to reductions in the objective value and the radius of biggest ball inside the feasible region.

THEOREM 2.3.2. *Starting from an interior point in the feasible region for (2.1), if the centering step is carried to optimality, the spherical method converges to an optimal solution of (2.1).*

PROOF. For detail of proof refer to [**8, 12**]. □

2.4. Conclusion

The spherical method looks like a promising method in theory, but convergence is highly dependent on the centering step. The procedure that Murty proposed does not give any information about the accuracy of the approximation. In the next chapter we show that the strategy he proposed for QP has difficulties and may lead to points that are not optimal.

CHAPTER 3

Spherical Method for QP

3.1. Introduction

In this chapter we explain how to adapt the spherical method to QP. As mentioned in the previous chapter, in the spherical method the most important step in each iteration is the centering step. The centering step is computationally more expensive than the descent step, so it is important to carry out the centering step quickly, yet with good accuracy. Murty [7] proposed a procedure to get an approximation for the centering step, but there is a fundamental difficulty in his procedure that makes it inefficient. In the next section we try to explain the spherical method for QP, after that in section three we try show its difficulty and suggest a change in the procedure that makes it more efficient.

3.2. QP spherical method

We consider the QP (1.1) with H positive definite so that the QP is strictly convex. The unconstrained minimizer of $Q(x)$ is $y^* = -\mathrm{H}^{-1}c$. If $y^* \in \mathcal{R}$, then the problem is done. We assume that $y^* \notin \mathcal{R}$, since if it is feasible then solution of (1.1) is y^* and can be determined by Cholesky decomposition for solving a positive definite linear system of equations or by any other algorithm that can solve unconstrained minimization problems.

In the previous chapter we described the spherical method for LP. The spherical method for QP consists of the same steps, a centering step and a descent step. But the centering step is different and the descent directions are changed. The aim of the centering step is to determine point within the feasible region that has most possible distance from boundaries of feasible region and at same time has the objective value less than initial point. Let y be the current feasible interior point for $\mathcal{R}$. The problem of finding the largest ball inscribed within $\mathcal{R}$ with objective value less than y, is the min-max problem as follow

$$\max \{ \min\{b_i - a_i^\top x \mid Q(x) \leq Q(y) \text{ and } \forall\, i \in \mathcal{I}\} \}$$

which can be rewritten as the following non-linear problem

$$\begin{aligned} &\textit{Maximize} \quad \delta \\ &\textit{subject to} \quad \delta + a_i^\top x \leq b_i \quad i \in \mathcal{I} \\ &\qquad\qquad\quad Q(x) \leq Q(y). \end{aligned} \tag{3.1}$$

If $(\bar{x}, \bar{\delta})$ is an optimal solution of (3.1), then the biggest ball inscribed within $\mathcal{R}$ is $B(\bar{x})$ and its radius is equal to $\bar{\delta}$. The optimal solution $\bar{x}$ may not unique, but δ is unique. As you can see, the centering step requires the solution of a non-linear optimization problem. Since we must solve this type of model several times, like for the LP, it is not rational to solve it exactly with contemporary methods for non-linear optimization problem. To make this method efficient, a procedure should be developed to get an approximation to the optimal solution of (3.1) without matrix inversion.

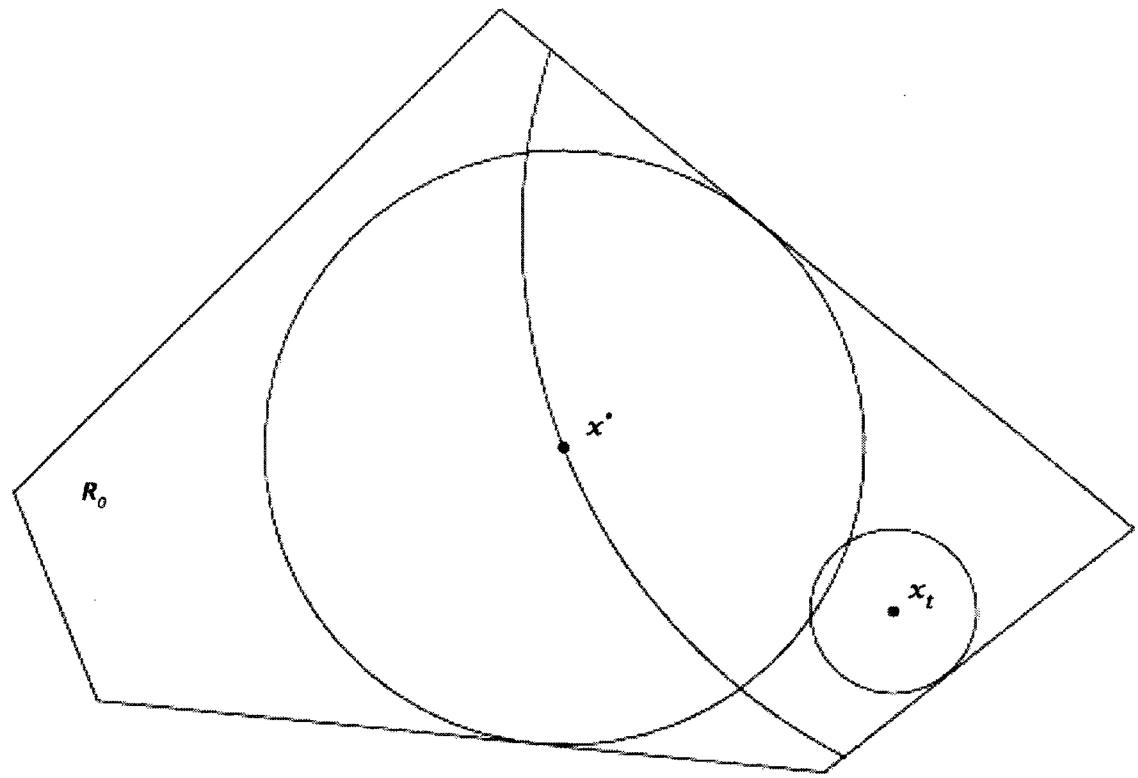

FIGURE 3.1. The QP centering step.

