INTERIOR POINT ALGORITHMS FOR LINEAR PROGRAMMING AND THEIR BOUNDARY BEHAVIOUR

BY

CHARLES GWANDI

B.Sc. (Hons.), A.B.U., Zaria. 1994

A thesis submitted to the Post-Graduate School,
Ahmadu Bello University, Zaria, Nigeria
in partial fulfilment of the requirement for the degree of
Master of Science (M.Sc.) In Mathematics.

Department of Mathematics
Faculty of Science,
Ahmadu Bello University,
Zaria-NIGERIA

JULY 19 97

DECLARATION

This thesis is not a published or unpublished work of another author. No part of this thesis has been submitted for a degree in any other University. Information derived from the published or unpublished work of others has been acknowledged in the text.

Signature:.......................

CERTIFICATION

THIS THESIS TITLED "INTERIOR POINT ALGORITHMS FOR LINEAR PROGRAMMING AND THEIR BOUNDARY BEHAVIOURS" BY Charles Gwandi meets the regulation governing the award of the degree of master of science (M.Sc) of Ahmadu Bello University and is approved of its contribution to scientific knowledge and literary presentation.

(Name and Signature)
Chairman, Supervisory Committee.
.... Date: 15/7/97.

Name and signature of External Examiner
Dr. M. Y. Bello .... Date: 15/7/97

Name and Signature of Supervisor
Dr. B. Sani .... Date: 15/7/97

(Name and Signature)
Head, Department of Mathematics
.... Date: 15/7/97

(Name and Signature)
Dean, Post-Graduate School
.... Date: 6/4/98

DEDICATION

My Father in memoriam

My Mother and

My brothers and sisters.

CHARLES GWANDI

ACKNOWLEDGMENT

I thank God Almighty for his guidance and protection throughout this research work. I do express my boundless gratitude to my Research Supervisor Dr. J.N. Singh for suggesting this topic and for his guidance during the course of study but had to travel half way due to some unforseen circumstances.

I also thank Dr. Babangida Sani who had to take over from Dr. Singh and worked tirelessly to make sure the work is successfully completed.

My thanks also goes to my Assistant Supervisor Professor M.A. Rashid who is also the present Head of Mathematics Department, for his encouragement, helpful advice and constructive criticisms.

I wish also to thank the entire members of staff of the Mathematics Department and my colleagues for their concern and moral support throughout the period this work is being done.

I am also indeed most grateful and wish to express my gratitude and appreciation to Professor M. Kojima of the Department of Mathematical and Computing Sciences, Tokyo Institute of Technology, Japan, Professor J.M. Todd of School of Operations Research and Industrial Engineering, Cornell University, Ithaca, New York, U.S.A. for sending me research materials that enabled me to successfully complete this program.

Unless I forget, I wish to thank Mr. Wunde John A; Karl for his untiring effort and encouragement throughout the period this work is being done and also my beloved, Umeh Mirian Uchenna for her tremendous moral support and for compiling the bibliography.

I am also indebted to my family for their support, understanding and encouragement throughout this program.

Finally, I wish to thank Isah Abdulkarim who typed the original version of this work.

TABLE OF CONTENTS

SYMBOL PAGE

A	=	m x n constraint matrix
Z	=	Objective function
X, Y, Z, W	=	Feasible solutions
α_j	=	Column vectors of matrix A
λ	=	Lagrange multiplier
J_F	=	Jacobian matrix
Γ	=	Curve in path-following algorithm
C	=	Price vector
b	=	Requirement vector
S	=	Convex set
Δ_j	=	Optimality condition for the simplex algorithm
R^n	=	n-dimensional Euclidean space for real numbers
Z^n	=	n-dimensional Euclidean space for integers
Ω	=	Affine subspace
Δ	=	Simplex
P	=	Polytope
Σ	=	Hyperplane
D_n	=	Diagonal matrix of order n x n
L^1	=	Input length
( )	=	Equations
[ ]	=	References
δ	=	Improvement in Karmarkar's potential function
η	=	Search direction in transformed space
ζ	=	Search direction in original space
α	=	Step length
π	=	Deviation

ABSTRACT

This work presents a survey of recent developments in linear programming. The whole project is divided into four chapters.

Chapter one introduces the formulation of the linear programming problem. The graphical method is mentioned. Also mentioned is the simplex method procedure for solving linear programming problems.

In chapter two, the concept of duality is introduced. Proof of the fundamental theorem of duality and complementarity slackness theorem are provided.

In chapter three, the various interior point algorithms recently developed are critically analyzed.

In chapter four, the boundary behaviour of the interior point algorithms is discussed.

INTRODUCTION

Linear programming deals with that class of programming problems for which all relations among the variables are linear. The relations must be linear both in the constraints and in the function to be optimized.

The term "Linear" means that all the relations in the particular problem are linear the and term "Programming" refers to the process of determining a particular programme or plan of action.

Interior point algorithms for linear programming usual update a point X, interior to the feasible polyhedron P, by moving a straight line in the direction of a vector V(X) defined at X. The new point depends of course not only on the direction of V(X) but also on the step size $\alpha(x)$ which is assigned at X. Thus the new point can be represented in the form $X^1 = X + \alpha(X) V(X)$ where $\alpha(X)$ denotes a real number that determines the step size. For a problem in the minimization form the new point has the form $X - \alpha(X) V(X)$ where $\alpha(X)$ is positive.

There is a great variety of derivations of the different interior point algorithms. Originally, these algorithms were designed to apply to various different forms of linear programming problems. But when they are adapted to the standard form linear programming problem, most of them have the common property that the search direction along which new interior feasible solutions are generated from a current feasible solution $X \in \mathbb{R}^n$ is composed of the two direction XPXC (steepest descent direction) for the objective function in the transformed space and XPc (centering direction), where X is a diagonal matrix; P is an orthogonal projection matrix and e is a vector of 1's. This observation by Todd is due to Gonzaga [15]. See also Mitchell and Todd [31], Kojima, Mizuno and Yoshise [22].

In general, projective methods require $0(nL')$ iterations and path following methods $0(\sqrt{n}L')$ iterations to solve a problem with n variables and integer data with input length L'. Affine methods in their original form are not believed to be polynomially bounded, Todd [37]. Each iteration requires $0(n^3)$ arithmetic operations to solve a linear system of equations, in the basic methods. But, modified versions of the algorithm use a trick due to Karmarkar that needs only $0(n^{5/2})$ arithmetic operations per iteration on average. Hence, the overall complexity of the projective method and path following method are $0(n^{7/2}L')$ and $0(n^3L')$ respectively. These bounds are however established in very different ways. Projective method assures a fixed decrease in Karmarkar's potential function at each iteration by performing line searches. Path following methods stay close to a central trajectory, while driving the objective function value to optimal value by measuring the Optimality gap or duality gap directly. The affine methods use only diagonal scaling and projected gradient ideas. All the interior point algorithms continuously extsome vertex and moves from vertex to vertex improving objective function units it arrives at optimal vertex, an interior point algorithm starts at an interior point and goes almost directly to the optimal vertex making a few corrections to the path.

CHAPTER ONE

1.0 Definitions and Preliminaries

We begin this chapter by discussing the basic concepts and techniques involved in linear programming, that will facilitate the understanding of the theoretical and computational aspects of interior point algorithms for linear programming.

In this write up, we will be maximizing or minimizing the objective function (Z) subject to some given constraints. Whatever case we consider, the same result apply to the other.

1.1 General Linear Programming Problem

A linear programming problem consist of a set of simultaneous linear equations which represent the conditions of the problem and a linear function which expresses the objective function of the problem. The linear function which is to be optimized is called the objective function, while the conditions of the problem expressed as simultaneous linear equations (or inequalities) are referred to as constraints.

Note that by optimization we mean maximization or minization.

A general linear programming problem can be stated as follows:

Find real numbers x_1, x_2, ..., $_n$ which optimize the linear function

$$Z = c_1 x_1 + c_2 x_2 + \ldots + c_n x_n \qquad \ldots \quad (1)$$

subject to the constraints

$$a_{11}\ x_1 + a_{12}\ x_2 + \dots + a_{1n}\ x_n\ (\le, =, \ge)\ b_1$$
$$a_{21}\ x_1 + a_{22}\ x_2 + \dots + a_{2n}\ x_n\ (\le, =, \ge)\ b_2$$
$$\dots \quad \dots \quad \dots \quad \dots \quad \dots$$
$$\dots \quad \dots \quad \dots \quad \dots \quad \dots \qquad \dots \quad (2)$$
$$a_{m1}\ x_1 + a_{m2}\ x_2 + \dots + a_{mn}\ x_n\ (\le, =, \ge)\ b_m$$

and non-negativity restrictions

$$x_j \ge 0,\ j = 1, 2, \dots, n \qquad \dots \quad (3)$$

where all a_{ij}'s, b_i's and c_j's are constants and x_j's are variables.

In the conditions given by (2), there may be any of the three symbols, $\le, =, \ge$.

The function Z given by (1) is called the objective function and the conditions given by (2) are termed constraints of the linear programming problem.

We shall always assume that all $b_i \ge 0$. If any one is negative, we make it positive by multiplying both sides of the inequality by -1. By this, the inequality is also reversed. In matrix notation, the above linear programming problem may also be stated as follows:

Optimize $Z = C^T X$

subject to $AX\ (\le, =, \ge) b$

$X \ge 0$

where $A = [a_{ij}]$ is an m x n matrix of coefficients

$C = (c_1,\ c_2,\ \dots,\ c_n)$ is the price vector $X=[x_1, x_2, \dots, x_n]$ constitutes the decision variables, $b = [b_1,\ b_2,\ \dots,\ b_m]$ is the requirement vector and 0 is an n-dimensional null vector. The constraints a_{ij}'s are called activities. The column vector formed by the coefficients of x_j in the constraint matrix A is denoted by α_j.

That is $\alpha_j = [a_{1j}, a_{2j}, \ldots, a_{mj}]$

$$j = 1, 2, \ldots, n$$

Than $A = (\alpha_1, \alpha_2, \ldots, \alpha_n)$.

In a general linear programming problem, it is assumed that the number of rows of the coefficient matrix A is less than its number of columns.

We next look at some basic definitions connected with the linear programming problem.

(1) **Basic Solution (BS)**

Consider a system AX = b of m equations in n unknowns (n > m) and rank (A,b)=m. Then none of the equations is redundant.

If any m x m non-singular submatrix is chosen from A and if all the (n-m) variables not associated with the columns of this matrix are set equal to zero, the solution to the resulting system of equations is called a basic solution. Thus in a basic solution, (n-m) variables must vanish.

The m variables associated with the columns of the above non-singular matrix which may be different from zero are called basic variables.

If B is the matrix of m linearly independent vectors of

A and X_B is the vector of corresponding variables (basic variables) then the basic solution is given by $X_B = B^{-1}b$. A basic solution is called non-degenerate if none of the basic variables is zero and degenerate if one or more of the basic variables are zero.

(2) **Feasible Solution (FS)**

A feasible solution to a linear programming problem is any set of values of the variables which satisfies the set of constraints and the non-negativity restrictions of the problem.

(3) **Basic Feasible Solution (BFS)**

In a linear programming problem, a feasible solution which is also basic is called a basic feasible solution.

(4) **Optimum (or optimal) Solution**

A basic feasible solution to a linear programming problem is said to be an optimal solution if it gives a maxima or minima of the objective function $Z = C^T X$.

(5) **Jacobian Matrix**

Considering all vectors to be column vectors, let $D_1, D_2, \ldots, D_n$ be the usual parital derivatives. Thus $D_i = \partial/\partial \quad x_i$.

Let $F: \mathbb{R}^n \rightarrow \mathbb{R}^m$ be a mapping. We can represent F as a coordinate function. In order words, there exist functions $f_1, \ldots, f_m$ such that

$$F(X)\begin{pmatrix} f_1(X) \\ f_2(X) \\ \vdots \\ f_m(X) \end{pmatrix} = (f_1(X), \ldots, f_m(X))^T$$

e view X as a column vector $X = (x_1, x_2, \ldots, x_n)^T$. Let us assume that the partial derivatives of each function $f_i(i=1,2,\ldots,m)$ exists. We can then form the matrix of partial derivatives.

$$\frac{\partial f_i}{\partial x_j} = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \vdots & \vdots \\ \frac{\partial f_m}{\partial x_1} & \frac{\partial f_m}{\partial x_2} & \frac{\partial f_m}{\partial x_n} \end{pmatrix} = \begin{pmatrix} D_1 f_1(X) & \cdots & D_n f_1(X) \\ \vdots & & \\ D_1 f_m(X) & \cdots & D_n f_m(X) \end{pmatrix}$$

where $i = 1,2,\ldots, m$ and $j = 1, 2, \ldots, n$

This matrix is called the Jacobian matrix of F, and is denoted by $J_F(X)$. In the case of two variables (x,y), say F(x,y)=(f(x,y), g(x,y)), then Jacobian matrix is

$$J_F(x,y) = \begin{pmatrix} \frac{\partial f}{\partial x} & \frac{\partial f}{\partial y} \\ \frac{\partial g}{\partial x} & \frac{\partial g}{\partial y} \end{pmatrix}$$

(6) <u>Open Ball</u>

Let $P \in \mathbb{R}^n$ and let a be a number >0. The set of points X such that $||X-P|| < a$ is called an open ball of radius a and centre P.

The set of points X such that $||X-P|| = a$ is called a sphere of radius a and centre P.

(7) **Hyperplane**

In $\mathbb{R}^n$ the set of points $X=(x_1, x_2, \ldots, x_n)$ satisfy $c_1x_1 + c_2x_2 + \ldots + c_nxn = Z$ (not all $\epsilon_i = 0$) defines a hyperplane for given values of c_i's and Z.

Note that, if Z = 0, then the hyperplane is said to pass through the origin and its equation is then written as

$c_1x_1 + c_2x_2 + \ldots + c_nx_n = 0$

$$C^TX = 0$$

or this implies that C is orthogonal to every vector X in the hyperplane

(8) **Interior Point**

Let $S \epsilon \mathbb{R}^n$ be a set of points. A point $P \epsilon S$ is said to be an interior point of S if there exists an open ball B of positive radius, centred at P and such that B is contained in S.

A point P (not necessarily in S) is called a boundary point of S if every open ball B centred at P includes a point of S and also a point which is not in S.

If a set contains all its boundary points, then we shall say that the set is closed.

Finally, a set is said to be bounded if there exists a number b > 0 such that for every point X of the set, we have $||X|| \leq b$.

(9) **Line Segment in $\mathbb{R}^n$**

The line segment joining two given points $X_1=(x_1^{(1)}, x_2^{(1)}, \ldots, x_n^{(1)})$ and $X_2=(x_1^{(2)}, x_2^{(2)}, \ldots, x_n^{(2)})$, is the collection of all points $X = (x_1, x_2, \ldots, x_n)$ where $X = \lambda X_1 + (1-\lambda) X_2$ for all values of λ such that $0 \le \lambda \le 1$.

(10) Extreme Point

A point X in $S \in \mathbb{R}^n$ is called an extreme point if it cannot be expressed as a convex combination of any two points of S.

(11) Point-to-Set Mapping

Let X and Y be two sets in $\mathbb{R}^n$ and $\mathbb{R}^m$ respectively. If with each element $x \in X$ we associate a subset $F(x) \subset Y$, we call this correspondence a point-to-set mapping of points in X into subsets of Y.

1.3 Convex Sets and their properties

Convex Set

A subset S of $\mathbb{R}^n$ is called a convex set if the line segment joining any two points X_1, X_2 in S lies in S. In other words, a subset S of $\mathbb{R}^n$ is called a convex set if $X_1, X_2 \in S$ implies that

$$\lambda X_1 + (1-\lambda) X_2 \in S \text{ for all } \lambda \text{ such that } 0 \le \lambda \le 1.$$

Theorem 1 A hyperplane is a convex set.

Proof Consider the hyperplane.

$$S = \{X \in \mathbb{R}^n \mid C^T X = Z\}$$

Let X_1 and X_2 be any two points in the hyperplane S, i.e

$C^T X_1 = Z$ and $C^T X_2 = Z$.

Now, we consider $\lambda X_1 + (1-\lambda) X_2$ which is a convex combination of X_1 and X_2.

$$C^T(\lambda X_1 + (1-\lambda)X_2) = \lambda C^T X_1 + (1-\lambda)C^T X_2$$

$$= \lambda Z + (1-\lambda)Z$$

$$= Z$$

This implies that $\lambda X_1 + (1-\lambda)X_2 \in S$. Hence by definition, the hyperplane S is a convex set.

Theorem 2 The set of all feasible solutions to a linear programming problem is a convex set.

Proof Let S be the set of all feasible solutions to the linear programming problem

$$\text{Maximize } Z = C^T X$$

$$\text{subject to } AX \leq b \qquad \ldots \quad (4)$$

$$X \geq 0$$

Let X_1, $X_2 \in S$. Then X_1 and X_2 are feasible solutions to linear programming problem (4) i.e.

$$AX_1 \leq b, \ X_1 \geq 0$$

$$AX_2 \leq b, \ X_2 \geq 0$$

Now, we consider $X = \lambda X_1 + (1-\lambda)X_2$

Then obviously, $X \geq 0$,

and

$$AX = A(\lambda X_1 + (1-\lambda)X_2)$$

$$= \lambda AX_1 + (1-\lambda)X_2$$

$$= \lambda b + (1-\lambda)b$$

$$\leq b$$

This implies X is a feasible solution.

That is $X \in S$.

Hence S is a convex set.

Theorem 3

Intersection of any two convex sets is a convex set.

Proof

Let S_1 and S_2 be any two convex sets. We consider $S_1 \cap S_2$.

Let $X_1, X_2 \in S_1 \cap S_2$

$\Rightarrow X_1, X_2 \in S_1$ and $X_1, X_2 \in S_2$

i.e

$\lambda X_1 + (1-\lambda)X_2 \in S_1$

$\lambda X_1 + (1-\lambda)X_2 \in S_2$

$\Rightarrow \lambda X_1 + (1-\lambda)X_2 \in S_1 \cap S_2$

Hence $S_1 \cap S_2$ is a convex set.

Theorem 4

Every basic feasible solution to the system $AX=b$, $X\geq 0$ is an extreme point of the convex set of feasible solutions and vice versa.

Theorem 5

If a linear programming problem has at least one feasible solution then it has at least one basic feasible solution. Proofs of these theorems can befound in Gupta [16], Hadley [17].

Theorem 6 (Fundamental theorem of linear programming).

If a linear programming problem has an optimal solution, then at least one basic feasible solution must be optimal.

Proof

Let $X^* = [x_1, x_2, \ldots, x_n]$ be an optimal feasible solution to linear programming problem

Maximize $Z = C^T X$

subject to $AX = b$

$X \geq 0$

If α_i denotes the i^{th} column of the matrix A, then $A = (\alpha_1, \alpha_2, \ldots, \alpha_n)$.

Let $Z^k = C^T X^* = \sum c_i x_i$ be the maximum value of the objective function corresponding to X^*. Without any loss of generality, we can assume that the first k components of X^* are non-zero and remaining (n-k) are zero. So, we have

$$\sum_{j=1}^{k} x_j \alpha_j = b \qquad \ldots \quad (5)$$

and

$$\bar{Z}^* = \sum_{j=1}^{k} c_j x_j \qquad \ldots \quad (6)$$

Now, if $\alpha_1, \alpha_2, \ldots, \alpha_k$ are linearly independent, then by definition X^* is a basic feasible solution which is also optimal. Hence theorem is true in this case.

If $\alpha_1, \alpha_2, \ldots, \alpha_k$ are linearly dependent, then there exist scalars $\lambda_1, \lambda_2, \ldots, \lambda_k$ (not all zero) such that

$$\sum_{j=1}^{k} \lambda_j \alpha_j = 0 \qquad \ldots \quad (7)$$

Now, we assume that at least one λ_i is positive (because if all are negative, then we can multiply (7) by -1 to get positive λ_i).

$$Let\ V = \max_{1 \le i \le k} \left(\frac{\lambda_i}{X_i}\right), \quad then\ V > 0 \qquad \dots (8)$$

Multiplying (7) by 1/V and subtracting from (5), we get,

$$\sum_{i=1}^{k} \left(X_i - \frac{\lambda_i}{V}\right) \alpha_i = b \qquad \dots (9)$$

$$i.e.\ X^1 = \left[X_1 - \frac{\lambda_1}{V}, \dots, X_k - \frac{\lambda_k}{V}, 0, 0, \dots, 0\right] \qquad \dots (10)$$

is also a solution of the system AX = b.

$$Since \quad V \ge \frac{\lambda_i}{X_i}$$

This implies

$$X_i - \frac{\lambda_i}{V} \ge 0$$

which imples $X^1 \ge 0$.

