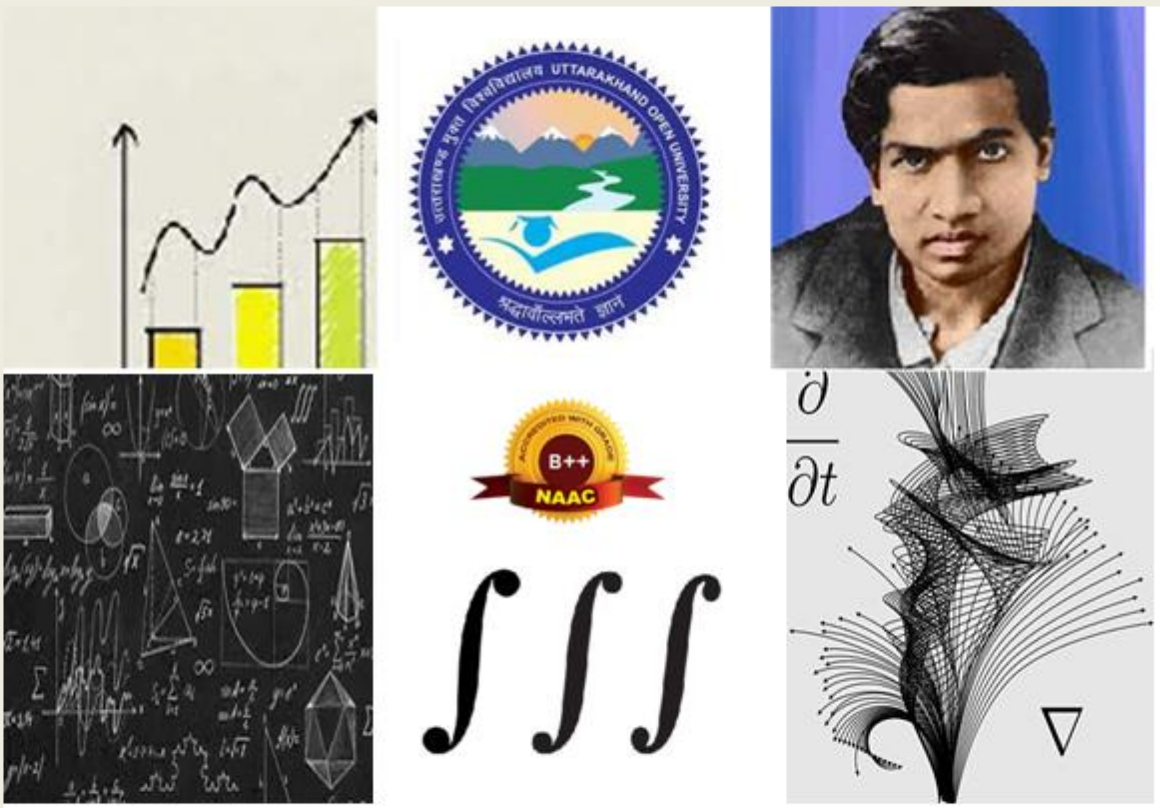
Master of Science
(THIRD SEMESTER)

MAT 603
OPERATIONS RESERACH



DEPARTMENT OF MATHEMATICS
SCHOOL OF SCIENCES
UTTARAKHAND OPEN UNIVERSITY
HALDWANI, UTTARAKHAND
263139

COURSE NAME: OPERATIONS RESEARCH

COURSE CODE: MAT603

Department of Mathematics
School of Science
Uttarakhand Open University
Haldwani, Uttarakhand, India,
263139

BOARD OF STUDIES-FEBRUARY 2023

Editor

Dr. Jyoti Rani
Assistant Professor
Department of Mathematics,
School of Sciences, Uttarakhand Open University

Unit Writer	Block	Unit
Dr. Kamlesh Bisht Assistant Professor (AC) Department of Mathematics Uttarakhand Open University	**I, II, III, IV**	**01, 02, 03, 04, 05, 06, 07, 08, 09, 10, 11, 12, 13 & 14**

Course Title and Code : OPERATIONS RESEARCH (MAT 603)

ISBN :

Edition : 2024

Contents

Course: Operations Research **Course code: MAT603**
Credit: 4

<u>COURSE INFORMATION</u>

The present self-learning material **"Operations Research"** has been designed for M.Sc. (Third Semester) learners of Uttarkhand Open University, Haldwani. This self learning material is writing for increase learner access to high-quality learning materials. This course is divided into 14 units of study. The first block is devoted to introduction of linear programming problem (LPP) and various methods to solve LPP and their application in operational research. Unit 6, Unit 7 and Unit 8 are focussed on the real life problems in LPP. The aim of Unit 9, 10 and 11 are to introduce the non-linear programing problem with various applications like assignment problem, quadratic programming etc. Unit 12 and Unit 13 are based on dynamical programming problems and also include two important methods Wolfe's modified Simplex Methods & Bell's Method which are essential to solved the dynamical programming problem. Unit 14 will explain the goal programming problem. This material also used for competitive examinations. The basic principles and theory have been explained in a simple, concise and lucid manner. Adequate number of illustrative examples and exercises have also been included to enable the leaners to grasp the subject easily.

SYLLABUS

OPERATIONS RESEARCH

Introduction to Linear Programming, Linear Programming Problems and Mathematical Formulation, Graphical Solution, Lines and Hyper Plane, Convex Set, Extreme Points of convex set, Convex Combination of vector Convex Hull. Basic solution and basic feasible solution. Fundamental Theorem of Linear Programming, Simplex Method, Big-M Method, Two Phase Method, Degeneracy, Concept of Duality, Dual Simplex Method, Revised Simplex Method.

Sensitivity Analysis, Integer Linear Programming, Branch and Bound Method, Travelling Salesman Method, Transportation, Test of Optimality, Degeneracy in Transportation Problem, Balanced and Unbalanced Transportation Problem, North West Corner method, Vogel's approximation method, Transhipment Problem.

Dynamical Programming, Decision Tree and Bellman's Principle of Optimality, Decomposition, Non Linear Programming, Quadratic Programming, Kuhn-Tucker Condition, Dynamic Programming. Quadratic Programming, Bell's Method. Goal Programming, Assignment Problem.

BLOCK- I

LINEAR PROGRAMMING

UNIT-1: INTRODUCTION TO LINEAR PROGRAMMING & OPERATION RESEARCH

CONTENTS:

1.1 *INTRODUCTION*

It's fascinating how linear programming has found applications across such a wide array of industries, from finance to manufacturing to transportation. The flexibility of LP in optimizing resource allocation makes it a versatile tool indeed. The simplex algorithm, in particular, revolutionized the field by providing an efficient method for solving these problems.

In essence, linear programming involves optimizing a linear objective function subject to linear constraints. The graphical method provides an intuitive way to visualize and solve two-variable

LP problems, while the simplex algorithm extends this to problems with any number of variables.

Moreover, the concept of duality in LP is quite powerful. It allows us to look at the same problem from different perspectives, providing valuable insights and sometimes simplifying the solution process. The dual simplex and revised simplex methods further enhance the toolbox for solving LP problems efficiently.

Overall, the mathematical foundation and practical applications of linear programming make it an indispensable tool in various domains, contributing to more efficient resource utilization and decision-making processes.

Linear programming (LP) is widely used for optimizing specific types of problems. In 1947, George Bernard Dantzig developed the simplex algorithm, a highly effective method for solving linear programming problems (LPP). Since then, LP has been applied in diverse industries such as banking, education, forestry, petroleum, manufacturing, and trucking. The primary challenge in these fields often involves distributing limited resources among various activities in the most optimal manner. Real-world scenarios where LP is applicable vary widely, including everything from assigning production facilities to products to allocating national resources for domestic needs, from portfolio selection to determining shipping patterns, and beyond. This unit will cover the mathematical formulation of LPP, the graphical method for solving two-variable LPP, as well as the simplex algorithm, duality, dual simplex, and revised simplex methods for solving LPP with any number of variables.

George Bernard Dantzig, born on November 8, 1914, and passing away on May 13, 2005, was an American mathematical scientist renowned for his contributions to a range of fields including industrial engineering, operations research, computer science, economics, and statistics.

His most notable achievement is the development of the simplex algorithm, a groundbreaking method for solving linear programming problems. Dantzig's work in linear programming has had a profound impact across numerous industries and disciplines.

In addition to his work in optimization, Dantzig made significant contributions to statistics. Interestingly, he famously solved two open problems in statistical theory, mistaking them for homework after arriving late to a lecture by Jerzy Neyman. At the time of his passing, Dantzig held the prestigious positions of Professor Emeritus of Transportation Sciences and Professor of Operations Research and Computer Science at Stanford University.

George Bernard Dantzig
(8 November 1914 – 13 May 2005)
https://en.wikipedia.org/wiki/George_Dantzig

1.2 *OBJECTIVE*

After reading this unit learners will be able to

- Understand the basic concept of mathematical formulation of LPP.
- Visualized the concept of general linear programming problem.
- Implementation of graphical method.

1.3 *MATHEMATICAL FORMULATION OF LPP*

There are some basic components of an LPP:

Basic variables: Decision variables are the amounts that must be found in order to solve the LPP.

Objective function: The objective function, in optimization, is a linear function of the decision variables that you aim to either maximize or minimize.

Constraints: Constraints are factors such as labor, machinery, raw materials, space, and finances that act as limitations or restrictions, affecting the extent to which an objective can be achieved.

Sign restriction: When a decision variable x_i is limited to nonnegative values, we express this constraint as $x_i \geq 0$. If a variable x_i can take on positive, negative, or zero values, we refer to it as being unrestricted in sign.

A linear programming problem (LPP) involves optimizing a linear objective function subject to a set of constraints. Here are the key components:

(i) The objective function is either to be maximized or minimized and is linear.
(ii) The decision variables must satisfy to a set of constraints, each of which is expressed as a linear equation or inequality.
(iii) Each decision variable has a sign restriction associated with it.

Within linear programming, two fundamental concepts are the feasible region and the optimal solution.

Feasible Region: The feasible region encompasses all the points in the decision variable space that satisfy the constraints and sign restrictions.

Optimal Solution: The optimal solution represents the point within this feasible region where the objective function achieves its maximum (or minimum) value.

OR

An optimal solution in linear programming refers to a point within the feasible region that either maximizes or minimizes the objective function, depending on whether it's a maximization or minimization problem. For maximization problems, the optimal solution corresponds to the point with the highest value of the objective function within the feasible region. Conversely, for minimization problems, the optimal solution is the point with the lowest value of the objective function within the feasible region.

1.4 *GENERAL LINEAR PROGRAMMING PROBLEM*

Mathematically general linear programming can be represented as follows:

Maximize (Or Minimize) Z= $c_1x_1 + c_2x_2 + ... + c_nx_n$

Subject to,

$a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + ... + a_{1j}x_j + ... + a_{1n}x_n \ (\leq, =, \geq) b_1$

$a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + ... + a_{2j}x_j + ... + a_{2n}x_n \ (\leq, =, \geq) b_2$

…………………………………………………………

$a_{i1}x_1 + a_{i2}x_2 + a_{i3}x_3 + ... + a_{ij}x_j + ... + a_{in}x_n \ (\leq, =, \geq) b_i$

…………………………………………………………

$a_{m1}x_1 + a_{m2}x_2 + a_{m3}x_3 + ... + a_{mj}x_j + ... + a_{mn}x_n \ (\leq, =, \geq) b_m$

And $x_1, x_2, x_3, ..., x_n \geq 0$

The above programming problem can be rewritten in compact form as,

Maximize (Or Minimize) Z= $\sum_{j=1}^{n} c_j x_j$ …………….. (1)

Subject to,

$\sum_{j=1}^{n} a_{ij} x_j \ (\leq, =, \geq)\ b_i;\ i = 1,2,...,m$ ………………. (2)

$x_j \geq 0;\ j = 1,2,...,n$ ………………. (3)

The objective is to determine the values of x_j that optimize (maximize or minimize) the objective function (1). These values must satisfy to the constraints (2) as well as non-negativity restrictions (3). In this context, the coefficients c_j are termed as cost coefficients, while a_{ij} represents technological coefficients; a_{ij} denotes the quantity of the i^{th} resource utilized per unit of variable *xj*, and *bi* signifies the overall availability of the i^{th} resource.

Example 1: An oil company possesses two refineries - refinery A and refinery B. Refinery A can produce 20 barrels of petrol and 25 barrels of diesel daily, while refinery B can produce 40 barrels of petrol and 20 barrels of diesel per day. The company has a minimum requirement of 1000 barrels of petrol and 800 barrels of diesel. Operating refinery A costs Rs. 300 per day and refinery B costs Rs. 500 per day. How many days should each refinery be operated to minimize costs? Formulate this scenario as a linear programming model.

