LINEAR PROGRAMMING (THEORY AND COMPUTATIONS)

A Thesis

Presented to the

Department of Mathematics

In Partial Fulfillment of the Requirements for the Degree of

MASTER OF SCIENCE INDUSTRIAL MATHEMATICS

By

BISMARK AMEYAW (BSC. MATHEMATICS AND STATISTICS HONS.)

INSTITUTE OF DISTANCE LEARNING, KNUST

[June 2012]

DECLARATION

I hereby declare that this submission is my own work towards the MSc. and that, to the best of my knowledge, it contains no material previously published by another person nor materials which have been accepted for the award of any other degree of the university, except where due acknowledgement has been made in the text.

BISMARK AMEYAW (PG4062410) 05-06-12

Student Name & ID **Signature** **Date**

Certified by:

KWAKU DARKWAH 5/6/2012

Supervisor's Name **Signature** **Date**

Certified by:

KWAKU DARKWAH .... 5/6/2012.

Head of Dept. Name **Signature** **Date**

ABSTRACT

For quite a long time students have been using linear programming models without understanding the details of the theory. Most students even find it difficult to explain solutions of linear programming question they have solved on their own.

This thesis reviewed the theory and solution methods of linear programming. These methods include the graphical method, the simplex method, the revised simplex method, the dual simplex method, Karmarkar's algorithm, the decomposition principle, the bounded variable technique and the column generation method. All the algorithms involved in the application of the methods mentioned above were systematically given. All the necessary steps to be followed in the formulation of linear programming models were also explained.

Users of this thesis will, therefore, be provided with easy to understand and logical review of essentials of linear programming theory with illustrations.

It was found out that errors are avoided when linear programming models are formulated properly and also when the details of the theory of linear programming are applied.

TABLE OF CONTENTS

LIST OF TABLES

CAPTION **PAGE**

LIST OF FIGURES

CAPTION **PAGE**

DEDICATION

To Bismark Ameyaw Junior, Josephine Dwomo Ameyaw and Susana Ofumaah Ameyaw.

ACKNOWLEDGEMENT

I would like to thank my supervisor, Mr. Kwaku Darkwah, a lecturer and head of Department of Mathematics of Kwame Nkrumah University of Science and Technology for his great supervision and continuous support. His invaluable pieces of advice were instrumental during my studies. His ideas and motivation were great source of inspiration to me. He shared his time, knowledge, and experience with me and also paid attention to all the steps of my thesis. May God richly bless him.

I am also grateful to my family and all my friends for their encouragement and support. They were always there for me whenever I needed them throughout my studies.

Finally, I would like to thank all my lecturers in Kwame Nkrumah University of Science and Technology, Kumasi for sharing their knowledge and ideas with me. They have made me who I am today.

CHAPTER 1

INTRODUCTION

1.1BACKGROUND OF THE STUDY

1.1.1 THE ORIGIN OF LINEAR PROGRAMMING

In spite of its wide applicability to everyday problems, linear programming was not known before 1947.Fourrier may have been aware of its potential in 1823. In 1939, Kantorovich of U.S.S.R, made proposals that were neglected during the two decades that witnessed the discovery of linear programming and its firm establishment elsewhere.

After World War II the US Air Force consolidated the statistical control, programming and budgeting functions under the staff of the Air force Comptroller, General E W Rawlings who was the president of Mills foundation. It became clear to members of this organization that efficiently coordinated the energies of whole nations in the event of total war would require scientific programming techniques. Undoubtedly this need had occurred many times in the past, but this time there were two concurrent developments that had a profound influence: (a) the development of large scale electronic computers and (b) the development inter-industry model. The latter is a method of describing inter-industry relation of an economy and was originated by Leontief (1951). According to Dantzig(1963), intensive work began in June 1947, in a group that later (October 1948) was given the official title of project SCOOP (Scientific Computation of Optimum Programs). Principals in this group were Marshall Wood and author and soon thereafter John Norton and Murray Geisler.

The simplex computational method for choosing the optimal feasible program was developed by the end of summer of 1947. Interest in linear programming began to spread rapidly. During this period the Air Force sponsored work at the U.S Bureau of standards on electronic computers and on mathematical techniques for solving such models. John Curtiss and Albert Cahn of the Bureau played an active role in generating interest in the work among economics and mathematicians.

Contact with Tjalling Koopmans of the Cowles Commission, then at the University of Chicago and Robert Dorfman, then of the Air Force and the interest of such economics as Paul Samuelson of the Massachusetts Institute of Technology, initiated an era of intense re-examination of classical economic theory using results and ideas of linear programming (Dantzig, 1963).

Contact with john von Neumann at the Institute of Advanced Study gave fundamental insight into the mathematical theory and sparked the interest of A.W Tucker of Princeton University and a group of his student who attacked problems in linear inequality theory and game theory. Since that time his group has been a focal point of work in these related fields.

Since 1948, theAir Force Staff has been making more and more active use of mechanically computed programs. The triangular models are in constant use for the computation of detailed programs, while the general linear programming models have been applied in certain areas such as (a) contract binding, (b) balanced air craft, crew training and wing employment procedures, (c) scheduling of maintenance overhaul cycles, and (d) personnel assignment and (e) airlift routing problems (U.S.A Air Force, 1954; Jacobs and Natrella, 1951).

Mathematical Origins and Developments

The linear programming model, when translated into purely mathematical terms requires a method for finding a solution of simultaneously linear equations and linear inequalities which minimizes a linear form. This central mathematical problem of linear programmingwas not known to be an important one with many practical applications until the advent of linear programming in 1947.

The literature of mathematics contains thousands of papers concerned with techniques for solving linear equation systems, with the theory of matrix algebra with linear approximation methods, etc. on the other hand, the study of linear inequality systems excited virtually no interest until the advent game theory in 1944 and linear programming in1947. For example T Motzkin, in his doctoral thesis on linear inequalities in1936,was able to cite after diligent search only some thirty references for the period 1900-1936, and about forty two in all (Motskin, 1936). In the 1930's, four papers dealt with the building of a comprehensive theory linear inequalities and with an appraisal of earlier works. These were by R. w. Stokes (1931), Dines-McCOY (1931-1), H .Weyl (1935), and T. Motzkin (1933). As evidence that mathematicians were unaware of the importance of the problem of seeking a solution to an inequality system that also minimized a linear form, we maynote that none of these papers made any mention of such a problem, although there had been earlier instances in the literature.

The famous mathematician, Fourrier (1826), while not going into the subject deeply, appears to have been the first to study linear inequalities systematicallyand to point out their importance to mechanics and probability theory. He was interested in finding the least maximum deviation fit to a system of linear equations, which he reduce to the problem of finding the lowest point of a polyhedral set. He suggested a solution by a vertex-to vertex descent to a minimum, which is the principle behind the simplex method used today. This is probably the earliest knwon instance of a linear programming problem. Lather another famous mathematician, de la ValleePoussin (1911), considered the sameproblem and proposed a similar solution.

A good part of the early mathematical literature is concerned with finding conditions under which a general homogeneous linear inequality system can be solved. All the results obtained express, in one form or another, a relationship between the original (or primal) system and another system (called the dual) which uses the column s of the original matrix of coefficients to form a new linear equations or inequalities according certain rules. Typical is the derived theorem of P Gordan (1873) showing that a solution with at least one variable positive if the dual possesses no solution with strict inequalities. Siemke (1915) added a theorem on the existence of a solution with all variables positive. These results are expressed in a sharper form in Motskin's Transportation Theorem (1936) and theorems on dual Systems by Tucker (1956). Specifically designed for algebraic proof of the Minimaxtheorem are the results of Ville (1938) and of (von Neumann and Mergenstern, 1944). Essentially, these theorems stated that either the original (primal) possesses a nontrivial solution or the dual system possesses a strict inequality solution. Because of this ''either-or,'' von Neumann and Morgenstern called their result the Theorem of the Alternative for Matrices.

The work of Kantorovich

The Russian mathematician L. V Kantorovich had for number of years been interested in the application of mathematics to programming problems. He published an extensive monograph in 1939 entitledMathematical Methods in the Organization and Planning of Production (1939).

In his introduction Kantorovich states, ''There are two ways of increasing efficiency of the work of a shop, an enterprise, or a whole branch of industry. One way is by various improvements in technology, that is, new attachments for individual machines, changes in technological processes, and the discovery of new, better kinds of raw materials. The other way, thus for much less used, is by improvement in the organization of planning and production. Here are included such questions as the distribution of work among individual machines of the enterprise, or among mechanisms, orders among enterprises, the correct distribution of different kinds of raw materials, fuels and other factors'' (Kantorovich, 1939).

Kantorovich should be credited with being the first to recognize that certain important broad classes of production problems had well defined mathematical structures which, he believed, were amendable to practical numerical evaluation and could be numerically solved.

In the first part of his work Kantorovich is concerned with what he now calls the weighted two index distribution problems. These were generalized first to include a single linear side condition, then a class of problems with processes having several simultaneous outputs (mathematically the latter is equivalent to general linear program). He outlined a solution approach based on having on hand an initial feasible solution to the dual. (For the particular problems studied, the latter did not present any difficulty). Although the dual variables were not called ''prices,'' the general idea is that the assigned values of these resolving multipliers for resources in short supply can be increased to a point where it pays to shift to resources that are in surplus. Kantorovich showed on simple examples how to make the shifts to surplus resources. In general however how to shift turns out to be a linear program in itself forwhich no computational method was given. The report contains an outstanding collection of potential applications. His 1942 paper ''On the Translation of masses'' (Kantorovich, 1942) is the forerunner of his joint paper with M.K Gavurin on '' The Application of Mathematical Methods to problems of Freight Flow Analysis'' (Kantorovich and Gavurin, 1948). Here can be found a very complete theory of the transshipment problem, the relations the primal and the dual price system, the use of the linear graph of the network, and the important extension to capacitated networks. Moreover, it is clear that the authors had developed considerable facility with the adjustment of freight flow patterns from nonoptimal to optimal patterns for elaborate systems of the kind commonly encountered in practice. However, again, incomplete computational algorithm was given. It is commendable that the paper is written in a nontechnical manner, so as to encourage those responsible for routing freight to use the proposed procedures.

In 1959, twenty years after the publication of his first work, Kantorovich published a second entitled Economic Computation of the Optimal Utilization of resources, a book primarily intended for economists (1959). If Kantorovich's earlier efforts had been appreciated at the time they were first presented, it possible that linear programming would be more advanced today. However, his early work in this field remained unknown both in the Soviet Union and elsewhere for nearly two decades while linear programmingbecame a highly developed art. According to The New Times, ''The scholar, professor L .V Kantorovich, said in a debate that, Soviet economists had been inspired by a fear of mathematics that left the Soviet Union far behind the United States in applications of mathematics to economic problems. It could have been a decade ahead'' (New York Times, 1959).

The National Bureau of standards played an important role in the development of linear programming theory. Not only did it arrangethrough John H Curtiss and Albert Kahn the important initial contact between workers in the field, but it provided for the testing of a number of computational proposals in their laboratories. In the fall of 1947, Laderman of the Mathematical Tables Project in New York computed the optimal solution of Stigler's diet problem Stigler (1945)in a test of the newly proposed simplex method. At the institute of Numerical Analysis, Professor Theodore Motzkin, whose work on the theory of linear inequalities has been mentioned earlier, proposed several computational schemes for solving linear programming problems such as the Relaxation '' Method'' Motzkin and Scheonberg (1954) and the Double Description Method (Motzkin et al., 1953). Alex Orden of the Air Force worked actively with National Bureau of Standards (N.B.B.S) group who prepared codes on the SEAC(National Bureau of Standards Eastern Automatic Computer) for the general simplex method and for the transportation problem. Alan J. Hoffiman, with a group at the N B S , was instrumental in having experiments run on a number of alternative computational methods (Hoffiman et al., 1953). He was also the first to establish the cycling can occur in simplex algorithm without special provisions for avoiding degeneracy (Hoffiman, 1953).

In June 1951, the first symposium in linear programming was held in Washington under the joint auspices of the Air Force and the N B S. By this time, interest in linear programming was widespread in government academic circles. A Charness and W Cooper had just begun their pioneering work on industrial applications. Aside from this work, they published numerous contributions to the theory of linear programming. Their lectures were published in An Introduction to Linear Programming (Charness et al., 1953).

Electronic Computers Codes

The special simplex method developed for the transportation problem Dantzig (1951) was first coded for the SEAC in 1950 and the general simplex method in 1951 under the general direction of A. Orden of the Air Force and A. J. Hoffiman of the Bureau of Standards. In 1952, W. Orchard-Hays of The RAND Corporation worked out a simplex code for the IBM-C.P.C, and for the 701 and 704 in 1954 and 1956, respectively. The later code was remarkably flexible and solved problems of two hundred equations and a thousand or more variables in five hours or so with great accuracy (Ochard-Hays, 1955).

Extension of Linear programming

If we distinguish between those types of generalizations on mathematics that have led to existence proofs and those that have led to constructive solutions of practical problems, then the period following the first decade marks the beginning of several important constructive generalizations of linear programming concepts to allied fields. These are: Network Theory:A remarkable property of a special class of linear programs, the transportation the equivalent network flow problem, is that their extreme point solutions are integer valued when their constant terms are integers (Birkhoff,1946; Dantzig, 1951). This has been a key fact in an elegant theory linking certain combinatorial problems of topology with the continuous processes of network theory. The field has many contributors. Of special mention is the work of Kuhn (1955) using an approach of Egervary on the problem of finding a permutation of ones in a matrix composed of zeros and the related work of Ford and Fulkerson (1954) for network flows.

Convex Programming: A natural extension linear programming occurs when the linear part of the inequality constraints and the objective are replaced by convex functions. Early work centered about a quadratic objective Dorfman (1951), Barankin and Dorfman(1959), Marowitz (1956) and culminated in an elegant procedure developed independently by Beale (1959), Houthakker (1959), and Wolfe (1959)who showed how aminor variant of the simplex procedure could be used to solve such problems. Also studied earlier was the case where the convex objective could be separated a nonnegative sum of terms, each convex in a single variable (Dantzig, 1956; Charnes and Lemke, 1954). The general case has been studied in fundamental by ~~Kuhn and~~ Tucker (1950) and Arrow et al. (1958)

Integer programming: important classes of nonlinear, nonconvex, discrete, combinatorial problems can be shown to be formally reducible to a linear programming type of problem, some or all whose variables must be integer valued. By the introduction of the concept of cutting planes, linear programming methods were used to construct an optimal tour for a salesman visiting Washington, D.C. and forty eight state capitals of the United States (Dantzig, Fulkerson, and Johnson, 1954). The theory was incomplete. The foundations for a rigorous theory were first developed by Gomory (1958)

Programming under Uncertainty: it has been pointed out by Madansky (1960) that the area of programming under uncertainty cannot be usefully stated as a single problem. One important class considered is a multistage where the technological matrix of input-output coefficients is assumed known, the values of the constant terms are uncertain, but joint probability distribution of their possible values is assumed to be known. A promising approach based on the decomposition principle was discussed by Dantzig and Madansky (1960).

Industrial Applications of Linear Programming

The history of the first years of linear programming would be incomplete without a brief survey of its use in business and industry. These applications began in1951 but have had such a remarkable growth in the years 1955-1960 that this use is now more important than its military predecessor.

Linear programming has been serving industrial users in several ways. First, it has provided a novel view of operations; second, it induced research in the mathematical analysis of the structure of industrial systems; and third, it has become an important tool for business and industrial management for improving the efficiency of their operations. Thus the application of linear programming to a business or industrial problem has required the mathematical formulation of the problem and an implicit statement of the desired objectives. In many instances such rigorous thinking about business problems has clarified aspects of management decision making which previously had remained hidden in a haze of verbal arguments. As a partial consequence some industrial firms have stated educational programs for their managerial personnel in which the importance of the definition of objectives and constraints on business policies is being emphasized. Moreover, scheduling industrial production traditionally has been, as in the military, based on intuitive and experience, a few rules, and the use of visual aids. Linear programming has induced extensive research in developing quantitative models of industrial systems for the purpose of scheduling production. Of course many complicated systems have not as yet been quantified, but sketches of conceptual models have stimulated widespread interest. An example of this is in the scheduling of job shop production, where M. E Salveson (1953) initiated research work with a linear programming-type tentative model. Research on job-shop scheduling is now being performed by several academic and industrial research groups (Jackson, 1957). Savings by business and industry through the use of linear programming for planning and scheduling operations are occasionally reported (Dantzig, 1957).

The first and the most fruitful industrial applications of linear programming have been to the scheduling of petroleum refineries. Mellon et al. started their pioneering work in this field in 1951 (Mollen et al., 1952). Two books have been written on the subject, one by Gifford Symonds (Symonds, 1955) and another by Alan Manne (Manne, 1956). So intense has been the development that a survey by Garvin, Crandall, John, and Spellman (1957-1) showed that there are applications by the oil industry in exploration and production and distribution as well as in refining.

The food processing industry is perhaps the second most active user of linear programming. In 1953 amajor producer first used it to determine shipping of catchup from six plants to seventy warehouses (Henderson and Schlaifer, 1954) and producer has considered applying it to a similar problem, except that in this case the number of warehouses is several hundred. A major meat packer determines by means of linear programming the most economical mixture of animals feeds (Fisher and Schruben, 1953).

In the iron and steel industry, linear programming has been used for the evaluation of various iron ores and of the pelletization of low grade ores (Fabian, 1955). Additions to coke ovens and shop loading of rolling mills have provided additional applications Fabian (1955), a linear programming model of an integrated steel mill is being developed (Fabian, 1958). It is reported that the British industry has used linear programming to decide what products their rolling mills should make in order to maximize profit.

Metalworking industries use linear programming for shop loading (Morin,1955) and for determining the choice between producing and buying a part (Lewis, 1955;Mynard, 1955). Paper mills use it to decrease the amount of trim losses (Eisemann, 1957; Land and Doig, 1957; Paul and Walter, 1955;Doig and Belz, 1956).

The optimal routing of messages in a communication network (Kalabaet al., 1956), contract award problems Goldstein (1952), Gainen (1955), and the routing of aircraft and ships Dantzig and Fulkerson(1954), Ferguson and Dantzig(1954; 1956)are problems that have been considered for application of linear programming methods by the military and are under consideration by industry. In France the best program of investment in electric power has been investigated by linear programming methods (Masse and Gibrat, 1957)

Since 1957 the number of applications has grown so rapidly that it is not possible to give an adequate treatment.

1.1.2 THE DEVELOPMENT OF LINEAR PROGRAMMING UP TILL NOW

The development of linear programming has been ranked among the most important scientific advances of the mid-20th century and we must agree to this assignment. Its impact since just 1950's has been extraordinary. Today it is a standard tool that has saved many thousands or millions of dollars for most companies or business of even moderate size in the various industrialized counties of the world; and its used in other sectors of society has been spreading rapidly. A major proportion of all scientific computations on computers are devoted to use of linear programming. Dozens of textbooks have been written about linear programming and published articles describing the important applications now number in the hundreds.

Briefly, the most common type of application of linear programming involves the general problem of allocation limited resources among competing activities in a best possible (that is optimal) way. More precisely, this problem involves selecting the level of certain activities that complete for scarce resources that are necessary to perform those activities. The choice of activity levels then dictates how much of each resource will be consumed by each activity. The variety of situations to which this description applies is diverse, indeed, ranging from the allocation of production facilities to products to the allocation of natural resources to domestic needs from portfolio selection to the selection of shipping patterns, from agricultural planning to the design of radiation therapy, and so on. However, the one common ingredient in each of these situations is the necessity for allocating resources to activities by choosing the levels of those activities.

Linear programming uses mathematical model to describe the problem of concern. The adjective linear means that all the mathematical functions in this model are required to be linear functions. The word programming does not refer here to a computer programming; rather, it is essentially the synonym for planning. Thus, linear programming involves the planning of activities to obtain an optimal results, that is a result that reaches the specific goal best (according to the mathematical model) among all feasible alternatives.

Although allocating resources to activities is the most common type of application, linear programming has numerous other important applications as well. In fact, any problem whose mathematical model fits the very general format for the linear programming model is a linear programming problem.

1.1.3 METHODS OF SOLUTION OF LINEAR PROGRAMMING

There are many available methods for solving linear programming problems. Some of these methods are described below.

1. The simplex method: the simplex algorithm is a basis-exchange algorithm that solves linear programming problems by constructing a feasible solution at a vertex of the polytope and then walking along a path on the edges of the polytope to vertices with non-decreasing values of the objective function until an optimum is reached

2. Revised simplex method: The revised simplex methodis a scheme for ordering the computations required for the simplex algorithm so that unnecessary calculations are avoided.

3. Dualsimplex method: The dual simplex method is based on the duality theory. It can be thought of as the minor image ~~of the~~ simplex method. It deals with a problem as if the simplex method were being applied simultaneously to its dual problem.

4. Decomposition Principle: For solving problems of large size, it is not advisable to use the simplex or the revised simplex method. For such problems, the decomposition principle is the appropriate method to use.

1.2 PROBLEM STATEMENT

Students have been using linear programming models without understanding the details of the theory. This thesis will, therefore, provide easy to understand and logical review of essentials of linear programming theory with illustrations to students.

1.3 OBJECTIVES OF THE STUDY

The specific objectives of this thesis are:

a) To review the theory of linear programming and solution method.

b) To illustrate solution methods with examples and solve a real life problem.

1.4 METHODOLOGY

Illustrative examples of linear programming will be used in this thesis. Linear programming models will be formulated from the illustrative examples. The Graphical method, Simplex method, Revised Simplex method, Dual Simplex method, Karmarkar's algorithm, Decomposition Principle and the Bounded Variable Technique will be used to solve the given linear programming example. Solution method in this thesis will be by hand calculations and matlab software. The Information on this thesis will be obtained from Kwame Nkrumah University of Science and Technology library, internet and College of Science library, K. N. U. S.T.The following books will be used in writing this thesis;Amponsah, K. O. (2007), Optimization Techniques 1.IDL, KNUST, Kumasi, pages 25-70,Dantzig, G.B. (1963), Linear Programming and Extensions. Princeton University Press Princeton, pages 1-166, Winston, W.L. (2003), Operations Research – Applications and Algorithms, pages 24-341,Jiri, M and Gartner, B, (2007), Understanding and Using Linear Programming. Springer, Berlin. pages1-184,Kasana, H. S. and Kumar, K.D. (2004), Introductory Operations Research – Theory and

Applications, pages 1 – 192, Lueberberg, D.G. and Ye, Y (2008), Linear and Non-Linear Programming. Springer, New York, pages 11-175, Micheal, C. F. etal. (2007), Linear Programming withMatlab.Mathematical. Society and Society for Industrial and Applied Mathematics, Philadelphia, pages1-252, Primal-Dual Interior-Point Methods, pages 1-63, Vanderbei, J.R. (2008), Linear Programming-Foundations and Extensions, pages 1-160, etc.

1.5 JUSTIFICATION OF THE STUDY

For quite a long time, students in Ghana have been using linear programming models without understanding the details of the linear programming theory. Most students even find it difficult to explain solutions of linear programming questions that they have solved on their own.

This thesis will contribute to the academic development of Ghanaian students by providing them with something that can be understood and can be used easily. Students will be provided with easy to understand and logical review of essentials of linear programming theory with illustrative examples.

1.6 THESIS ORGANISATION

This thesis contains five chapters. Chapter 2 provides the literature review on the views that other people have on linear programming and the different methods of solving linear programming problems. Chapter 3 contains the methodology. Here, we discuss the algorithms that will be employed to solve the linear programming illustrative examples given. It includes how to construct mathematical models out of linear programming examples. In chapter 4, mathematical models will be constructed from the linear programming problem given.Different methods of solving linear programming problems will be used to solve the linear programming problem given. All the computational procedures in getting the required results will also be outlined. Data from Express Savings and Loans Company Limmited would be analysed. A loan policy would be formulated for (ESLCL).Chapter 5 talks about the conclusions and the recommendations that would come out of this thesis. Here, we find out whether the title of the thesis is adequate and whether the objectives of this thesis are met. Other findings from the thesis would be discussed. Based on these findings recommendations would be made.

CHAPTER 2

LITERATURE REVIEW

In this chapter, we review the work that other people have done on linear programming and the viewes that other people have on linear programming.

Linear programming problems have been intensively studied by a diverse group of researchers including mathematicians, statisticians, economics, engineers, and computer scientists. Linear programming is applied in variety of fields such as energy, transportation, telecommunication, planning, assignment etc. Linear programming problems can be solved using different methods.

Dantzig (1963), cited that Kantorovich (1939), was the first to make proposals on linear programming which were neglected during the two decades that witnessed the discovery of linear programming and its first establishment elsewhere. The method was kept secret until Dantzig (1947) published the simplex method that allowed problems with linear constraints and linear objective to be solved in theory. The history of linear programming can be traced back to the 1930's and 1940's.McCloskey (1987) described the earlier part of the history while Gass and Assad (2005) provided the recent authoritative account. An interesting collection of personal reminiscences is found in Lenstra et al. (1991) including Dantzig's (1991) contributions. After early precursors that mostly dealt with linear inequalities by Fourier (1826), more closely related work on the subject began in the mid-1930s. Motzkin's (1936) solutions of systems of linear inequalities, Leontief's (1936)work on input-output models, Kantorovich's (1936) production assignments, and Hitchcock's(1941) transportation problem are the main contributions that dealt with still isolated attempts to quantify and solve practical problems that could be reached to linear systems.

In linear programming, Manne (1953) and Orchard-Hays (1955) dealt with parametric programming (the later author's results were based on his unpublished Master's Thesis three years prior) while Orden (1952) described the product form of the inverse. Charnes(1952) dealt with an aberration of linear programming called degeneracy, and 1953 saw the emergence of the computational efficient revised simplex method of linear programming by Dantzig and Ochard-Hays(1953). The same year saw one of the first tests on linear programming written by Charnes et al. (1953). Other ground break through development of linear programming were the development of the dual simplex method by Lemke (954), the design of cutting planes for integer programming problems by Dantzig et al. (1954) and the use of branch and bound by Land and Doig (1960) also for integer programming problems, seemingly straightforward and extensions of standard linear programming problems.

Progress was however not confined to improvements of the existing tool. Markowitz (1952)applied mathematical programming to portfolio selection in finance, Charnes et al.(1952) dealt with application of linear programming in the blending of aviation fuels, Ford and Fulkerson (1954)described a linear programming problem that would enable a planner to ship as many units as pcssible from an origin to a destination through a network, and Koopmans and Beckmann(1957) were the first to describe the quadratic assignment problem, and a nonlinear structure, that was found to be applicable to facility layout problems.

Eiselt and Sandblom (2007), cited that the 1960s saw more developments in many fields related to linear programming, particularly integer programming, nonlinear programming, and flow problems. In addition, computers were beginning to make their mark in the increase problem sizes that could be solved. This was the driving force of progress for linear programming models and applications as well as for operations research in general. Edmonds (1965) introduced the notion algorithms that require a number of operations that is some polynomial function in the length of the input in the worst case should be considered efficient.

In the 1970s, the issue of algorithmic efficiency began to be taken seriously with the contribution by Cook (1971) and Karp (1972). Starting with automata theory, Cook (1971) described a probably difficult problem called Satisfiability or SAT for short and Karp (1972) described a reduction theme, according to which problem may be reduced to satisfiability. In case such a reduction exists the problem under consideration is proved to be at least as difficult as SAT. This allowed scientists to classify some problems as difficult, while others would be easy even in the worst case. Given that distinction, it was of special interest to researchers and users of linear programming when Klee and Minty (1972) determined that at least one version of the simplex method was not efficient in the worst case.

Based on earlier work by Shor (1970), Khachian (1979) described an ''ellipsoid method'' that was able to solve linear programming problems even in the worst case. The method turned out to be painfully slow in practice, so the achievement was largely theoretical. This change with the work by Karmarker (1984) whose '' interior point methods'' had the property that computational effort they required would increase only marginally as the size of the problem increased. This makes them uniquely suited to large scale practical problems. Work on improvements to methods in this class and the search for more efficient implementations of interior point methods continues to this day.

The first known instant of the diet problem actually predated development of linear programming. A solution to a small problem was first calculated without linear programming by Stigler (1945) whose guess was shown to be very close to optimal.

Gallile and Gass (1981) provided an interesting account. The basic idea behind the diet problem is simple: choose quantities of food so as to satisfy nutritional requirements and ensure that prices of the resulting diets are within reason. The theory behind this is either we minimize the cost of the diet and ensure by way of the constraints that some nutritional constraints are satisfied , or we maximize the nutritional content of the diet subject to a budget constraint Gallile and Gass (1981). Walford (2007) considered nine nutrients that are among the standard nutritional components typically considered in diets.

Cutting stock problems were first described in the early days of linear programming. Gilmore and Gomory (1961) were the first to formulate cutting stock problems using linear programming. This problem is described by Gilmore and Gomory (1961) as, given materials that are available in certain shape and sizes, cut them in order to generate certain desire shape and sizes, so as to minimize some objective such as cost. Blending problems have a long history in the application of linear programming. One of the first descriptions of blending problems deals with the blending of gasoline (Charnes et al., 1952). Their paper described a linear programming that blends airlines fuels and adds chemicals, so as to ensure that prespecified performance levels are attained, example, vapour pressure, lead and sulphur content and other specifications. The objective was to maximize profit.

Hitchock (1941) worked on transportation problems, but his work was later shown to be aspecial case of it. Dantzig and Thapa (1997; 2003), Eiselt et al. (1987) also worked on a very largetransportation problem with special techniques referred to as modified distribution technique (MODI) or stepping stone method. Routing problems are discussed in Eiselt and Sandblom (2003). Transportation problems with reshipments using linear programming were first introduced by Dwyer (1975) and have subsequently been discussed by Finke (1977). Finke (1983) outlined conditions under which reshipments yield saving beyond the optimal solution of the standard transportation problems. The simple existence conditions derived by the author require dual variables. Another extension of the standard transportation problem is referred to as a problem with over shipments. It allows additional units to flow through the network beyond those presently available at the origins and those demanded at the destinations, (Finke, 1983). Such shipments may actually result in cost savings, sometimes dubbed more -for-less paradox. The paradox is due to Swarc (1971), Charnes and Klingman (1971). Eiselt and Sandblom (2000), Ahuja et al.(1993) and Murty(1992) described most transportation scenarios. They put models that deal with transportation in the context of network flow models.

Early work on assignment problems started with Egervary's (1931) combinatorial theorem and Kuhn's (1955) ''Hungarian algorithm'', so called in honour of Egervary's contribution. A story that was used tc sell the problem soon after its appearance in the 1950s is the ''Marriage problem''. Eiselt et al. (1987), Dantzig and Thapa (2003) described the standard solution method for assignment problems which exploits the special structure of the formulation and is highly efficient. Historical accounts of this method were given by Kuhn (1991) and Frank (2005). A generalization of the standard assignment problem is what is known as the generalized assignment problem (GAP), (Frank, 2005).

Dantzig (1951) proved that if the feasible set is nonempty and bounded, then at least one optimal solution is located at an extreme pointand no interior point can be optimal. Many attempts were made in the 1960sand 1990s to find one of the shortest paths, as this will certainly minimize computational effort. However, none of these attempts proved successful. One of the reasons is that none of the methods hasforesight much beyond what happens when moving from one extreme point to an adjacent extreme point (other than actually making the moves), for feature required to find short paths between the present solution and the optimal points. Hirsch conjecture (1957), states that it is generally possible to form any extreme point to any other in at most m steps where m is the number of constraints. Klee and Walkup (1968) proved the Hirsch conjecture, Hirsch (1957) for problems with $n\text{-}m \leq 5$, where n is the number of variables, and disproved it for problems with unbiased polytopes.

Eiselt et al. (1987) described a good collection of rules for the primal simplex method. One of the popular methods described is the ''greatest change method", which determines pivot elements for each pivot-eligible column and chooses the pivot that results in the largest increase of the value of the objective function. Extensive test revealed that some pivot column selection rules result in savings in terms of iterations; such savings are however, achieved at the cost of additional computations required to apply. Dantzig (1963) referred to a phenomenon in which for a number of iterations the value of the objective function does not change as circling, but most authors nowadays call it cycling. Dantzig also indicated that the cycling, if not avoided or properly dealt with will prevent the simplex algorithm from being finite. Hoffman (1953) was the first to report the occurrence of cycling. The smallest linear programming problem known to us in which cycling occurs was described by Beale (1955) and Dantzig (1963) restated it. Dantzig (1963) devised one of the first techniques to prevent cycling called the lexicographic selection rule or perturbation technique.However, nowadays most authors suggest the use of the Bland's rule which is simple and straightforward to implement.

Simply speaking, the rule chooses the entering variable as the pivot-eligible variable with the smallest subscript, and in case of a tie for the variable that leaves the basis; it selects the one with the smallest subscript as well. Zhang (1991) developed another well-known simple rule, the LIFO (Last In, First Out) for preventing cycling. This rule selects as entering variable the pivot-eligible variable that most recently left the basis and a tie for the leaving variable is broken by choosing the variable that entered the basis most recently,(Zhang, 1991).Dantzig and Thapa (1998) gave some further details on this subject. Dantzig and Thapa (1997) stated that primal degeneracy occurs frequently in practice, e.g.; in network flow problems, transportation problems, etc. It is of utmost importance that commercial software codes for linear programming that are based on the simplex method properly treats degeneracy.

Linear programming is based on the theory of duality. The origin of duality dates back to John von Neumann (1947) and independently by Gale et al (1951), followed by the contribution due to Dantzig (1953). The ''optimaleGeltungszalal'' of Schmalenbach (1948) represents yet another independent development of the theory of duality in a managerial context. Dantzig (1982) has given a historical perspective of this development. The competitive system of duality allows a glimpse at the strong connection between linear programming in general on one hand and, game theory on the other hand (Dantzig, 1982). Eiselt and Sandblom (2004) described the further details of the competitive system. The roots of dual simplex method date back to the work by Lemke (1954). The dual simplex method maintains dual feasibility and complementary slackness throughout the computations, and terminates once primal feasibility is reached (Lemke, 1954). In addition to the dual simplex method, many other simplex based methods exist. Most prominently, Dantzig et al. (1956) described the primal dual simplex method, the popular revised simplex method with the added and space-saving feature of product form of the inverse. Ziont (1969) also described the criss-cross method. Eiselt et al (1987), Rardin (1998), Sierksma(2002) and Dantzig and Thapa (1997; 2003) gave the further details of these methods which are mostly of technical interest.

In most linear programming problems, many of the variables are bounded from below or from above. Such additional restrictions are referred to as secondary constraints Dantzig and van Slyke (1967). Dantzig and van Slyke (1967) originally suggested techniques known as generalized upper bounding technique that are used for solving linear programming problems with variables bounded from above. Cooper and Steinberg (1974) elaborated upon the generalized upper bounding techniques. Eiselt et al. (1987) and Dantzig and Thapa (2003) have summarized all information on the upper bounding technique.