For at least one i,

$$V = \frac{\lambda_i}{X_i} \qquad implies \quad X_i - \frac{\lambda_i}{V} = 0$$

This means that X^1 cannot have more than (K-1) non-zero components.

$$Now\ \bar{Z}^1 = \sum_{i=1}^{k} C_i\left(x_i - \frac{\lambda_i}{V}\right) = \sum_{i=1}^{k} C_i x_i - \frac{1}{V}\sum_{i=1}^{k} C_i\lambda_i$$

Thus, Z^1 will be the optimal value of the objective function if and only if $Z^1=Z^*$.

$$i.e.\quad if\ and\ only\ if \sum_{i=1}^{k} C_i\lambda_i = 0$$

$$Suppose\ if\ possible \quad \sum_{i=1}^{k} C_i\lambda_i \neq 0$$

$$i.e\ either\ (i) \sum_{i=1}^{k} C_i\lambda_i < 0$$

Now we can find a real number r (negative in case (i) and positive in case (ii)) such that

$$r\sum_{i=1}^{k} C_i\lambda_i > 0 \qquad or$$

$$i.e.\ \sum_{i=1}^{k} C_i(r\lambda_i) > 0$$

This implies

$$\sum_{j=1}^{k} C_j(r\lambda_j)+\sum_{j=1}^{k} C_j x_j>\sum_{j=1}^{k} C_j x_j$$

which implies

$$\sum_{j=1}^{k} C_j(x_j+r\lambda_j)>\bar{Z}^* \qquad \ldots (12)$$

Multiplying (7) by r and adding to (5), we get

$$\sum_{j=1}^{k} (x_j+r\lambda_j)\alpha_j=b.$$

This means $[x_1 + r\lambda_1, \ldots, x_k + r\lambda_k, 0, 0, \ldots, 0]$ for all values of r is also a solution of AX = b.

If we choose r such that

$$\max_{\lambda_j>0}\left(\frac{(-x_j}{\lambda_j}\right)\leq r\leq \min_{\lambda_j<0}\left(\frac{-x_j}{\lambda_j}\right) \qquad \ldots (13)$$

then $x_i + r\lambda_i > 0$.

Thus if r lies in the interval given by (13) then the solution $X = [x_1 + r\lambda_1, x_2 + r\lambda_2, \ldots, x_k + \lambda_k, 0, 0, \ldots, 0]$ is also a feasible solution to the system.

AX = b.

It now follows from (12) that this feasible solution X gives a value of Z which is greater than the maximum value which is a contradiction.

Hence we must have

$$\sum_{i=1}^{k} C_i \lambda_i = 0.$$

ie $Z^1 = Z^*$. This shows that

X^1 given by (10) is also an optimal solution. Thus we have derived a new optimal solution containing less number of non-zero variables. This solution will be optimal basic feasible solution if column vectors associated to non-zero variables is new solution are linearly independent. Hence the theorem in true. If associated column vectors are not linearly independent, then by repeating the above process, we can get another optimal basic feasible solution of given system containing not more than (k-2) non-zero variables.

Continuing this process a finite number of times we will get an optimal basic feasible solution. This completes the proof.

1.4 Methods of Solving Linear Programming Problem

There are actually a variety of methods of solving a linear programming problem. The graphical method is one of such methods which is used to solve a linear programming problem involving two variables only.

The simplex method ranks among the best. It was developed by Dantzig G.B in 1947 which was made available in 1951 as an alternative method for solving linear programming problems. This method involves an iterative (step by step) procedure in which we proceed in systematic steps from an initial basic feasible solution to other basic feasible solutions and finally, in a finite number of steps, to an optimal basic feasible solution, in such a way that the value of the objective function at each step is better than the one at the last step. In other words, the simplex algorithm consists of the following main steps:

(1) Finding a trial basic feasible solution of the linear programming problem.

(2) Testing whether it is an optimal solution or not.

(3) Improving the first trial basic feasible solution (if it is not optimal) by a set of rules.

(4) Repeating the steps (2) and (3) till we get an optimal solution.

The method also indicates whether there is an unbounded solution of the given linear programming problem. A problem is said to have an unbounded solution if the objective function increases or decreases arbitrarily. That is, there is no finite optimal value of the objective function.

Before we look at the computational procedure of the simplex algorithm, it is nice we discuss the following:

(1) **Slack variables**

If a constraint has a sign $\leq$, then in order to make it an equality, we have to add something positive to the left hand side. The positive variables which are added to the left hand side constraints to convert them into equality constraints are called slack variables.

(2) **Surplus variables**

If a constraint has a sign $\geq$, then in order to make it an equality, we have to subtract something positive from its left hand side. The positive variables which are subtracted from the left hand side constraints to convert them into equations are called surplus variables.

(3) **Artificial variables**

In linear programming problems, some constraints may have the signs $\geq$ or $=$ with all b_i's positive. In such problems we introduce surplus variables in the constraints with signs $\geq$ and $=$. In these problems, we cannot get the starting basic matrix $B = I_m$. So to avoid this difficulty, we add one more variable to each other of such constraints. These variables are called "artificial variables".

The artificial variables technique is merely a device to get the starting basic feasible solution so that we may proceed with simplex method to get the optimal solution.

When artificial variables are applied to a problem, there are two techniques to deal with the artificial variables, viz:

(a) Two phase method

(b) The Big M method where M is usually considered to be very large.

1.5 Notations

(i) Let B be an m x n non-singular matrix where column vectors are linearly independent columns of A. If columns are denoted by β_1, $\beta_2, \ldots,$ β_m, then B =

$(\beta_1, \beta_2, \ldots, \beta_m)$.

The matrix B is called the basic matrix.

(ii) The variables corresponding to $\beta_1, \beta_2, \ldots, \beta_m$ called basic variables are denoted by $x_{\beta 1}, x_{\beta 2}, \ldots, x_{\beta m}$ respectively.

The vector of these m basic variables is denoted by X_B.

i.e $X_B = [x_{\beta 1}, x_{\beta 2}, \ldots, x_{\beta m}]^T$

where $X_B = B^{-1}b$ and is called basic feasible solution of the linear programming problem.

(iii) Since $B = (\beta_1, \beta_2, \ldots, \beta_m)$ is a non-singular matrix of order m x m, and the vector $\beta_1, \beta_2, \ldots, \beta_m$ are linearly independent, they form the basis of $\mathbb{R}^m$. Therefore, each vector in $\mathbb{R}^m$ can be expressed as a linear combination of vectors in B.

Let $\alpha_j = \beta_j y_{2j} + \beta_2 y_{2j} + \ldots + \beta_m y_{mj}$

$$= (\beta_1, \ldots, \beta_m)\begin{pmatrix} y_{1j} \\ y_{mj} \end{pmatrix} = BY_j \; Y_j = \begin{pmatrix} y_{1j} \\ y_{mj} \end{pmatrix}$$

where

$y_{1j}, y_{2j}, \ldots, y_{mj}$ are scalars required to express α_j $j=(1,2,\ldots,n)$ as linear combination of $\beta_1, \beta_2, \ldots, \beta_m$.

$$\therefore Y_j = B^{-1}\alpha_j.$$

(v) Finally, we write

$$Z_j = C_B Y_j = C_{B1} Y_{1j} + C_{B2} Y_{2j} + \ldots + C_{Bm} Y_{mj}$$

Next, we state a series of theorems which form the core of the simplex algorithm. The proofs can be found in Hadley [17], Kanti, Gupta, Man [18}.

<u>Theorem 7</u> (To determine improved BFS).

Let $X_B = B^{-1}b$ be a BFS to a linear programming problem with $Z=C_BX_B$ as the value of the objective function. If for any column α_j in A but not in B, the condition $c_j - z_j > 0$ holds and if at least one $y_{ij} > 0$, $i = 1, 2, \ldots, m$ then it is possible to obtain a new BFS by replacing one of the columns in B by α_j and if the new value of the objective function is Z^1 then $Z^1 \geq Z$. If given BFS is non-degnerate then $Z^1 > Z$.

<u>Theorem 8</u> (Optimality Condition).

Let $X_B = B^{-1}b$ be the BFS to the linear programming problem

Maximize $Z = C^TX$

subject to $AX = b$

$X \geq 0$

and $Z^* = C_BX_B$ be the value of the objective function for this BFS. If $C_j - Z_j \leq 0$ for every column α_j in A but not in B, then Z^* is the optimal (maximum) value of the objective function Z and this BFS X_B is an optimal BFS.

<u>Theorem 9</u> (Unbounded Solution)

If for any BFS to a linear programming problem

Maximize $Z = C^T X$

subject to $AX = b$

$X \geq 0$,

there are some columns α_j in A but not in B for which $C_j - Z_j > 0$ and $y_{ij} \leq 0$ $(i = 1, 2, \ldots, m)$ then the problem has an unbounded solution.

<u>Theorem 10</u> (Min-Max theorem)

The minimization of a function $Z = C^T X$, is equivalent to the maximization of the negative expression of this function, $-Z = C^T X$. i.e. Minimize Z = - Maximize (-Z).

<u>Computational procedure of the simplex method for the solution of a maximization linear programming problem</u>

<u>Step 1</u> If the linear programming problem is minimization one, convert it into maximization problem using min-max theorem (10) above.

<u>Step 2</u> Make all b_i's positive.

<u>Step 3</u> Convert the constraints into equations by introducing the non-negative slack or surplus variables. Also introduce artificial variables in the constraints where surplus variables are inserted and which do not form the column of the unit matrix.

<u>Step 4</u> Find initial starting BFS.

<u>Step 5</u> Construct a simplex table (see appendix A).

<u>Step 6</u> Test starting BFS for optimality.

(i) If $\Delta_j = C_j - Z_j \leq 0$ for each j, then the solution under test is optimal.

(a) If some of Δ_j are zero, then other optimal solutions exist with the same value of the objective function Z.

(b) If $\Delta_j < 0$ for every j, then unique solution will exist.

(ii) If $\Delta_j > 0$ for any j, then the solution under the test is not optimal and go to step 7.

Step 7 **To Find entering and outgoing vector**

The incoming vector will be taken as α_k if Δ_k=Max Δ_j. The outgoing vector β_r is taken corresponding to that value of r for which

$$\frac{X_{Br}}{Y_{rk}} = Min\left|\frac{X_{Bi}}{Y_{ik}}, Y_{ik}>0\right\}$$

Note that if minimum is not unique, solution will be degenerated.

Step 8 If α_k be the entering vector and β_r be the outgoing vector, then the element y_{rk} = (α_{rk}) is called key element or pivot element.

1.6 Degeneracy

If for any linear programming problem the choice of outgoing vector (vector to be deleted from the basis) is not unique, that is, if

$$Min\left\{\frac{X_{Bi}}{Y_{ik}}, Y_{ik}>0\right\}$$

occurs at $i = i_1, i_2, \ldots, i_s$, then the next solution may be a degenerate basic feasible solution. Such a problem is called the problem of degeneracy in linear programming. Sometimes, due to presence of degeneracy, the same sequence of simplex tables are repeated continuously without reaching the optimal solution. This problem is known as cycling problem.

Conditions for Occurence of degeneracy

(i) Degeneracy may appear in a linear programming problem at the very first iteration when some $b_i = 0$.

(ii) All b_i's are positive but the choice of the vector to be deleted from the basis β_r at any iteration is not unique.

Resolving Degeneracy

Degeneracy could be resolved in many ways. Sometimes, the degeneracy can be handled by simply ignoring it and going further as usual. The problem here is that getting an optimal solution without cycling cannot be guaranteed. Other methods of resolving degeneracy are

(i) Charne's perturbation method

(ii) Generalized simplex method.

(i) Charne's Perturbation Method

We consider a linear programming problem:

$$\text{Maximize } Z = C^T X$$
$$\text{subject to } AX = b \qquad \ldots (14)$$
$$X = 0,$$

Let $B = (\beta_1, \beta_2, \ldots, \beta_m)$ be the basis at any iteration and the choice of outgoing vector is not unique. Then the solution at this iteration will be

$$X_B = B^{-1} b \qquad \ldots (15)$$

We replace $b \geq 0$ by $b(\epsilon)$ given by

$$b(\epsilon) = b + \sum_{j=1}^{n} \alpha_j \epsilon^j \qquad \ldots (16)$$

where

$$\sum_{j=1}^{n} \alpha_j \epsilon^j$$

is a polynomial in ϵ with vector coefficients of α_j. The number ϵ is taken to be positive and very small. If ϵ_{max} is the maximum value of ϵ, then we may assume $0 < \epsilon < \epsilon_{max}$.

Thus the new perturbed linear programming problem is given by

$$\text{Maximize } Z = C^T X$$
$$\text{subject to } AX = b(\epsilon) \qquad \ldots (17)$$
$$X \geq 0,$$

Now, in order to solve linear programming problem (14), we first of all solve perturbed linear programming problem (17) and then put $\epsilon=0$ in the final solution of (17) to get the solution of linear programming problem (14).

If $X_B(\epsilon)$ is the basic feasible solution corresponding to the basis of problem (17), then

$$X_B(\epsilon)=B^{-1}\left(b+\sum_{j=1}^{n}\alpha_j\epsilon^j\right)$$

$$ie\ X_B(\epsilon)=B^{-1}+\sum_{j=1}^{n}B^{-1}\alpha_j\epsilon^j=X_B+\sum Y_j\epsilon^j \qquad \ldots (18)$$

Since $B^{-1}\alpha_j = Y_j$

$$X_{Bi}(\epsilon)=X_{Bi}+\sum_{j=1}^{n}Y_{ij}\epsilon^j \qquad \ldots (19)$$

$i = 1, \ldots, m$

Now if non of the x_{Bi} is zero then ϵ can be chosen small enough so that $x_{Bi}(\epsilon) > 0$ for $i = 1, 2, \ldots, m$. On the other hand if $X_{Bi} = 0$ for some i, then in order to make $X_{Bi}(\epsilon)$ non-degenerate, we will rearrange the simplex table as follows:

We first write unit vectors and then the other vectors. If $Y_1, Y_2, \ldots, Y_m$ denote the columns of newly arranged tableau then we have

$$Y_1=e_1,\ Y_2=e_2,\ \ldots,\ Y_m=e_m$$

where e_i is the unit vector having unity at the i^{th} place.

$$X_B(\epsilon) = X_B + \sum_{j=1}^{n} Y_j \epsilon^j$$

$$= X_B + \sum_{j=1}^{m} e_j \epsilon^j + \sum_{j=m+1}^{n} \bar{Y}_j \epsilon^j$$

$$i.e \quad X_{Bi}(\epsilon) = X_{Bi} + \epsilon^j + \sum_{j=m+1}^{n} \bar{y}_{ij} \epsilon^j \qquad \ldots (21)$$

where y_{ij} is the component of new Y_j.

Now it follows from (21) that even if $x_{Bi} = 0$, $x_{Bi}(\epsilon) > 0$, $i = 1, 2, \ldots, m$ which is non-degenerate basic feasible solution.

Computational Procedure

If α_k is the vector entering the basis and

$$Min\left\{\frac{x_{Bi}}{y_{ik}}, y_{ik} > 0\right\}$$

is not unique, then the following procedure is adopted.

Step 1

Renumber the columns of the simplex tableau starting with columns in the basis.

Step 2

$$If\ Min\left\{\frac{x_{Bi}}{y_{ik}}, y_{ik} > 0\right\} \qquad occurs\ at$$

$i = i_1, i_2, \ldots, i_s$, then put

$$I_1 = \{i_1, i_2, \ldots, i_s\}$$

and compute

$$\underset{i \in I_1}{Min}\left\{\frac{X_{Bi}}{Y_{ik}}, Y_{ik}>0\right\}$$

If this minimum is unique, then delete the corresponding vector from the basis, otherwise go to step 3.

Step 3 Compute

$$\underset{i \in I_2}{Min}\left\{\frac{X_{Bi}}{Y_{ik}}, Y_{ik}>0\right\}$$

where I_2 is the set of all those values $i \in I_1$ for which there is a tie in step 2. Obviously $I_2 \subseteq I_1$.

If this minimum is unique then delete the corresponding vector from the basis, otherwise go to step 4.

Step 4 Compute

$$\underset{i \in I_3}{Min}\left\{\frac{X_{Bi}}{Y_{ik}}, Y_{ik}>0\right\}$$

where I_3 is the set of those values $i \in I_2$ for which there is a tie in step 3. Clearly $I_3 \subseteq I_2 \subseteq I_1$.

Continuing in this way, we shall get unique vector to be deleted from the basis.

For details about the Generalized Simplex Method, see Dantzig [6].

CHAPTER TWO

2.0 Duality in Linear Programming

2.1 Introduction

The fundamental notion of duality in linear programming was introduced by Von J. Neumann (in a conversation with Dantzig in October 1947) and appeared implicitly in a working paper he wrote few weeks later. Subsequently, Gale, Kuhn and Tucker formulated explicit duality theorems which they proved using the classical lemma of Farkas.

A systematic presentation of theoretical properties of dual linear programs can also be found in Goldman and Tucker [14] and Gale [12].

Every linear programming problem is associated with another linear programming problem called the dual of the problem. The original problem is called "primal" while the other is called its "dual". If the optimal solution of either problem (primal or its dual) is known, then the optimal solution of the other is also available and the values of the two objective functions (Z_P and Z_D) are equal.

2.2 Symmetric Dual Problems

There is a symmetric relation between a dual and its primal.

Consider a linear programming problem stated as follows: Find real numbers x_1, x_2, ..., x_n which

$$\text{Maximize } Z_p = C^T X.$$
$$\text{subject to } AX \le b$$
$$X \ge 0.$$

The dual of the above linear programming problem is obtained by

(i) Transposing the constraint matrix

(ii) Interchanging the role of the components of the requirement vector and the coefficient of the decision variables in the objective function of the primal problem.

(iii) Reverting the inequalities.

(iv) Minimizing the objective function instead of maximizing it.

In matrix notation, the above linear programming problem and its dual can be written as follows:

Find a column vector X, which

$$\text{Maximize } Z_p = C^T X$$
$$\text{subject to } AX \le b \qquad \ldots (22)$$
$$X \ge 0 \quad \text{and}$$

Find a column vector w, which

$$\text{Minimize } Z_D = b^T w$$
$$\text{subject to } A^T w \ge C \qquad \ldots (23)$$
$$w \ge 0 \quad \text{respectively.}$$

where $w = [w_1, w_2, \ldots, w_m] \in \mathbb{R}^m$, A is an m x n matrix, $b \in \mathbb{R}^m$, $C \in \mathbb{R}^n$.

2.3 Unsymmetric Dual Problem

If we have a linear programming problem in standard form as:

$$\text{Maximize } Z_p = C^T X$$
$$\text{subject to } AX = b \qquad \ldots \quad (24)$$
$$X \geq 0,$$

then the dual of this problem (24) can be written as follows:

Find $w \in \mathbb{R}^m$ which

$$\text{Minimize } Z_p = b^T w$$
$$\text{subject to } A^T w \geq C \qquad \ldots \quad (25)$$
$$w \text{ is unrestricted in sign}$$

2.4 Dual of a mixed system

If a system consist of a mixture of equations, inequalities (in either direction), non negative variables or unrestricted variables, then the dual of the problem can be obtained by reducing the problem to the form (22). For this, the following procedure is adopted.

(a) If a constraint has a sign $\geq$, then multiply both sides by -1 and make the sign $\leq$.

(b) If a constraint has a sign =, then it is replaced by two constraints involving the inequalities going in opposite directions.

For example, an equation

$$\sum_{j=1}^{n} a_{ij}x_j = b_i$$

is replaced by two constraints (inequalities)

$$\sum_{j=1}^{n} a_{ij}x_j \leq b_i$$

and

$$\sum_{j=1}^{n} a_{ij}x_j \geq b_i$$

The second inequality can be written as

$$-\sum_{j=1}^{n} a_{ij}x_j \leq -b_i$$

(c) Every unrestricted variable is replaced by the difference of two positive variables.
Do also note that the dual variables which correspond to primal equality constraints must be unrestricted in sign and those associated with the primal inequalities must be non-negative.

A linear programming problem is said to be in standard primal form if

(i) all the constraints involve the sign ≤ if it is a problem of maximization or

(ii) all the constraints involve the sign $\geq$ if it is a problem of minimization.

2.5 Fundamental properties of dual problems

The fundamental properties are in the form of generalized results as follows:

Theorem 11

The dual of the dual of a given primal problem is the primal itself.

(Proof can be found in Hadley 17).