Solution: To formulate this problem as a linear programming model, let's define our decision variables:

Let x be the number of days refinery A is operated.

Let y be the number of days refinery B is operated.

Our objective is to minimize costs, so we want to minimize the total operating cost:

Minimize: $300x+500y$

Subject to the constraints:

1. Refinery A produces 20 barrels of petrol per day, and refinery B produces 40 barrels. The total petrol production should be at least 1000 barrels: $20x+40y\geq 1000$

2. Refinery A produces 25 barrels of diesel per day, and refinery B produces 20 barrels. The total diesel production should be at least 800 barrels: $25x+20y\geq 800$

3. Non-negativity constraints: $x\geq 0$, $y\geq 0$

This linear programming model represents the problem of minimizing costs while meeting the production requirements for petrol and diesel.

OR

Minimize $Z = 300x + 500y$

Subject to,

$$20x + 40y \geq 1000$$

$$25x + 20y \geq 800$$

$$x, y \geq 0$$

Example 2: In a particular factory, three machines, namely M_1, M_2, and M_3, are utilized in the manufacturing process of two products, P_1 and P_2. Machine M_1 is occupied for 5 minutes for producing one unit of P_1, while M_2 is used for 3 minutes and M_3 for 4 minutes. For one unit of P_2, the time requirements are 1 minute for M_1, 4 minutes for M_2, and 3 minutes for M_3. The profit earned per unit is Rs. 30 for P_1 and Rs. 20 for P_2, regardless of whether the machines operate at full capacity. How can we determine the production plan that maximizes profit? Frame this problem as a linear programming challenge.

Solution: To formulate this problem as a linear programming problem, let's define our decision variables:

Let x_1 be the number of units of product P_1 produced. Let x_2 be the number of units of product P_2 produced.

Our objective is to maximize profit, so we want to maximize the total profit:

Maximize: $30x_1+20x_2$

Subject to the constraints:

1. Time constraint for machine M_1: $15x_1+x_2 \le T_1$
Where T_1 is the total available time on machine M_1.
2. Time constraint for machine M_2: $3x_1+4x_2 \le T_2$
Where T_2 is the total available time on machine M_2.
3. Time constraint for machine M_3: $4x_1+3x_2 \le T_3$
Where T_3 is the total available time on machine M_3.
4. Non-negativity constraints: $x_1 \ge 0$, $x_2 \ge 0$

Note: Here, we can take total available time for all machines is 60 i.e., $T_1 = T_2 = T_3 = 60$

This linear programming model represents the problem of determining the production plan that yields the highest profit while considering the time constraints on each machine.

OR

Maximize $Z = 30x_1 + 20x_2$

Subject to,

$$5x_1 + x_2 \le 60$$

$$3x_1 + 4x_2 \le 60$$

$$4x_1 + 3x_2 \le 60$$

$$x_1, x_2 \geq 0$$

1.5 *LP SOLUTION*

First, we will learn some terminologies for solution.

Closed half plane: A linear inequality in two variables is known as a half plane. The corresponding equality or the line is known as the boundary of the half plane. The half plane along with its boundary is called a closed half plane.

In the context of linear inequalities in two variables, a half plane represents the region of the coordinate plane that satisfies the inequality. The boundary of this region is defined by the corresponding equality or line. When considering both the boundary and the region itself, it's termed as a closed half plane. This closed half plane includes the boundary line and all the points on one side of it.

Convex set: A set is convex if, for any two points within the set, the line segment connecting those points remains entirely within the set. This property holds true for all pairs of points in the set, making it a fundamental characteristic of convexity. Mathematically, A set S is said to be convex set if for all $x, y \in S$, $\lambda x + (1-\lambda) y \in S \, \forall \lambda \in [0,1]$.

For example, the set $S = \{(x, y) : 3x + 2y \leq 12\}$ is convex because for two points (x_1, y_1) and $(x_2, y_2) \in S$, it is easy to see that $\lambda(x_1, y_1) + (1-\lambda)(x_2, y_2) \in S \, \forall \lambda \in [0,1]$. While the set, $(S = \{(x, y) : x^2 + y^2 \geq 16\}$ is not convex. Note that two points (4,0) and (0,4) $\in S$ but $\lambda(4,0) + (i-\lambda)(0,4) \notin S$ for $\lambda = 1/2$.

Convex polygon: A convex polygon is indeed a convex set formed by the intersection of a finite number of closed half planes. Each side of the polygon corresponds to a boundary line of a half plane, and the polygon itself includes all the points within its boundaries. This property ensures that the polygon is convex, meaning that any line segment connecting two points within the polygon lies entirely within it.

Extreme points: The extreme points of a convex polygon are precisely the points where the lines that bound the feasible region intersect. These points are crucial because any point within the polygon can be expressed as a convex combination of the extreme points. Thus, they play a fundamental role in characterizing the polygon's shape and properties.

Feasible solution (FS): A feasible solution in optimization refers to any solution that meets all the constraints of the problem while maintaining non-negative values for the decision variables. It's essentially a valid solution that adheres to the problem's requirements.

OR

A feasible solution to the problem is any non-negative solution that complies with every restriction.

Basic solution (BS): In linear programming, particularly when dealing with a set of m simultaneous equations in n variables (where $n > m$), a basic solution is obtained by setting (n - m) variables equal to zero and then solving the resulting system of equations for the remaining m variables. These m variables are referred to as basic variables, while the (n - m) variables set to zero are called non-basic variables. Basic solutions play a crucial role in optimization algorithms such as the simplex method.

Basic feasible solution (BFS): A basic solution to a linear programming problem is termed a basic feasible solution (BFS) if it satisfies all the non-negativity constraints.

Furthermore, a BFS is classified as degenerate if at least one of the basic variables has a value of zero. Conversely, a BFS is considered non-degenerate if all of the basic variables have non-zero and positive values.

These distinctions are significant in understanding the behavior of optimization algorithms such as the simplex method.

Optimal basic feasible solution: An optimal basic feasible solution in linear programming is a basic feasible solution that optimizes (maximizes or minimizes) the objective function. It represents the best feasible solution among all basic feasible solutions in terms of achieving the highest (or lowest) objective function value.

In linear programming, the optimal value of the objective function occurs at one of the extreme points of the convex polygon formed by the set of feasible solutions of the linear programming problem (LPP). This property is fundamental and is exploited in optimization algorithms such as the simplex method to efficiently find the optimal solution. By examining the extreme points, we can determine the best feasible solution that maximizes or minimizes the objective function.

Unbounded Solution: An LPP is said to have an unbounded solution if its solution can grow infinitely large without violating any of the constraints. This means that there is no finite optimal solution, and the objective function can be increased (in case of maximization) or decreased (in case of minimization) indefinitely while still satisfying all the constraints.

1.6 *SOME IMPORTANT RESULTS*

Now we will discuss some important theorems and the results which are required to solve LPP.

Theorem 1: A hyperplane is a convex set.

Proof: Consider the hyperplane $S = \{x : cx = z\}$. Let x_1 and x_2 be two points in *S*. Then $cx_1 = z$ and $cx_2 = z$. Now, let a point x_3 be given by the convex combination of x_1 and x_2 as

$x_3 = \lambda x_1 + (1-\lambda)x_2, 0 \le \lambda \le 1$. Then

$$cx_3 = c\{\lambda x_1 + (1-\lambda)x_2\}$$
$$= c\lambda x_1 + (1-\lambda)x_2$$
$$= cz + (1-\lambda)z = z$$

Therefore, x_3 satisfies $cx = z$ and hence $x_3 \in S$. x_3 being the convex combination of x_1 and x_2 in *S*, *S* is a convex set. Thus a hyperplane is a convex set.

(Proof by another way) A hyperplane is typically defined as an affine subspace of dimension *n*−1 in an *n*-dimensional vector space.

To prove that a hyperplane is a convex set, we need to show that for any two points $\mathbf{x}_1$ and $\mathbf{x}_2$ in the hyperplane, the line segment connecting them lies entirely within the hyperplane.

Consider the equation of a hyperplane in *n*-dimensional space:

$\mathbf{a}^T\mathbf{x} = b$

where **a** is a non-zero vector normal to the hyperplane, **x** is a point in the hyperplane, and *b* is a scalar constant.

Now, let $\mathbf{x}_1$ and $\mathbf{x}_2$ be two arbitrary points in the hyperplane, satisfying:

$$\mathbf{a}^T\mathbf{x}_1 = b$$

$$\mathbf{a}^T\mathbf{x}_2 = b$$

Consider any point **x** on the line segment connecting **x**1 and **x**2. This point can be expressed as:

$\mathbf{x} = t\,\mathbf{x}_1 + (1-t)\mathbf{x}_2$

where, $0 \le t \le 1$.

Now, let's compute the dot product of **a** with **x**:

$\mathbf{a}^T\mathbf{x} = \mathbf{a}^T(t\mathbf{x}_1 + (1-t)\mathbf{x}_2)$

$= t\mathbf{a}^T\mathbf{x}_1 + (1-t)\mathbf{a}^T\mathbf{x}_2 = tb + (1-t)b = b$

Thus, $\mathbf{a}^T\mathbf{x}=b$, showing that $\mathbf{x}$ also lies on the hyperplane. Since this is true for any $\mathbf{x}$ on the line segment connecting $\mathbf{x}_1$ and $\mathbf{x}_2$, the entire line segment lies within the hyperplane.

Therefore, a hyperplane is indeed a convex set.

Theorem 2: Intersection of two convex sets is also a convex set.

Proof: The intersection of two convex sets is indeed a convex set. This property is a fundamental result in convex geometry and is known as the "intersection theorem" or "convexity preserving property of intersections."

To prove this, let's suppose we have two convex sets A and B in some vector space. We want to show that their intersection, denoted by $A \cap B$, is also convex.

Let x, y be any two points in $A \cap B$, and let λ be a scalar such that $0 \leq \lambda \leq 1$.

Since x and y belong to $A \cap B$, they must belong to both A and B. Because A and B are convex sets, the line segment connecting x and y lies entirely within both A and B.

Since A is convex, $\lambda x+(1-\lambda)y$ lies in A. Similarly, since B is convex, $\lambda x + (1-\lambda)y$ lies in B. Therefore, $\lambda x+(1-\lambda)y$ lies in both A and B, which means it lies in their intersection $A \cap B$.

Thus, $A \cap B$ is convex, as any point on the line segment between any two points in $A \cap B$ also lies within $A \cap B$.

Theorem 3: The set of all feasible solutions of an LPP is a convex set.

Proof: The set of all feasible solutions of a Linear Programming Problem (LPP) forms a convex set.

An LPP typically involves optimizing a linear objective function subject to linear constraints. The feasible region, which is the set of all points that satisfy these constraints, is typically a convex set.

To see why, consider the constraints of an LPP:

$$\mathbf{Ax} \leq \mathbf{b}$$

where $\mathbf{A}$ is a matrix of coefficients, $\mathbf{x}$ is the vector of decision variables, and $\mathbf{b}$ is a vector of constants.

Each constraint $\mathbf{a}_i^T\mathbf{x} \leq b_i$ represents a half-space in n-dimensional space, defined by a hyperplane. The intersection of all these half-spaces forms the feasible region.

Since each constraint defines a convex set (a half-space is convex), the intersection of convex sets (feasible region) is also convex. This means that any convex combination of feasible solutions remains feasible, ensuring the convexity of the feasible set.

Therefore, the set of all feasible solutions of an LPP is indeed a convex set. This property is crucial for the efficient solution of linear programming problems using convex optimization techniques.

Remark: An LPP has an infinite number of feasible solutions if it has two feasible solutions, since a feasible solution might be any convex combination of the two feasible solutions.

Theorem 4: The collection of all feasible solutions of an LPP constitutes a convex set whose extreme points correspond to the basic feasible solutions.

Proof: Let's break down the proof into two parts:

1. **The Feasible Region is Convex**: To prove that the collection of all feasible solutions of an LPP constitutes a convex set, we need to show that any convex combination of two feasible solutions is also a feasible solution.

Let $\mathbf{x}_1$ and $\mathbf{x}_2$ be two feasible solutions, meaning they satisfy all the constraints of the linear programming problem. Now, consider the convex combination:

$\mathbf{x}=\lambda\mathbf{x}_1+(1-\lambda)\mathbf{x}_2$

where $0 \leq\lambda \leq 1$.