Suppose an approximation $(\bar{x}, \bar{\delta})$ for centering step is available. Then in the descent step we solve the problem

$$\begin{aligned} &\textit{Minimize} && c^\top x + \tfrac{1}{2} x^\top \mathrm{H} x \\ &\textit{subject to} && (x - \bar{x})^\top (x - \bar{x}) \leq \bar{\delta}^2. \end{aligned} \tag{3.2}$$

This is a well known trust-region subproblem and efficient polynomial algorithms exist for its solution, see [**1, 5**]. Let $\hat{x}$ be the optimal solution of (3.2). Then there are two possible cases

(i) If $\hat{x}$ is boundary point of $\mathcal{R}$, then $\hat{x}$ is an optimal solution of (1.1)

(ii) $\hat{x} \in \mathcal{R}_0$ then $s_1 = (\hat{x} - \bar{x})$ is a descent direction for $Q(x)$.

In case (i), the optimal solution is found and we terminate, otherwise we do a line search to minimize $Q(x)$ on line segment $\{x = \bar{x} - \lambda s_1 \mid x \in \mathcal{R}_0\}$, we take $\bar{\lambda} = \min\{\lambda_1, \lambda_2\}$ where λ_1 is the optimal stepsize and λ_2 is maximum feasible step size. If $\bar{x} - \lambda s_1$ is a boundary point of $\mathcal{R}$, let index set $J = \{i \in \mathcal{I} \mid b_i = a_i^\top(\bar{x} - \lambda s_1)\}$. If there exists a solution for the following system of equations (i.e. KKT condition for (1.1))

$$\begin{aligned} -c - (\bar{x} - \lambda s_1)^\top \mathrm{H} &= \sum_{j \in \mathcal{J}(\bar{x} - \bar{\lambda} s_1)} \omega_j a_j \\ & \omega_j \geq 0 \quad \forall j \in \mathcal{J}(\bar{x} - \bar{\lambda} s_1), \end{aligned} \tag{3.3}$$

where ω_j are corresponding Lagrange multipliers, then $\bar{x} - \lambda s_1$ is optimal solution of (1.1) and terminate, otherwise move to $x_{new} = \bar{x} - (\lambda - \epsilon_0)s_1$, where ϵ_0 is same as chapter 2, and set it as output of descent step. Repeat centering step with x_{new} as initial point. The algorithm runs until the stopping conditions are satisfied. The stopping conditions can be the same as other interior point methods.

3.2.1. Murty's Centering Step. As mention before in chapter 2 and above, to make the spherical methods efficient in theory and practice, we need a procedure to carry out the centering step without using matrix inverses or current non-linear optimization algorithms. Murty [**7**] proposed a procedure by using the concept he used for LP (i.e. see [**8, 9**]). Now we describe this procedure in detail.

Suppose $x_0 \in \mathcal{R}_0$. Let $\delta(x)$ and the index set $T(x)$ be the same as in chapter 2. The special structure of (3.1) leads to a strategy of moving perpendicular to the facetal hyperplanes of $\mathcal{R}$, so Murty just considers the normals of the constraints as directions to move. Define $D = \{\pm a_i \mid i \in \mathcal{I}\}$. For convenience, we refer Murty's considered. Murty defines s to be a profitable direction at the point x_j if it satisfies conditions

c_1): $\nabla Q(x_j)^\top s > 0$,

c_2): $\delta(x_j - \alpha s) > \delta(x_j)$ for some $\alpha > 0$.

The first condition can be easily checked for each candidate direction, and by using Theorem (2.2.1), the second condition also can be checked. Procedure M starts with an interior point, uses conditions c_1 and c_2 to check whether or not a profitable direction from D exists. If it determines no profitable direction, it terminates the procedure and uses the current point as the new center. Otherwise, $s \in D$ is a profitable direction for current point x_j. We need a step size to move from point x_j in the direction s_j to next point. The step size is defined to be $\alpha = \min\ \{\alpha_1, \alpha_2\}$ where

$$\begin{aligned}\alpha_1 = &\ \arg\ \min\ \{Q(x_j - \alpha s_j) \mid \alpha \geq 0\}\\ \alpha_2 = &\ \arg\ \max\ \{\delta(x_j - \alpha s_j) \mid \alpha \geq 0\}.\end{aligned}$$

Finding α_1 is the minimization quadratic function in single variable and it is easily calculated (i.e. $\alpha_1 = \frac{\nabla Q(x)^\top s_j}{s_j^\top \mathrm{H} s_j}$). To calculate α_2 is a bit more complicated as we need to solve LP

$$\begin{aligned}&\textit{maximize} && \delta\\ &\textit{subject to} && \delta + \alpha a_i^\top s_j \leq b_i - a_i^\top x_j \quad \textit{for}\ i \in \mathcal{I}\\ & && \delta, \alpha \geq 0,\end{aligned} \tag{3.4}$$

where (δ, α) are variables. This LP is the same as (2.6) in chapter 2. Once α_1 and α_2 are determined, set $\alpha = \min\ \{\alpha_1, \alpha_2\}$ and move to the next point $x_{j+1} = x_j - \alpha s_j$, and repeat. In the next section we make suggestion to get a better approximation of solution (3.1). The following theorem provides an existence proof for (3.1).

and repeat. In the next section we make suggestion to get a better approximation of solution (3.1). The following theorem provides an existence proof for (3.1).

THEOREM 3.2.1. *Consider following parametric formulation of (3.1) with parameter t replacing $Q(x)$.*

$$\begin{aligned} \delta(t) = \quad & \max \delta \\ \textit{subject to} \quad & \delta + a_i^\top x \leq b_i \quad \textit{for } i \in \mathcal{I}, \\ & Q(x) \leq t. \end{aligned} \tag{3.5}$$

Then $\delta(t)$ is a concave function of t.