Theorem 12

If X is any feasible solution to the primal problem

$$\text{Maximize } Z_p = C^T X$$
$$\text{subject to } AX \leq b$$
$$X \geq 0$$

and w is feasible solution to the dual problem

$$\text{Minimize } Z_p = b^T w$$
$$\text{subject to } A^T w \geq C$$
$$w \geq 0$$

then $C^T X \leq b^T w$.

Proof

Let $X = [x_1, x_2, \ldots, x_n]$ and $w = [w_1, w_2, \ldots, w_m]$ be feasible solution to the primal problem

$$\text{Maximize } ZD = C^T X$$
$$\text{subject to } AX \leq b \qquad \ldots \quad (26)$$
$$X \geq 0$$

and dual problem:

$$\text{Minimize } Z_P = b^T w$$

$$\text{subject to } A^T w \geq C \quad \ldots \quad (27)$$

$$w \geq 0$$

Now, we have $AX \leq b$ ... (28)

Also w is a feasible solution to dual problem (27). So on multiplying both sides of (28) by w^T, we get

$$w^T(AX) \leq (b^T w)^T$$

This implies $(A^T w)^T X \leq (b^T w)^T$ ... (29)

Also we have

$$A^T w \geq C \quad \ldots \quad (30)$$

Since X is a feasible solution to the problem (26), on multiplying both sides of (30) by X^T, we get

$$X^T(A^T w) \geq X^T C$$

ie $X^T(w^T A)^T \geq (C^T X)^T$

ie $[(w^T A)X]^T \geq (C^T X)^T$

ie $(w^T A)X \geq C^T X$ ... (31)

Now, from (29) and (31) we have

$$C^T X \leq (A^T w)^T X \leq (b^T w)^T = b^T w$$

This implies $C^T X \leq b^T w$.

Theorem 13

If X is a feasible solution to the primal problem

$$\text{Maximize } Z_p = C^T X$$

$$\text{subject to } AX \leq b$$

$$X \geq 0$$

and w is a feasible to the dual problem

$$\text{Minimize } Z_p = b^T w$$
$$\text{subject to } A^T w \geq C$$
$$w \geq 0$$

such that $CT = b^T w$, then X and w are optimal solutions. (Proof can be found in Gupta [16], Hadley [17]).

Theorem 14 (Fundamental duality theorem)

If a finite optimal feasible solution exists for the primal, then there exists a finite optimal feasible solution for the dual and conversely. The values of the two objective functions are equal.

Proof

We consider a pair of primal-dual problem as follows:

$$\text{Maximize } ZD = C^T X$$
$$\text{subject to } AX \leq b \qquad \ldots (32)$$
$$X \geq 0$$

and

$$\text{Minimize } Z_p = b^T w$$
$$\text{subject to } A^T w \geq C \qquad \ldots (33)$$
$$w \geq 0$$

Now, introducing slack vector $X_s = (X_{s1}, X_{s2}, \ldots, X_{sm})$ the primal problem (32) can be restated as follows:

$$\text{Maximize } Z_p = C^T X$$
$$\text{subject to } AX + IX_s = b \qquad \ldots (34)$$
$$X \geq 0,\ X_s \geq 0$$

Let X_B be the finite optimal feasible solution to the primal problem (32) if B denotes the basis matrix corresponding to the feasible solution X_B. Let $C_B = (C_{B1}, C_{B2}, \ldots, C_{Bm})$ be the vector containing the prices of basic variables. Since X_B is the optimal solution to the primal problem,

$\therefore \quad C_j - Z_j \leq 0$

ie., $C_j - C_B Y_j \leq 0$

$C_j - C_B B^{-1} \alpha_j \leq 0$ for every α_j including the surplus variables

ie., $C_B B^{-1} \alpha_j \geq C_j \qquad \ldots \quad (35)$

This implies $C_B \; B^{-1} \; (\alpha_1, \alpha_2, \ldots, \alpha_n) \geq (C_1, \ldots, C_n)$

which implies $C_B \; B^{-1} A \geq C^T \qquad \ldots \quad (36)$

Now, if we write $C_B B^{-1} = (\hat{w})^T = [w_1, w_2, \ldots, w_m]^T$ then it follows from (36) that

$(\hat{w})^T A \geq C^T$

ie $\quad [(\hat{w})^T A]^T \geq (C^T)^T$

ie $\quad A^T \hat{w} \geq C$

This means $\hat{w}$ satisfies the constraint of dual problem (33).

We again consider the equation (35) with α_j corresponding to slack variables. So in this case $C_j = 0$. Hence it follows from (35) that

$C_B B^{-1} e_j \geq 0, \qquad j = 1, 2, \ldots, m$

or $\quad [w_1, w_2, \ldots, w_m]^T e_j \geq 0$

or $\quad w_j \geq 0$

This means $\hat{w} = [w_1, w_2, \ldots, w_m]$ is the feasible solution to the dual (33).

$$
\begin{aligned}
\text{Now } Z_D = b^T\hat{w} &= [(\hat{w})^T b]^T \\
&= (\hat{w})^T b \\
&= (C_B\ B^{-1})\, b \\
&= C_B\ (B^{-1}b) \\
&= C_B X_B \\
&= Z_P
\end{aligned}
$$

Hence it follows that $\hat{w}$ and X_B are feasible solutions of the dual (33) and the primal (32) respectively such that

$$Z_p = Z_D$$

Next, we discuss the complementary slackness theorem which is very important in a dual system.

Theorem 15 (Complementary slackness theorem)

For the optimal feasible solutions of the primal and dual systems,

(a) Whenever inequality occurs in the i^{th} relation of either system, if the corresponding slack or surplus variable is positive, then the i^{th} variable of its dual is zero.

(b) If the j^{th} variable is positive in either system, the j^{th} relation of its dual holds as a strict equality (ie. the corresponding slack or surplus variable w_{m+j} vanishes).

Alternatively, the above theorem can be stated as follows:

A necessary and sufficient condition for any pair of feasible solutions to the primal and dual to be optimal is that

$$w_i\ x_{n+1} = 0 \qquad i = 1, 2, \ldots, m$$

where x_{n+i} is the slack variable in the primal and

$$x_j\ w_{m+j} = 0 \qquad j = 1, 2, \ldots, n$$

where w_{m+j} is the surplus variable for the dual.

Proof

The primal and dual problems in explicit form after introducing the non negative slack variables x_{n+1}, x_{n+2}, ..., x_{n+m} in primal constraints and introducing non negataive surplus variables w_{m+1}, w_{m+2}, ..., w_{m+n} in the dual constraints can be written as follows:

$$\text{Maximize } Z_p = C^T X \qquad \ldots \quad (37)$$

$$\text{subject to } AX + IX_s = b \qquad \ldots \quad (38)$$

$$X \geq 0$$

$$\text{Minimize } Z_D = b^T w \qquad \ldots \quad (39)$$

$$\text{subject to } A^T w - Iw_t = C \qquad \ldots \quad (40)$$

$$w \geq 0$$

Multiplying equations of (38) by w_1, w_2, ..., w_m respectively and then adding we have

$$X_1 \sum_{I=1}^{M} a_{i1} W_i + X_2 \sum_{I=1}^{M} a_{i2} W_i + \ldots + X_n \sum_{I=1}^{M} a_{\epsilon} W_i$$

$$= \sum_{I=1}^{M} b_i W_i \qquad \ldots (41)$$

Now, subtracting (41) from (37), we get

$$\left(C_1 - \sum_{I=1}^{M} a_{i1} W_i\right) X_1 = \left(C_2 - \sum a_{i2} W_i\right) X_2 + \ldots + \left(C_n - \sum a_{\epsilon} W_t\right) X_n$$

$$- X_{n+1} W_1 - X_{n+2} W_2 + \ldots - X_{n+m} W_M$$

$$= \bar{Z}_P - \sum_{i=1}^{M} b_i W_i$$

$$= Z_P - Z_D \qquad \ldots (42)$$

It now follows from (40) that

$$- W_{m+j} = C_j - \sum_{i=1}^{M} a_{ij} W_i \qquad \ldots (43)$$

Using (43) in (42), we have

$-(w_{m+1}\ x_1 + w_{m+2}\ x_2 + \ldots + w_{m-n}\ x_n)$

$-(w_1\ x_{n+1} + w_2\ x_{n+2} + \ldots + w_m\ x_{n+m}) = Z_p - Z_D \qquad \ldots (44)$

If $X^* = (x^*_1, x^*_2, \ldots, x^*_n)$ and $w^* = (w^*_1, w^*_2, \ldots, w^*_m)$ be the optimal solutions to the primal and dual problems with objective function values Z^*_p and Z^*_D respectively, then it follows from fundamental theorem on duality that

$$Z^*_p = Z^*_D \qquad \ldots (45)$$

and we get

$(w^*_{M\mu x}{}^*_{1+} W^{*m+2} x^*_2 + \ldots + w^*_{m+nx*n}) + (w^*_1\ x^*_{n+1} + w^*_2 x^*_{n+2} + \ldots + w^*_n x_{n+m}) = 0$

Since $x^*_i \geq 0$, $w^*_i \geq 0$, this is possible only when

$w^*_{m+j} x^*_j = 0 \qquad$ for every $j = 1, 2, \ldots, n \ \ldots \ (46)$

$w^*_i x^*_{n+i} = 0 \qquad$ for every $i = 1, 2, \ldots, n \ \ldots \ (47)$

From (47), if $x^*_{n+i} > 0$ then we must have $w^*_i = 0$ i.e. if the slack variable in the i^{th} relation of the prima is positive then the i^{th} variable in the dual is zero. Again from (46), if $w^*_{m+j} > 0$ then we must have $x^*_j = 0$ i.e. if the surplus variable in the j^{th} relation of the dual is positive then the j^{th} variable of the primal is zero. This proves part (a) of the theorem.

From (46), if $x^*_j > 0$ then $w^*_{m+j} = 0$ i.e. if j^{th} variable in the primal is >0 then j^{th} relation in dual is strict equality (as $w^*_{m+j} = 0$). From (47), if $w^*_i > 0$ then $x^*_{n+1} = 0$ i.e. the i^{th} relation in the primal is a strict equality (as $x^*_{n+1} = 0$). This proves part (b) of the theorem.

2.6 The Dual Simplex Algorithm

The dual simplex algorithm was discovered by C.E. Lemke [23], a student of Charne while applying the simplex method to the dual of a linear programming problem.

The algorithm starts with an optimal solution (not necessarily feasible) and tries to attain feasibility. That is, it decreases the number of negative variables iteratively. Computationally, let the primal problem in standard form be written as

$$\begin{aligned} &\text{Maximize } Z_p = C^T X \\ &\text{subject to } AX = b \qquad \ldots \quad (48) \\ &\qquad\qquad X \geq 0 \end{aligned}$$

and its dual as

$$\begin{aligned} &\text{Minimize } Z_D = b^T w \qquad \ldots \quad (49) \\ &\text{subject to } A^T w \geq C \\ &\qquad\qquad w_i \text{ unrestricted in sign.} \end{aligned}$$

After introducing the surplus variables to the problem (49), it can be restated in the form used in the application of the simplex method as follows:

$$\text{Minimize } Z_p = b^T w$$
$$\text{subject to } A^T w - I w_s = C \qquad \ldots \quad (50)$$
$$w_s \geq 0$$

where $w_s = [w_{s1}, w_{s2}, \ldots, w_{sn}]$ is the column vector of surplus variables.

Let X be any solution (not necessarily feasible) to AX = b and let w be any soluton to the dual.

Then from (50), we have

$$X^T A^T w - X^T w_s = X^T C \qquad \ldots \quad (51)$$

If $X^T w_s = 0$ then from (51) we have

$$X^T A^T w = X^T C$$
$$(AX)^T w = X^T C$$
$$b^T w = X^T C$$

Then for any solution to the primal and any solution to the dual, the objective functions for both problems are equal if $X^T w_s = 0$.

Interatively, the dual simplex algorithm operates as follows:

Step 1

Convert the minimization linear programming problem into that of maximization if it is in the minimization form. Convert the ≥ type inequalities, representing the constraints of the given linear programming problem if any, into ≤ type by multiplying the corresponding constraints by -1.

Step 2

Introduce slack variables in the constraints of the given problem and obtain an initial basic feasible solution. Put this solution in the starting dual simplex table. The dual simplex table is similar to the simplex method table.

Step 3

Test the nature of $\Delta_j = C_j - Z_j$ in the starting simplex table. ie.

(i) if all $C_j - Z_j \leq 0$ for every j and all $x\beta_i$ are non-negative for all i, then an optimal basic feasible solution has been obtained.

(ii) If $\Delta_j \leq 0$ for every j and at least one basic variable, say x_{Bi} is negative then go to step 4.

Step 4

Select the most negative of x_{Bi}'s. The corresponding basis vector then leaves the basis set Y_B. Let x_{Bk} be the most negative basic variable so that y_k leaves Y_B.

Step 5

Test the nature of Y_{kj}, j=1 2,...,n.

(i) If all Y_{kj} are non-negative, there does not exist any feasible solution to the given problem.

(ii) If at least one y_{kj} is negative, compute the replacement ratios.

Leaving vector

$y_k = \text{Min } \{x_{Bi}, x_{Bi} < 0\}$ = key row.

Entering vector

$$\alpha_k = Min\left\{\frac{\Delta_j}{elements\ of\ key\ row}\right\} = key\ column.$$

Step 6

Test the new iterated dual simplex table for optimality.

Repeat the procedure until either an optimum feasible solution has been obtained (in a finite number of steps) or there is an indication of the non existence of a feasible solution.

Note that the sole advantage of the dual simplex algorithm over the simplex method of solving a linear programming problem is that, here we do not require any artificial variables. Hence much labour is saved whenever this method is applicable.

2.7 The Primal Dual Algorithm

The primal dual algorithm works simultaneously on the primal and dual problems. It operates as follows:

(a) We start with a solution to the dual problem. That is, we first of all determine an initial dual solution.

(b) We then proceed to find an infeasible solution to the primal problem such that the theorem of complementary slackness remains satisfied. The procedure is as follows:

(i) We associate with the initial dual solution a restricted primal problem which is deduced from the original primal by zeroing certain variables to satisfy the complementary slackness theorem and by replacing the original linear form by the sum of artificial variables.

(ii) We maximize the sum of the artificial variables by using simplex algorithm.

(iii) In case the optimal solution thus obtained is not optimal solution of the original primal, then we determine a new dual solution and apply step (i) and (ii).

Determining initial dual solution

We consider a pair of primal dual problems (53) and (54) as follows:

$$\text{Maximize } Z_p = C_1x_1 + C_2x_2 + \ldots + C_nx_n$$
$$\text{subject to } AX = b \qquad \ldots \quad (53)$$
$$X \geq 0$$

$$\text{Minimize } Z_D = b_1w_1 + b_2w_2 + \ldots + b_mwm$$
$$\text{subject to } A^Tw \geq C \qquad \ldots \quad (54)$$

Beale [5] suggested a very simple modification of the primal problem (53) which provides an immediate solution to the dual problem.

The procedure is as follow:

Introducing an additional constraint $x_o+X = x_o+x_1+x_2+\ldots+x_n=b_o$ in the given primal (53), we obtain the following new primal called the modifed primal problem.

$$\text{Maximize } Z_p = 0x_o + C_1x_1 + \ldots + C_nx_n = (0,C).[x_o,X]$$
$$\text{subject to } x_o + X = b_o$$

$$Ax = b \qquad \ldots \quad (55)$$

$$[x_o, X] \geq 0$$

Here, it is assumed that the new constraint b_o have arbitrarily large value.

Now, (55) can also be written as

$$\text{Maximize } Z_p = C^T X$$

$$\text{subject to} \begin{bmatrix} 1 & 1 \\ 0 & A \end{bmatrix} \begin{bmatrix} x_o \\ X \end{bmatrix} = \begin{bmatrix} b_o \\ b \end{bmatrix}$$

$$[x_o, X] \geq 0$$

The dual of the modified primal (55) or (56) called the modified dual is as follows:

$$\text{Minimize } Z_D = b_o w_o + b^T w = b_o w_o + b_1 w_1 + \ldots + b_m w_m$$

$$\text{subject to} \begin{bmatrix} 1 & 0 \\ 1 & A \end{bmatrix} \begin{bmatrix} w_o \\ w \end{bmatrix} \geq \begin{bmatrix} 0 \\ C \end{bmatrix} \qquad \ldots (56)$$

which can also be written as

$$\begin{aligned} 1.w_o + 0.w_1 + 0.w_2 + \ldots + 0.w_m &\geq 0 \\ 1.w_o + a_{11}.w_1 + a_{21}.w_2 + \ldots + a_{m1}.w_m &\geq C_1 \\ \vdots \qquad \vdots \qquad \vdots \qquad & \vdots \qquad \vdots \\ 1.w_o + a_{1n}.w_1 + a_{2n}.w_2 + \ldots + a_{mn}.w_m &\geq C_n \end{aligned} \qquad \ldots \quad (57)$$

where w_1, w_2, ..., w_m are unrestricted in sign and by virtue of the first constraint in (57). (57) implies $w_o = 0$, then modified dual reduces to original dual problem (54).

Now, we get an immediate solution to the dual (57) as

$w_o = \text{Max}\{0, C_j\}$, $w=0$ i.e. $w_1 = w_2 = \ldots = w_m = 0$ $j=1,2,\ldots,n$ ...(58)

To solve (57) we add surplus variables w_{sj}, $j=1,2,\ldots,n$ to the constraints of (57) and then the resulting problem is

$$
\begin{aligned}
\text{Minimize} \quad & Z_D = b_o w_o + b_1 w_1 + \ldots + b_m w_m + 0.w_{so} + \ldots + 0.w_{sn} \\
\text{subject to} \quad & 1.w_o + 0.w_1 + 0.w_2 + \ldots + 0.w_m - w_{so} = 0 = C_o \\
& 1.w_o + a_{11}.w_1 + a_{21}.w_2 + \ldots + a_{m1}.w_m \quad -w_{s1} = C_1 \qquad \ldots (59) \\
& \vdots \\
& 1.w_o + a_{1n}.w_1 + a_{2n}.w_2 + \ldots + a_{mn}.w_m \qquad -w_{sn} = C_n \\
& w_{so}, \; w_s \geq 0
\end{aligned}
$$

$w_1, w_2, \ldots, w_m$ are unrestricted in sign.

From (58) and (59), we have

$$
\begin{aligned}
w_{s0} &= w_o - C_o \\
w_{s1} &= w_o - C_1 \\
w_{s2} &= w_o - C_2 \\
&\vdots \\
w_{sn} &= w_o - C_n
\end{aligned}
\qquad \ldots (60)
$$

The solution (58), is the initial dual solution with which we start the primal dual method. Since w_o = Max $(0, C_j)$ $j=1,2,\ldots,n$, it follows that at least one of $w_{s0}, w_{s1}, \ldots, w_{sn}$ is zero.

From equation (51) in the dual simplex algorithm, we know that if $X^T w_s = 0$ or equivalently for dual problems (56) and (59), if

$$\begin{bmatrix} X_o \\ X \end{bmatrix}^1 \begin{bmatrix} W_{SO} \\ W_S \end{bmatrix} = 0$$

$$\text{or } x_o.w_{so} + Xw_s = 0 \qquad \ldots (61)$$

then $Z_p = Z_D$ where X is any solution (not necessarily feasible) to the primal.

If along with the condition (61) $[x_o, X]$ is a feasible solution to the modified primal (55) then $[x_o, X]$ is an optimal feasible solution to the modified primal and $[w_o, w]$ is an optimal solution to the modified dual.

If in addition $w_o = 0$, then $Z_p = C^TX = Z_D = b^Tw$ and from (61) $X^Tw_s = 0$. Consequently X is an optimal solution to the given primal and w is an optimal solution to the modified dual. If we get a solution to the modified dual with $w_o \neq 0$ and a feasible solution to the modified primal such that (61) is satisfied then it follows that $x_o = 0$ and then $Z_p = Z_D = b_o w_o + b^T w$.

To determine the restricted primal problem

To solve the modified primal problem (56), we add artificial variables to each of the constraints and assign price -1 to each artificial variable and zero prices coresponding to every legitimate variable. We thus obtain the following primal problem:

$$Maximize\ Z^* = -1x_{a0} - 1x_{a1} - 1x_{a2} + \ldots - 1x_{am} = -1x_{a0} - 1X_a$$

$$subject\ to \begin{bmatrix} 1 & 1 \\ 0 & A \end{bmatrix} \begin{bmatrix} x_o \\ X \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & I_M \end{bmatrix} \begin{bmatrix} x_{ao} \\ X_a \end{bmatrix} = \begin{bmatrix} b_o \\ b \end{bmatrix}$$

and $\qquad [x_o, X] \geq 0, \; [x_{ao}, X_a] \geq 0$

Now, we apply simplex method on the above restricted primal and go on improving Z^* till $Z^* = 0$, which follows that $x_{ao} = 0 = x_{a1} = \ldots = x_{am}$.

