EXTENDED CUTTING PLANE ALGORITH FOR NON-LINEAR PROGRAMMING PROBLEMS

BY

OLASUNKANMI PAUL JIMOH
M.TECH/SSSE/2000/582

A THESIS SUBMITTED TO THE DEPARTMENT OF MATHEMATICS AND COMPUTER SCIENCE, FEDERAL UNIVERSITY OF TECHNOLOGY, MINNA, NIGERIA. IN PARTIAL FULFILMENT OF THE REQUIREMENT FOR THE AWARD OF MASTERS OF TECHNOLOGY (MTECH) DEGREE IN MATHEMATICS.

June, 2004

CERTIFICATION

This thesis titled – "EXTENDED CUTTING PLANE ALGORITHM FOR NON-LINEAR PROGRAMMING PROBLEMS", by Olasunkanmi Paul Jimoh, meets the regulations governing the award of the degree of Masters of Technology, in Mathematics, Federal University of Technology, Minna and is approved for its contribution to knowledge and literacy presentation

Dr. S. A. Reju
Supervisor

24-06-2004
Date

L. N. Ezeako
Head of Department.

24-6-2004
Date

Prof.(Mrs) H.O. Akanya
Dean, School of Science & Science Education

20th/9/04
Date

Prof. J. A. Abalaka
Dean, Postgraduate School

Date

Dr.(Mrs) O. O. Ugbebor
External Examiner

24-06-04
Date

DEDICATION

I dedicate this work to God Almighty, the God of all creation who saw me through and made it possible for me to complete this course. May His name be forever glorified, amen.

LORD JESUS YOU ARE TOO GOOD TO ME.

ACKNOWLEDGEMENTS

I wish to acknowledge the contributions of my supervisor, Dr S.A. Reju, in the research work for the pains he bore in supervising and correcting this work, his style of supervision made this project work a success.

I also want to express my sincere gratitude to all the staff of the Department of Mathematics and Computer Science, Federal University of Technology, Minna. Especially, the Head Of Department, Mr L.N. Ezeako and all my lecturers, namely Prof. K.R. Adeboye, Dr. NN. Akinwande, and Dr. Yomi Aiyesimi, for all their contributions towards my academic pursuit in the Department in particular and in the University at large, I shall always remain grateful to them all.

My sincere thanks also to my pastor, Pastor Seyi Ojobe, and my prayer group members for their support and prayers throughout this course, for reaching this point and for all the journeys.

My thanks also to Miss Janet and her colleagues at Boye Business Centre for painstakingly typing all the prekiminary works in this research.

Jimoh O.P.
Minna, June 2004.

TABLE OF CONTENTS

ABSTRACT

In this work, we used the Extended Cutting Plane Algorithm (Gradient Method) to solve Non–Linear Integer Programming Problem via linearization. The method was used to linearize both the objective and constraints functions and it was shown that the method gave more rapid convergence to the optimum solution than the Hookes and Jeev's or the Bound and Branch methods.

CHAPTER ONE

INTRODUCTION TO OPTIMIZATION THEORY

1.1 GENERAL INTRODUCTION

1.1.1 PREAMBLE

Optimization is a concept to describe optimal and best way of achieving result amongst alternatives under given conditions.

Optimization is an aspect of Operations Research, a branch of Mathematics which is concerned with the applications of scientific technique and methods to decision making problems and finding ways of establishing the best or optimal solution to such problems.

Areas of applications of Optimization theory includes among others: Constructions, Maintenance of Engineering systems, Cost – Profit Manufacturing problems and Transportation problems.

An Optimization problem can take the following form:

Find

$$x = \begin{Bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{Bmatrix}$$

which minimizes $f(x)$ subject to

$$g_j(x) \leq 0; \quad j = 1,\ 2,\ \ldots\ n$$

$$h_k(x) = 0; \quad k = 1,\ 2,\ \ldots\ n \tag{1.1}$$

$F(x)$ is called the objective function while $g_j(x)$ $h_k(x)$ are the constraints functions.

The Optimization problem (1.1) is said to be linear if the objective and constraints functions are all linear, and it is non–linear if any of them is of non–linear relationship.

1.1.2 LINEAR OPTIMIZATION PROBLEM

Linear Programming Problems (LPP) can be of all–integer values, mixed integer or zero–one problem. It is of the all–integer value type if all the design variables
$x_1, x_2, \ldots, x_n$
are constrained to take only integer values. It is of the mixed–integer type when some of the variables are constrained to take integer values; and it is of the zero–one type when all the design variables are allowed to take on valuesof either one or zero.

A linear optimization problem can take the form:

Optimize

$$f(x_1, x_2, \ldots, x_n) = \sum_{j=1}^{n} c_j x_j$$

subject to

$$\sum_{j=1}^{n} a_{ij} x_j \leq b_i, \quad i = 1, 2, \ldots m$$

$x_j \geq 0, \ \ j = 1, 2, \ldots n;$ x_j integers

(1.2)

If $m = n$, the problem (1.2) is an all–integer problem, and if $m < n$ it is of the mixed integer type.

Example

Minimize

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

subject to

$$3x_1 + x_2 \leq 10$$
$$2x_1 + 3x_2 = 6$$

(1.3)

1.1.3 NON–LINEAR OPTIMIZATION PROBLEM

When the objective and constraint functions are non–linear, the programming problem is said to be non–linear.

The non–linear programming problem can be (i) polynomial or (ii) general non–linear in nature.

The general non–linear programming problem can be all–integer or mixed integer type.

A Polynomial Programming problem can take the form:

Minimize

$$f(x) = \sum_{j=1}^{n_0} c_j \left(\Pi x_j^{P_{ij}}\right); \quad c_i > 0, \ \ x_j > 0$$

subject to

$$\sum a_{ik} \left(\Pi x_k^{a_{ik}}\right) \leq 0$$

$a_{ik} \geq 0$

(1.4)

where n_0, n_j denote the number of polynomial terms in the objective and j^{th} constraint function respectively.

The Geometric Programming problem take the form (1.4) above while the quadratic equivalent will take the form

Minimize

$$f(x) = c + \sum_{i=1}^{m} q_i x_i + \sum_{i=1}^{m}\sum_{j=1}^{n} Q_{ij} x_i x_j$$

subject to

$$\sum a_{ij} x_j \leq b_i, \quad i = 1,\ 2,\ \ldots\ m$$

$x_j \geq 0, \quad j = 1,\ 2,\ \ldots\ n;\ c_i,\ q_i,\ Q_{ij},\ a_{ij}$ and b_j are constants

(1.5)

The general quadratic programming problem can also take the form:
Minimize

$$z = x'Dx + C'x$$

subject to

$$Ax \leq b \text{ or } \geq$$

$$x \geq 0$$

(1.6)

where
D = n x n matrix which can be assumed symmetric
A = m x n matrix
b = m – component column matrix
C = n – component row vector

Example – Geometric Programming Problem

Minimize

$$f(x) = \rho w \frac{\pi}{4}(d_1^2 + d_2^2 + d_3^2 + d_4^2$$

$$+d_1d_2 + d_1d_3 + d_1d_4 + d_2d_3 + d_2d_4 + d_3d_4)$$

subject to

$$C_i \approx \frac{nd_i}{2}\left(1 + \frac{w_i}{w}\right) + \frac{w_i - d_i^2}{4a} + 2a \tag{1.7}$$

Example – Quadratic Programming Problem

Minimize

$$f(x) = \frac{1}{2}(x_1^2 + x_2^2 + x_3^2 + x_4^2)$$

subject to

$$\begin{aligned} g_1(x) &= x_1 + 2x_2 + 3x_3 + 4x_4 = 0 \\ g_2(x) &= x_1 + 2x_2 + 5x_3 + 6x_4 - 15 = 0 \end{aligned} \tag{1.8}$$

The general non-linear programming problem can be expressed in the form:

Find

$$x = \left\{ \begin{array}{c} x_1 \\ x_2 \\ \vdots \\ x_n \end{array} \right\} = \left\{ \begin{array}{c} x_i \\ x_c \end{array} \right\}$$

which minimizes $f(x)$ subject to

$$g_j(x) \geq 0; \quad j = 1,\ 2,\ \ldots\ n$$

$$x_i \in S_i \text{ and } x_c \in S_c \tag{1.9}$$

The vector variables $\{x_i\}$ and $\{x_c\}$ are the sets of integer and continuous variables respectively.

Example – General Non–Linear Programming Problem

Minimize

$$f(x_1,\ x_2) = x_1^2 + x_2^2$$

subject to

$$g_1(x_1,\ x_2) = (x_1 - 2)^2 + x_2 - 1 = 0$$
$$g_2(x_1,\ x_2) = x_1^2 + (x_2 + 1)^2 - 2 = 0 \tag{1.10}$$

1.2 CONSTRAIN FUNCTIONS

1.2.1 TYPES OF CONSTRAINTS

An Optimization problem that is subject to certain restrictions is said to be a constrained optimization problem, it is unconstrainned if otherwise.

Any constraint which is represented by the behaviour or performance of the system is called behaviour of functional constraint while any constraint represented by physical limitations on the design variable like availability, portability and fabricability is called side or geometric constraint.

1.2.2 CONSTRAINT SURFACES

The constraint surfaces are classified as follow:

(i) $g_j(x) = 0$ is the hypersurface called the composite constraint surface. It divides the design space into two regions: $g_j(x) < 0$ $g_j(x) > 0$

(ii) $g_j(x) > 0$ is called the infeasible or unacceptable region.

(iii) $g_j(x) < 0$ is called the feasible or acceptable region.

1.3 CLASSIFICATION OF NON–LINEAR OPTIMIZATION PROBLEM

1.3.1 CLASSIFICATION BASED ON THE NATURE OF CONSTRAINTS

The non–linear optimization problem can take any of the following form:

(i) Non–linear optimization problem in one–dimension.

This can be of the form:
Minimize

$$f(x) = \sum_{i}^{n} a_i x_i$$

subject to

$$g_j(x) = \mu_j(x) - \delta_j \leq 0$$
$$x_i \geq 0, \;\; j = 1,\, 2,\, \ldots\, n \;\; i = 1,\, 2,\, \ldots\, m \tag{1.11}$$

(ii) Non–linear optimization problem with inequality constraint.
This can be of the form:
Minimize

$$f(x) = (x_1 - 1)^2 + (x_2 + 2)^2$$

subject to

$$3 - x_1 - 2x_2 > 0$$
$$2 - 3x_1 + x_2 > 0$$
$$x_1,\;\; x_2 \geq 0 \tag{1.12}$$

(iii) Non–linear optimization problem with equality constraint.
This can be of the form:
Minimize

$$f(x) = (x_1,\, x_2,\, \ldots,\, x_n)$$

subject to

$$g_j(x) = 0; \; j = 1,\, 2,\, \ldots\, m \tag{1.13}$$

Usually m = n. if m ¡ n the problem is said to be undefined and there will be no solution.