PROOF. The proof is same as (2.3.1). $c^\top x$ replaced by $Q(x)$, since $Q(x)$ is convex. See [**14, 7**] for complete proof. □

It can be concluded from (3.2.1) that there exists a biggest ball inside feasible region for every t in the interval for what that problem has feasible solution. Also, let $R(t)$ denote feasible region of (3.5), it is obvious that for $t_1 < t_2$, $R(t_1) \subset R(t_2)$. Hence, $\delta(t)$ decrease monotonically as t decreases.

3.2.2. General Iteration. Here is a short description of spherical method.

Algorithm 2 Spherical Method for QP

Let x_0 be an initial feasible point.
Set $j = 0$ and $x_j = x_0$.
Centering step
Get an approximation solution for (3.1), beginning with x_j as initial point. Let $\bar{x}_j$ and $\delta(\bar{x})$ be the approximations. Update $T(x_j)$ and move to the descent step.
Descent step
Apply the described strategy with ball $B(x_j, \delta(x_j))$. If termination doesn't occur in this step, let $\hat{x}_j$ denote the interior feasible point for $\mathcal{R}_0$. Move to the next step.
Next iteration
Check the stopping conditions. If they are satisfied set $x^* = \hat{x}_j$ as the optimal solution. Otherwise set $x_{j+1} = \hat{x}_j$ and $j = j + 1$. Go to centering step

3.3. Improvement in Procedure M

Convergence of spherical methods is proved under the assumption that in the centering step the optimal solution is obtained. In [**14**] it was shown that when the centering step is not carried out with good accuracy convergence does not hold and is not guaranteed. Consider the following numerical example

$$
\begin{array}{lll}
\max & \delta & \\
\textit{subject to} & \delta - x & \leq 0 \\
& \delta - y & \leq 0 \\
& \delta + y & \leq 1 \\
& \delta + 0.573x - 0.819y & \leq 0.245 \\
& (x-2)^2 + (y-2)^2 & \leq 4 \\
& \delta \geq 0. &
\end{array}
\tag{3.6}
$$

We start from point $x_1 = (1, 0.5)$. The direction $s_1 = (0, 1)^\top$ is a profitable direction as it satisfies conditions c_1 and c_2. We move in the direction s_1 to the point $x_2 = (1, 0.73)$ with $\delta = 0.27$. The optimal solution to (3.6) is $x^* = (0.576, 0.576)$ with $\delta^* = 0.403$. There does not exist a direction that satisfies condition c_1 and in that direction we move from point x_2 to x^*. As you can see in figure (3.2) direction s_1 moves point x_1 to a lower level set, so condition c_1 will not be satisfied by any other direction and it is not possible to move to optimal solution of (3.6). Any procedure is used in the centering step, must determine as much as possible a good approximation for centering step. In this section, we provide some suggestion that improve Murty's procedure. In procedure M, the direction s should satisfy $\nabla Q(x)^\top s > 0$ to be a profitable direction, which mean $-s$ must be a descent direction for $Q(x)$. But in (3.1) points must satisfy

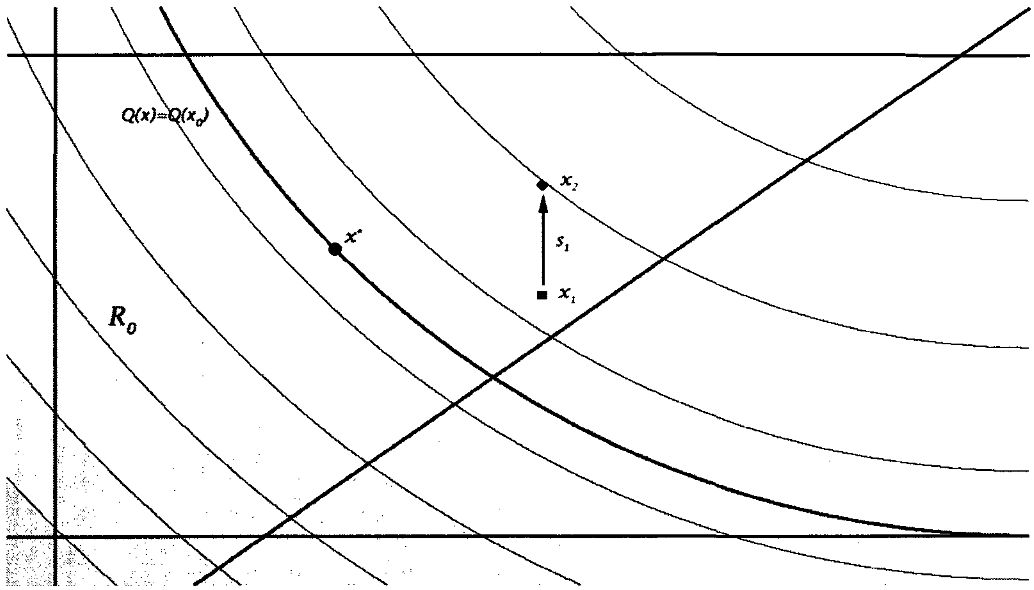

FIGURE 3.2. Murty's procedure failure.