The upper bounding technique is based on the work by Charnes and Lemke (1954) and Dantzig (1954). Many practical problems lead to linear programming models of a large scale with thousands or even tens of thousands of rows. Worse, the number of columns (i.e., variables) may easily reach millions or even billions. Invariably, larger scale problems possess special structures and have coefficient matrices with a sparsity of only a few percent, often just a fraction of a percent. The revised simplex method is then preferable to the regular simplex methods since it has lower storage requirements and is computationally more efficient, however, for really large – scale problems, not more specialized techniques are needed (Eiselt and Sandblom, 2007). Considering the computational capability at the time, the decomposition method by Dantzig and Wolfe (1960; 1961) held the promise of being able to solve quite large linear programming problems quickly and efficiently. But the method turned out to be more successful numerically than had first been hoped, while at the same time the simplex method had been improved (e.g. by virtue of the inverse in product from LU decomposition, etc.) and major linear programming software packages refined to such an extent that the Dantzig –Wolfe decomposition method was never really able to compute. The decomposition or partitioning method of Benders (1962), which can be seen as a dual to the Dantzig-Wolfe method met with the same fate. Eiselt et al. (1987) treated the Dantzig – Wolfe decomposition method and the decomposition or partitioning method of Benders (1962) in full. A method that withstood the test of time and which appears to be the only workable strategy for very large-scale problems with millions or even billions of columns is the column-generation technique. Gilmore and Gomory (1961, 1963) described a cutting stock problem that canbe used to illustrate this technique. Sierksma (2002) and Eiselt et al.(1987) provided the details accounts of the column generation technique.

The best known successful application of the column generation is the airline industry, where airplane flight crews are assigned to sequences of flights. Anbil et al. (1991)described a situation in which savings in the order of thirteen million dollars per year were reported by one major airline when improved schedules were computed. In general column generation procedures have been found as a widespread applications in software for solving large-scale linear programming problems in the area of vehicle routing and scheduling, Anbil et al (1991).

Eiselt and Sandblom (2007) stated that one of the assumptions in linear programming is that all parameters are deterministic, i.e., assumed to be known with certainty and one popular way to get around the problem, i.e., dealing with uncertainty while keeping the simple structureand efficient solution methods of linear programming, are sensitivity analysis. In essence, sensitivity analysis deals with the effects of changes on the optimal solution.

Manne (1953) originally worked on postoptimality analysis. Manne (1953) described postoptimality as investigations that deal with the changes in the optimal solution due to variation in the data. From the original work of Manne (1953), Gass and Saaty (1955), Dinkelbach (1969) and others have made important contributions on this topic. Bradley etal.(1977) also described another possibility to perform sensibility analysis. In contrast to the standard sensitivity, analysis, Bradley et al (1977) method allows simultaneous changes of the right hand values or, similarly, of the objective function coefficient. In order to initialize their hundred percent rule, the interval of the allowable changes are calculated first.

While being extremely successful in practice for the last half century, the simplex method is by no means the only solution method for linear programming. Interestingly enough, Brown and

Koopmans (1951) published the first alternative solution technique to the simplex method. Since then, a variety of methods have been suggested, none of which were able to compete with the simplex method as far as computational speed and general practicability was concerned. This changed in 1984, when Karmarker (1984) proposed a technique that has turned out efficient not only in the worst case, but is also able to solve large-scale methods within a reasonable amount of time. The idea of the transversal method by Brown and Koopmans (1951) is fairly simple. It works directly on the primal problem. It starts with some interior point $x^0$and moves in the direction of the gradient of the objective function, through the interior of the feasible set until it reaches its boundary at some x^1on source H_i.

De Marr (1983) was the first to suggest another attempt that allows movements through the interior of the feasible region. Later, Eiselt and Sandblom (1985, 1990) elaborated upon it. Interestingly, the external pivoting method ''tricks'' the simplex method into moving through the interior even though the method is designed to follow the boundary of the feasible set. According to Eiselt and Sandblom (1990), the advantage of this approach is that it enables the user to move through the interior of the feasible set to its boundary, rather than along its boundaries, thus potentially saving computational efforts. Furthermore, the method can easily be incorporated in the existing software that uses the simplex method. The disadvantage is that, due to its construction, the external column will create some degree of dual degeneracy. Murty (1986) proposed a different approach. Murty (1986) gravitational method is modeled after a physical equivalent, using an idea from ~~Fourier~~ in the 1820s. Again, the method works directly on the primal problem. In particular, it first rotates the space of decision variables, so that the gradient points directly downwards. The method starts with an interior solution, which is thought of as a ball of a liquid that follows the path of gravity.

This means that the ball will first fall onto one of the hyperplanes that bounds the feasible set, and from that point on, it will follow a path along the hyperplanes that define the feasible set to the lowest point. This point is the optimal solution. Eiselt and Sandblom (2000) proposed another technique that is also based on a physical technique. Their bounce method mimics the path of a ball that bounces of inside of the surfaces of that define the feasible set. Ignoring friction the ball will also come to rest at the lowest point of the feasible set, which is the optimal. The ellipsoid method, proposed by Khachian (1979) made quite a stir in the scientific community. Its major contribution was that, for the first time, it was proved that it was possible to solve linear programming problems in polynomial time in the worst case. However, once it became implemented, it soon turned out that the method, while making a significant theoretical contribution by performing efficiently in the worst case, did not perform on average and thus was not suitable for the solution of linear programming problems in practice. Still based on its landmark contribution, the ellipsoid method is designed to first find feasible solutions toa set of simultaneous linear equations efficiently. Khachian (1979), Nemhauser and Wolsey (1988) have provided the details the ellipsoid method. Unfortunately, this initial ellipsoid is extremely large even for small problems. This feature, coupled with a slow rate of shrinkage of the ellipsoids, is responsible for the exceeding slow rate of convergence of the ellipsoid method. However, the method will converge in a number of steps that is polynomial in the size of the problem, making it an efficient method in the worst case.

Although the simplex method is still dominant for solving linear programming problems, it has a major shortcoming: it cannot be guaranteed to find an optimal solution in polynomial time. On average, a problem with m structural constrains will require $\frac{1}{2}$ (3m) iterations until an optimal solution is reached. Vanderbei (2001) specifies the number as $\frac{1}{2}$ (m+n), which for a square problemwith m=n with slack variables amounts to $\frac{1}{2}$ (3m). Dantzig and Thapa (2003) have provided further details on this topic. While this number may apply on average, there are problems for which much iterationis required. For example, Klee and Minty (1972) provided an example in which the feasible region is a distorted hypercube in R^n, will force the simplex method to visit all 2^n extreme points of the feasible region before reaching optimum. This is the reason why researchers have been trying to find methods that can be proved to exhibit polynomial behavior. Apparently, this was already accomplished by the logarithmic potential method of Frisch (1956), which was further developed by Parisot (1961); the simplex splitting technique due to Levin (1965) as well as the center method of Huard (1967) which was fully developed by Reneger (1988).

However, it was the ellipsoid method by Khachian (1979) that was the first widely known polynomial-time algorithm for general linear programming problems.

The simplex method was thus unrivalled for the solution of practical linear programming problem, until the modern development of interior point methods. They were first rediscovered by Karmarker (1984), whose projectivescaling method was able to compete with the simplex method as applied to realistic problems. A remarkable feature of computer implementations of the interior point methods is that the number of iterations required by the method does not increase very rapidly with the size of the problem Karmarker (1984). As a matter of fact, it appears that less than one hundred iterations are sufficient even for the solution of very large problems with millions of variables. This is in stark contrast to the number of iterations required by the simplex method. The simplex method moves along the boundary of the feasible region from one extreme to an adjacent extreme point. Typically, real-life problemsinclude an astronomical number of extreme points McMullem (1970) has shown that for a linear programming problem with n variables and m constraint, there could be as many as

$\begin{pmatrix} m- & \left[\dfrac{n+1}{2}\right] \\ m- & n \end{pmatrix} + \begin{pmatrix} m- & \left[\dfrac{n+2}{2}\right] \\ m- & n \end{pmatrix}$ extreme points. Even for small problems with n and m in the hundreds, this number can easily surpass10^{100}, a number of extreme points that is far too large to be examined, even if only a tiny fraction would have to be dealt with. It is one of the great achievements of the simplex method that, on average, it generally agreed that it needs to explore no more than about $\frac{3}{2}$m of the existing extreme points. Since that time, the class of interior point methods has been developed to the extent that some commercial software packages for large scale linear programming now offer interior point methods as alternatives to the simplex method. Padberg (1995), Saigal (1995), Radin (1998), Bhatti (2000), Vanderbei (2001), Sierksma (2002), Dantzig and Thapa (2003), and Roos et al. (2006) have treated the interior point methods in details. According to Roos et al (2006), interior point methods approach an optimal point through a sequence of interior points. Starting with some initial interior point, the method moves through the interior of the feasible set along some improving direction to another interior point. There, a new improving direction is found, along which a move made to yet another interior point. This process is repeated, resulting in a sequence of interior points that converge to an optimal boundary point. There are essentially two different approaches. Dikin (1967) used an approach of interior point method called affine scaling method. Dikin (1967) rescaled the problem in order to make the current point stay some distance away from any boundary constraint and then restrict the step length, so that the next move will not reach the boundary.

The other possibility is used in what is known as the Newton step barrier method. Although the affine scaling method does not have a polynomial worst case bound, Karmaker (1984) used a similar method which has a polynomial worst-case bound. With that method, Roos et al. (2006) have described versions with a complexity of$O(n^{3.5}$ L), where L is the length of a binary encoding of the problem. For an instructive survey regarding barrier methods, Fiacco (1979) used the barrier method to ensure that he stayed away from the boundary of the feasible set. Fiacco and McMormick (1968) showed that when using the barrier method, the resulting maximal point will depend on the choice of U, and if U were to be allowed to parametrically approach zero, the optimal point would tend to the true optimum for the original for the original objective function z = cx. Although the affine scaling and primal-dual interior point methods are used in matrix version, Choleski and Bunch Parlet (1971) avoided this in practice by solving the related set of simultaneous linear equations by efficient numerical techniques known from linear algebra. Dantzig and Thapa (2003) and Roos et al (2006) have treated the details on this approach. Roos et al. (2006) reported that implementations of variants of the primal-dual interior point method rarely use more than 50 and mostly around 20 iterations for convergence to optimal solutions with 8-digit accuracy.

Eiselt et al. (1987) and Cooperand Steinberg (1970) have described in detail about variables that are unrestricted in sign problems. It has been explained that; in linear programming, variables that do not fit into the standard or canonical form, constraint and objective functions that do not appear to lend themselves to ~~linear~~ programming can be transformed to fit into standard or canonical form. These variables are said to be unrestricted in sign Eislet (1987). Appa and Smith (1973), Norback and Morris (1980) stated that applications of variable found in the reshipment models of transportation problems can be found in the reshipment models of transportation problems. A survey by Hobson and Weinkam (1979) also indicated that application of reformulations can be found in the area of curve fitting. It has been explained further that the actual formulation of the objective will depend on the definition of proximity. One possibility is to minimize the sum of absolute deviations of the observed points from the hyperplane (Weinkam, 1979).

Basow (1959) is the first author to describe bottleneck programming problems.Eiselt and Sandblom (2007) defined Bottleneck linear programming problems as mathematical formulations with a special type of objective functions that minimizes the maximal cost coefficient of any variable with strictly positive value. Glicksberg and Gross (1953) were the first to describe a special case of bottleneck problems and then by Gross (1959). Hammer (1969) later made a new variety of theoretical and methodological developments about bottleneck programming.

Edmond and Fulkerson (1970), Galfinkel and Rao (1971; 1976), Kaplan (1976) and Posner and Wu (1981) stated that applications of bottleneck linear programming problems are found in areas such as political districting and location models. Wu (1981) formulated two simple numerical problems on bottleneck linear programming. Simmons (1972) put forwardMinimaxandMaximin linear programming models. Minimax and Maximin linear programming models have certain remembrance to the bottleneck problems. They also have a Minimax or Maximin objective functions, but in the basic model, the objective is to either minimize the maximal value of any of the given decision variables; while bottleneck problems minimize the attribute of the variable with the largest subscript Simmons (1972). Simmons (1972) stated that application of minimax problems in general are found in fields such as the routing towards emergency facilities where the most congestion in the system is districts, so as to maximize the district with the smallest number of potential customers.

Frenk and Schaible (2004) gave a more general and in-depth treatment of fractional programming. A useful transformation from a nonlinear to a linear programming problem is possible if the objective is fractional Frenk and Schaible (2004). They used a popular application of this transformation to derive a linear programming problem on data envelopment analysis.

Multi-objective linear programming problems are very useful areas in optimization (Cohon, 1998). Eiselt and Sandblom (2004) and Cohon(1998) summarized multi-objective linear programming models and techniques and described the following terminologies: multi-criteria-decision making or MCDM occurs whenever multiple concerns (objective or criteria) exist. The field of multi-criteria decision making is commonly subdivided into multi attribute decision making or MADM on the one hand and multi-objective (linear) programming or MO(L)P on the other hand. The difference is that in multi-attribute decision making, the decision maker is to choose between a finite numbers of already existing solutions, while multi- objective decision programming problems include a number of objectives that are to be optimized, typically in continuous space. Ballestero and Romero (1998) provided a brief account of the history multi-objective optimization methods. The origins of the field can be traced back to Koopmans (1951) and Kuhn and Tucker (1951). The formal contribution introduces the notion of domination in the field; while the second paper develops optimality conditions.Geoffrion (1967) was the first to discuss Bicfiterion models, i.e., models with two objectives. A major impact was the first conference on multi-criteria decision making at the University of South Carolina in 1972. Cochrane and Zeleny (1973) published the proceedings of conference. After that milestone, activity in the field increased tremendously, witnessed by the thousands of references collected Sadler (1984) who already collected about 1,700 references in the mid-1980s.

One important feature of optimization models with multi-objectives is that the concept of optimality, in the way it is for single–objective optimization problems, does no longer applyEiselt and Sandblom (2007). Usually, decision makers employ the concept of parato-optimality first put forward by the Italian economics Parato (1906). Yu (1973) appears to be the first to determine parato-optimaal solutions for multi-objective programming problems. Yu (1973) stated that a solution or decision is called parato-optimal, if there is no other feasible solution that is equal or better to all objectives included in the model. One possibility to generate all efficient extreme points is the multi-objective simplex method. This technique is due to Evans and Steuer (1973), Philip (1972) and Zeleny (1973). Ehrgott (2005) gave a more recent account of the multi-objective simplex method. As Ballestero and Romero (1998) report, the multi-objective simplex method has been to solve problems to about 50 decision variables and three objective functions by the ADBASE problem developed by Steuer (1995). A variety of methods exist that approximate the efficient frontier. Most prominently among them are the weighting method and the constraint method. Zedeh (1963) suggested that in case the decision maker is able to specify a finite tradeoff between any pair of objectives, it is possible to apply the weighting method. Collette and Siarry (2003) give a good description of the weighting method. Collette and Siarry (2003) indicated that the drawbacks of the weighting method are that; first, the decision maker has to make some very strong statements regarding the ranking of the objectives, and secondly, the lower- ranking objectives are highly unlikely to be considered at all. The constraint method dates back to Marglin (1967). Marglin (1967) stated that the basic idea is to transform all but one objective to constraints with unknown right-hand side value which represent different target values or achievement levels.

Benayoun et al. (1971) developed another multi-objective simplex methods called the interactive methods which were used in order to incorporate the decision maker's input in a procedure that shuttles back and forth between the decision maker, who specifies some input and the analyst, who incorporate the decision maker's input in the model and resolves it. Steuer (1986), Shin and

Ravindran (1991) and Hussein and Al-Ghaffer (1996) offered reviews of the interactive procedures. Reeves and Franz (1985) provide a good summary of interactive methods while Sakawa (2002) and Chen et al. (2005) are the recent references. Eiselt and Sandblom (2004) describe another method call the Reference point programming. Eiselt and Sandvlon (2004) indicated that the Reference point programming methods require that the decision maker specifies ideal or least ideal points for each of the objectives, as well as a metric that measures the distance between the actual achievement and the yardstick, defined by the ideal point.

Buchanan and Gardiner (2003) is the recent reference that applies the concept of Reference point programming method.

Bellman and Zedeh (1970) introduced Fuzzy programming which they described as a concept in which desired achievements of the given objectives are assumed to be stated in an ambiguous way and the approach attempts to reconcile them. Much of the early work on the use of Fuzzy programming was done by Zimmerman (1976, 1978). This is an active area that even has its own journal ''Fuzzy Set and Systems''. Good surveys can be found in Zedeh (1979) and Inuiguchi and Ramik (2000) where the concept of fuzzy sets and a basic model that allows us to incorporate fuzziness in a linear optimization model are introduced. Vasant et al. (2005) presented a nonlinear fuzzy member functions which had to be dealt with by techniques from nonlinear optimization, while the inclusion of fuzzy parameters lead to possibility programming. Pawlak (1991) introduced other extension of fuzziness which is derived from the theory of rough sets. Their use in linear programming or optimization in general has, however, not been established. Charnes et al. (1955) and Charnes and Cooper (1961) described goal programming which is another way to address optimization problems in case the decision maker has provided some guidelines concerning values of objective functions. Ijiri (1965) used a preemptive priority structure for the individual goals. Ignizio (1982) summarized his work in the 1960s and 1970s, which included many applications of goal programming. Other milestones are the books by Lee (1972) and Schniederjans (1984). Newer contributions are thebooks by Ignizio and Cavalier (1994), Schniederjans (1995), and the edited collections by Trzaskalik and Michnik (2002) and Tanino et al. (2003). Simon (1957) is the first to provide the underlying concept in the goal programming approach which is the concept of satisficing. Simply speaking, the main idea of satisficing

(= satisfying + sufficing) is that the decision makers do not necessary maximize their utility, but instead, are satisfied when a predetermined tangent value has been achieved. Many extensions and practical approaches have been suggested to the basic goal programming model. Dyer (1972) discussed the early interactive techniques while Reeves and Hedin (1993) and Zykina (2004) described more recent interactive approaches. According to Eiselt and Sandblom (2007) interactive techniques are in order to incorporate in a decision maker's input in a procedure that shattles back and forth between the decision maker, who specifies some input and the analyst, who incorporates the decision maker's input in the model and resolve it. Jones and Tamiz (2002) provided a survey of the field in years up to 2000. The journal INFOR (2000) devoted two issues to a variety of aspects of goal programming.

Bilevel programming problems have a long history in economics. The economist Von Stackelberg (1934) was the first to describe bilevel programming and McGill (1973) introduced the problems into operations research field. He gave the following explanation; the structure of bilevel programming includes two levels on which decisions are made. On the upper level, there is the leader while the follower occupies the lower level.

Dempe (2002) provideda comprehensive account of bilevel programming.Vicentte (2001), Colson et al. (2005) and Fliege and Vicentte (2006) provided up-to-date surveys of bilevel programming. Vicentte (2006) gave the basic idea in bilevel programming as ''the leader is able and willing to make his decision first, not knowing what the followers will do and how they will react. Once the leader has made his decision, the followers will take the leaders decision as given and react in a way that optimizes their own objective''.

Feylizedeh and Bagherpour (2011) applied optimization techniques in production planning. They then proposed extended approach in which Multi Period Multi-Product (MPMP) problem was converted into a project management as an extension of MPMP modeling considering both network concepts and multi-objective modeling. Two objective functions were proposed by Feylizedeh and Bagherpour (2011). The first one was to minimize the cost which included inventory holding, lost sale, network crushing and overhead cost and the second one was to minimize the time completion of the planning period. The approach they proposed simultaneously integrated both project management network and linear programming modeling through a production planning context.

According to Feylizedeh and Bagherpour (2011) planning is one of the most important activities in a production factory. Also it represents the beating heart of any manufacturing process. Its purpose is to minimize production time and costs, efficiently organize the use of resources and maximize efficiency in the work place. Production planning incorporates a multiplicity of production elements, ranging from the everyday activities of staff to the ability to realize accurate delivery for the customer (Feyhlizedeh and Bagherpour, 2011). Feylilzedeh et al. (2008) stated that Multi-Period-Multi-Product (MPMP) problems consist of matching production levels of individual products to the fluctuation of demand for a number of periods, subject to the capacity constraints. However, the machine centers capacity constraints and predecessor relationship may not correctly represent the actual solution in practical cases and can lead to an infeasible solution (Feylizedeh et al., 2011). To overcome this issue, an MPMP problem can be transformed to into project work. It is thus possible to determine the sequence of operations considering the dependencies and precedence logics. These models generally can be divided into deterministic and uncertain ones. Deterministic models are analyzed by optimization techniques, usually based on linear programming

According to Feylizedeh and Baghpour (2011) many linear programming models had been carried out in several areas of production planning and control such as traditional material requirement planning. These models usually consider single objective function that minimizes total costs including production costs, inventory costs, and shortage costs subject to some constraints. For example, inventory balance, demand quantity, and capacity constraints at each period of time during production planning horizon. According to Chen and Ji (2007) traditional Material Requirement Planning (MRP) started with Master Production Schedule (MPS) which showed the quantity required to deliver to a customer within specific dates. The MPS was then translated into specific plan started with and due dates for all subassemblies and components on the basis of the product and subsequently a detailed scheduling problem was solved to meet those due dates (Chen and Jil, 2007). However, MRP normally was not considered capacity constraints that do not consider operation sequences of items (Billington et al., 1983; Taal and Wartmann, 1997). This created many problems on the shop flow for later production, such as varying workloads, changing bottleneck, etc. Moreover, the main problem was to face with infeasible production plan which caused that commitment to a customer would not be delivered on time.

In this way, Faaland and Schmit (1987) proposed a two stage heuristic model to generate feasible schedule. Sum and Hill (1993) proposed a framework to plan manufacturing processes and scheduling systems. Agrawal et al. (1996) applied precedence network to consider operation sequence and developed a heuristic algorithm based on critical path concept. According to

Feylizedeh and Bagherpour (2011) most researches have been focused on applying optimization techniques or developing efficient heuristics approaches to overcome issues available in MRP context in order to generate a feasible plan. Shanthikumar and Sargent (1983) discussed an integrated approach namely hybrid simulation/analytical modeling tempting to use advantages of both simulation and analytical modeling through a unique system. Many investigations carried out incorporated optimization models in MPMP problem. As a good example, initially, Bryne and Bakir (1999) developed a hybrid algorithm by combining mathematical programming and simulation model of a manufacturing system. They pointed out analytical methods working in co-operation with the simulation model results a better solution in comparison with the individual ones. The obtained production plan can be simultaneously both mathematically optimal and practically feasible.

Also in this respect, Kim and Kim (2001) proposed an extended linear programming model for hybrid problems. At each simulation run, actual workload of the jobs and utilization of the resources are identified. Information is then passed to the linear programming model for calculating the optimal production plan with the minimal total cost. Bryne and Hosian (2005) proposed an extended linear programming model over (Bryne and Kakir, 1999; Kim and Kim, 2001). In their model, in order to introduce the unit load concept of JIT, work load of jobs was sub-divided. While an optimal plan is sought, due to this unit load concept, the model takes account of the requirement of small lot sizes which is one factor of the JIT approach. Incorporation of the unit load concept and the modification of resource requirements and constraints in the proposed linear programming formulation are expected to help the improvement of the planning model by reducing the level of Work in Process (WIP) and total flow time (Feylizadeh et al., 2008).

As a related work in considering project and production principle through an analyzing unique system, Noori et al. (2008) proposed a fuzzy control chart application to MPMP problems. However, they considered uncertainty associated with fuzzy control chart and implemented their approach by using earned value analysis. Although, there are some works regarding crushing, this concept has not been applied in production planning or especially in MPMP problem directly. However some of the studies are as follows: Goyal (1996) gave a procedure for shortening the duration of a project at low cost. This procedure allows shortening of activities which may have been shortened initially and they happen to be exclusively in the path which has been shortened excessively. Taraghian and Taheri (2006) developed a solution procedure to study the tradeoffs of time, cost and quality in management of a project. This problem assumes the duration of tasks and quality of project activities to be discrete, non-increasing functions of a single non-renewable resource as normally assumed. Three inter-related mathematical models were developed such that each model optimized one of the given entities by assigning desired bounds on the other hand.

Feylizadeh and Bagherpour (2011) studied different forms of quality aggregations and effect of activity model reductions. Deineko and Woeginger (2001) considered the discrete modeling of the well-known time-cost tradeoff problem for project networks which had been extremely studied. Bagherpour et al. (2006) presented a new approach to adapt linear programming to solve cost time tradeoff problems. The proposed approach used two different modeling flow shops scheduling into a leveled project management network. The first model minimized make-span subject to budget limitation and the second model minimized total cost to determine optimum make span over production planning horizon (Feylizadeh et al., 2008). Abbassi and Mukattash (2001) introduced and developed a method for investigating the application of mathematical programming to the concept of crushing in Programme Evaluation and Review Technique. The main objective was the maximization of the pessimistic time estimate in Programme Evaluation and Review Technique network by investing additional amounts of money in the activities on the critical path. Azaron et al. (2007) developed a multi-objective model for the time-cost tradeoff problem in a dynamic Programme Evaluation and Review Technique network using an interactive approach.

Feylizadeh et al. (2007) presented an application of Fuzzy Goal Programming (FGP) in a flow shop scheduling problem where two objectives, namely minimization completion time and minimizing crashing costs were assumed to be considered simultaneously. Laslo (2003) described a stochastic extension of the critical path method time-cost tradeoff model. This extension included fundamental formulations of time-cost tradeoff models that represented different assumptions of the effect of the changing performance speed on the frequency distribution parameters of the activity duration, as well as the effect ot the random duration on the activity cost (Feylizadeh et al., 2008).

Coopersmith and Sumutka (2010) applied a linear programming model in tax-efficient retirement withdrawal planning. Coopersmith and Sumutka (2010) stated that a common rule for withdrawing retirement savings before tax-deferred savings, but this strategy can inflate required minimum distributions and reduce tax efficiency and wealth. However, tax-efficient withdrawing schemes can determine withdrawals that maximize the final total account balance over a retirement horizon. With US population aging and baby boomers reaching retirement age, attention focused on how nest eggs can best provide income during retirement (Coopersmith and Sumutka, 2010). Articles and papers such as Davis (2009) and Seibert and Meredith (2010) discussed policies and procedures used in retirement planning. They included distribution (or decumulation) plans that consist of sequencing withdrawing from retirement savings needed to satisfy a desired lifestyle. Distribution planning is simple if one has only tax-deferred accounts and pension. However, nest eggs often consist of various accounts that often produce different rates of return and tax liabilities (Seibert and Meredith, 2010).

Little et al. (1994) examined mathematically optimal tax- efficient withdrawals using linear programming model. Ragsdale et al. (1994) referred to a withdrawals plan as heuristic when based on a subjective rule. A tax-efficient plan is mathematically optimal when it provides the best outcome for all possible plans (Ragsdale, 1994). Subsequent papers that used heuristic tax-efficient methods expressed concern that the Ragsdale model was not applicable to the current code. This concern was not valid because optimization models could easily be adapted to changes in federal tax and estate laws. Guyton (2010) emphasized the need for tax efficiency on both a short and long-term basis using linear programming. Arvesen et al (2009) used mathematical optimization to achieve tax-efficiency for retirement savings that included taxable and tax-deferred savings. Feedback from this paper suggested comparing tax-efficient and common rule plans for a wide range of variables. Sumutka et al. (2009) demonstrated this by using linear programming models which included taxable savings and social security and allowed for tax deductions and tax brackets. The linear programming model used by Coopersmith and Sumukta (2010) made it quite time efficient for generating many alternative results. Spritzer and Singh (2006) considered initial wealth divided between taxable and tax-deferred savings and compared longevity for various portfolio scenarios assuming a flat rate of taxation using linear programming.Horan (2006) included multiple tax brackets, deductions and exemptions, but not required minimum distributions, to evaluate six rules for sequencing withdrawals.

Fletcher (2000) reviewed and applied linear programming to the Assessment Tools for Teaching and Learning Projects. Fletcher's (2000) paper reviewed the international research on linear programming as applied to the issues of banking assessment items. He outlined the mathematical procedures needed to obtained feasible solutions to selection made by teachers and constraints imposed by assessment developers. The various algorithms and heuristic procedures necessary for feasible solutions in an item band of only 500 items with testlets were discussed and exemplified. The following recommendations were made by Fletcher (2000). The use of detailed items mapping, limiting the number of ability levels, use of the simultaneous selection of items and sets method, use of the maximin model, and use of the optimal rounding method in finding solutions.

Balistreri et al. (2008) looked at various oil market simulation and optimization models conducted to date with more emphasis on the optimization model. According to Balistreri et al (2008), in 1990, Stephen Powel noted that most existing oil market models were either inter-temporal optimization or behavioural simulation and listed three models as inter-temporal optimization models. Baldwin and Prosser (1998) conducted a similar survey and followed the same classification as that of Powel (1990) and believed that most of the oil market models belonged to either recursive simulation models or inter-temporal optimization models.

CHAPTER 3

METHODOLOGY

In this chapter, we review the theory of linear programming, some methods of solution of linear programming, duality of linear programming and sensitivity analysis of linear programming.

3.1THE LINEAR PROGRAMMING CONCEPT

3.1.1DEFINITION OF LINEAR PROGRAMMING

Before defining linear programming, we first explain a linear function. A linear function is a function in which the unknown quantity in the function has an exponent of 1. The mathematical formulation to optimize profit, loss, production etc., under given set of conditions is called mathematical programming. The mathematical programming problem is written as

$$\text{Optimize } f(X) \tag{3.1a}$$

$$\text{Subject to} \quad g_i(X) \geq b_i \quad i = 1, 2, \ldots, m \tag{3.1b}$$

$$h_i(x) = b_i \quad i = m+1, \ldots, p \tag{3.1c}$$

$$q_i(x) \leq b_i \quad i = p+1, \ldots, s \tag{3.1d}$$

$$X \geq 0,$$

where $X = (x_1, x_2, \ldots, x_n)^T$ is the column vector in n – dimensional real linear space R^n. Thus, $X^T = (x_1, x_2, \ldots, x_n)$ is the row vector.

Now, from the mathematical formulation above;

(1) The function f(X) to be optimized is termed as objective function.

(2) The relations $g_i(X) \geq b_i h_i(x) = b_i,\ q_i\ (x) \leq b_i$, I = 1, 2, . . . , s are called constraints.

(3) The conditions $X \geq 0$ are nonnegativety restrictions.

(4) The variables$x_1, x_2, \ldots, x_n$ are decision variables.

(5) The terminology (optimize) stands for minimization or maximization.

If the objective function f(X) and all the constraints g_i (X) are linear in a mathematical programming problem, we call the problem **a linear programming problem.**

Linear programming involves the planning of activities to obtain optimal results, that is a result that reaches the specific goal best (according to the mathematical model) among all feasible alternatives.Linear programming is a mathematical method for determining a way to achieve the best outcome (such as maximum profit or lowest cost) in a given mathematical model for some list of requirements represented as linear relationships.

Any linear programming has the general form:

Optimize $z = c_1x_1 + c_2x_2 + \ldots + c_nx_n$ (3.2a)

Subject to

$$a_{i1}x_1 + a_{i2}x_2 + \ldots + a_{in}x_n \geq b_i,\ i = 1, 2, \ldots, m \qquad (3.2b)$$

$$x_1, x_2, \ldots, x_n \geq 0,$$

where c_k, k = 1, 2, . . . , n and b_i , i = 1, 2, . . ., m are real numbers.

Standard form of linear programming.

The standard form of linear programming is written as

$$\text{Optimize} \quad z = c_1x_1 + c_2x_2 + \ldots + c_nx_n \tag{3.3a}$$

Subject to

$$a_{i1}x_1 + a_{i2}x_2 + \ldots + a_{in}x_n = b_i \text{ , I} = 1, 2, \ldots, \text{m} \tag{3.3b}$$

$$x_1, x_2, \ldots, x_n \geq 0, \; b_1, b_2, \ldots, b_m \geq 0$$

or

$$\text{Optimize } z = c_1x_1 + c_2x_2 + \ldots + c_nx_n$$

Subject to

$$a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n = b_1$$

$$a_{21}x_1 + a_{22}x_2 + \ldots + a_{2n}x_n = b_2$$

$$\vdots \qquad \vdots \qquad \vdots$$

$$a_{m1}x_1 + a_{m2}x_2 + \ldots + a_{mn}x_n = b_m$$

$$x_1, x_2, \ldots, x_n \geq 0, \; b_1, b_2, \ldots, b_m \geq 0$$

or , in the matrix form:

$$\text{Optimize } z = C^TX$$

Subject to

$$AX = b$$

$$X \geq 0, \;\; b \geq 0,$$

where

$$C = (c_1, c_2, \ldots, c_n)^T \;\; \text{(cost vector)}$$

$$X = (x_1, x_2, \ldots, x_n) \;\text{(decision variables)}$$

$$A = (a_{ij}) \;\; \text{coefficient matrix of order m by n and}$$

$$b = (b_1, b_2, \ldots, b_m)^T \;\text{(right hand side of functional constraints).}$$

Converting to the standard form

The standard form of linear programming problem deals with nonnegativity decision variables and linear equality constraints. Here we explain how to convert the linear programming problem into the standard form in case any or both of these conditions are not available in the linear programming problem.

A linear inequality can easily be converted into an equation by introducing **slack and surplus variables.** If the ith constraint has the form

$$a_{i1}x_1 + a_{i2}x_2 + \ldots + a_{in}x_n \leq b_i,$$

We can add a nonnegative variable $s_i \geq 0$ to have

$$a_{i1}x_1 + a_{i2}x_2 + \ldots + a_{in}x_n + s_i = b_i. \tag{3.4}$$

Here, the variable s_i is called **slack variable**. Similarly, if ith constraint has the form

$$a_{i1}x_1 + a_{i2}x_2 + \ldots + a_{in}x_n \geq b_i,$$

A nonnegative variable $s_i \geq 0$ is subtracted to have

$a_{i1}x_1 + a_{i2}x_2 + \ldots + a_{in}x_n - s_i = b_i$ (3.5)

This time s_i is termed as the **surplus variable**.

Terminologies of linear programming

We explain some terminologies that are often used in linear programming.

1. Decision variables: Decision variables are variables whose values are under our control and influence the performance of the system.
2. Constraints: In most situations, only certain values of decision variables are possible. Restrictions on the values of decision variables are constraints.
3. Objective function: In any linear programming problem, the decision maker wants to maximize (usually revenue or profit) or minimize (usually cost) some function of the decision variables. The function to be maximized or minimized is the objective function.
4. Feasible region: Feasible region is the set of all points that satisfies all the constraints and sign restrictions.
5. Feasible solution: A feasible solution is the solution for which all the constraints are satisfied.
6. Infeasible solution: An infeasible solution is a solution for which at least one constraint is violated.
7. Basic feasible point: Abasic feasible point is a basic point that satisfies all the constraints and sign restrictions.
8. Basic feasible solution: A basic feasible solution is a point in the feasible region which does not give the largest objective function value.
9. Optimal solution: An optimal solution is a point in the feasible region with the largest objective function value.

10. Slack variables: Slack variables are variables that represent the unused raw materials that do not contribute to the objective function.

3.1.2 FORMULATION OF LINEAR PROGRAMMING MODEL

In this section we discuss the general characteristicsof linear programming problems, including the various legitimate forms of the mathematicalmodel for linear programming.

We begin with some basic terminology and notation.

The key terms are ***resources*** and *activities*, where m denotes the numberof different kinds of resources that can be used and n denotes the number of activitiesbeing considered. Some typical resources are money and particular kinds of machines, equipment, vehicles, and personnel. Examples of activities include investing in particularprojects, advertising in particular media, and shipping goods from a particular source toa particular destination. In any application of linear programming, all the activities maybe of one general kind (such as any one of these three examples), and then the individualactivities would be particular alternatives within this general category.

The most common type of applicationof linear programming involves allocating resources to activities. The amount availableof each resource is limited, so a careful allocation of resources to activities must bemade. Determining this allocation involves choosing the *levels* of the activities that achievethe best possible value of the *overall measure of performance.*

Certain symbols are commonly used to denote the various components of a linearprogramming model. These symbols are listed below, along with their interpretation for the general problem of allocating resources to activities.

Z = value of overall measure of performance.

x_j = level of activity j (for j =1, 2, . . . , n).