Since $\mathbf{x}_1$ and $\mathbf{x}_2$ satisfy the constraints, it follows that:

$\mathbf{Ax}_1 \leq \mathbf{b}$

$\mathbf{Ax}_2 \leq \mathbf{b}$

Multiplying these inequalities by λ and $1-\lambda$ respectively and summing them, we get:

$\lambda(\mathbf{Ax}_1)+(1-\lambda)(\mathbf{Ax}_2) \leq \lambda\mathbf{b}+(1-\lambda)\mathbf{b}$

Simplifying:

$\mathbf{A}(\lambda\mathbf{x}_1+(1-\lambda)\mathbf{x}_2) \leq \mathbf{b}$

Thus, $\mathbf{x}$ also satisfies the constraints, making it a feasible solution. Since this holds for any λ in the range $0\leq \lambda \leq 1$, the feasible region is convex.

2. **Extreme Points Correspond to Basic Feasible Solutions**: To prove that extreme points of the feasible region correspond to basic feasible solutions, we need to show that each extreme point is indeed a basic feasible solution, and conversely, every basic feasible solution is an extreme point.

- **Extreme Points as Basic Feasible Solutions**: Any convex combination of two distinct feasible solutions lies strictly within the line segment connecting those two solutions. Since an extreme point cannot be expressed as such a convex combination of two distinct points, it must satisfy a minimal set of constraints, making it a basic feasible solution.

- **Basic Feasible Solutions as Extreme Points**: Basic feasible solutions are those solutions where a minimal set of constraints is active. If a solution is not a basic feasible solution, it means it can be expressed as a convex combination of two distinct basic feasible solutions. Therefore, it cannot be an extreme point.

Therefore, the extreme points of the feasible region correspond to the basic feasible solutions, completing the proof.

1.7 *GRAPHICAL METHOD*

The graphical method is indeed suitable for solving Linear Programming Problems (LPPs) with only two decision variables because it allows us to visualize the feasible region and the objective function contour lines on a two-dimensional graph. By graphically identifying the corner points of the feasible region and evaluating the objective function at these points, we can determine the optimal solution.

However, when dealing with three or more decision variables, graphical methods become impractical due to the difficulty of visualization. In such cases, the simplex method is commonly used. The simplex method is an iterative algorithm that systematically moves from one basic feasible solution to another along the edges of the feasible region until the optimal solution is reached. It's a powerful algorithm for solving linear programming problems of any size efficiently.

The simplex method will indeed be discussed further in the next section, as it provides a robust and efficient approach for solving LPPs with three or more variables.

Example 3: Solve the following LPP by graphical method.

$$\text{Minimize } Z = 20x_1 + 10x_2$$

$$\text{Subject to, } x_1 + 2x_2 \le 40$$

$$3x_1 + x_2 \ge 30$$

$$4x_1 + 3x_2 \ge 60$$

$$x_1, x_2 \ge 0$$

Solution: To solve this Linear Programming Problem (LPP) graphically, we'll start by plotting the feasible region defined by the given constraints and then find the optimal solution within this region.

Let's begin by plotting the constraint equations:

1. $x_1+2x_2 \leq 40$
2. $3x_1+x_2 \geq 30$
3. $4x_1+3x_2 \geq 60$

To plot these equations, we'll first find their intercepts on the axes.

For $x_1+2x_2=40$, intercepts are:

- When $x_1=0$, $2x_2=40 \Rightarrow x_2=20$
- When $x_2=0$, $x_1=40$

For $3x_1+x_2=30$, intercepts are:

- When $x_1=0$, $x_2=30$
- When $x_2=0$, $3x_1=30 \Rightarrow x_1=10$

For $4x_1+3x_2=60$, intercepts are:

- When $x_1=0$, $3x_2=60 \Rightarrow x_2=20$
- When $x_2=0$, $4x_1=60 \Rightarrow x_1=15$

Now, we'll plot these points and draw the lines connecting them.

Next, we'll shade the region that satisfies all the inequalities. Since we're minimizing $Z=20x_1+10x_2$, we're looking for the region where Z is the smallest.

Let's get to graphing it!

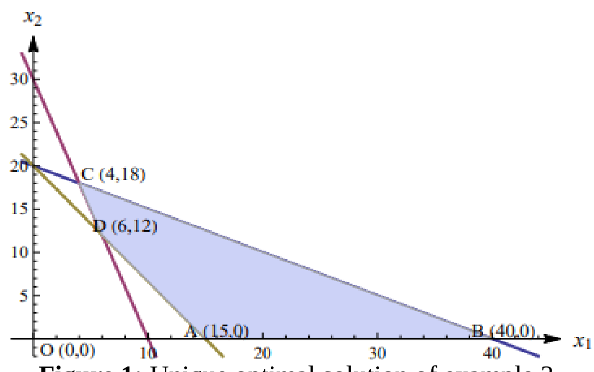


Figure 1: Unique optimal solution of example 3

Hence, the optimal solution of the shaded region is determined by the following table which shows the given LPP has minimum value is $Z_{min} = 240$ at the points $x_1 = 6$, $x_2 = 12$.

Extreme point	Objective function $Z = 20x_1 + 10x_2$
A (15,0)	300
B (40,0)	800
C (4,18)	260
D (6,12)	240

Example 4: Solve the following LPP by graphical method.

$$\text{Minimize } Z = 4x_1 + 3x_2$$

$$\text{Subject to, } x_1 + 2x_2 \leq 6$$

$$2x_1 + x_2 \leq 8$$

$$x_1 \geq 7$$

$$x_1, x_2 \geq 0$$

Solution: As seen in Figure 2, the limitations are plotted on the graph. There is no possible solution to the problem because there is no feasible region in the solution space.

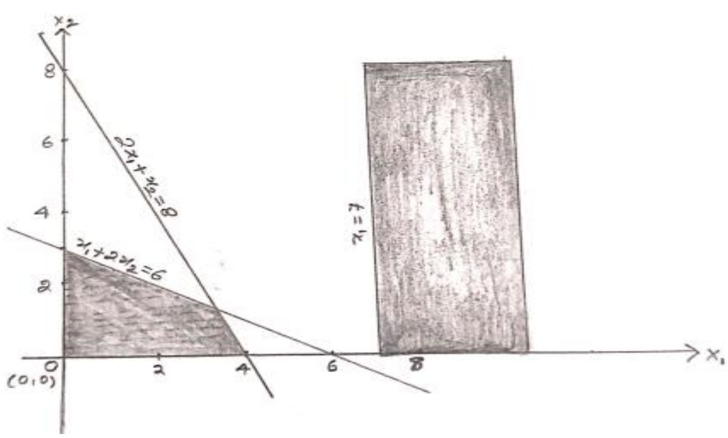


Figure 2: Feasible region of example 4

Example 5: Solve the following LPP by graphical method.

$$\text{Minimize } Z = 3x_1 + 5x_2$$

$$\text{Subject to, } x_1 + 2x_2 \geq 10$$

$$x_1 \geq 5$$

$$x_2 \leq 10$$

$$x_1, x_2 \geq 0$$

Solution: It is evident from the graph in Figure 3 that the feasible region is open-ended. As a result, Z's value can be increased indefinitely without going against any of the restrictions. Therefore, the LPP has an infinite solution.

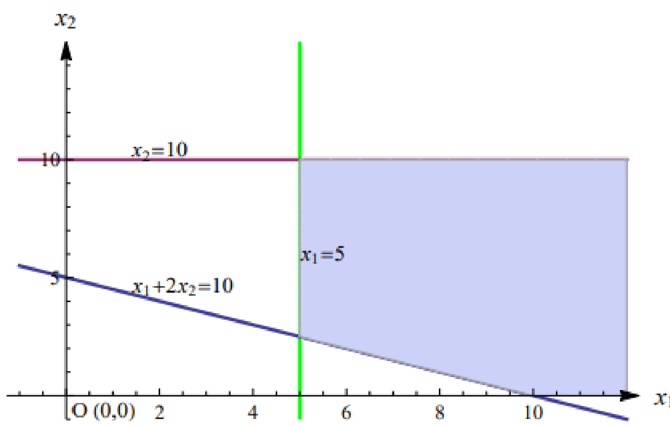


Figure 3: Unbounded solution of example 5

Note: It should be noted that an unbounded feasible zone does not always indicate the absence of a finite optimal solution for an LP problem. Examine the subsequent LPP, which although having an infinite feasible region, has an optimal viable solution.

$$\text{Minimize } Z = 2x_1 - x_2$$

$$\text{Subject to, } x_1 - x_2 \leq 1$$

$$x_1 \leq 3$$

$$x_1, x_2 \geq 0$$

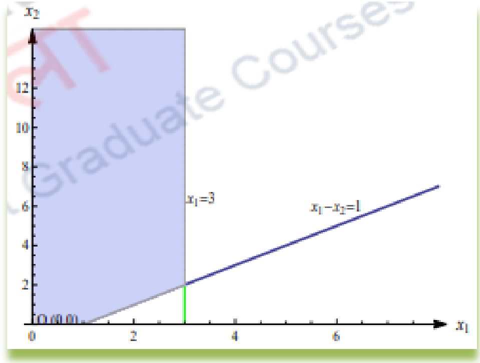


Figure 4: Finite optimal solution

Example 6: Solve the following LPP by graphical method.

$$\text{Minimize } Z = 3x_1 + 2x_2$$

$$\text{Subject to, } 6x_1 + 4x_2 \le 24$$

$$x_2 \ge 2$$

$$x_1 \le 3$$

$$x_1, x_2 \ge 0$$

Solution: As seen in Figure 5, the constraints are plotted on a graph by considering them as equations, and the feasible region is then identified using the signs of their inequality.

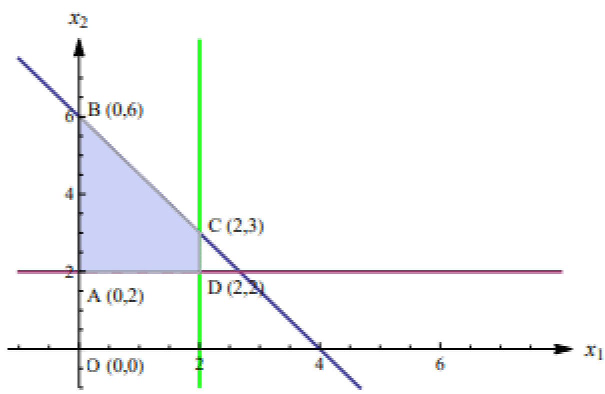


Figure 5: An infinite number of optimal solution of example 6

The extreme points of the region are A(0,2), B(0,6), C(2,3) and D(2,2). As we can easily find that slope of the objective function and one of the constraint $6x_1 + 4x_2 = 24$ coincide at line BC. Also from figure BC is the boundary line of the feasible region. So we can say that the optimal solution of LP problem can be obtained at any point of the line segment BC. From the following table

Corners (x, y)	Objective Function $Z = 3x_1 + 2x_2$
A (0,2)	4
B (0,6)	12
C (2,3)	12
D (2,2)	10

The optimal solution Z=12 is same at two different extreme points B and C. As a result, there exist several combinations of any two locations on the line segment BC that yield identical values for the objective function, thereby serving as optimal solutions for the linear programming problem. As a result, the provided LP issue has an endless number of optimal solutions.

Check your progress

Problem 1: Using the graphical method solve the following LPP

Minimize, $z = -x + 2y$

Subject to the constraint, $-x + 3y \leq 10$; $x + y \leq 6$; $x - y \leq 2$; $x, y \geq 0$

Answer: $x = 2, y = 0$ and minimum $z = -2$

Problem 2: Using the graphical method solve the following LPP

Minimize, $z = 2x + 3y$

Subject to the constraint, $x + y \leq 30$; $x - y \geq 0$; $y \geq 3$; $0 \leq x \leq 20, 0 \leq y \leq 12$

Answer: $x = 18, y = 12$ and maximum $z = 72$

1.7 *SUMMARY*

Linear Programming & Operations Research provides a comprehensive overview of two interconnected disciplines essential for optimizing decision-making processes. Linear programming offers a mathematical approach for resource allocation through the formulation and solution of linear optimization problems. Operations research, on the other hand, extends beyond linear programming to encompass a broader range of mathematical techniques aimed at addressing complex operational challenges across various industries. By exploring these fields, individuals gain valuable insights into modeling real-world problems and devising optimal solutions to enhance organizational efficiency and decision-making effectiveness in diverse domains such as manufacturing, logistics, finance, and healthcare. In this unit we have learned about the basic definitions of LPP, Feasible region, optimal solution, convex set, basic feasible solution, optimal feasible solution and more useful definitions used to solve the linear programming problem. The overall summarization of this units are as follows:

- A hyper plane is a convex set.
- Intersection of two convex sets is also a convex set.
- The set of all feasible solutions of an LPP is a convex set.
- The collection of all feasible solutions of an LPP constitutes a convex set whose extreme points correspond to the basic feasible solutions.