$Q(x) \leq Q(x_0)$ where x_0 is initial point. So the optimal solution of (3.1) can have the same objective function value as x_0, but condition $\nabla Q(x)^\top s > 0$ says that for the sequence x_j for $j = 1, 2 \ldots$, $Q(x_1) \geq Q(x_2) \geq \ldots \geq Q(x_r) \geq \ldots$ which is unnecessary. So condition c_1 is too strict and we need an alternative way to prevent violating $Q(x) \leq Q(x_0)$. Since $Q(x)$ is a strictly convex function $Q(x) \leq Q(x_0)$ is a convex bounded region. Let x be an interior point of this region. For every direction such s from x there exist two values of α with $Q(x - \alpha s) = Q(x_0)$, and more precisely these two values, α_{min} and α_{max}, are

$$\begin{aligned} \alpha_{max} &= \frac{s^\top \nabla Q(x) + \sqrt{(s^\top \nabla Q(x))^2 - 2(s^\top \mathrm{H} s)(\Delta Q(x))}}{s^\top \mathrm{H} s} \\ \alpha_{min} &= \frac{s^\top \nabla Q(x) - \sqrt{(s^\top \nabla Q(x))^2 - 2(s^\top \mathrm{H} s)(\Delta Q(x))}}{s^\top \mathrm{H} s}. \end{aligned} \tag{3.7}$$

Now suppose direction s satisfy condition c_2, then we need to determine step size that maximize $\delta(x)$. Murty used (3.4) to calculate step size. Murty suppose that step size is bigger that zero, but the only reason for that is to satisfy condition c_2(i.e.

$Q(x)s > 0)$. we need solve

$$\max\ \{\min\{b_i - a_i^\top (x - \alpha s) \mid i \in \mathcal{I},\ \alpha_{min} \le \alpha \le \alpha_{max} \text{ and } x \in \mathcal{R}\}\},$$

which has same answer as following 2-variable problem

$$\begin{aligned} & \textit{maximize} && \delta \\ & \textit{subject to} && \delta + \alpha a_i^\top s \le b_i - a^\top x_j \quad \textit{for}\, i \in \mathcal{I} \\ & && \alpha_{min} \le \alpha \le \alpha_{max} \\ & && \delta \ge 0. \end{aligned} \tag{3.8}$$

By restricting α between α_{min} and α_{max} we are sure that the constraint $Q(x) \le Q(x_0)$ is satisfied. So direction s is a profitable direction at point x if $\min\{a_i^\top s \mid i \in T(x)\} > 0$. After that if a profitable direction found, calculate upper and lower bound of step size and solve corresponding (3.8) problem for finding step size.

3.4. Conclusion

Procedure M can determine an estimate for the centering step, but it is not accurate enough, and also it will not provide how accurate is approximation. Murty's procedure is good to find a warm-start initial point or as pre-procedure. It also can provide a lower bound for δ. The main difficulty of procedure M is in the calculation of stepsize for the profitable directions. To make spherical method a practical and reliable algorithm, a computationally inexpensive procedure is needed to carry out the centering step.

CHAPTER 4

A probabilistic procedure for approximation the center

4.1. Introduction

In the previous chapter we discussed Murty's method for QP. We supposed that the feasible region is bounded , and by theorem (3.2.1,2.3.1) we showed that the biggest ball inside the feasible region always exists. We are interested in the procedure that without using matrix inversion get an accurate approximation of (3.1). As shown in **[14]** and assumption of convergence proof, the centering step plays an important part in convergence, so it is necessary to propose a procedure which be able to carry out the centering step accurately and get a good approximation of solution (3.1). We will use a probabilistic method to develop such a procedure. In the next section we provided some interesting theorem and result about $\delta(x)$. In the third section, we explain the idea behind the probabilistic method, after that in section 4 we proposed our procedure and at the last chapter we provide the numerical results from the implementation of our procedure.

4.2. Notes on properties of $\delta(x)$

The function $\delta(x)$ satisfies following properties

i) $\delta(y) = 0$ if y is on the boundary of $\mathcal{R}$;

ii) $\delta(y) > 0$ if $y \in \mathcal{R}_0$;

iii) $\mathcal{R}' \subset \mathcal{R}$ then $\delta(y)_{\mathcal{R}'} \leq \delta(y)_{\mathcal{R}}$.

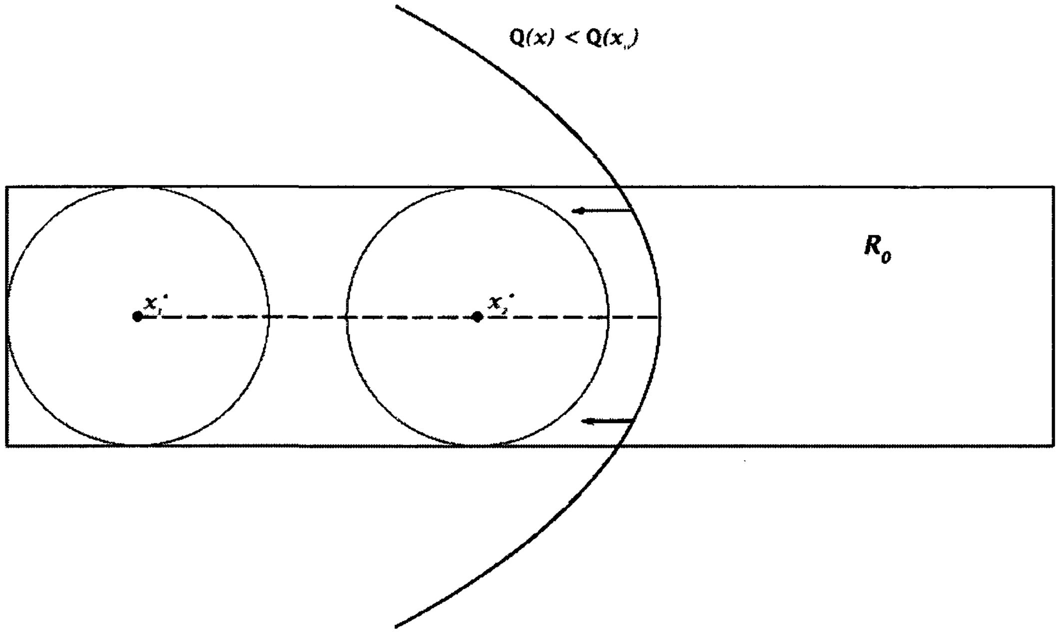

FIGURE 4.1. Center are not unique.