The criteria for the selection of the entering and leaving vector will be different from that in the regular simplex method. We shall make use of the following notations as we proceed further.

(a) (m+1) component legitimate activity vectors will be denoted by $Y_j^{(1)}$ j = 1, 2, ..., n.

(b) a basis matrix for the constraints will be denoted by B_1.

(c) the vector containing the price C^*_j of the variables in the basis will be denoted by C^*_{Bj}.

(d) $\Delta^*_j = C^*_j - Z^*_j = C^*_j - C^*_{Bj}B^{-1}\,\alpha_j^{(1)} = C^*_j - C^*_{Bj}\;Y_j^{(1)}$.

Entering vectors

We know that $w_o = \max\{0, C_j\}$, $w = 0$ j=1,2,...,n, which implies that at least one component w_{aj} will be zero. The vector $Y_j^{(1)}$ for which the corresponding $w_{aj} = 0$ is taken as the vector entering the restricted primal basis.

Leaving vectors

The vector leaving the basis corresponding to the entering vector $Y_j^{(1)}$ is selected one by one in the same manner as in phase I of ordinary simplex method.

Improved value of Z_D and Z^*

(a) The new dual solution $\hat{w}_{aj}$ is obtained by using the formula

$$\hat{w}_{aj} = w_{aj} - \theta\;\Delta^*_j$$

where

$$\theta = Min\left\{\frac{W_{sj}}{\Delta^*_j},\ \Delta^*_j > 0\right. \qquad \ldots (62)$$

Here also, at least one w_{aj} will be zero and hence the corresponding $Y_j^{(1)}$ will be selected to enter the basis at the next iteration.

(b) The improved value of Z^* will be obtained by the formula

$$Z^{*} = C_{B1}\ X_{B}^{(1)} \qquad \ldots \quad (63)$$

(c) The value of Z_D will be obtained by the formula

$$Z_D = C_D\ \theta Z^k$$

where θ and Z^{*} are given by (62) and (63) respectively.

Test of Optimality

The algorithm terminates with one of the following results:

(i) when $z^{*} = 0$, we get an optimal solution to the primal and $Z_p = Z_D$ provided $w_{s0} = 0 = w_o$.

(ii) when $Z^{*} = 0$ but $w_{s0} \neq 0$, the primal problem has an unbounded solution.

(iii) when $Z^{*} < 0$, $\Delta_j' \leq 0$ for all j (at any iteration), the primal have no solution.

For worked examples using this algorithm, see Gupta [16].

See appendix B for the starting table.

CHAPTER THREE

3.0 INTERIOR POINT ALGORITHMS FOR LINEAR PROGRAMMING

3.1 Introduction

Since Karmarkar's [19] discovery of a polynomial time interior point algorithm for linear programming in 1984, the field of interior point methods has been a major area of research in optimization. The central path concept first proposed in McLinden [27, 28] and later rediscovered by Sonnevend [36] in primal setting and Megiddo [29] in primal dual context has played a major role in the development of many interior point algorithms for linear programming. We carefully look at the various interior point algorithms for linear programming.

3.2 Karmarkar's projective rescaling algorithm

3.3 Introduction

This is a new polynomial-time algorithm for linear programming. It came after the ellipsoid algorithm by Khachian [20] and is much faster than the ellipsoid algorithm so long as computation is concerned. With this algorithm, we prove that given a polytope P and a strictly interior point $a \in P$, there is a projective transformation of the space that maps P, a to P′,a. It consists of repeated application of such projective transformations each followed by optimization over an inscribed sphere to create a sequence of points which converge to the optimal solution.

Now, we informally discuss the main ideas involved in the algorithm before the algorithm itself.

3.4 Problem Definition

Karmarkar's algorithm works on a linear programming problem of the form

$$\text{Minimize } C^T X$$

$$\text{subject to } AX = 0 \qquad \ldots \quad (64)$$

$$\Sigma x_i = 1 \qquad i=1,2,\ldots,n$$

$$X \geq 0$$

where $X = \{x_1, x_2, \ldots, x_n\}^T \in \mathbb{R}^n$, $C \in Z^n$, $A \in Z^{m\times n}$.

This form of the problem is called canonical form which can be obtained from standard form of linear programming using projective transformation.

Let Ω denote the affine subspace

$\{X \in \mathbb{R}^n : AX = 0\}$

Let Δ denote the simplex

$\{X \in \mathbb{R}^n : X \geq 0, \ \Sigma x_i = 1\}$

Let P denote the polytope defined by $P = \Omega \cap \Delta$.

The problem (64) can again be reduced to the form

Minimize $C^T X$

subject to $X \in P$

3.5 Bounds on the objective function

Let a_o be a strictly interior point in the polytope P. Suppose we draw an ellipsoid E with centre a_o that is contained in the polytope and solve the optimization problem over the restricted region E instead of P. To derive a bound another ellipsoid E' is created by magnifying E by a sufficiently large factor ν so that E' contains P.

This implies $E \subseteq P \subseteq E'$; $E' = \nu E$

Let f_E, f_P, f_E denote the minimum values of the objective function f(x) on E, P and E′ respectively.

$f_E \geq f_P \geq f_E$

ie $f(a_o) - f_E \leq f(a_o) - f_P \leq f(a_o) - f_E = \nu[f(a_o) - f_E]$

The above inequality is as a result of linearity of f(x). Considering the last two inequalities we have

$$\frac{f(a_o) - f_P}{f(a_o) - f_E} \leq \nu$$

subtracting both sides from 1, we have

$$1 - \frac{f(a_o) - f_E}{f(a_o) - f_P} \leq 1 - \frac{1}{\nu}$$

which implies

$$\frac{f_E - f_P}{f(a_o) - f_P} \leq \left(1 - \frac{1}{\nu}\right)$$

Thus by going from a_o to the point a, that minimizes the objective function f(x) over E, we come closer to the minimum value of the objective function by a factor $(1 - \nu^{-1})$. The same process could be repeated with a as a centre. The rate of convergence of this method depends on ν which is equivalent to the dimension of the space. The smaller the value of ν, the faster the convergence.

3.6 Projective transformation

Projective transformation of $\mathbb{R}^n$ are described by formulae of the form

$$X \to \frac{DX + b}{f^T X + d}$$

where $D \in \mathbb{R}^{n x n}$, $f, b \in \mathbb{R}^n$, $d \in \mathbb{R}$ and the matrix

$$\begin{bmatrix} D & b \\ f^T & d \end{bmatrix} \text{ is non-singular.}$$

What we actually require is projective transformation of the hyperplane $\Sigma = \{X \in \mathbb{R}^n | \Sigma x_i = 1\}$ which are described by

$$X \to \frac{Dx}{e^T DX}$$

where $e = (1,1,\ldots,1)^T$ and D is a non-singular n x n matrix. For each interior point say $a \in \Delta$, there is a unique projective transformation $T(a, a_o)$ of the hyperplane Σ moving $a=(a_1, a_2, \ldots, a_n)^T$ to the centre

This transformation is thus given by

$$X^1 = T(X) = \frac{D^{-1}X}{e^T D^{-1} X}$$

The corresponding matrix D is now diagonal. That is $D=diag[a_1, a_2, \ldots, a_n]$ and $T(a, a_o)$ has the following properties.

(1) T is one-to-one and maps the simplex onto itself. Its inverse is given by

$$x_i = \frac{a_i x_i^1}{\sum_j a_j x_j^1}$$

$$i = 1, 2, \ldots, n.$$

(2) Each facet of the simplex given by $x_i = 0$ is mapped onto the corresponding facet $x_i^1 = 0$.

(3) Let A_i denote the i^{th} column of A.

ie. $A = [A_1, A_2, \ldots, A_n]$

The system of equations $\Sigma A_i x_i = 0$ now becomes

$$\frac{\sum A_i a_i x_i^1}{\sum_j a_j x_j^1} = 0$$

Since

$$x_i = \frac{a_i x_i^1}{\sum_j a_j x_j^1}$$

This implies $\Sigma A_i^1 x_i^1 = 0$

where $A_i^1 = a_i A_i$.

We denote the new affine subspace $\{X' \epsilon \mathbb{R}^{n+1} | A'X'=0\}$ by Ω'.

(4) If $a \epsilon \Omega$ then the centre of the simplex $a_o \epsilon \Omega'$.

(5) Image of the point x=a is the centre of the simplex given by $x'=a_o$.

3.7 Invariant Potential Function

Karmarkar's projective rescaling algorithm generates a sequence of points $X^{(0)}, X^{(1)}, \ldots, X^{(k)} \ldots$ having decreasing values of the objective function. In the k^{th} step, the point $X^{(k)}$ is brought into the centre by a projective transformation.

Then we optimize the objective function over intersection of the inscribed ball and the affine subspace to find the next point $X^{(k+1)}$. From previous results, the objective function is expected to reduce by a factor of $(1-n^{-1})$ at least in each step. Since the set of linear function is not invariant under projective transformation, Karmarkar [19] introduced a potential function as the sum of the logarithms of the ratio of two linear functions which is invariant under projective transformation in order to measure the progress of the algorithm.

$$g(X) = \sum_{j=1}^{n} \ln\left(\frac{C^T X}{x_j}\right) + k$$

where k = constant.

The potential function has the following properties:

(1) Any desired amount of reduction in the value of the objective function can be achieved by sufficient reduction in the value of g(X).

(2) g(X) is mapped into a function of the same form by projective transformation $T(a, a_o)$ described above.

(3) Optimization of g(X) in each step can be done approximately by optimizing a linear function (but a different linear function in different steps).

3.8 Complexity of the algorithm

The value of the objective function is reduced by a constant factor 0(n) steps. As in the ellipsoid algorithm, Karmarkar defined

$$L=\log(1+|D_{max}|) + \log(1+\alpha)$$

where D_{max} = Max $\{|\det(X)|$: X is a square submatrix of constraint matrix A$\}$.

$$\alpha = \max\{|C'_i|,\ |b_i|\ i = 1,2,\ldots,n\}.$$

where $C' = DC$. $D \in \mathbb{R}^{nxn}$ is a diagonal matrix, $C \in \mathbb{R}^n$.

Clearly, L is not greater than the input length

$$l = \left\lceil \sum_{i=1}^{m}\sum_{j=1}^{n} \log_2(|a_{ij}|+1) + \sum_{i=1}^{m} \log_2(|b_i|+1) + \sum_{j=1}^{n} \log_2(|C_j|)+1) + \log_2 mn \right\rceil +$$

where $\lceil . \rceil$ is a celling function (celling function of a real number P is the smallest interger greater than or equal to P).

It has been claimed in Karmarkar [19] that his algorithm is better than that of Khachian by a factor of $0(n^{2.5}L)$. This is however based on Khachian's earlier paper [20]. But in his revised paper Khachian [21], the complexity of his algorithm is given as $0(n^4L)$. Thus theoretically, the improvement of Karmarkar's algorithm over Khachian's is $0(n^{.5}L)$.

3.9 The Main Algorithm

There are some important points we have to note before the algorithm.

(1) The feasible region is the intersection of an affine space with a simplex rather than the positive orthant.

(2) The linear system of equations defining Ω is homogeneous.

(3) The target minimum value of the objective function is zero.

The algorithm operates in such a way that it generates a sequence of points $X^{(0)}$, $X^{(1)}, \ldots,$ $X^{(k)}$ in the following steps:

Step 1 Initialize

$X^{(0)}$ = centre of the simplex = e/n

Step 2

Compute next point in the sequence; $X^{(k+1)} = \phi(X^{(k)})$.

Step 3

Check for infeasibility.

If feasible, go to step 4. If not stop and conclude that the objective function is strictly positive which implies the problem is unbounded.

Step 4

Check for optimality.

If optimality is not achieved, go to step 2.

Now, we describe the various steps in details.

Step 2

Let $b = X^{(k+1)}$ and $a = X^{(k)}$.

The function $b = \phi(a)$ is defined by the following sequence of operations:

Let $D = \text{diag}(a_1, a_2, \ldots, a_n)$ be a diagonal matrix whose i^{th} diagonal entry is a_i.

Substep 1

Let

$$B = \begin{bmatrix} AD \\ e^T \end{bmatrix}$$

That is, the matrix AD is augmented with a row of all 1's. This guarantees that ker B (null space of B) is contained in the hyperplane.

Substep 2

Compute the orthogonal projection of DC into the null space of B.

ie. $C_p = [I - B^T (BB^T)^{-1}B]DC$

Substep 3

$$\hat{C} = \frac{C_p}{\|C_p\|}$$

ie c is the unit vector in the direction of C_p.

Substep 4

$$b' = a_o - \alpha r c.$$

ie. take a step of length α in the direction c where

$$r = \frac{1}{\sqrt{n(n-1)}}$$

is the radius of the largest inscribed sphere in the simplex Δ and $\alpha \in (0,1)$ is a parameter which Karmarkar suggests to set equal to ¼.

Substep 5

Apply inverse projective transformation to b^1.

$$\textbf{ie} \qquad b = \frac{Db^1}{e^T Db^1}$$

Return b.

Step 3

To check for infeasibility, we defined a "potential function" by

$$g(X) = \sum_j \ln \frac{C^T X}{x_j}$$

We expect certain improvement δ in the potential function at each step. The value of δ depends on the choice of the parameter α. For example, if α=¼, then δ = ⅛ = 1/8 according to Karmarkar. If this expected improvement is not observed, that is, if $f(X^{K+1}) > f(X^k) - \delta$ then we stop and say that the minimum value of the objective function must be strictly positive. In such case, we conclude that the standard form linear programming problem from which we got our canonical form does not have finite optimal solution. That is, it is either infeasible or unbounded.

Step 4

The algorithm stops when we meet the required convergence check.

$$\frac{C^T X}{C^T a_o} \le 2^{-q}$$

where q is a positive real number. Based on fixed potential function decrease $\delta(\alpha) = \ln 2$ at each iteration as achieved in McDiamid [26] and Anstreicher [1], Singh and Singh [34,35] observe that the algorithm will converge to an optimal solution in at most k=nq iterations. For linear convergence of this algorithm, see Singh [35].

3.10 Transformation of a standard linear programming problem to canonical form

Consider the standard linear programming problem

$$\text{Minimize } C^T X$$
$$\text{subject to } AX = b$$
$$X \ge 0$$

where $C \in Z^n . A \in Z^{mxn}$, $b \in Z^m$.

Let X = a be a strictly interior point of the polytope P and the target minimum value of the objective function we are interest in is zero.

Consider the transformation

$X' = T(X)$ where $X \in \mathbb{R}^n$, $X' \in \mathbb{R}^{n+1}$ are defined by

$$x_i^1 = \frac{x_i | a_i}{\sum_j x_j | a_j + 1}$$

i=1,2,...,n.

$x'_{n+1} = 1 - \Sigma x'_i$ i=1,2,...,n.

Let $P_+ = \{X \in \mathbb{R}^n : X \geq 0\}$ denote the positive orthant, and

$$\Delta = \left\{ X' \in \mathbb{R}^{n+1} / X' \geq 0, \sum_{i=1}^{n+1} x_i^1 = 1 \right\} \quad \textit{denote the simplex}$$

The transformation T(X) has the following properties:

(1) It is one-to-one and onto. The inverse transformation is given by

$$x_i = \frac{a_i x_i^1}{x_{n+1}^1} \qquad i=1,2,...,n.$$

(2) The image of point a is the centre of the simplex.

(3) Let A_i denote the i^{th} column of A, then AX = b can be written as.

$$\sum_{i=1}^{n} A_i x_i = b$$

$$ie. \quad \frac{\sum A_i a_i x_i^1}{x_{n+1}^1} = b$$

$$ie. \quad \sum_{i=1}^{n} A_i a_i x_i^1 - b x_{n+1}^1 = 0$$

$$ie. \quad \sum_{i=1}^{n+1} A_i x_i^1 = 0$$

where $A^1_i = (a_i A_i)$ $\qquad i=1,2,\ldots,n.$

$A_{n+1} = -b.$

If Ω' denotes the affine subspace corresponding to the matrix A' then

$$\Omega' = \{X' \in \mathbb{R}^{n+1}: A'X'=0\}.$$

Thus we get a system of homogeneous equations in the transformed space. Since we are interested in target value of zero, we define Z to be the affine subspace corresponding to the zero set of the objective function in the original space.

$$Z = \{X \in \mathbb{R}^n : C^T X = 0\}$$

substituting

$$x_i = \frac{a_i x_i^1}{n_{n+1}^1} \; in \; \sum_{i=1}^{n} C_i x_i = 0$$

we get

$$\sum_{i=1}^{n} \frac{C_i a_i x_i^1}{x_{n+1}^1} = 0$$

ie. $$\sum_{i=1}^{n+1} C_i a_i x_i^1 = 0$$

ie. $$\sum_{i=1}^{n+1} C_i x_i^1 = 0$$

where $C'_i = a_i C_i$ $\quad i=1,2,\ldots,n$

Then $C^T X = 0$ implies $C'^T X' = 0$. Therefore the image of Z is given by $Z' = \{X' \in \mathbb{R}^{n+1} : C'^T X' = 0\}$.

Thus the transformed problem in canonical form can be stated as follows:

Minimize $C'^T X'$

subject to $A'X' = 0$

$$\sum_{i=1}^{n+1} x_i^1 = 1$$

where $C \in \mathbb{R}^{n+1}$, $A' \in \mathbb{R}^{m \times (n+1)}$.

The center of the simplex is a feasible starting point. Application of the main algorithm to this problem either gives a solution X′ such that $C'^T X' = 0$ or we conclude that minimum value of $C'^T X'$ is strictly positive which implies that the original problem from which we have obtained canonical form is either infeasible or unbounded.

3.11 <u>The Affine Rescaling Algorithm</u>

3.12 <u>Introduction</u>

The affine rescaling algorithm which is a variation on the Karmarkar's algorithm was proposed by a number of authors including Vanderbei, Meketon and Freedman [39] and Barnes [3], soon after the appearance of Karmarkar's work. However, it turned out that the method had been proposed almost twenty years before by a student of Kantorovich Dikin [8,9]. The algorithm has several advantages over Karmakar's original algorithm. It starts directly on linear programming problem in standard form. It is assumed that a point X° is known such that $AX^{\circ} = b$ and $X^{\circ} > 0$. The algorithm produces a Monotone decreasing sequence of values of the objective function and the minimum value of the objective function does not have to be known in advance.

3.13 Derivation of the algorithm

We derive the algorithm following the approach of Barnes [3].

Consdier the linear programming problem of the form

$$\begin{aligned} &\text{Minimize } C^T X \\ &\text{subject to } AX = b \\ &\qquad X \geq 0 \end{aligned} \qquad \ldots \quad (65)$$

where $C, X \in \mathbb{R}^n$, $b \in \mathbb{R}^m$ and $A \in \mathbb{R}^{m \times n}$ is a matrix of rank m. The j^{th} column of A will be denoted by a_j.

We make the following assumptions. The problem (65) has no degenerate basic feasible solutions and its dual

$$\begin{aligned} &\text{Maximize } b^T\lambda \\ &\text{subject to } A^T\lambda \geq C \end{aligned} \qquad \ldots \quad (66)$$

has no degenerate basic solutions. By these assumptions, we mean that b cannot be written as a positive combination of fewer than m columns of A and that at most m of the equations $C_i - a_i^T\lambda = 0$ $i = 1,2,\ldots, n$ can be satisfied simultaneously.

Let $Y = (y_1, y_2, \ldots, y_n)^T$ be a feasible solution of (65) satisfying $y_i>0$ $i = 1,2,\ldots, n$.

If $0 < R < 1$, the ellipsoid

$$\sum_{i=1}^{n} \frac{(x_i - y_i)^2}{y_i^2} \leq R^2 \text{ is in } \mathbb{R}^n \qquad \ldots (67)$$

This is true since if $x_j \leq 0$ for some j, then

$$\sum_{i=1}^{n} \frac{(x_i - y_i)^2}{y_i^2} \geq \frac{(x_j - y_j)^2}{y_j^2} \geq 1 > R^2$$

This implies that a feasible solution of (65) satisfying $C^TX < C^TY$ can be obtained by solving the following problem.

Minimize C^TX

subject to $AX = b$

$$\sum_{i=1}^{n} \frac{(x_i - y_i)^2}{y_i^2} \leq R^2 \qquad \ldots (68)$$

Notice that the constraint $X \geq 0$ has been replaced by (67), which is an easier way to handle the probelm.