(iv) Non–linear optimization problem which are unconstrained.
This can be of the form:
Minimize

$$f(x) = c_i + \sum \sum Q_{ij} x_i x_j$$

$$j = 1,\ 2,\ \ldots\ n \quad i = 1,\ 2,\ \ldots\ m$$

$$(1.14)$$

1.3.2 CLASSIFICATION BASED ON THE NATURE OF DESIGN VARIABLES AND SYSTEMS

(i) Non–linear optimization problem which are based on the nature of design variables.
i.e. Finding values to a set of parameters which make some prescribed function of these parameters minimum or maximum subject to certain constraint.
e.g. find the maximum load $(x_1 + x_2 + x_3)$ that can be supported by the system if the weight of the supporting beam and the ropes are negligible.
The design vector is

$$x = \begin{Bmatrix} x_1 \\ x_2 \\ x_3 \end{Bmatrix}$$

This can be of the form:
Minimize

$$f(x) = -(x_1 + x_2 + x_3)$$

subject to $x_1 + x_2 + x_3 \geq 0$
or $x_3 - (x_1 + x_2) \geq 0$
or $x_1,\ x_2,\ x_3 \geq 0$

$$(1.15)$$

e.g. If the cross sectional dimension of a rectangular beam is allowed to vary along its length:

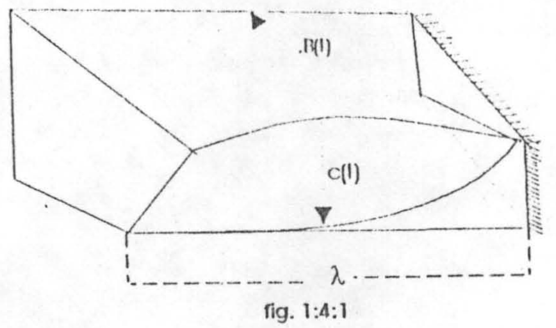


Figure 1.1 A rectangular beam

The optimization problem can be stated as:

Find

$$x(t) = \left\{ \begin{array}{c} b(t) \\ c(t) \end{array} \right\}$$

which minimizes

$$f(x(t)) = \int b(t)c(t)dt$$

subject to

$$\lambda_{tp}[x(t)] \leq \delta_{\max}$$

$$0 \leq t \leq 1, \quad b(t) \geq 0, \quad c(t) \geq 0 \qquad (1.16)$$

(ii) Non-linear optimization problem which are based on the equations of design system.

i.e. Equations expressing the objective and constraint functions

(iii) Non-linear optimization problem which are based on the physical structure of the design system.

e.g. Bringing the speed of a system of motion under control

1.4 OPTIMALITY AND CONTROL PROBLEM

There are some control theory problems which are designed to bring about some physical changes in a system. e.g. applying break to a moving vehicle suddenly, to avoid collision or bring about minimum impact or force, [2], [10].

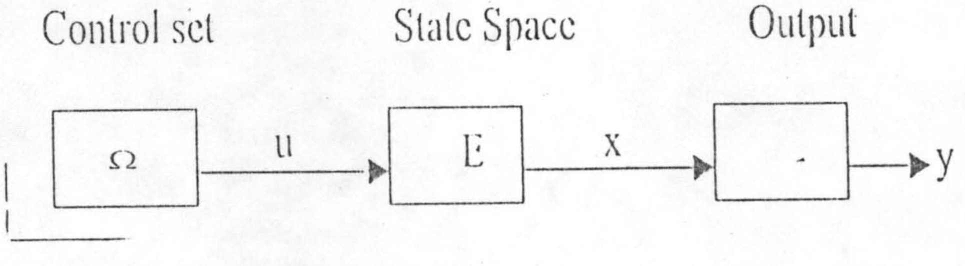


Figure 1.2 – Engineering design of locomotive

An example is the engineering design of a locomotive as depicted in the figure 1.2 above.

A linear optimization problem arising from above is:

Optimize

$$x(t) = u_1(t) - u_2(t)$$

and by transformation may assume

$$x(t) = Ax(t) + Bu(t)$$

where

$$x(0) = \begin{Bmatrix} x_0 \\ y_0 \end{Bmatrix}$$

$$x = \begin{Bmatrix} x_1 \\ x_2 \end{Bmatrix}$$

$$A = \begin{Bmatrix} 0 & 1 \\ 0 & 0 \end{Bmatrix}$$

$$B = \begin{Bmatrix} 0 & 0 \\ 1 & -1 \end{Bmatrix}$$

$$u = \begin{Bmatrix} u_1 \\ u_2 \end{Bmatrix} \tag{1.17}$$

A general non–linear problem arising from the above can take the form:

$$x'(t) = f(x,\ u)$$
$$f(0,\ 0) = 0 \tag{1.18}$$

For example

$$x_1' = -x_1 + u_1$$
$$x_2' = -2x_1 + u_1 + 2u_2$$

(1.18) can be expressed as

$$x'(t) = f\{t,\ x(t),\ u(t)\} \tag{1.19}$$

1.5 AIMS AND OBJECTIVE OF THIS STUDY

The aims and objectives of this study are:

1. To appraise the Extended Cutting Plane Methods in Optimization theory and

2. To use the method for the linearization of non–linear objective and constraint functions in Optimization theory.

CHAPTER TWO

GENERAL OPTIMIZATION TECHNIQUE FOR NON–LINEAR PROGRAMMING PROBLEMS

2.1 INTERIOR PENALTY FUNCTION METHOD

Given the problem

Find

$$x = \begin{Bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{Bmatrix} = \begin{Bmatrix} x_I \\ x_c \end{Bmatrix}$$

which minimizes $f(x)$ subject to the constraints

$$g_j(x) \geq 0; \quad j = 1,\ 2,\ \ldots\ m$$

$$x_c \in S_c; \quad x_I \in S_I \tag{2.1}$$

x_I and x_c are two vectors representing integer and continuous variables respectively.

S_I and S_c denote the feasible sets of integer and continuous variables respectively. We expect that either of these sets can be empty if the variables are all integers or all continuous [12].

To introduce penalty parameters, we define

Minimize $Q(x,\ r_k,\ S_k)$ as

$$Q(x,\ r_k,\ S_k) = f(x) + r_k \sum_{j=1}^{m} G_j[g_j(x)] + S_k Q(x_d) \tag{2.2}$$

r_k is the weighing factor called the penalty parameter and

$$r_k \sum_{j=1}^{m} G_j[g_j(x)] \tag{2.3}$$

is the contribution of the constraints to the Q_k function and is equivalent to

$$r_k \sum_{j=1}^{m} G_j[g_j(x)] + r_k \sum_{j=1}^{m} \frac{1}{g_j(x)} \tag{2.4}$$

this term is positive for all x satisfying
$g_j(x) > 0$ and $\to \infty$ whenever any of the constraints tend to zero value. This implies that if the minimization of the Q_k function starts from the feasible point, the point remains in the feasible region always.

The term $S_k Q_k$ is the penalty term with the weighing factor or penalty parameter and $Q_k(x_d)$ will be the penalty anytime variables in x_d take values other than integer values.

Therefore

$$Q_k(x_d) = \begin{cases} 0 \text{ if } x_d \in S_d \\ \mu > 0 \text{ if } x_d \notin S_d \end{cases} \tag{2.5}$$

Essentially the function is to be minimized for a sequence of values of r_k and S_k such that for $k \to \infty$ we obtain:

$$\min Q_k(x,\, r_k,\, S_k) \to \min f(x)$$

$$g_j(x) \geq 0; \quad j = 1,\, 2,\, \ldots\, m$$

and

$$Q_k(x_I) \to 0 \tag{2.6}$$

Generally the penalty function method can be defined as:

Given $f(x)$, $g_1(x)$, $g_2(x)$, $\ldots$, $g_m(x)$ having continuous first partial derivatives in $\mathbb{R}^n$ then a constraint problem

Minimize $f(x)$
subject to
$g_1(x) \leq 0,\ g_2(x) \leq 0,\ \ldots,\ g_m(x) \leq 0$

$$(2.7)$$

can be solved as follow:

1. For each positive integer k suppose that $x_k^\star$ is a global minimizer of
$$P_k(x) = f(x) + k\sum_{i=1}^{m}[g_i(x)]^2 \tag{2.8}$$

2. It is to be shown that subsequent $(x_k^\star)$ will converge to the solution $x^\star$.

2.2 APPLICATION OF PENALTY FUNCTION METHOD TO CONVEX PROGRAMMING PROBLEM

Suppose that $f(x),\ g_1(x),\ g_2(x),\ \ldots,\ g_m(x)$ are convex functions with continuous first partial derivatives on $I\!R^n$ and suppose that $f(x)$ is coersive, i.e.
$$\lim_{[x]\to+\infty} f(x) = +\infty \tag{2.9}$$

If the convex programme given by equations (2.7) is consistent then the dual programme is also consistent and the minimum value of the programme is given as [8], [12]

$$\inf f(x)\ :\ g_i(x) \leq 0;\ i = 1,\ 2,\ \ldots,\ m;\ x \in I\!R^n \tag{2.10}$$

If the constraint functions $g_i(x)$ have continuous first partial derivative, so also is the function given by

$$h(x) = [g^\star(x)]^2 \tag{2.11}$$

and

$$\frac{\partial h(x)}{\partial x_i} = 2g^\star(x)\frac{\partial g(x)}{\partial x_i};\ i = 1,\ 2,\ \ldots,\ n \tag{2.12}$$

and also the function given by

$$P_k(x) = f(x) + k\sum_{i=1}^{m}[g_i^\star(x)]^2$$

for a bounded sequence $\{x_k\}$

$$P_k(x_k) = \min\{P_k(x) : x \in I\!R^n\} \tag{2.13}$$

If $\{x_{k_j}\}$ is a convergent subsequence then

$$\nabla P_k(x) = \nabla f(x) + k\sum_{i=1}^{m} 2g_i(x)\nabla g_i(x) \tag{2.14}$$

$\{x_{k_j}\}$ are the convergent subsequence of $\{x_k\}$ and their limits are the solutions of the problem

If $\{x_{k_j}\}$ is the minimizer for $P_{k_j}(x)$ then

$$0 = \nabla P_{k_j}(x_{k_j}) = \nabla f(x_{k_j}) + \sum_{i=1}^{m} 2k_j g_i(x)\nabla g_i(x) \tag{2.15}$$

2.3 METHOD OF TRANSFORMATION OF VARIABLES

Given a quadratic or polynomial programming problem of the form:

Minimize $f(x) = x^2 + 2x$
subject to

$$g(x) = 1 - x \leq 0,\ \ x \in I\!R \tag{2.16}$$

Solution

If $x^\star = 1$, the minimum value is given by $x = 3$

Applying penalty function method as in (2.8)

$$P_k(x) = F(x) + k\sum_{i=1}^{m}[g_i(x)]^2$$

$$P_k(x) = x^2 + 2x + k[(1-x)]^2$$
$$= x^2 + 2x + k - 2kx + kx^2$$

or

$$= \begin{cases} x^2 + 2x + k[1-x]^2 & \text{for } \ x = 1 \\ x^2 + 2x & \text{for } \ x > 1 \end{cases}$$

$P_k(x)$ is continuously differentiable everywhere, it is an increasing function at $x = 1$ and has a unique minimizer $x^\star$ at $x = 1$; then

$$0 = P_k'(x) = 2x + 2 - 2k(1-x)$$

$$x + 1 - k + kx = 0$$
$$x = \frac{k-1}{1+k}$$

taking limit, $x_k^\star = 0$ as $k \to \infty$
i.e. the sequence converges and the higher the value of k the closer is

$$x_k^\star = \frac{k-1}{1+k}$$

to becoming feasible.