So $\delta(x)$ is a position function. Therefore, instead of the classic analytic center, one can use a point that has the maximum value of $\delta(x)$ and determine the biggest ball can inscribed inside $\mathcal{R}$. From our assumption, we know that optimal solution of (1.1) is not an interior point of $\mathcal{R}$ and it must be quasistationary point, so when we are at optimal point, the function $\delta(x)$ become zero. The most important problem with concept of biggest ball inside the feasible region is this ball can be non-unique, so there will be different centers, and it is not possible to define path of center like analytical center path. Therefore properties of it still are unknown. In figure (4.1) you can see that all of line segment are the optimal solutions. But it is possible, by putting some restriction, to define path of center. The following theorems help us to define this path.

THEOREM 4.2.1. *Let S be the set of all feasible and optimal solutions to (2.2). Let (x^*, δ^*) be such that $Q(x^*) = \min\{Q(x) \mid (x, \delta) \in S\}$. If $x_j \in \mathcal{R}_0$ and $Q(x_j) < Q(x^*)$ and if $(\bar{x}, \bar{\delta})$ is a solution to (3.1), then $Q(\bar{x}) = Q(x_j)$. Furthermore, $(\bar{x}, \bar{\delta})$ is unique.*

PROOF. Suppose that $Q(\bar{x}) < Q(x_0)$. Then $(\bar{x}, \bar{\delta})$ satisfies optimally condition

$$\begin{aligned}
&\sum_{i \in T(\bar{x})} \omega_i = 1 && w_i \geq 0 \\
&\sum_{i \in T(\bar{x})} \omega_i a_i = 0 && i \in T(\bar{x}), \\
&\omega_i = 0 && i \in \mathcal{I} \setminus T(\bar{x}).
\end{aligned} \tag{4.1}$$

which is same as (2.2), therefore $(\bar{x}, \bar{\delta})$ is a solution for (2.2). Since $Q(\bar{x}) < Q(x^*)$, then $\bar{x} \neq x^*$, which is contradiction. So, $Q(\bar{x}) = Q(x_0)$. Suppose $(\bar{x}, \bar{\delta})$ is not unique, so there exists another solution $(\bar{x}, \hat{\delta})$ where $\bar{x} \neq \hat{x}$. Since (3.1) is convex then

$$y = \lambda\bar{x} + (1 - \lambda)\hat{x}, \ \mathit{for}\ \lambda \in [0, 1],$$

$(y, \bar{\delta})$ is a solution for (3.1). From above we know at optimal solution $Q(y) = Q(x_0)$, but set of point on quadratic curve are not a convex set. Therefore $\bar{x} = \hat{x}$ and optimal solution is unique. □

DEFINITION 4.2.2. *Center of Polyhedron If (x^*, δ^*) is the unique optimal solution to (2.2), we call x^* the center of polyhedron.*

Now if the initial point of procedure is equal center of polyhedron, we can define path of center, since (x, δ) are unique in each centering step by theorem (4.2.1). LP is a special case of QP, so if in centering step we use (2.3) instead (2.4), (4.2.1) holds for LP except that the optimal solution (x, δ) maybe is not unique, since the set of point on straight line is convex, so we can not define path of center. Also if the optimal solution of (2.2) is not unique we can define center of polyhedron corresponding to a function like $f(x)$ where $f(x)$ is convex and scalar function.

4.3. Improving Hit-and-Run

Random Search algorithms offer powerful methods for optimization. Random search methods walk around in feasible region and try to improve a solution. Examples of random search are Hit-and-Run, Hide-and-Seek, Pure Random Search. By modification of these methods, new genres are derived. The differences between various random search methods are in how they sample the feasible region. These kinds of methods widely used in global optimization where the chance of being trapped in local optimum is high. To prevent from trapping in local maximal it is quiet common to use a parameter ,called "Temperature". Usually in the beginning of a procedure, the temperature is high and random search is almost unbiased. As the temperature goes down, each iteration of simulated annealing more likely goes toward an optimal solution. In other words, if next random generated point has worse objective value, still there is a positive probability to move that point.

Now suppose we want to determine

$$\textit{minimize}\ f(x)$$
$$x \in S,$$

where S is a convex, compact, full dimensional subset of R^n, x a vector of order n and $f(x)$ is real value convex function on S. Since S is convex, every local optimum is a global optimum. So we shouldn't be worried about local optimal. A class of Random

Search algorithms for solving this problem is sequential random search method. The concept of sequential random search methods is to generate next random point by taking a random direction and move by a step size from previous point. The general iteration in each step of algorithms, for $i = 1, 2, \ldots$, is

$$X_{i+1} = \begin{cases} X_i + \alpha_i D_i & \textit{if} \quad f(X_i + \alpha_i D_i) < f(X_i), \\ X_i & \textit{otherwise}, \end{cases}$$

where X_i is current point, D_i is random direction obtained, not necessarily, by sampling from a uniform distribution on the unit sphere and α_i is the step size. The method of choosing the step size is different in each algorithm.

Improving Hit-and-Run (IHR) proposed by Zabinsky [**16**], is a sequential random search method that take advantage efficiency of HR(Hit-and-Run) [**2, 3**] and PAS(Pure Adaptive Search) [**10, 15**] simultaneously. The difficulty of implementation PAS is efficiently generating a uniform sample of feasible region. A good alternative way to generates this sample is using HR algorithm in each iteration. The structure of IHR is to generate a candidate point along a random direction with a random step in that direction. If next point has better objective value accept it otherwise stay in current point. A brief description of Improving Hit-and-Run is in algorithm 3 below.