C_j= increase in Z that would result from each unit increase in level of activity j.

b_i= amount of resource i that is available for allocation to activities (for i = 1, 2, . . . , m).

a_{ij} = amount of resource i consumed by each unit of activity j.

The models pose the problem in terms of making decisions about the levels of the activities,

so x_1 , x_2, . . . , x_n are called the **decision variables.**

The values of C_j, b_i, and a_{ij}(for i = 1, 2, . . . , m and j =1, 2, . . . , n) are the *input constants* for the model. The C_j, b_i, and a_{ij} are also referred to as the **parameters of the model.**

We formulate the mathematical model for the general problem of allocating resources to activities as follows. In particular, this model is to select the values x_1, x_2, …, x_n so as to

$$\text{maximize } Z = c_1x_1 + c_2x_2 + \ldots + c_nx_2$$

Subject to the restrictions

$$a_{11}x_1 + a_{12} + \ldots + a_{1n}x_n \leq b_1$$

$$a_{21}x_1 + a_{22} + \ldots + a_{2n}x_n \leq b_2$$

$$\vdots \quad \vdots \quad \vdots \quad \vdots$$

$$a_{m1}x_1 + a_{m2}x_2 + \ldots + a_{mn}x_n \leq b_m$$

$$\text{and } x_1 \geq 0, x_2 \geq 0, \ldots, x_n \geq 0$$

Any situation whose mathematical formulation fits this model is a linear programming model.

The function being maximized $c_1x_1 + c_2x_2 + \ldots + c_xx_n$ is the objective function. The restrictions normally are referred to as constraints. The first m constraints (those with a function of all the variables

$a_{i1}x_1 + a_{i2}x_2 + \ldots + a_{in}x_n$ on the left hand side) are sometimes called functional constraints or structural constraints. Similarly, the $x_j \geq 0$ restrictions are called nonnegativity constraints or nonnegativity conditions. Table 3.1 shows the data needed for a linear programming model involving the allocation of resources to activities.

Table 3.1: Data needed for a linear programming model involving the allocation of resources to activities.

	Resources Usage per Unit of Activity				
	Activity				
Resources	1	2	…	n	Amount of Resources available
1	a_{11}	a_{12}	…	a_{1n}	b_1
2	a_1	a_{22}	…	a_{2n}	b_2
⋮	…	…	…	…	⋮
m	a_{m1}	a_{m2}	…	a_{mn}	b_m
Contribution to Z per unit of Activity	c_1	c_2	…	c_n	

3.1.3 ASSUMPTIONS OF LINEAR PROGRAMMMING

All the assumptions of linear programming actually are implicit in the model formulation given. However, it is good to highlight these assumptions so that we can moreeasily evaluate how well linear programming applies to any given problem.

Proportionality assumption

Proportionality is an assumption about both the objective function and the functional constraints. The contribution of each activity to the *value ofthe objective function Z* is *proportional* to the *level of the activity* x_j, as representedby the $c_j x_j$ term in the objective function. Similarly, the contribution ofeach activity to the *left-hand side of each functional constraint* is *proportional*to the *level of the activity*x_j, as represented by the $a_{ij} x_j$ term in the constraint.

When the function includes any *cross-product terms (*terms involving the product of two or more variables),proportionality should be interpreted to mean that *changes* in the function value are proportional to *changes* in each variable (x_j) individually, given any fixed values forall the other variables. Therefore, a cross-product term satisfies proportionality as long as each variable in theterm has an exponent of 1. However, any cross-product term violates the *additivity assumption.*

Consequently, this assumption rules out any exponent other than 1 for any variablein any term of any function (whether the objective functions or the function the left-hand side of a functional constraint) in a linear programming model.

Additivity assumption

Although the proportionality assumption rules out exponents other than 1, it does not prohibit

Cross-product terms (terms involving the product of two or more variables). The additivityassumption does rule out this latter possibility.*Every*function in a linear programming model (whether the objective functions or the function on the left-hand side of a functional constraint) is the *sum* of the *individual contributions* of the respective activities.

Divisibility assumption

This assumption concerns the values allowed for the decision variables.Decision variables in a linear programming model are allowed to have *any* values, including *noninteger*values, that satisfy the functional and nonnegativity constraints. Thus, these variables are *not* restricted to just integer values. Since each decision variable represents the level of some activity, it is being assumed that the activities can be run at *fractional levels.*

In certain situations, the divisibility assumption does not hold because some of or allthe decision variables must be restricted to *integer values.* Mathematical models with these restrictions are called integer programming models.

Certainty assumption

The certainty assumption concerns the *parameters* of the model, namely, the coefficients in the objective function c_j, the coefficients in the functional constraints a_{ij} and the right-hand sides of the functional constraints b_i. The value assigned to each parameter of a linear programming model is assumed to be a *known constant.*In real applications, the certainty assumption is seldom satisfied precisely. Linear programming models usually are formulated to select some future course of action. Therefore,the parameter values used would be based on a prediction of future conditions, which inevitably introduces some degree of uncertainty.

3.1.4 GEOMETRIC INTERPRETATION OF LINEAR PROGRAMMING

Let R^n denote the n-dimensional vector space defined on a field of real numbers. Suppose X, Y $\in R^n$.

For X = $(x_1, x_2, \ldots, x_n)^T$ and Y = $(y_1, y_2, \ldots, y_n)^T$we define the distance between X and Y as

$|X - Y| = ((x_1 - y_1)^2 + (x_2 - y_2)^2 + \ldots + (x_n - y_n)^2)^T$

Neighbourhood.Let $X_O$be a point in R^n , then σ –neighbourhood of X_O, denoted by $N_\sigma(X_O)$ is defined as the set of points satisfying

$N_\sigma(X_O) = \{X \in R^n : |X - X_O| < \sigma, \sigma > 0\}$.

$N_\sigma(X_O) \backslash X_O = \{X \in R^n : 0 < |X - X_O| < \sigma\}$ is termed as the neibourhood of X_O.

In R^2, $N_\sigma(X_O)$ is a circle without circumference, and in R^3 , $N_\sigma(X_O)$ is a sphere without boundary, and for R , an open interval on the line. For n > 3, figures are hypothetical.

Let S $\in R^n$. We give the following definitions.

Boundary point: A point X_O is called a boundary point of S if each deleted neighbourhoodof $X_O$intersects S and its compliment S^C

Interior point. A point $X_O \in S$ is said to be an interior point of S, if there exists a neighbourhood of X_O which is contained in S.

Open set: A set S is said to be open if for each X$\in$ S there exist a neighbourhood of X which is contained in S.

Close set. A set S is closed if its compliment S^C is open.

Bounded set.A set S in R^n is bounded if there exists a constant M> 0 such that $|X| \leq$ M for all X in S.

Definition: A line joining X_1 and X_2 in R^n is a set of points given by the **linear combination**

$$L = \{X \in R^n : X = \propto_1 X_1 + \propto_2 X_2, \propto_1 + \propto_2 = 1\}.$$

Obviously,

$L^+ = \{X \in X = \propto_1 X_1 + \propto_2 X_2, \propto_1 + \propto_2 = 1, \propto_2 \geq 0\}$ is a half-line originating from X_1 in the direction of X_2. For $\propto_2 = 0$, X = X_1 and $\propto_2 = 1$, X = X_2.

Similarly,

$L^- = \{X : X = \propto_1 X_1 + \propto_2 X_2, \propto_1 + \propto_2 = 1, \propto_1 \geq 0\}$ is a half-line emanating from X_2 in the direction of X_1. For $\propto_1 = 0$, X = X_2 and $\propto_1 = 1$, X = X_1.

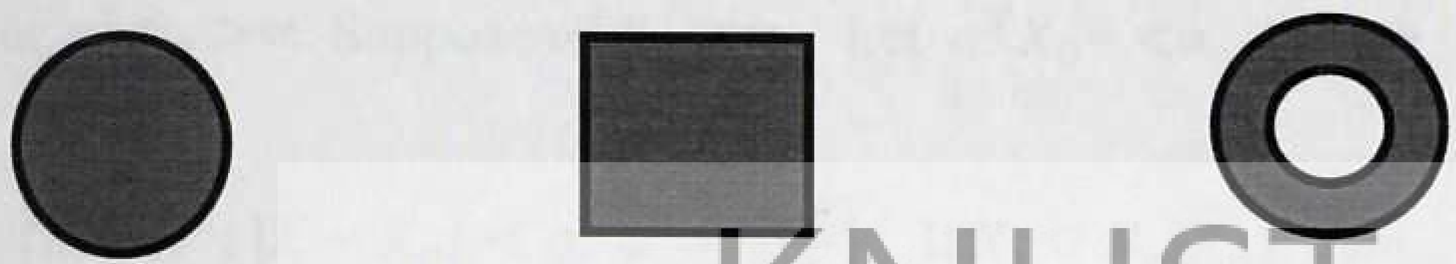
Definition: A point X$\in R^n$ is called a **convex linear combination** of two points X_1 and X_2, if it can be expressed as

$$X = \propto_1 X_1 + \propto_2 X_2, \propto_1, \propto_2 \geq 0, \propto_1 + \propto_2 = 1.$$

Geometrically speaking, convex linear combination of any points X_1 and X_2 is a line segment joining X_1andX_2.

Convex sets

Definition: A set S is said to be convex, if the linear combination of any two elements of S is also an element of S. That is, if x, y $\in$ S and $\lambda \in [0, 1]$, then z = λx + (1- λ)y $\in$ S. Geometrically speaking, a set is a convex , if all points on a straight line segment that joints any pair of arbitrary element of S are also elements of S. Figure 3.1 below gives more illustrations.



Convex set　　　　Convex set　　　　non convex set

Figure 3.1(a)　　　　Figure 3.1 (b)　　　　Figure 3.1 (c)

Hyperplanes and Half- spaces: A hyperplane is a flat geometric shape in n- dimensional space. A hyperplane in R^n is the set of points $(x_1, x_2, \ldots, x_n)^T$ satisfying the linear equation

$a_1x_1 + a_2x_1 + \ldots + a_nx_n = \beta$ or $\alpha^T X = \beta$, where $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_n)^T$. Thus, a hyperplane in R^n is the set

$$H = (\{X \in R^n : \alpha^T X = \beta\}.$$

A hyperplaneseparates the whole space into two closed half-spaces

$$H_L = \{X \in R^n:\ \alpha^T X \leq \beta\},\quad H_U = \{X \in R^n \geq \beta\}$$

Proposition: A hyperplane in R^n is a closed convex set.

Proof: The hyperplane in R^n is the set $S = \{X \in R^n : \alpha^T X = \beta\}$.

We prove that S is closed convex set. First, we show that S is closed. To do this we prove that S ComplimentS^c is open, where

$S^C = \{X \in R^n : \alpha^T X < \propto\} \cup \{X \in R^n : \alpha^T X > \propto\} = S_1 \cup S_2$. Let $X_O \in S^C$. Then $X_O \notin S^C$.

This implies

$\alpha^T X_O < \propto$ or $\alpha^T X_O > \propto$. Suppose $\alpha^T X_O < \propto$. Let $\alpha^T X_O = < \propto$. Define

$N_\sigma(X_O) = \left\{X \in R^n : |X - X_O| < \sigma,\ \sigma = \frac{\propto - \beta}{|\alpha|}\right\}$. If $X_1 \in N_\sigma(X_O)$, then

$\alpha^T X_1 - \alpha^T X_O \leq |\alpha^T X_1 - \alpha^T X_0| = |\alpha^T (X_1 - X_O| = |\alpha^T||X_1 - X_O| < \alpha - \beta$.

But $\alpha^T X_O = \beta$. This implies $\alpha^T X_1 < \propto$ and hence $X_1 \in S_1$. Since X_1 is arbitrary, we conclude that $N_\sigma(X_O) \subset S_1$. This implies S_1 is open.

Similarly, it can be shown that $S_2 = \{X : \alpha^T X_O > \propto\}$ is open.

Now, $S^C = S_1 \cup S_2$ is open (being union of open sets) which proves that S is closed.

Let $X_1, X_2 \in$ S. then $\alpha^T X_1 = \propto$ and $\alpha^T X_2 = \propto$ and considering

X = $\beta_1 X_1 + \beta_2 X_2$, $\beta_1, \beta_2 \geq 0$, $\beta_1 + \beta_2 = 1$. We note that

α^TX = $\beta_1 \alpha^T X_1 + \beta_2 \alpha^T X_2 = \beta_1 \beta + \beta_2 \beta = \beta(\beta_1 + \beta_2) = \beta$.

Thus, X$\in$ S and hence S is convex.

Proposition: Ahalf-space S = $\{X \in R^n : \alpha^T X \leq \propto\}$ is a closed convex set.

Proof: Let S $= \{X \in R^n : \alpha^T \leq \propto\}$. Suppose $X_O \in S^C$. Then $\alpha^T X_O > \propto$. Now, $\alpha^T X_O = \beta > \propto$.

Consider the neighbourhood $N_\sigma(X_O)$ defined by

$N_\sigma(X_O) = \left\{X \in R^n: \left|X - X_O\right| < \sigma, \sigma = \frac{\beta-\alpha}{|\alpha|}\right\}$. Let X_1 be an arbitrary point in $N_\sigma(X_O)$. Then

$\alpha^T X_O - \alpha^T X_1 \leq |\alpha^T X_1 - \alpha^T X_O| = |\alpha^T||X_1 - X_O| < \beta - \alpha$.

Since $\alpha^T X_O = \beta$, we have

$-\alpha^T X_1 < -\alpha \Rightarrow \alpha^T X_1 \geq \alpha \Rightarrow X_1 \in S^C \Rightarrow N_\sigma(X_O) \subset S^C$.

This implies S^C is open and hence S is closed. Take $X_1, X_2 \in$ S. Hence $\alpha^T X_1 \leq \alpha$, $\alpha^T X_2 \leq \alpha$.

For

$X = \alpha_1 X_1 + \alpha_2 X_2, \alpha_1, \alpha_2 \geq 0, \alpha_1 + \alpha_2 = 1$.

We note that

$\alpha^T X = \alpha^T(\alpha_1 X_1 + \alpha_2 X_2) = \alpha_1 \alpha^T X_1 + \alpha_2 \alpha^T X_2$

$\leq \alpha_1 \alpha + \alpha_2 \alpha = \alpha(\alpha_1 + \alpha_2) = \alpha$.

This implies $X \in$ S, and hence S is convex.

Polyhedral set. A set formed by the intersection of finite number of closed half-spaces is termed as **polyhedron or polyhedral**.

If the intersection is nonempty and bounded, it is called a **polytope**.

Remarks:

i. Generally speaking we may observe that the sets in R^n are convex if they contain no ''hole'', ''indentation'' or ''protrusion'' and non-convex otherwise.

ii. The intersection of any family of convex sets in R^n is convex.

iii. A close half-space or an open half-space in R^n is convex. Hence a hyperplane, being the intersection of two close half-space is convex.

iv. If A is an m × n matrix and b is an m-vector, then the set of solution of the linear system Ax = b , being the intersection of finite number of hyperplanes in R^n, is convex. Hence the set of all x satisfying the condition AX = b, X ≥ 0, is convex, since it is the intersection of a convex set and a half-space, which is convex set.

Extreme points

Definition: For any convex set S, a point P in S is an extreme point if each line segment that lies completely in S and contains the point P has P as an end point of the line segment. Extreme points are sometimes **called corner points** because if the set S is a polygon, the extreme point of S will be the vertices, or corners of the polygon. For example, each point on the circumference of a circle is an extreme point of the circle.

3.1.5 FUNDAMENTAL THEOREM OF LINEAR PROGRAMMING

The maximum of the objective function f(X) of a linear programming problem occurs at least at onevertex of the feasible region P_F, provided the feasible region P_F is bounded.

Proof. Given that the linear programming problem is a maximization problem. Suppose that maximum of f(X) occurs at some point X_0 in feasible region P_F.

Thus,

$$F(X) \leq f(X_O) \;\; \forall \; X \in P_F.$$

We show that this X_O is nothing but some vertex of the feasible region P_F. Since P_F is bounded and the problem is a linear programming problem, it contains finite number of vertices $X_1, X_2, \ldots, X_n$. Hence,

$$F(X_i) \leq f(X_O), \;\; i = 1, 2, \ldots, n. \qquad (3.6)$$

By linear combination,

$$X_O = \alpha_1 X_1 + \alpha_2 X_2 + \ldots + \alpha_n X_n, \; \alpha_i \geq 0, \; \sum_{i=1}^{n} \alpha_i = 1.$$

Using linearity of f, we have

$$f(X_O) = \alpha_1 f(X_1) + \alpha_2 f(X_2) + \ldots + \alpha_n f(X_n).$$

Let

$$f(X_K) \; max\{f(X_1), f(X_2), \ldots, f(X_n)\},$$

where $f(X_K)$ is one of the values $f(X_1), f(X_2), \ldots, f(X_n)$. Then

$$f(X_O) \leq \alpha_1 f(X_K) + \alpha_2 f(X_k) + \ldots + \alpha_n f(X_k) = f(X_k) \qquad (3.7)$$

Combining (3.6) and (3.7), we have $f(X_O) = f(X_K)$. This implies that X_O is the vertex X_K and hence the results.

The minimization case can be ~~treated in~~ on parallel lines just by reversing the inequalities. Thus, we have proved that the optimum of a linear programming occurs at some vertex of the feasible region P_F, provided P_F is bounded.

Remark. The above theorem does not rule out the possibility of having an optimal solutionat a point which is not vertex. It simply says among all optimal solutions to a linear programming problem, at least one of them is a vertex.

Proposition. In a linear programming problem, if the objective function f(X) attains its maximum at an interior point of the feasible region P_F, them f is constant , provided P^F is bounded.

Proof. Given that the problem is maximization, and let X_O be an interior point of the feasible region P_F, where maximum occurs, i.e.

$$f(X) \leq f(X_O) \ \forall \ X \in P_F.$$

Assume contrary that f(X) is not constant. Thus, $X_1 \in P_F$ such that

$$F(X_1) \neq f(X_O), \quad f(X_1) < f(X_O)$$

Since P_F is nonempty bounded closed convex set, it follow that X_O can be written as a clc of two points X_1 and X_2 of P_F

$$X_O = \alpha X_1 + (1 - \alpha)X_2, \quad 0 < \alpha < 1.$$

Using linearity of f, we get

$$F(X_O) = \alpha(X_1) + (1 - \alpha)f(X_2) \Rightarrow f(X_O) < \alpha f(X_O) + (1 - \alpha) f(X_2).$$

Thus, $f(X_O) < f(X_2)$. This is a contradiction and hence the theorem.

3.1.6 GRAPHICAL METHOD OF LINEAR PROGRAMMING

The graphical method is convenient in case of two variables. The optimum value of the objective function occurs at one of the vertices of the feasible region. We exploit this result to find an optimal solution of any linear programming problem. First, we sketch the feasible region and identify its vertices. Compute the value of the objective function at each vertex and take largest of these values to decide optimal value of the objective function, and the vertex at which this large value occurs is the optimal solution. For a minimization problem, we consider the smallest value.

3.2 THE METHODS OF SOLUTION OF LINEAR PROGRAMMING

There are so many methods that can be applied to solve linear programming problems. Some of these methods are described below.

3.2.1 THE SIMPLEX ALGORITHM

The simplex method is a basis-exchange algorithm that solves linear programming problem by constructing a feasible solution a vertex of a polytope and then walking along a path on the edges of the polytope to vertices with non-decreasing values of the objective function until an optimum is reached.

To find the optimal solution by simplex method one starts from some convenient basic feasible solution (vertex), and goes to another adjacent basic feasible solution (vertex) so that the value of the objective function value is improved.

Consider the linear programming problemin standard form as the basic feasible solution is calculated in after writing the problem in this format

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

$$\text{Subject to } AX = b$$

$$X \geq 0$$

where

$$C = (c_1, c_2, \ldots, c_n)^T$$

$$X = (x_1, x_2, \ldots, x_n)^T$$

A = (a_{ij})m x n, the constraint matrix of order m x n and

$$b = (b_1, b_2, \ldots, b_m)^T$$

The above linear programming problem can also be written in the form

$$\text{maximize } Z = c_1x_1 + c_2x_2 + \ldots + c_nx_n \tag{3.8a}$$

$$\text{Subject to } x_1A_1 + x_2A_2 + \ldots + x_nA_n = b \tag{3.8b}$$

$$x_1, x_2, \ldots, x_n \geq 0, \; b \geq 0.$$

Where A $= (A_1, A_2, \ldots, A_n)$ is mxn matrix and A_1, A_2 and A_n are the first, second and nth columns of A respectively.

Let m be the rank of A, and every set of m column vectors is linearly independent. The number of equations in the variables and n – m nonbasic variables. The total number of basic feasible solutions of linear programming problem cannot exceed n!/m!(n – m)!.

Suppose a basic feasible solution $s_1, x_2, \ldots, x_m$ is avairable at our disposal. This implies

$$x_{m+1} = x_{m+2} = \ldots = x_n = 0 \tag{3.9}$$

are left as nonbasic variables. Thus, $X_B = (x_1, x_2. \ldots, x_m)^T$ is a basic vector with basis matrix given as

$$B = \begin{bmatrix} a_{11} & a_{12} \ldots & a_{1m} \\ a_{21} & a_{22} \ldots & a_{2m} \\ \vdots & & \\ a_{m1} & a_{m2} \ldots & a_{mm} \end{bmatrix}$$

Using 3.3 in 3.4, we have $x_1A_1 + x_2A_2 + \ldots, + x_mA_m = b$, or

$$(A_1, A_2, \ldots, A_m)\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{bmatrix} = \mathrm{b},$$

Since, $(A_1, A_2, \ldots, A_m) = \mathrm{B}$, this has the compact form

$$\mathrm{B}X_B = \mathrm{b} \Leftrightarrow X_B = B^{-1}\mathrm{b} \qquad (3.10)$$

All column vectors of B are linearly independent (by assumption)), and hence , B is a nonsingular matrix and all column vectors of B generate R^m , an m – dimensional real linear space. This ensures that each A_j , j = 1, 2, . . . , A_n can uniquely be expressed as a linear combination of the elements of ordered basis

$\{A_1, A_2, \ldots, A_m\}$.

$$A_{m+1} = \alpha_1^{m+1}A_1 + \alpha_2^{m+1}A_2 + \ldots + \alpha_m^{m+1}A_m$$

$$A_{m+2} = \alpha_1^{m+2}A_1 + \alpha_2^{m+2}A_2 + \ldots + \alpha_m^{m+2}A_m$$

$$\vdots \quad =$$

$$A_n = \alpha_1^n A_1 + \alpha_2^n A_2 + \ldots + \alpha_m^n A_m.$$

The compact form of the above system is

$$A_j = \alpha_1^j A_1 + \alpha_2^j A_2 + \ldots + \alpha_m^j A_m, \ \mathrm{j} = \mathrm{m}+1, \mathrm{m}+2, \ldots, \mathrm{n}$$

or

$$A_j = (A_1, A_2, \ldots, A_m)\begin{bmatrix} \alpha_1^j \\ \vdots \\ \alpha_m^j \end{bmatrix} \Rightarrow A_j = B\alpha^j$$, where $\alpha^j = (\alpha_1^j, \alpha_2^j, \ldots, \alpha_m^j)^T$ is the coordinate vector of A_j. Thus we get the relation

$$\alpha^j = B^{-1}A_j \qquad (3.11)$$

Also, in view of (3.5) we have

$f(X_B) = c_1x_1 + c_2x_2 + \ldots + c_mx_m = C_B^TX_B = C_B^TB^{-1}$ b is the value of objective function at X_B,

where $C_B^T = (c_1, c_2, \ldots, c_m)$ is the cost of basic vector X_B .

Proposition 1. If $z_j = C_B^T\alpha^j$, j = 1, 2, . . ., n, then $z_j - c_j = 0$ for all basic variables.

Proof. Since $A_1, A_2, \ldots, A_m$ is an ordered basis, the coordinate vector α^j of A_j , j = 1, 2, . . . , m is (0, 0, . . ., . . . , 0), where 1 is at jth place. Hence

$$z_j = C_B^T\alpha^j = (c_1, c_2, \ldots, c_j, \ldots, c_m)\begin{bmatrix} 0 \\ \vdots \\ 1 \leftarrow \\ \vdots \\ 0 \end{bmatrix} = c_j .$$

Here, the arrow toward 1 indicates that it is at jth place. From the above relation, $z_j - c_j = 0$.

Remark. We note that $z_j - c_j$ may or may not be zero for a nonbasic variable.

In order to introduce the simplex method in algebraic terms, we introduce some notations here. For a given basic feasible solution X^* , we can always denote it by

$$X^* = \begin{bmatrix} X_B^* \\ X_N^* \end{bmatrix},$$

where the elements of vector X_B^* represent the basic variables and the elements of vector X_N^* represent nonbasic variables. Needless to mention $X_B^* \geq 0$ and $X_N^*= 0$ for the basic feasible solution.

Also, for a given cost vector C (column vector) and the coefficient matrix A, we can always denote these as

$$C = \begin{bmatrix} C_B \\ C_N \end{bmatrix} \text{ and } A = [B|N],$$

where B is a mxm nonsingular matrix that is referred to as basis and N is referred to as nonbasis with dimensionality m x (n – m).

Proposition 2. Let the linear programming problem be maximize $C^T X$ subject to AX = b, X $\geq$ 0. If, for any basic feasible solution X_B^* all $z_j - c_j \leq 0$ then X_B is the optimal solution of the problem.

Proof. Once a basis is known, every feasible solution X $\in P_F$ is arranged in an order as

$$X = \begin{bmatrix} X_B \\ X_N \end{bmatrix}.$$

We note that both X_B and X_N are nonnegative. Hence the linear programming becomes

$$\text{maximize } z = C_B^T X_B - C_N^T X_N \qquad (3.12a)$$

$$\text{Subject to} \qquad B B_X + N X_N = b \qquad (3.12b)$$

$$X_B \geq 0, \; X_N \geq 0 \qquad (3.12c).$$

Equation (3.12b) implies that

$$X_B = B^{-1}b - B^{-1}NX_N \qquad (3.13)$$

Substituting (3.8) back into (3.7a) results in

$$Z = C_B^T\,(B^{-1}b - B^{-1}N\,X_N) + C_N^T X_N$$

$$= C_B^T B^{-1}b + (C_N^T - C_B^T B^{-1}N)\,X_N$$

$$= C_B^T B^{-1}b - r^T \begin{bmatrix} X_B \\ X_N \end{bmatrix} \qquad (3.14)$$

where

$$r = \begin{bmatrix} 0 \\ (B^{-1}N)^T C_B - C_N \end{bmatrix}.$$

We observe that r is an n – dimensional vector. Its m components, corresponding to basic variables are set to be zero and the remaining n – m components correspond to nonbasic variables. Also, note that objective value z^* at current basic feasible solution X^* is $C_B^T B^{-1}$ b, since $X_B^* = B^{-1}$b and $X_N^* = 0$. Consequently (7) becomes

$$z^* - z = r^T \begin{bmatrix} X_B \\ X_N \end{bmatrix} \quad \text{for each } X \in P_F$$

It is apparent that $r^T \geq 0$, i.e., every component of $C_B^T B^{-1}N - C_N^T$ is nonnegative, then $z^* - z \geq 0$ for each feasible solution $X \in P_F$. Hence z^* is optimal value and X^* is optimal solution.

Equivalently, for optimal solution to exist, we write

$$r_j = z_j - c_j = C_B^T B^{-1} A_j - c_j \geq 0,$$

where j runs over nonbasic variables as $z_j - c_j = 0$ for all basic variables. This proves the proposition.

Remark. For a minimization problem all$z_j - c_j \leq 0$ in the last iteration table is the desired condition for any basic feasible solution, X_B^* to be optimal.

Proposition3.Let $B = (X_1, X_2, \ldots, X_N)$ be a basis in a linear space R^n and let $X \notin B$ such that $X = X_1 + \alpha_2 X_2 + \ldots + \alpha_n X_n$. If $\alpha_1 = 0$ then the vector X_i can not be replaced by X to form a new basis of R^n.

Proof. Given that $B = (X_1, X_2, \ldots, X_n)$ is a basis of R^n . For any vector $X \in B$, we have the unique representation

$$2X = \alpha_1 X_1 + \alpha_2 X_2 + \ldots + \alpha_n X_n \quad (3.15).$$

If $\alpha_i = 0$, then (3.10) becomes

$$X = \alpha_1 X_1 + \alpha_2 X_2 + \ldots + \alpha_{i-1} X_{i-1} + \alpha_{i+1} X_{i+1} + \ldots + \alpha_n X_n.$$

By using commutativity and associativity repeatedly in the above, we have

$$\alpha_1 X_1 + \alpha_2 X_2 + \ldots + \alpha_{i-1} X_{i-1} + (-1)X + \alpha_{i+1} X_{i+1} + \ldots + \alpha_n X_n = 0. \qquad (3.16)$$

From (3.16), we conclude that $(X_1, X_2, \ldots, X_{i-1}, X, X_{i+1}, \ldots, X_n)$ is linearly dependent, and hence, it is not a basis. Thus, we have established that the vector whose coefficient is zero in linear representation for X can be by X to form a new basis.

Rules for entering and leaving variables

Suppose we are considering maximization and the basic vector $X_B = (x_1, x_2, \ldots, x_n)$ is at our disposal. With the help of proposition (2) we can check whether this basic feasible solution is optimal or not. If not our next step is to some other basic feasible solution (vertex) so that the current value of the objective function improves (increases) or optimal basic feasible solution is obtained. The basic matrix associated with X_B is

$$B = (A, A_2, \ldots, A_m).$$

To go to other vertex one of the basic variables from $x_1, x_2, \ldots, x_n$ is suppose to leave and other nonbasic variable will occupy its position. This is equivalent to saying that some of the column vectors from B will leave the basis and a column vector from $A_{m+1}, A_{m+2}, \ldots, A_{m+n}$ occupies its place to form a new basis as shown in figure 3.2 below.

Enters B

$B = (A_1, A_2, \ldots, A_r, \ldots, A_m)$ $\qquad$ $A = (A_1, \ldots, A_m, \ldots, A_j, \ldots, A_n)$

Leaves B

Figure 3.2: B leaving and entering basis

$$\boldsymbol{A_j} = \alpha_1^j A_1 + \alpha_2^j A_2 + \ldots + \alpha_r^j A_r + \ldots + \alpha_m^j A_m. \qquad (3.17)$$

Preposition 3 ensures that $A_1, A_2, \ldots, A_j, \ldots, A_m$ forms a new basis provided $\alpha_r^j \neq 0$. From the original basic feasible solution, we have

$$x_1A_1 + x_2A_2 + \ldots + \ldots + x_mA_m = \text{b}.$$

We use equation (10) to replace A_r in the above equation by A_j

$$(x_1 - x_r\frac{\alpha_1^j}{\alpha_r^j})A_1 + (x_2 - x_r\frac{\alpha_2^j}{\alpha_r^j})A_2 + \ldots + \frac{x_r}{\alpha_r^j}A_j + \ldots + (x_m - x_r\frac{\alpha_m^j}{\alpha_r^j})A_m = \text{b}.$$

The position of A_r has been occupied by A_m and

$$x_1 - x_r \frac{\alpha_1^j}{\alpha_r^j},\ x_2 - x_r \frac{\alpha_2^j}{\alpha_r^j}, \ldots, \frac{x_r}{\alpha_r^j}, \ldots, x_m - x_r \frac{\alpha_m^j}{\alpha_r^j} \tag{3.18}$$

isa new basic solution, i.e., it satisfies AX = b when n – m variables are keep at zero level, and (3.18) gives the new set of basic variables. Now, we have to choose A_r such that the set in (3.18) defines a feasible solution, ie, each term must be nonnegative. For this, the first requirement is that in rth term of (11), α_r^j must be positive (this is nonnegative by assumption). Note that the first basic variable $\geq$ 0 if $\alpha_1^j < 0$.

To ensure that each term in (11) is nonnegative, ie,

$$x_1 - x_r \frac{\alpha_1^j}{\alpha_r^j} \geq 0,\ x_2 - x_r \frac{\alpha_2^j}{\alpha_r^j} \geq 0, \ldots, \frac{x_r}{\alpha_r^j} \geq 0, \ldots,\ x_m - x_r \frac{\alpha_m^j}{\alpha_r^j} \geq 0$$

or

$$\frac{x_1}{\alpha_1^j} \geq \frac{x_r}{\alpha_r^j}, \frac{x_2}{\alpha_2^j}, \geq \frac{x_r}{\alpha_r^j}, \ldots, \frac{x_r}{\alpha_r^j} \geq 0, \ldots, \frac{x_m}{\alpha_m^j} \geq \frac{x_r}{\alpha_r^j}$$

Choose r(1 to m) such that x_r / α_r^j is minimum of the right side entries in the above inequalities, ie,

$$\frac{x_r}{\alpha_r^j} = \min_i \left(\frac{x_i}{\alpha_i^j},\ \alpha_i^j > 0 \right) = \theta_j,\ \text{I} = 1, 2, \ldots, \text{m}. \tag{3.19}$$

This determines r, ie, which variable x_r or column A_r leaves the basis, when its position is to be occupied by A_j(fixed) so that the resulting set $A_1, A_2, \ldots, A_j, \ldots, A_m$ forms a basis and the set in (2.12) forms a basic feasible solution.

In the above analysis A_j is arbitrary but fixed. Now our main purpose is the selection of proper A_j , i.e, the entering variable so that there is maximum improvement in the value of the objective function as compared to its value on the earlier basic feasible solution. Improving condition (3.13), the new basic feasible solution is

$\tilde{x}1 = x_1 - \theta_j\alpha_1^j , \tilde{x}2 = x_2 - \theta_j\alpha_2^j, \ldots, \tilde{x}r = \theta_j , \ldots, \tilde{x}m = x_m - \theta_j\alpha_m^j$. The value of the objective function at this new basic feasible solution is

$$\tilde{x}0 = c_1\,\tilde{x}1 + c_2\,\tilde{x}2 + \ldots + c_j\,\tilde{x}r + \ldots + c_m\,\tilde{x}m .$$

Inserting the values of $x_1 , x_2 , \ldots , x_m$, we have

$\bar{x}_0 = c_1x_1 - \frac{x_r}{\alpha_r^j}c_1\alpha_1^j + c_2x_2 - \frac{x_r}{\alpha_r^j}c_2\alpha_2^j + \ldots, + c_j\frac{x_r}{\alpha_r^j} + \ldots + c_mx_m - \frac{x_r}{\alpha_r^j}c_m\alpha_m^j + c_rx_r - \frac{x_r}{\alpha_r^j}c_r\alpha_r^j$

Here we added and subtracted c_rx_r.

Since $x_o = c_1x_1 + c_2x_2 + \ldots + c_rx_r + \ldots + c_mx_m$ and $z_j = C_B^T\alpha^j = c_1\alpha_1^j + c_2\alpha_2^j + \ldots + c_r\alpha_r^j + \ldots + c_m\alpha_m^j$. It follows that

$$\bar{x}_0 - x_0 = \theta_j\,(z_j - c_j\,).$$

Since $x_r\,/\,\alpha_r^j = \theta_j \geq 0$, for maximum increase in x_0 , we select that A_j as the entering variable for which

$$(z_j - c_j\,)$$

is the most negative.

Thus, theoretically we permit the entry of the nonbasic variable determined by the most negative (for maximization) or most positive (for minimization) nature of $\theta_j(z_j - c_j)$. But this is not convenient because we have to compute θ_j for all nonbasic variables. This will make our task time consuming. The only thing we do is to see the most negative or the most positive nature of $z_j - c_j$.