2.4 ITERATIVE (NON-GRADIENT) METHOD USING HOOKE AND JEEVE ALGORITHM [11]

We consider an optimization problem of the form

Minimize $f(x_1, x_2, \ldots, x_n)$; a multrivariable non-linear function, the algorithm is as follow:

Step 1

Choose an initial base point b_i and step length h_j for the respective x_j and for numerical accuracy h_j can be chosen to equalise the quantities

$$f(b_i + h_j e_j) - f(b_i)$$

Step 2

After evaluating $f(b_i)$, call it exploratory success (s) if it gives a decrease in the value of f(x) and it is failure (f) if otherwise.

Step 3

Exploratory move for the variable x_1

E(i) - Evaluate $f(b_i + h_j e_j)$ if the move from b_i to $b_i + h_j e_j$, be a success; replace the base point b_i by $b_i + h_j e_j$, then evaluate $f(b_i + h_j e_j)$ otherwise, i.e. failure retain the original base point b_i.

E(ii) - Repeat E(i), for the variable x_2 by considering variables $b_2 + h_2 e_2$ from the point $b_1 + h_1 e_1$ considered to be a success in E(i).

Applying this procedure to each variable x_j in turn to arrive finally at new base point b_n after 2(n+1) functions.

E(iii) - If $b_2 = b_1$ for the step length h_j return to E(i) and terminate the algorithm when the step length have been reduced to a prescribed level.

Step 4

P(i) - Move from b_2 to $P - 1 = 2b_2 - b_1$ and it can continue with new sequence of exploratory move about P_1.

P(ii) - If the lowest function value obtained during the pattern and exploratory process of P(i) is less than $f(b_2)$ then a new base point b_2 has been reduced then return to P(i) increasing the suffixes by a unit otherwise the move is abandoned i.e. the pattern move from b_3 then continue with a new sequence of exploratory move about b_2.

Step 5

Stop the iteration when the chosen stopping condition is recorded e.g. $h_1 = h_2 < 1/4$

Example

Minimize $f(x) = 4x_1^2 - 3x_1x_2 + x_2^2 + 3x_1 + x_2$

Solution

take $b_1 = (0,\ 0)$ as the initial base point
$h_1 = h_2 = 1$ as initial step length

Denote exploratory move by $E(x_r)$ about the point x_1 and $P(b_r)$ pattern move from the base point b_r, let s and f denote success and failure when $f(x_r)$ is evaluated at x_r.
$f(b_1) = 0$ for $E(b_1)$
$f(1,\ 0) = 7$ (f)
$f(-1,\ 0) = 1$ (f)
$f(-1,\ 1) = 9$ (f)
$f(-1,\ -1) = -2$ (s)

$E(b_1)$ is a success at $(-1,\ -1)$, the new base point is
$b_2 = (-1,\ -1)$ and $f(b_2) = -2$
$f(-1,\ -2) = -3$ (s)
$f(-1,\ -1) = -2$ (s)
$f(-1,\ -3) = -2$ (s)
$f(-1,\ 3) = 22$ (f)
then $b_2 = (-1,\ -2)$ as the new base point, $f(b_2) = -2$
Making a further pattern move
$P_2 = 2b_2 - b_1 = (-1,\ -3)$ $f(P_2) = -2$
For $E(P_2)$ decreasing the step length by 1/2
$f(-1/2,\ -2) = -2$ (s)
$f(-1/2,\ -1/2) = -4$ (s)
$f(-1/2,\ 3/2) = 19/2$ (f)
$P_2 = (-1/2,\ -3/2) = -4$ (s) is the best pattern move then
$P_3 = 2b_3 - b_2 = (0,\ -2)$ $f(P_3) = 2$
Evaluate $E(b_3)$ i.e
$f(-1/4,\ -1/2) = -19/8$ (s)
$f(-1/4,\ -1/4) = -17/8$ (s)
$f(-1/4,\ -3/4) = -10/4$ (s)
$f(-1/4,\ 3/4) = -1$ (s)

The function $f(x) = 4x_1^2 - 3x_1x_2 + x_2^2 + 3x_1 + x_2$ can be minimized at $(-1/4,\ -3/4)$ being the least exploratory move.

Since the variables are independent each can be allowed to vary to obtain

$$\frac{\partial f}{\partial x_{m+1}} = \ldots = \frac{\partial f}{\partial x_n} = 0 \tag{2.21}$$

Using (2.18) and (2.19) we obtain

$$\frac{\partial F}{\partial x_j} = \frac{\partial f}{\partial x_j} + \sum_{i=1}^{n} \lambda_j \frac{\partial g_j(x_i)}{\partial x_j} = 0 \tag{2.22}$$

and

$$\frac{\partial F}{\partial \lambda_j} - g_j(x) = 0 \tag{2.23}$$

and so

$$\frac{\partial f}{\partial x_j} = 0, \quad \frac{\partial f}{\partial \lambda_j} = 0 \tag{2.24}$$

Given (2.17) and (2.18) a sufficient condition for $f(x)$ to have a point $x^\star$ is that

$$Q = \sum_{i=1}^{m} \sum_{j=1}^{n} h_i h_j \frac{\partial^2 f(x,\, \lambda)}{\partial x_i \partial x_j} \tag{2.25}$$

or the Hessian

$$H = \frac{\partial^2 f(x,\, \lambda)}{\partial x_i \partial x_j} \tag{2.26}$$

evaluated at $x = x^\star$ must be positive definite

2.6 KUHN TUCKER THEORY ON OPTIMALITY

The Kurn Tucker theory gives necessary condition for any quadratic programming problem to have relative or global maximum, [4], [12]:

Given the problem:
Maximize $z = x'Dx + Cx$
subject to

$$g(x) = Ax \leq b$$

A point $(x^\star, \lambda^\star)$ is a global maximum if

$$f(x, \lambda) = x'Dx + C'x + \lambda^\star(b - Ax)$$

and

$$\partial f(x^\star, \lambda^\star) = 2D + C - A\lambda^\star \leq 0$$

The following conditions can be imposed:

(i) $k - T(i)$,

$$v^\star = \nabla_x f(x^\star, \lambda^\star) \leq 0$$

(ii) $k - T(ii)$,

$$v^\star x^\star = \nabla_x f(x^\star, \lambda^\star) x^\star \leq 0$$

(iii) $k - T(iii)$,

$$v_j^\star x_j^\star = \nabla_x f(x^\star, \lambda^\star) \leq 0$$

(iv) $k - T(iv)$,

$$\lambda_j^\star x_{n+1}^\star = \nabla_x f(x^\star, \lambda^\star) \lambda^\star \leq 0$$

CHAPTER THREE

CUTTING PLANE METHOD IN OPTIMIZATION

3.1 PREAMBLE

The cutting plane method is developed to solve non–linear programming problems [1], [12]. The problem is linearized using the Taylor series expansion which leads to the approximation of the feasible region by the linearized envelope or region.

3.2 GRAPHICAL SOLUTION TO LINEAR PROGRAMMING PROBLEM (INTERIOR)

Given the problem

Minimize $z(x_1,\ x_2) = -(5x_1 + 7x_2)$
subject to

$$4x_1 + 11x_2 \leq 77$$
$$3x_1 - 2x_2 \leq 12$$
$$x_1 > 0,\ x_2 > 0$$

x_1 and x_2 are integers.

(3.1)

Table 3.1

x_1	-2	-1	0	1	2	3	4	5
$(77 - x_1)/11$	7.7	7.3	7.0	6.6	6.3	6.0	5.6	5.2
$(3x_1 - 12)/2$	-9	-7.5	-6	-4.5	-3	-1.5	0	1.5

The feasible points in the feasible region are:
(0, 7), (0, −5.5), (3.5, 0), (6.5, 4.5)
Simultaneously, the solutions to the constraints are

$$x_1 = \frac{858}{123} = 6.6$$

$$x_2 = \frac{183}{41} = 4.46$$

These are fractions but many real life problems are integers. So the variabl;es are truncated.

$$z(x_1,\ x_2) = -(5x_1 + 7x_2)$$

$$= -(5 \times 4.46 + 7 \times 6.6) = -68.5$$

max $z(x_1,\ x_2) = 68.5$

$$z(x_1,\ x_2) = -(5x_1 + 7x_2)$$

$$= -(5 \times 4 + 7 \times 7) = -69$$

max $z(x_1,\ x_2) = 69$

3.3 GOMORY'S CUTTING PLANE METHOD

From the graphical illustration below of the last problem in section 3.3, the feasible region is AOBCD with solution to $z(x_1,\ x_2) = -68.5$, this is without integer requirement. But when truncated to integer value
$x_1 = 4$ $x_2 = 7$ $z(x_1,\ x_2) = -69$
when
$x_1 = 5$ $x_2 = 7$ $z(x_1,\ x_2) = -74$
using $x_1 = 4$ and $x_2 = 7$ to reduce or cut the original feasible region AOBCD to AOCEF and further cutting could reduce the region. This approximation lead to to a more feasible solution.