1.8 *GLOSSARY*

- Linear Programming Problem
- Feasible Region
- Optimal Solution
- Convex Set
- Basic Feasible Solution
- Optimal Basic Feasible Solution
- Graphical Method

1.9 *REFERENCES*

- Mokhtar S. Bazaraa, John J. Jarvis and Hanif D. Sherali: *Linear Programming and Network Flows* (4th edition). John Wiley and Sons, 2010.
- Hamdy A. Taha: *Operations Research: An Introduction (10th edition). Pearson*, 2017.
- Paul R. Thie and Gerard E. Keough: *An Introduction to Linear Programming and Game Theory* (3rd edition), Wiley India Pvt. Ltd, 2014.

- Kanti swarup, P. K. Gupta and Man Mohan: Introduction to Management Science "Operations Research", S. Chand & Sons, 2017.
- OpenAI. (2024). *ChatGPT (August 2024 version) [Large language model]*. OpenAI. https://www.openai.com/chatgpt

1.10 *SUGGESTED READING*

- G. Hadley, *Linear Programming*, Narosa Publishing House, 2002.
- Frederick S. Hillierand Gerald J. Lieberman: *Introduction to Operations Research* (10th edition). McGraw-Hill Education, 2015.
- https://epgp.inflibnet.ac.in/Home/ViewSubject?catid=ZLCHeZEhCZ8yCri36nSF3A==

1.11 *TERMINAL QUESTION*

Long Answer Type Question:

1: Solve graphically the following LPP and find maximum and minimum value of objective function:

Maximize (or minimize) $z = 5x + 3y$

Subject to: $x + y \leq 6;\ 2x + 3y \geq 3;\ 0 \leq x \geq 3;\ 0 \leq y \geq 3$

2: Solve graphically the following LPP and find maximum value of objective function:

Maximize $z = 5x + 3y$

Subject to: $x + y \leq 6;\ 2x + 3y \geq 6;\ 0 \leq x \leq 4;\ 0 \leq y \leq 3$

3: Solve graphically the following LPP and find maximum value of objective function:

Maximize (or minimize) $z = 3x + 2y$

Subject to: $-2x + y = 1;\ x \leq 2;\ x + y \leq 3;\ x, y \geq 0$

Short answer type question:

1: Solve graphically the following LPP and find maximum value of objective function:

Maximize $z = 2x + 4y$

Subject to: $x + 2y \le 5;\ x + y \le 4;\ x, y \ge 0$

2: Solve graphically the following LPP and find maximum value of objective function:

Maximize $z = 6x + y$

Subject to: $2x + y \ge 3;\ y - x \ge 0;\ x, y \ge 0$

Objective type question:

1: The graphical method of solving a linear programming problem is applicable when the number of decision variables is:

A) 1

B) 2

C) 3

D) Any number

2: In the graphical method, the feasible region is:

A) The entire plane

B) The area where all constraints overlap

C) The intersection of the objective function and one constraint

D) The area outside the constraints

3: The optimal solution to a linear programming problem using the graphical method is found:

A) At the center of the feasible region

B) At any point within the feasible region

C) At a corner point (vertex) of the feasible region

D) Along the boundary of the feasible region

4: If the feasible region is unbounded, the linear programming problem:

A) Has no solution

B) Always has an optimal solution

C) May have an optimal solution if the objective function is bounded

D) Will have an infinite number of solutions

5: In a maximization problem using the graphical method, the objective function line is shifted:

A) Parallel to itself towards the origin

B) Parallel to itself away from the origin

C) In any random direction

D) To the nearest constraint line

6: If two constraints intersect at a point in the feasible region, this point is called:

A) A feasible solution

B) An infeasible solution

C) A corner point

D) The optimal solution

7: In the graphical method, the area where no constraints overlap is called:

A) The feasible region

B) The infeasible region

C) The optimal region

D) The objective region

8: When solving a linear programming problem graphically, the constraints are represented by:

A) Straight lines

B) Curved lines

C) Dotted lines

D) Points

9: If the objective function is parallel to one of the constraints in the feasible region, then:

A) The problem has a unique solution

B) The problem has no solution

C) The problem has infinitely many solutions

D) The feasible region is empty

10: In a linear programming problem, the feasible region is bounded if:

A) The feasible region extends infinitely in one or more directions

B) The feasible region is a closed polygon

C) The feasible region lies entirely within the first quadrant

D) The objective function has a finite value

Fill in the blanks:

1: The graphical method of solving linear programming problems is only applicable when the number of decision variables is ___________.

2: In the graphical method, the ___________ region is the area where all constraints overlap.

3: The optimal solution in the graphical method is typically found at a __________ point of the feasible region.

4: The __________ function line is shifted parallel to itself in the graphical method to find the optimal solution.

5: If the feasible region is __________, the problem may have no finite optimal solution**.**

6: In the graphical method, each constraint is represented by a __________ on the graph.

7: If the objective function is __________ to one of the constraints in the feasible region, the problem may have infinitely many optimal solutions.

8: The area on the graph that does not satisfy all the constraints is called the __________ region.

9: The point of intersection of two or more constraints in the graphical method is called a __________ point.

10: In the graphical method, a linear programming problem is said to be __________ if the feasible region is a closed and bounded area.

1.12 *ANSWERS*

Answer of short answer type question

Answer 1: Maximum z = 10.

2: Problem has unbounded solution.

Answer of Long answer type question

Answer 1: $x = 3, y = 3$; Optimum $z = 24$

2: $x = 4, y = 2$; Optimum $z = 2$

3: $x = 2, y = 1$; Maximum $z = 8$

Answer of objective type question

Answer 1: B) **2:** B) **3:** C) **4:** C)

5:	B)	**6:**	C)	**7:**	B)	**8:**	A)
9:	C)	**10:**	B)				

Answer of fill in the question

Answer 1:	**2**	**2:**	Feasible	**3:**	Corner	**4:**	Objective
5:	unbounded	**6:**	straight line	**7:**	Parallel	**8:**	Infeasible
9:	Corner (or vertex)	**10:**	Bounded				

UNIT-2: SIMPLEX METHOD

CONTENTS:

2.1 *INTRODUCTION*

In linear programming, the simplex method is a basic strategy for addressing optimization problems. These problems usually involve a function and many constraints expressed as inequalities. The answer is usually found at one of the vertices of the polygonal region defined by the inequalities. A methodical process for evaluating the vertices as potential solutions is the simplex approach.

Drawing the restrictions on a graph can help solve certain straightforward optimization issues. Nevertheless, this approach is limited to two-variable systems of inequalities. An enormous number of extreme points can arise from the hundreds of equations with thousands of variables that are frequently involved in real-world problems. The simplex approach was developed in 1947 by George Dantzig, a mathematics adviser to the U.S. Air Force, to limit the number of extreme points that need to be checked. One of the most practical and effective algorithms ever created, the simplex approach is still the go-to technique for solving optimization issues on computers.

The approach first presumes that an extreme point is identified. (If no extreme point is provided, the Phase I form of the simplex approach is employed to identify one or to conclude that no workable solutions exist.) Next, a test establishes whether or not that extreme point is optimal using an algebraic formulation of the problem. An nearby extreme point is searched along an edge in the direction where the objective function's value increases at the fastest rate if the optimality test is unsuccessful. On occasion, it is possible to increase the objective function value without bound by moving along an edge. Should this happen, the process ends with a prescription of the boundary where the goal reaches positive infinity. If not, a new extreme point with an objective function value at least as high as the previous one is reached. Next, the described sequence is repeated. When the unbounded case arises or an ideal extreme point is discovered, termination takes place. In reality, the approach usually converges on the best solution in a number of steps that is just a tiny multiple of the number of extreme points, although in principle the number of steps required may grow exponentially with the number of extreme points.

2.2 *OBJECTIVE*

After reading this unit learners will be able to

- Various types of variables like slack and surplus variable.
- Visualized the canonical and standard forms of an LPP.
- Implementation of simplex method and visualized the algorithm to solve the given LPP by simplex method.

2.3 *SIMPLEX METHOD*

George Dantzig created the simplex approach in 1947 as an effective way to solve LP problems with many of variables. The graphical technique and the simplex method both involve examining the extreme points of the feasible region in order to get the best possible solution. In this case, the ideal solution is located at a multi-dimensional polyhedron's extreme point. The foundation of the simplex technique is the fact that, in the event that an ideal solution exists, it can always be found inside one of the most basic feasible options.

2.3.1 *CANONICAL AND STANDARD FORMS OF AN LPP*

Any linear programming problem is said to be in canonical form if it can be expressed as,

Maximize, $Z = c_1x_1 + c_2x_2 + ... + c_nx_n$

Subject to,

$a_{i1}x_1 + a_{i2}x_2 + ... + a_{ij}x_j + ... + a_{in}x_n \le b_i, i = 1,2,...,m$

$x_1, x_2,..., x_n \ge 0$

These are some characteristics of this form,

(i) The objective function is of maximization type Or Maximize Z. If we have given minimize Z, we convert it to maximize by taking negative of Z i.e., Maximize (-Z).

(ii) All constraints should be of the type $"\le"$, except the non-negative restrictions.

(iii) All variables are non-negative.

An LPP in such form known as **Standard form**:

Maximize (Or Minimize), $Z = c_1x_1 + c_2x_2 + ... + c_nx_n$

Subject to,

$$a_{i1}x_1 + a_{i2}x_2 + ... + a_{ij}x_j + ... + a_{in}x_n = b_i, i = 1,2,...,m$$

$$x_1, x_2,..., x_n \ge 0$$

OR

Maximize (Or Minimize), $Z = cx$

Subject to,

$$Ax = b, i = 1,2,...,m$$

$$x \geq 0 \text{ (null vector)}$$

Where $c = (c_1, c_2, ..., c_n)$ and n-component row vector; $x = (x_1, x_2, ..., x_m)$ an m-component column vector; $b = [b_1, b_2, ..., b_m]$ an m-component column vector and the matrix $A = (a_{ij})_{m \times n}$. The characteristic of this form are as follows:

(i) All constraints are expressed in the form of equations, except the non-negative restrictions.

(ii) The RHS of each constraint equation is non-negative.

2.3.2 *SLACK AND SURPLUS VARIABLES*

1. Slack Variables:

- Slack variables are introduced to convert inequality constraints into equations.
- For each inequality constraint of the form $\mathbf{a}_i^T\mathbf{x} \leq b_i$, a slack variable s_i is added such that $\mathbf{a}_i^T\mathbf{x} + s_i = b_i$.
- Slack variables represent the amount by which the left-hand side of the constraint falls short of the right-hand side to satisfy the constraint.
- In the simplex method, slack variables start with a value of zero in the initial basic feasible solution.

OR

Slack variable: A variable which is added to the LHS of a "≤" type constraint to convert the constraint into an equality is called slack variable.

2. Surplus Variables:

- Surplus variables are introduced to convert inequality constraints into equations when the inequalities are of the form $\mathbf{a}_i^T\mathbf{x} \geq b_i$.

- For each inequality constraint of the form $\mathbf{a}_i^T\mathbf{x} \geq b_i$, a surplus variable s_i is added such that $\mathbf{a}_i^T\mathbf{x} - s_i = b_i$.
- Surplus variables represent the amount by which the left-hand side of the constraint exceeds the right-hand side.
- In the simplex method, surplus variables start with a value of zero in the initial basic feasible solution.

Surplus variable: A variable which is subtracted from the LHS of a "≥" type constraint to convert the constraint into an equality is called surplus variable.

These additional variables allow the LP problem to be formulated in canonical form, where all constraints are equations. The simplex method operates on LP problems in canonical form, making it easier to identify and move between basic feasible solutions efficiently.

2.3.3 *BASIC SOLUTION*

Consider a set of m linear simultaneous equations of n ($n > m$) variables.

$$Ax = b,$$

where A is an $m \times n$ matrix of rank m. If any $m \times m$ non-singular matrix B is chosen from A and if all the $(n - m)$ variables not associated with the chosen matrix are set equal to zero, then the solution to the resulting system of equations is a basic solution (BS).

Basic solution has not more than m non-zero variables called basic variables. Thus the m vectors associated with m basic variables are linearly independent. The variables which are not basic, are termed as non-basic variables. If the number of non-zero basic variables is less than m, then the solution is called degenerate basic solution. On the other hand, if none of the basic variables vanish, then the solution is called non-degenerate basic solution. The possible number of basic solutions in a system of m equations in unknowns is ${}^nC_m = \frac{n!}{m!(n-m)!}$.

Theorem 1: The necessary and sufficient condition for the existence and non-degeneracy of all the basic solutions of $Ax = b$ is that every set of m columns of the augmented matrix $[A, b]$ is linearly independent.

Proof: To prove the theorem stating that the existence and non-degeneracy of all basic solutions of $Ax = b$ depend on every set of m columns of the augmented matrix $[A, b]$ being linearly independent, we'll need to establish both the necessity and sufficiency of this condition.