SAMMARY OF THE SIMPLEX METHOD

The following points should be noted in the application of the simplex method.

1. Write the linear programming problem in standard form.
2. The coefficient matrix A must contain the identity submatrix. The variables constituting the identity submatrix give the starting basis (BV), and the solution is b.
3. The objective function must be expressed in terms of nonbasic variables.

After the all the above is done, we summarize here how to complete all simplex iterations to reach at optimality for a maximization problem.

i. the variable (column) with the most negative coefficient will enter as basic variable. This ensures largest possible increase in objective function.

ii. The leaving variable is decided by

$$\min\left\{\frac{\text{entries of the solution}}{\text{corresponding entering column entries} > 0}\right\}.$$

This ensures feasibility.

iii. if all the entries in the z –row are ≥ 0, the optimality is reached, and the optimal solution can be reached from the tableau.

The minimization problem can be solved by converting into maximization problem as

$$\text{Min } f(x) = -\text{maximize } (-f(x)).$$

Therefore, we simply multiply the cost coefficients by -1 to convert a minimization problem into maximization. But once maximization of the problem is found, remember to multiply the maximization by - 1 for the original minimization.

Similarly,

$$\text{Maximize } f(x) = -\text{minimize } (-f(X)).$$

We can also solve minimization problem directly without converting into maximization problem. The rules for minimization problem are

i. The variable (column) with the most positive coefficient will enter as a basic variable. This ensures largest positive decrease in objective function.

ii. The leaving variable is decided by the same rule as for maximization problem which ensures feasibility.

iii. If all the entries in the in the z – row are ≤ 0, the optimality is reached and the optimal solution can be reached from the tableau.

EXCEPTIONAL CASES IN THE APPLICATION OF THE SIMPLEX METHOD

Here, we discuss some situations which are encountered during the application of the simplex method.

1. **Unbounded solution**. This may happen when the feasible region is unbounded. The feasible region is unbounded if while applying the simplex method, it is observed that all entries of the column body matrix corresponding to some nonbasic variable are nonpositive. In other words, if in any simplex iteration, the minimum ratio rule fails, the linear programming problem has unbounded solution, i.e., appropriate nonbasic variable desires to enter the basis but $\alpha^j \leq 0$ do not permits its entry. The solution becomes unbounded because the entering variable can enter the basis at an arbitrary level.
2. **Alternative optimal solution**. If in the optimal tableau (obtained from any method), the relative cost $z_j - c_j = 0$ for at least of the nonbasic variables, then alternative optimal solution exists provided PF is bounded. Bring this nonbassic variable into basis and find a new optimal solution. Again, if the optimality occurs at two or more vertices, then it also occurs at convex linear combination of these vertices. However, convex linear combination of optimal basic feasible solution may not be a basic solution.
3. **Degeneracy**. Oneof the reasons for degeneracy may be due to the presence of some redundant constraint. The system is redundant if one or more constraints in a linear programming problem are not at all essential to find the optimal solution. Such constraints should be eliminated before proceeding for simplex iterations. However, in the absence of redundant the degeneracy may occur in the linear programming problem. If in any simplex tableau, there is a tie between two or more leaving variables we can select any one of them to leave the basis but the new solution thus obtained will have

remaining such variables (tied) at zero level in the next simplex tableau. It means new basic feasible solution will degenerate.

Suppose at some stage of simplex iterations s_2 enters, and the minimum ratios for two basic variables are same, i.e., two basic variables are candidates for leaving at the same time. This will give next basic feasible solution as degenerate.

The following table is self-explanatory in this regard.

Table 3.2: Degeneracy of feasible solution

BV	x_1 x_2 $s_1 \downarrow s_2$ s_3	Solution
z	$\propto$	
$\leftarrow x_1$	β	$\propto'$
$\leftarrow x_2$	γ	β'
s_1		γ'
z	0	
s_2	1	$\frac{\propto'}{\propto}$
x_2	0	$\beta' - \beta \propto' /\propto = 0$
s_1	0	$\gamma' - \gamma \propto' /\propto$

We summarize the above discussions.

Degeneracy due to tie among leaving variables causes three possibilities:

i. Temporary degeneracy. After some iteration degeneracy disappears and

ii. Nondegenerate optimal solution is obtained.

iii. Permanent degeneracy. Degenerate optimal solution is obtained.

iv. Cyclic degeneracy. Simplex tableau starts repeating after some iterations.

In the case of permanent and temporary degeneracy the nonbasic variables which have a tie

to leave the basis can be chosen at random as the basic variable. However, it cannot be done when cycle is detected. Cycling can be detected at early stage by noting the fact that for tied variables

$$\min_{i} \left\{ \frac{x_i}{\alpha_i^j} \, , \; \alpha_i^j > 0 \right\} = 0.$$

Remark. Whatever type of degeneracy occurs in a linear programming problem, the Charne's perturbation method should be applied to solve the problem.

Considering the problem:

$$\text{Optimize} \quad z = C^T X$$

$$\text{Subject to} \quad A X = b$$

$$X \geq 0,$$

The requirement vector b, perturbed to z is given by

$$b(\varepsilon) = b + \sum_{j=1}^{n} \varepsilon^j A_j + \sum_{i=1}^{n+s} \varepsilon^{n+1} q_i,$$

whereq_i is the ith column corresponding to ith artificial variable. Since $X_B = B^{-1}$b, we have

$$X_B\,(\varepsilon) = B^{-1}\mathrm{b} + \sum_{j=1}^{n} \varepsilon^j B^{-1} A_j + \sum_{i=1}^{n+s} \varepsilon^{n+1} B^{-1} q_i$$

or

$$X_{B\,(\varepsilon)} = X_B + \sum_{j=1}^{n+s} \varepsilon^j \alpha^j .$$

Now, we take the kth component of $X_{B\,(\varepsilon)}$ as

$$X_{Bk\,(\varepsilon)} = X_{Bk} + \sum_{j=1}^{n+s} \varepsilon^j \alpha_k^j .$$

Let x_k (A_k) enter the basis. The variable $X_{Br\,(\varepsilon)}$ will leave the basis, if

$$\frac{X_{Br\,(\varepsilon)}}{a_r^k} = \min_i \left\{\frac{X_{Bi\,(\varepsilon)}}{\alpha_r^k},\ \alpha_i^k > 0\right\}$$

and

$$Q_k(\varepsilon) = \min_i \left\{\frac{X_{Bi\,(\varepsilon)}}{\alpha_i^k} + \sum_{j=1}^{n+s} \varepsilon^j \frac{\alpha_i^j}{\alpha_i^k},\ \alpha_i^k > 0\right\}$$

$$= \min_i \left\{Q_k + \sum_{j=1}^{n+s} \varepsilon^j \frac{\alpha_i^j}{\alpha_i^k},\ \alpha_i^k > 0\right\}$$

If Q_k is not unique then cycling may occur. To avoid this, we examine

$$\min\left\{\frac{\alpha_i^1}{\alpha_i^k},\ \alpha_i^k > 0\right\} \qquad \text{(i)}$$

$$\vdots$$

$$\min\left\{\frac{\alpha_i^2}{\alpha_i^k},\ \alpha_i^k > 0\right\} \qquad \text{(ii)}$$

$$\vdots$$

$$\min\left\{\frac{\alpha_i^{n+s}}{\alpha_i^{ki}},\ \alpha_i^k > 0\right\} \qquad (n+s)$$

If (i) is unique, then we stop and the leaving variable has been decided. In case (i) is not unique then we proceed to (ii) and so on until uniqueness is achieved. Charnes has claimed that in proceeding like this way the uniqueness is necessarily obtained.

The Big M Method

After the linear programming problem is written in standard form and the coefficient matrix A in X = b does not contain identity submatrix then we extend the idea of solving the problem by the Big M method or the Two phase method.

For any equation I that does not have the slack variable, we augment an artificial variable $R_i(\geq 0)$. With the induction of artificial variables the matrix A is modified and now, it contains identity submatrix. The artificial variables then become part of the starting basic feasible solution. Because artificials are extraneous in linear programming model, we assign penalties to them in objective function to force to come to zero level at later simplex iterations. As M is sufficiently large positive number, the variable R_i is penalized in the objective function using $-MR_i$ in maximization problem, while by $+MR_i$ in minimization.

The Two Phase Method

As usual we write the linear programming problem in standard form and seek the presence of identity submatrix in coefficient matrix A. When A does not contain the identity submatrix, the addition of artificial variable is used to do so. During phase – I, we find a basic feasible solution of the system of constraint with the help of an auxiliary objective function to be minimized by using simplex iterations.

Once Phase-I is done, we go for Phase-II that tests whether the basic feasible solution obtained in Phase-I is optimal in reference to the main objective function. In case this basic feasible solution is not optimal, we continue further simplex iterations to reach at optimality.

3.2.2 THE REVISED SIMPLEX METHOD

The revised simplex method is a scheme for ordering the computations required for the simplex method so that unnecessary calculations are avoided. In fact, if pivoting is eventually required in all columns, but the number of rows m is smaller compared to the number of columns n; the revised simplex method can frequently save computational effort.

The way of executing algorithm of the simplex method, either in algebraic or tabular form, is not the most efficient computational procedure for computers because it computes and stores many numbers that are not needed at the current iteration and that may not even become relevant for decision making at subsequent iterations. The only pieces of information relevant at each iteration are the coefficients of the nonbasic variables, the coefficients of the entering basic variable in the other equations; and the right-hand sides of the equations. It would be very useful to have a procedure that could obtain this information efficiently without computing and storing all other coefficients. These considerations motivated the development of the revised simplex method. This method was designed to accomplish exactly the same things as the original simplex method, but in a way that is more efficient for execution on computer. Thus, it is a streamlined version of the original procedure. It computes and stores only the information that is currently needed, and it carries along the essential data in a more compact form.

The revised simplex method explicitly uses matrix manipulations, so it is necessary to describe the problem in matrix notations. Using matrices, the general linear programming model becomes

$$\text{Maximize} \quad Z = cx$$

$$\text{Subject to}$$

$$Ax \leq b$$

$$X \geq 0,$$

Where c is the row vector, c = $[c_1, c_2, \ldots, c_n]$. X, b and O are column vectors such that

$$X=\begin{bmatrix} x1 \\ x2 \\ . \\ . \\ . \\ xn \end{bmatrix}, \quad b=\begin{bmatrix} b1 \\ b2 \\ . \\ . \\ . \\ bm \end{bmatrix}, \quad O=\begin{bmatrix} 0 \\ 0 \\ . \\ . \\ . \\ 0 \end{bmatrix} \text{ and A is the matrix } A=\begin{bmatrix} a_{11} & a_{12} & a_{1n} \\ a_{21} & a_{22} & a_{2n} \\ \cdots & \cdots & \cdots \\ a_{m1} & a_{m2} & a_{mn} \end{bmatrix}.$$

To obtain the augmented form of the problem, we introduce the column vector of slack variables

$X_s = \begin{bmatrix} xn+1 \\ xn+2 \\ . \\ . \\ . \\ xn+m \end{bmatrix}$ so that the constraints become $[A, I]\begin{bmatrix} X \\ Xs \end{bmatrix} = b$ and $\begin{bmatrix} X \\ Xs \end{bmatrix} \geq 0$, where I is the m x m identity matrix, and the null matrix O now has the n + m elements.

Solving for a Basic Feasible Solution

We recall that the general approach of the simplex method is to obtain a sequence of improving basic feasible solutions until an optimal solution is reached. One of the key features of the revised simplex method involves the way in which it solves each new basic feasible solution after identifying its basic and nonbasic variables. Given these variables, the resulting basic solution is the solution of the m equations

$[A, I]\begin{bmatrix} X \\ Xs \end{bmatrix} = b$, in which the n nonbasic variables from the n + m elements of $\begin{bmatrix} X \\ Xs \end{bmatrix}$ are set to zero. Eliminating these n variables by equating them to zero leaves a set of m equations in m unknowns (the basic variables).

This set of equations can be denoted by

BX_B = b, where the vector of basic variables $X_B = \begin{bmatrix} XB_1 \\ XB_2 \\ \vdots \\ XB_m \end{bmatrix}$ is obtained by eliminating the nonbasic variables from

$\begin{bmatrix} X \\ Xs \end{bmatrix}$ and the basic matrix B = $\begin{bmatrix} B_{11} & B_{12} & \cdots & B_{1m} \\ B_{21} & B_{22} & \cdots & B_{2m} \\ \cdots & \cdots & \cdots & \cdots \\ B_{m1} & B_{m2} & \cdots & B_{mm} \end{bmatrix}$ is obtained by eliminating the columns corresponding to coefficients of nonbasic variables from [A,I]. In addition, the elements of X_B, and therefore, the columns of B may be placed in a different order when the simplex method is executed.

The simplex method introduces only basic variables such that b is nonsingular, so B^{-1} always will exist. Therefore, to solve B_X = b, both sides are premultiplied by B^{-1}.

$B^{-1}BX_B = B^{-1}$b.

Since B^{-1}B = I, the desired solution for the basic variables is $X_B = B^{-1}$b.

Let C_B be the vector where elements are the objective function coefficients (including zeros for slack variables) for the corresponding elements of B_X. The value of the objective function for this basic solution is

$$Z = C_B X_B = C_B B^{-1} b$$

Matrix Form of the Current Set of Equations

The last preliminary before we summarize the revised simplex method is to show the matrix form of the set of equations appearing in the simplex tableau for any iteration of the original simplex method.

For the original set of equations, the matrix form is

$$\begin{bmatrix} 1 & -c & O \\ O & A & I \end{bmatrix} \begin{bmatrix} Z \\ X \\ Xs \end{bmatrix} = \begin{bmatrix} O \\ b \end{bmatrix}.$$

The algebraic operations performed by the simplex method (multiply an equation by a constant and add a multiple of one equation to another equation) are expressed in matrix form by premultiplying both sides of the original set of equations by the appropriate matrix. This matrix would have the same elements as identity matrix, except that each multiple for an algebraic operation would go into the sport needed to have the matrix multiplication perform this operation. Even after a series of algebraic operations over several iterations, we still can deduce what this matrix must be (symbolically) for the entire series by using what we already know about the right-hand sides of the new set of equations. In particular, after any iteration,

$X_B = B^{-1}$b and Z = $C_B B^{-1}$b, so the right-handsides of the new set of equations have become

$$\begin{bmatrix} Z \\ XB \end{bmatrix} = \begin{bmatrix} 1 & C_{BB^{-1}} \\ O & B^{-1} \end{bmatrix} \begin{bmatrix} O \\ b \end{bmatrix} = \begin{bmatrix} C_{BB^{-1}}b \\ B^{-1}b \end{bmatrix}$$

Because we performed the same series of algebraic operations on both sides of the original set of equations, we use this same matrix that premultiplies the original right-hand side to premultiply the original left- hand side. Consequently, since

$$\begin{bmatrix} 1 & C_{BB^{-1}} \\ O & B^{-1} \end{bmatrix}\begin{bmatrix} 1 & -c & O \\ O & A & I \end{bmatrix}=\begin{bmatrix} 1 & C_B B^{-1}A-c & C_B B^{-1} \\ O & B^{-1}A & B^{-1} \end{bmatrix}$$, the desired matrix form of the set of equation is

$$\begin{bmatrix} 1 & C_B B^{-1}A-c & C_B B^{-1} \\ O & B^{-1}A & B^{-1} \end{bmatrix}\begin{bmatrix} Z \\ X \\ Xs \end{bmatrix}=\begin{bmatrix} C_B B^{-1}b \\ B^{-1}b \end{bmatrix}$$

The overall procedure

There are two key implications from the matrix form of the current set of equations.

The first is that only B^{-1} needs to be derived to be able to calculate all the numbers in the simplex tableau original parameters of the problem.

The second is that any one of these numbers can be obtained individually, usually by performing only a vector multiplication (one row times one column) instead of a complete matrix multiplication. Therefore, the required numbers to perform an iteration of the simplex method can be obtained as needed without expending the computational effort to obtain all the numbers. These two key implications are incorporated into the following summary of the overall procedure.

Summary of the Revised Simplex Method

1. Iterations

Step 1. Determine the entering basic variable. This is the least right-hand-side ratio.

Step 2.Determine the leaving basic variable. This is the most negative z_j - c_j value for maximization problem and most positivetive z_j - c_j value for minimization problems.

Step 3. Determine the new basic feasible solution: Derive B^{-1} and set $X_B = B^{-1}$b.

2. Optimality test: Check if all thez_j - c_j values are zeros and positives for maximization and zeros and negatives for minimization problems.

In step 3 of an iteration, B^{-1} could be derived each time by using a standard computer routine for inverting a matrix. However, since B (and thereforeB^{-1}) changes a little from one iteration to the next, it is much more efficient to derive the new B^{-1} (denoted it by B_{new}^{-1}) from B^{-1} at the preceding iteration (denote it by B_{old}^{-1}). For the initial basic feasible solution,

$$B = I = B^{-1}.$$

The method for doing this is described below:

Let x_k = entering basic variable

a'_{ik} = coefficient of x_k in current equation (i), for I = 1, 2, . . . , m (calculated in step 2 of an iteration)

r = number of equation containing the leaving basic variable.

We recall that the new set of equations can be obtained from the preceding set by subtracting a'_{ik}/a'_{rk}times equation (r) from equation (i), for all i= 1, 2, . . . ,m except i = r, and then dividing equation (r) bya'_{rk}

Therefore, the element in row i and column j of B_{new}^{-1} is

$$(B_{new}^{-1})_{ij} = \begin{cases} (B_{old}^{-1})_{ij} - {a'_{ik}}/{a'_{rk}}\,(B_{old}^{-1})_{rj}\ , & \text{if } i \neq r, \\ \\ {1}/{a'_{rk}}\,(B_{old}^{-1})_{rj}\ , & \text{if } i = r \end{cases}$$

These formulas are expressed in matrix notation as

$$B_{new}^{-1} = EB_{old}^{-1},$$

where matrix E is an identity matrix except that its rth column is replaced by the vector,

$$\omega = \begin{bmatrix} \omega 1 \\ \omega 2 \\ \vdots \\ \omega m \end{bmatrix}, \text{ where } \omega_i = \begin{cases} -{a'_{ik}}/{a'_{rk}}\ , & \text{if } i \neq r, \\ \\ {1}/{a'_{rk}}\ , & \text{if } i = r \end{cases}$$

Thus E = $[\mu_1, \mu_2, \ldots, , \ldots, \mu_{r-1}, \mu, \mu_{r+1}, \ldots, \mu_m]$, where the m elements of each of the μ_i column vectors are 0 except for a 1 in the ith position.

3.2 .3 THE BOUNDED VARIABLE TECHNIQUE

We consider the linear programming problem in which the variables x_j are bounded by their lower bounds l_j and upper bounds u_j .

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

Subject to

$$a_{i1}x_i + a_{i2}x_2 + \ldots + a_{in}x_n \geq, =, \leq b_i$$

$$l_i \leq x_j \leq u_j, \quad i = 1, 2, \ldots, m, \; j = 1, 2, \ldots, n$$

Since $x_j \geq l_j \Rightarrow y_i = x_j - l_j \geq 0$, it follows that the lower bounds may be converted to 0 just replacing x_j by $y_j + l_j$. We note that $0 \leq y_j \leq u_j - l_j$ and the simplex method is not applicable directly. Such type of linear programming problem in which some or all of the variables are having lower pound 0 and upper bound a finite number are solved using the bounded variable technique described as follows:

Assume that we have a basic feasible solution of the above system which satisfies the upper bounds. We note that no upper bound is mentioned for slack or surplus or artificial variable, and is taken at infinity. Suppose that basic feasible solution

$$X_B = \{xB_1, xB_2, \ldots, xB_m\}$$

is available by introducing slack, surplus and artificial variables. Suppose x_j is the nonbasic variable with most negative relative cost $z_j - c_j$. Then x_j enters. To decide the leaving variable the following conditions are desired.

i. the next solution must be basic feasible solution.

ii. All basic variables in next basic feasible solution must satisfy the upper bound limits.

Table 3.3: Initial Bounded Variable Technique Tableau

B V	x_1 or A_1	x_j or A_j	solution
Z	z_1 - c_1	z_j - c_j	f (X_B)
x_i or A_i	α_1^1	α_1^j	x_{B1}
$\vdots$	$\vdots$	$\vdots$	$\vdots$
x_r or A_r	α_r^1	α_r^j	x_{Br}
$\vdots$	$\vdots$	$\vdots$	$\vdots$
x_m or A_m	α_m^1	α_m^j	x_{Bm}

The basic matrix corresponding to X_{B1} from the table above is given by

B = (A_1, . . . , A_r , . . ., A_r), where A_i are the column s of the coefficient matrix A and α_i^j is the coordinate vector of the column A_j of A with respect to B.

$$A_j = \alpha_1^j A_1 + . . . + \alpha_r^j A_r + . . . + \alpha_m^j A_m.$$

We have assumed that x_j enters (column A_j) the basis and x_r(columnA_r) leaves. The new basis is

$$B^* = (A_1, . . ., A_r - 1, A_j , A_j + 1, . . ., A_m).$$

So far we are performing simplex iterations. The value of the basic variables are given by

$$x_{Bi}^* = x_{Bi} - \alpha_i^j x_j = x_{Bi} - \alpha_i^j \frac{xB_r}{\alpha_r^j}. \qquad (3.20)$$

From the above, the leaving variables xB_r should be determined so that the desired conditions (i) and (ii) are satisfied. From the theory of the simplex method, we know that the next solution will be a basic feasible solution if the leaving variable is decided by the minimum ratio rule, i.e.,

$$\theta_1 = \frac{xB_r}{\alpha_r^j} = \min\left\{\frac{xB_i}{\alpha_i^j},\ \alpha_i^j > 0\right\} = x_{Br}^* = x_j . \qquad (3.21)$$

The variable enters at level θ_1. This satisfies (1), but we have the additional condition that no variable must exceed its upper limit. This is achieved by (1) as

$$x_{Bi}^* = xB_i - \alpha_i^j x_j \le u_i. \qquad (3.22)$$

If $\alpha_i^j \ge 0$, then condition (ii) is met since $xB_i \le u_i$. In case $\alpha_i^j < 0$, then the bound may exceed. The relation (3) holds true in this situation, provided

$x_j \le \frac{u_i - xB_i}{-\alpha_i^j}$, for all those i such that $\alpha_i^j < 0$.

This means that

$$x_j \le \min_i \left\{\frac{u_i - xB_i}{-\alpha_i^j},\ \alpha_i^j < 0\right\} = \theta_2 \qquad (3.23)$$

Thus, x_j should not exceed its upper limit, i.e.,

$$x_j \le u_j. \qquad (3.24)$$

From (3.23), (2) and (3.24), it follows that the largest value of x_j which meets conditions (i) and (ii) is

$$\theta = \min(\theta_1, \theta_2, u_j). \qquad (3.25)$$

Now, we discuss all three possibilities:

Situation $\theta = \theta_1$, since minimum is θ_1 and x_j enters at θ_1 and $\theta_1 \leq \theta_2$ and $\theta \leq u_j$, the conditions (i) and (ii) will be met. Thus, iteration after deciding leaving and entering variables is nothing but simplex iteration.

Situation $\theta = \theta_2$ ensures that minimum in (3.25) is θ_2. Hence $\theta_1 > \theta_2$ and x_j enters at θ_2 level such that

$$\frac{u_r - xB_r}{-\alpha_r^j},\ \alpha_r^j < 0.$$

Since $\theta_1 \geq \theta_2$, the solution may not be a basic solution. The next solution will be a basic feasible solution provided x_j enters at θ_1 level. It can be made basic by the substitution

$$x_r = u_r - x_r^l,\ 0 \leq x_r^l \leq u_r\,. \qquad (3.26)$$

The substitution (3.26) means replacing x_r by x_r^l and the column to the solution column.

Situation $\theta = u_j$ implies that x_j enters at its upper bound. To make x_j at its upper bound, make the substitution

$$x_j = u_j - x_j^l,\ 0 \leq x_j^l \leq u_j\,.$$

In fact x_j does not enter the basis but remains nonbasic at its upper limit u_j. Since $\theta = u_j < \theta_1$, the new solution will not be basic. If it enters at θ_1, then the new solution will be a basic feasible solution.

3.2.4 THE DECOMPOSITION PRINCIPLE

In many linear programming problems, the constraints and variables may be decomposed in the following manner:

Constraints in set 1 only involve variables in variable set 1.

Constraints in set 2 only involve variables in variable set 2.

⋮

Constraints in set k only involve variables in variable set k.

Constraints in set k+1 may involve any variable.

The constraints in set k+1 are referred to as the central constraints. The linear programming problems that can be decomposed in this fashion can often be solved efficiently by the Dantzig – Wolfe decomposition principle.

The decomposition algorithm proceeds as follows:

Step 1: Let the variables in variable set 1 be $x_1, x_2, \ldots, x_n$.

Express the variables as a convex combination of the extreme points of the feasible region for constraint set 1 (the constraints that involve only the variables in variable set 1).

That is, if we let $P_1, P_2, \ldots, P_K$ be the extreme points of this feasible region, then any point

$\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$ in the feasible region for constraint set 1 may be written as

$$\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \mu_1 P_1 + \mu_2 P_2 + \ldots + \mu_k P_k \quad (3.32)$$

Where $\mu_1 + \mu_2 + \ldots + \mu_k = 1$, and $\mu_i \geq 0$ (I = 1, 2, . . . ,k)

Step 2: express the variables in variable set 2, $x_{n1+1}, x_{n1+2}, \ldots, x_n$ as a convex combination of the extreme points of constraint set 2's feasible region. If we let the extreme points of the feasible region be $Q_1, Q_2, \ldots, Q_m$, then any point in constraint set 2's feasible region may be written as

$$\begin{bmatrix} x_{n1+1} \\ x_{n1+2} \\ \vdots \\ x_n \end{bmatrix} = \gamma_1 Q_1 + \gamma_2 Q_2 + \gamma_m Q_m \quad (3.33)$$

Where $\gamma_i \geq 0$ (i = 1, 2, . . . , m) and $\gamma_1 + \gamma_2 + \ldots \gamma_m = 1$.

Using (3.32) and (3.33), we express the linear programming problems' objective function and centralized constraints in terms of $\mu_{i's}$ and γ_i's. After adding the constraints (called convexity constraints) $\mu_1 + \mu_2 + \ldots + \mu_k = 1$ and $\gamma_1 + \gamma_2 + \ldots + \gamma_m = 1$ and sign restrictions $\mu_i \geq 0$(i = 1, 2, . . ., k) and $\gamma_i \geq 0$ (I = 1, 2, . . ., m), we obtain the following linear programming, which is referred to as the restricted master:

Max (or min) [objective function in terms of μ_i;s and γ_i's]

Subject to [central constraint in terms of μ_i's and γ_i's].

$\mu_i \geq 0$ (i = 1, 2, . .., k)$\mu_1 + \mu_2 + \ldots + \mu_k = 1$

$\gamma_1 + \gamma_2 + \ldots + \gamma_m = 1$ (convexity constraints)

$$\gamma_i \geq 0 \quad (i = 1, 2, \ldots, m) \qquad \text{(sign restrictions)}$$

Step 4: Assume that a basic feasible solution for the restricted master is readily available, then we solve the restricted master by the standard techniques available for the optimal values of μ_i's and γ_i's.

Step 5: Substitute the optimal value of μ_i's and γ_i's found in step 4 into (3.32) and (3.33). This will yield the optimal values of $x_1, x_2, \ldots, x_n$.

3.2.4 KARMARKAR INTERIOR POINT ALGORITHM

An increase in the number of variables or constraints causes an increase inmultiplications and additions required for any iteration. The complexity of the simplex method is exponential.

In 1984, Karmarkar proposed an algorithm known as the interior point algorithm to solve large-scale linear programming problems efficiently. The beauty of the approach is that it gives the polynomial time complexity for the solution. This is remarkably an excellent improvement over the simplex method. However, the analysis is not simple but it requires projective geometry. In the simplex method, we move from a vertex to another vertex to find the vertex where the optimal solution lies. For large linear programming problems the number of vertices will be very large and this makes and this makes the simplex method very expensive in terms of computational time. In fact it has been found that the karmarkar's algorithm is fifty times faster than the simplex method.

The Karmarkar's algorithm is based on the following two observations:

1. If the current solution is near the center of the polytope, we can move along the direction of steepest descent to reduce value of f by maximum amount.
2. The solution can always be transformed without changing the nature of the problem so that the current solution lies near the center of the polytope.

The Karmarkar's algorithm requires the linear programming problem in a specific format:

$$\text{Minimize} \quad f = C^T X \qquad (3.26a)$$

Such that

$$AX = 0 \qquad (3.26b)$$

$$e^T X = 1 \qquad (3.26c)$$

$$X \geq 0 \qquad (3.26d)$$

where $X = (x_1, x_2, \ldots, x_n)$, $C = (c_1, c_2, \ldots, c_n)$, $e = (1, 1, \ldots, 1)^T$ and A is a m x n matrix.

A feasible solution vector X of the above problem is defined to be an interior solution if every variable $x_i \geq 0$. Here the feasible domain is bounded, and hence a polytope. A consistent problem in the Karmarkar's standard form certainly has a finite infimum.

Karmarkar made two assumptions for his algorithm:

(A1) $Ae = 0$, so that $X_O = (1/n, 1/n, \ldots, 1/n)T$ is an initial interior solution.

(A2) The optimal objective value of the problem is zero.

Conversion of a linear programming problem into required form.

Let the linear programming problem be given in standard form

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

Subject to

$$AX = b$$

$$X \geq 0$$

Our objective is to convert this problem into the standard form required by Karmarkar, where satisfying the assumption (A1) and (A2).

The key feature of the Karmarkar's standard form is the simplex structure, of course results in a bounded feasible region. Thus, we have to regularize the above linear programming problem by adding a bounding constraint

$$e^T X = x_1 + x_2 + \ldots + x_n \leq Q$$

For some positive integer Q derived from the feasibility and optimality considerations. In the worst case, we can choose $Q = 2^L$, where L is the problem size (number of variables). If this constraint is binding at optimality with objective value $-2^{O(L)}$, then we can show that the linear programming problem has unbounded solution.

By introducing a slack a variable x_{n+1}, we have a new linear program

$$\text{Minimize } f = C^T X \qquad (5.27a)$$

Subject to

$$AX = b \qquad (3.27b$$

$$e^T X + x_{n+1} = Q \qquad (3.27c)$$

$$X \geq 0, \; x_{n+1} \geq 0$$

In order to keep the matrix structure of A undisturbed for sparsity manipulation, we introduce a new variable $x_{n+2} = 1$ and rewrite the above constraints as

$$A X - b x_{n+2} = 0 \qquad (3.28a)$$

$$e^T X + x_{n+1} + Q x_{n+2} = 0 \qquad (3.28b)$$

$$e^T X + x_{n+1} + x_{n+2} = Q + 1 \qquad (3.28c)$$

$$X \geq 0, \; x_{n+1}, \; x_{n+2} \geq 0 \qquad (3.38d)$$

To normalize $e^T X + x_{n+1} + x_{n+2} = Q + 1$ for the required simplex structure, we apply the transformation

$$x_j = (Q + 1)\, y_j, \quad j = 1, 2, \ldots, n + 2.$$

In this way, we have an equivalent linear program

$$\text{Minimize } f = (Q + 1) C^T Y \qquad (3.29a)$$

Subject to

$$AY - by_{n+2} = 0 \quad (3.29b)$$

$$Y + y_{n+1} - Qy_{n+2} = 0 \quad (3.29c)$$

$$e^T Y + y_{n+1} + y_{n+2} = 1 \quad (3.29d)$$

$$Y \geq 0, \; y_{n+1}, y_{n+2} \geq 0. \quad (3.29d)$$

The problem above is now in the standard form required by the Karmarkar's algorithm. In order to satisfy the assumption (A1), we may introduce an artificial variable y_{n+3} with a large scale coefficient M and consider the following problem

$$\text{Minimize } f \; (Q+1)C^T Y + g \; My_{n+3} \quad (3.30a)$$

$$\text{Subject to} \quad AY - by_{n+2} - [Ae - b] \, y_{n+3} = 0 \quad (3.30b)$$

$$e^T Y + y_{n+1} - Qy_{n+2} - (n+1-Q)y_{n+3} = 0 \quad (3.30c)$$

$$e^T Y + y_{n+1} + y_{n+2} + y_{n+3} = 1 \quad (3.30d)$$

$$Y \geq 0, \; y_{n+1}, \; y_{n+2} \geq 0, \; y_{n+3} \geq 0$$

We observe that this form satisfies assumption (A1) as (1/n+3, 1/n+3, . . . , 1/n+3) is the interior point solution. It minimum value is zero (assumption (A2)).

Algorithm

The Karmarkar algorithm proceeds as

Step 1. Set k = 0, x^0 = (1/n, 1/n, . . . , 1/n)^T.

Step 2. If the desired accuracy $\varepsilon \geq 0$ such that $C^T x^k \leq \varepsilon$ is achieved, then stop withx^k as an approximation to the optimal solution.

Otherwise, go to step

Step 3. Here, we find a better solution.

D_k= diagonal matrix formed with diagonal elements as the components of x^k.

$$B_k = \begin{bmatrix} AD_k \\ 1, \ 1, \ \ldots, \ 1 \end{bmatrix}$$

$$\boldsymbol{d_k} = - \left[I - B_k^T \left(B_k B_k^T\right)^{-1} B_k \right] D_K C$$

$$\boldsymbol{y^k} = \left(\frac{1}{n}, \frac{1}{n}, \ldots, \frac{1}{n}\right)^T + \frac{\alpha}{n}\left(\frac{d_k}{\|d^k\|}\right) \text{ for some } 0 < \alpha \le 1$$

$$x^{k+1} = \frac{D_k y^k}{e^T D_k y^k}$$

Set k = k+1 and go to step 2.

Note that in this computational procedure x^k is always an interior feasible solution, D_k is an is an n – dimensional matrix with ith element of vector x^k as its ith diagonal matrix. $B_k$is the constraint matrix of ith linear programming problem in the Karmarkar's standard form; d_k is the feasible direction of the projected negative gradient; y^k is a new interior feasible solution the transformed space and x^{k+1} is a new interior feasible solution in original space.

3.2.4: COLUMN GENERATION ALGORITHM

For linear programming problems that have many variables, column generation can be used to increase the efficiency of the revised simplex method. The theoretical background of column generation algorithm can be illustrated as follows.

Minimize z = $\sum_{j=1}^{n} C_j \, x_j$(3.31)

Subjet to $\sum_{j=1}^{n} A_j\, x_j = b$ (3.31b)

$X \geq 0,$ (3.31)

where A_j, j = 1, 2, . . ., n, and b are m (< n) dimensional vectors. Assume that there is a large feasible solution x_B available, with an associated basis matrix B and cost coefficient C_B. Then the simplex multiplier (dual vector) associated with this basis is given by

$$\pi = C_B B^{-1} \qquad (3.2)$$

Now, if the reduced cost coefficients $C_j - \pi A_j \geq 0$ for all of the nonbasic variables, then the optimal solution of the linear program (3.2) is at hand. On the other hand, if is a k such that

$$\pi A_K - C_K = {}^{max}{}_{j} \pi A_j - C_j > 0, \qquad (3.33)$$

Then, the standard simplex method can be applied, by entering x_k into the basis through a pivot operation, and an adjacent (improved) basis can be found. For large – sized problems, finding the maximum in (3.33) by improving each πA_j - C_j may be computational expensive. In some cases, however, these columns can be identified as vertices of another polytope S. In such a case, the column to be entered into the basis can be chosen by solving the following subproblem

$$\text{Mazimize } \pi A - C_j. \qquad (3.34)$$

The solutions of the subproblem are then sent to the problem master for pivoting and updating. These iterations continue until optimality is reached.