To solve (68), let $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_m)^T$ be a vector of Lagrange multipliers corresponding to the constraints AX=b. Let $D = \text{diag}\ (y_1, y_2, \ldots, y_n)$ be a diagonal matrix whose diagonal entries are the components of Y. Since $Y > 0$, D is non-singular. For any X satisfying (68) we have

$$C^TY - C^TX = \{C - A^T\lambda\}^T(Y - X)$$
$$= [D(C - A^T\lambda)]^T D^{-1} (X - Y) \quad \ldots \quad (69)$$
$$\leq ||D(C-A^T\lambda)|| \; ||D^{-1}(X-Y)|| \leq ||D(C-A^T\lambda)||R,$$

The inequalities being obtained from Schwartz inequality and (68) above.

Equality holds throughout (69) if

$$D(C - A^T\lambda) = \gamma D^{-1}(X-Y) \quad \ldots \quad (70)$$

for some constant γ and if $||D^{-1}(X-Y)|| = R$.

These conditions imply that

$$\gamma = \frac{||D(C-A^T\lambda)||}{R}$$

substituting this in (70) gives

$$X = Y - R\,\frac{D^2(C-A^T\lambda)}{||D(C-A^T\lambda)||} \quad \ldots (71)$$

The condition $AX = AY = b$ implies that $AD^2(C-A^T\lambda) = 0$ or

$$\lambda = (AD^2A^T)^{-1}AD^2C \quad \ldots \quad (72)$$

writing (69) as

$$C^TX \geq C^TY - R\,||D(C-A^T\lambda)|| \quad \ldots \quad (73)$$

it can be seen that the minimum is given by the right hand side (73) and is attained when X and λ are given by (71) and (72) respectively. This indeed suggests an algorithm for interatively finding the solution of (65) and is called the affine rescaling algorithm.

Interatively, the algorithm can therefore be formally stated as follows: Let $X^\circ > 0$ satisfying $AX^\circ = b$ be given. In general, if $X^{(k)}$ is known, define

D_k = diag (x_1^k, x_2^k, ..., x_n^k)

and compute $x^{(k+1)} > 0$ by the formula

$$X^{k+1} = X^k - \frac{RD_k^2(C - A^T\lambda_k)}{||D_k(C - A^T\lambda_k)||} \qquad \ldots (74)$$

where $\lambda_k = (AD_k^2A^T)^{-1}AD_k^2C$

Theorem 16

If (65) has a bounded solution, the sequence $\{x^k\}$ defined by (74) converges to a solution of (65) that is an extreme point of the constraint set defined by AX = b, X ≥ 0.

Proof can be found in Barnes [3].

3.14 Convergence of the algorithm

The theorem below best describes the way the algorithm converges.

Theorem 17

Let X^* denote the solution of (65). The sequence $\{x^k\}$ generaged by (74) satisfies

$$C^TX^{k+1} - C^TX^* \leq \left(1 - \frac{R}{\sqrt{n-m+\epsilon_k}}\right)(C^TX^k - C^TX^*) \qquad \ldots (75)$$

where $\{\epsilon_k\}$ is a sequence of positive numbers converging to 0 as $k \to \infty$.

Proof

From the non-degeneracy assumptions in the beginning, we know that

$$X^* = \lim_{k \to \infty} X^k$$

has (n-m) components equal to zero. For simplicity, assume that $X^* = (x^*_1, x^*_2, \ldots, x^*_n, 0, 0, \ldots, 0)$

$$\text{Then } C^TX^k - C^TX^* = [D_k(C-A^T\lambda_k)]^T D_k^{-1} (X^kX^*)$$

$$\leq ||D_k(C-A^T\lambda_k)||(n-m+\epsilon_k)^{1/2}$$

$$= 1/R \ (C^TX^k - C^TX^{k+1})(n-m+\epsilon_k)^{1/2} \qquad \ldots \qquad (76)$$

where

$$\epsilon_k = \sum_{i=1}^{M} \left(\frac{x_i^k - x_i^*}{x_i^k} \right)^2 \rightarrow 0 \ as \ k \rightarrow \infty$$

Thus (76) can be written as

$$R \frac{(C^TX^k - C^TX^*)}{(n-m+\epsilon_k)^{1/2}} \leq C^TX^k - C^TX^{k+1}$$

which implies

$$C^TX^{k+1} - C^TX^* - (C^TX^k - C^TX^*) \leq \frac{-R}{\sqrt{n-m+\epsilon_k}} (C^TX^k - C^TX^*)$$

which is equivalent to (75)

From this theorem, it can be seen that the amount by which the objective function C^TX decreases at each step of the algorithm increases if R is increased. This suggests that at each step of the algorithm, we should increase R as much as possible subject to the condition that all variables remain non-negative.

The next interior point algorithm we have to discuss is the path following algorithm. But before this, it is necessary we discuss the pathways leading to the optimal set in linear programming.

3.15 Pathways to the optimal set in linear programming

Derivation of the path leading to the optimal set makes use of the logarithmic barrier function technique usually used in non linear constrained optimization. This method of applying the logarithmic barrier function technique to the linear programming problem recently came up in [13] where Karmarkar's algorithm [19] was analysed from the barrier function view point. But, the idea of using this function in the context of linear programming is usually attributed to Frisch [11].

We consider the linear programming problem in the standard form

$$\text{Maximize } C^T X$$
$$\text{subject to } AX = b \qquad \ldots \quad (76)$$
$$X \geq 0$$

where A is a constant m x n matrix, $b \in \mathbb{R}^m$ and C, $X \in \mathbb{R}^n$.
The barrier function technique give rise to the following problem.

$$\text{Maximize } C^T X + \mu \Sigma \ln X_j \qquad j=1,2,\ldots,n.$$
$$\text{subject to } AX = b \qquad \ldots \quad (77)$$
$$X > 0$$

where $\mu > 0$ is typically small. It is called the barrier parameter. The barrier function approach is valid only if there exists an $X > 0$ such that $AX = b$.

For any n-dimensional vector X, that is $X=(x_1,x_2,\ldots,x_n)$, let D_x denote the diagonal matrix of order n x n whose diagonal entries are the components of X. A vector $X > 0$ is an optimal solution for (77) if and only if there exists a vector $Y \in \mathbb{R}^m$ such that the following optimality condition is satisfied:

$$\mu D_x^{-1}e - A^T Y = -C \qquad \ldots \quad (78)$$
$$AX = b$$

where e is a vector of 1's of any dimension. To arrive at the above optimality condition, we construct a lagrangian function with Y as a lagrange multiplier and then solve for the stationary points.

The problem (77) may be unbounded. For a moment, let us assume that the feasible domain $\{X \in \mathbb{R}^n: AX = b, X > 0\}$ is bounded. Since the objective function in (77) is strictly concave, at least in this case, both (76) and (77) have optimal solutions, (for every μ). Under the boundedness assumption, (77) has a unique optimal solution for every $\mu > 0$ since its objective function is strictly concave.

Thus under the boundedness assumption the system (78) has a unique solution for X for every $\mu > 0$. If the matrix A is of full row rank $m(m \leq n)$, the value of Y is uniquely determined by the value of X.

Proposition 18

The problem (77) is either unbounded for every $\mu > 0$ or has a unique optimal solution for every $\mu > 0$.

Proof

Consider the interval I of values t for which the set

$$L(t) = \{X \in \mathbb{R}^n: X \geq 0,\ C^T X = t,\ AX = b\}$$

has a nonempty interior. I is obviously an open interval.

If for any $t \in I$ the function $\phi(X) = \Sigma \ln X_j$ is unbounded on $L(t)$, then of course (77) is unbounded for all values of $\mu>0$.

Without any loss of generality, we can assume that $\phi(X)$ is bounded over every $L(t)$ $(t \in I)$. Strict concavity of $\phi(X)$ implies that for each $t \in I$, there is a unique maximizer $X = X(t)$ of ϕ over $L(t)$. Let $g(t)$ denote the maximum value of $\phi(X)$ over $L(t)$.

Let us consider first the case where $\infty \in I$ that is, the function $C^T X$ is unbounded. Here, there is a ray contained in the interior of the feasible region along which $C^T X$ tends to infinity. Since the domain is a polyhedral, the ray is bounded away from the boundary. Thus, on the ray the function $\phi(X)$ is bounded from below and hence (77) is unbounded for every $\mu > 0$. In the remaining case, we observe that strict concavity of $g(t)$ implies that $t + \mu g(t)$ is bounded for every $\mu > 0$ if t is bounded. Thus in the later case (77) has a unique optimal solution for every $\mu > 0$. This completes the proof.

It therefore follows from the above proposition that if the system (78) has a solution for any positive value of μ, then it determines a unique and continuous path $X = X(\mu)$ where μ varies over the positive reals. When A is of full row rank also, a continuous path $Y = Y(\mu)$ is determined. What we are very much interested in now is the limits of $X(\mu)$ and $Y(\mu)$ as μ tends to zero which will give us the optimal solution. Let us assume for a moment that the limits of $X(\mu)$ and $Y(\mu)$ as μ tends to zero exist and are denoted by x and y respectively. It follows that $Ax = b$, $x \geq 0$ and $A^TY \geq C$. Moreover, for each j such that $x > 0$, $A_j^TY = C_j$ where A_j^T is the j^{th} column of A^T. It follows that x and Y are optimal solutions for (76) and its dual respectively.

To relate these paths to an algorithm for the linear programming problem, we have to at least address two issues. First, we have to know a solution for say $\mu = 1$. Secondly, the limit of $X(\mu)$ (as μ tends to zero) should exist. To see this, we modify the objective function so that an initial solution becomes available. For this, instead of (77) we can work with a problem of the form:

$$\begin{aligned} &\text{Maximize } C^TX + \mu\Sigma w_j \ln X_j \\ &\text{subject to } AX = b \qquad \ldots \quad (79) \\ &\qquad X > 0 \end{aligned}$$

where $W \in \mathbb{R}^n$ is any vector with positive components. Suppose X^o and Y^o are interior feasible solutions for the primal and the dual problems respectively. We can choose w so that the vectors X^o and Y^o satisfy the optimality condition with respect to (79) at $\mu = 1$. The optimality condition is given as

$$\begin{aligned} &\mu D_x^{-1}w - A^TY = -C \qquad \ldots \quad (80) \\ &AX = b \end{aligned}$$

Specifically, $w = D_xo(A^TY^o - C)$. Thus given any pair of interior feasible solutions for the primal and the dual problems, we can easily calculate a suitable weight vector w which in turn determines paths $X(\mu)$ and $Y(\mu)$ as explained above.

Now, if for every $\mu > 0$ the system (78) has a unique solution $(X(\mu), Y(\mu))$ then it can be seen that $C^T X(\mu)$ tends to the optimal value of the objective function of the primal problem (76). To see this, we multiply the first row of (78) by $X(\mu)$, the second row by $Y(\mu)$ and then add them up. We get $b^T Y(\mu) - C^T X(\mu) = n\mu$. The optimal value lies between $b^T Y(\mu)$ and $C^T X(\mu)$ and this implies that $C^T X(\mu)$ tends to the optimal value as μ tends to zero. Now, we look at the conditions under which the point $X(\mu)$ tends to an optimal solution.

Let $V(\mu) = C^T X(\mu)$ (where $X(\mu)$ is the optimal solution of (77) and let $V(0)$ denote the optimal value of (76). We have shown that $V(\mu)$ tends to $V(0)$, as μ tends to 0 above. So, obviously $X(\mu)$ is also the optimal solution of the following problem:

$$\begin{aligned} &\text{Maximize } C^T X + \mu \Sigma \ln x_j \\ &\text{subject to } AX = b \qquad \qquad \ldots \quad (81) \\ &\qquad C^T X = v(\mu) \\ &\qquad X > 0 \end{aligned}$$

The latter is equivalent to

$$\begin{aligned} &\text{Maximize } \Sigma \ln x_j \\ &\text{subject to } AX = b \\ &\qquad C^T X = V(\mu) \\ &\qquad X > 0 \end{aligned}$$

Our assumption of existence of the path $V(\mu)$ is equivalent to existence of an optimal solution for the problem (81). From proposition (18) the function $\phi(X)$ is bounded on $L(t)$ where $t = V(\mu)$ for some $\mu > 0$. We assert that this implies that the set L(t) itself is bounded.

Let N denote the set of all indices j such that $x_j = 0$ in every optimal solution. Thus the optimal face is the intersection of the feasible domain with subspace $\{X | x_j = 0,\ j \epsilon N\}$

Let $\phi_N(X) = \Sigma \ln x_j$

and $\phi_B(X) = \phi(X) - \phi_N(X)$.

Let $\xi_j(\mu)$ denote the j^{th} component of the vector $V(\mu)$, that is the optimal solution at μ. Since $\phi_N(X)$ is constant on the set $\{X | x_j = \xi_j(\mu),\ j \in N\}$, it follows that the point $X(\mu)$ is actually the optimal solution of the problem.

$$\begin{aligned} &\text{Maximize } \phi_B(X) \\ &\text{subject to } AX = b \\ &\quad C^T X = V(\mu) \\ &\quad x_j = \xi_j(\mu) \qquad (j \epsilon N) \\ &\quad X > 0 \end{aligned}$$

Since the optimal set is bounded, it follows that the problem

$$\begin{aligned} &\text{Maximize } \phi_B(X) \\ &\text{subject to } AX = b \\ &\quad C^T X = V(0) \\ &\quad x_j = 0 (j \epsilon N) \\ &\quad x_j > 0 \quad (j \notin N) \end{aligned}$$

has a unique optimal solution which we denote by $x(0)$.

$$ie \lim_{\mu \to 0} x(\mu) \to x(0).$$

We can thus state the following proposition:

Proposition 19

If for some $\mu > 0$ the system (78) has a solution $X > 0$ then for every $\mu > 0$ there is a solution $X(\mu)$ so that the path $X(\mu)$ is continuous and the limit of $x(\mu)$ as μ tends to zero exists and constitutes an optimal solution to the linear programming problem (76).

The implication of the above proposition is that starting from say $\mu = 1$, where we readily have an optimal solutions to problems of the form (79), we follow the path of optimal solutions for such problems while μ varies from 1 to 0. The limit as μ tends to zero is an optimal solution to the linear programming problem, namely the point $X(0)$.

3.16 A Primal Dual Interior Point Algorithm for Linear Programming (Path following algorithm)

3.17 Introduction

Based on the work of Megiddo [28], on central path, Kojima *et al* [27] have proposed a primal-dual interior point algorithm for linear programming.
The algorithm works simultaneously on primal and dual linear programming problems and generates a sequence of pairs of their interior feasible solutions. Along the sequence generated, the duality gap converges linearly to zero.

3.18 Information Discussion

We consider a pair of primal-dual programming problems as follows:

Minimize C^TX ... (82)

subject to $X \in S = \{X \in \mathbb{R}^n : AX = b,\ X \geq 0\}$

Maximize b^TY ... (83)

subject to $(Y,Z) \in T = \{(Y,Z) \in \mathbb{R}^{m+n} : A^TY + Z = C,\ Z \geq 0\}$

where $C,\ X \in \mathbb{R}^n$ $b \in \mathbb{R}^m$ and A is an m x n constraint matrix. We impose the following assumptions on these problems throughout this algorithm.

(a) The set $S^1 = \{X \in S:\ X > 0\}$ of strictly positive feasible solutions of the primal problem (82) is non-empty.

(b) The set $T^1 = \{Y,Z) \in T : Z > 0\}$ of the interior point of the feasible region T of the dual problem (83) is non-empty.

(c) Rank (A) = M.

In order to solve the problem (82), we consider the problem minimizing the objective function $C^TX - \mu\Sigma \ln x_j$ over the interior of feasible region S as follows:

Minimize $C^TX - \mu\Sigma \ln x_j$

subject to $X \in S^1$ ... (84)

The objective function of (84) is convex. Hence each problem of (84) and its dual has at most one global optimal solution which is completely characterized by Karush-Kuhn-Tucker stationary conditions (for example see Mangasarian [24]).

$$C - \mu\ D_x^{-1}e - A^TY = 0$$

$$Ax - b = 0$$

$$X > 0,$$

where $D_x = \text{diag}\ (x_1, x_2, \ldots, x_n)$ denotes the diagonal matrix with the diagonal elements $x_1, x_2, \ldots, x_n$ and $e \in \mathbb{R}^n$ is the vector of 1's. Introducting a slack variable vector $Z=\mu D_x^{-1}e$, the stationary conditions can be reduced to

$$D_z X - \mu e = 0$$
$$AX - b = 0 \qquad \ldots \quad (85)$$
$$A^T Y + Z - C = 0$$
$$X > 0$$

where D_z = diag $(z_1, z_2, \ldots, z_n)$. The first equality can be further written component wise as $x_i z_i = \mu \quad i=1,2,\ldots,n$.

Let $(X(\mu), Y(\mu), Z(\mu))$ denote a unique solution of the system (85) for each $\mu > 0$. Then each $(Y(\mu), Z(\mu))$ lies in the interior of the dual feasible region **T**, that is $(Y(\mu), Z(\mu)) \in T^1$. The duality gap turns out to be $n\mu$. In fact,

$$\begin{aligned} C^T X(\mu) - b^T Y(\mu) &= (C^T - Y(\mu)^T A) X(\mu) \\ &= Z(\mu)^T X(\mu) \\ &= e^T Z(\mu) X(\mu) \\ &= n\mu. \end{aligned}$$

Let $\Gamma = \{X(\mu), Y(\mu), Z(\mu) : \mu > 0\}$.

As we take the limit $\mu \to 0$, the curve Γ leads to a pair of optimal solution X^* of problem (82) and (Y^*, Z^*) of problem (83) which satisfies the strong complementarity such that $z_i^* = 0$ if and only if $x_i^* > 0$. Thus we have observed that if $(X(\mu), Y(\mu), Z(\mu))$ is a solution to the system (85) above, then

$X(\mu) \in S^1$.

$(Y(\mu), Z(\mu)) \alpha T^1$

and $C^T X(\mu) - b^T(\mu) = n\mu$.

Given an initial point $(X^o, Y^o, Z^o) \in S^1 \times T^1$ the algorithm generates a sequence (X^k, Y^k, Z^k) in $S^1 \times T^1$. To measure a deviation from the curve Γ at each iteration $(X^k, Y^k, Z^k) \in S^1 x T^1$ we introduce the quantity π^k defined as follows:

$$\pi^k = f^k_{ave} | f^k_{min} \qquad \ldots (86)$$

$$\text{where } f_i^k = x_i^k z_i^k \; (i=1,2,\ldots,n)$$
$$f^k_{min} = \min | f_i^k : i=1,2,\ldots,n |$$

$$\ldots (87)$$

$$f^k_{ave} = \left\{ \left(\sum_{i=1}^{n} f_i^k \right) / n \right\}$$

We can easily see that $\pi^k \geq 1$ and that $(X^k, Y^k, Z^k) \in S^1 \times T^1$ lies on the curve Γ if and only if $\pi^k = 1$. When the deviation π^o at the initial point $(X^o, Y^o, Z^o) \in S^1 \times T^1$ is large, we have to reduce not only the duality gap but also the deviation. Keeping a small deviation, we can then direct our main effort to reducing the duality gap.

The algorithm we are going to see next makes use of the common rescaling matrix $(D_x D_z^{-1})^{½}$ which is the geometric mean of the primal rescaling matrix D_x and the dual rescaling matrix D_z^{-1} and is applied simultaneously to the primal and dual spaces even if the point (X, Z) does not lie on the curve Γ.

3.19 The Algorithm

The algorithm begins by defining the Newton direction of the system of equations (85) at $(X^k, Y^k, Z^k) \in S^1 \times T^1$. If we denote the left hand side of equations (85) by $H(X,Y,Z)$, then the Newton direction $(\Delta X, \Delta Y, \Delta Z)$ at (X^k, Y^k, Z^k) is defined as the unique solution of the system of linear equations:

$$H(X^k, Y^k, Z^k) - J_{FX} H\Delta X - J_{FY} H\Delta Y - J_{FZ} H\Delta Z = 0$$

where $J_{FX} H$, $J_{FY} H$ and $J_{FZ} H$ denote the Jacobian matrices of the mapping H at (X^k, Y^k, Z^k) with respect to the variable vectors X,Y and Z respectively.

$$ie \begin{pmatrix} A \\ 0 \\ z \end{pmatrix} \Delta X + \begin{pmatrix} 0 \\ A_T \\ 0 \end{pmatrix} \Delta Y + \begin{pmatrix} 0 \\ I \\ X \end{pmatrix} \Delta Z = \begin{pmatrix} AX^k - b \\ A^T Y^k + Z_k^k - C \\ D z^k - \mu e \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ V(\mu) \end{pmatrix}$$

which on simplification gives

$$\begin{aligned} D_z \Delta X + D_x \Delta Z &= V(\mu) \\ A\Delta X &= 0 \\ A^T \Delta Y + \Delta Z &= 0 \end{aligned} \qquad \ldots \quad (88)$$

where $D_x = \text{diag}\ (x_1^k, x_2^k, \ldots, x_n^k)$

$D_z = \text{diag}\ (z_1^k, z_2^k, \ldots, z_n^k)$

$V(\mu) = D_x z^k - \mu e \in \mathbb{R}^n$

Since a solution of this system depends linearly on the parameter μ, so the solution can be written as $(\Delta X(\mu), \Delta Y(\mu), \Delta Z(\mu))$.