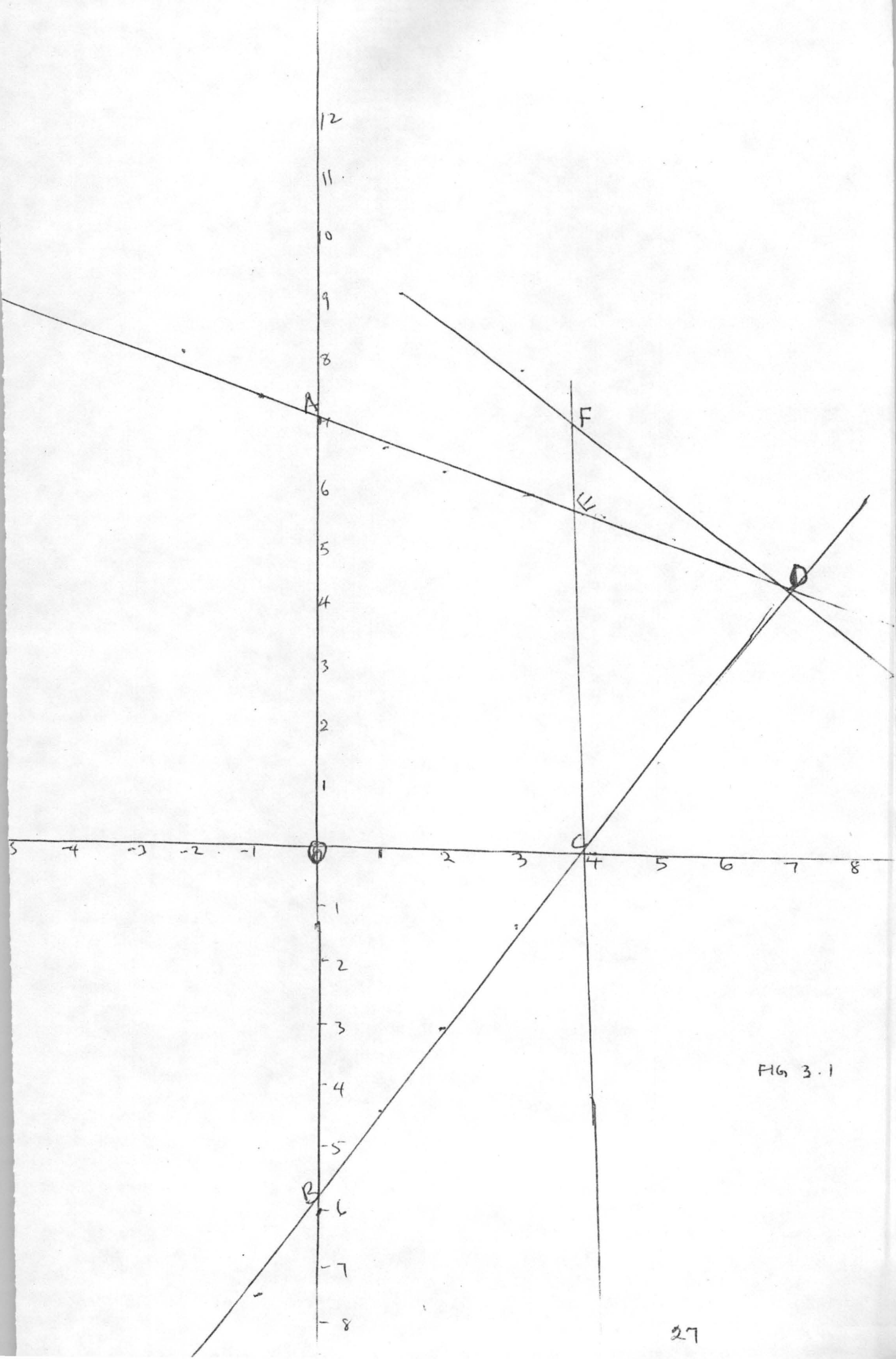


FIG 3.1

3.4.1 GOMORY'S CUTTING PLANE ALGORITHM

An algorithm was developed by Gomory [1], [5], [11]; to solve All integer or mixed integer programming problem with rational data. Consider the problem

Optimize $z = C^T x$
subject to

$$Ax = B$$

$$x \geq 0$$

$x_1,\ x_2,\ \ldots,\ x_q$ integers.
where $x = (x_1,\ x_2,\ \ldots,\ x_q,\ \ldots,\ x_n)^T$.
and
$C = n \times 1$ matrix
$A = n \times n$ matrix
$B = m \times 1$ matrix

It is All–integer if $q = n$

3.4.2 CONSTRUCTION OF GOMERY'S CUTS OR CONSTRAINTS FOR ALL INTEGER PROGRAMMING PROBLEMS

If the associated linear programming problem is solved with one of the variables resulting in non–integer form, suppose that the variable is x_i and is occuring in the i^{th} row of the optimal tableau, let the corresponding equation of such row be given by

$$x_i + \bar{a}_{j1}y_1 + \bar{a}_{j2}y_2 + \ldots + \bar{a}_{jp}y_p = \bar{b}_j \tag{3.2}$$

where
$y_k,\ k = 1,\ 2, \ldots,\ p$ are the basic variables and

$\bar{a}_{jk}$, $k = 1,\ 2, \ldots,\ p$ are the coefficients of y_k in this j^{th} row and $\bar{b}_j$ is the value of x_i, the solution is thus given by:

$$x_i = b_j - \alpha_{j1}y_1 - \alpha_{j2}y_2 - \ldots - \alpha_{jp}y_p \tag{3.3}$$

If $[\alpha]$ is the largest integer not greater than $\alpha \in I\!R$

$$\alpha = [\alpha] + \alpha'$$

e.g

$$3\frac{1}{3} = [3] + \frac{1}{3}'$$

With this definition, the solution now takes the form:

$$x_i = [\bar{b}_j]+\bar{b}_j'-\{([\bar{a}_{j1}]+\bar{a}_{j1}')y_1+([\bar{a}_{j2}]+\bar{a}_{j2}')y_2+\ldots+([\bar{a}_{jp}]+\bar{a}_{jp}')y_p\} \tag{3.4}$$

Collecting the integer terms gives:

$$x_i = [\bar{b}_j] - [\bar{a}_{j1}]y_1 - [\bar{a}_{j2}]y_2 - \ldots - [\bar{a}_{jp}]y_p$$

$$+\{(\bar{b}_j' - \bar{a}_{j1}')y_1 - (\bar{b}_j' - \bar{a}_{j2}')y_2 - \ldots - (\bar{b}_j' - \bar{a}_{jp}')y_p\} \tag{3.5}$$

and this gives the first part as

$$x_i = \{(\bar{b}_j) - (\bar{a}_{j1})y_1 - (\bar{a}_{j2})y_2 - \ldots - (\bar{a}_{jp})y_p\} \tag{3.6}$$

and it is an integer if all variables $y_1,\ y_2,\ \ldots,\ y_p$ are integers which is true by assumption.

For x_i to be an integer the second part

$$\bar{b}_j' - \{\bar{a}_{j1}'y_1 - \bar{a}_{j2}'y_2 - \ldots - \bar{a}_{jp}'y_p\} \tag{3.7}$$

must be an integer but $0 < \bar{b}_j' < 1$ as $\bar{b}_j'$ was assumed to be non–negative integer. Also because $0 \leq \bar{a}_{ji}' < 1$ for $i = 1,\ 2,\ \ldots, p$. Hence as the $y_1,\ y_2,\ \ldots,\ y_p$ are constrained to be non–negative integers it follows that

$$\bar{b}_j - \{\bar{a}_{j1}y_1 - \bar{a}_{j2}y_2 - \ldots - \bar{a}_{jp}y_p\} \leq 0 \tag{3.8}$$

holds in any feasible region or integer solution and we introduce a slack variable x_k such that

$$\bar{b}_j - \{\bar{a}_{j1}y_1 - \bar{a}_{j2}y_2 - \ldots - \bar{a}_{jp}y_p\} + x_k = \tag{3.9}$$

x_k as an integer, equation (3.9) will now be added to the final tableau of the set of constraints to obtain optimal solution to the modified LPP using simplex method algorithm.

Given the problem in (3.4.1), suppose that one of the constraints has a non–integer variable, then equation (3.4) can be written as

$$[\bar{b}_j] + b'_j - x_r = \sum_{j=1} \bar{a}_{jk}x_k \tag{3.10}$$

Since not all the variables y_k may be constrained to be integer, then let

$$S^+ = k; \;\; a_{jk} \geq 0$$

$$S^- = k; \;\; a_{jk} \leq 0$$

(3.10) then takes the form:

$$[\bar{b}_j] + b'_j - x_r = \sum_{S^+} \bar{a}_{jk}x_k + \sum_{S^-} \bar{a}_{jk}x_k \tag{3.11}$$

and two cases emerge:

Case 1

$$[\bar{b}_j] - \bar{b}'_j - x_r < 0 \tag{3.12}$$

Case 2

$$[\bar{b}_j] - \bar{b}'_j - x_r > 0 \tag{3.13}$$

<u>Case 1</u>

$$[\bar{b}_j] - \bar{b}'_j - x_j < 0 \tag{3.14}$$

Since $[\bar{b}_j]$ is an integer, x_j is constrained to be an integer in a feasible region and $\bar{b}'_j$ is a non–negative function hence

$$[\bar{b}_j] - x_j$$

must be non–negative integer v, say.

$$[\bar{b}_j] - \bar{b}'_j - x_j = \bar{b}_j - v \tag{3.15}$$

thus

$$\bar{b}_j - v = \sum_{k \in S_+} \bar{a}_{jk} y_k + \sum_{k \in S_-} \bar{a}_{jk} y_k \tag{3.16}$$

Since $v \geq 1$, then

$$\bar{b}_j - v \geq \sum_{k \in S_+} \bar{a}_{jk} y_k + \sum_{k \in S_-} \bar{a}_{jk} y_k \tag{3.17}$$

From the definition of S_+ and since $y_k \geq 0$ for all k we have that

$$\bar{b}_j - 1 \geq \sum_{k \in S_-} \bar{a}_{jk} y_k \tag{3.18}$$

as $\bar{b}_j - 1 < 0$ we get that

$$1 \leq (\bar{b}_j - 1)^{-1} \sum_{k \in S_-} \bar{a}_{jk} y_k \tag{3.19}$$

which gives

$$\bar{b}_j \leq \bar{b}_j (\bar{b}_j - 1)^{-1} \sum_{k \in S_-} \bar{a}_{jk} y_k \tag{3.20}$$

<u>Case 2</u>

$$[\bar{b}_j] - \bar{b}'_j - x_j \geq 0 \tag{3.21}$$

As x_j is constrained to be an integer in a feasible region we have that

$$[\bar{b}_j] - \bar{b}'_j - x_j = \bar{b}'_j + w \tag{3.22}$$

where $w \in (1, 2, \ldots)$ thus

$$\bar{b}'_j + w = \sum_{k \in S_+} \bar{a}_{jk} y_k + \sum_{k \in S_-} \bar{a}_{jk} y_k \tag{3.23}$$

Since $w \geq 0$, then

$$\bar{b}'_j \leq \sum_{k \in S_+} \bar{a}_{jk} y_k + \sum_{k \in S_-} \bar{a}_{jk} y_k \tag{3.24}$$

From the definition of S_+ and since $y_k \geq 0$ for all k we have that

$$\bar{b}_j - 1 \leq \sum_{k \in S_-} \bar{a}_{jk} y_k \tag{3.25}$$

which gives

$$\bar{b}'_j \leq \bar{b}'_j(\bar{b}'_j - 1)^{-1} \sum_{k \in S_-} \bar{a}_{jk} y_k \tag{3.26}$$

This inequality must be satisfied for x_j to be an integer and this is the Gomory cut or constraint to be introduced in the final tableau.