3.3 DUALAITY OF LINEAR PROGRAMMING

The notation of duality is one of the most important concepts in linear programming. To each linear program defined by the matrix A, right hand side vector b and the cost vector C, there corresponds another linear program known as the dual, with the same set of data A, b and C.

3.3.1 DEFINITION OF THE DUAL PROBLEM

The dual problem of a linear programming defined directly and symmetrically from the primal (or original) linear programming model. The two problems are closely related that the optimal solution of one problem automatically provides the optimal solution to the other.

Inmost linear programming treatments, the dual is defined for various forms of the primal depending on the sense of optimization (maximization or minimization), types of constraints (≤ , ≥ or =) and orientation of the variables (nonnegative or unrestricted).

The dual problem is constructed as

$$\text{Maximize or minimize } Z = \sum_{j=1}^{n} C_j x_j$$

Subject to

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

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

The variables x_j, j = 1, 2, …, n include the surplus, slack and artificial variables, if any.

How dual problem is constructed from the primal

1. A dual variable is defined for each primal (constraint) equation.
2. The dual constraint is defined for each primal variable.
3. The constraint (column) coefficients of a primal variable define the left-hand-side coefficients of the dual constraint and its objective coefficients define the right-hand-side.
4. The objective coefficients of the dual equal the right-hand-side of the primal constraint equations.

Table 3.4: Construction of the Dual from the Primal

	Primal variables						
	x_1	x_2	$\cdots$	x_j	$\cdots$	x_n	
Dual variables	c_1	c_2	$\cdots$	c_j	$\cdots$	c_n	RHS
y_1	a_{11}	c_{12}	$\cdots$	a_{1j}	$\cdots$	a_{1n}	b_1
y_2	a_{21}	c_{22}	$\cdots$	a_{2j}	$\cdots$	a_{2n}	b_2
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\cdots$	$\vdots$	$\vdots$
y_m	a_{m1}	a_{m2}	$\cdots$	a_{mj}	$\cdots$	b_{mn}	b_m

If the primal problem is a normal maximization problem with m constraints and n variables, then the dual problem will be a normal minimization problem with m variables and n constraints. In this case, the primal and the dual may be written as follows:

Primal problem

Maximize Z = $C_1x_1 + C_2x_2 + \ldots + C_nx_n$

Subject to

$$a_{11}x_1 + a_{22}x_2 + \ldots + a_{1n}x_n \leq b_1$$

$$a_{21}x_2 + a_{22}x_2 + \ldots + a_{2n}x_n \leq b_2$$

$$\vdots \qquad \vdots \qquad \vdots$$

$$x_1 + a_{i2}x_2 + \ldots + a_{in}x_n \leq b_i$$

$$\vdots \qquad \vdots \qquad \vdots$$

$$a_{m1}x_1 + a_{m2}x_2 + \ldots + a_{mn}x_n \leq b_m$$

$$x_j \geq 0 \text{ (j = 1, 2, …, n)}$$

Dual problem

Minimize W = $b_1y_1 + b_2y_2 + \ldots + b_my_m$

Subject to

$$a_{11}y_1 + a_{21}y_2 + \ldots + a_{m1}y_m \geq C_1$$

$$a_{12}y_1 + a_{22}y_2 + \ldots + a_{m2}y_2 \geq C_2$$

$$\vdots \qquad \vdots \qquad \vdots$$

$$a_{1j}y_1 + a_{22}y_2 + \ldots + a_{mj}y_m \geq C_j$$

$$\vdots \qquad\qquad \vdots \qquad\qquad\qquad \vdots$$

$$a_{1n} + a_{2n}y_2 + \ldots + a_{mn}y_m \geq C_n$$

$$y_i \geq 0 \text{ (i = 1, 2, …, m)}$$

Table 3.5: Rules for constructing the Dual problem

Primal problem	Dual problem		
Objective	Objective	Constraint type	Variable sign
Maximization	Minimization	$\geq$	Unrestricted
Minimization	Minimization	$\leq$	Unrestricted

We must know that all primal constraints are equations with nonnegative right-hand-side and all the variables are nonnegative.

Table 3.6: Rules for constructing the dual problem

Primal problem	Dual problem
Maximization	Minimization
Constraints	Variables
$\geq$	$\leq$
$\leq$	$\geq$
Variables	Unrestricted
≥ 0	$\geq$
≤ 0	$\leq$
Unrestricted	$=$

Theorem 1. The dual of a dual is primal

Proof: Let the linear programming problem be given in canonical form

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

$$\text{Subject to } AX \geq b$$

$$X \geq 0,$$

Where A is a matrix of order m x n and $C = (c_1, c_2, \ldots, c_n)^T$, $X = (x_1, x_2, \ldots, x_n)^T$, $b = (b_1, b_2, \ldots, b_m)^T$.

Then its dual is written as

$$\text{Maximize } W = b^T Y$$

Subject to

$$A^T Y \leq C$$

$$Y \geq 0,$$

where $Y = (y_1, y_2, \ldots, y_m)^T$, and y_i is the dual variable associated with the ith constraint.

Now, we want to write the dual of the dual. To write the dual of the system above, we first write it in the form

$$\text{Minimize } W' = - b^T Y$$

Subject

$$-A^T \geq Y - C$$

$$Y \geq 0$$

The dual of the above linear programming problem is

Maximize $U' = (-C)^T U$ or minimize $U = C^T U$

Subject to $(-A^T)^T U \leq (-b^T)^T$ Subject to $AU \geq b$

$U \geq 0$, $U \geq 0$

The last linear programming problem is the primal. Hence we have proved that the dual of the dual is the primal.

3.3.2 DUALITY THEOREMS

Theorem 2 (Weak duality theorem). This theorem states that if X_O is a feasible solution for the primal problem and any Y_0 is a feasible solution for dual problem, then

$$C^T X_0 \geq b^T Y_O$$

Proof. The dual feasibility of Y_O implies that $A^T Y_O \leq C$, $Y_O \geq 0$.

If X_O is a primal feasible, then $X_O \geq 0$, and

$$X_O^T A^T Y_O \leq X_O^T C.$$

We note that $AX_O \geq$ b. Hence

$$C^T X_O = X_O^T C \geq X_O^T A^T Y_O = (AX_O)^T Y_O \geq b^T Y_O.$$

Corollary 1. If X_O is primal feasible, Y_O is dual feasible, and $C^TX_O = b^TY_O$, then X_O and Y_O are the optimal solutions to the respective problems.

Proof. Theorem 2 indicates that $C^TX \geq b^TY_O = C^TX_o$, for each primal feasible solution X. Thus, X_O is an optimal solution to the primal. A similar argument holds for the dual.

Corollary 2. If the primal is unbounded below, then the dual is infeasible.

Proof. Whenever the dual problem has a feasible solution Y_O, the weak duality theorem prevents the primal objective from failing before b^TY_O.

Corollary 3. If the dual problem is unbounded above, then the primal problem is infeasible.

The converse statement of the two corollaries above is not true. When the primal problem is infeasible, the dual could be either unbounded above or infeasible.

For a given cost vector C (Column vector) and the coefficient matrix A, we can always denote these as

$$C = \begin{bmatrix} CB \\ \hline CN \end{bmatrix} \quad \text{and} \quad A = [B/N],$$

where C_B is the cost vector of basic variables, C_N is the cost vector if nonbasic variables, B is m x m nonsingular and N is the matrix corresponding to nonbasic variables with dimensionality m x (n –m).

Theorem 3 (Strong Duality theorem)

1. If either the primal or the dual linear program has a finite optimal solution, then so does the other and they achieve the same objective value.
2. If either problem has an unbounded objective value, then the other has no feasible solution.

Proof. For the first part, without loss of generality, we can assume that the primal problem has reached a finite optimum at a basic feasible solution X. If we utilize the simplex algorithm at X and define

$$Y^T = C^T_B B^{-1}, \text{ then,}$$

$$A^T Y - C = \begin{bmatrix} B^T \\ N^T \end{bmatrix} Y - \begin{bmatrix} C_B \\ C_N \end{bmatrix} = r \leq 0.$$

Therefore Y is dual feasible. Moreover, since X is a basic feasible solution, we have

$$C^T X = C^T_B X_B = C^T_B B^{-1} b = Y^T b = b^T Y.$$

Due to corollary 1, we can say Y is an optimal solution to the dual linear problem.

3.3.3 COMPLEMENTARY SLACKNESS THEOREM

This theorem explains how the primal and the dual are closely related. The relationship between the primal and the dual reveals somany facts involving optimal solution of one from the other.

Theorem 4 (Complementary slackness conditions)

a. If, in optimal tableau of primal the decision variable x_k appears as basic variable then the kth dual constraint is satisfied as equality constraint, ie, slack or surplus variable associated with kth dual constraint assumes zero value.

b. If, in optimal tableau of primal the slack or surplus variable s_k appears as basic variable then the dual variable y_k associated with kth primal constraint assumes zero value in the optimal solution of dual.

Proof:

a. Since z_k - $c_k = 0$ for all basic variables, it follows that if z_k is a basic variable then z_k = c_k. It means $C_B^T B^{-1} A_K = c_k \Rightarrow Y^T A_K = c_k$. This implies

$y_1 a_{1k} + y_2 a_{2k} + \ldots + y_m a_{mk} = c_k \ldots$ (*), ie, kth dual constraint is satisfied as equality constraint.

b. If s_k is slack or surplus variable, then $c_k = 0$ and $A_k = (0, 0, \ldots 1, \ldots 0)^T$. Using this data in (*), we have

$y_1 \text{x}\, 0 + y_2 \text{ x } 0 + \ldots + y_k \text{ x } 1 + \ldots + y_m \text{ x } 1 = 0 \Rightarrow y_k = 0.$

Remarks: At any simplex iteration of the primal or dual, the direct consequence of complementary slackness theorem is

(Objective coefficient [relative cost] of variable j in one problem) = (Left- hand side minus right hand side of constraint j in other problem).

This property is very useful for finding optimal solution of primal or dual when the optimal solution of one is known.

Symmetric property of duality

Consider the primal in the form

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

$$\text{Subject to} \quad A X \geq b, \quad X \geq 0.$$

Its dual is given by

$$\text{Maximize} \quad b^T Y$$

$$\text{Subject to} \quad A^T Y \leq C, \; Y \geq 0$$

This is called symmetric pair of primal and dual programs.

Define the primal slackness vector

$$s = AX - b, X \geq 0 \text{ and}$$

Dual slackness vector

$$r = C - A^T Y$$

$$Y \geq 0$$

Theorem 5 (Complementary slackness theorem)

Let X be primal feasible solution and Y be dual feasible solution to a symmetric pair of linear programs. Then X and Y become an optimal solution pair if and only if

Either $\quad rr_j = (C - A^T Y)_j = 0$

or

$$X_j = 0, \ j = 1, 2, \ldots, n$$

and

either

$$s_i = (AX - b)_j = 0$$

or

$$Y_i = 0, \ i = 1, 2, \ldots, m \quad \text{are satisfied.}$$

Here X_j and Y_j represent the jth and ith component of the feasible X and Y respectively.

Proof. For any primal feasible X and dual feasible Y, we have

$$0 \leq r^T X + s^T Y = (C^T - Y^T A)X + Y^T(AX - b)$$

$$= C^T X - b^T Y$$

Therefore, the quantity $r^T + b^T Y$ is equal to the duality gap between the primal feasible solution X and dual feasible solution Y. The duality gap vanishes if and only if

$$r^T X = 0 \ \text{ and } \ s^T Y = 0.$$

In this case Xand Y become optimal solution of primal and dual respectively. $r^T X = 0$ and $s^T Y$ require that either $r_j = 0$ or $X_j = 0$ for j = 1, 2, . . .,n and either $s_i = 0$ or $Y_i = 0$ for i = 1, 2, . . .,m.

This proves the theorem.

Remark. If the primal is given in standard form

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

$$\text{Subject to } \text{ A X } = 0$$

$$\text{X } \geq 0,$$

It dual is given in the form

$$\text{Maximize } b^T \text{ Y}$$

$$\text{Subject to } A^T \text{ Y } \leq 0$$

$$\text{Y unrestricted.}$$

Since the primal has zero slackness (being tight equalities), the condition $s^T\text{Y} = 0$ is automatically met. Thus, complimentary slackness is simplified to $r^T\text{X} = 0$.

3.3.4 THE DUAL SIMPLEX METHOD

The dual simplex method is applicable when in the starting simplex tableau the optimal criterion is satisfied but the feasibility remains disturbed, while identity sub-matrix is manipulated to exist in A, the coefficient matrix. Hence, the objective function is immediately observed.

We adopt the following procedure to find the optimal solution.

1. After introducing slack or surplus variable, we write the problem in the format

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

$$\text{Subject to A X } = \text{ b}$$

$$\text{X } \geq 0,$$

Where A contains the identity matrix as sub matrix and at least one of the b_i in the right-hand side vector b = $(b_1, b_2, \ldots, b_m)^T$ is negative.

2. The objective function is expressed in terms of nonbasic variables.

Algorithm

Step 1. The leaving variable is decided to be the most negative entry of the solution column, ie,

$$x_r = \min_i (x_i \text{ , } x_i < 0).$$

Step 2. To decide the entering variable, we look for negative entries in row of leaving variable and find the ratio of these entries with the corresponding $(z_j - c_j)$'s in x_o row. Fix the entering variable by

$$\min_j \left\{ \left| \frac{z_j - c_j}{\alpha_r^j} \right|, \ \alpha_r^j < 0 \right\}.$$

Step 3. When the entering and leaving variables are decided by steps 1 and2, perform the simplex iterations to have the next tableau.

If all the entries in solution column of the resulting tableau after the iteration nonnegative values, then stop otherwise continue iterations through steps 1 and 2 till all the entries in solution column are nonnegative, i.e., the feasibility is attained.

Remarks.

1. Use of artificial variables should be avoided to produce the identity sub matrix, while applying the dual simplex method.

2. Suppose jth variable is qualified to leave the basis, but all the entries $\propto_j^k$, k = 1, 2, . . . ,n are positive which means that no variable can enter basis. In this situation the linear programming problem has no feasible solution.

3.4 SENSITIVITY ANALYSIS

Sensitivity analysis is a process which is applied to an optimal tableau of any linear programming problem when some changes are proposed in the original problem. It is sometimes referred to as **post optimal analysis.**

Given a linear programming problem in standard form, the problem is completely specified by the constraint matrix A, the right hand side vector b, and the cost vector C. We assumed that the linear programming problem has an optimal has an optimal solution with the data set (A, b,C). In many cases, we find the data set (A, b, C) needs to be changed within a range after we obtained the optimal solution, and we are interested to find the new optimal solution.

Thus, the possible changes are

(i) Change in the cost vector;

(ii) Change in the right hand side vector;

(iii) Change in the constraint matrix.

3.4.1 CHANGE IN THE COST VECTOR

The change in cost of variables has ~~a direct~~ impact on optimal criteria (z-row) which has its entries as $z_j - c_j = C_B^T B^{-1} A_j - c_j$. If the optimal criterion is disturbed due to cost change, then we use simplex method to restore optimality which results in a new solution.

Two types of changes are possible:

(i) *Change in cost of a nonbasic variables*. With the change in of nonbasic variables the relative cost of this variable is changed. Obviously, there is no change in relative cost of any other variable. If the sign of the relative cost is changed, then we bring this variable into the basis to get the new optimal solution.

Let the linear programming problem be

$$\text{maximize} \quad z = C^T X$$

$$\text{Subject to} \quad A X = b,$$

$$X \geq 0$$

Suppose x_k is a nonbasic variable and its cost c_k is changed to $c_k \Delta c_k$, where $k \in \bar{N}$, the index set of nonbasic variables. The new relative cost of x_k turns up

$$C_B^T B^{-1} A_K - (c_k + \Delta c_k), \quad k \in \bar{N}$$

Since C_B^T is fixed and cost of all remaining variables are kept fixed, there will be no change relative of any of any other variable. The optimal remains same if

$$C_B^T B^{-1} A_K - (c_k + \Delta c_k) \geq 0,$$

Otherwise the optimality is disturbed which can be restored by simplex method to find new optimal solution.

(ii) *Change in cost of a basic variable*. With change in the cost of a basic variable, all $z_j - c_j$ will change except for the basic variables. Note that $C_B^T B^{-1}$ cannot be taken from the optimal table as C_B^T has changed. There will also be a change in the objective function value.

Let c_j, the cost of jth basic variable be shifted to $c_j + \Delta c_j$, where $j \in \bar{B}$, the index set of basic variables. Then, relative cost of each nonbasic variable is changed as

$$(C_B^T + \Delta c_j e_j)\alpha^k - c_k, \quad j \in \bar{B}, \quad K \in \bar{N},$$

Where α^k and c_k are the coordinate vector and cost of kthnonbasic variable, respectively.

To stay optimal solution as it is, we must have

$$C_B^T \alpha^K - c_k + \Delta c_j \alpha_j^k \geq 0$$

This implies

$$\Delta c_j \geq \frac{z_k - c_k}{-\alpha_j^k}, \quad \text{for kth nonbasic variable.}$$

Hence, we can define

$$\underline{\Delta c\, j} = \max\left\{\frac{z_k - c_k}{-\alpha_j^k}, \ \alpha_j^k \geq 0, \ k \in \bar{N}\right\}.$$

$$\overline{\Delta c\, j} = \min\left\{\frac{z_k - c_k}{-\alpha_j^k}, \ \alpha_j^k \leq 0, \ k \in \bar{N}\right\}.$$

Which ensures the variation limits in cost of jth basic variable

$$\underline{\Delta c\, j} \leq \Delta c_j \leq \overline{\Delta c\, j}.$$

If Δc_j goes out of these limits for at least one nonbasic variable, this implies optimality is disturbed. Calculate fresh objective function value using

$(c_1, c_2, \ldots, c_j + \Delta c_j, \ldots, c_m) B^{-1}$b, and apply simplex method to restore optimality which results in new optimal solution.

3.4.2 CHANGES IN THE RIGHT HAND SIDE VECTOR

In linear programming problem, if the change in right hand side of the constraints is made then the solution column, B^{-1}b, and the objective function value, f(X_B) = $C_B^T B^{-1}$b are affected. This change corresponds to two cases.

(i) if all entries of the new solutioncolumn turn out to be nonnegative, then the existing table remains optimal with the new solution and new optimal value.

Let the linear programming problem be

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

$$\text{Subject to } AX = b,$$

$$X \geq 0.$$

Suppose right hand side entry b_k of the vector b = $(b_1, b_2, \ldots, b_m)$T is shifted to $(b_1 + b_2 + \ldots + b_k + \Delta b_k, \ldots, b_m)$T. Then the new solution column is

X_B'= B^{-1}(b + $\Delta b_k e_k$), where e_k = (0. 0, . . . , 1, . . . , 0)T has 1 at kth position.

Let B^{-1}= (β_{ik})m xm. The optimal basis remains the same if

$$X_B' = B^{-1}b + \Delta b_k B^{-1} e_k \geq 0$$

or

$$X_B + \Delta b_k \begin{bmatrix} \beta 1k \\ \beta 2k \\ \vdots \\ \beta mk \end{bmatrix} \geq \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

If $\bar{B}$ denotes theindex set of basic variables, then we have

$$X_q + \Delta b_k \beta_{ik} \geq 0, \mathrm{q} \in \bar{B}, \ \mathrm{i} = 1, 2, \ldots, \mathrm{m}$$

which gives

$$\Delta b_k \geq \frac{X_q}{-\beta_{ik}}.$$

We define

$$\underline{\Delta b k} = \max_i \left\{ \frac{Xq}{-\beta_{ik}}, \ \beta_{ik} > 0 \right\}.$$

$$\overline{\Delta b k} = \min_i \left\{ \frac{X_q}{-\beta_{ik}}, \ \beta_{ik} < 0 \right\}$$

Thus, if Δb_k satisfies

$$\underline{\Delta b k} \leq \Delta b_k \leq \overline{\Delta b k}$$

The same basis remains intact.

In this way the same set of basic variable with changed values according to $X_q + \Delta b_k \beta_{ik}$ produce the new solution and new objective value is $C_B^T X_B'$.

(ii) the second possibility is that Δb_k is assigned beyond the above limits. Then at least one of the entries in new B^{-1}b is negative, ie, feasibility is disturbed. Restore the feasibility using the dual simplex method to get the optimal solution of the revised problem.

3.4.3 CHANGE IN THE CONSTRAINT MATRIX

In general, changes made in constraint matrix may result in different optimal basis and optimal solutions. We discuss five possible changes here. They are; adding a constraint or variable, adding a variable, removing a constraint or variable and change in some column of the constraint matrix A.

1. **Addition of a constraint**. If a new constraint is added to a linear programming problem, then we have to make two observations:

i) If the constraint to be added is satisfied by the given optimal solution, then there will be no effect on adding this constraint.

ii) If the constraint is not satisfied, then addition will affect the optimal solution. Addition of such a constraint first will disturb the simplex format. When the simplex format is restored the feasibility will get disturbed. Restore the feasibility by the dual simplex method to find the new optimal solution

Remarks: 1. It is worth mentioning that addition of a constraint (provided it affect the optimal solution) always worsens the current optimal value of the objective function as the set of basic feasible solution has shrunk.

2. For equality constraint to be added, we split into two inequalities constraints. Certainly one of the inequality constraints will be satisfied by the optimal solution and other one is considered for addition in the linear programming problem.

2. Addition of a variable. Addition of a new variable causes addition of some column in the optimal table which may affect the optimal criteria. Suppose that a new variable say x_{n+1} is identified after we obtained the optimal solution X^* of the original linear programming problem. Assume that c_{n+1} is the cost coefficient associated with x_{n+1}, and A_{n+1} is the associated column in the new constraint matrix. Our aim is to find an optimal solution of the new linear program

$$\text{Max } z = C^T X + c_{n+1}x_{n+1}$$

Subject to

$$AX + A_{n+1}x_{n+1} = b$$

$$X \geq 0, x_{n+1} \geq 0.$$

Observe that we can set $x_{n+1} = 0$, then $(X, 0)^T$ becomes a basic feasible solution to the new linear programming problem. Hence, the simplex algorithm can be started right away. Also, note that X^* is an optimal solution to the original problem, the relative cost $z_j - c_j$, j = 1, 2, . . . ,n+1 must be nonnegative. Therefore, we have to check additional relative cost

$$z_{n+1} - c_{n+1} = C_B^T B^{-1} A_{n+1}.$$

If $z_{n+1} - c_{n+1} \geq 0$, then the current solution X^* with $x_{n+1} = 0$ is the optimal solution to the new problem and we don't have to do anything. On the other hand if $z_{n+1} - c_{n+1} < 0$, then z_{n+1} should be included in the basis. Continue simplex iterations till an optimal solution to the new linear programming problem is available.

2. **Deletion of a constraint**: While deleting a constraint we observe two situations:

   i) If any constraint is satisfied on the boundary, i.e. slack or surplus variable corresponding to this constraint is at zero level then deletion of such a constraint may cause change in the optimal solution.

   ii) If any constraint is satisfied in interior of the feasible region, ie, slack or surplus variable corresponding to this constraint are positive, then deletion of such a constraint will not affect the optimal solution.

3. **Deletion of a variable**. We observe for two situations:

   i) If we delete nonbasic variable or a basic variable (at zero level) in the optimal, there will be no change in the optimal solution.

   ii) However, deletion of a positive basic variable will change the optimal solution. Deleting a basic variable with positive value is equivalent to converting this into nonbasic. For the purpose, first we remove the entire column from the optimal table associated with the basic variable to be deleted and then multiply the entire row in front of this variable by - 1. This will certainly disturb the feasibility. Now, use the dual simplex method to restore feasibility.

4. Change in column of the constraint matrix. The change in coefficients associated with a variable may affect the optimal criteria. We consider the linear programming problem

$$\text{Max } z = C^T X$$

Subject to

$$AX = b$$

$$X \geq 0.$$

Two cases arise:

First, we discuss the change in coefficient of constraint matrix associated with a nonbasic variable. This will change the whole column below this variable in the optimal table.

Suppose the a_{ik}th entry of the column A_k corresponding to kth nonbasic variable is shifted to $A'_K = A_k + \sigma_{ik} e_i$, here k $\in \bar{N}$, the index set of nonbasic variables and e_i is the column vector with 1 at ith position and zero elsewhere. We decide the limits of variation a_{ik} such that the optimal solution remains same.

When A_K is changed to A'_K, this will affect the relative cost of x_k. The new relative cost of x_k become

$C_B^T B^{-1} A'_K - c_k$. The optimal table remains same if

$$C_B^T B^{-1}(A_K + \sigma_{IK} e_i) - c_k \geq 0,$$

or

$$C_B^T \propto^k + \sigma_{ik} C_B^T B^{-1} e_i - c_k \geq 0.$$

Let $B^{-1} = (\beta_1, \beta_2, \ldots, \beta_m)$. Then the above expression is simplified to

$$z_k - c_k + \sigma_{ik} C_B^T \beta_i.$$

But β_i is the coordinate vector of ith variable in the starting basis. Hence,

$$\sigma_{ik} \geq -\frac{z_k - c_k}{\beta_i}.$$

This gives the variation in element α_{ik} of the column A_K in constraint matrix to that the optimal solution remains same. Restore the optimal criteria by the simplex method to get the new optimal solution. Suppose the coefficients of constraint matrix associated with some basic variable in a linear programming problem considered above are changed. Let the column A_K associated with some basic variable x_k is changed to A'_K, where $k \in \bar{B}$. Then add a variable x'_k with same cost as that of x_k and column A'_K in constraint matrix. Compute

$$z'_k - c_k = C_B^T B^{-1} A'_K - c_k.$$

If $z'_k - c_k > 0$, there is no effect of such change, otherwise the optimality is disturbed and to restore optimality bring z'_k into the basis. Treat x_k as the artificial variable and force to come out of the basis. In the last optimal table the value of x'_k in the solution is nothing but x_k with the new column.

CHAPTER 4

NUMERICAL EXAMPLES AND RESULTS

4.1: INTRODUCTION

In this chapter, illustrative examples will be used to illustrate some methods of solution of linear programming. These methods include the Graphical method, the Simplex method, the Revised Simplex method, Dual Simplex method, Karmarkar's algorithm, the Decomposition Principle, the Bounded Variable Technique and the Column Generation method.

ILLUSTRATIVE EXAMPLE

To illustrate the Graphical method, the Simplex method and the Dual simplex method, we consider the following example.

Example 4.1: (Class assignment formulated by C Sibel, October 2010) A company manufactures two soft drinks A and B. Two resources R_1 and R_2 are required to make the drinks. Each unit of soft drink A requires 1 unit of resource R_1 and 3 units of resource R_2. Each unit of soft drink B requires 1 unit of resource R_1 and 2 units of resourceR_2. The company has 5 units of resource R_1 and 12 units of resource R_2 available. The company also makes a profit of GHC6 per unit of soft drink A sold and GHC5 per unit of soft drink B sold. How many soft drink A and soft drink B should the company manufacture to ensure a maximum profit?

CONSTRUCTION OF LINEAR PROGRAMMING MODELS

The formulation table 4.1 for the example 4.1 is shown below.

Table 4.1: Amount of resources available to produce soft drinks A and B.

Types of soft drink	Resources		Profit per unit
	R_1	R_2	
Soft drink A (x_1)	1	3	GHC6
Soft drink B (x_2)	1	2	GHC5
Amount of resources available.	5	12	

The owners of the company would like to maximize profit as much as soft drinks A and B can be but they are restricted by the availability of resources.

Let x_1 = number of units of soft drink A produced.

Let x_2 = number of units of soft drink B produced.

Let Z = total profit per day for manufacturing these two soft drinks.

Thus x_1 and x_2 are the decision variables for the model. The objective is to choose the values of x_1 and x_2 so as to maximize profit, subject to the restrictions imposed on their values by the limited production capacity of the two resources R_1 and R_2.

Profit function $Z = 6x_1 + 5x_2$,

Restrictions: $x_1 + x_2 \leq 5$

$3x_1 + 2x_2 \leq 12$

$x_1, x_2 \geq 0.$

Since negative soft drinks cannot be produced, x_1 and x_2 are called non-negativity constraints.

The canonical form (Kasana and Kumar, 2004) of the maximization linear programming model thus becomes;

$$P_1\text{:Maximize } Z = 6x_1 + 5x_2$$

$$\text{Subject to } x_1 + x_2 \leq 5$$

$$3x_1 + 2x_2 \leq 12$$

$$x_1, x_2 \geq 0,$$

Where Z= $6x_1 + 5x_2$ is the objective function, $x_1 + x_2 \leq 5$ and $3x_1 + 2x_2 \leq 12$ are the constraints and x_1, x_2 are the decision variables.

4.2: ILLUSTRATION OF THE GRAPHICAL METHOD

Replacing the inequalities in the constraints of problemP_1 by equality, we have

$$Z = 6x_1 + 5x_2$$

$$x_1 + x_2 = 5 \quad \ldots 4.1$$

$$3x_1 + 2x_2 = 12 \quad \ldots 4,2$$

Solving for points in Feasible Set

When $x_1 = x_2 = 0$, we have the point (0, 0)

From 4.1, when $x_1 = 0$, $x_2 = 5$ and when $x_2 = 0$, $x_1 = 5$. This gives the points (0, 5) and (5, 0).

From 4.2, when x_1= 0, $2x_2 = 12$, $x_2 = \frac{12}{2} = 6$ and when $x_2 = 0$, $3x_1$= 12, $x_1 = \frac{12}{3} = 4$. This also gives us the points (0,6) and(4, 0).

Solving of Convex Feasible Region

Plotting the above points on a graph, we have figure 4.1

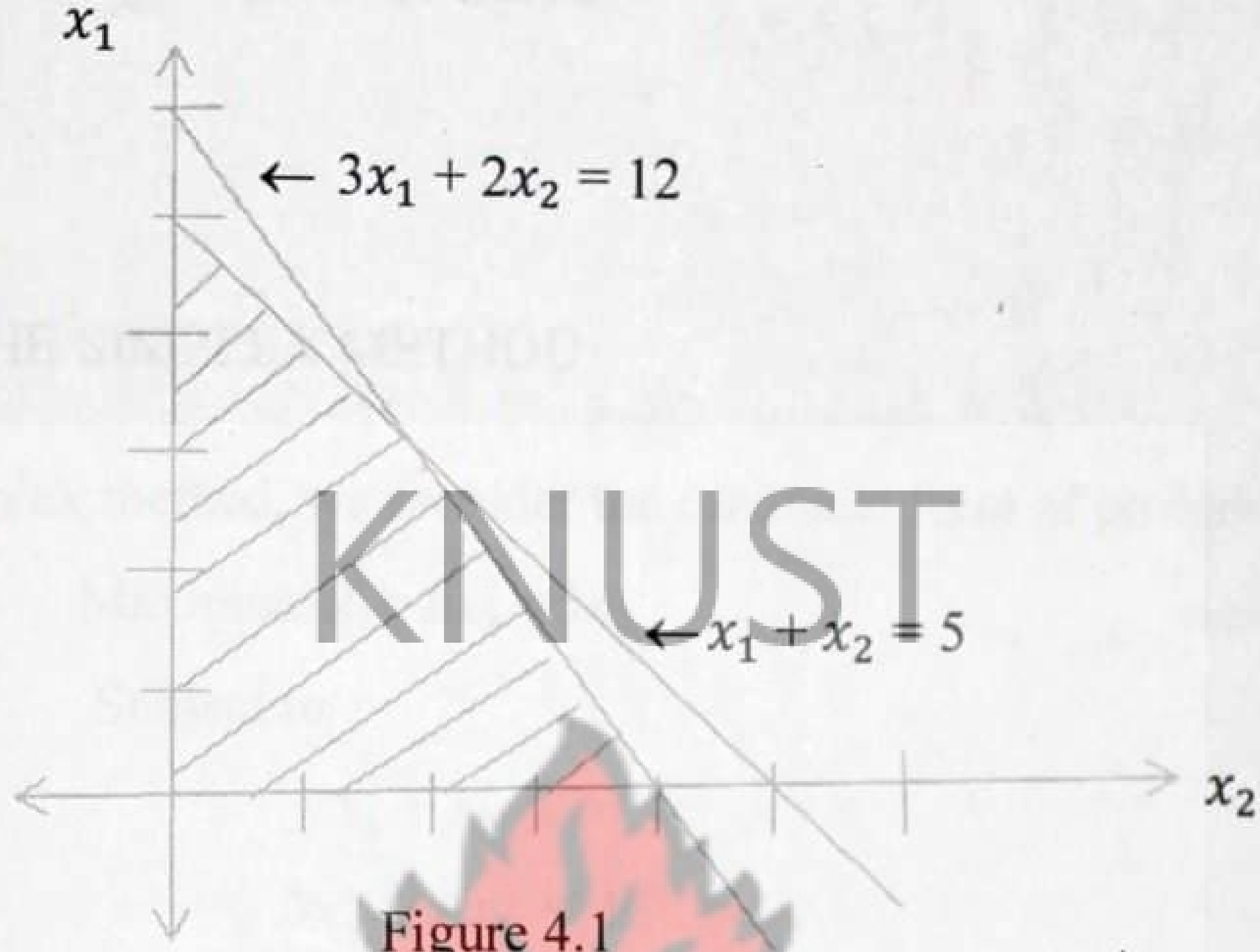


Figure 4.1

The two lines $3x_1 + 2x_2 = 12$ and $x_1 + x_2 = 5$ are meeting at the point (2, 3)

The feasible region is the shaded portion. The points (0,0), (0,5), (4,0) and (2,3) are the basic points while (5,0) and (0,6) are the nonbasic points.

$$Z = 6x_1 + 5x_2$$

For (0, 0) $\quad Z = 6(0) + 5(0)$

$\quad Z = 0$

For (4, 0) $\quad Z = 6(4) + 5(0)$

$\quad Z = 20$

For (0, 5) $\quad Z = 6(0) + 5(5)$

$\quad Z = 25$

For (2, 3) $\quad Z = 6(2) + 5(3)$

$\quad Z = 12 + 15$

$\quad Z = 27$

Therefore the feasible objective values are 0, 20, 25 and 27. The basic feasible values are 0, 20, and 25 while 27 is both basic and objective value optimal solution and it occurred at the solution$x_1 = 2$ and $x_2 = 3$. This means the company must manufacture 2 units of soft drink A and 3 units of soft drink B in order to get a profit of GHC27.