Necessary condition: Let's first prove the necessity part. We want to show that if all basic solutions exist and are non-degenerate, then every set of m columns of $[A, b]$ must be linearly independent.

Suppose that there exists a set of m columns of $[A, b]$ that is linearly dependent. This implies that there exists a nontrivial linear combination of these columns that equals the zero vector. Without loss of generality, let's assume that the linear combination involves the last column, corresponding to the vector b.

$$c_1A_1+c_2A_2+\ldots+c_mA_m+c_{m+1}b=0$$

Where A_i represents the i^{th} column of A and c_i are coefficients not all zero.

Since the last column of $[A, b]$ is linearly dependent on the other columns, it means that the system $Ax = b$ has at least one redundant equation. In other words, the last component of b can be expressed as a linear combination of the other components, rendering the system degenerate.

Hence, if every set of m columns of $[A, b]$ is linearly independent, then the system $Ax = b$ cannot have any redundant equations, ensuring the existence and non-degeneracy of all basic solutions.

Sufficient condition: Now let's prove the sufficiency part, i.e., if every set of m columns of $[A, b]$ is linearly independent, then all basic solutions exist and are non-degenerate.

Suppose all sets of m columns of $[A, b]$ are linearly independent. This implies that each component of b cannot be expressed as a linear combination of the other components. Therefore, each equation in the system $Ax = b$ contributes uniquely to the determination of the solution.

Since there are no redundant equations, every basic solution of the system $Ax = b$ corresponds to a unique set of pivot variables, making the solution non-degenerate. Furthermore, since each equation is necessary for determining the solution, all basic solutions exist.

Therefore, the sufficiency part is proven.

Conclusion: Combining the necessity and sufficiency proofs, we conclude that the existence and non-degeneracy of all basic solutions of $Ax = b$ are guaranteed if and only if every set of m columns of the augmented matrix $[A, b]$ is linearly independent.

2.3.4 *BASIC FEASIBLE SOLUTION (BFS)*

As we know that, if a feasible solution x is also basic, meaning that it corresponds to a set of linearly independent columns of the constraint matrix A, then it is termed a basic feasible solution (BFS). Basic feasible solutions are important in LP because they often correspond to the vertices of the feasible region (in the case of bounded LP problems), and they serve as starting points for various optimization algorithms such as the simplex method.

In summary, a basic feasible solution is a feasible solution that satisfies the additional condition of being basic, implying that it corresponds to a set of linearly independent constraints.

OR

An LPP's feasible solution is one that meets all of its constraints and non-negativity restrictions. A viable solution is referred to as basic feasible solution (BFS) if it is basic once more.

Theorem 2: The necessary and sufficient condition for the existence and non-degeneracy of all possible basic feasible solutions of $Ax = b$, $x \geq 0$ is the linear independent of every set of m columns of the augmented matrix [A, b], where A is the m × n coefficient matrix.

Proof: To prove the theorem that the existence and non-degeneracy of all possible basic feasible solutions of the linear programming problem $Ax = b$, $x \geq 0$ depend on the linear independence of every set of m columns of the augmented matrix $[A, b]$, we need to establish both the necessity and sufficiency of this condition.

Necessary condition: Let's first prove the necessity part. We want to show that if all possible basic feasible solutions exist and are non-degenerate, then every set of m columns of $[A, b]$ must be linearly independent.

Suppose that there exists a set of m columns of $[A, b]$ that is linearly dependent. This implies that there exists a nontrivial linear combination of these columns that equals the zero vector. Without loss of generality, let's assume that the linear combination involves the last column, corresponding to the vector b.

$$c_1A_1+c_2A_2+\ldots+c_mA_m+c_{m+1}b=0$$

Where Ai represents the i^{th} column of A and c_i are coefficients not all zero.

Since the last column of $[A, b]$ is linearly dependent on the other columns, it means that the system $Ax = b$ has at least one redundant equation. In other words, the last component of b can be expressed as a linear combination of the other components, violating the non-negativity constraint $x \geq 0$ and rendering the system degenerate.

Hence, if every set of m columns of $[A, b]$ is linearly independent, then the system $Ax = b$ cannot have any redundant equations, ensuring the existence and non-degeneracy of all possible basic feasible solutions.

Sufficient condition: Now let's prove the sufficiency part, i.e., if every set of m columns of $[A, b]$ is linearly independent, then all possible basic feasible solutions exist and are non-degenerate.

Suppose all sets of m columns of $[A, b]$ are linearly independent. This implies that each component of b cannot be expressed as a linear combination of the other components. Therefore, each equation in the system $Ax=b$ contributes uniquely to the determination of the solution.

Since there are no redundant equations, every basic feasible solution of the system $Ax = b$ corresponds to a unique set of pivot variables, making the solution non-degenerate. Furthermore, since each equation is necessary for determining the solution, all possible basic feasible solutions exist.

Therefore, the sufficiency part is proven.

Conclusion:

Combining the necessity and sufficiency proofs, we conclude that the existence and non-degeneracy of all possible basic feasible solutions of the linear programming problem $Ax = b$, $x \geq 0$ are guaranteed if and only if every set of m columns of the augmented matrix $[A, b]$ is linearly independent.

Example 1: Find out the basic feasible solution for the system of linear equations

$$2x_1 + 3x_2 - x_3 + 4x_4 = 8$$

$$x_1 - 2x_2 + 6x_3 - 7x_4 = -3$$

$$x_1, x_2, x_3, x_4 \geq 0$$

Proof: The given system of equations can be written as

$$a_1x_1 + a_2x_2 + a_3x_3 + a_4x_4 = b$$

where $a_1 = [2,1], a_2 = [3,-2], a_3 = [-1,6], a_4 = [4,-7]$ and $b = [8,-3]$. The maximum number of basic solutions that can be obtained is ${}^4C_2 = 6$. The six sets of 2 vectors out of 4 are

$B_1 = [a_1, a_2]$, $B_2 = [a_1, a_3]$, $B_3 = [a_1, a_4]$

$B_4 = [a_2, a_3]$, $B_5 = [a_2, a_4]$, $B_6 = [a_3, a_4]$.

Here $|B_1| = -7$, $|B_2| = 18$, $|B_3| = -18$, $|B_4| = 16$, $|B_5| = -13$, and $|B_6| = -17$. Since none of these determinants vanishes, hence every set B_i of two vectors is linearly independent. Therefore, the vectors of the basic variables associated to each set B_i, i = 1,2,3,4,5,6 are given by,

$$x_{B_1} = B_1^{-1}b = -\frac{1}{7}\begin{bmatrix} -2 & -3 \\ -1 & 2 \end{bmatrix}\begin{bmatrix} 8 \\ -3 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$$

$$x_{B_2} = B_2^{-1}b = \frac{1}{13}\begin{bmatrix} 6 & 1 \\ -1 & 2 \end{bmatrix}\begin{bmatrix} 8 \\ -3 \end{bmatrix} = \begin{bmatrix} 45/13 \\ -14/13 \end{bmatrix}$$

$$x_{B_3} = B_3^{-1}b = \frac{1}{18}\begin{bmatrix} -7 & -4 \\ -1 & 2 \end{bmatrix}\begin{bmatrix} 8 \\ -3 \end{bmatrix} = \begin{bmatrix} 22/9 \\ 7/9 \end{bmatrix}$$

$$x_{B_4} = B_4^{-1}b = \frac{1}{16}\begin{bmatrix} 6 & 1 \\ 2 & 3 \end{bmatrix}\begin{bmatrix} 8 \\ -3 \end{bmatrix} = \begin{bmatrix} 45/16 \\ 7/16 \end{bmatrix}$$

$$x_{B_5} = B_5^{-1}b = -\frac{1}{13}\begin{bmatrix} -7 & -4 \\ 2 & 3 \end{bmatrix}\begin{bmatrix} 8 \\ -3 \end{bmatrix} = \begin{bmatrix} 44/13 \\ -7/13 \end{bmatrix}$$

And $x_{B_6} = B_6^{-1}b = -\frac{1}{17}\begin{bmatrix} -7 & -4 \\ -6 & -1 \end{bmatrix}\begin{bmatrix} 8 \\ -3 \end{bmatrix} = \begin{bmatrix} 44/17 \\ 45/17 \end{bmatrix}$

From above, we see that the possible basic feasible solutions are $x_1 = [1, 2, 0, 0]$, $x_2 = [22/9, 0, 0, 7/9]$, $x_3 = [0, 45/16, 7/16, 0]$ and $x_4 = [0, 0, 44/17, 45/17]$ which are also non-degenerate. The other basic solutions are not feasible.

Theorem 3: (Fundamental theorems of LP): If a linear programming problem has an optimal solution, then the optimal solution will coincide with at least one basic feasible solutions of the problem.

Proof: Let us consider that x^* is an optimal solution of the following LPP:

Maximize, $z = cx$

Subject to $Ax = b, x \geq 0$ … (1)

Without loss of generality, we assume that the initial *p* component of optimal solution x^* are non-zero and the remaining $(n-p)$ component of x^* are non-zero. Thus,

$$x^* = [x_1, x_2, ..., x_p, 0, 0, ..., 0]$$

Then, from (1), $Ax^* = b$ gives $\sum_{j=1}^{p} a_{ij} x_j = b_i, i = 1, 2, ..., m$

Also, $A = [a_1, a_2, ..., a_p, a_{p+1}, , ..., a_n]$ gives

$$a_1 x_1 + a_2 x_2 + ..., a_p x_p = b \qquad \text{… (2)}$$

$$\text{Also, } z^* = z_{\max} = \sum_{j=1}^{p} c_j x_j \qquad \text{… (3)}$$

Now, if the vectors $a_1, a_2, ..., a_p$ correspond to the non-zero components of x^* are linearly independent, then, by definition of x^* is a basic solution by definition, and the theorem is valid in this instance. The simplest possible solution is non-degenerate if *p* = *m*. Conversely, in the event where *p* is less than *m*, a degenerate basic feasible option will be formed, wherein the basic variables (*m* - *p*) equal zero.

If vectors are not linearly independent, then they must be linearly dependent i.e., there exists scalars $\lambda_j, j = 1, 2, ..., p$ of which at least one of the $\lambda_j{}'s$ is non-zero such that

$$\lambda_1 a_1 + \lambda_2 a_2 + ... + \lambda_p a_p = 0 \qquad \text{… (4)}$$

Suppose that at least one $\lambda_j > 0$. If the non-zero λ_j is not positive, then we can multiply (4) by (-1) to get a positive λ_j.

$$\text{Let } \mu = \underset{1 \le j \le p}{Max} \left\{ \frac{\lambda_i}{x_j} \right\} \qquad \text{… (5)}$$

Then μ is positive as $x_j > 0 \, \forall \, j = 1, 2, ..., p$ and at least one λ_j is positive. Dividing (4) by μ and subtracting it from (2), we get

$$\left(x_1 - \frac{\lambda_1}{\mu} \right) a_1 + \left(x_2 - \frac{\lambda_2}{\mu} \right) a_2 + ... + \left(x_p - \frac{\lambda_p}{\mu} \right) a_p = b$$

And hence $x = \left[\left(x_1 - \frac{\lambda_1}{\mu}\right)a_1, \left(x_2 - \frac{\lambda_2}{\mu}\right)a_2, ..., \left(x_p - \frac{\lambda_p}{\mu}\right), 0, 0, ..., 0\right]$ … (6)

is a solution of the systems of equations $Ax = b$

Again from (5), we have

$$\mu \geq \frac{\lambda_i}{x_j} \text{ for } j = 1, 2, ..., p$$

Or $\mu \geq \frac{\lambda_i}{x_j}$ for $j = 1, 2, ..., p$

This implies that all the components of x_1 are non-negative and hence x_1 is feasible solution of $Ax = b, x \geq 0$. Again, for at least one value of j, we have, from (5), $x_j - \frac{\lambda_i}{\mu} = 0$, for at least one value of j.

As a result, we may observe that the feasible solution x_1 will have one extra zero than what was demonstrated in (6). Therefore, there can be no more than $(p-1)$ non-zero variables in the possible solution x_1. As a result, we have demonstrated that it is possible to decrease the number of positive variables that provide an optimal solution.

$$z' = cx_1 = \sum_{j=1}^{p} c_j \left(x_j - \frac{\lambda_j}{\mu}\right) = \sum_{j=1}^{p} c_j x_j - \sum_{j=1}^{p} c_j \frac{\lambda_j}{\mu} = z^* - \frac{1}{\mu}\sum_{j=1}^{p} c_j \lambda_j \text{, by (3)} \quad \text{...... (7)}$$

Now, if we can show that,

$$\sum_{j=1}^{p} c_j \lambda_j = 0 \quad \text{... (8)}$$

Then $z' = z^*$ and this will prove that x_1 is an optimal solution.