It has been shown [**16**] for class of elliptical programs, IHR has search effort that is polynomially bounded. Solis and Wets in [**11**] provide sufficient conditions for convergence of random search methods to solution which are satisfied by IHR. In next section we are going to use IHR to solve (3.1) and (2.3).

Algorithm 3 Improving Hit-and-Run

Step 0. Let $X_0 \in S$,$Y_0 = f(X_0)$, and Set $i := 0$.
Step 1. Pick random direction D_i from uniform distribution on a unit sphere.
Step 2. Generate a step size α_i uniformly form L_i, the set of feasible step sizes. from current iteration point X_i in the Direction D_i, where

$$L_i = \{\lambda \in \mathcal{R} : X_i + \lambda D_i \in S\}.$$

if $L_i = \emptyset$, then go step 1.
Step 3. Update the new point as follow

$$X_{i+1} = \begin{cases} X_i + \alpha_i D_i & \textit{if} \quad f(X_i + \alpha_i D_i) < f(X_i), \\ X_i & \textit{otherwise}, \end{cases}$$

set $Y_{i+1} = f(X_{i+1})$
step 4. Check the stopping criterion, if stratifies,stop.Otherwise $i = i + 1$ and go to step 1.

4.4. determining $\delta(x)$ maximizer

In this section, we present the modification of IHR to solve (3.1). Also at the end of this section, we discuss how to modify this algorithm to solve (2.3). In each step of procedure Murty proposed it should be check that for direction s at current point x, $As < 0$ or $As > 0$ where $A^\top = [a_{j_1}, a_{j_2}, \ldots, a_{j_k}]$ for $j_k \in T(x)$. Determining a solution to satisfy this condition is equivalent to checking Gordan's Theorem, which seems not efficient in every iteration. Instead solving $As < 0$, we pick a random direction from uniform distribution on a unit sphere and we check whether or not it is a profitable direction. If it is, we determine the optimum step size for it, otherwise we pick another random direction and repeat it in same way. Let initial point x_0 be provided. We are looking for a point that maximize $\delta(x)$ and also has the objective value less or at least equal to x_0. Corresponding to each interior point x we have index set $T(x)$, so we can define matrix $A_{T(x)}$ as follow

$$A^\top = [a_{j_1}, a_{j_2}, \ldots, a_{j_k}] \quad \textit{for}\, j_k \in T(x).$$

From theorem (2.2.1) we know that if $A_{T(x)}s > 0$ at point x, then s is a profitable direction. If s, we mean s or $-s$, satisfies condition, then we choose s otherwise pick another random direction and check the condition and repeat until we find a profitable direction. Since we pick the direction randomly uniform, if $Ay > 0$ has a feasible solution, the probability of determining a feasible solution is greater than zero and we will find a profitable direction. So we say direction s is profitable at point if $\exists\, d \in \{s, -s\}$ that satisfies $A_{T(x)}d > 0$. Now, suppose s is a profitable direction. Since $\mathcal{R}$ is bounded, for direction s, there exist two numbers that $x_0 - \alpha s$ is on boundary and are

$$\begin{aligned} \alpha_1 &= \min\{\tfrac{b_i - a_i^\top x}{a_i^\top s} \mid \text{for } i = 1, \dots, m \text{ and } \quad as^\top > 0\} \\ \alpha_2 &= \max\{\tfrac{b_i - a_i^\top x}{a_i^\top s} \mid \text{for } i = 1, \dots, m \text{ and } \quad as^\top < 0\}. \end{aligned} \tag{4.2}$$

It is obvious that $\alpha_1 \geq 0$, $\alpha_2 \leq 0$ and for all number $\alpha_2 \leq \alpha \leq \alpha_1$, $x_0 - \alpha s$ is feasible. This line segment lying completely inside the feasible region. Also we need to satisfy constraint $Q(x) \leq Q(x_0)$. Since we are inside the contour level $Q(x_0)$ and $Q(x)$ is strictly convex for each x inside the contour level and direction s there exist two value that $Q(x_0) = Q(x - \alpha s)$. These two values are

$$\begin{aligned} \alpha_3 &= \frac{\nabla Q(x)^\top s + \sqrt{\nabla Q(x)^\top s - 4 s^\top \mathrm{H} s \Delta}}{s^\top \mathrm{H} s}, \\ \alpha_4 &= \frac{\nabla Q(x)^\top s - \sqrt{\nabla Q(x)^\top s - 4 s^\top \mathrm{H} s \Delta}}{s^\top \mathrm{H} s}, \end{aligned} \tag{4.3}$$

where $\Delta = Q(x) - Q(x_0)$. The step sizes α_3 and α_4 should have different sign and also $\alpha_3 > \alpha_4$, therefore $\alpha_3 \geq 0$ and $\alpha_4 \leq 0$ but both can not be zero at a same time. Now we choice step size as follow

$$\begin{cases} \alpha_{max} = \min\{\alpha_1, \alpha_3\} \quad \alpha_{min} = 0 \quad \text{if } s \text{ is profitable} \\ \alpha_{min} = \max\{\alpha_2, \alpha_4\} \quad \alpha_{max} = 0 \quad \text{if } -s \text{ is profitable.} \end{cases} \tag{4.4}$$

then the line segment $x_0 - \alpha s$ for $\alpha_{min} \leq \alpha \leq \alpha_{max}$ is feasible and satisfy constraint $Q(x) < Q(x_0)$ at same time. Now suppose the feasible region reduce to this line segment and we want to

$$\begin{aligned} &maximize \quad \delta(x) \\ &subject\ to \quad x_0 - \alpha s \quad for \quad \alpha_{min} \leq \alpha \leq \alpha_{max}. \end{aligned}$$

$\delta(x)$ for the line segment is

$$\min \{b - a_i^\top (x_0 - \alpha s) \mid \text{for } i = 1, \ldots, m \text{ and } \alpha_{min} \leq \alpha \leq \alpha_{max}\}.$$