A slack variable x_r is to be added to (3.26) as follow

$$\bar{b}'_j \leq \bar{b}'_j(\bar{b}'_j - 1)^{-1} \sum_{k \in S_-} \bar{a}_{jk} y_k + x_r \tag{3.27}$$

As $y_k = 0,\ k = 1,\ 2,\ \ldots,\ p$, we have that

$$x_r = -\bar{b}_j$$

which is non feasible then it remains to apply the dual simplex algorithm to remedy this outcome.

The above process is repeated until either

(i) A tableau is obtained where $x_i = 0,\ i = 1,\ 2,\ \ldots,\ q$ are integers in which case the corresponding is optimal or

(ii) The use of the dual simplex technique leads to the conclusion that no feasible solution exists in which case we conclude that the original mixed integer programming problem has no feasible solution

3.5.1 DAKIN'S METHOD OF BRANCH AND BOUND ENUMERATION

Optimal solutions to any integer problem can be obtained by listing all possible solutions and choosing the best i.e. by exhaustive enumeration, it is also possible to examine the set of all possible solutions so that whole sets of solutions can be discarded without specific evaluation of the all the solution in each of the sets, this technique is known as implicit enumeration.

An implicit enumeration is called the branch and bound enumeration and is designed for integer programming [2].

Given a programming problem

Maximize $C^T x$
subject to

$$Ax = B$$

$$x \geq 0$$

$x_1, x_2, \ldots, x_q$

$$(3.28)$$

where $x = (x_1, x_2, \ldots, x_q, \ldots, x_n)^T$ and
$C = n \times 1$, $B = m \times 1$ and $A = m \times n$

Solution to the above problem can be solved using Dakin's method enumerating as follows:

1 Solve the problem as Linear Programming problem ignoring integer requirementsusing simplex method.

2 value of solutions obtained is the bound which is assigned to the first point of the decision tree representing all feasible solution to the original LP.

The two constraints (3.31) are called the Dakin's cut but x_i cannot take value $\bar{b}_i$ in either of the two cases but two new points are created in the decision tree both joined by lines to the original point.

The first represent all feasible solutions to problem 1 and the second to problem 2.

Optimal solution to the original LP if it exists lie in one of these sets S_1 and S_2, i.e. partitioning the feasible solution S to the original problem into two sets S_1 and $S-2$, so that

$$S_1 \cup S_2 = S; \quad S_1 \cap S_2 = \phi$$

3.5.2 EXAMPLE I

Maximize $x_1 + 2x_2$
subject to

$$2x_1 + 2x_2 \leq 7$$
$$2x_1 - x_2 \leq 5$$

x_i integers.

(3.33)

By simplex algorithm:

Table 3.2

constraints	x_1	x_2	x_3	x_4	b_j
x_3	2	2	1	0	7
x_4	2	-1	0	1	5
z	1	2	0	0	0

Table 3.3

constraints	x_1	x_2	x_3	x_4	b_j
x_1	1	1	1/2	0	7/2
x_4	0	3/2	1/2	-1/2	2
z	0	-1	-3/2	0	7/2

Table 3.4

constraints	x_1	x_2	x_3	x_4	b_j
x_1	1	0	1/6	1/3	13/6
x_2	0	1	1/3	-1/3	4/3
z	0	0	-7/6	-1/3	29/6

feasible solution:

$$x_1 = \frac{13}{6}, \quad x_2 = \frac{4}{3}, \quad z = \frac{29}{6}$$

but this is not feasible and not optimal since x_1 and x_2 are non–integers. A cut is introduced and x_1 can be written as

$$\frac{13}{6} - x_1 = \frac{1}{6}x_3 + \frac{1}{3}x_4$$

i.e.

$$2 + \frac{1}{6} - x_1 = \frac{1}{6}x_3 + \frac{1}{3}x_4 \tag{3.34}$$

therefore

$$[\bar{b}_j] = 2$$

$$\bar{b}'_j = \frac{1}{6}$$

$j = 2,\ i = 1,\ p = 2$

$$\bar{a}_{j1} = \frac{1}{6}$$

$$\bar{a}_{j2} = \frac{1}{3}$$

$y_1 = x_3$, $y_2 = x_4$, $S_+ = \{4\}$, $S_- = \{3\}$;
The cut now becomes

$$\begin{aligned}\frac{1}{6} &= \frac{1}{6}(\frac{1}{6}-1)^{-1}(\frac{1}{6})x_3 + \frac{1}{3}x_4 - x_5 \\ &= \frac{1}{6}(-\frac{5}{6})^{-1}(\frac{1}{6})x_3 + \frac{1}{3}x_4 - x_5 \\ &= \frac{1}{30}x_3 + \frac{1}{3}x_4 - x_5 \qquad (3.35)\end{aligned}$$

giving rise to the tableau:

Table 3.5

constraints	x_1	x_2	x_3	x_4	b_j
x_1	1	0	1/6	1/3	13/6
x_2	0	1	1/3	-1/3	4/3
x_3'	0	0	-1/30	-1/3	-1
z	0	0	-7/6	-1/3	29/6

Table 3.6

constraints	x_1	x_2	x_3	x_4	b_j
x_1	1	0	29/180	5/18	2
x_2	0	1	1/3	-1/3	4/3
x_4'	0	0	-1/30	-1/3	-1
z	0	0	-7/6	-1/3	29/6

giving

$$x_1^* = 2$$

$$x_2^* = \frac{4}{3}$$

$$z = x_1 + 2x_2 = 2 + 2(\frac{4}{3}) = \frac{14}{3} \approx \frac{29}{6}$$

3.5.3 EXAMPLE II

Maximize $f(x_1,\ x_2) = 4x_1 + 3x_2$
subject to

$$3x_1 + 4x_2 \leq 12$$
$$4x_1 - 2x_2 \leq 9$$

$x_1 > 0,\ x_2 > 0,\ x_i,\ i = 1,\ 2$ integers.

(3.36)

By simplex algorithm:

Table 3.7

	x_1	x_2	x_3	x_4	b_j
x_2	0	1	2/5	-3/10	21/10
x_1	1	0	-1/5	-1/5	6/5
f	0	0	2/5	7/10	111/10

Optimal solution

$$x_1^\star = \frac{6}{5} = 1 + \frac{1}{5}$$
$$x_2^\star = \frac{21}{10} = 2 + \frac{1}{10}$$
$$f^\star = \frac{111}{10} = 11 + \frac{1}{10}$$

(3.37)

Take the largest of the x_1 and x_2, i.e.

$$x_2^\star = 2 + \frac{1}{10}$$

the integer part is 2 and so

$$x_2 \leq 2,\ \ x_3 \geq 3$$

and the decision tree will be designed as follow:

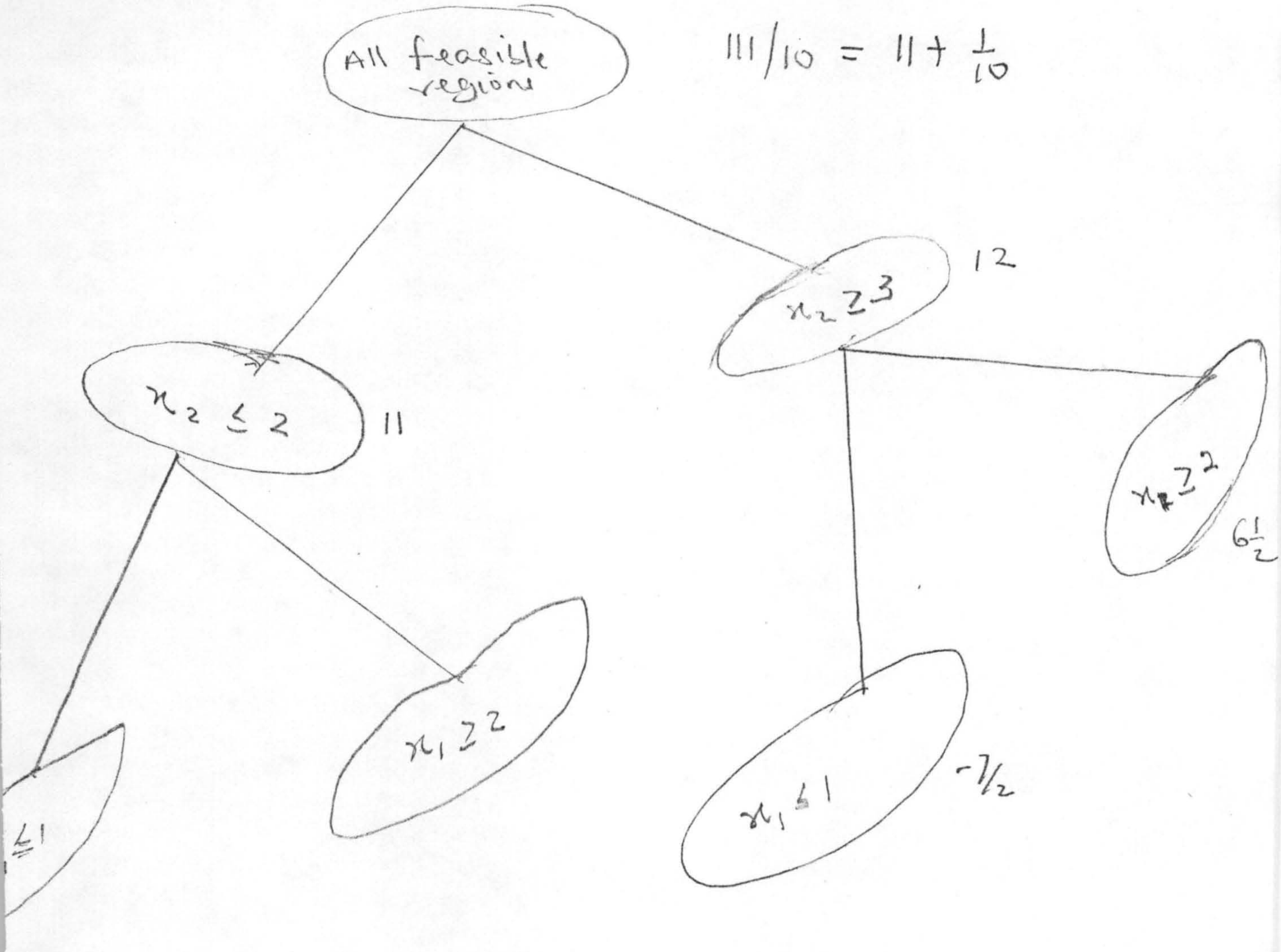


Figure 3.2

The two new problems emanating from these new cuts are:

I Maximize $4x_1 + 3x_2$
subject to

$$3x_1 + 4x_2 \leq 12$$
$$4x_1 - 2x_2 \leq 9$$

$$x_2 \leq 2,\ x_i \geq 0,\ i = 1,\ 2 \tag{3.38}$$

II Maximize $f(x_1,\ x_2) = 4x_1 + 3x_2$
subject to

$$3x_1 + 4x_2 \leq 12$$
$$4x_1 - 2x_2 \leq 9$$

$$x_2 \geq 3,\ x_i \geq 0,\ i = 1,\ 2 \tag{3.39}$$

3.6 REMARKS

Problem I has optimal solution

$$x_1^\star = \frac{5}{4}$$
$$x_2^\star = 2$$
$$f^\star = 11$$

Problem II has optimal solution

$$x_1^\star = \frac{3}{4}$$
$$x_2^\star = 3$$

$$f^{\star} = 12$$

From the above the two problems still have non–integer solution, hence the procedure is repeated.