Introducing the diagonal matrix

$D = (D_x D_z)^{-1½} = \text{diag}\ \{(x_1^k/z_1^k)^{½}, \ldots, (x_n^k/z_n^k)^{½}\}$,

the vector-valued linear function

$U(\mu) = (D_xD_z)^{1/2} - \mu(D_xD_z)^{-1/2} e \in \mathbb{R}^n$ or $U_i(\mu = (x_i^kz_i^k)^{1/2} - \mu(x_i^kz_i^k)^{-1/2}$

$i=1,2,...,n.$

in the variable μ, and the orthogonal projection matrix

$Q = DA^T (AD^2A^T)^{-1} AD$ onto the subspace $\{DA^TY : Y \in \mathbb{R}^m\}$

Then the newton direction $(\Delta X, \Delta Y, \Delta Z)$ can be written as

$$\Delta X(\mu) = D(I - Q)\ U(\mu)$$
$$\Delta Y(\mu) = -(AD^2A^T)^{-1}ADU(\mu) \qquad \ldots \quad (90)$$
$$\Delta Z(\mu) = D^{-1}QU(\mu)$$

These equalities and (88) imply

$$D^{-1}\Delta X(\mu) + D\Delta Z(\mu) = U(\mu) \qquad \ldots \quad (91)$$
$$0 = \Delta X(\mu)^T \Delta Z(\mu) = (D^{-1}\Delta X(\mu))^T (D\Delta Z(\mu))$$

We fix two real parameters σ and τ such that $0 \le \tau < \sigma < 1$ which control the parameter μ and the step length respectively.

We define π^k, f_i^k $(i=1,2,...,n)$, f_{ave}^k and f_{min}^k by (86) and (87). We will set the parameter $\mu = \sigma\ f_{ave}^k$ and then compute the $(k+1)^{th}$ iterate from (X^k, Y^k, Z^k) in the direction $-(\Delta X, \Delta Y, \Delta Z)$. That is $(X^{k+1}, Y^{k+1}, Z^{k+1})$ can be written as follows:

$$X^{k+1} = X(\alpha,\mu) = X^k - \alpha\Delta X(\mu)$$
$$Y^{k+1} = Y(\alpha,\mu) = Y^k - \alpha\Delta Y(\mu) \qquad \ldots \quad (92)$$
$$Z^{k+1} = Z(\alpha,\mu) = Z^k - \alpha\Delta Z(\mu)$$

where α is the step length.

$$f_i(\alpha,\mu) = x_i(\alpha,\mu)\, z_i\ (\alpha,\mu)$$
$$= x_i^kz_i^k - \alpha(x_ik\Delta z_i(\mu) + z_i^k\ \Delta x_i(\mu)) + \Delta x_i(\mu)\Delta z_i^k(\mu)\alpha^2$$
$$f_i^k - (f_i^k - \mu)\alpha + \Delta x_i(\mu)\ \Delta z_i(\mu)\alpha^2$$
$$i = 1,2,..,n.$$

(Since $x_i^k\ \Delta z_i(\mu) + z_i^k\ \Delta x_i(\mu) = f_i^k - \mu$ by (88)) and the average f_{ave} of f_i $(i=1,2,...,n)$ as well as the duality gap

$C^TX - b^TY$ changes linearly.

$$f_{ave}(\alpha,\mu) = \left\{\sum_{i=1}^{n} f_i(\alpha,\mu)\right\}/n$$

(Since $\Delta X(\mu)^T \Delta Z(\mu) = 0$ by (91)

$$C^TX(\alpha, \mu) - b^TY(\alpha, \mu) = X(\alpha, \mu)^T Z(\alpha, \mu)$$
$$= X^{kT}Z^k - (X^{kT}Z^k - (n\mu))\alpha$$
$$= (C^TX^k - b^TY^k) - \{(C^TX^k - b^TY^k) - n\mu\}\alpha$$

The step length α^k is determined as follows:

Let $\beta^k = \max \{\alpha : f_i(\alpha', \mu) \geq \psi(\alpha')$ for all $\alpha' \in (0,\alpha)$ and $i=1,2,\ldots,n\}$

then $\alpha^k = \min \{1,\beta^k\}$ ... (93)

$$\psi(\alpha) = f_{min}^k - (f_{min}^k - \tau f_{ave}^k)\alpha.$$

The algorithm can now be formally stated as follows:

Step 0

Let $(X^o, Y^o, Z^o) \in S^1 \times T^1$ be an initial point and ϵ a tolerance for duality gap. Let σ and τ be real numbers such that $0 \leq \tau < \sigma < 1$,

Let $k = 0$

Step 1

$C^TX^k - b^TY^k < \epsilon$ then stop.

Step 2

Compute $f_{ave}{}^k$ and$f_{min}{}^k$ by (87)

Step 3

Let $\mu = \sigma f_{ave}{}^k$ and compute

$(\Delta X(\mu), \Delta Y(\mu), \Delta Z(\mu))$ by (90)

Step 4

Compute β^k and α^k by (93).

Step 5

$X^{k+1} = X(\alpha^k, \mu) = X^k - \alpha^k \Delta X(\alpha^k, \mu)$

$Y^{k+1} = Y(\alpha^k, \mu) = Y^k - \alpha^k \Delta Y(\alpha k, \mu)$

$Z^{k+1} = Z(\alpha^k, \mu) = Z^k - \alpha^k \Delta Z(\alpha^k, \mu)$

Step 6

k = k+1. Go to step 1.

Experience with the methods in solving problems

Having analysed the various interior point algorithms, it is worth noting that solving a linear programming problem, for instance

$$\text{Minimize } Z = x_1 + x_2$$
$$\text{subject to } 2x_1 + x_2 \geq 4$$
$$x_1 + 7x_2 \geq 7$$
$$x_1,\ x_2 \geq 0$$

using the above methods requires many iterations to arrive at optimality. Meanwhile, using Dantzig's simplex algorithm requires just three iterations as shown below

Simplex Method

Solution

First we convert the problem into maximization i.e.

$$\text{Max } Z' = -Z = -x_1 - x_2.$$

Introducing surplus variables x_3 and x_4 and also artificial variables x_5 and x_6, the constraints of the problem can be rewritten as

$$2x_1 + x_2 - x_3 + 0x_4 + x_5 + 0x_6 = 4$$
$$x_1 + 7x_2 - 0x_3 - x_4 + 0x_5 + x_6 = 7$$

Equation non-basic variables to zero, we have

$x_5 = 4,\ x_6 = 7$

Phase I

1st Iteration:

	X_B	Y_1	Y_2	Y_3	Y_4	$A_1(B_1)$	$A_2(B_2)$
$A_1(B_1)$	4	2	1	-1	0	1	0
$A_2(B_2)$	7	1	7	0	-1	0	1
	x_j	0	0	0	0	4	7

Y_2 enters the basis while $A_2(B_2)$ leaves the basis.

2nd Iteration

	X_B	Y_1	Y_2	Y_3	Y_4	$A_1(B_1)$
$A_1(B_1)$	3	13/7	0	-1	1/7	1
Y_2	1	1/7	1	0	1/7	0
	x_j	0	1	0	0	3

Y_1 enters the basis while $A_1(B_1)$ leaves the basis.

3rd Interation

	X_B	Y_1	Y_2	Y_3	Y_4
Y_1	21/13	1	0	-7/13	1/13
Y_2	10/13	0	1	1/13	-14/91
	x_j	21/13	10/13	0	0

Thus we have the following solution in phase I.

$X_1 = 21/13,\ x_2 = 10/13,\ x_3 = 0,\ x_4 = 0$

Phase II

Starting simplex table

B	C_B	X_B	$Y_1(B_1)$	$Y_2(B_2)$	Y_3	Y_4	Min. Ratio
Y_1	-1	21/13	1	0	-7/13	1/13	
		x_j c_j	21/13 -1	10/13 -1	0 0	0 0	
		$\Delta_j = c_j - 7_j$	0	0	-6/13	-7/71	

All values of $\Delta_j \leq 0$. Hence optimality has been achieved. Therefore, the required solution is $x_1 = 21/13$ $x_2 = 10/13$

Min Z = -Max Z' = (-31/13) = 31/13.

However, using two of the interior point algorithms outlined above, we have the following results

Using Karmarkar's projective rescaling algorithm

First, we change the problem from standard to canonical form.

$$\text{Minimize} \quad Z = x_1 + x_2$$

$$\text{Subject to } 2x_1 + x_2 \geq 4$$

$$x_1 + x_2 \geq 7$$

$$x_1,\ x_2 \geq 0.$$

Introducing surplus variables, we have

$$\text{Minimize} \quad Z = x_1 + x_2$$

$$\text{Subject to } 2x_1 + x_2 - x_3 = 4$$

$$x_1 + x_2 - x_4 = 7$$

$$x_i \geq 0 \qquad I = 1, \ldots, 4.$$

Let X = a = (2,2,2,9) be a strictly interior starting point.

$$A = \begin{pmatrix} 2 & 1 & -1 & 0 \\ 1 & 7 & 0 & -1 \end{pmatrix}$$

$$A_1^1 = A_1 a_1 = 2\begin{pmatrix}2\\1\end{pmatrix} = \begin{pmatrix}4\\2\end{pmatrix}$$

$$A_2^1 = A_2 a_2 = 2\begin{pmatrix}1\\7\end{pmatrix} = \begin{pmatrix}2\\14\end{pmatrix}$$

$$A_3^1 = A_3 a_3 = 2\begin{pmatrix}-1\\0\end{pmatrix} = \begin{pmatrix}-2\\0\end{pmatrix}$$

$$A_4^1 = A_4 a_4 = 9\begin{pmatrix}0\\-1\end{pmatrix} = \begin{pmatrix}0\\-9\end{pmatrix}$$

$$A_5^1 = \begin{pmatrix}-4\\-7\end{pmatrix}$$

$$A_1 = \begin{pmatrix}4 & 2 & -2 & 0 & -4\\2 & 14 & 0 & -9 & -7\end{pmatrix}$$

C = (1, 1, 0, 0)

$$C^1 = DC = \begin{pmatrix}2 & & & \\ & 2 & & \\ & & 2 & \\ & & & 9\end{pmatrix}\begin{pmatrix}1\\1\\0\\0\end{pmatrix} = \begin{pmatrix}2\\2\\0\\0\end{pmatrix} \textit{since } C_{n+1} = 0$$

Canonical form of the problem is now stated as follows:

Minimize $Z = 2x_1 + 2x_2$

subject to $4x_1 + 2x_2 - 3x_3 - 4x_j = 0$

$2x_1 + 14x_2 - 9x_4 - 7x_5 = 0$

$$\sum x_i = 1$$

$$X_j \geq 0 \qquad I=1, \ldots, 5.$$

$$A_5^1 = \begin{pmatrix} -4 \\ -7 \end{pmatrix}$$

$$A_1 = \begin{pmatrix} 4 & 2 & -2 & 0 & -4 \\ 2 & 14 & 0 & -9 & -7 \end{pmatrix}$$

C = (1, 1, 0, 0)

$$C^1 = DC = \begin{pmatrix} 2 & & & \\ & 2 & & \\ & & 2 & \\ & & & 9 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 2 \\ 2 \\ 0 \\ 0 \end{pmatrix} since\ C_{n+1} = 0$$

Canonical form of the problem is now stated as follows:

Minimize $Z = 2x_1 + 2x_2$

subject to $4x_1 + 2x_2 - 3x_3 - 4x_j = 0$

$2x_1 + 14x_2 - 9x_4 - 7x_5 = 0$

$$\sum x_i = 1$$

$$X_j \geq 0 \qquad I=1, \ldots, 5.$$

a_o = e/n = (1/5, 1/5, 1/5, 1/5, 1/5) which is a feasible starting point.

D = diag (1/5, 1/5, 1/5, 1/5, 1/5)

e^T (1, 1, 1, 1, 1)

q is a real number 1 set equal to 25.

$$AD = \begin{pmatrix} 4 & 2 & -2 & 0 & -4 \\ 2 & 14 & 0 & -9 & -7 \end{pmatrix} \begin{pmatrix} 0.2 & & & & \\ & 0.2 & & & \\ & & 0.2 & & \\ & & & 0.2 & \\ & & & & 0.2 \end{pmatrix}$$

$$= \begin{pmatrix} 0.8 & 0.4 & -0.4 & 0 \\ 0.4 & 2.8 & 0 & -1.8 \\ 1 & 1 & 1 & 1 \end{pmatrix}$$

$$B^T = \begin{pmatrix} 0.8 & 0.4 & 1 \\ 0.4 & 2.8 & 1 \\ -0.4 & 0 & 1 \\ 0 & -1.8 & 1 \\ 0.8 & -1.4 & 1 \end{pmatrix}$$

$$BB^T = \begin{pmatrix} 1.6 & 2.56 & 0 \\ 2.56 & 13.2 & 0 \\ 0 & 0 & 5 \end{pmatrix}$$

$$(BB^T)^{-1} = \begin{pmatrix} 0.9 & 0.17 & 0 \\ 0.17 & 0.1 & 0 \\ 0 & 0 & 0.2 \end{pmatrix}$$

$$B^T(BB^T)^{-1} = \begin{pmatrix} 0.8 & 0.4 & 1 \\ 0.4 & 2.8 & 1 \\ 0.4 & 0 & 1 \\ 0 & -1.8 & 1 \\ -0.8 & -1.4 & 1 \end{pmatrix} \begin{pmatrix} 0.9 & -0.17 & 0 \\ -0.17 & 0.1 & 0 \\ 0 & 0 & 0.2 \end{pmatrix}$$

$$B^T(BB^T)^{-1}B = \begin{pmatrix} 0.65 & -0.09 & 0.2 \\ -0.11 & 0.21 & 0.2 \\ -0.36 & 0.6 & 0.2 \\ 0.3 & -0.18 & 0.2 \\ -0.48 & -0.004 & 0.2 \end{pmatrix} \begin{pmatrix} 0.8 & 0.4 & -0.4 & -0 & -0.8 \\ 0.4 & 2.8 & 0 & -1.8 & -1.4 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

$$= \begin{pmatrix} 0.68 & 0.2 & -0.6 & 0.36 & -0.19 \\ 0.19 & 0.74 & 0.24 & -0.17 & -0.006 \\ -0.06 & 0.22 & 0.34 & 0.09 & 0.4 \\ 0.36 & -0.18 & 0.08 & 0.52 & 0.21 \\ -0.18 & -0.003 & 0.39 & 0.2 & 0.58 \end{pmatrix}$$

$$= = \begin{pmatrix} 0.24 \\ 0.14 \\ -0.32 \\ -0.36 \\ 0.36 \end{pmatrix} = C_p.$$

$||C_p|| = 0.662$

$$\hat{C} = \frac{C_p}{||C_p||} = \begin{pmatrix} 0.36 \\ 0.21 \\ -0.48 \\ -0.54 \\ 0.54 \end{pmatrix}$$

$\alpha = 0.25$

$$r = \frac{1}{\sqrt{n(n-1)}} = \frac{1}{\sqrt{5(4)}} = \frac{1}{\sqrt{20}}$$

$\alpha r = 0.056$

$$b^1 = \begin{pmatrix}0.2\\0.2\\0.2\\0.2\end{pmatrix} - 0.056\begin{pmatrix}0.36\\0.21\\-0.48\\-0.54\\0.54\end{pmatrix} = \begin{pmatrix}0.18\\0.19\\0.23\\0.23\\0.17\end{pmatrix}$$

$$X^1 = \frac{D_o b^1}{e^T D_o b^1} = \frac{\begin{pmatrix}0.2&&&&\\&0.2&&&\\&&0.2&&\\&&&0.2&\\&&&&0.2\end{pmatrix}\begin{pmatrix}0.18\\0.19\\0.23\\0.23\\0.17\end{pmatrix}}{\sum Db^1} = \frac{\begin{pmatrix}0.036\\0.038\\0.046\\0.046\\0.034\end{pmatrix}}{0.2} = \begin{pmatrix}0.18\\0.19\\0.23\\0.23\\0.17\end{pmatrix}$$

Assuming problem is feasibly we check the optimality.

Optimality

We require the convergence check

$$\frac{C^{1T}X^1}{C^{1T}X^o} \le 2^{-q}$$

where $X^o = a_o$

$$C^{1T}X^o = (2,2,0,0,0)\begin{pmatrix}0.2\\0.2\\0.2\\0.2\\0.2\end{pmatrix} = 0.8$$

$$\frac{0.8}{0.8} = 1 > 2^{-q}$$

$$C^{1T}X^{1} = (2,2,0,0,0)\begin{pmatrix}0.18\\0.19\\0.23\\0.23\\0.17\end{pmatrix} = 0.74$$

Checking for optimality

$$0.74/0.8 = 0.925 > 2^{-q}$$

Optimality has not been achieved.

Next Iteration

$X^1 = (0.18,\ 0.19,\ 0.23,\ 0.23,\ 0.17)$

$$AD_1 = \begin{pmatrix}4 & 2 & -2 & 0 & -4\\2 & 14 & 0 & -9 & -7\end{pmatrix}\begin{pmatrix}0.18 & & & & \\ & 0.19 & & & \\ & & 0.23 & & \\ & & & 0.23 & \\ & & & & 0.17\end{pmatrix}$$

$$BB^{T} = \begin{pmatrix}1.33 & 2.07 & -0.04\\2.07 & 12.91 & -0.24\\-0.04 & -0.24 & 5\end{pmatrix}$$

$$B^{T} = \begin{pmatrix}0.72 & 0.36 & 1\\0.38 & 2.66 & 1\\-0.46 & 0 & 1\\0 & -2.07 & 1\\0.68 & -1.19 & 1\end{pmatrix}$$

$$B = \begin{pmatrix} AD \\ e^T \end{pmatrix} = \begin{pmatrix} 0.72 & 0.38 & -0.46 & 0 & -0.68 \\ 0.36 & 2.66 & 0 & -2.07 & -1.19 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

$$B^T(BB^T)^{-1} = \begin{pmatrix} 0.72 & 0.36 & 1 \\ 0.38 & 2.66 & 1 \\ 0.46 & 0 & 1 \\ 0 & -2.07 & 1 \\ -0.68 & -1.19 & 1 \end{pmatrix} \begin{pmatrix} 1 & -0.16 & 0.0003 \\ 0.16 & 0.1 & 0.004 \\ 0.0003 & 0.0004 & 0.2 \end{pmatrix}$$

$C^{1T}X^3 = 0.36$

Checking for optimality

$0.36/0.8 = 0.45 > 2^{-q}$

Optimality is still not achieved.