We assume that equation (8) does not hold and we find a suitable real number γ, such that

$$\gamma(c_1\lambda_1 + c_1\lambda_2 + ... + c_p\lambda_p) > 0$$

i.e., $c_1(\gamma\lambda_1) + c_2(\gamma\lambda_2) + ... + c_p(\gamma\lambda_p) > 0$.

Adding $(c_1x_1 + c_2x_2 + ... + c_px_p)$ to both sides, we get

$$c_1(x_1 + \gamma\lambda_1) + c_2(x_2 + \gamma\lambda_2) + ... + c_p(x_p + \gamma\lambda_p) > c_1x_1 + c_2x_2 + ... + c_px_p = z^* \qquad \dots (9)$$

Again, multiply (4) by γ and adding to (2), we get

$$(x_1 + \gamma\lambda_1)a_1 + (x_2 + \gamma\lambda_2)a_2 + ... + (x_p + \gamma\lambda_p)a_p = b$$

So that

$$[(x_1 + \gamma\lambda_1),(x_2 + \gamma\lambda_2),...,(x_p + \gamma\lambda_p),0,0,0] \qquad \dots (10)$$

is also a solution of the system $Ax = b$

Now, we choose γ such that,

$$x_j + \gamma\lambda_j \geq 0 \; \forall \; j = 1,2,..., p$$

or $\gamma \geq -\dfrac{x_j}{\lambda_j} \, if \; \lambda_j > 0$

and $\gamma \leq -\dfrac{x_j}{\lambda_j} \, if \; \lambda_j > 0$

and γ is unrestricted, if $\lambda_j = 0$.

Now equation (10) becomes a feasible solution of $Ax = b, x \geq 0$.

Thus choosing γ is such a way that,

$$\underset{\substack{j \\ \lambda_j > 0}}{Max}\left\{-\frac{x_j}{\lambda_j}\right\} \leq \gamma \leq \underset{\substack{j \\ \lambda_j < 0}}{Max}\left\{-\frac{x_j}{\lambda_j}\right\}$$

We see from equation (9) that the feasible solution equation (10) gives a greater value of the objective function than z^*. Which is the contradiction our assumption that z^* is optimal value. Thus we can say that

$$\sum_{j=1}^{p} c_j \lambda_j = 0$$

Thus, x_1 is likewise the best option. As a result, we demonstrate that the number of non-zero variables in the given optimal solution is fewer than that of the one that was provided. If the additional non-zero variables' corresponding vectors are linearly independent, then the theorem follows because the new solution will be a fundamentally workable solution. We can further reduce the number of non-zero variables as previously mentioned to obtain a new set of optimal solutions if the new solution is once more not a fundamentally possible option. We can keep going until we arrive at an ideal solution that is also a basically feasible solution.

2.4 *SIMPLEX ALGORITHM*

Any LP problem that may be solved using a simplex algorithm always assumes the existence of a starting BFS. We shall talk about the LP issue of maximizing kind using the simplex approach here. Here's a simplified explanation of how it works:

1. **Initialization**: Start with an initial feasible solution. This can be achieved by solving a set of linear equations or inequalities that satisfy the constraints of the problem.
2. **Iteration**: The algorithm iterates through a series of steps to improve the solution. At each iteration, it selects a variable to enter the solution and a variable to leave the solution, moving towards the optimal solution.
3. **Optimality Test**: At each iteration, the algorithm checks if the current solution is optimal. If it is, the algorithm terminates. Otherwise, it proceeds to the next step.
4. **Pivoting**: If the current solution is not optimal, the algorithm performs a pivoting operation to improve the solution. This involves selecting a pivot element in the current tableau (a table representing the problem), and using it to update the tableau in a way that improves the objective function value.
5. **Repeat**: Steps 3 and 4 are repeated until an optimal solution is found.

The Simplex algorithm is efficient and can handle large-scale linear programming problems with thousands or even millions of variables and constraints. However, it's worth noting that in some cases, the algorithm may take exponential time to find the optimal solution, although this is rare in practice.

The following are the steps involved in computing an optimal solution:

Step 1: If the given LPP is of minimization type, then convert the objective function to maximizing type. Additionally, change all m constraints to non-negative b_i's (i = 1, 2, ..., m).

Next, create an equation for each inequality constraint by adding a slack or surplus variable, and give that variable a zero cost coefficient in the objective function.

Step 2: If necessary, introduce artificial variable(s) and take (-*M*) as the coefficient of each artificial variables in the objective function.

Step 3: Obtain the initial basic feasible solution $x_B = B^{-1}b$, where *B* is the basis matrix (Which is here an identity matrix).

Step 4: Calculate the net evaluation $z_j - c_j = c_B x_{Bj} - c_j$.

(i) If $z_j - c_j \geq 0 \ \forall\ j$ then x_B is an optimum BFS.

(ii) If at least once $z_j - c_j < 0$. Then to improve the next solution we proceed in next step.

Step 5: If there are more than one negative $z_j - c_j$, then choose the most negative of them. Let it be $z_k - c_k$ for some $j = k$.

(i) If all $a_{ik} < 0$ (*i* = 1, 2, …., *m*), then there exist an unbounded solution to the given problem.

(ii) If at least one $a_{ik} > 0 \, (i = 1, 2, ..., m)$ then the corresponding vector a_k enters the basis *B*. This column is called the *key* or *pivot column*.

Step 6: Divide each value of x_B (i.e., b_i) by the corresponding (but positive) number in the key column and select a row which has the ratio non-negative and minimum i.e.,

$$\frac{x_{Br}}{a_{rk}} = Min\left\{\frac{x_{Bi}}{a_{ik}}; a_{ik} > 0\right\}$$

We refer to this rule as the minimum ratio rule. This kind of row selection is known as the pivot or key row, and it stands for the variable that will be eliminated from the fundamental solution. The key or pivot element (let's say ark) is the element that is located where the simplex table's key row and key column intersect.

Step 7: Use the relation to convert all other elements in its column to zeros and the leading element to unity by dividing its row by the key element itself:

$$\hat{a}_{rj} = \frac{a_{rj}}{a_{rk}} \text{ and } \hat{x}_{Br} = \frac{x_{Br}}{a_{rk}},\ i = r;\ j = 1, 2, ...n$$

$\hat{a}_{ij} = a_{ij} - \frac{a_{rj}}{a_{rk}} a_{ik}$ and $\hat{x}_{Br} = x_{Bi} - \frac{x_{Br}}{a_{rk}} a_{ik}$, $i = 1, 2, ...m; i \neq r$

Step 8: Now go to the step 4 and repeat the procedure until all entries in $(z_j - c_j)$ are either positive or zero, or there is an indication of an unbounded solution.

2.4.1 *SIMPLEX TABLE*

The simplex for a standard LPP

Maximize, $z = cx$

Subject to, $Ax = b$

$x \geq 0$

is given below:

Where, $B = (a_{B1}, a_{B2}, ..., a_{Bm})$, basis matrix

$x_B = (x_{B1}, x_{B2}, ..., x_{Bm})$, basic variables

$c_B = [c_{B1}, c_{B2}, ..., c_{Bm}]$

$A = (a_{ij})_{m \times n}$

$b = [b_1, b_2, ..., b_m]$

$c = (c_1, c_2, ..., c_n)$

$x = (x_1, x_2, ..., x_n)$

			$c_j \rightarrow$	c_1	c_2	...	c_n
$\mathbf{c_B}$	$\mathbf{B}$	$\mathbf{x_B}$	$\mathbf{b}$	a_1	a_2	...	a_n
c_{B1}	a_{B1}	x_{B1}	b_1	a_{11}	a_{12}	...	a_{1n}
c_{B2}	a_{B2}	x_{B2}	b_2	a_{21}	a_{22}	...	a_{2n}
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
c_{Bm}	a_{Bm}	x_{Bm}	b_m	a_{m1}	a_{m2}	...	a_{mn}
			$z_j - c_j$	$z_1 - c_1$	$z_2 - c_2$	...	$z_n - c_n$

Table 2.1: Simplex table

Example 2: Using simplex method solve the following LPP.

Maximize, $Z = x_1 - 3x_2 + 2x_3$

Subject to,

$3x_1 - x_2 + 2x_3 \leq 7$

$-2x_1 + 4x_2 \leq 12$

$-4x_1 + 3x_2 + 8x_3 \leq 10$

$x_1, x_2, x_3 \geq 0$

Solution: We have given the problem of minimization. So, at first we convert the problem into maximization.

So, we have Max$(Z_1) = Min(-Z) = -x_1 + 3x_2 - 2x_3$. Now, introduced the slack variable x_4, x_5 and x_6, then problem can be put in the standard form as

Max$(Z_1) = -x_1 + 3x_2 - 2x_3 + 0x_4 + 0x_5 + 0x_6$

Subject to,

$3x_1 - x_2 + 2x_3 + x_4 = 7$

$$-2x_1 + 4x_2 + x_5 = 12$$

$$-4x_1 + 3x_2 + 8x_3 + x_6 = 10$$

$$x_1, x_2, x_3, x_4, x_5, x_6 \geq 0$$

Now, we apply the simplex algorithm (Step 2 to Step 8). The outcomes of each iteration are displayed in Table 2.2. Since, $z_j - c_j \geq 0 \; \forall \; j$ in the last iteration Table 2, condition of optimality is satisfied. The optimal solutions are $x_1 = 4, x_2 = 5, x_3 = 0$ and the corresponding objective function is $(Z_1)_{\max} = 11$. Hence the solution of the original problem is $x_1 = 4, x_2 = 5, x_3 = 0$ and $(Z_1)_{\min} = -11$

			$c_j \rightarrow$	-1	3	-2	0	0	0	Mini
c_B	**B**	x_B	**b**	a_1	a_2	a_3	a_4	a_5	a_6	Ratio
0	a_4	x_4	7	3	-1	2	1	0	0	-
0	a_5	x_5	12	-2	4	0	0	1	0	12/4=3
0	a_6	x_6	10	-4	3	8	0	0	1	10/3=3.33
			$z_j - c_j$	1	-3	2	0	0	0	
0	a_4	x_4	10	5/2	0	2	1	1/4	0	
3	a_2	x_2	3	-1/2	1	0	0	1/4	0	
0	a_6	x_6	1	-5/2	0	8	0	-3/4	1	
			$z_j - c_j$	-1/2	0	2	0	3/4	0	
-1	a_1	x_1	4	1	0	4/5	2/5	1/10	0	
3	a_2	x_2	5	0	1	2/5	1/5	3/10	0	
0	a_6	x_6	11	0	0	10	1	-1/2	1	
			$z_j - c_j$	0	0	12/5	1/5	16/20	0	

Table 2.2: Simplex table

Example 3: Niki works at Job I and Job II, two part-time employment. She has a strict limit of 12 hours per week that she would never work. She has calculated that she requires two hours of preparation time for every hour she works at Job I, and one hour of preparation time for every hour she works at Job II. She has also decided that she cannot spend more than sixteen hours preparing. How many hours a week should she work at each job to optimize her income if she makes $40 an hour at Job I and $30 an hour at Job II?

Solution: (*Solution of this example is described in step wise procedure and also described many questions which can be arise on mind during solving by simplex method***)** We will use the above-mentioned algorithm to solve this problem.
Step 1: Define the issue. Write the constraints and the goal function.

Since the simplex method is used for problems that consist of many variables, it is not practical to use the variables *x*, *y*, *z* etc. We use symbols x_1, x_2, x_3 and so on.

$x_1 =$ The number of hours per week Niki will work at Job I and

$x_2 =$ The number of hours per week Niki will work at Job II.

Traditionally, Z is selected as the variable to be maximized. The formulation of the problem is the same as it was in the previous chapter.

Maximize, $Z = 40x_1 + 30x_2$

Subject to,

$x_1 + x_2 \le 12$

$2x_1 + x_2 \le 16$

$x_1, x_2 \ge 0$

Step 2: Convert the inequalities into equations: For every inequality, one slack variable is added to achieve this. Convert the equality into inequality $x_1 + x_2 \le 12$. we add a non-negative variable y_1, and we get

$$x_1+x_2+y_1=12$$

Here the variable y_1 picks up the slack, and it represents the amount by which x_1+x_2 falls short of 12. In this problem, if Niki works fewer than 12 hours, say 10, then y_1 is 2. Later when we read off the final solution from the simplex table, the values of the slack variables will identify the unused amounts.

We rewrite the objective function $Z = 40x_1+30x_2$ = 401+302 as $-40x_1-30x_2+Z = 0$

Subject to constraints: $-40x_1-30x_2+Z=0$

$x_1+x_2+y_1=12$

$2x_1+x_2+y_2=16$

$x_1\geq 0$; $x_2\geq 0$

Step 3: Construction of initial table of simplex method: Every inequality constraint is displayed in a separate row. (In the simplex tableau, the non-negativity constraints are not represented as rows.) Assign the bottom row to the objective function.