Problem I – choose $x_1 \leq 1$ and $x_2 \geq 2$
we create new LP's as

III Maximize $4x_1 + 3x_2$
subject to

$$3x_1 + 4x_2 \leq 12$$
$$4x_1 - 2x_2 \leq 9$$

$x_2 \leq 2,\ x_1 \leq 1$

IV Maximize $4x_1 + 3x_2$
subject to

$$3x_1 + 4x_2 \leq 12$$
$$4x_1 - 2x_2 \leq 9$$

$x_2 \leq 2,\ x_1 \geq 2$

V Maximize $4x_1 + 3x_2$
subject to

$$3x_1 + 4x_2 \leq 12$$
$$4x_1 - 2x_2 \leq 9$$

$x_2 \leq 3,\ x_1 \leq 1$

VI Maximize $4x_1 + 3x_2$
subject to

$$3x_1 + 4x_2 \leq 12$$
$$4x_1 - 2x_2 \leq 9$$

$x_2 \leq 3,\ x_1 \geq 1$

Problem III – choose $x_1 = 1$ and $x_2 = 2$

$f(x_1, x_2) = 4(1) + 3(2) = 10$

Problem IV – choose $x_1 = 2$ and $x_2 = 2$

$f(x_1, x_2) = 4(2) + 3(2) = 14$

Problem V – choose $x_1 = 1$ and $x_2 = 3$

$f(x_1, x_2) = 4(1) + 3(3) = 13$

Problem VI – choose $x_1 = 1$ and $x_2 = 3$

$f(x_1, x_2) = 4(1) + 3(3) = 13$

The decision tree is given below:

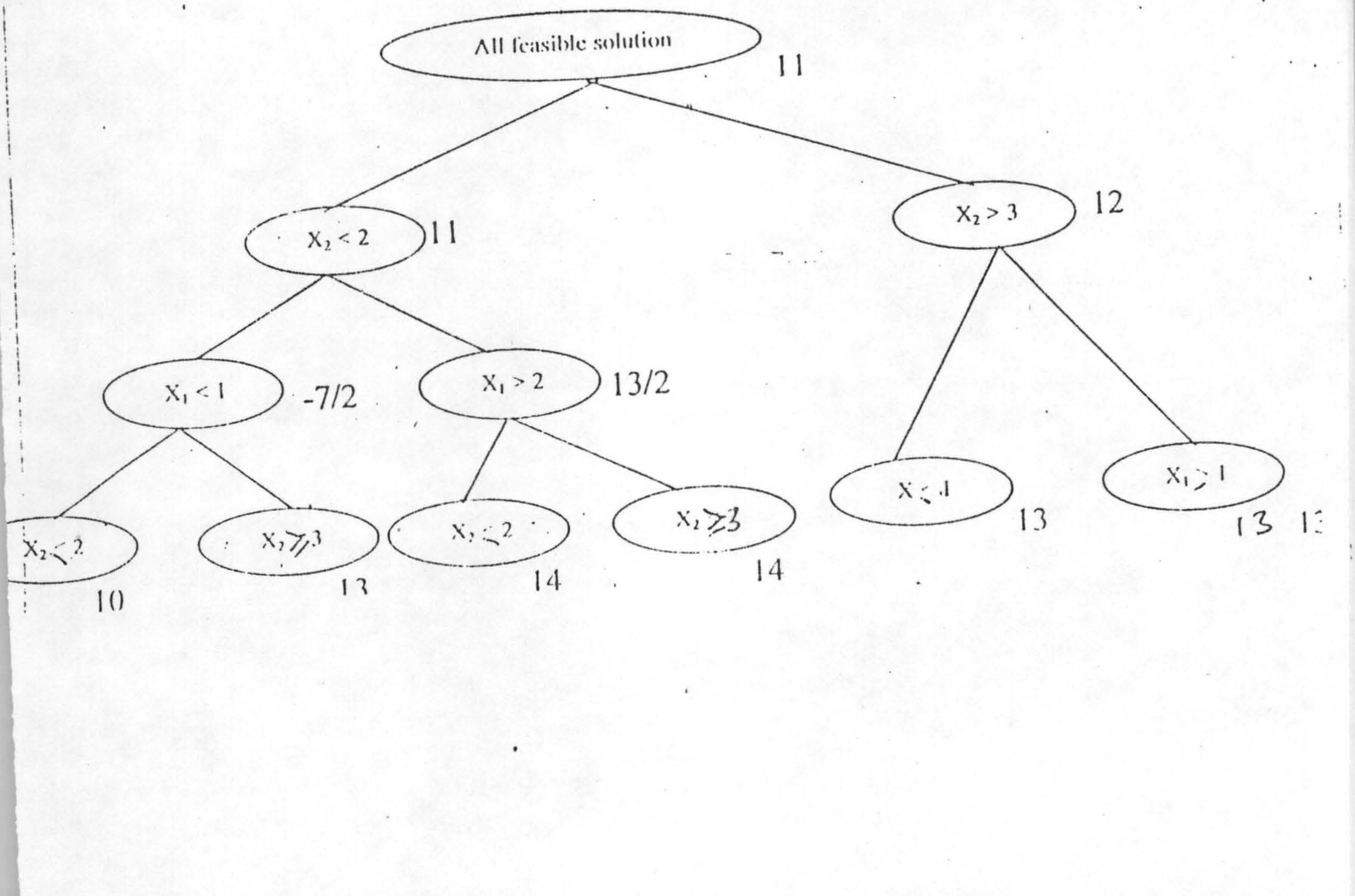


Figure 3.4

Optimal solutions:

Problem III

$$x_1^\star = 1$$
$$x_2^\star = 2$$
$$f^\star = 10$$

Problem IV

$$x_1^\star = 2$$
$$x_2^\star = 2$$
$$f^\star = 14$$

Problem V

$$x_1^\star = 1$$
$$x_2^\star = 3$$
$$f^\star = 13$$

Problem VI

$$x_1^\star = 1$$
$$x_2^\star = 3$$
$$f^\star = 13$$

CHAPTER FOUR

EXTENDED CUTTING PLAN ALGORITHM FOR NON–LINEAR PROGRAMMING PROBLEM

4.1.1 INTRODUCTION

The method adopted by Gomory for LPP can be extended to non–linear programming problem, this is the main focus of this research work.

4.1.2 LINEARIZATION OF THE NON–LINEAR OBJECTIVE FUNCTION

Given a problem

Minimize $F(x_1, x_2, \ldots, x_n)$
subject to

$$g_j(x_1, x_2, \ldots, x_n) \leq 0$$

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

$$\tag{4.1}$$

A new variable x_{n+1} is introduced as the original problem is transformed into an equivalent form [4], [12]

Find $(x_1, x_2, \ldots, x_n, x_{n+1})$
which minimizes x_{n+1}
subject to

$$g_j(x_1, x_2, \ldots, x_n) \leq 0$$

$j = 1, 2, \ldots, m$
and

$$g_{m+1}(x_1, x_2, \ldots, x_{n+1}) = f(x_1, x_2, \ldots, x_n) - x_{n+1} \leq 0$$

(4.2)

Generally, the original problem (4.1) may be stated as:

Minimize

$$f(x) = C^T x = \sum_{i=1}^{n} c_i x_i$$

subject to

$$g_j(x) \leq 0$$

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

(4.3)

where

$$x = (x_1, x_2, \ldots, x_n, x_n)$$

$C = m \times 1$ matrix

4.1.3 EXAMPLE

Minimize $f(x) = x_1^2 - 2x_1x_2$
subject to

$$x_1 + x_2 \leq 5$$
$$2x_1 - x_2 \leq 3$$

(4.4)

Transforming the problem by introducing an additional variable x_3 as illusrated above gives the simultaneous inequalities (i), (ii) and (iii) below:

$$x_1 + x_2 \leq 5 \qquad (i)$$
$$2x_1 - x_2 \leq 3 \qquad (ii)$$
$$x_1^2 - 2x_1x_2 - x_3 \leq 0 \qquad (iii)$$

Solving the inequalities give

$$x_1 = \frac{8}{3}, \; x_2 = \frac{7}{3}, \; x_3 = -\frac{48}{9}$$

Next we introduce slack variables into (i) and (ii) as follow:

$$x_1 + x_2 + x_3 = 5 \qquad (iv)$$

$$2x_1 - x_2 + x_4 = 3 \qquad (v)$$

$$x_1 + x_2 + x_5 = \frac{31}{3} \qquad (vi)$$

Table 4.1

	x_1	x_2	x_3	x_4	x_5	b_j
x_3	1	1	1	0	0	5
x_4	2	-1	0	1	0	3
x_5	1	1	0	0	1	-31/3
z	1	1	0	0	0	0
x_1	1	1	1	0	0	5
x_2	0	1/2	1	-1/2	0	7/2
x_5	0	0	1	0	-1	-16/3
z	0	0	1	0	0	5
x_1	1	0	-1	1	0	-2
x_2	0	1	2	-1	0	7
x_5	0	0	1	0	-1	-16/3
z	0	0	1	0	0	5

$$x_1 = -2, \;\; x_2 = 7, \;\; f(x) = x_1 + x_2 = 5$$

4.1.4 LINEARIZATION OF CONSTRAINT FUNCTION OF A NON–LINEAR PROGRAMMING PROBLEM

The following steps can be used in linearizing the constraint functions of a non–linear programming problem, [1], [18]:

Step 1 Start with initial point x_1 and set the iteration number as i = 1, this point need not be feasible.

Step 2 Linearize the constraint function $g_j(x)$ as

$$g_j(x) = g_j(x_1) + \nabla g_j(x_1)^T(x - x_1), \quad j = 1, 2, \ldots, n$$

Step 3 Formulate the approximating LPP constraint as

$$g_j(x_1) + \nabla g_j(x_1)^T(x - x_1) \leq 0, \quad j = 1, 2, \ldots, n$$

Step 4 Solve the approximating LPP to obtain the solution vector x_i.

Step 5 Evaluate the original constraints at x_{i+1}. If

$$g_j(x_{i+1}) \leq e$$

where e is a prescribed small positive tolerance.