Using the affine rescaling algorithm

Solution

Introducing surplus variables, we have

Minimize $Z = x_1 + x_2$

Subject to $2x_1 + x_2 - x_3 = 4$

$x_1 + 7x_2 - x_4 = 7$

$x_1 \geq 0 \qquad i = 1_j, \ldots, 4.$

Let $X^o = (2, 2, 2, 9)$ be an interior point satisfying $Ax^o = b$.

$$A = \begin{pmatrix} 2 & 1 & -1 & 0 \\ 1 & 7 & 0 & -1 \end{pmatrix}$$

$$D_2^o = \begin{pmatrix} 4 & & & \\ & 4 & & \\ & & 4 & \\ & & & 81 \end{pmatrix}$$

$$C^T = (1,\ 1,\ 0,\ 0)$$

$$D_o = diag\ (2,2,2,9) = \begin{pmatrix} 2 & & & \\ & 2 & & \\ & & 2 & \\ & & & 9 \end{pmatrix} \qquad AD_o^2 = \begin{pmatrix} 2 & 1 & -1 & 0 \\ 1 & 7 & 0 & -1 \end{pmatrix} \begin{pmatrix} 4 & & & \\ & 4 & & \\ & & 4 & \\ & & & 9 \end{pmatrix}$$

$$AD_o^2 A^T = \begin{pmatrix} 8 & 4 & -4 & 0 \\ 4 & 28 & 0 & -81 \end{pmatrix} \begin{pmatrix} 2 & 1 \\ 1 & 7 \\ -1 & 0 \\ 0 & -1 \end{pmatrix}$$

$$(AD_o^2 A^T)^{-1} AD_o^2 = \begin{pmatrix} 0.05 & -0.006 \\ -0.006 & 0.004 \end{pmatrix} \begin{pmatrix} 8 & 4 & -4 & 0 \\ 4 & 28 & 0 & -81 \end{pmatrix}$$

$$\lambda = (AD_o^2 A^T)^{-1} AD_o^2 C = \begin{pmatrix} 0.37 & 0.03 & -0.2 & 0.48 \\ -0.03 & 0.08 & 0.02 & -0.32 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \end{pmatrix}$$

$$= \begin{pmatrix} 0.4 \\ 0.05 \end{pmatrix}$$

$$= \begin{pmatrix} 24 & 36 \\ 36 & 281 \end{pmatrix}$$

$$A^T \lambda = \begin{pmatrix} 2 & 1 \\ 1 & 7 \\ -1 & 0 \\ 0 & -1 \end{pmatrix} = \begin{pmatrix} 0.4 \\ 0.05 \end{pmatrix} \begin{pmatrix} 0.85 \\ 0.75 \\ -0.4 \\ -0.05 \end{pmatrix}$$

$$(C-A^T\lambda) = \begin{pmatrix}1\\1\\0\\0\end{pmatrix} - \begin{pmatrix}0.85\\0.75\\-0.4\\-0.05\end{pmatrix} = \begin{pmatrix}0.15\\0.25\\0.4\\0.05\end{pmatrix}$$

$$D_o^2(C-A^T\lambda) = \begin{pmatrix}4 & & & \\ & 4 & & \\ & & 4 & \\ & & & 81\end{pmatrix}\begin{pmatrix}0.15\\0.25\\0.4\\0.05\end{pmatrix} = \begin{pmatrix}0.6\\1\\1.6\\4.05\end{pmatrix}$$

$$= 1.087$$

$$D_o^2(C-A^T\lambda) = \begin{pmatrix}2 & & & \\ & 2 & & \\ & & 2 & \\ & & & 9\end{pmatrix}\begin{pmatrix}0.15\\0.25\\0.4\\0.05\end{pmatrix} = \begin{pmatrix}0.6\\1\\1.6\\4.05\end{pmatrix}$$

$$||D_o(C-A^T\lambda)|| = \sqrt{(0.3)^2+(0.5)^2+(0.8)^2+(0.45)^2}$$

$$\frac{D_o^2(C-A^T\lambda)}{||D_o(C-A^T\lambda)||} = \begin{pmatrix}0.55\\0.91\\1.47\\3.72\end{pmatrix}$$

$$X^1 = \begin{pmatrix}2\\2\\2\\9\end{pmatrix} -0.25\begin{pmatrix}0.55\\0.91\\1.47\\3.72\end{pmatrix} = \begin{pmatrix}1.9\\1.8\\1.6\\8.07\end{pmatrix}$$

$$C^T X^\circ = (1,1,0,0)\begin{pmatrix}2\\2\\2\\9\end{pmatrix} = 4$$

$$C^T X^1 = (1,1,0,0)\begin{pmatrix}1.9\\1.8\\1.6\\8.07\end{pmatrix} = 3.7$$

Next Iteration

$$D_1 = \begin{pmatrix}1.9 & & & \\ & 1.8 & & \\ & & 1.6 & \\ & & & 8.07\end{pmatrix}$$

$$D_1^2 = \begin{pmatrix}3.61 & & & \\ & 3.24 & & \\ & & 2.56 & \\ & & & 65.12\end{pmatrix}$$

$$AD_1^2 = \begin{pmatrix}2 & 1 & -1 & 0\\ 1 & 7 & 0 & -1\end{pmatrix}\begin{pmatrix}3.61 & & & \\ & 3.24 & & \\ & & 2.56 & \\ & & & 65.12\end{pmatrix}$$

$$AD_1^2A^T = \begin{pmatrix}7.22 & 3.24 & -2.56 & 0\\ 3/61 & 22.68 & 0 & -65.12\end{pmatrix}\begin{pmatrix}2 & 1\\ 1 & 7\\ -1 & 0\\ 0 & -1\end{pmatrix}$$

$$= \begin{pmatrix}20.24 & 29.9\\ 29.9 & 227.5\end{pmatrix}$$

$$(AD_1^2A^T)^{-1}AD_1^2 = \begin{pmatrix} 0.06 & -0.008 \\ -0.008 & 0.005 \end{pmatrix} \begin{pmatrix} 7.22 & 3.24 & -2.56 & 0 \\ 3.61 & 22.68 & 0 & -65.12 \end{pmatrix}$$

$$\lambda = (AD_o^2A^T)^{-1}AD_o^2C = \begin{pmatrix} 0.37 & 0.03 & -0.2 & 0.48 \\ -0.03 & 0.08 & 0.02 & -0.32 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \end{pmatrix}$$

$$= \begin{pmatrix} 0.41 \\ 0.05 \end{pmatrix}$$

$$A^T\lambda = \begin{pmatrix} 2 & 1 \\ 1 & 7 \\ -1 & 0 \\ 0 & -1 \end{pmatrix} \begin{pmatrix} 0.41 \\ 0.05 \end{pmatrix} = \begin{pmatrix} 0.87 \\ 0.76 \\ -0.41 \\ -0.05 \end{pmatrix}$$

$$(C-A^T\lambda) = \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \end{pmatrix} - \begin{pmatrix} 0.87 \\ 0.76 \\ -0.41 \\ -0.05 \end{pmatrix} \begin{pmatrix} 0.13 \\ 0.24 \\ 0.41 \\ 0.05 \end{pmatrix}$$

$$D_1^2(C-A^T\lambda) = \begin{pmatrix} 3.61 & & & \\ & 3.24 & & \\ & & 2.56 & \\ & & & 65.12 \end{pmatrix} \begin{pmatrix} 0.13 \\ 0.24 \\ 0.41 \\ 0.05 \end{pmatrix} = \begin{pmatrix} 0.46 \\ 0.77 \\ 1.04 \\ 3.25 \end{pmatrix}$$

$$D_1(C-A^T\lambda) = \begin{pmatrix} 1.9 & & & \\ & 1.8 & & \\ & & 1.6 & \\ & & & 8.07 \end{pmatrix} \begin{pmatrix} 0.13 \\ 0.24 \\ 0.41 \\ 0.05 \end{pmatrix} = \begin{pmatrix} 0.24 \\ 0.43 \\ 0.65 \\ 0.4 \end{pmatrix}$$

$||D_1(C-A^T\lambda)|| = -.908.$

$$\frac{D_1^2(C-A^T\lambda)}{||D_1(C-A^T\lambda)||} = \begin{pmatrix} 0.5 \\ 0.84 \\ 1.14 \\ 3.57 \end{pmatrix}$$

$$X^2 = \begin{pmatrix} 1.9 \\ 1.8 \\ 1.6 \\ 8.07 \end{pmatrix} - 0.25 \begin{pmatrix} 0.5 \\ 0.84 \\ 1.14 \\ 3.57 \end{pmatrix} \begin{pmatrix} 1.8 \\ 1.6 \\ 1.32 \\ 7.18 \end{pmatrix}$$

$C^TX^2 = 3.4.$

Next Iteration

$$D_2 = \begin{pmatrix} 1.8 & & & \\ & 1.6 & & \\ & & 1.32 & \\ & & & 7.18 \end{pmatrix}$$

$$D_2^2 = \begin{pmatrix} 3.24 & & & \\ & 2.56 & & \\ & & 1.74 & \\ & & & 51.55 \end{pmatrix}$$

$$AD_2^2 = \begin{pmatrix} 2 & 1 & -1 & 0 \\ 1 & 7 & 0 & -1 \end{pmatrix} \begin{pmatrix} 3.24 & & & \\ & 2.56 & & \\ & & 1.74 & \\ & & & 51.55 \end{pmatrix}$$

$$AD_2^2A^T = \begin{pmatrix} 6.48 & 2.56 & -1.74 & 0 \\ 3.24 & 17.92 & 0 & -5.55 \end{pmatrix} \begin{pmatrix} 2 & 1 \\ 1 & 7 \\ -1 & 0 \\ 0 & -1 \end{pmatrix}$$

$$= \begin{pmatrix} 17.26 & 24.4 \\ 24.4 & 180.23 \end{pmatrix}$$

$$(AD_2^2A^T)^{-1}AD_2^2 = \begin{pmatrix} 0.07 & -0.009 \\ -0.009 & 0.006 \end{pmatrix} \begin{pmatrix} 6.48 & 2.56 & -1.74 & 0 \\ 3.24 & 17.92 & 0 & -51.55 \end{pmatrix}$$

$$\lambda = (AD_2^2A^T)^{-1}AD_2^2C = \begin{pmatrix} 0.42 & 0.01 & -0.12 & 0.46 \\ -0.03 & 0.08 & 0.01 & -0.3 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \end{pmatrix}$$

$$= \begin{pmatrix} 0.43 \\ 0.05 \end{pmatrix}$$

$$A^T\lambda = \begin{pmatrix} 2 & 1 \\ 1 & 7 \\ -1 & 0 \\ 0 & -1 \end{pmatrix} \begin{pmatrix} 0.43 \\ 0.05 \end{pmatrix} = \begin{pmatrix} 0.91 \\ 0.78 \\ -0.43 \\ -0.05 \end{pmatrix}$$

$$D_2(C-A^T\lambda) = \begin{pmatrix} 1.8 & & & \\ & 1.6 & & \\ & & 1.32 & \\ & & & 7.18 \end{pmatrix} \begin{pmatrix} 0.09 \\ 0.12 \\ 0.43 \\ 0.05 \end{pmatrix} = \begin{pmatrix} 0.16 \\ 0.19 \\ 0.56 \\ 0.35 \end{pmatrix}$$

$$(C-A^T\lambda) = \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \end{pmatrix} - \begin{pmatrix} 0.91 \\ 0.78 \\ -0.43 \\ -0.05 \end{pmatrix} \begin{pmatrix} 0.093 \\ 0.12 \\ 0.43 \\ 0.05 \end{pmatrix}$$

$||D_2(C-A^T\lambda)|| = 0.705$

$$D_2^2(C-A^T\lambda) = \begin{pmatrix} 3.24 & & & \\ & 2.56 & & \\ & & 1.74 & \\ & & & 51.55 \end{pmatrix} \begin{pmatrix} 0.09 \\ 0.12 \\ 0.43 \\ 0.05 \end{pmatrix} = \begin{pmatrix} 0.29 \\ 0.3 \\ 0.74 \\ 2.57 \end{pmatrix}$$

$$\frac{D_2^2(C-A^T\lambda)}{||D_2(C-A^T\lambda)||} = \begin{pmatrix} 0.41 \\ 0.42 \\ 1.04 \\ 3.64 \end{pmatrix}$$

$$X^3 = \begin{pmatrix} 1.8 \\ 1.6 \\ 1.32 \\ 7.18 \end{pmatrix} - 0.25 \begin{pmatrix} 0.41 \\ 0.42 \\ 1.04 \\ 3.64 \end{pmatrix} = \begin{pmatrix} 1.7 \\ 1.5 \\ 1.06 \\ 6.27 \end{pmatrix}$$

$C^TX^3 = 3.2.$

Next Iteration

$$D_3 = \begin{pmatrix} 1.7 & & & \\ & 1.5 & & \\ & & 1.06 & \\ & & & 6.27 \end{pmatrix}$$

$$D_2^3 = \begin{pmatrix} 2.89 & & & \\ & 2.25 & & \\ & & 1.12 & \\ & & & 39.31 \end{pmatrix}$$

$$AD_2^3 = \begin{pmatrix} 2 & 1 & -1 & 0 \\ 1 & 7 & 0 & -1 \end{pmatrix} \begin{pmatrix} 2.89 & & & \\ & 2.25 & & \\ & & 1.12 & \\ & & & 39.31 \end{pmatrix}$$

$$AD_3^2A^T = \begin{pmatrix} 5.78 & 2.25 & -1.12 & 0 \\ 2.89 & 15.75 & 0 & -39.31 \end{pmatrix} \begin{pmatrix} 2 & 1 \\ 1 & 7 \\ -1 & 0 \\ 0 & -1 \end{pmatrix}$$

$$= \begin{pmatrix} 14.93 & 21.53 \\ 21.53 & 152.45 \end{pmatrix}$$

$$(AD_3^2A^T)^{-1}AD_3^2 = \begin{pmatrix} 0.08 & -0.001 \\ -0.001 & 0.006 \end{pmatrix} \begin{pmatrix} 5.78 & 2.25 & -1.12 & 0 \\ 2.89 & 15.75 & 0 & -39.31 \end{pmatrix}$$

$$\lambda = (AD_3^3A^T)^{-1}AD_2^2C = \begin{pmatrix} 0.43 & 0.02 & -0.08 & 0.39 \\ -0.03 & 0.01 & 0.01 & -0.3 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \end{pmatrix}$$

$$= \begin{pmatrix} 0.45 \\ 0.07 \end{pmatrix}$$

$$A^T\lambda = \begin{pmatrix} 2 & 1 \\ 1 & 7 \\ -1 & 0 \\ 0 & -1 \end{pmatrix} \begin{pmatrix} 0.45 \\ 0.07 \end{pmatrix} = \begin{pmatrix} 0.97 \\ 0.94 \\ -0.45 \\ -0.07 \end{pmatrix}$$

$$||D_3(C-A^T\lambda)|| = 0.645$$

$$(C-A^T\lambda) = \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \end{pmatrix} - \begin{pmatrix} 0.97 \\ 0.94 \\ -0.45 \\ -0.07 \end{pmatrix} = \begin{pmatrix} 0.03 \\ 0.06 \\ 0.45 \\ 0.07 \end{pmatrix}$$

$$D_3^2(C-A^T\lambda) = \begin{pmatrix} 2.89 & & & \\ & 2.25 & & \\ & & 1.12 & \\ & & & 39.31 \end{pmatrix} \begin{pmatrix} 0.03 \\ 0.06 \\ 0.45 \\ 0.07 \end{pmatrix} = \begin{pmatrix} 0.08 \\ 0.13 \\ 0.5 \\ 2.75 \end{pmatrix}$$

$$D_3(C-A^T\lambda) = \begin{pmatrix} 1.7 & & & \\ & 1.5 & & \\ & & 1.06 & \\ & & & 6.27 \end{pmatrix} \begin{pmatrix} 0.03 \\ 0.06 \\ 0.45 \\ 0.07 \end{pmatrix} = \begin{pmatrix} 0.05 \\ 0.09 \\ 0.47 \\ 0.43 \end{pmatrix}$$

$$\frac{D_3^2(C-A^T\lambda)}{||D_3(C-A^T\lambda)||} = \begin{pmatrix} 0.12 \\ 0.2 \\ 1.77 \\ 4.26 \end{pmatrix}$$

$$X^4 = \begin{pmatrix}1.7\\1.5\\1.06\\6.27\end{pmatrix} - 0.25\begin{pmatrix}0.12\\0.2\\0.77\\4.26\end{pmatrix} = \begin{pmatrix}1.67\\1.45\\0.87\\5.2\end{pmatrix}$$

$$C^T X^4 = 3.12.$$

Up to this point, convergence is yet to be achieved. Observe also, that the solution of the above problem using the simplex algorithm is 31/13. On the other hand, the two interior point algorithms are arriving at a solution a bit far from that of the simplex algorithm.

This is probably due to the numerous round-offs involved when applying the algorithms manually, most especially the projective rescaling algorithms. Moreover, the rate at which the algorithms converge becomes much slower as they approach optimality. Thus if there were no roundoffs, and the algorithm is patiently allowed to converge, all the three answers would be the same.

From the foregoing, it is therefore needless using the interior point algorithms to solve small linear programming problems. They are even more tedious, time consuming and boring as the number of iterations increases. They are therefore recommended only for large linear programming problems since Karmarkar himself claims that his algorithm can handle the largest problem four hundred times faster than the simplex algorithm. From his claim, we can conclude that the interior point algorithm are inevitable when the problem becomes too large for the simplex algorithm.

CHAPTER FOUR

4.0 Boundary Behaviour of Interior Point Algorithms

We begin the boundary behaviour of interior point algorithms by discussing the search directions along which we search for new interior point solutions from current interior point solution. We consider linear programming problem in standard form

$$\text{Minimize } C^TX$$
$$\text{subject to } AX = b \qquad \ldots \quad (94)$$
$$X \geq 0$$

where $A \in \mathbb{R}^{m\times n}$ $(m\leq n)$, $b\in {}^m$ and $C,\ X\in\mathbb{R}^n$.

Let $X\in\mathbb{R}^n$ and $D = \text{diag}\ (x_1,\ x_2,\ \ldots,\ x_n)$ be a diagonal matrix of order n whose diagonal entries are the components of X. The affine rescaling algorithm assigns to the point X a "search direction" that is a vector ζ which is computed as follows:

Consider a transformation of space $T_x: \mathbb{R}^n \to \mathbb{R}^n$ given by $T_x(Y) = D^{-1}Y$. In the transformed space the direction $\eta = T_x(\zeta)$ is obtained by projecting the vector DC orthogonally into the linear subspace $\{\eta:\ AD\eta = 0\}$. Thus η is the solution of the following least-squares problem.

$$\text{Minimize } ||DC - \eta||^2$$
$$\text{subject to } AD\eta = 0$$

Assuming A is of full row rank, the solution is

$$\eta_1 = [I - (DA^T(AD^2A^T)^{-1}AD]DC)$$

In the original space, the linear rescaling algorithm assigns to a point X the vector $\zeta_1 = \zeta_1(x) = D[[I - DA^T(AD^2A^T)^{-1}AD]DC]$ to define a search direction.

The projective rescaling algorithm is stated with respect to the linear programming problem in the canonical form.

$$\begin{aligned} &\text{Minimize } C^TX \\ &\text{subject to } AX = 0 \qquad \qquad \ldots \quad (95) \\ &\qquad e^TX = 1 \\ &\qquad X \geq 0 \end{aligned}$$

where $A \in \mathbb{R}^{m \times n}$, C, $X \in \mathbb{R}^n$ and $e = (1, \ldots, 1)^T \in \mathbb{R}^n$.

Let $X \in \mathbb{R}^n$ be such that $AX = 0$, $e^{Tx}x = 1$ and $X > 0$ and D is as defined above. Consider a transformation of space $T_x: R^n \rightarrow R^n$ given by

$$T_X(Y) = \frac{D^{-1}Y}{e^TD^{-1}Y}$$

Obviously $T_x(X) = e/n$. In the transformed space, the direction is obtained by projecting the vector DC into the null space of the matrix.

$$\bar{A} = \begin{pmatrix} AD \\ e^T \end{pmatrix}$$

Thus $\eta_p = [I - A^T(A\ A^T)^{-1}A]DC$

The null space of the matrix A equals the intersection of the null spaces of the matrices AD and e^T. However, e is orthogonal to every row of AD since $ADe = AX = 0$. This implies that η_p can be obtained by projecting on the null space of AD and then projecting the projection on null space of e^T. See lemma 21 below.

The search direction in the transformed space is thus given by

$$\eta_P = [I - A^T(A A^T)^{-1}A]DC$$
$$= [I - 1/n\ ee^T][I - DA^T(AD^2A^T)^{-1}AD]DC$$

The search directin in the original space is obtained as follows:

The algorithm move in the transformed space from the point e/n to a point of the form

$$q = e/n - \rho \frac{\eta_p}{||\eta_p||}$$

where ρ is a certain positive constant. The step in the original space is given by the vector

$$U = T_X^{-1}(q) - X$$

where

$$T_X^{-1}(q) = \frac{Dq}{e^T Dq}$$

$$Let\ \psi = \frac{\rho\eta_p}{||\eta_p||}$$

we have

$$U = \frac{De/n - D\psi}{e^T(De/n - D\psi)} - X$$

$$= - \frac{D\psi + (X^T\psi)X}{1/n - X^T\psi}$$

Since $e^T D = X^T$.

Now, ignoring the size of the step and considering just a vector η_p in the opposite direction of U, we have

$$\zeta_p = D\eta_p - (X^T\eta_p)X = [D - X X^T]\eta_p$$

Note that $X^T e = 1$, so we have

$$[D - X X^T][I - 1/n\ ee^T] = D - 1/n\ X\ e^T - X X^T + 1/n\ X X^T ee^T$$

$$= D - X X^T.$$

Thus the algorithm assigns to the point X the vector

$$\zeta_p = \zeta_p(x) = [D - X X^T][I - DA^T(AD^2A^T)^{-1}AD]DC$$

to define a search direction in the original space.