4.2: ILLUSTRATION OFTHE SIMPLEX METHOD

As an illustration of the simplex method, we consider the canonical form of problem P_1.

$$\text{Maximize } Z = 6x_1 + 5x_2$$

Subject to

$$x_1 + x_2 \leq 5$$

$$3x_1 + 2x_2 \leq 12$$

$$x_1, x_2 \geq 0.$$

We introduce slacks to convert the canonical form of example 4.1 to standard form (Kasana and Kumar, 2004). Let s_1 and s_2 be the slacks. The model thus becomes;

$$\text{Maximize } Z = 6x_1 + 5x_2 + 0s_1 + 0s_2$$

Subject to

$$x_1 + x_2 + s_1 = 5$$

$$3x_1 + 2x_2 + s_2 = 12$$

$$x_1, x_2, s_1, s_2 \geq 0$$

Table 4.2: The initial simplex tableau

	c_j	6	5	0	0		
c_B	B_v	x_1	x_2	s_1	s_2	RHS	Ratio
0	s_1	1	1	0	0	5	5
0	s_2	(3)	2	0	1	12	4 $\rightarrow$
	Z_j	0	0	0	0	0	
	c_j - Z_j	6$\uparrow$	5	0	0		

From Table 4.2, the basic variables are s_1 and s_2 while the nonbasic variables are $x_1$and x_2. The most positive (c_j-Z_j) value is 6. Therefore the third column is the pivot column. Dividing the elements in the right hand side by their respective elements in the pivot column and comparing the ratios, it is observed that the smaller ratio is 4. Therefore the row containing the smaller ratio 4 (the fifth row) is the pivot row and 3 is the pivot element. This means x_1 is entering basis and s_2 leaving basis. [The pivot element 3 should be reduced to 1 and the element above the pivot element should also be reduced to zero. By performing Gauss Jordan elimination row operations, that is dividing the fifth row by 3, the pivot element is reduced one. Also by multiplying the fourth row by – 1 and adding the result to the row obtained after the pivot row has been divided by the pivot element 3], table 4.3 below is obtained as the second simplex tableau of Table 4.3

Table 4.3: The second simplex tableau

	C_j	6	5	0	0		
c_B	B_v	x_1	x_2	s_1	s_2	RHS	Ratio
0	s_1	0	$(\frac{1}{3})$	1	$-\frac{1}{3}$	1	3 →
6	x_1	1	$\frac{2}{3}$	0	$\frac{1}{3}$	4	6
	Z_j	6	4	0	2	24	
	c_j - Z_j	0	1 ↑	0	-2		

From Table 4.3 the variables in basis are s_1 and x_1 while the nonbasic variaables are x_2 and s_2. The most positive c_j - Z_j is 1. Comparing the two ratios 3 and 4, we observed that 3 is less than 4. This means x_2 is entering basis while s_1 is leaving basis. The pivot column is the fourth column and the pivot row is the third row. The value at where the pivot column and the pivot row are meeting is $\frac{1}{3}$. Therefore the pivot element is $\frac{1}{3}$. [The pivot element $\frac{1}{3}$ should be change to 1 by multiplying the third row by 3. The value below the pivot element (i.e., $\frac{2}{3}$) is changed to zero by performing Gauss- Jordan row operations. We multiply the elements in the fourth row by -$\frac{3}{2}$and add the results to the new pivot row to change the value below the pivot element to zero] to obtain the Table 4.4 below.

Table 4.4: The optimal tableau

	C_j	6	5	0	0		
c_B	B_v	x_1	x_2	s_1	s_2	RHS	Ratio
5	x_2	0	1	3	-1	3	3
6	x_1	1	0	-2	1	2	2
	Z_j	6	5	3	1	27	
	C_j - Z_j	0	0	-3	-1		

From Table 4.4, since all the c_j - Z_j values are zeros and negatives, we stop. The values of x_2 and x_1 are 3 and 2 respectively and the maximum profit is 27. This means the company should manufacture 4 units of soft drink A and 3 units of soft drink B to obtain a maximum profit of GHC27.

4.3: ILLUSTRATION OF THE REVISED SIMPLEX METHOD

OPTIMAL LOAN ALLOCATION FOR EXPRESS SAVINGS AND LOANS COMPANY LIMITED (A REAL LIFE PROBLEM)

Emample 4.2: Express Savings and Loans Company Limited wants to formulate a loan policy which would enable them disburse a total amount of GHC 200,000 as loans to its customers. The table below gives the types of loans, interest rates charge and the probability of bad debts of ESLC Las estimated in the 2012 financial year.

Table 4.5: Types of loan, interest rates and probability of bad debt of ESLCL

Types of loans	Interest rate (r_i)	Probability of bad debts (P_i)
personal loan: x_1	0.30	0.04
Production loan: x_2	0.32	0.25
Susu loan: x_3	0.29	0.02
Business loan: x_4	0.40	0.10
Education: x_5	0.35	0.15

The company wants the money for the loan to be disbursed in the following manner:

1. At most 50% of the total amount should be allocated to Susu and Production loans.
2. The sum of business and educational must not exceed 20% of the total amount.
3. Susu loan must be at least 40% of personal, Business, and small Educational loans to ensure maximum profit.
4. Production loans must not exceed 6% of the total amount.
5. The sum of production and small Education loans should be at most 40% of the total amount.

6. The sum ofSusu and Business loans must not exceed 10%.

7. The total ratio of bad debt must not exceed 6% on all types of loans.

FORMULATION OF LINEAR PROGRAMMING MODEL

OBJECTIVE FUNCTION

The objective is to maximize the net return comprising the difference between revenue from interest and money which would be lost as a result of bad debts for each amount of money given as loan.

By letting

x_1 = amount given as Personal loans

x_2 = amount given as Production loans

x_3 = amount given as Susu loans

x_4= amount given as Business loans

x_5 = amount given as educational loans,

the following table is obtained on amount of loan, amount of bad debts and amount contributing to profit.

Table 4.6: Amount of loans, bad debt and amount contributing to profit of ESLCL

Amount of loan (x_i)	Amount of bad debts ($P_i x_i$)	Amount contributing to profit ($1 - P_i)x_i$
x_1	$0.04x_1$	$0.96x_1$
x_2	$0.25x_2$	$0.75x_2$
x_3	$0.02x_3$	$0.98x_3$
x_4	$0.10x_4$	$0.90x_4$
x_5	$0.15x_5$	$0.85x_5$

Total profit on loans is given as:

$Z = r_1(1 - P_1)x_1 + r_2(1 - P_2)x_2 + r_3(1 - P_3)x_3 + r_4(1 - P_4)x_4 + r_5(1 - P_5)x_5.$

The objective function is therefore given as

Maximize $Z = 0.3(0.96x_1) + 0.32(0.75x_2) + 0.29(0.98x_3) + 0.40(0.90x_4) + 0.35(0.85x_5)$

$Z = 0.288x_1 + 0.240x_2 + 0.284x_3 + 0.360x_4 + 0.298x_5$

CONSTRAINTS

Total amount is GHC 200,000

1. Limit on total amount (x_1, x_2, x_3, x_4, x_5)

   $x_1 + x_2 + x_3 + x_4 + x_5 \leq 200,000$

2. Limit on Business(x_4) and education (x_5):

$x_4 + x_5 \leq 0.2(200000)$

$x_4 + x_5 \leq 40000$

3. Limit on Production (x_2) and Education (x_5) loans

$x_2 + x_5 \leq 0.40(200{,}000)$

$x_2 + x_5 \leq 80{,}000$

4. Limit on Susu (x_3) and Production (x_4) loans:

$x_2 + x_3 \leq 0.5(200000)$

$x_2 + x_3 \leq 100000$

5. Limit on Susu (x_3), Personal (x_1), Business (x_4) and Education(x_5):

$x_3 \geq 0.4(x_1 + x_4 + x_5)$

$0.4x_1 - x_3 + 0.4x_4 + 0.4x_5 = 0$

6. Limit on Production loans (x_2):

$x_2 \leq 1200$

7. Limit on Susu (x_3) and business (x_4) loans:

$x_3 + x_4 \leq 0.1(200000)$

$x_3 + x_4 \leq 20000$

8. Limit on bad debt:

$$\frac{0.04x_1 + 0.25x_2 + 0.02x_3 + 0.10x_4 + 0.15x_5}{x_1 + x_2 + x_3 + x_4 + x_5} \leq 0.06$$

$-0.02x_1 + 0.19x_2 - 0.04x_3 + 0.04x_4 + 0.09x_5 \leq 0$

9. Non- negativity constraints: $x_1, x_2, x_3, x_4, x_5 \geq 0$.

RESULTING LINEAR PROGRAMMING MODEL

Maximize Z = $0.288x_1 + 0.240x_2 + 0.280x_3 + 0.360x_4 + 0.340x_5$

Subject to

$x_1 + x_2 + x_3 + x_4 + x_5 \leq 200{,}000$

$x_4 + x_5 \leq 40000$

$x_2 + x_5 \leq 80{,}000$

$x_2 + x_3 \leq 100000$

$0.4x_1 - x_3 + 0.4x_4 + 0.4x_5 \leq 0$

$x_2 \leq 1200$

$x_3 + x_4 \leq 20000$

$-0.02x_1 + 0.19x_2 - 0.04x_3 + 0.04x_4 + 0.09x_5 \leq 0$

$x_1, x_2, x_3, x_4, x_5 \geq 0$.

RESULTS

The initial iteration of the solution of the revised simplex method is given in appendix.

Table 4.7 gives the variables (column one), the optimal value of the variables(column two) and the status of the variables (column three). The variables show that funds for loans should be allocated to personal, business and education with the amounts shown on the table.

Optimal solution after four (4) iterations

Table:4.7: Optimal value (Z) = 611200

Variables	Optimal value	Status
x_1 : Personal loans	140000.00	Basic
x_2 : Production loans	0.00	Nonbasic
x_3 : Susu loans	0.00	Nonbasic
x_4 : Business loans	20000.00	Basic
x_5 : Educational loans	40000.00	Basic

Table 4.8 shows the variables (column one),objective coefficient (column two) and the objective value contribution (column three).

Table 4.8: Variables, objective coefficient and objective value contribution

Variable	Objective coefficient	Objective value contribution
x_1 : Personal loans	0.288	40320.00
x_2 : Production loans	0.240	0.00
x_3 : Susu loans	0.284	0.00
x_4 : Business loans	0.360	7200.00
x_5 : Educational loans	0.298	11920.00
Total		59440

DISCUSSION

The optimal solution is x_1 = 140000, x_2 = 0, x_3 = 0, x_4 = 400, x_5 = 40000 and the objective function value, Z is 611200. This shows that Express Savings and Loans Company Limited (ESLCL) should allocate GHC 140000 to personal loans, GHC 20000 to business loans and GHC40000 to educational loans. The company should not allocate any fund to production loans and Susu loans

4.4: ILLUSTRATION OF THE DUAL SIMPLEX METHOD

In this section, we use problem P_1 to illustrate the application of the Dual Simplex method.

That is

$$\text{Maximize } z = 6x_1 + 5x_2$$

Subject to

$$x_1 + x_2 + s_1 = 5$$

$$3x_1 + 2x_2 + s_2 = 12$$

$$x_1,\ x_2,\ s_1,\ s_2 \geq 0.$$

Writing the problem in the appropriate Dual Simplex format, we have

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

Subject to

$$-AX = -b, \quad X \geq 0$$

This implies $\quad$ Maximize $z = 6x_1 + 5x_2$

Subject to

$$-x_1 - x_2 - s_1 = -5$$

$$-3x_1 - 2x_2 - s_2 = -12$$

$$x_1,\ x_2,\ s_1,\ s_2 \geq 0$$

Organization structure of tables

In the following tables, the elements in the bracket are the pivot elements. An arrow pointing towards outside a row means the element in that row is the pivot element and that row is the pivot row. An arrow pointing towards inside a column means the variable in that column is the entering variable and that column is the pivot column.

Table 4.9: initial tableau

	c_j	6	5	0	0	
CB_V	B_V	x_1	x_2	s_1	s_2	solution
0	s_1	-1	-1	-1	0	-5
0	s_2	(-3)	-2	0	-1	-12 →
	Z_j	0	0	0	0	0
	Z_j - c_j	-6 ↑	-5	0	0	

From the above Tableau, the more negative entry in the solution column is - 12 and that corresponds to s_2, hence s_2 is the leaving variable. Next, we look for negative entries in s_2–row and compute

$$\text{Min }(|-6/-3|, |-5/-2|) = 2.$$

This corresponds to x_1 , hence x_1 is the entering variable. Thus, s_2 leaves and x_1 enters and -3 is the pivot element. [Now, we obtain the next tableau exactly like the simplex method by performing Gauss Jordan row operations. That is dividing the fourth row by -3 to convert the pivot element to 1 and adding the result to the third row] Table 4.10 is obtained.

Table 4.10: The second tableau

	c_j	6	5	0	0	
CB_V	B_V	x_1	x_2	s_1	s_2	solution
0	s_1	0	-1/3	- 1	1/3	-1 →
6	x_1	1	2/3	0	$\frac{1}{3}$	4
	Z_j	6	4	0	2	24
	$z_j - c_j$	0	-1 ↑	0	2	

From the Table above, since -1 is the negative entry in the solution column and corresponds to s_1, it means s_1 is the leaving variable. We look for all negative entries in s_1-row and compute

$$\text{Min}\,(|-1/-1/3|) = 3.$$

This corresponds to x_2 therefore $x_2$is the entering variable. Thus,s_1 leaves and $x_2$enters and the pivot element is $-1/3$. [We multiply the s_1-row by -3 to convert the pivot element to 1, multiply the result by $-2/3$ and to the fourth row] to obtain the Table 4.11 below.

Table 4.11: The optimal tableau

	c_j	6	5	0	0	
C_{B_V}	B_V	x_1	x_2	s_1	s_2	solution
5	x_2	0	1	3	-1	3
6	x_1	1	0	-2	1	2
	z_j	6	5	3	1	27
	$z_j - c_j$	0	0	3	1	

Since all the solution columnvalues are nonnegative, the table above gives the optimal solution. That is

$x_1 = 2,\ x_2 = 3,$ maximum value is 27.

4.6: ILLUSTRATION OF KARMARKAR'S ALGORITHM

As an illustration ofKarmarkar's algorithm in solving linear programming problems, let us consider the following minimization examples since Karmarkar's algorithm is very useful in solving minimization problems.

Example 4.3:(Eiselt and Sandbrom, 2007) An individual wants to plan his diet, which consists of only pork and beans. The two nutrients that are considered in this diet are protein and carbohydrates. Details concerning the nutritional contents of the food stuffs, their prices and the required nutritional contents are shown in the table below.

Table 4.12: Nutritional content of food stuffs and their prices

	Pork	Beans	Nutrients needed (at least)
Protein (x_1)	2	3	8
Carbohydrate (x_2)	5	2	12
Unit Selling Price	GHC 3	GHC 4	

By denoting with x_1 and x_2 the respective number of servings of pork and beans respectively, a cost-minimizing diet problem can be formulated as a linear programming problem as follows;

Minimize $Z = 3x_1 + 4x_2$

Subject to

$$2x_1 + 3x_2 \geq 8$$

$$5x_1 + 2x_2 \geq 12$$

$$x_1, x_2 \geq 0$$

Since both constraints contain greater than or equal to signs (≥), it means resources have been over used. We must therefore subtract the excess resources by introducing surplus. Then model then becomes:

$$\text{Minimize } Z = 3x_1 + 4x_2$$

Subject to

$$2x_1 + 3x_2 - s_1 = 8$$

$$5x_1 + 2x_2 - s_2 = 12$$

$$x_1, x_2, s_1, s_2 \geq 0$$

We convert the model into the standard form required by the Karmarkar algorithm so that the assumption $Ae = 0$ will be satisfied. That is

$$\text{Minimize } f = (Q + 1)C^T Y + My_{n+3} \quad \ldots 4.3$$

Subject to

$$AY - by_{n+2} - [Ae - b]y_{n+3} = 0 \ldots \quad 4.4$$

$$e^T Y + y_{n+1} - Qy_{n+2} - (n + 1) - Q)y_{n+3} = 0 \ldots \quad 4.5$$

$$e^T Y + y_{n+1} + y_{n+2} + y_{n+3} = 1 \ldots \quad 4.6$$

$$Y \geq 0, \; y_{n+1}, \; y_{n+2}, y_{n+3} \geq 0.$$

Where $Q = 2^L$, L is the number of variables in the linear programming model above, n is the number of columns in matrix A shown below. This implies $n = 4$.

From the linear programming problem,

$A = \begin{bmatrix} 2 & 3 & -1 & 0 \\ 5 & 2 & 0 & -1 \end{bmatrix}$, $Q = 2^L$, where $L = 4$. This implies $Q = 16$, $C = [2\ 3\ 0\ 0]^T$

$$\text{Min } f = (16 + 1)[2\ 3\ 0\ 0][y_1 y_2 y_3 y_4]^T + My_7$$

$$\text{Min } f = (17)(2y_1 + 3y_2) + My_7$$

From 4.4,

$$\begin{bmatrix} 2 & 3 & -1 & 0 \\ 5 & 2 & 0 & -1 \end{bmatrix}\begin{bmatrix} y1 \\ y2 \\ y3 \\ y4 \end{bmatrix} - \begin{bmatrix} 8 \\ 12 \end{bmatrix} y_6 - \left[\begin{pmatrix} 2 & 3 & -1 & 0 \\ 5 & 2 & 0 & -1 \end{pmatrix}\begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} - \begin{pmatrix} 8 \\ 12 \end{pmatrix}\right] y_7 = 0$$

$$\begin{bmatrix} 2y_1 + 3y_2 - y_3 \\ 5y_1 + 2y_2 - y_4 \end{bmatrix} - \begin{bmatrix} 8y_6 \\ 12y_6 \end{bmatrix} - \left[\begin{pmatrix} 2+3-1 \\ 5+2-1 \end{pmatrix} - \begin{pmatrix} 8 \\ 12 \end{pmatrix}\right] y_7 = 0$$

$$2y_1 + 3y_2 - y_3 - 8y_6 + 4y_7 = 0$$

$$5y_1 + 2y_2 - y_4 - 12y_6 + 6y_7 = 0$$

From 4.5, we have

$$[1\ 1\ 1\ 1]\begin{bmatrix} y1 \\ y2 \\ y3 \\ y4 \end{bmatrix} + y_5 - 16y_6 - (5 - 16)y_7 = 0$$

$$y_1 + y_2 + y_3 + y_4 + y_5 - 16y_6 + 11y_7 = 0.$$

From 4.6, $[1\ 1\ 1\ 1]\begin{bmatrix} y1 \\ y2 \\ y3 \\ y4 \end{bmatrix} + y_5 + y_6 + y_7 = 1$

$$y_1 + y_2 + y_3 + y_4 + y_5 + y_6 + y_7 = 1.$$

Therefore the model becomes

$$\text{Minimize f} = 34y_1 + 51y_2 + My_7$$

Subject to

$$2y_1 + 3y_2 - y_3 - 8y_6 + 4y_7 = 0$$

$$5y_1 + 2y_2 - y_4 - 12y_6 + 6y_7 = 0$$

$$y_1 + y_2 + y_3 + y_4 + y_5 - 16y_6 + 11y_7 = 0$$

$$y_1 + y_2 + y_3 + y_4 + y_5 + y_6 + y_7 = 1$$

$$y_1, y_2, y_3, y_4, y_5, y_6, y_7 \geq 0$$

which satisfies the standard form of Karmarkar's algorithm.

To solve the model above, we let M = 10000000000. n = 7,

Matrix $A = \begin{bmatrix} 2 & 3 & -1 & 0 & 0 & -8 & 4 \\ 5 & 2 & 0 & -1 & 0 & -12 & 6 \\ 1 & 1 & 1 & 1 & 1 & -16 & 11 \end{bmatrix}$, C = [34 51 0 0 0 0 10000000000]

Step 1: we set k = 0, this means $x^0 = (1/7, 1/7, 1/7, 1/7, 1/7, 1/7, 1/7)^T$

Step 2: $F(x^0) = 34(1/7) + 51(1/7) + 10000000000(1/7)$

$= 4.85714 + 7.28571 + 1.42857 \times 10^9$

$= 1.42857\times 01^{9} > 0.075.$

Step 3:

$$D_0 = \begin{bmatrix} 1/7 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1/7 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1/7 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1/7 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1/7 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1/7 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1/7 \end{bmatrix}$$

$$\mathrm{A}D_0 = \begin{bmatrix} 2 & 3 & -1 & 0 & 0 & -8 & 4 \\ 5 & 2 & 0 & -1 & 0 & -12 & 6 \\ 1 & 1 & 1 & 1 & 1 & -16 & 11 \end{bmatrix} \begin{bmatrix} 1/7 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1/7 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1/7 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1/7 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1/7 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1/7 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1/7 \end{bmatrix}$$

$$= \begin{bmatrix} 2/7 & 3/7 & -1/7 & 0 & 0 & -8/7 & 4/7 \\ 5/7 & 2/7 & 0 & -1/7 & 0 & -12/7 & 6/7 \\ 1/7 & 1/7 & 1/7 & 1/7 & 1/7 & -16/7 & 11/7 \end{bmatrix}$$

$$B_0 = \begin{bmatrix} 2/7 & 3/7 & -1/7 & 0 & 0 & -8/7 & 4/7 \\ 5/7 & 2/7 & 0 & -1/7 & 0 & -12/7 & 6/7 \\ 1/7 & 1/7 & 1/7 & 1/7 & 1/7 & -16/7 & 11/7 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

$$
B_0^T = \begin{bmatrix} 2/7 & 5/7 & 1/7 & 1 \\ 3/7 & 2/7 & 1/7 & 1 \\ -1/7 & 0 & 1/7 & 1 \\ 0 & -1/7 & 1/7 & 1 \\ 0 & 0 & 1/7 & 1 \\ -8/7 & -12/7 & -16/7 & 1 \\ 4/7 & 6/7 & 11/7 & 1 \end{bmatrix}, \quad B_0 B_0^T = \begin{bmatrix} 94/49 & 136/49 & 176/49 & 0 \\ 136/49 & 30/7 & 264/49 & 0 \\ 176/49 & 264/49 & 282/40 & 0 \\ 0 & 0 & 0 & 7 \end{bmatrix}
$$

$$
(B_0 B_0^T)^{-1} = \begin{bmatrix} 465255/38546 & -42238/19273 & -86240/19273 & 0 \\ -422238/19273 & 733187/115638 & -215600/57819 & 0 \\ -86240/19273 & -215600/57819 & 304780/57819 & 0 \\ 0 & 0 & 0 & 1/7 \end{bmatrix}.
$$

$$
D_0 C^T = \begin{bmatrix} 34/7 \\ 51/7 \\ 0 \\ 0 \\ 0 \\ 0 \\ 10000000000/7 \end{bmatrix}, \quad I = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}
$$

$B_0^T(B_0B_0^T)^{-1}B_0 =$

$$
\begin{bmatrix}
\frac{-2972850}{1888754} & \frac{-4582541}{944377} & \frac{1436866}{944377} & \frac{-214273}{269822} & \frac{-42187}{134911} & \frac{15420073}{944377} & \frac{-8798330}{944377} \\
\frac{-402541}{944377} & \frac{-1545639}{1888754} & \frac{133769}{1888754} & \frac{-30252}{134911} & \frac{-23707}{134911} & \frac{6261559}{944377} & \frac{-3830993}{94437} \\
\frac{-66162}{134911} & \frac{-198033}{269822} & \frac{549953}{809466} & \frac{50339}{134911} & \frac{138319}{404733} & \frac{15455}{404733} & \frac{320501}{404733} \\
\frac{20089}{1888754} & \frac{928236}{944377} & \frac{-27627}{944377} & \frac{143553}{269822} & \frac{44053}{134911} & \frac{-2999659}{944377} & \frac{2222576}{944377} \\
\frac{-42187}{134911} & \frac{-23707}{134911} & \frac{138319}{404733} & \frac{44053}{134911} & \frac{101359}{404733} & \frac{26459}{404733} & \frac{204119}{404733} \\
\frac{9340073}{944377} & \frac{13861559}{944377} & \frac{-13571815}{2833131} & \frac{5763}{134911} & \frac{26459}{404733} & \frac{-106909451}{2833131} & \frac{53403265}{2833131} \\
\frac{-48357968240}{9634628597} & \frac{-7369651785}{9634628597} & \frac{9083507}{2833131} & \frac{727133228}{1376375511} & \frac{204119}{404733} & \frac{156674666855}{9634628597} & \frac{-6582479946}{9634628597}
\end{bmatrix}
$$

$$
\mathrm{I} - B_0^T(B_0B_0^T)^{-1} B_0 = \begin{bmatrix}
1.72 & 4.85 & -1.52 & 0.79 & 0.31 & -16.33 & 9.32 \\
0.43 & 0.96 & -0.07 & 0.22 & 0.18 & -6.63 & 40.57 \\
0.49 & 0.73 & -0.54 & -0.37 & -0.35 & -0.04 & -0.79 \\
-0.01 & -0.98 & 0.03 & -0.39 & -0.33 & 3.00 & -2.35 \\
0.31 & 0.18 & -0.34 & -0.33 & -0.11 & -0.07 & -204119404733 \\
-9.90 & -14.68 & 47.90 & -0.04 & -0.07 & 37.88 & -18.85 \\
5.12 & 7.65 & -3.21 & -0.05 & -0.50 & -16.26 & 0.83
\end{bmatrix}
$$

$$d_0 = -[I - B_0^T (B_0 B_0^T)^{-1} B_0] D_0 C^T = \begin{bmatrix} \frac{-8998330028835015}{6610639} \\ \frac{-10336861934078397884809}{178368261498} \\ \frac{6410019956203903}{5666262} \\ \frac{2222257600476815 49}{6610639} \\ \frac{2753795301193376299973565 85}{94437} \\ \frac{5340326530735059 73}{19831917} \\ \frac{-795885599966933171123}{674424001419} \end{bmatrix}$$

Example 4.4: (Kasana and Kumar) Find the solution of the following linear programming problems using the Karmarkar's algorithm.

Minimize $f = 2x_1 + x_2 - x_3$

Subject to $x_2 - x_3 = 0$

$x_1 + x_2 + x_3 = 1$

$x_1, x_2, x_3 \geq 0$

From the problem above, A = [0, 1, -1], C = (2, 1, -1), n = 3, e = (1, 1, 1).

Ae = 0. Therefore, the problem is in Karmarkar's standard form.

Step 1: If k = 0, $x^o = (1/3, 1/3, 1/3)^T$

Using tolerance of 0.075, we check step 2.

step 2: $f(x^o) = 2(1/3) + 1/3 - 1/3 = 2/3 > 0.075$, we check step 3.

Step 3: we define

$$D_0 = \begin{bmatrix} 1/3 & 0 & 0 \\ 0 & 1/3 & 0 \\ 0 & 0 & 1/3 \end{bmatrix}.$$

$$AD_0 = [0\ 1\ -1]\begin{bmatrix}1/3 & 0 & 0\\ 0 & 1/3 & 0\\ 0 & 0 & 1/3\end{bmatrix} = [0, 1/3, -1/3],\ B_0 = \begin{bmatrix}0 & 1/3 & -1/3\\ 1 & 1 & 1\end{bmatrix}$$

$$B_0B_0^T = \begin{bmatrix}0 & 1/3 & -1/3\\ 1 & 1 & 1\end{bmatrix}\begin{bmatrix}0 & 1\\ 1/3 & 1\\ -1/3 & 1\end{bmatrix} = \begin{bmatrix}2/9 & 0\\ 0 & 3\end{bmatrix},\ (B_0B_0^T)^{-1} = \begin{bmatrix}9/2 & 0\\ 0 & 1/3\end{bmatrix}.$$

$$D_0\,C = \begin{bmatrix}1/3 & 0 & 0\\ 0 & 1/3 & 0\\ 0 & 0 & 1/3\end{bmatrix}\begin{bmatrix}2\\ 1\\ -1\end{bmatrix} = \begin{bmatrix}2/3\\ 1/3\\ -1/3\end{bmatrix}$$

$$I - B_0^T(B_0B_0^T)^T B_0 = \begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & 1\end{bmatrix} - \begin{bmatrix}0 & 1\\ 1/3 & 1\\ -1/3 & 1\end{bmatrix}\begin{bmatrix}9/2 & 0\\ 0 & 1/3\end{bmatrix} = \begin{bmatrix}0 & 1/3 & -1/3\\ 1 & 1 & 1\end{bmatrix}$$

$$= \begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & 1\end{bmatrix} - \begin{bmatrix}1/3 & 1/3 & 1/3\\ 1/3 & 5/6 & -1/6\\ 1/3 & -1/6 & 5/6\end{bmatrix} = \begin{bmatrix}2/3 & -1/3 & -1/3\\ -1/3 & 1/6 & 1/6\\ -1/3 & 1/6 & 1/6\end{bmatrix}.$$

The moving direction is given by

$$d_0 = -[\,I - B_0\,(B_0B_0^T)T\,B_0]\,D_0C = = \begin{bmatrix}-2/3 & 1/3 & 1/3\\ 1/3 & -1/6 & -1/6\\ 1/3 & -1/6 & -1/6\end{bmatrix} = \begin{bmatrix}2/3\\ 1/3\\ -1/3\end{bmatrix} = \begin{bmatrix}-4/3\\ 2/9\\ 2/9\end{bmatrix}$$

The length of the vector d_0 is given by

$$\| d_0 \| = \sqrt{(4/9)^2 + (2/9)^2 + (2/9)^2} = 2\sqrt{6/9},\quad \alpha = 1/\sqrt{n\,(n-1)} = 1/\sqrt{6}$$

$$y^0 = \begin{bmatrix}1/3\\1/3\\1/3\end{bmatrix} + (1/3)\,(^1/_{\sqrt{6}})\ ^9/_{2\sqrt{6}}\begin{bmatrix}-4/9\\2/9\\2/9\end{bmatrix} = \begin{bmatrix}1/3\\1/3\\1/3\end{bmatrix} + \begin{bmatrix}-1/9\\1/18\\1/18\end{bmatrix} = \begin{bmatrix}2/9\\7/18\\7/18\end{bmatrix}$$

Hence, the new feasible solution is given by

$$x^1 = \frac{D_0\, y^0}{e^T D_0\, y^0} = \frac{(\frac{2}{27}, \frac{7}{54}, \frac{7}{54})}{1/3} = (2/9,\ \ 7/18,\ 1/18)^T$$

$f(x^1) = 2(2/9) + 7/18 - 7/18 = 4/9 > 0.075.$

Certainly, there is improvement over the first starting estimation but the tolerance limit is not satisfied.

If $\alpha = {}^2/_6$, then

$$y^0 = \begin{bmatrix}1/3\\1/3\\1/3\end{bmatrix} + (1/3)\ (^2/_{\sqrt{6}})(^9/_{\sqrt{6}}) \begin{bmatrix}-4/9\\2/9\\2/9\end{bmatrix} = \begin{bmatrix}1/9\\4/9\\4/9\end{bmatrix}$$

$$D^0 y^0 = \begin{bmatrix}1/3 & 0 & 0\\0 & 1/3 & 0\\0 & 0 & 1/3\end{bmatrix}\begin{bmatrix}1/9\\4/9\\4/9\end{bmatrix} = \begin{bmatrix}1/27\\4/27\\4/27\end{bmatrix},\quad e^T D_0 y^0 = [1\ 1\ 1]\begin{bmatrix}1/27\\4/27\\4/27\end{bmatrix} = 1$$

$$x^1 = (1/27, 4/27, 4/27)^T$$

$F(x^1) = 2(1/27) + 4/27 - 4/27 = 2/27 < 0.075.$

Thus, $x^1 = (1/27,\ 4/27,\ 4/27)$ is the optimal solution.

4.7: ILLUSTRATION OF THE DECOMPOSITION PRINCIPLE

To illustrate the decomposition principle, we consider the following example

Example 4.5: Write the following linear programming problem amendable to the Decomposition Principle and find its optimal solution.

$$\text{Maximize } f = x_1 + 2x_2 + 2x_3 + 3x_4$$

$$\text{Subject to } x_1 + x_2 + x_3 + x_4 \leq 100$$

$$x_1 + x_3 \leq 50$$

$$x_1 + x_2 \leq 60$$

$$x_1 - 2x_2 \leq 0$$

$$-2x_3 + x_4 \leq 0$$

$$x_i \geq 0, \; i = 1, 2, 3, 4.$$

The above problem can be written in the form of the Decomposition Principle as;

$$\text{Maximize } f(X) = C_1^T X_1 + C_2^T X_2$$

$$\text{Subject to } \quad A_1 X_1 + A_2 X_2 = D_0$$

$$B_1 X_1 = D_1$$

$$B_2 X_2 = D_2$$

$$X_1, X_2 \geq 0,$$

where

$A_1 = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}$; $A_2 = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}$; $B_1 = \begin{bmatrix} 1 & 1 \\ 1 & -2 \end{bmatrix}$; $B_2 = [-2 \ 1]$; $C_1 = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$; $C_2 = \begin{bmatrix} 2 \\ 3 \end{bmatrix}$; $D_o = \begin{bmatrix} 100 \\ 50 \end{bmatrix}$; $D_1 = \begin{bmatrix} 60 \\ 0 \end{bmatrix}$; $D_2 = [0]$; $X_1 = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$; $X_2 = \begin{bmatrix} x_3 \\ x_4 \end{bmatrix}$

Iteration 1. We consider the subsidiary constraints sets $B_j X_j = D_j$, $j = 1, 2$.

When j = 1, $B_1 X_1 = D_1$, but $B_1 = \begin{bmatrix} 1 & 1 \\ 1 & -2 \end{bmatrix}$, $X_1 = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$, $D_1 = \begin{bmatrix} 60 \\ 0 \end{bmatrix}$.

$\begin{bmatrix} 1 & 1 \\ 1 & -2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 60 \\ 0 \end{bmatrix}$. Therefore $x_1 = 40$ and $x_2 = 20$.

The vertices of

Feasible region 1, are $X_1^{(1)} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$; $X_2^{(1)} = \begin{bmatrix} 0 \\ 60 \end{bmatrix}$; $X_3^{(1)} = \begin{bmatrix} 40 \\ 20 \end{bmatrix}$;

In the same way the vertices of

Feasible region 2, are $X_1^{(2)} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$; $X_2^{(2)} = \begin{bmatrix} 100 \\ 0 \end{bmatrix}$; $X_3^{(2)} = \begin{bmatrix} 100 \\ 200 \end{bmatrix}$.

The sketches for feasible region 1 and feasible region 2 are given below.