4.1.5 EXAMPLE

Minimize $f(x) = x_1 + x_2$
subject to

$$g(x_1,\ x_2) = x_1^2 - 4x_1 + x_2^2 - 3 < 0 \tag{i}$$

Solution

Step 1 – Start with an initial solution x_1

Step 2 – To avoid unbounded solution let x_1 and x_2 be bounded as

$$-1 < x_1 < 1, \quad -1 < x_2 < 1 \tag{ii}$$

Step 3 – The problem becomes
Minimize $f(x) = x_1 + x_2$
subject to

$$-1 < x_1 < 1, \ \ -1 < x_2 < 1 \tag{iii}$$

Step 4 – Solving this LPP at $(-1, \ 1)$ gives f(x) = 0

$$g(x_1, \ x_2) = (-1)^2 - 4(-1) + (1)^2 - 3 = 1 + 4 + 1 - 3 = 3 > 0$$

let the choice of e = 0.02
since

$$g(x_1, \ x_2) = 3 > e$$

then

Step 5 – we linearize about the point x_2 as

$$g_1(x) = g_1(x_2) + \nabla g_1(x_2)^T (x_1 - x_2) < 0$$

$$\frac{\partial g}{\partial x_1} = 2x_1 - 4x_2 = -6$$

$$\frac{\partial g}{\partial x_2} = -4x_1 + 2x_2 = -2$$

$$g_1(x_1, \ x_2) = -6x_1 - 2x_2 + 7$$

adding this constraint to the first two:

$$-1 < x_1 < 1, \ \ -1 < x_2 < 1, \ \ -6x_1 - 2x_2 + 7 < 0$$

Step 6 – Set the iteration number i = 2. (step 4 recalls); solve the LPP
at $x_2 = 1$.
$-6x_1 - 2x_2 + 7 = -6x_1 - 2 + 7 = 0$
$-6x_1 = -5$
$x_1 = 5/6 = 0.8333$
which gives
$f_2(x) = x_1 + x_2 = 11/6$

then step 5

$$g_1(x_1) = 25/36 - 20/6 = 107/36 < e$$

the iteration stops since $g_1(x_1) < e$.
Result in tabular form:

Table 4.2

New linearised constraints	solution of the approximating LPP	F(x_1+1)	$g_1(x_1+1)$
$-1 < x_1 < 1$	-1.000, 1.000	0	3
$-1 < x_2 < 1$	-1.000, 1.000	0	3
$-6x_1 - 2x_2 + 7 < 0$	0.8333, 1.000	1.8333	-3¡e=0.02

4.2 GEOMETRIC INTERPRETATION OF THE CUTTING PLANE METHOD [12]

Given the one variable problem

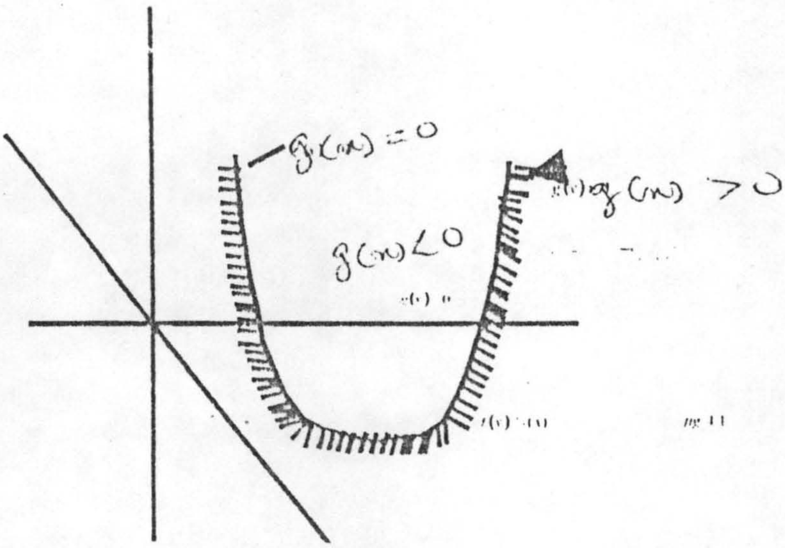
Maximize $f(x) = c_1 x$
subject to

$$g(x) \leq 0 \tag{4.5}$$

c_1 a constant and $g(x)$ a non–linear function of x.

This problem is represented by the following graph:

figure 4.1



The feasible region and the contour of the objective function are as shown in the graph.

In order to avoid unboundedness of the solution we can introduce additional constraints

$$l_i \leq x \leq u_i$$

where l_i and u_i are lower and upper bounds repectively.

The programming problem then takes the form:

Maximize $f(x) = c_1 x$

subject to

$$l_i \leq x \leq u_i$$

$$(4.6)$$

The optimum solution of the approximating linear programming problem can be taken as

$$x^* = l_1$$

Next is to linearize the constraint $g(x)$ about the point l_1 and add it to the previous constraint, and the problem now takes the form:

Maximize $f(x) = c_1 x$

subject to

$$l_1 \leq x \leq u_1$$

and

$$g(l_1) + \frac{dg(l_1)}{dx}(x - l_1) \leq 0$$

$$(4.7)$$

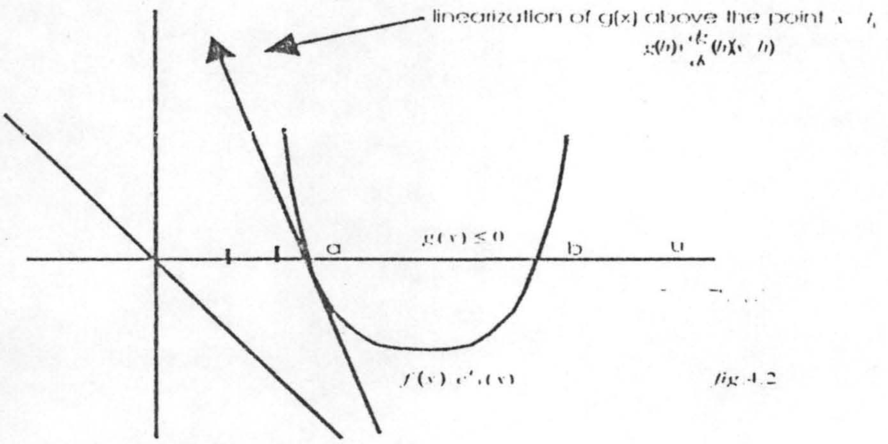
The feasible region of x as a result of the constraints is given in the graph below by

$$l_2 \leq x \leq u$$

The optimum solution of the approximating linear programming problem is

$$x^* = l_1$$

figure 4.2



Linearization of constraint above the point

97

4.3 SIMULTANEOUS LINEARIZATION OF THE OBJECTIVE AND CONSTRAINT FUNCTIONS

4.3.1 INTRODUCTION

In this method, the objective and constraint functions are expanded about a point using the Taylor series expansion in the following form:

Given a non–linear programming problem:

Maximize $z = f(x)$

subject to

$$g(x) \leq b_i \tag{4.8}$$

This non–linear programming problem can be approximated to a linear programming problem which is solved to obtain a trial point x_2.

Repeating the procedure using x_2 in the place of x_1 until the problem is reduced to the solution of a sequence of linear programming problem.

4.3.2 THE LINEARIZATIONS

Let x_1 be a feasible solution, then obtain dual form from (4.8)

$$F(x) = f(x_1) + [\nabla f(x_1)(x - x_1)]$$

and

$$G(x) = g_1(x_1) + [\nabla g(x_1)(x - x_1)] \tag{4.9}$$

a new variable y_j can be introduced with a non restrictive sign as follows

$$y = [y_1 \dots y_n] = x - x_1$$

we introduce the following notations:

$$a_i = \nabla g_1(x_i)$$

$$\hat{b}_i = b_i - g_i(x_i)$$
$$c = \nabla f(x_i)$$
$$\hat{z} = z - f(x_i)$$

$$(4.10)$$

substituting in (4.9) gives:

Maximize $\hat{z} = c'y$

subject to

$$a_i'y \leq b_i \tag{4.11}$$

y is unrestricted in sign.

To ensure the validity of the linear approximations, we impose upper bounds on the magnitudes of the variables y_j as

$$\{y_j\} \leq m_j \tag{4.12}$$

Let $y_1^\star$ be the optimal solution of the problem subject to the additional constraint

$$x_2 = x_1 + y_1^\star \tag{4.13}$$

is then taken as the next trial point and the constraint a_i, b_i and c are evaluated at this point x_2 and a new linear programming problem similar to (4.11) with conditions in (4.12) is formulated but no guarantee that the new trial point will satisfy the constraints of the problem. To absorb this, either decrease the upper bound m_j or proceed to the next stage ignoring x_2 as not being feasible.

Iteration will terminate when the difference between two successive solution is acceptably small, i.e.

$$\{x_{i+1} - x_i\} < \epsilon \tag{4.14}$$

prescribed or when the difference between two successive values of objective function is small, i.e.

$$\{z_{i+1} - z_i\} < \delta \tag{4.15}$$

is prescribed.

$$x'_j \le x_j \le x''_j \tag{4.16}$$

since y is unrestricted in sign but bounded in magnitude, we replace variables y_j by the non-negative variables

$$w_j = y_j + m_j \tag{4.17}$$

and then

$$\begin{aligned} x''_j &\le x_{1j} + w_j \\ -m_j &\le x''_j \end{aligned} \tag{4.18}$$

where x_{1j} is the j^{th} component of x_1 and

$$-m_j \le w_j - m_j \le m_j \tag{4.19}$$

then

$$\begin{aligned} &\max\{x'_j - x_{1j} + m_j,\ 0\} \\ &\le w_j \le \min\{x''_j - x_{1j} + m_j,\ 2m_j\} \end{aligned} \tag{4.20}$$

this leads to the linear programming problem

Maximize $\hat{z} = c'(w - m)$

subject to

$$\begin{aligned} a_1 w &\le b_i + a'_i m \\ w'_j &\le w_j \le w''_j \end{aligned} \tag{4.21}$$

where $w = \{w_1, \ldots, w_n\}$
$m = \{m_1, \ldots, m_n\}$
$w'_j = \max\{x'_j - x_{1j} + m_j, 0\}$
$w''_j = \min\{x''_j - x_{1j} + m_j, 2m_j\}$

4.3.3 EXAMPLE I

Maximize $z = x_1^2 - x_1x_2 + 2x_2^2$
subject to

$$3x_1 + 4x_2 \leq 10$$
$$x_1^2 - x_2^2 \geq 1$$

$x_1, \ x_2 \geq 0$

(4.22)