After the inequalities have been transformed into equations, we can use the following augmented matrix representation of the issue to create the initial simplex tableau.

x1	x2	y1	y2	Z	C
1	1	1	0	0	12
2	1	0	1	0	16
–40	–30	0	0	1	0

The left and right sides of the equations are divided in this instance by a vertical line. The goal function and constraints are divided by the horizontal line. Column C is the representation of the right side of the equation.

It is important for the reader to note that the final four columns of this matrix resemble the final matrix obtained by solving a system of equations. If we select $x_1 = 0$ and $x_2 = 0$ at random, we obtain

$$\begin{bmatrix} y_1 & y_2 & Z & | & C \\ 1 & 0 & 0 & | & 12 \\ 0 & 1 & 0 & | & 16 \\ 0 & 0 & 1 & | & 0 \end{bmatrix}$$

Which reads

$y_1=0$, $y_2=16$, $Z = 0$

The basic solution related to the tableau is the result of solving for the remaining variables after randomly allocating values to some of the variables. Thus, the basic solution for the original simplex tableau is the one mentioned above. As indicated in the table below, we can identify the fundamental solution variable to the right of the final column.

x_1	x_2	y_1	y_2	Z		
1	1	1	0	0	12	y_1
2	1	0	1	0	16	y_2
–40	–30	0	0	1	0	Z

Step 4: The pivot column is indicated by the lowest row's most negative entry:

Since the bottom row's most negative entry is -40, column 1 is recognized.

x_1	x_2	y_1	y_2	Z		
1	1	1	0	0	12	y_1
2	1	0	1	0	16	y_2
–40	–30	0	0	1	0	Z
↑						

Question: Why do we select the lowest row entry that is the most negative?

Answer: The coefficient whose entry will raise the value of the objective function the fastest is the greatest coefficient in the objective function, and it is represented by the most negative entry in the bottom row.

The simplex approach starts at a corner where all of the primary variables, variables with symbols like x_1, x_2, x_3, etc. have zero values. Next, it advances from one corner point to the next, always raising the goal function's value. Increasing the value of x_1 will make more sense in the case of the objective function $Z = 40x_1+30x_2$ than x_2 will. The number of hours a week that Niki works at Job I is represented by the variable x_1. The variable x_1 will raise the goal function by $40 for every unit increment in the variable x_1, as Job I pays $40 per hour whereas Job II only pays $30.

Step 5: Do the quotient calculations. The row is identified by the least quotient. The pivot element is the one that is found at the intersection of the row found in this step and the column found in step 4.

We divide the items in the far right column by the entries in column 1, omitting the entry in the bottom row, in accordance with the algorithm to find the quotient.

x1	x2	y1	y2	Z			
1	1	1	0	0	12	y1	12 ÷ 1 = 12
[2]	1	0	1	0	16	y2	← 16 ÷ 2 = 8
–40	–30	0	0	1	0	Z	
↑							

Of the two quotients, 12 and 8, 8 is the smallest. Row 2 is thus identified. The highlighted entry number two is located at the junction of row 2 and column 1. This is the key component for us.

Question: Why do we look for quotients, and how does a row become identified by its smallest quotient?

Answer: By adding the variable x_1, we want to raise the value of the objective function when we select the entry in the bottom row that is the most negative. However, we are unable to select a value for x_1. Can we allow for $x_1=100$? Absolutely not! This is due to Niki's insistence on never working more than 12 hours at both jobs put together: $x_1+x_2 \leq 12$. Can we allow for $x_1=12$? Once more, the answer is no, as the time required to prepare for task I is twice that of the actual task. Niki can work no more than 16 ÷ 2 = 8 hours since she never wants to spend more than 16 hours preparing.

You now understand why it is necessary to compute the quotients; doing so ensures that we do not go against the limitations when identifying the pivot element.

Question: Why is the pivot element identified?

Answer: The simplex approach, as previously discussed, starts at one corner point and advances to the next, always increasing the value of the objective function. By altering the number of units of the variables, the objective function's value is increased. One variable's number of units may be increased while the units of another are subtracted. We can accomplish just that by pivoting. The variable that is being added units is referred to as the entering variable, while the variable that is being replaced units are referred to as the departing variable. The highest negative item in the bottom row of the above table indicates that x_1 is the entering variable. The lowest of all quotients was used to identify the departure variable, y_2.

Step 6: Perform pivoting to make all other entries in this column zero

To get the row echelon form of an augmented matrix, we pivot the matrix. Getting a 1 at the pivot element's location and then setting all other values in that column to zeros is the process of pivoting. It is now our task to divide the entire second row by two in order to turn our pivot element into a 1. The outcome is as follows.

x_1	x_2	y_1	y_2	Z	
1	1	1	0	0	12
[1]	1/2	0	1/2	0	8
–40	–30	0	0	1	0

We add row 1 to the second row after multiplying it by -1 to get a zero in the entry above the pivot element. We obtain

x_1	x_2	y_1	y_2	Z	
0	1/2	1	–1/2	0	4
[1]	1/2	0	1/2	0	8
–40	–30	0	0	1	0

We multiply the second row by 40 and add it to the last row in order to get a zero in the element below the pivot.

x_1	x_2	y_1	y_2	Z		
0	1/2	1	–1/2	0	4	y_1
[1]	1/2	0	1/2	0	8	x_1
0	–10	0	20	1	320	Z

We now ascertain the fundamental solution linked to this tableau. Upon selecting $x_2 = 0$ and $y_2 = 0$ at random, we arrive at $x_1 = 8$, $y_1 = 4$, and $z = 320$. The following matrix states the same thing if we write the augmented matrix, whose left side is a matrix with one column having a 1 and all other entries zeros.

$$\begin{bmatrix} x_1 & y_1 & Z & | & C \\ 0 & 1 & 0 & | & 4 \\ 1 & 0 & 0 & | & 8 \\ 0 & 0 & 1 & | & 320 \end{bmatrix}$$

The answer linked to this matrix can be restated as follows: $z = 320$, $y_1 = 4$, $y_2 = 0$, $x_1 = 8$, and $x_2 = 0$. At this point in the game, Niki's profit Z is \$320 if she works 8 hours at Job I and 0 hours at Job II.

Step 7: We are done when the bottom row contains no more negative entries; if not, we go back to step 4 and repeat the process.

We must start over at step 4 because there is still a negative entry, -10, in the bottom row. Instead of going over each step in detail again, this time we will find the column and row that contain the pivot element and highlight it. This is the outcome.

x1	x2	y1	y2	Z			
0	[1/2]	1	–1/2	0	4	y1	← 4 ÷ 1/2 = 8
1	1/2	0	1/2	0	8	x1	8 ÷ 1/2 = 16
0	–10	0	20	1	320	Z	
	↑						

By multiplying row 1 by 2, we create the pivot element 1, and we obtain

x1	x2	y1	y2	Z	
0	[1]	2	–1	0	8
1	1/2	0	1/2	0	8
0	–10	0	20	1	320

We are done because there are no more negative entries in the bottom row.

Question: When there are no negative entries in the bottom row, why are we done?

Answer: The bottom row has the solution. The equation and the bottom row match.

$0x_1 + 0x_2 + 20y_1 + 10y_2 + Z = 400$ **or**

$Z = 400 - 20y_1 - 10y_2$

The maximum number Z can ever reach is 400 because all variables are non-negative, and that can only occur when both y_1 and y_2 are 0.

Step 8: Now that we have determined the fundamental solution linked to the final simplex tableau, we read off our solutions. Once more, we examine the columns containing a 1 and all other entries are zeros. We choose $y_1 = 0$ and $y_2 = 0$ at random because the columns with the labels y_1 and y_2 are not such columns, and we obtain

$$\begin{bmatrix} x_1 & x_2 & Z & | & C \\ 0 & 1 & 0 & | & 8 \\ 1 & 0 & 0 & | & 4 \\ 0 & 0 & 1 & | & 400 \end{bmatrix}$$

The values of the matrix are z = 400, x_1 = 4, and x_2 = 8. According to the final solution, Niki will maximize her income to $400 if she works 4 hours at Job I and 8 hours at Job II. She would have used up all of the working and preparation time because both of the slack variables are 0, meaning that none will be left.

Example 4: Using Simplex method solve the following LP problem

Maximize, $Z = 4x_1 + 10x_2$

Subject to,

$2x_1 + x_2 \le 50$

$2x_1 + 5x_2 \le 100$

$2x_1 + 3x_2 \le 90$

$x_1, x_2 \ge 0$

Solution Step 1: Introducing the slack variable.

Maximize, $Z = 4x_1 + 10x_2 + 0s_1 + 0s_2 + 0s_3$

Subject to,

$2x_1 + x_2 + s_1 = 50$

$2x_1 + 5x_2 + s_2 = 100$

$2x_1 + 3x_2 + s_3 = 90$

$x_1, x_2, s_1, s_2 \ge 0$

So, the given L.P.P. converted to the following system of linear equations.

$$\begin{pmatrix} 2 & 1 & 1 & 0 & 0 \\ 2 & 5 & 0 & 1 & 0 \\ 2 & 3 & 0 & 0 & 1 \end{pmatrix} \begin{bmatrix} x_1 \\ x_2 \\ s_1 \\ s_2 \\ s_3 \end{bmatrix} = \begin{bmatrix} 50 \\ 100 \\ 90 \end{bmatrix}$$

Step 2: So, the basic feasible solution is given by $x_B = B^{-1}b$

i.e., $\begin{bmatrix} s_1 \\ s_2 \\ s_3 \end{bmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{bmatrix} 50 \\ 100 \\ 90 \end{bmatrix} = \begin{bmatrix} 50 \\ 100 \\ 90 \end{bmatrix}$

Here, $B = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = B^{-1}$ and $b = \begin{bmatrix} 50 \\ 100 \\ 90 \end{bmatrix}$

Step 3: Now compute y_j and $(z_j - c_j)$ as follows:

$$y_1 = B^{-1}a_1 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{bmatrix} 2 \\ 2 \\ 2 \end{bmatrix} = \begin{bmatrix} 2 \\ 2 \\ 2 \end{bmatrix}$$

$$y_2 = B^{-1}a_2 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{bmatrix} 1 \\ 5 \\ 3 \end{bmatrix} = \begin{bmatrix} 1 \\ 5 \\ 3 \end{bmatrix}$$

$y_3 = B^{-1}e_1 = e_1, y_4 = B^{-1}e_2 = e_2$ and $y_5 = B^{-1}e_3 = e_3$

$$z_1 - c_1 - = c_B y_1 - c_1 = (0,0,0) \begin{bmatrix} 2 \\ 2 \\ 2 \end{bmatrix} - 4 = -4$$

$$z_2 - c_2 - = c_B y_2 - c_2 = (0,0,0) \begin{bmatrix} 1 \\ 5 \\ 3 \end{bmatrix} - 10 = -10$$

$$z_3 - c_3 - = c_B y_3 - c_3 = (0,0,0) \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} - 0 = 0$$

$$z_4 - c_4 - = c_B y_4 - c_4 = (0,0,0)\begin{bmatrix}0\\1\\0\end{bmatrix} - 0 = 0$$

$$z_5 - c_5 - = c_B y_5 - c_5 = (0,0,0)\begin{bmatrix}0\\0\\1\end{bmatrix} - 0 = 0$$

Step 4: The simplex table below now displays the initial basic feasible answer.

		c_j	4	10	0	0	0
c_B	y_B	x_B	y_1	y_2	y_3	y_4	y_5
0	y_3	50	2	1	1	0	0
0	y_4	100	2	**5**	0	1	0
0	y_5	90	2	3	0	0	1
	z_j		0	0	0	0	0
	$z_j - c_j$	z (=0)	-4	-10	0	0	0

It is clear from the tableau that two of the zj-cj are negative. We select -10, which is the most negative of these. The corresponding column vector (y_2) enters the basis.

Step 5: Given that y_2's entries are all positive. We compute $Min\left\{\frac{x_{Bi}}{y_{ir}}; y_{jr} > 0\right\}$ i.e.,

$Min\left\{\frac{50}{1}, \frac{100}{5}, \frac{90}{3}\right\} = \frac{100}{5}$. This occur for the element y_{22} = (=5) i.e. (element of second row and second column). As a result, the column element becomes the leading element for the first iteration and the vector y_4 departs from the basis y_B.