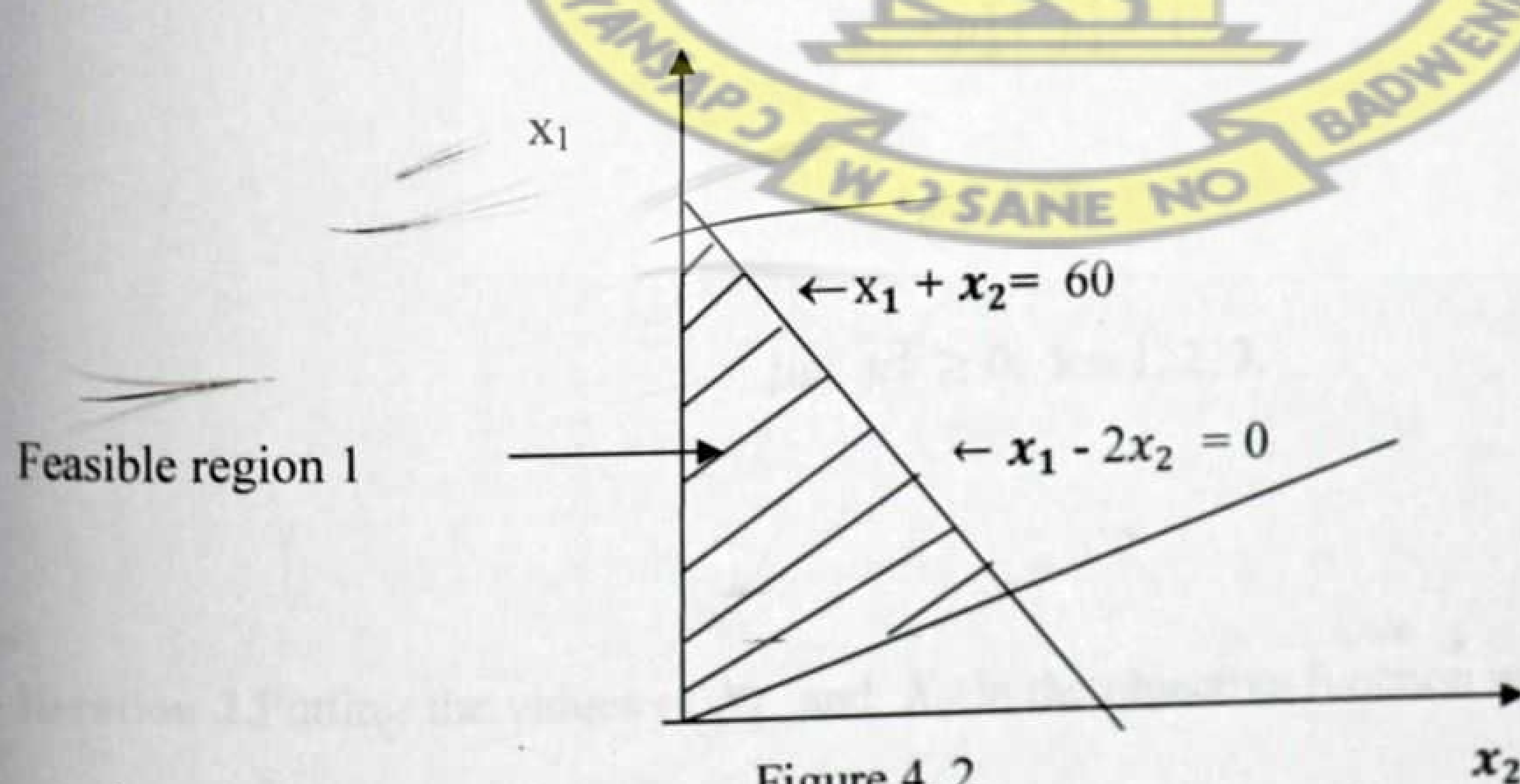


Figure 4. 2

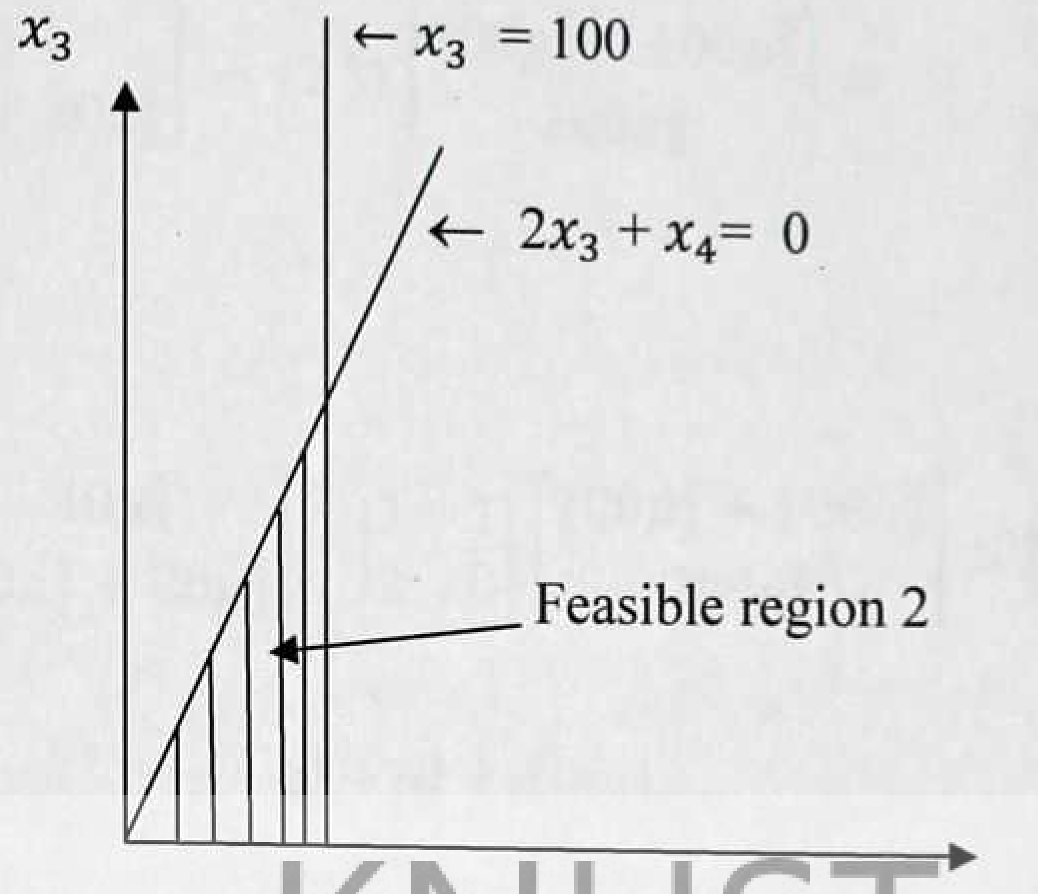


Figure 4.3

Hence, any point X_1 belonging to feasible region 1 and X_2 belonging to feasible region 2 can be given as

$$X_1 = \mu_1^1 \begin{bmatrix} 0 \\ 0 \end{bmatrix} + \mu_2^1 \begin{bmatrix} 0 \\ 60 \end{bmatrix} + \mu_3^1 \begin{bmatrix} 40 \\ 20 \end{bmatrix}$$

$$X_2 = \mu_1^2 \begin{bmatrix} 0 \\ 0 \end{bmatrix} + \mu_2^2 \begin{bmatrix} 100 \\ 200 \end{bmatrix} + \mu_3^2 \begin{bmatrix} 100 \\ 0 \end{bmatrix}$$

$$\mu_1^1 + \mu_2^1 + \mu_3^1 = 1$$

$$\mu_1^2 + \mu_2^2 + \mu_3^2 = 1$$

$$\mu_K^1,\ \mu_K^2 \geq 0,\ \text{k} = 1, 2, 3.$$

Iteration 2. Putting the values of X_1 and X_2 in the objective function and the first constraint of the problem (amendable to decomposition principle) and including restrictions μ_k^j, we have

$$\text{Maximize } f = (1\ 2)\begin{bmatrix} 40\mu_3^1 \\ 60\mu_2^1 + 20\mu_3^1 \end{bmatrix} + (2\ 3)\begin{bmatrix} 100\mu_2^2 + 100\mu_3^2 \\ 200\mu_2^2 \end{bmatrix}$$

Subject to

$$\begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}\begin{bmatrix} 40\mu_3^1 \\ 60\mu_2^1 + 20\mu_3^1 \end{bmatrix} + \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}\begin{bmatrix} 100\mu_2^2 + 100\mu_3^2 \\ 200\mu_2^2 \end{bmatrix} \leq \begin{bmatrix} 100 \\ 50 \end{bmatrix}$$

$$\mu_1^1 + \mu_2^1 + \mu_3^1 = 1$$

$$\mu_1^2 + \mu_2^2 + \mu_3^2 = 1$$

all variables ≥ 0,

or in simplified form

$$\text{maximize } f = 120\mu_2^1 + 80\,\mu_3^1 + 800\mu_2^2 + 200\mu_3^2$$

subject to

$$60\mu_2^1 + 60\,\mu_3^1 + 300\,\mu_2^2 + 100\,\mu_3^2 \leq 100$$

$$40\,\mu_3^1 + 100\,\mu_2^2 + 100\,\mu_3^2 \leq 50$$

$$\mu_1^1 + \mu_2^1 + \mu_3^1 = 1$$

$$\mu_1^2 + \mu_2^2 + \mu_3^2 = 1$$

$$\mu_k^1, \mu_k^2 \geq 0, \ k = 1, 2, 3.$$

Any standard method can be applied to solve the above linear programming problem. We use the simplex method to solve the problem.

Organization of the tables

In table 4.13below, the value in the bracket is the pivot element. An arrow pointing outside a row means the corresponding variable is the leaving variable while that row is the pivot row. An arrow pointing towards a column means the corresponding variable is the entering variable while that column is the pivot column.

Table 4.13: Initial simplex Tableau

	c_j	120	80	800	200	0	0		
CB_V	B_V	μ_2^1	μ_3^1	μ_2^2	μ_3^2	s_1	s_2	solution	Ratio
0	s_1	60	60	(300)	100	1	0	100	$\frac{1}{3}\rightarrow$
0	s_2	0	40	100	100	0	1	50	$\frac{1}{2}$
	z_j	0	0	0	0	0	0	0	
	z_j - c_j	-120	-80	-800 ↑	-200	0	0	0	

From Table 4.13, the pivot element, 300 is changed to 1 by dividing the elements in the pivot row by 300. The element below the pivot element is changed to 0 by performing Gauss Jordan row operations to obtain table 4.14.

From table4.10, s_1 = 100, s_2 = 50 and z = 0.

Table 4 .14: The optimal tableau

	C_j	120	80	800	200	0	0	
CB_V	B_V	μ_2^1	μ_3^1	μ_2^2	μ_3^2	s_1	s_2	solution
800	μ_2^2	1/5	1/5	1	1/3	1/300	0	1/3
0	s_2	-20	20	0	200/3	-1/3	1	50/3
	z_j	160	160	800	800/3	8/3	0	800/3
	z_j - c_j	40	80	0	200/3	8/3	0	

From the Table 4.14 above, since all the z_j - c_j values are positive, Table 4.14 is the optimal tableau.

The optimal solution of the above problem is $\mu_1^1 = 1, \mu_2^1 = 0, \mu_1^2 = \frac{2}{3}, \mu_2^2 = \frac{1}{3}, \mu_3^2 = 0,$ $f = \frac{800}{3}.$

Inserting these values for the expressions of X_1 and X_2, we have

$X_1 = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$. This implies $x_1 = 0,\ x_2 = 0;$

$X_2 = \begin{bmatrix} x_3 \\ x_4 \end{bmatrix} = \frac{1}{3}\begin{bmatrix} 100 \\ 200 \end{bmatrix} = \begin{bmatrix} \frac{100}{3} \\ \frac{200}{3} \end{bmatrix}$. This implies $x_3 = \frac{100}{3},\ x_4 = \frac{200}{3}$.

Thus the optimal solution is $x_1 = 0, x_2 = 0,\ x_3 = \frac{100}{3},\ x_4 = \frac{200}{3}$ and $f = \frac{800}{3}$.

4.8 ILLUSTRATION OF THE BOUNDED VARIABLE TECHNIQUE

To illustrate the Bounded Variable Techniquewe consider the example below.

Solve the linear programming problem

Example 4.6 (Kasana and Kumar)

$$\text{Maximize } z = 4x_1 + 2x_2 + 6x_3$$

$$\text{Subject to } 4x_1 - x_2 - 3x_3 \leq 9$$

$$-x_1 + x_2 + 2x_3 \geq 8$$

$$-3x_1 + x_2 + 4x_3 \leq 11$$

$$1 \leq x_1 \leq 3,\ 0 \leq x_2 \leq 5,\ 0 \leq x_3 \leq 2.$$

The two phase simplex method is convenient to apply to solve this problem.

Since the variables are bounded, the bounded variable technique is the appropriate method to use to find the optimal solution.

First, we make lower bounds in $1 \leq x_1 \leq 3$ at zero level. By replacing x_1 by $y_1 + 1$, we have $1 \leq y_1 + 1 \leq 3$, which gives us $0 \leq y_1 \leq 2$.

Also, replacing x_1 by $y_1 + 1$ and substituting slackss_1, s_3, surplus $s_2$and artificial variable R_2 into the constraints, the model shown in phase 1 is obtained.

Phase 1: To find the initial basic feasible solution, we solve the auxiliary problem

Minimize $r = R_2$

Subject to $4y_1 - x_2 - 3x_3 + s_1 = 5$

$-y_1 + x_2 + 2x_3 - s_2 + R_2 = 9$

$-3y_1 + x_2 + 4x_3 + s_3 = 14$

$0 \le y_1 \le 2,\ 0 \le x_2 \le 5,\ 0 \le x_3 \le 2$

$s_1,\ s_2,\ s_3,\ R_2 \ge 0.$

Organization of the tables

In the tables below, the elements in brackets are the pivot elements. An arrow pointing towards inside a column means the variable in that column is the entering variable and that column is the pivot column. An arrow pointing towards outside a row means the variable in that row is the leaving element while that row is the pivot row.

Table 4.15: The initial Tableau

BV	y_1	x_2	$x_3 \downarrow$	s_2	s_1	R_2	s_3	Solution
z	-1	1	2	-1	0	0	0	9
s_1	4	-1	-3	0	1	0	0	5
R_2	-1	1	2	-1	0	1	0	9
s_3	-3	1	4	0	0	0	1	14

In Table4.15, x_3 enters basis since it has the largest z-value. To decide the leaving variable, we compute

$$\theta_1 = \min\left(\frac{9}{2}, \frac{14}{2}\right) = \frac{7}{2}$$

$$\theta_2 = \min\left(\frac{\infty - 5}{-3}\right) = \infty, \text{ upper bound on } s_1$$

$$u_3 = 2, \text{ upper bound on the entering variable } x_3.$$

Thus, $\theta = \min(\theta_1, \theta_2, \mu_3) = 2$. This is the case where $\theta = \mu_3$. Thus x_3 enters basis at its upper bound as nonbasic. We use the relation

$$x_3 = \mu_3 - x_3^1, \; 0 \le x_3^1 \le 2.$$

In table 4.15, we change x_3 to x_3^1 with column of x_3^1 just negative of the column of x_3. In fact, x_3 does not enter into basis but it remains as nonbasic at its upper level as indicated in Table 4.16 below.

Table 4.16: The second tableau

BV	y_1	$x_2 \downarrow$	x_3^1	s_2	s_1	R_2	s_3	solution
z	-1	1	-2	0	0	0	0	5
$\leftarrow s_1$	4	-1	3	0	1	0	0	11
R_2	-1	1	-2	-1	0	1	0	5
s_3	-3	1	-1	0	0	0	1	6

In Table4.16, x_2 is the entering variable since it has the largest z-row value. To decide the leaving variable, we compute

$$\theta_1 = \min\left(\frac{5}{1}, \frac{6}{1}\right) = 5$$

$$\theta_2 = \min\left(\frac{\infty - 11}{-(-1)}\right) = \infty, \text{ upper bound on } s_1$$

$$u_2 = 5, \text{ upper bound on } s_2$$

$$\theta = \min(\theta_1, \theta_2, u_2) = 5.$$

The minimum is for $\theta_1$and u_2. Obviously θ_1 is preferred, and R_2 leaves. We make simplex iteration and this is the end of phase 1. We discontinue with R_2 column toobtain table 4.17 below.

Table 4.17: The third Tableau

BV	y_1	x_2	x_3^1	s_2	s_1	s_3	solution
z							
s_1	3	0	1	-1	1	0	16
x_2	-1	1	-2	-1	0	0	5
s_3	-2	0	-2	1	0	1	1

Phase II: We rewrite the objective function as

Maximize $z = 4(y_1 + 1) + 2x_2 + 6(2 - x_3^1) = 4y_1 + 2x_2 - 6x_3^1 + 16$. Inserting this into the objective function in table 4.14,

that is from Table 4.14, $y_1 = -1$, $x_2 = 1$, $x_3^1 = -2$. This implies

$Z = 4(-1) + 2(1) - 6(-2) + 16 = 26$.

We obtain table below.

Table 4.18: The initial tableau

BV	$y_1 \downarrow$	x_2	x_3^1	s_2	s_1	s_3	solution
z	-6	0	2	-2	0	0	26
s_1	3	0	1	-1	1	0	16
$\leftarrow x_2$	(-1)	1	-2	-1	0	0	5
s_3	-2	0	-2	1	0	1	1

In Table4.18, y_1 is the entering variable since it has the most negative z value. To decide the leaving variable, we compute

$$\theta_1 = \min\left(\frac{16}{3}\right) = \frac{16}{3}$$

$$\theta_2 = \min\left(\frac{5-5}{1}, \frac{\infty-1}{2}\right) = 0.$$

$$u_1 = 2$$

The minimum value is 0 corresponding to x_2. Thus, y_1 enters and x_2 leaves. This is the case $\theta = \theta_2$. To achieve this, we just make simplex iterations and obtain table 1.16.

Tableau 4.19: The second Tableau

BV	y_1	x_2	x_3^1	s_2	s_1	s_3	solution
z	0	-6	14	4	0	0	-4
s_1	0	3	-5	-4	1	0	31
y_1	1	-1	2	1	0	0	-5
s_3	0	-2	2	3	0	1	-9

We make x_2 at its upper bound and by the substitution $x_2 = u_2 - x_2^1$, $0 \le x_2^1 \le 5$.

y_1– row gives

$$y_1 - x_2 + 2x_3^1 + s_2 + 0s_1 + 0s_3 = -5.$$

Since $u_2 = 5$, we make the substitution $5 - x_2^1$ in the above constraint equation to have Table 4.20 shown below.

Table 4.20: The final Tableau

BV	y_1	x_2^1	x_3^1	s_2	s_1	s_3	solution
z	0	6	14	4	0	0	26
s_1	0	3	-5	-4	1	0	16
y_1	1	1	2	1	0	0	0
s_3	0	2	2	3	0	1	1

Table 4.20 is the optimal tableau since all z – row values are positive. The optimal solution is;

$y_1 = 0$, this implies $x_1 - 1 = 0$, that is $x_1 = 1$

$x_2^1 = 0$, this implies $5 - x_2 = 0$, that is $x_2 = 5$

$x_3^1 = 0$, this implies $2 - x_3 = 0$, that is $x_3 = 2$, $z = 26$.

4.9: ILLUSTRATION OF THE COLUMN GENERATION METHOD

To illustrate the column generation method, we consider the following example.

Example 4.5: (Winston, 2003): Woodco sells 3-ft, 5-ft, and 9-ft pieces of lumber. Woodco's customers demand 25 3-ft boards, 20 5-ft boards, and 15 9-ft boards. Woodco, who must meet its demands by cutting up 17-ft boards, wants to minimize the waste incurred. Formulate a Linear Programmingmedel to help Woodco accomplish its goal, and solve the LP by column generation.

Woodco must decide how each 17-ft board should be cut. Hence, each decision corresponds to a way in which a 17-ft board can be cut. For example, one decision variable would correspond to a board being cut into three 5-ft boards, which would incur waste of 17 - 15 = 2 ft. Many possible ways of cutting a board need not be considered. For example, it would be foolish to cut a board into one 9-ft and one 5-ft piece; we could just as easily cut the board into a 9-ft piece, a 5-ft piece, *and* a 3-ft piece. In general, any cutting pattern that leaves 3 ft or more of waste need not be considered because we could use the waste to obtain one or more 3-ft boards.

Table 4.21 show the ways to cut a 17-ft board.

Table 4.21: Ways to cut a 17-ft board

	Number of			
combination	3-ft board	5-ft board	9-ft board	Waist (feet)
1	5	0	0	2
2	4	1	0	0
3	2	2	0	1
4	2	0	1	2
5	1	1	1	0
6	0	3	0	2

We now define

x_i = number of 17-ft boards cut according to constraint i and formulate Woodco's linear programming model.

Woodco's waste + total customer demand = total length of boards cut.

Total customer demand = 25(3) + 20(5) + 15(9) = 310ft.

Total length of board cut = $17(x_1 + x_2 + x_3 + x_4 + x_5 + x_6)$.

Woodco's waste (in feet) = $17x_1 + 17x_2 + 17x_3 + 17x_4 + 17x_5 + 17x_6$ - 310.

Then Woodco's objective function is to

Minimize z = $17x_1 + 17x_2 + 17x_3 + 17x_4 + 17x_5 + 17x_6$- 310.

This is equivalent to minimizing $17(x_1 + x_2 + x_3 + x_4 + x_5 + x_6)$ which is equivalent to

Minimizing $x_1 + x_2 + x_3 + x_4 + x_5 + x_6$.

Hence, Woodco's objective function is

Minimize z $= x_1 + x_2 + x_3 + x_4 + x_5 + x_6$

This means Woodco can minimize its total waste by minimizing the number of 17-ft boardsthat are cut.

Woodco faces the following three constraints.

Constraint 1: At least 25 3-ft boards must be cut.

Constraint 2: At least 20 5-ft boards must be cut.

Constraint 3: At least 15 9-ft boards must be cut.

Because the total number of 3-ft boards that are cut is given by

$5x_1 + 4x_2 + 2x_3 + 2x_4 + x_5$, constraint 1 becomes $5x_1 + 4x_2 + 2x_3 + 2x_4 + x_5 \geq 25$.

Similarly, constraint 2 becomes $x_2 + 2x_3 + x_5 + 3x_6 \geq 20$ and constraint 3 becomes

$x_4 + x_5 \geq 0$.

The linear programming model is thus

$$\text{Minimize } z = x_1 + x_2 + x_3 + x_4 + x_5 + x_6$$

Subject to

$$5x_1 + 4x_2 + 2x_3 + 2x_4 + x_5 \geq 25 \quad \text{(3 –ft constraint)}$$

$$x_2 + 2x_3 + x_5 + 3x_6 \geq 20 \quad \text{(5-ft constraint)}$$

$$x_4 + x_5 \geq 0. \quad \text{(9-ft constraint)}$$

We note that $x1$ only occurs in the 3-ft constraint (because combination 1 yields only 3-ft boards), and $x6$ occurs in the 5-ft constraint (because combination 6 yields only 5-ft boards). This means that $x1$ and $x6$ can be used as starting basic variables for the 3-ft and 5-ft constraints. Unfortunately, none of combinations yields only 9-ft boards, so the

9-ft constraint has no obvious basic variable. To avoid having to add an artificial variable to the 9-ft constraint, we define combination 7 to be the cutting combination that yields only one 9-ft board. Also, define $x7$ to be the number of boards cut according to combination

7. Clearly, x_7 will be equal to zero in the optimal solution, but inserting x_7 in the starting basis allows us to avoid using the Big M or the two-phase simplex method.

Note that the column for $x7$ in the LP constraints will be

$\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$ and a term x_7 will be added to the objective function. We can now use basic variable BV = $\{x_1, x_6, x_7\}$ as a starting basis for the model. If we let the tableau for this basis be the tableau 0, then we have

$$B_0 = \begin{bmatrix} 5 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \qquad B_0^{-1} = \begin{bmatrix} 1/5 & 0 & 0 \\ 0 & 1/3 & 0 \\ 1 & 0 & 1 \end{bmatrix}.$$

Then

$$C\,B_V B_0^{-1} = [1\ 1\ 1] \begin{bmatrix} 1/5 & 0 & 0 \\ 0 & 1/3 & 0 \\ 1 & 0 & 1 \end{bmatrix} = [1/5\ \ 1/3\ \ 1].$$

If we now price out each nonbasic variable it would tell us which variable should enterthe basis. However, in a large-scale cutting stock problem, there may be thousands of variables, so pricing out each nonbasic variable would be an extremely tedious chore. This is the type of situation in which column generation comes into play. Because we are solving a minimization problem, we want to find a column that will price out positive (have a positive coefficient in row0). In the cutting stock problem, each column, or variable, represents a combination for cutting up a board:

A variable is specified by three numbers:

a_3, a_5, and a_9, where $a_i$is the number of i-ft boards yielded by cutting one 17-ft board according to the given combination. For example, the variable $x_2$is specified by a_3 =4,

a_5= 1, and a_9=0. The idea of column generation is to search efficiently for a column that will price out favourably (positive in minimization problem and negative in maximization problem). For our current basis, a combination specified by a_3, a_5 and a_9 will price out as

$$CB_V B_0^{-1} = \begin{bmatrix} a_3 \\ a_5 \\ a_9 \end{bmatrix} - 1 = 1/5a_3 + 1/3\, a_5 + a_9 - 1.$$

We note that a_3, a_5, and a_9 must be chosen so they don't use more than 17-ft wood. We also know that by a_3, a_5 and a_9 must be nonnegative integers. In short, for any combination, a_3, a_5 and a_9 must satisfy

$3a_3 + 5a_5 + 9a_9 \leq 17$ ($a_3 \geq 0, a_5 \geq 0$, $a_9 \geq 0$, a_3, a_5, a_9 are integers).

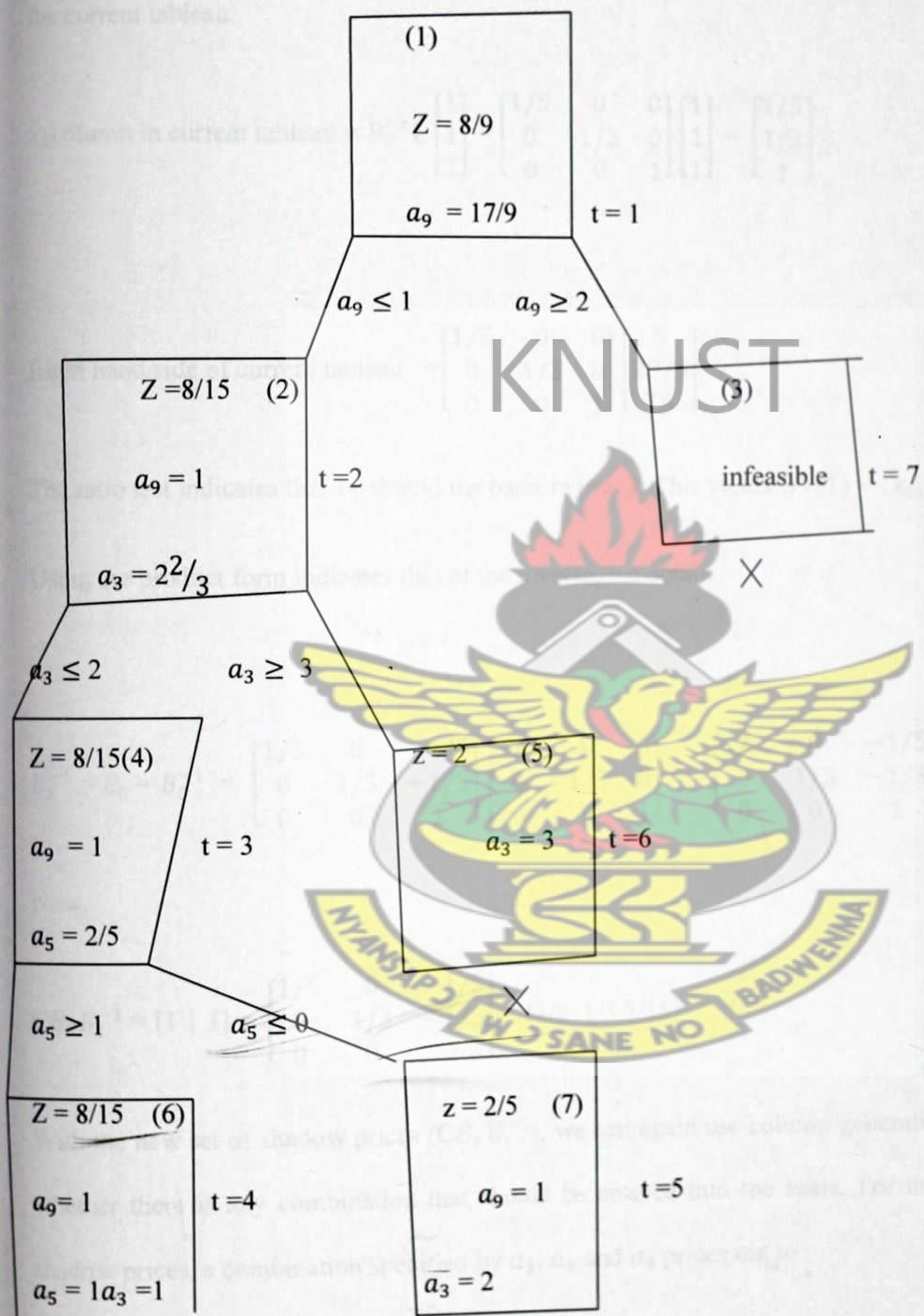
We can now find the combination that prices out most favourably by solving the following knapsack problem:

$$\text{Max } z = 1/5a_3 + 1/3\, a_5 + a_9 - 1.$$

$$\text{Subject to } \quad 3a_3 + 5a_5 + 9a_9 \leq 17$$

$a_3 \geq 0, a_5 \geq 0$, $a_9 \geq 0$, a_3, a_5, a_9 are integers

Because the above problem is knapsack (without 0 -1 restrictions on the variables), it can be solve by using the branch and bound procedure. The resulting branch and bound tree is shown in fig 4.4. From fig 4.4 we find that the optimal solution is z = 8/5, $a_3 = a_5 = a_9$= 1. This corresponds to 5 combination and variable x_5. Hence, x_5 = prices out 8/15 and entering x_5 into basis will decrease Woodco's waste.



Candidates Figure4.4 Branch and Bound Tree for example 4.5

To enter x_5 into basis, we create the right hand side of the current tableau and the x_5 column of the current tableau.

$$x_5\text{column in current tableau} = B_0^{-1}\text{b}\begin{bmatrix}1\\1\\1\end{bmatrix} = \begin{bmatrix}1/5 & 0 & 0\\0 & 1/3 & 0\\0 & 0 & 1\end{bmatrix}\begin{bmatrix}1\\1\\1\end{bmatrix} = \begin{bmatrix}1/5\\1/3\\1\end{bmatrix}$$

$$\text{Right hand side of current tableau} = \begin{bmatrix}1/5 & 0 & 0\\0 & 1/3 & 0\\0 & 0 & 1\end{bmatrix}\begin{bmatrix}5\\20/3\\15\end{bmatrix}$$

The ratio test indicates that x_5 should the basis in row 3. This yields BV(1) = $\{x_1, x_6, x_5\}$.

Using the product form indicates that of the inverse, we obtain

$$B_1^{-1} = E_0 = B_0^{-1} = \begin{bmatrix}1/5 & 0 & -1/5\\0 & 1/3 & -1/3\\0 & 0 & 1\end{bmatrix}\begin{bmatrix}1/3 & 0 & 0\\0 & 1/3 & 0\\0 & 0 & 1\end{bmatrix} = \begin{bmatrix}1/5 & 0 & -1/5\\0 & 1/3 & -1/3\\0 & 0 & 1\end{bmatrix}.$$

Now,

$$CB_VB_1^{-1} = [1\ 1\ 1] = \begin{bmatrix}1/5 & 0 & -1/5\\0 & 1/3 & -1/3\\0 & 0 & 1\end{bmatrix} = [1/5\ \ 1/3\ 7/15].$$

With ~~the new~~ set of shadow prices ($CB_VB_1^{-1}$), we can again use column generation to determine whether there is any combination that should be entered into the basis. For the current set of shadow prices, a combination specified by a_3, a_5 and a_9 prices out to

$$[1/5\ 1/3\ 7/15]\begin{bmatrix}a_3\\a_5\\a_9\end{bmatrix} - 1 = 1/5a_3 + 1/3a_5 + 7/15a_9 - 1$$

For the current tableau, the column generation procedure yields the following problem:

Maximize $z = 1/5a_3 + 1/3a_5 + 7/15a_9 - 1$

Subject to $3a_3 + 5a_5 + 9a_9 \leq 17$

$a_3, a_5, a_9 \geq 0$; a_3, a_5, a_9 are integers.

The branch and bound tree for the above problem is shown below.

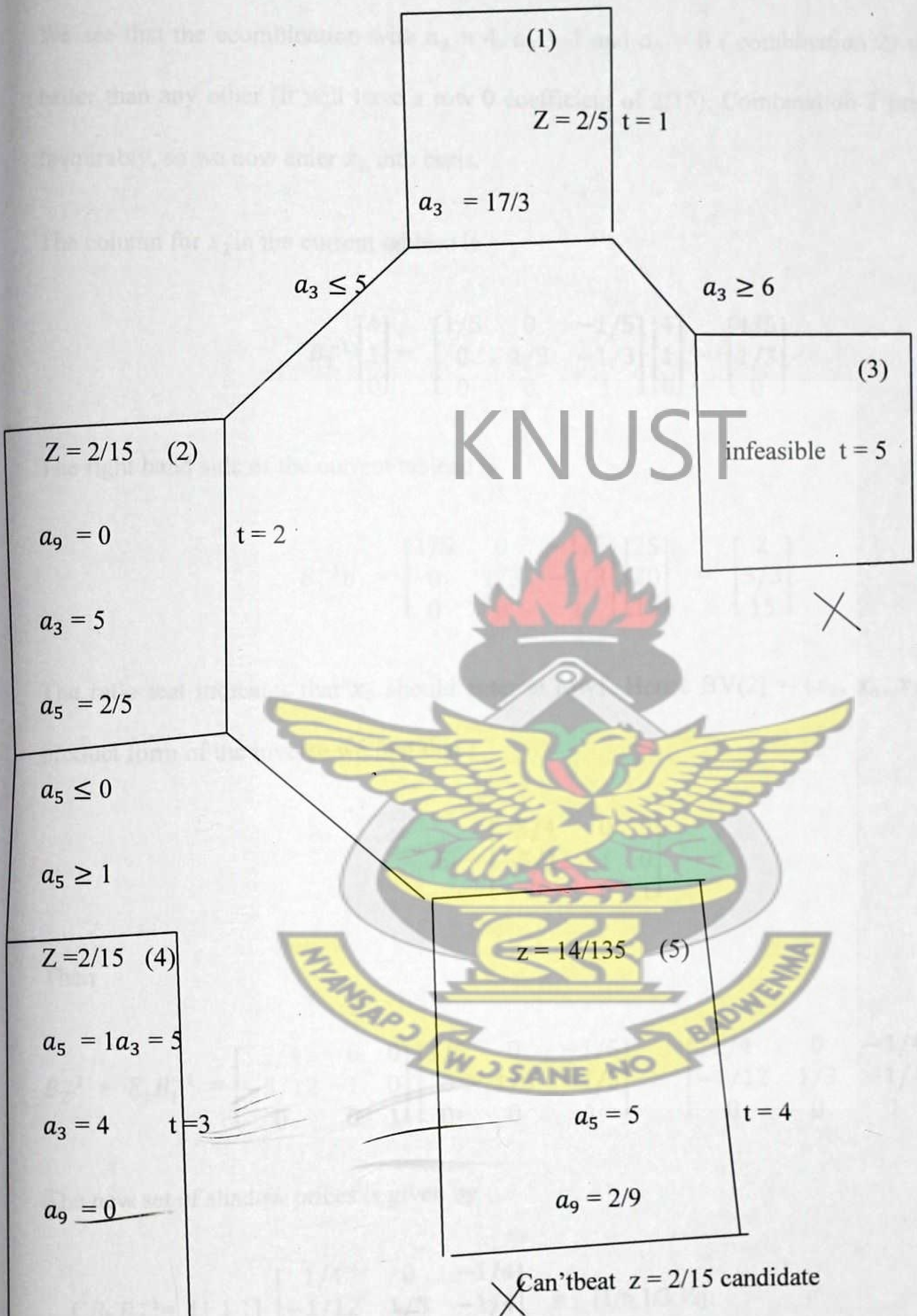


Figure 4.5 Branch and bound tree for example 4.5

We see that the ccombination with $a_3 = 4$, $a_5 = 1$ and $a_9 = 0$ (combination 2) will price out better than any other (it will have a row 0 coefficient of 2/15). Combination 2 prices out most favourably, so we now enter x_2 into basis.