Let β and μ are vectors of order m, also $\beta_i = b - a_i^\top x_0$, $\mu_i = a_i^\top s$. So we can rewrite $\delta(x)$ for line segment as a function of α as follow

$$\delta(\alpha) = \min\{\beta_i + \alpha\mu_i \mid \text{for } i = 1, \ldots, m\} \tag{4.5}$$

therefore for determining the maximizer of $\delta(x)$ on a line segment, we can solve the following problem

$$\begin{aligned} &maximize \qquad \delta(\alpha) \\ &subject\ to \quad \alpha_{min} \leq \alpha \leq \alpha_{max} \end{aligned} \tag{4.6}$$

where α_{min} and α_{max} are same as we mention above. $\delta(\alpha)$ is a concave function and to determine solution of (4.6) we can use any kind repetitive methods. It is good to mention that β and μ are calculated and used in determining the maximum feasible step size, so we do not need to carry out another multiplication. We are able to determine the maximizer of (4.6) easily(for example it can be done with bisection).

The general iteration in each step of procedure starts with point like x_j from previous iteration, generate a random direction s_i form uniform distribution on a unit sphere, calculate step sizes α_{max} and α_{min}. Determine the optimum value of α in (4.6) and set $x_{i+1} = x_i - \alpha s_i$. If the stop criterion satisfy stop, Otherwise repeat these procedure. A brief version of procedure come as follow

---

Algorithm 4

Let x_0 be an initial feasible point.
Set $j = 0$ and $x_j = x_0$,$T(x_j) = T(x_0)$,Flag=FALSE.
while stopping criterion satisfy **do**
  Pick random direction s_j from uniform distribution on a unit sphere .
  Check weather or not s_j is profitable. If yes,Flag=TRUE.
  if Flag=TRUE **then**
    if $A_{T(x_j)}s_j > 0$ **then**
      calculate step sizes for s_i as follow

$$\begin{aligned}
\alpha_1 &= \min\left\{\tfrac{a_i^\top x_j - b_i}{a_i^\top s_j} \mid \text{for } i = 1,\ldots,m \text{ and } a_i^\top s_j < 0\right\} \\
\alpha_3 &= \tfrac{s_j^\top \nabla Q(x_j) + \sqrt{(s_j^\top \nabla Q(x_j))^2 - 2(s_j^\top \mathrm{H} s_j)(\Delta Q(x_j))}}{s_j^\top \mathrm{H} s_j} \\
\alpha_{max} &= \min\{\alpha_1, \alpha_3\} \quad \text{and} \quad \alpha_{min} = 0
\end{aligned}$$

    else
      calculate step size for $-s_j$ as follow

$$\begin{aligned}
\alpha_2 &= \max\left\{\tfrac{a_i^\top x_j - b_i}{a_i^\top s_j} \mid \text{for } i = 1,\ldots,m \text{ and } a_i^\top s > 0\right\} \\
\alpha_4 &= \tfrac{s_j^\top \nabla Q(x_j) - \sqrt{(s_j^\top \nabla Q(x_j))^2 - 2(s_j^\top \mathrm{H} s_j)(\Delta Q(x_j))}}{s_j^\top \mathrm{H} s_j} \\
\alpha_{min} &= \max\{\alpha_2, \alpha_4\} \quad \text{and} \quad \alpha_{max} = 0
\end{aligned}$$

    end if
    Set
    $\alpha = arg\ max\{X + \alpha S \mid \alpha_{min} \leq \alpha \leq \alpha_{max}\}$.
    $x_{j+1} = x_j - \alpha s_j$.
    Update $T(x_{j+1})$.
  end if
  $j = j + 1$.
end while

The stopping criterion can be certain number of iteration like mn or the variation on last hundred iteration be less some arbitrary ϵ_0. Another good stopping criterion is $\frac{\delta(x_{i+1}) - \delta(x_i)}{\delta(x_i) - \delta(x_{i-1})}$. This procedure exactly same as IHR expect the step size generation.

It is possible to choose step size randomly in each iteration, but the calculation on optimum step size is easy and using optimum value of step size increase speed of convergence.

To apply this procedure to determine solution of (2.3) just need to manipulate α_{max} and α_{min}. The only different between (3.1) and (2.3) is in the last constraint (i.e. $Q(x) \leq Q(x)$ and $c^\top x \leq c^\top x_0$), So we just need to change the definition of α_3 and α_4 and at iteration i^{th} there can be define as follow

$$\begin{cases} \alpha_3 = \frac{c^\top x_0 - c^\top x_i}{c^\top s_i} & \alpha_4 = -\infty \quad \text{if} \quad c^\top s_i < 0, \\ \alpha_4 = \frac{c^\top x_0 - c^\top x_i}{c^\top s_i} & \alpha_3 = +\infty \quad \text{if} \quad c^\top s_i > 0, \\ \alpha_3 = +\infty & \alpha_4 = -\infty \quad \mathit{if} \quad c^\top s_i = 0 \end{cases} \tag{4.7}$$

At each iteration of this procedure, we calculate maximum feasible step size. We can use this information to find which constraint are necessary. For direction s, we know the index of first constraint of direction s reaches, then that constraint is necessary. Although you may not find all unnecessary and necessary condition, but this information help to mange number of iteration or relax your problem to a problem with necessary constraint.