Lemma 21

Let $A \in \mathbb{R}^{m \times n}$ and $B \in \mathbb{R}^{m \times n}$ be matrices such that $AB^T = 0$. Under these conditiions, the orthogonal projection of any vector V $\in \mathbb{R}^n$ on the intersection of the null spaces of A and B can be obtained as follows: First, project V orthogonally into the null space of B and then project this projection orthogonally into the null space of A.

Proof

Without any loss of generality, we can assume that A and B are of full rank.

$$\text{Let } M = \begin{pmatrix} A \\ B \end{pmatrix}$$

Since $AB^T = 0$, it follows that

$$MM^T = \begin{pmatrix} AA^T & 0 \\ 0 & BB^T \end{pmatrix}$$

Also, since AA^T and BB^T are non-singular (even positive definite) MM^T is non-singular. The orthogonal projection of V into the null space of M is given by $[I - M^T(MM^T)^{-1}M]V$. It follows that $M^T(MM^T)^{-1}M = A^T(AA^T)^{-1}A + B^T(BB^T)^{-1}B$.

On the other hand, the sequence of projections stated in the lemma result in the vector $[I - A^T(AA^T)^{-1}A][I-B^T(BB^T)^{-1}B]V$. The lemma now follows since $AB^T = 0$.

4.1 Interior Point Algorithms Continuously extend to the boundary

The central feature of the interior point algorithms under consideration is the projection of a vector on a certain subspace. We focus more attention on the behaviour of the resulting vector as the current point of the algorithm tends to a boundary point.

Let A denote a constraint matrix of order m x n. Let $N=(1,2,...,n)$ and let I_1 and I_2 define a partition of N i.e. $I_1 \neq \phi$, $I_2 \neq \phi$, $I_1 \cup I_2 - N$ and $I_1 \cap I_2 = \phi$. Let A_i denote a submatrix of A consisting of the columns of A with indices in $I_i (i=1,2)$. Similarly, for any n-vector V, let V_1 and V_2 denote subvector of V. Let $D(V) = \text{diag}(V)$ and let C_1 and C_2 denote subvectors of C as defined above.

Given $X \in \mathbb{R}^n$, a step of the linear rescaling algorithm amounts to the evaluation of the orthogonal projection of a vector DC on a linear subspace $L(X) = \{Y:ADY = 0\}$. We look at the behaviour of this projection when X tends to a limit point x. The interesting case is when some of the components of x are zero. let I_1 denote the set of indices j for which $x_j \neq 0$. The orthogonal projection of DC on L(x) is equal to the point in L(x) which is closest to DC. Thus, it is the solution of the following optimization problem (where decision variables are the components of Y).

$$\text{Minimize } ||DC - Y||^2$$
$$\text{subject to } ADY = 0$$

With the notations intoruced above, the latter problem is equivalent to

$$\text{Minimize } ||D(x_1)C_1 - Y_1||^2 + ||D(x_2)C_2 - Y_2||^2$$
$$\text{subject to } A_1D(x_1)Y_1 + A_2D(x_2)Y_2 = 0$$

Note that D = D(x) to show its dependence on x.

Let us denote this projection by y(x) and also let $y_1(x)$ and $y_2(x)$ denote the restriction to the set of indices I_1 and I_2 respectively. Obviously, if x tends to x then the point D(x)C tends to the point D(x)C. The distance between D(x)C and y(x) is always less than or equal to $||D(x)C||$. Now, it follows that the point y(x) is bounded while x tends to x. Since x_2 tends to zero, the vector$A_2D(x_2)y_2(x)$ also tends to zero (since $y_2(x)$ is bounded). It is importnt to note that the point $y_1(x)$ is the orthogonal projection of the point D(x)C on the affine subspace.

$$\phi(X) = \{U | A_1D(x_1)U = -A_2D(x_2)Y_2(x)\}.$$

Now, in order to prove the continuity of the function $\phi(x)$ at x, we discuss the following propostions:

Proposition 22

Let ψ be a point-to-set mapping that takes points $X \epsilon \mathbb{R}^n$ to the subsets $\psi(x)$ of R^m. The mapping ψ is continuous at x if there exists a sequence $\{x^k\}$ in $\mathbb{R}^n$ such that $x^k \to x$, then the following statements are true:

(1) for any convergent sequence $\{z^k\}$, where $z^k \epsilon \psi(x^k\}$, necessarily $z = \lim z^k \epsilon \psi(x)$.

(2) for any point $z' \epsilon \psi(x^k)$, there exists a sequence $\{z^k\}$ converging to z' where $z^k \epsilon \psi(x^k)$.

Proposition 23

The mapping $\phi(x)$ is continuous at (x).

Proof

Let $\{x^k\}$ be any sequence converging to x. By the above assumption, $x_1 > 0$ and $x_2 = 0$. Notice that

$\phi(x) = \{U: A_1 D(x_1) U = 0\}$.

Obviously, condition (1) is satisfied since $A_2 D(x_2{}^k)\ Y_2(x^k\_$ tends to zero. In other words, the set β of all limits of sequence $\{U^k\}$ such that $U^k \epsilon \phi(x^k)$ is contained in the subspace $\phi(x)$. Thus β is a linear subspace which is in a sense the limit of the affine subspaces $\phi(x^k)$. The dimension of β is the same as the common dimension of all the $\phi(x^k)$'s for k sufficiently large. $\phi(x)$ is a linear subspace of the same dimension (since $x_1 > 0$). It follows that $\phi(x) = \beta$ and ths complete the proof.

Proposition 24

If x tends to x (where $x_1 > 0$ and $x_2 = 0$) then the point $y_1(x)$ tends to the projection of $D(x_1)C_1$ onthe linear subspace $\phi(x)$. (Proof can be found in Megiddo, Shub [30]).

Corollary 25

The limit of the orthogonal projection of D(x)C on the subspace $\{Z: AD(x)Z = 0\}$ is equal to the orthogonal projection of D(x)C on the subspace $\{Z: AD(x)Z = 0\}$.

Proof

Since the orthogonal projection of D(x)C onto $\{Z: AD(x)Z=0)\}$ is of the form $(y_1(x), 0)$ and $y_1(x)$ tends to $y_1(x)$ by proposition 24, it is sufficient to show that $y_2(x)$ tends to zero.

Suppose that $y_2(x)$ has a limit point $y^*_2 \neq 0$. Then $||D(x)C-(y_1(x), y_2(x))||^2$ tends to $||D(x)C - (y_1(x), 0||^2 + y^*_2||^2$. On the other hand, let $\tilde{y}_1(x) = D^{-1}(x_1)D(x_1)y_1(x)$. We have that $||D(x)C - (\tilde{y}_1(x), 0)||$ tends to $||D(x)C-(y_1(x), 0)||$. Together, these imply that for x sufficiently close to x.

$||D(x)C - (y_1(x), y_2(x)|| > ||D(x)C - \tilde{y}_1(x), 0||$.

However $(\tilde{y}_1(x), 0) \in \{Z: AD(x)Z - 0\}$.

The vector $\zeta = \zeta(x)$ asigned by the linear rescaling algoritm to a point x can be described as $\zeta(x) = D(x)y$ where y is the projection of the vector D(x) on the subspace $\{y | AD(x)y = 0\}$. Thus we have the following proposition.

Proposition 26

Suppose $x \in \mathbb{R}^n$ satisfying AX = b has positive components and tend to a point x such that $x_1 > 0$ and $x_2 = 0$.

Then the vector $\zeta(x)$ of the linear rescaling algorithm at $x > 0$ in the problem (94) tends to the vector $\zeta(x_1)$ assigned by this algorithm at x_1 in the problem.

$$\text{Minimize } C_1^T Z$$
$$\text{subject to } A_1 Z = b$$
$$Z \geq 0$$

(Proof can be found in Megiddo, Shub [30]).

The same argument above applies to the projective rescaling algorithm and the path following algorithm. For details see Megiddo, Shub [30]).

4.2 Behaviour near vectices

For the study of the behaviour of path following algorithm near vertices, it is convenient to consider the linear programming problem in standard form.

$$\text{i.e. Maximize } C^T X$$
$$\text{subject to } AX = b$$
$$X \geq 0$$

were $A \in \mathbb{R}^{m \times n}$ $(m \leq n)$, $C, X \in \mathbb{R}^n$ and $b \in \mathbb{R}^m$. Let B denote the square matrix of order m, consisting of the first m columns of A. We assume B is non-singular and $B^{-1}b > 0$. In other words, B is a non-degenerate feasible basis. Let N denote the matrix of order m x (n-m) consisting of the last (n-m) columns of A.

We denote the restriction of any n-vector V to the first m coordinates by V_B and its restriction to the last n-m coordinates by V_N. Thus, the vectors C_B, C_N, X_B and X_N are defined with respect to the vectors C and X. We denote by $D = D(x)$ a diagonal matrix (of order n) whose diagonal entries are the components of th vector X. Also, D_B and D_N are diagonal matrices of orders m and (n-m) respectively, corresponding to the vector X_B and X_N. We assume that both the primal and dual problems have feasible regions of full dimension. The path is defined whenever a pair of interior feasible solutions for the primal and dual problems is given. Thus let $X^\circ \epsilon \mathbb{R}^n$ be such that $AX^\circ = b$ and $X^\circ > 0$ and let $Y^\circ \epsilon\ \mathbb{R}^m$ be such that $A^T Y^\circ \geq C$. The path starting at (X°, Y°) is given by the equations.

$$x_j (A_j^T\ Y - C_j) = \mu x_j\ (A^T{}_j Y^\circ - C_j)$$

$$AX = b \qquad (j = 1, 2, \ldots, n).$$

It is obvious that for any point on this path, if we "restart" the path according to this definition then nothing changes since the products of complementary variables remain in the same proportions.

Let $w_j = x_j{}^\circ (A_j^T\ Y^\circ - C_j)$

we have $BX_B + NX_N = b$

which implies $X_B = B^{-1}\ (b - NX_N)$

Also, along the path

$$B^T Y - C_B = \mu \left(\frac{w_1}{x_1}, \ldots, \frac{w_m}{x_m} \right)^T$$

it is convenient to denote the vector in the right hand side of the latter by (w_B/x_B).

we now have

$$Y = B^{-T}\left[\mu\left(\frac{w_B}{X_B}\right) + C_B\right]$$

On the other hand, for every j

$$x_j = \frac{\mu w_j}{A_j^T Y - C_j}$$

Thus

$$= \mu \frac{w_j}{A_j^T B^{-T}\left[\mu\left(\frac{w_B}{X_B}\right) + C_B\right] - C_j}$$

$$= \mu \frac{w_j}{-C_j + A_j^T B^{-T}\mu\left(\frac{w_B}{X_B}\right)}$$

Suppose $B^{-1}b > 0$ is the unique primal optimal solution and $B^{-T}C_B$ is the unique dual optimal sulution. So the paths of the x_j's and y_i's converge to these points respetively. Asymptotically, as μ tends to zero, the "non-basic" variables, that is x_j, $j = m+1,\ldots,n$ are

$$x_j \sim \mu \frac{w_j}{A_j^T B_{CB}^{-T} - C_j} \qquad (j=m+1,\ldots,n)$$

The denominator in the right hand side is sometimes called the reduced cost with respect to the basis B, that is

$$C_j = C_j - A_j^T B^{-T} C_B.$$

So

$$x_j \sim \mu \frac{w_j}{C_j} \quad (j=m+1,\ldots,n)$$

Note tht if Y^o is close to the dual optimal solution $B^{-T}C_B$ then we have

$$X_j \sim \mu x_j^o \qquad (j=m+1,\ldots,n)$$

In a more explcit way, if we start close enought to an optimal solution, the path takes us appriximately in a straight line to the optimal solution. This is different from the affine and projective rescaling algorithms where all pahts tends to a unique asymptotic direction of approach to the optimal solution.

CONCLUSTION

In preceeding chapters, we have attempted to give an overview of the recent developments in linear programming. For practically useful algorithms with theoretical guarantees projective methods have a considerable advantage over path following methods. Indeed, line searches in practice allow large decreases in the potential functon to be realized and constant decreases are guaranteed from any interior feasible point.

The simplex algorithm practically seems to be more effective than the interior point algorithms for small linear programming problems. But the interior point algorithms become more effective and much faster than the simplex algorithm in solving very large linear programming problems. They even solve very big problems that the simplex algorithm cannot handle. The projective rescaling algorithm is for example 400 times faster than the simplex algorithm in solving the largest problem that the simplex algorithm can handle. Still in line, while the projective method make use of steepest descent direction in searching for new inerior feasible solutions from a current one, the path following algorithm makes use of the centering direction.

There are in deed many avenues for further work on this thesis. In particular, one can find out whether the complexity bounds of the various interior point algorithms can be further improved. One can also find out the maximum step length size of the projective rescaling algorithm that can give maximum effectiveness of the algorithm and also guarantees polynomiality.

Perhaps, there are open questions yet to be answered. Can a linear programming problem be solved as a strongly polynomial time algorithm? Also, can exponential functions be used in place of logrithmic functions in the path following algorithm? Good luck to who ever finds this field interesting.

REFERENCES

[1] Anstreicher, K.M., (1989), "A worst case in Karmarkar's Algorithm", Mathematics of Operations Research 14(2) pp. 294-302.

[2] Avriel, M., (1976), "Non Linear Programming: Analysis and Methods", Prentice Hall, Inc, Eaglewood Cliffs, New Jersey.

[3] Barnes, E.R., (1986). "A variation on Karmarkar's algorithm for solving linear programming problems", Mathematical Programming 36, 174-182.

[4] Bazaraa, M.S. and C.M. Shetty, (1979), "Non Linear Programming, Theory and Algorithms", John Wiley and Sons Inc, New York, U.S.A.

[5] Baele, E.M.L., 1954-1, "Linear Programming by the method of leading variables", Report of th Conference on Linear Programming, arranged by Ferranti, Ltd, London.

[6] Dantzig, G.B., (1963), "Linear Programming and Extensions", Princeton University Press, Princeton, N.J.

[7] David, G.L., (1973), "Introduction to Linear and Non Linear Programming", Addison-Wesley Publishing Company, Reading, Mascachusetts, Manlo Park, California.

[8] Dikin, I.I., (1974), "On the convergence of an iterative process", Unpraulyaemve Sistems 12, 54-60 (In Russia).

[9] Dikin, I.I., (1967Q0, "Iterative solution of problems of linear and quadratic programming", Doklady Akademija Nauk SSSR 174, 747-748 [English translation: Soviet Mathematics Doklady 8, 674-675].

[10] Dorfman, R.P.S. and R. Solow, (1958), "Linear Programming and Economic analysis", McGraw Hill Book Co., New York.

[11] Frisch, K.R., (1955), "The logarithmic potential method of convex programming", Unpublished manuscript, University Institute of Economics, Oslo, Norway.

[12] Gale, D. 1956-1. "Neighbouring vertices on a convex polyhedron", in H.W. Kuhn and A.W. Tucker (eds). Linear inequalities and Related Systems, Annals of Mathematics study No 38, Princeton University Press, Princeton, New Jersey, 1956 pp. 255-263.

[13] Gill, P.E., Murray, W., Saunders, M.A., Tomlin, J.A. and Wright, M;H., (July 1985), "On projected Newton barrier

methods for linear programming and an equivalence to Karmarkar's projective method". Technical Report SOL 85-11, Systems Optimization Laboratory, Department of Operations Research, Stanford University, Stanford, Calif.

[14] Goldman, A.J. and A.W. Tucker, 1956-1, "Polyhedral Convex Cones", Ibid, pp. 19-39.

[15] Gonzaga, C. (1988), "Search directions for interior linear programming methods", Memorandum VCB/ERL M87/44, Electronics Laboratory, College of Engineering, University of California, Berkeley, California.

[16] Gupta, R.K. (1986-87), "Linear Programming", Krishma Prakashan Mandir, Krishma Vihar, 11, Shiraji Road, Meerut (U.P), India.

[17] Hadley, G., (1962), "Linear Programming", Addison Wesley Reading, Mass.

[18] Kanti, S., P.K. Gupta, Man M., (1994), "Operations Research", Sultan Chand and Sons 4793/23, Daryaganj, New Delhi - 110 002.

[19] Karmarkar, N., (1984), "A new polynomial time algorithm for linear programming", Combinatorica 4, 373-395.

[20] Khachian, L.G., (1979), "A polynomial Algorithm for Linear Programming", Doklady Akad 20 pp. 191-194.

[21] Khachian, L.G., (1980), "Polynomial Algorithms in Linear Programming", (In Russia), Zhurnal Vvchissditd-noi Matematikii Matematisches Koi Fizizi 20, 51-68 [English translation: USSR Computational Mathematics and Mathematical Physics 20, 53-72].

[22] Kojima, M., Mizuno, S. and Yoshise, A., (1988), "A primal-dual interior point method for linear programming", in: Progress in Mathematical Programming, (N. Megiddo, ed.), Springer Verlay, Berlin, 29-47.

[23] Lemke, C.E. 1954-1. "The Dual Method of solving the linear programming problems", Naval Res. Logist. Quart., Vol. 1, 1954, pp. 36-47.

[24] Mangasarian, O.L., (1970), "Non linear programming", McGraw-Hill, New York.

[25] McCarthy, M.D. and R.C. Geary, (1964), "Elements of linear programming with Economic Applications", Charles Griffin and Company Limited, 42 Drury Lane, London, W.C.2.

[26] McDiamid, C., (1990), "On the improvement per iteration in Karmarkar's method for linear programming", Mathematical Programming 40, pp. 290-320.

[27] McLinden, L., (1980), "An analogue of Moreau's approximation theorem with application to non linear complementarity problem", Pacific Journal of Mathematics 88, pp. 101-161.

[28] __________, (1980), "The complementarity problem for maximal monotone multifunctions", in Cottle, R.W., Giannessi, F., and Lions, J.L. (eds.) Variational Inequalities and Complementarity problems, John Wiley and Sons, New York, pp. 251-270.

[29] Megiddo, N., (1988), "Patheways to the optimal set in linear programming", in: Progress in Mathematical Programming (N. Megiddo, ed.), Springer-Verlay, Berlin, 131-158.

[30] Megiddo, N. and Shub, M., (1986), "Boundary behaviour of interior point algorithms for linear programming", UBM Research Report RJ 5319, T.J. Watson Research Center, Yorktown Heights, New York.

[31] Mitchell, J.E. and Todd, M.J., (1986), "On the relationship between the search directions in the affine and projective variants of Karmarkar's linear programming algorithm", Technical Report No. 725, School of Operations Research and Industrial Engineering, Cornell University, Itacha, NY.

[32] Serge, L., (1968), "Calculus of several variables", Addison - Wesley Publishing Company, Reading Massachusetts, Menlo Park, California.

[33] Shinji, M., Masakazu, K. and Michael, J.T., (1995), "Infeasible-interior-point primal-dual potential-reduction algorithms for linear programming", SIAM J, Optimization, Vol. 5, No. 1, pp. 52-67.

[34] Singh, J.N. and Singh, D. (1992), "A modified termination rule for Karmarkar's agorithm", Paper presented in the fourth SIAM Conference on Optimization, Vol. held in Chicago (May, 1992).

[35] Singh, J.J., (1994), "Convergence in Karmarkar's algorithm: a review, International Journal of Mathematical Education in Science and Technology, Vol 25, No. 3, 333-341.

[36] Sonnevend, G., (1985), "An 'analytic centre' for polyhedrons for linear (smooth, convex) programming", in lecture notes in Control and Information Sciences, (Springer Verlag), New York, pp. 866-876.

[37] Todd, M.J., (1988), "Recent developments and new directions in linear programming", Technical Report No. 827, School of Operations Research and Industrial Engineering, Cornell University, Ithaca, New York, 14854.

[38] Todd, M.J., (1994), "Interior-point algorithms for semi-infinite programming", Revised manuscript, Mathematical programming 65, 217-245, School of Operations Research and Industrial Engineering, College of Engineering, Cornell University, Ithaca, NY 14853-3801, USA.

[39] Vanderbei, R.J., Meketon, M.S., and Freedman, B.A. (1986), "A modification of Karmarkar's linear programming algorithm", Algorithmica 1, 395-407.

[40] Von, N.J., 1937-1, "On a maximization problem" (Manuscript), Institute for Advanced Study, Princeton, New Jersey, November 1937.

[41] Walsh, G.R. (1975), "Method of optimization", John Wiley and Sons, London, England.