The column for $x_2$in the current tableau is

$$B_1^{-1}\begin{bmatrix}4\\1\\0\end{bmatrix} = \begin{bmatrix}1/5 & 0 & -1/5\\0 & 1/3 & -1/3\\0 & 0 & 1\end{bmatrix}\begin{bmatrix}4\\1\\0\end{bmatrix} = \begin{bmatrix}4/5\\1/3\\0\end{bmatrix}$$

The right hand side of the current tableau is

$$B_1^{-1}\mathrm{b} = \begin{bmatrix}1/5 & 0 & -1/5\\0 & 1/3 & -1/3\\0 & 0 & 1\end{bmatrix}\begin{bmatrix}25\\20\\15\end{bmatrix} = \begin{bmatrix}2\\5/3\\15\end{bmatrix}.$$

The ratio test indicates that x_2 should enter at row1. Hence BV(2) = $\{x_2, x_6, x_5\}$. Using the product form of the inverse we find that

$$E_1 = \begin{bmatrix}5/4 & 0 & 0\\-5/12 & 1 & 0\\0 & 0 & 1\end{bmatrix}.$$

Then

$$B_2^{-1} = E_1B_1^{-1} = \begin{bmatrix}5/4 & 0 & 0\\-5/12 & 1 & 0\\0 & 0 & 1\end{bmatrix}\begin{bmatrix}1/5 & 0 & -1/5\\0 & 1/3 & -1/3\\0 & 0 & 1\end{bmatrix} = \begin{bmatrix}1/4 & 0 & -1/4\\-1/12 & 1/3 & -1/4\\0 & 0 & 1\end{bmatrix}.$$

The new set of shadow prices is given by

$$CB_VB_2^{-1} = [1\ 1\ 1]\begin{bmatrix}1/4 & 0 & -1/4\\-1/12 & 1/3 & -1/4\\0 & 0 & 1\end{bmatrix} = [1/6\ 1/3\ ½].$$

For this set of shadow prices, a combination specified by a_3, a_5, and a_9 will price out to

$1/6a_3 + 1/3a_5 + 1/2a_9 - 1$. Thus the column generation procedure required us to solve the following problem

$$\text{Maximize } z = 1/6a_3 + 1/3a_5 + 1/2a_9 - 1$$

$$\text{Subject to } 3a_3 + 5a_5 + 9a_9 \leq 17$$

$a_3, a_5, a_9 \geq 0;\ a_3, a_5, a_9$ integers.

The optimal z value is found to be $z = 0$. This means that no combination can price out favourably. Hence, our current basic solution must be an optimal solution. To find the values of the basic variables in the optimal solution, we find the right hand side of the current tableau;

$B_2^{-1}\text{b} = \begin{bmatrix} 1/4 & 0 & -1/4 \\ -1/12 & 1/3 & -1/4 \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} 25 \\ 20 \\ 15 \end{bmatrix} = \begin{bmatrix} 5/2 \\ 5/6 \\ 15 \end{bmatrix}$. Therefore the optimal solution to Woodco's cutting stock problem is given by $x_2 = 5/2,\ x_6 = 1,\ x_5 = 15$.

CHAPTER 5

CONCLUSION AND RECOMMENDATIONS

This chapter is the concluding part of this thesis. In this chapter, we find out whether the title of this thesis has adequately been dealt with. We also find out whether the objectives of the thesis have been achieved and other findings of this thesis will also be discussed. Recommendations based on the findings of the thesis will also be made.

5.1: CONCLUSION

Chapter 3 of this thesis contains all the important theories of linear programming. Some methods of solutions of linear programming have also been treated in chapter three. All the necessary algorithms, procedures and steps involved in the applications of these methods have also been given in details in chapter three. All the necessary steps involved in the formulation of linear programming models have also been given in chapter three. These methods include the Graphical method, the Revised Simplex method, the Dual Simplex method, the Bounded Variable Technique, the Decomposition Principle, the Karmardar's method and the Column Generation method.

In chapter 4, illustrative examples have been given. The required linear programming models needed in solving the given examples have also been formulated out of the given examples. These examples have been used to illustrate the Graphical method of linear programming, the simplex method of linear programming, the Revised method of linear programming, the dual simplex method of linear programming and the Karmarkar's method of linear programming. All the necessary computations have been made to clearly illustrate the methods mentioned above for better understanding. All the tables involved in computing the solutions of the given examples that are used to illustrate these methods have also been given and systematically explained.

Data from Express Savings and Loans Company Limited (ESLCL) has been analysed. A loan policy has been formulated for ESLCL using linear programming model.

It can, therefore, be concluded that the title of this thesis has adequately been dealt with.

It can also be concluded that the objectives of the thesis have been achieved.

So many findings have come out of this thesis. This thesis has shown that there are so many methods available that can be used to solve linear programming problems of all kinds. It has been found out of this thesis that linear programming is a very powerful mathematical tool which can be applied to enable us maximize profit and minimize cost in our everyday activities. It has also been found that linear programming can be applied in various fields of study. It can be used most extensively in businesses and economics. Linear programming can also be utilized for some engineering problems. Industries such as energy, telecommunications and manufacturing can also use linear programming models to solve problem to ensure the growth. Linear programming can be useful in diverse types of problems in planning and designing. Linear programming can also help us to think logically and that will enable us to solve many problems in our societies.

5.2 RECOMMENDATIONS

The following recommendations are made out of this thesis.

It is recommended that when solving linear programming problems with any of the linear programming methods such as the graphical method, the simplex method, the revised simplex method, the dual simplex method etc., students should make sure they apply the details of the theory behind the method that they are using.

It is also recommended to students that they should give logical illustrations of their solutions for clear understanding. That is, they should ensure that the linear programming models are properly formulated to avoid any errors.

All the procedures leading to the solution of the problem should be carefully followed in order to obtain a very good solution.

This thesis will benefit Ghanaian students in so many ways. The theory of linear programming and some methods of solution have been clearly explained in chapter three of the thesis to give students clear understanding of linear programming.

In Chapter 4 of the thesis, the methods of linear programming have been used to solve a linear programming problem. All the necessary procedures involved in the application of the methods have been followed to illustrate the methods to give logical and easy understanding to students.

It is further recommended that linear programming should be applied in all our everyday activities to enable us maximize profit and reduce cost.

It is also further recommended that industries such as energy, telecommunications etc. in Ghana must applied linear programming models to ensure maximum growth.

It is also recommended that all businesses and engineering firms in Ghana must apply linear programming models since that will enable them to maximize profit and reduce cost.

It is also recommended that we must apply linear programming whenever we want to solve problems in our societies.

REFERENCES

Ackoff, R.L. (1974), Redesigning the future: a systems approach to societal problems. Wiley, New York, NY

Ackoff, R.L. (1978), The art of problem solving. Wiley, Inc., New York, NY

Ahuja et al. (1993), Network flows: theory, algorithms, and applications. Prentice-Hall, Englewood Cliffs, New Jersey.

Amponsah, S. K. (2009), Optimization Techniques 1. IDL, KNUST, Kumasi.

Ballestero, E. and Romero, C. (1998), Multiple criteria decision making and its application to economic problems. Kluwer, Boston Dordrecht London.

Bazaraa et al. (1993), Nonlinear programming: theory and algorithms. Wiley, New York, NY

Bhatti, M.A. (2000), Practical optimization methods with Mathematica applications. Springer, New York, NY

Bland, R.G. (1977), New finite pivoting rules for thesimplex method. Mathematics of Operations Research 2: 103-107

Bracken, J. and McGill, J. (1973), Mathematical programs with optimization problems in the constraints. Operations Research 21: 37–44

Bradley, S.P. and Magnanti, T.L. (1977), Applied mathematical programming. Addison Wesley, Reading, MA

Bretscher, O. (1997), Linear algebra with applications. Prentice-Hall, Upper Saddle River, NJ

Brown, G.W. and Koopmans, T.C. (1951), Computational suggestions for maximizing a linear function subject to linear inequalities. Wiley, New York, NY,

Buchanan, J. and Gardiner, L. (2003), A comparison of two reference point methods in multiple objective mathematical programming. European Journal of Operational Research 149: 17-34

Bunch, J.R. and Parlett, B.N. (1971), Direct methods for solving symmetric indefinite systems of linear equations. SIAM Journal on Numerical Analysis 8: 639-655

Charnes, A. (1952), Optimality and degeneracy in linear programming. Econometrica 20: 160-170

Charnes, A. and Cooper, W.W. (1961), Management models and industrial applications of linear programming. Wileys, New York, NY

Charnes and Cooper, W.W. (1955), Optimal estimation of executive compensation by linear programming. Management Science 1, 138-151

Charnes, A. and Henderson, A. (1953), An introduction to linear programming. Wiley, New York

Charnes, A. (1952), Blending aviation gasolines—a study in programming interdependent activities in an integrated oil company.Econometrica 20: 135-159

Charnes,A. and Rhodes, E. (1978), Measuring the efficiency of decision-making units. European Journal of Operational Research 2: 429-444

Charnes, A. and Klingman, D. (1971), The more-for-less paradox in the distribution model. Cahiers de centre d'études recherche operationnelle 13: 11-22

Charnes, A. and Lemke, C.E. (1954), Computational theory of linear programming I: the bounded variables problem. ONR Research Memo No. 10, Graduate School of Industrial Administration, Carnegie-Mellon University, Pittsburgh, PA

Chen, G. and Yang, X. (2005), Vector optimization. Lecture Notes in Economics & Mathematical Systems 541, Springer, Berlin

Cochrane, J.L. andZeleny, M. (1973), Multiple criteria decision making. University of South Carolina Press, Columbia, SC

Cohon, J. L. (1978),Multiobjective Programming Planning. In: Mathematics in Science and Engineering, Vol. 140, Academic Press, New York.

Collette, Y. and Siarry, P. (2003), Multiobjective optimization: principles and case studies. Springer, Berlin Heidelberg New York

Colson, B. and Savard, G. (2005), Bilevel programming: a survey. 4OR 3: 87-107

Coopersmith, W. and Sumutka, A.R. (2011), Tax – Efficient Retirement Withdrawal Planning using Linear Programmig. http:www.fpanet.org/journal, (accessed 2011 September 25).

Cook, S.A. (1971), The complexity of theorem-proving procedures. Proceedings of the 3rd Annual ACM Symposium on the Theory of Computing, New York,

Cooper, L. and Steinberg, D. (1970), Introduction to methods of optimization. W. B. Saunders Co., Philadelphia, PA

Dantzig, G.B. (1951), Application of the simplex method to a transportation problem. In: Koopmans TC (ed), Activity analysis of production and allocation. Cowles Commission Monograph 13, Wiley, New York, pp 359-373

Dantzig, G.B. (1954), Notes on linear programming, Parts VIII, IX, X, Upper bounds, Secondary constraints and block triangularity in linear programming. The Rand Corporation, Research Memorandum RM-1367, Oct 4, 1954. Published under the same title in Econometrica 23, 1955, 174-183

Dantzig, G.B. (1963), Linear programming and extensions. Princeton University Press, Princeton, NJ

Dantzig, G.B. (1982), Reminiscences about the origins of linear programming. The State of the Art.Springer, Berlin Heidelberg New York.

Dantzig, G.B. (1991), Linear programming. In: Lenstra JK, RinnooyKan AHG, Elsevier Science, Amsterdam, Dantzig, G.B. and Orchard-Hays, W. (1956), A primal-dual algorithm for linear programming. Princeton University Press, Princeton, NJ, pp 171-181

Dantzig, G.B. and Orchard-Hays, W. (1953), Noteson linear programming: part V – alternate algorithm for the revised simplex method using product form of the inverse. TM-1268, The RAND Corporation, Santa Monica, CA

Dantzig, G.B and Ordens, A. (1954), The solution of a large-scale traveling salesman problem. Operations Research 2: 393-410

Dantzig, G.B. andOrden, A. (1953), Duality theorems, Rand report RM-1265. The Rand Corporation, Santa Monica, CA

Dantzig, G.B. andThapa, M.N. (1997), Linear programming. I: Introduction. Springer, New York Berlin Heidelberg

Dantzig, G.B. andThapa, M.N. (2003),Linear programming, 2: Theory and extensions. Springer, New York Berlin Heidelberg Dantzig GB, Van Slyke RM (1967) Generalized upper bounded techniques for linear programming. Journal of Computer and Systems Sciences 1: 213-226.

Dantzig, G.B. (1963), Linear Programming and Extensions. Princeton University Press, New York.

Dantzig, G.B. and Wolfe, P. (1961), The decomposition algorithm for linear programs. Econometrica 29:767-778

Dantzig, G.B. and Wolfe, P. (1960), Decomposition principle for linear programs.Operations Research 8:101-111.

David, G.L. and Yingu, Y. (2008), Linear and Nonlinear Programming, 3rd ed., Springer Science + Business Media Publications, LLC233, Spring Street, New York.

DeMarr, R. (1983), External pivoting in linear programming. Abstracts presented to the American Mathematical Society, issue 22, vol 4, no 1

Dempe, S. (2002), Foundations of bilevel programming, Kluwer, Boston, MA

Dikin, I. (1967), Iterative solution of problems of linear and quadratic programming. Soviet Mathematics Doklady 8: 674-675

Dwyer, P.S. (1975), Transportation problems with some *xij*negative and transshipment problems. Naval Research Logistics Quarterly 22: 751-776

Dyer, J.C. (1972), Interactive goal programming. Management Science 19: 62-70

Edmonds, J. (1965), Paths, trees, and flowers. Canadian Journal of Mathematics 17: 449-467

Ehrgott, M. (2005), Multicriteriaoptimization.Springer, Berlin Heidelberg New York

Eiselt, H.A. andLaporte, G. (1987), Combinatorial optimization problems with soft and hard requirements. Journal of the Operational Research Society 38: 785-795

Eiselt, H.A. andSandblom, C-L.(1985), External pivoting in the simplex algorithm. StatisticaNeerlandica 39: 327-341

Eiselt, H.A. andSandblom, C-L.(1990), Experiments with external pivoting. Computers &Operations Research 17: 325-332

Eisel, H.A. and Sandblom, C.L. (2000), The bounce algorithm for mathematical programming. Mathematical Methods of Operations Research 52: 173-183

Eiselt, H.A. and Sandblom, C.L. (2000), Integer programming andnetwork models. Springer, Berlin Heidelberg New York

Eiselt, H.A. and Sandblom, C.L. (2004), Decision analysis, location models, and scheduling problems.Springer, Berlin Heidelberg New York.

Eise~~lt, H.A~~. and Sandblom C.L. (2007), Linear Programming and its Applications.Library of Congress, Springer Berlin Heiderberg NY.

Evans, J.P. andSteuer, R.E. (1973), A revised simplex method for linear multiple objective programming. Mathematical Programming 5: 54-72

Farrell, J.M. (1957), The measurement of productive efficiency. Journal of the Royal Statistical Society 120: 253-281.

Ferris et al. (2007), Linear Programming with Matlab.Society for industrial and Applied Mathematics Publications, Philadelphia.

Feylizadeh, M.N. and Bagherpour, (2011), ''Application of Optimization Techniques in Production Planning Context: A Review and Extension'',(http://scialert.net/abstract/?doi=ijmsaj.2011.18), (accessed 2011 September 24).

Fiacco, A.V. (1979), Barrier methods for nonlinear programming. In: Holzman AG (ed) Operations research methodology. Marcel Dekker, New York Basel, pp377-440

Fiacco A.V and McCormick GP (1968), Nonlinear programming: sequential unconstrained minimization techniques. Wiley, New York, NY

Finke, G. (1977), A unified approach to reshipment, overshipment and post-optimization problems. In: StoerJ (ed) Optimization techniques. Proceedings of the 8th IFIP Conference, Part 2. Lecture Notes in Control and Transportation Sciences 7, Springer, Berlin Heidelberg New York, pp 201-208

Finke, G. (1983), Minimizing overshipments in bottleneck transportation problems. INFOR 21: 121-135

Fliege, J. and Vicente, L.N. (2006), Multicriteria approach to bilevel optimization. Journal of Optimization Theory and Applications 131: 209-225

Ford Jr, L.R. and Fulkerson, D.R.(1954), Maximal flow through a network. Rand Research Memorandum 1400, The RAND Corporation, Santa Monica Subsequently published under the same title in the Canadian Journal of Mathematics 8, 1956, 3999

Frank, A. (2005), On Kuhn's Hungarian method—a tribute from Hungary. Naval Research Logistics 52: 2-5

Gal, T. (1979), Postoptimality analyses, parametric programming and related topics. McGraw Hill, New York

Gal, T. (1984), Linear parametric programming - A brief survey. Mathematical Programming Study **21:** 43-68

Gale, D. (1960), The theory of linear economic models. Mc-Graw-Hill, New York,NY

Gale et al. (1951), Linear programming and the theory of games: Koopmans, T.C. (ed), Activity analysis of production and allocation. Cowles Commission Monograph 13: 317-329

Garey, M.R and Johnson, D.S. (1979), Computers and intractability: A guide to the theory of NP-completeness. Freeman, San Francisco.

Garfinkel, R.S. andRao, M. (1976), Bottleneck linear programming. Mathematical Programming 11: 291-298

Gass, S.I and Assad, A.A. (2005), An annotated timeline of operations research: An informal history. Kluwer, New York, NY

Gass, S.I. andSaaty, T. (1955), The computational algorithm for the parametric objective function. Naval Research Logistics Quarterly 2: 39

Garner, G. S. and Gass, S.I. (1981), Stigler's diet problem revisited. Operations Research 49: 1-13

Geoffrion, A.M. (1967), Solvingbicriterion mathematical programs. Operations Research 15: 39-54

Geoffrion et al. (1972), An interactive approach for multi-criterion optimization with an application to the operation of an academic department. Management Science 19: 357-368

Gilmore, P.C. andGomory, R.E. (1961), A linear programming approach to the cutting stock problem. Operations Research 9: 849-859

Gilmore,P.C. andGomory, R.E. (1963), A linear programming approach to the cutting stock problem − Part II. Operations Research 11: 863-888

Grattan-Guinness, I. (1970), Joseph Fourier's anticipation of linear programming. Operational Research Quarterly 21: 361-364

Gutin, G. andPunnen, A. (2002),The traveling salesman problem and its variations. Kluwer, Boston Dordrecht London.

Hillier, F.S. and Lieberman, G.J. (2000), Introduction to Operations Research, 7th ed., McGraw-Hill Company Ltd, New York.

Hirsch, W.M. (1957), Hirsch conjecture, verbal communication to DantzigReferences 370

Hitchcock, F.L. (1941),The distribution of a product from several sources to numerous localities. Journal of Mathematical Physics 20: 224-230

Hobson, R.F. andWeinkam, J.J. (1979), Curve fitting. In: Holzman AG (ed) Operations research methodology. Dekker, New York Basel, pp 335-362

Hoffman, A.J. (1953), Cycling in the simplex algorithm. National Bureau of Standards, Report No. 2974, Washington, D.C

Hoffman, A.J. andKruskal, J.G. (1956), Integral boundary points of convex polyhedra. In: Kuhn, H.W and Tucker, A.W. (eds) Linear inequalities and related systems. Annals of Mathematics Studies, Princeton University Press, Princeton, NJ, pp 223-246

Huard, P. (1967), Resolution of mathematical programming with nonlinear constraints by the method of centers. In: Abadie J (ed) Nonlinear programming. North-Holland, Amsterdam, pp 209-219

Hussein, M.L. and Al-Ghaffar, F.S.A. (1996), An interactive approach for vector optimization problems. European Journal of Operational Research 89: 185-192

Ignizio, J.P. (1982), Linear programming in single- & multiple-objective systems.

Prentice-Hall, Englewood Cliffs, NJ

Ignizio, J.P. and Cavalier, T.M. (1994), Linear programming. Prentice-Hall, Englewood Cliffs, NJ

Ijiri, Y. (1965), Management goals and accounting for control. Rand McNally, Chicago, IL

Inuiguchi, M. andRamík, J. (2000), Possibilistic linear programming: A brief review offuzzy mathematical programming and a comparison with stochastic programming in portfolio selection problem. Fuzzy Sets and Systems 111: 3-28

Jeroslow, S. and R.G. (1973), There cannot be any algorithm for integer programming with quadratic constraints. Operations Research 21: 221-224.

Jiri, M. and Gartner, B. (2007), Understanding and Using Linear Programming.Springer publications Inc., New York.

Jones, D.F. andTamiz, M. (2002), Goal programming in the period 1990-2000. In: Ehrgott

Kantorovich, L.V. (1939, 1960), Mathematical methods in the organization and planning of production.Translation of the 1939 original (in Russian). In: Management Science **6:** 36

Karmarkar, N. (1984),A new polynomial-time algorithm for linear programming. Combinatorica 4: 373-395

Karp, R.M. (1972), Reducibility among combinatorial problems. Plenum Press, New York.

Kasana H.S and Kumar K.D. (2004), Introductory Operations Research – Theory and Applications.Springer-Verlag Berlin Heidelberg.

Khachian, L.G. (1979), Polynomial algorithm for linear programming. DokladyAkademiiNauk SSSR 244: 1093-1096 (in Russian), translated into English in the same year in Soviet Mathematics Doklady 20: 191-194

Klee, V. and Minty, G.J. (1972), How good is the simplex algorithm. In: Shisha O (ed) Inequalities III. Academic Press, New York, pp 159-175

Klee, V.L. and Walkup, D.W. (1967), The*d*-step conjecture for polyhedra of dimension *d* 6. ActaMathematica 117: 53-78

Koopmans, T.C.(1951), Activity analysis of production and allocation. Wiley, New York, NY

Koopmans, T.C. (1951), Analysis of production as an efficient combination of activities.

Koopmans, T.C. and Beckmann, M. (1957), Assignment problems and the location of economic activities.Econometrica 25: 53-76

Kuhn, H.W. (1955), The Hungarian method for the assignment problem. Naval Research Logistics Quarterly 2: 83-97

Kuhn, H.W. (1991), On the origin of the Hungarian method. In: Lenstra JK, RinnooyKan AHG,

Kuhn, H.W. and Tucker, A.W. (1951), Nonlinear programming. In: Neyman J (ed) Proceedings of second Berkeley symposium on mathematical statistics and probability. University of California Press, Berkeley, CA, pp 481 – 491

Kuhn, H.W. and Tucker, A.W. (1956), Linear inequalities and related systems. Annals of Mathematics Studies 38. Princeton, NJ

Ladner, R.E. (1977), The computational complexity of provability in systems of modal propositional logics. SIAM Journal on Computing 6: 467-480

Land, A.H. andDoig, A.G. (1960), An automatic method of solving discrete linear programming problems. Econometrica 28: 496-520

Lee, ~~S.M.~~ (1972), Goal programming for decision analysis. Auerbach, Philadelphia, PA

Lemke, C.E. (1954), The dual method of solving linear programming problems. Naval Research Logistics Quarterly 1: 36-47

Leontief, W.W. (1936), Quantitative input and output relations in the economic system of the United States. Review of Economic Statistics 18: 105-125

Levin, A. and Yu (1965), On an algorithm for the minimization of convex functions. Soviet Mathematics Doklady 6: 286-290

Lootsma, F.A. (1972), A survey of methods for solving constrained minimization problems via unconstrained minimization. In: Lootsma FA (ed) Numerical methods for non-linear optimization. Academic Press, London

Mangasarian, O.L. (1969), Non linear programming.McGraw-Hill Publ. Co., New York, NY. Reprinted as SIAM Classic in applied mathematics 10, 1994, Philadelphia, PA

Manne, A.S. (1953), Notes on parametric linear programming. Rand Paper. The Rand Corporation, Santa Monica, CA

Marglin, J.A. (1967), Public investment criteria. MIT, Cambridge, MA

Markowitz,H.M.(1952), Portfolio selection. Journal of Finance 7: 77-91

McCloskey, J.F. (1987), The beginnings of operations research: 1934-1941. Operations Research 35: 143-152

McMullen, P. (1970), The maximum numbers of faces of a convex polytope. Mathematika 17: 179-184

Moore, J.H. and Weatherford, L.R. (2001), Decision modeling with Microsoft® Excel.Prentice Hall, Englewood Cliffs, NJ

Murty, K.G. (1986), The gravitational method for linear programming. Opsearch 23: 206-214

~~Murty, K.~~G. (1992), Network programming. Prentice Hall, Englewood Cliffs, NJ

Nemhauser, G.L.and Wolsey, L.A. (1988), Integer and combinatorial optimization. Wiley, New York, NY

Neumann, J. and von Morgenstern, O.(1944), Theory of games and economic behavior.Princeton University Press, Princeton, NJ

Nicholson, K. (2006), Linear algebra with applications. 5th edn. McGraw Hill, Boston, MA

Norback, J.P. and Morris, J.G. (1980), Fitting hyperplanes by minimizing orthogonal deviations. Mathematical Programming 19: 102-105

Orchard-Hays. W. (1955), Notes on linear programming (Part 6) The RAND vode for the simplex method (SX4). Report #1440, The RAND Corporation, Santa

Monica, C.A. (Based on the author's unpublished Master's Thesis in 1952)

Orden, A. (1952), Application of the simplex method to a variety of matrix problems.

Orden. A. and Goldstein, L. (eds) Proceedings of the Symposium on linear inequalities and programming.

Padberg, M. (1995), Linear optimization and extensions. Springer, Berlin Heidelberg New York

Parkinson, C.N. (1957), Parkinson's law. Houghton Mifflin, Boston, MA

Pawlak, Z. (1991), Rough sets: Theoretical aspects of reasoning about data. Kluwer, Dordrecht

Philip, J.(1972), Algorithms for the vector maximization problem. Mathematical Programming 2: 207-229

Rardin, R.L. (1998), Optimization in operations research. Prentice-Hall, Upper Saddle River, NJ

Reeves,G.R. and Franz, L.S. (1985), A simplified interactive multiple objective linear programming procedure. Computers & Operations Research 12: 589-601

Reeves, G.R. and Hedin, S.R. (1993), A generalized interactive goal programming procedure. Computers & Operations Research 20: 247-253

Renegar, J. (1988), A polynomial-time algorithm based on Newton's method for linear programming. Mathematical Programming 40: 59-93

Roos et al. (2006), Interior point methods for linear optimization. 2nd edn. Springer, New York, NY

Saigal, R. (1995), Linear programming. Kluwer, Boston, MA

Sakawa, M. (2002), Fuzzy multiobjective and multilevel optimization. In Ehrgott M., Gandibleux, X. (eds) Multiple criteria optimization: state of the art annotated bibliographic survey, Kluwer, The Netherlands.

Schniederjans, M.J. (1984), Linear goal programming.Petrocelli Books, Princeton, NJ

Schniederjans, M. (1995), Goal programming: methodology and applications. Kluwer, Boston, MA

Shin, W.S. andRavindran, A. (1991), Interactive multiobjective optimization survey I: continuous case. Computers & Operations Research 18: 97-114

Shor,N..1970), Utilization of the operation of space dilation in the minimization of convex functions. Kibernetika 1: 6-12 (in Russian), translated into English in the same year in Cybernetics 6: 7-15

Sierksma, G.(2002), Linear and integer programming: theory and practice (2nd ed.) Marcel Dekker, New York, NY

Simmons, D.M. (1972), Linear programming for operations research, Holden-Day, San Francisco, CA

Simonnard, M. (1966), Linear programming. Prentice-Hall, Englewood Cliffs, NJ

Simon, H.A. (1957), Models of man. Wiley, Yew York, NY

Stadler, W. (1984), A comprehensive bibliography on multicriteria decision making.

Steuer, R.E. (1986), Multiple criteria optimization: theory, computation, and application. Wiley, New York, NY

Steuer, R.E. (1995), The ADBASE multiple objective linear programming package. In Gu J, Chen J, Wei Q, Wang S (eds), Multiple criteria decision making, SCI-TECH, Windsor, England

Stigler, G. (1945), The cost of subsistence. Journal of Farm Economics 25: 303-314

Strassen, V. (1969), Gaussian elimination is not optimal. NumerischeMathematik13: 354-356

Sun et al. (1996), Solving multiple objective programming problems using feed-forward artificial neural networks: The interactive FFANN procedure. Management Science 42: 835-849.

Swarc, W. (1971), The transportation paradox. Naval Research Logistics Quarterly 18: 185-202

Tanino et al. (2003), Multi-objective programming and goal programming: theory and applications. Advances in soft computing 21, Springer, Berlin Heidelberg

Trzaskalik, T. andMichnik, J.(2002),Multipleobjective and goal programming. Advances in soft computing 12, Springer, Berlin Heidelberg

Vanderbei, R.J. (2001), Linear programming: Foundations and Extensions. 2nd edn.Kluwer, Boston, MA.

Vanderbei, R.J. and Hillier, F. (2007), Linear Programming: Foundations and Extensions, 3rd ed., Springer Science + Business Media, LLC 223, Spring Street, New York.

Vasant et al.(2005), Fuzzy linear programming with vague objective coefficients in an uncertain environment. Journal of the OperationalResearch Society 56: 597-603

Vicente, L.N. (2001), Bilevel programming: introduction, history, and overview. In: Floudas, C.A. andPardalos, P.M (eds) Encyclopedia of optimization. Springer, New York

Vicente, L.N. andCalamai, P.H. (1994),Bilevel and multilevel programming: A bibliographic review. Journal of Global Optimization 5: 291-306

von Neumann, J. (1947), On a maximization problem (manuscript). Institute for Advanced Study, Princeton, NJ, Nov 1947

Wright, S.J. (1997), Primal Dual – Interior Point Methods. Society for Industrail and Applied Mathematics Publications, University City Science Center, Philadelpia.

Zadeh, M. (1963), Optimality and non-scalar-valued performance criteria. IEEE Transactions on Automatic Control AC 8: 59-60

Zadeh, L.A. (1979), Fuzzy sets. In: Holzman AG (ed) Operations research methodology. Marcel Dekker, New York Basel, pp 569-606

Zeleny, M. (1973), Compromise programming. In: Cochrane JL, Zeleny M (eds), Multiple criteria decision making. University of South Carolina Press, Columbia, SC, 262-301

Zhang, S.(1991),On anti-cycling rules for the simplex method. Operations Research Letters 10: 189-192

Zimmermann, H.J. (1976), Description and optimization of fuzzy systems. International Journal of General Systems 2: 209-215

Zimmermann, H.J. (1978), Fuzzy programming and linear programming with several objectives. Fuzzy Sets and Systems 1: 45-55

Zionts, S. (1969), Thecriss-cross method for solving linear programming problems. Management Science 15: 426-445

Zykina, A.V. (2004), A generalized solution of an interactive goal programming problem. Cybernetics and Systems Analysis 40: 277-283

APPENDIX

INITIAL SOLUTION OF THE REVISED SIMPLEX METHOD IN OF EXAMPLE

Maximize $Z = 0.288x_1 + 0.240x_2 + 0.280x_3 + 0.360x_4 + 0.340x_5 + 0s_1 + 0s_2 + 0s_3 + 0s_4$

$+0s_6 + 0s_7 + 0s_8$

Subject to

$$x_1 + x_2 + x_3 + x_4 + x_5 + s_1 = 200{,}000$$

$$x_4 + x_5 + s_2 = 40000$$

$$x_2 + x_5 + s_3 = 80{,}000$$

$$x_2 + x_3 + s_4 = 100000$$

$$0.4x_1 - x_3 + 0.4x_4 + 0.4x_5 + s_5 = 0$$

$$x_2 + s_6 = 1200$$

$$x_3 + x_4 + s_7 = 20000$$

$$-0.02x_1 + 0.19x_2 - 0.04x_3 + 0.04x_4 + 0.09x_5 + s_8 = 0$$

COAMPUTATIONAL PROCEDURES

Impute parameters

$$A = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 \\ 0.4 & 0 & -1 & 0.4 & 0.4 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ -0.02 & 0.19 & -0.04 & 0.04 & 0.09 \end{bmatrix}, \quad b = \begin{bmatrix} 200000 \\ 40000 \\ 80000 \\ 100000 \\ 0 \\ 1200 \\ 20000 \\ 0 \end{bmatrix}$$

Iteration 0

(i) Basic variables in iteration 0, BV(0) $= \{s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8\}$

Nonbasic variables in iteration 0, NBV(0) = $\{x_1, x_2, x_3, x_4, x_5\}$

$$B_o = \mathrm{I} = B_o^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad CB_V = [0\,0\,0\,0\,0\,0\,0\,0]$$

$$CB_V B^{-1} = [0\,0\,0\,0\,0\,0\,0\,0] \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} = [0\,0\,0\,0\,0\,0\,0\,0]$$

(ii) From $CB_V B^{-1} a_j - C_j$,

$$\text{Coefficient of } x_1 \text{ in } z-\text{row} = [0\,0\,0\,0\,0\,0\,0\,0] \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0.4 \\ 0 \\ 0 \\ -0.02 \end{bmatrix} - 0.288 = -0.288.$$

$$\text{Coefficient of } x_2 \text{ in z-row} = [0\,0\,0\,0\,0\,0\,0\,0]\begin{bmatrix}1\\0\\1\\1\\0\\0\\0\\0.19\end{bmatrix} - 0.240 = -0.240.$$

$$\text{Coefficient of } x_3 \text{ in z-row} = [0\,0\,0\,0\,0\,0\,0\,0]\begin{bmatrix}1\\0\\0\\1\\-1\\0\\1\\-0.04\end{bmatrix} - 0.284 = -0.284.$$

$$\text{Coefficient of } x_4 \text{ in z-row} = [0\,0\,0\,0\,0\,0\,0\,0]\begin{bmatrix}1\\1\\0\\0\\0.4\\0\\1\\0.04\end{bmatrix} - 0.360 = -0.360$$

$$\text{Coefficient of } x_5 \text{ in z-row} = [0\,0\,0\,0\,0\,0\,0\,0]\begin{bmatrix}1\\1\\1\\0\\0.4\\0\\0\\0.09\end{bmatrix} - 0.340 = -0.34$$

Since x_4 has the most negative coefficient in z –row, it enters basis.

(iii) Using ratio test to determine the leaving variable.

$$
\text{Column of } x_4 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0.4 \\ 0 \\ 1 \\ 0.04 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0.4 \\ 0 \\ 1 \\ 0.04 \end{bmatrix}
$$

$$
B^{-1}\text{b} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 200000 \\ 40000 \\ 80000 \\ 100000 \\ 0 \\ 1200 \\ 20000 \\ 0 \end{bmatrix} = \begin{bmatrix} 200000 \\ 40000 \\ 80000 \\ 100000 \\ 0 \\ 1200 \\ 20000 \\ 0 \end{bmatrix}
$$

Row1 : $\frac{200000}{1}$ = 200000, row2: -, row 3:-, row3: -, row4: -, row 5: -, row 6: -, row7: $\frac{20000}{1}$ = 20000, row 8: -.

Since row 7 has the smallest ratio, s_7 leaves basis and x_4 enters basis.

(iv) The new basic variables are $\{s_1, s_2, s_3, s_4, s_5, s_6, x_4, s_8\}$

New nonbasic variables are $\{x_1, x_2, x_3, s_7, x_5\}$.

$$
B_{new} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0.4 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.04 & 1 \end{bmatrix}, \quad B_{new}^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -0.4 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.04 & 1 \end{bmatrix}.
$$

CB_V = [0 0 0 0 0 0 0.36 0],

$$B_{new}^{-1}\mathrm{b} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -0.4 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.04 & 1 \end{bmatrix} \begin{bmatrix} 200000 \\ 40000 \\ 80000 \\ 100000 \\ 0 \\ 1200 \\ 20000 \\ 0 \end{bmatrix} = \begin{bmatrix} 180000 \\ 40000 \\ 80000 \\ 100000 \\ -8000 \\ 1200 \\ 2000 \\ -888 \end{bmatrix}$$

(v) From $CB_V B^{-1}$b, the objective function value Z after first iteration is:

$$[0\ 0\ 0\ 0\ 0\ 0\ 0.36\ 0] \begin{bmatrix} 180000 \\ 40000 \\ 80000 \\ 100000 \\ -8000 \\ 1200 \\ 2000 \\ -888 \end{bmatrix} = 7200$$ and $x_4 = 2000$, $s_1 = 180000$, $s_2 = 40000$, $s_3 = 80000$,

$s_4 = 100000$, $s_5 = -8000$, $s_6 = 1200$, $s_8 = -800$.