4.5. Experimental Result

we implement our algorithm on Matlab 7.8 R2009a and tested it with randomly generated examples using a Toshiba Satellite with Pentium processor (2.1GHz, 4 GB RAM). Each matrix $H_i = \bar{H}_i^\top \bar{H}_i$ where $\bar{H}_i$ is non-zero matrix from order n sampled from normal distribution using Matlab routine "randn". So H_i will be Positive definite. Vector c_i is from order n sampled in same way. To generated Feasible region, we pick a random point x, and generated a random matrix A as coefficient matrix with n columns and m rows. Suppose $b = Ax + e$ where e is vector of order m and all of its entries are 1, then $Ax \leq b$ is a non-empty feasible region. The First 5 examples solved are described in table 1. The columns give the example number, the dimension n of the space, the number of constraints m, the minimum number of necessary constraint N, the value of δ_1 approximation that is found by our procedure, the value of δ_2 is exact solution and last one is $|\delta_2 - \delta_1|$

TABLE 1. Description of the First set of examples : Normal problems

	variables n	constraints m	Necessary Cons. N	Approx. δ_1	Exact sol. δ_2	$\lvert\delta_2 - \delta_1\rvert$
Ex. 1	2	7	4	0.934666	0.93502	0.0003
Ex. 2	5	15	13	1.658348	1.68496	0.02
Ex. 3	10	30	30	1.218375	1.254201	0.03
Ex. 4	20	60	54	0.767861	0.804017	0.03
Ex. 5	50	300	288	1.487063	1.515898	0.02

Redundancy sometimes cause the algorithm does not convergence to the optimal solution. We test our procedure with a set of highly redundant problems. The Second set contains of five highly redundant problems. The Second set examples solved are described in table 2. The column R denote minimum number of redundant constraint. As you can see our procedure can handel highly redundant problems.

TABLE 2. Description of the Second set of examples : Highly redundant problems

	variables n	constraints m	Redundant Con. R	Approx. δ_1	Exact sol. δ_2	$\lvert\delta_2 - \delta_1\rvert$
Ex. 6	2	26	20	3.359262	3.366827	0.007
Ex. 7	5	45	30	0.802441	0.804669	0.002
Ex. 8	10	90	60	0.952450	0.959655	0.007
Ex. 9	20	180	120	1.700652	1.733748	0.03
Ex. 10	50	900	600	1.447781	1.485494	0.03

CHAPTER 5

Conclusion and Future work

In this thesis we have analyzed Murtys proposed procedure for approximation the centering step in QP. His procedure is not able to provide a good approximation in the centering step, since it has difficulties in calculation of stepsize for profitable direction. We suggested a modification in calculation of stepsize that can improve Murty's procedure. Also we have introduced a new procedure for the centering step. This new procedure guarantied the accuracy of the approximation. However the procedure we proposed still must be improved. Also we discovered some of the properties of the new centering strategy that Murty introduced. Murty said that sometimes it is not possible to define the path of centers, but we determined assumptions that the path of center exists and we proved the uniqueness of center. So we are able to define the path of center. Further work should be aimed to developed a faster procedure for carrying out the centering step. From theorem (4.2.1) we know that the optimal solution always is on the quadratic surface, so new procedure should suggest a approach that be able to search the surface quadratic to solve (3.1). Another further work that can be done in this area is to prove that if $\delta \to 0$, then $x \to x^*$ where x^* is optimal solution of (1.1).

Bibliography

[1] ANDREW R. CONN, NICHOLAS I. M. GOULD, P. L. T. *Trust-Region Methods.* MPS-SIAM Series on Optimization, 2000.

[2] CLAUDE J. P. BELISLE, H. EDWIN ROMEIJN, R. L. S. Hit-and-run algorithms for generating multivariate distributions. *Mathematics of Operations Research 18*, 2 (May 1993), 255–266.

[3] H. C. P. BERBEE, C. G. E. BOENDER, A. H. G. R. R. C. L. S. R. L. S. J. T. Hit-and-run algorithms for the identification of nonredundant linear inequalities. *Mathematical Programming 3*, 2 (June 1987), 184–207.

[4] J. BEST, M. Equivalence of some quadratic programming algorithms. *Mathematical Programming 30* (1984), 71–87.

[5] J.MORE, J., AND SORENSEN, D. C. Computing a trust region step. *SIAM Journal on Scientific and Statistical Computing 3* (1983), 553–572.

[6] MICHAEL J. BEST, K. R. *Quadratic Programming : Active set Analysis and Computer programs.*

[7] MURTY, K. G. A new practically efficient interior point method for quadratic programming. *Mathematical Programming and Game Theory for Decision Making 1* (2006), 21–33.

[8] MURTY, K. G. A new practically efficient interior pointmethod for lp. *Algorithmic Operations Research 1* (2006), 3–19.

[9] MURTY, K. G. New sphere methods for linear programs. *Tutorial In Operation Research* (2009), 62–81.

[10] N. PATEL, R.L. SMITH, Z. B. Z. Pure adaptive search in monte carlo optimization. *Mathematical Programming 43* (1988), 317–328.

[11] SOLIS, F. J., AND WETS, R. J.-B. Minimization by random search techniques. *Mathematics of Operations Research 6*, 1 (February 1981), 10–30.

[12] S.Y. CHENG, K. M. The steepest descent gravitational method for linear programming. *Discrete Applied Mathematics 25*, 3 (November 1989), 211–239.

[13] T. L. MORIN, N. PRABHU, Z. Z. Complexity of the gravitational method for linear programming. *Journal of Optimization Theory and Applications 108*, 3 (March 2001), 633–658.

[14] VASILYEVA, V. An analysis of Murtys proposed interior point method for quadratic programming. Master's thesis, University of Windsor, 2008.

[15] ZELDA B. ŻABINSKY, R. L. S. Pure adpative search in global optimization. *Mathematical Programming 53*, 1-3 (January 1992), 323–338.

[16] ZELDA ZABINSKY, ROBERT SMITH, F. M. E. R. D. K. Improving hit-and-run for global optimization. *Journal of Global Optimization 3*, 2 (June 1993), 171–192.

Vita Auctoris

Mr. Pooyan Shirvani Ghomi was born in 1985 in the city of Tehran, Iran. He graduated from the K.N.Toosi University of Technology where he obtained a B.Sc. in Applied Mathematics in 2008. He expects to graduate from the University of Windsor with a Master of Science degree in Mathematics in Summer 2010.