Solution

Take $x_1 = [2, \ 1]$ to be the initial feasible solution, then

$$a_1 = \nabla g_1(x_1), \ \ \hat{b}_i = b_i - g_i(x) = [3, \ 2]$$
$$\hat{b}_1 = 10 - [3x_1 + 4x_2] = 10 - [6 + 4] = 0$$
$$a_2 = [2x_{11}, \ -2x_{22}] = [4, \ -2]$$
$$\hat{b}_2 = 1 - \left[2^2 - 1^2\right] = -2$$
$$c = \nabla f(x_1) = [2x_{11} - x_{12}, \ -x_{11} + 4x_{12}] = [3, \ 2]$$

the first constraint and non–negativity restriction for the above problem imply that
$0 \leq x_1 \leq 4$ and $0 \leq x_2 \leq 3$
therefore take
$x_1' = 0, \ x_1'' = 4, \ x_2' = 0, \ x_2'' = 3$
to be the upper and lower bounds as defined earlier.
$m_1 = m_2 = 1/2, \ m = [1/2, \ 1/2]$, the LPP then takes the form:
Maximize $\hat{z} = 3w_1 + 2w_2$
subject to

$$3w_1 + 4w_2 \leq b_1 + a_{11}m_1 + a_{12}m_2 = 7/2$$
$$4w_1 - 2w_2 \geq b_2 + a_{21}m_1 + a_{22}m_2 = -1$$

i.e.
Maximize $\hat{z} = 3w_1 + 2w_2$
subject to

$$3w_1 + 4w_2 \leq 7/2$$
$$4w_1 - 2w_2 \geq -1$$

$0 \leq w_1 \leq 1,\ 0 \leq w_2 \leq 1$

(4.23)

Optimal solution is

$$w_1 = \frac{3}{22},\ w_2 = \frac{17}{22}$$

hence

$$x_2 = x_1 + w_1^\star - m = [2,\ 1] + \left[\frac{3}{22},\ \frac{17}{22}\right] - \left[\frac{1}{2},\ \frac{1}{2}\right] = \left[\frac{18}{11},\ \frac{14}{11}\right]$$

The second iteration begins by replacing x_1 by x_2.
$a_1 = [3,\ 4]$

$$a_2 = [2x_{11},\ -2x_{12}] = \left[2(\frac{18}{11}),\ -2(\frac{14}{11})\right] = \left[\frac{36}{11},\ -\frac{28}{11}\right]$$
$$\hat{b}_1 = \frac{7}{2} - [3w_1 + 4w_2]\,\frac{7}{2} - \left[3(\frac{3}{22}) + 4(\frac{17}{22})\right] = 0$$
$$\hat{b}_2 = 1 - [4w_1 - 2w_2] = 1 - \left[4(\frac{3}{22}) - 2(\frac{17}{22})\right] = 2$$

$$c = \nabla f(w_i) = 3w_1 + 2w_2 = 3\,[2x_{11} - x_{12}]\,,\ 2\,[-x_{11} + 4x_{12}] = \left[-\frac{3}{2},\ \frac{65}{11}\right]$$

and so we have that:

$$z = -cw = -\frac{3}{2}w_1 + \frac{65}{11}w_2$$

subject to

$$3w_1 + 4w_2 \leq 0$$

$$\frac{36}{11}w_1 - \frac{28}{11}w_2 \geq 2$$

$0 \leq w_1 \leq 1,\ 0 \leq w_2 \leq 1$

$$w_1^* = \frac{129}{26},\ w_2^* = -\frac{19}{13}$$

this completes the second iteration.

4.3.4 EXAMPLE II

Maximize $z = 2x_1^2 - 2x_1x_2 + x_2^2 - 1$
subject to

$$2x_1 - x_2 \leq 10$$

$$x_1^2 - 2x_1x_2 + x_2^2 \leq 6$$

$x_1,\ x_2 \geq 0$

(4.24)

Solution

Take $x_1 = [3,\ 2]$ to be the initial feasible solution, then $a_1 = \nabla g_1(x_1)$

$$a_{11} = \frac{\partial g}{\partial x_{11}} = 2$$

$$a_{12} = \frac{\partial g}{\partial x_{12}} = -1$$

$$a_{21} = \frac{\partial g}{\partial x_{21}} = 2$$

$$a_{22} = \frac{\partial g}{\partial x_{22}} = -2$$

$a_1 = [2,\ 2]$, $a_2 = [-1,\ -2]$

$$\hat{b}_i = b_i - g_i(x)$$

$$\hat{b}_1 = 10 - [2x_1 - x_2] = 4$$

$$\hat{b}_2 = 6 - \left[x_1^2 - 2x_1x_2 + x_2^2\right] = 8$$

$$c = \nabla f(x_1) = [8,\ 2]$$

$$\hat{z} = z - f(x_i) = z - (2x_1^2 - 2x_1x_2 + x_2^2 - 1)$$

$0 \leq x_1 \leq 2$, $0 \leq x_2 \leq 2$
we take
$x_1' = 0$, $x_1'' = 2$, $x_2' = 0$, $x_2'' = 2$; $c = [8,\ 2]$
as lower and upper bounds to satisfy the non negativity conditions. We take also $m_1 = m_2 = 1/2$, and the problem takes the form:

Minimize $\hat{z} = c(w - m)$
$z = 8w_1 + 2w_2$
subject to

$$2w_1 + 2w_2 \leq \hat{b}_1$$

$$-w_1 - 2w_2 \leq \hat{b}_2$$

$$\hat{b}_1 = b_1 + a_{11}m_1 + a_{12}m_2 = 9/2$$

$$\hat{b}_2 = b_2 + a_{21}m_1 + a_{22}m_2 = 8$$

i.e.
Minimize $\hat{z} = c(w - m) = 8w_1 + 2w_2$
subject to

$$2w_1 + 2w_2 \leq 9/2$$

$$-w_1 - 2w_2 \leq 8$$

$0 \leq w_1 \leq 1$, $0 \leq w_2 \leq 1$
Optimal solution is feasible at

$$w_1^\star = \frac{25}{2},\ w_2^\star = -\frac{41}{4}$$

Therefore:

$$x_2 = x_1 + w_i^\star - m$$
$$= [3,\ 2] + \left[\frac{25}{2},\ -\frac{41}{4}\right] + \left[\frac{1}{2},\ \frac{1}{2}\right] = \left[\frac{23}{3},\ -\frac{31}{4}\right]$$

This completes the first iteration.

For the second iteration, we replace x_1 by x_2;
$a_1 = [3,\ 2]$

$$a_2 = [2x_{11} - 2x_{12},\ -2x_{11} + 2x_{12}]$$
$$= \left[2(\frac{23}{3}) - 2(\frac{31}{4}),\ -\frac{23}{3} + 2(\frac{-31}{4})\right] = \left[\frac{185}{6},\ \frac{185}{6}\right]$$
$$\hat{b}_i = b_i - c'c$$
$$\hat{b}_1 = 9/2 - (2w_1 - 2w_2) = 9/2 - \left[2(\frac{25}{2}) + 2(\frac{-41}{4})\right] = -1$$
$$\hat{b}_2 = 8 - (-w_1 - 2w_2) = 8 - \left[-\frac{25}{2} - 2(\frac{-41}{4})\right] = 0$$
$$c = \nabla f(x_i) = [4x_{11} - 2x_{12},\ -2x_{11} + 2x_{12}]$$
$$= \left[4(\frac{23}{3}) - 2(\frac{-31}{4}),\ 2(\frac{23}{3}) + 2(\frac{-31}{4})\right] = \left[\frac{277}{6},\ \frac{-185}{6}\right]$$

therefore the problem becomes: Minimize $z = \frac{277}{6}w_1 - \frac{185}{6}w_2$ subject to

$$\frac{185}{6}w_1 - \frac{185}{6}w_2 \leq 0$$
$$w_1^\star = -\frac{1}{25},\ w_2^\star = -\frac{1}{25}$$
$$x_3 = x_2 + w_2^\star - m$$
$$\left[\frac{23}{3},\ -\frac{31}{4}\right] + \left[-\frac{1}{25},\ -\frac{1}{25}\right] - \left[\frac{1}{2},\ \frac{1}{2}\right] = \left[\frac{23}{3},\ -\frac{31}{4}\right]$$

CHAPTER FIVE

CONCLUSION AND RECOMMENDATION

5.1 CONCLUSION

This work shows that the Gradient method such as the Extended Cutting Plane Algorithm is very useful in solving Integer Programming Problem via linearization.

The method appears direct and easily applicable though with great care for accuracy of results.

The method was used to linearize both the objective and constraints functions giving more rapid convergence to the optimum solution than the Hookes and Jeev's or the Bound and Branch methods.

The application of convexity theory, imposition of necessary and sufficient conditions as illustrated in the work provides a global or optimal feasible solution. The penalty function method using Lagrangian and Kutn Tucker provided opportunity to appreciate the importance of the Extended Cutting Plane Method when compared.

5.2 RECOMMENDATION

The Extended Cutting Plane Method using Taylor series expansion could also be applied to any financial based problem. The Price – Yield, Risk – Return relationship in any stock issue lead to a Quadratic Programming Problem which can then be modified to know at what price to trade and the yield expected on number of issues traded. This highlighted problem naturally leads to All Integer, Mixed Integer or Zero–One Polynomial Non–Linear Programming Problems which is the main subject of this study.

Further research work can be carried out directly applying this study to Stocks related and other financial problems.

REFERENCES

1. Cheney E.W & Goldstein A.A – Newton's Method of Convex Programming and Tchebycheff Approximation; Numeriche Mathematics, Vol 1, (1959).

2. Eke A.N. – Contributions to Non–Linear Differential Equations; PhD Thesis, University of Nigeria, Nsukka, (1990).

3. Flacco A.V. & McCormarck G.P. – Non–Linear Programming Sequential Unconstrained Minimization Techniques, Wiley & Sons, (1968).

4. Foulds L.R. – Optimization Technique, An Introduction, Springer–Verlag, (1981).

5. Gomory R.E. – An All–Integer Programming, New Jersey, (1963) (Edited by Mullhad J.F. & Thompson G.I.)

6. Hadley G. – Linear Programming, Addison Wesley Publishing co. 9^{th} Edition, (1975).

7. Kalman R.E. – Contribution to the Theory of Parasitic Distributed Parameter by Discrete–Time Imput–Output Data, SIAM Control Optimiz 22; 509 – 522, (1989).

8. Kelly J.E. – The Cutting Plane Method for Solving Convex Programs, SIAM, Vol No. 4, (1960).

9. Kuester J.L & Joe H. – Optimization Techniques with FORTRAN, McGraw–Hill Book Co. (1973).

10. Owuatu J.U. – Methods and Application of Optimal Control Theory; Lecture Notes, National Mathematical Centre, Abuja, (2001).

11. Reju S.A. – Computational Methods in Optimization Theory; Lecture Notes, National Mathematical Centre, Abuja, (2001).

12. Rao S.S. – Optimization and Applications; Wiley & Sons, pp 557 – 560, (1984).