Step 6: Utilizing the following transformation, convert all of y_2's elements to zeros and the leading element, y_{22}, to unity:

$$\hat{y}_{ij} = y_{ij} - \frac{y_{2j}}{y_{22}} y_{i2}; i = 1, 2, 3, 4 \,\&\, i \neq 2$$

$$\hat{y}_{2j} = \frac{y_{2j}}{y_{22}};\ j = 0,1,2,3,4,5$$

$$\therefore \hat{y}_{21} = \frac{y_{21}}{y_{22}} = \frac{2}{5};\ y_{20} = \frac{y_{20}}{y_{22}} = \frac{100}{5}\ or\ 20\ etc.$$

$$\hat{y}_{10} = y_{10} - \frac{y_{20}}{y_{22}}\, y_{12} = 50 - \frac{100}{5} \times 1 = 30$$

$$\hat{y}_{30} = y_{30} - \frac{y_{20}}{y_{22}}\, y_{32} = 90 - \frac{100}{5} \times 3 = 30$$

$$\hat{y}_{31} = y_{31} - \frac{y_{21}}{y_{22}}\, y_{32} = 2 - \frac{2}{5} \times 3 = \frac{4}{5}$$

$$\hat{y}_{11} = y_{11} - \frac{y_{21}}{y_{22}}\, y_{12} = 2 - \frac{2}{5} \times 1 = \frac{8}{5}$$

$$\hat{y}_{14} = y_{14} - \frac{y_{24}}{y_{22}}\, y_{12} = 0 - \frac{1}{5} \times 1 = -\frac{1}{5}$$, and so on.

Step 7: By using above mentioned calculation, the simplex table is given below:

		c_j	4	10	0	0	0
c_B	y_B	x_B	y_1	y_2	y_3	y_4	y_5
0	y_3	30	8/5	0	1	-1/5	0
10	y_2	20	2/5	1	0	1/5	0
0	y_5	30	4/5	0	0	-3/5	1
	z_j		4	10	0	2	0
	$z_j - c_j$	z (=200)	0	0	0	2	0

With an increasing value of z, the following simplex table produces a new fundamental feasible solution. Moreover, since $z_j - c_j > 0$, there is no chance for z to increase any further. Thus, using only the most basic variables x_2, s_1 and s_3 we have arrived at our ideal answer. So, the optimal/maximal basic feasible solution of given LPP is $x_1 = 0, x_2 = 20$ with maximum $z = 200$.

Example 5: Using Simplex method solve the following LP problem

Minimize, $Z = x_2 - 3x_3 + 2x_5$

Subject to,

$3x_2 - x_3 + 2x_5 \leq 7$

$-2x_2 + 4x_3 \leq 12$

$-4x_2 + 3x_3 + 8x_5 \leq 10$

$x_2, x_3, x_5 \geq 0$

Solution Step 1: Introducing the slack variable.

Maximize, $Z^* = -(x_2 - 3x_3 + 2x_5) + 0s_1 + 0s_2 + 0s_3$

Subject to,

$3x_2 - x_3 + 2x_5 + s_1 = 7$

$-2x_2 + 4x_3 + 0.x_5 + s_2 = 12$

$-4x_2 + 3x_3 + 8x_5 + s_3 = 10$

$x_2, x_3, x_5, s_1, s_2, s_3 \geq 0$

So, the given L.P.P. converted to the following system of linear equations.

$$\begin{pmatrix} 2 & 1 & 2 & 1 & 0 & 0 \\ 2 & 5 & 0 & 0 & 1 & 0 \\ 2 & 3 & 8 & 0 & 0 & 1 \end{pmatrix} \begin{bmatrix} x_2 \\ x_3 \\ x_5 \\ s_1 \\ s_2 \\ s_3 \end{bmatrix} = \begin{bmatrix} 7 \\ 12 \\ 10 \end{bmatrix} \text{ or } Ax = b$$

So, the obvious initial basic feasible solution is $x_B = B^{-1}b$ where $B = I_3$, and

$x_B =$ basic variable corresponding to columns of basis matrix $B(=I)$.

Step 2: So, the basic feasible solution is given by $x_B = B^{-1}b$

i.e., $\begin{bmatrix} s_1 \\ s_2 \\ s_3 \end{bmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{bmatrix} 50 \\ 100 \\ 90 \end{bmatrix} = \begin{bmatrix} 50 \\ 100 \\ 90 \end{bmatrix}$

Here, $B = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = B^{-1}$ and $b = \begin{bmatrix} 50 \\ 100 \\ 90 \end{bmatrix}$

$$\begin{bmatrix} s_1 \\ s_2 \\ s_3 \end{bmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}^{-1} \begin{bmatrix} 7 \\ 12 \\ 10 \end{bmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{bmatrix} 7 \\ 12 \\ 10 \end{bmatrix} = \begin{bmatrix} 7 \\ 12 \\ 10 \end{bmatrix}$$

Now, the simplex table is:

		c_j	-1	3	-2	0	0	0
c_B	y_B	x_B	y_1	y_2	y_3	y_4	y_5	y_6
0	y_4	7	3	-1	2	1	0	0
0	y_5	12	-2	**4**	0	0	1	0
0	y_6	10	-4	3	8	0	0	1
	z	0	1	-3	2	0	0	0

Since there is at least one negative z_j-c_j, or z_2-c_2, the existing basic feasible option is not optimal. We choose the column corresponding to z_2-c_2, *i.e.*, column vector y_2 enters the basis y_B (Since at least one y_{i2}>0). Further, since minimum $Min\left\{\frac{x_{Bi}}{y_{i2}}; y_{i2} > 0\right\}$ is $\frac{12}{4}(=3)$, current basis vector y_5 leaves the basis. y_{22} (= 4) is thus identified as the leading element. We now change all other elements of the incoming column vector y_2 to zero and the leading element to unity using E-Row operations. As seen in the following simplex table, we obtain the improved basic feasible solution.

		c_j	-1	3	-2	0	0	0
c_B	y_B	x_B	y_1	y_2	y_3	y_4	y_5	y_6
0	y_4	10	**5/2**	0	2	1	1/4	0
3	y_2	3	-1/2	1	0	0	1/4	0
0	y_6	1	-5/2	0	8	0	-3/4	1
	z	9	-1/2	0	2	0	3/4	0

Over that z_1-c_1 is negative and thus the current basic feasible solution is not optimum. The column corresponding to z_1-c_1 enters the next basis y_B (since $y_{i\ 1}$>0). Further, since only y_{11}>0 both y_{12}<0 and y_{13}<0); current basis vector y_4 leave the basis. This gives $y_{1\ 1}$ (= 5/2) as the leading element. We change the leading element to unity and the other values in its column y_1 to zero using E-row operations. The next simplex table displays the improved basic feasible solution.

c_B	y_B	x_B	y_1	y_2	y_3	y_4	y_5	y_6
-1	y_1	4	1	0	4/5	2/5	1/10	0
3	y_2	5	0	1	2/5	1/5	3/10	10
0	y_6	11	0	0	10	2/5	-1/2	1
	z	11	0	0	12/5	1/5	8/10	0

In this table, all z_i-c_i ≥ 0, an optimal BFS has been attained. So, the optimal solution of the given L.P.P. is,

Minimum Z = - Maximum $Z^* = -11$ with $x_2 = 4, x_3 = 5$ and $x_5 = 0$.

Check your progress

Problem 1: Using Simplex method solve the following LP problem

Maximize, $Z = 107x_1 + x_2 + 2x_3$

Subject to,

$14x_1 + x_2 - 6x_3 + 3x_4 = 7$

$16x_1 + x_2 - 6x_3 \le 5$

$3x_1 - x_2 - x_3 \le 0$

$x_1, x_2, x_3, x_4 \ge 0$

Answer: Unbounded Solution

Problem 2: Using Simplex method solve the following LP problem

Maximize, $Z = 3x_1 + 2x_2$

Subject to,

$x_1 + x_2 \le 4$

$x_1 - x_2 \le 2$

$3x_1 - x_2 - x_3 \le 0$

$x_1, x_2 \ge 0$

Answer: $x_1 = 3, x_2 = 1, Z = 11$

2.5 *SUMMARY*

The simplex method is a powerful algorithm used to solve linear programming problems by iteratively improving upon a feasible solution until an optimal solution is reached. The overall summarization of this units are as follows:

- The necessary and sufficient condition for the existence and non-degeneracy of all the basic solutions of $Ax = b$ is that every set of m columns of the augmented matrix $[A, b]$ is linearly independent.

- The necessary and sufficient condition for the existence and non-degeneracy of all possible basic feasible solutions of $Ax = b$, $x \geq 0$ is the linear independent of every set of m columns of the augmented matrix [A, b], where A is the m × n coefficient matrix.
- If a linear programming problem has an optimal solution, then the optimal solution will coincide with at least one basic feasible solutions of the problem.

After completion of this Unit learners will be able to solve the given LPP by using the simplex method more effectively.

2.6 *GLOSSARY*

- Slack and Surplus variable
- Simplex Method

2.7 *REFERENCES*

- Mokhtar S. Bazaraa, John J. Jarvis and Hanif D. Sherali: *Linear Programming and Network Flows* (4th edition). John Wiley and Sons, 2010.
- Hamdy A. Taha: *Operations Research: An Introduction (10th edition). Pearson*, 2017.
- Paul R. Thie and Gerard E. Keough: *An Introduction to Linear Programming and Game Theory* (3rd edition), Wiley India Pvt. Ltd, 2014.
- Swarup, K., Gupta, P. K., & Mohan, M. (2017). An introduction to management science operation research. *Sultan Chand & Sons educational publishers, New Delhi*.
- OpenAI. (2024). *ChatGPT (August 2024 version) [Large language model]*. OpenAI. https://www.openai.com/chatgpt

2.8 *SUGGESTED READING*

- G. Hadley, *Linear Programming*, Narosa Publishing House, 2002.
- Frederick S. Hillierand Gerald J. Lieberman: *Introduction to Operations Research* (10th edition). McGraw-Hill Education, 2015.
- https://epgp.inflibnet.ac.in/Home/ViewSubject?catid=ZLCHeZEhCZ8yCri36nSF3A==

2.9 *TERMINAL QUESTION*

Long Answer Type Question:

1. Using Simplex method solve the following LP problem

Maximize, $Z = 3x_1 + 2x_2 + 5x_3$

Subject to,

$x_1 + 2x_2 + x_3 \le 430$; $3x_1 + 2x_3 \le 460$; $x_1 + 4x_3 \le 420$

$x_1, x_2, x_3 \ge 0$

2. Using Simplex method solve the following LP problem

Maximize, $Z = 2x_1 + 4x_2 + x_3 + x_4$

Subject to,

$x_1 + 3x_2 + x_4 \le 4$; $2x_1 + x_2 \le 3$; $x_2 + 4x_3 + x_4 \le 3$

$x_1, x_2, x_3, x_4 \ge 0$

3. Using Simplex method solve the following LP problem

Maximize, $Z = 4x_1 + 3x_2 + 4x_3 + 6x_4$

Subject to,

$x_1 + 2x_2 + 2x_3 + 4x_4 \le 80$; $2x_1 + 2x_3 + x_4 \le 60$; $3x_1 + 3x_2 + x_3 + x_4 \le 80$

$x_1, x_2, x_3, x_4 \ge 0$.

4. Using Simplex method solve the following LP problem

Maximize, $Z = 4x_1 + 5x_2 + 9x_3 + 11x_4$

Subject to,

$x_1 + x_2 + x_3 + x_4 \le 15$; $7x_1 + 5x_3 + 3x_3 + 2x_4 \le 120$; $3x_1 + 5x_2 + 10x_3 + 15x_4 \le 100$

$x_1, x_2, x_3, x_4 \ge 0$.

5. Using Simplex method solve the following LP problem

Maximize, $Z = 15x_1 + 6x_2 + 9x_3 + 2x_4$

Subject to,

$2x_1 + x_2 + 5x_3 + 0.6x_4 \le 10; 3x_1 + x_2 + 3x_3 + 0.25x_4 \le 12; \ 7x_1 + x_4 \le 35$

$x_1, x_2, x_3, x_4 \ge 0.$

6. Using Simplex method solve the following LP problem

Maximize, $Z = 15x_1 + 6x_2 + 9x_3 + 2x_4$

Subject to,

$2x_1 + x_2 + 5x_3 + 0.6x_4 \le 10; 3x_1 + x_2 + 3x_3 + 0.25x_4 \le 12; \ 7x_1 + x_4 \le 35$

$x_1, x_2, x_3, x_4 \ge 0.$

7. Using Simplex method solve the following LP problem

Maximize, $Z = 4x_1 + x_2 + 3x_3 + 5x_4$

Subject to,

$4x_1 - x_2 - 5x_3 - 4x_4 \le -20; 3x_1 - 2x_2 + 4x_3 + x_4 \le 10; \ 8x_1 - 3x_2 + 3x_3 + 2x_4 \le 20$

$x_1, x_2, x_3, x_4 \ge 0$

has an unbounded solution.

Short answer type question:

1. Using Simplex method solve the following LP problem

Maximize, $Z = 3x_1 + 2x_2$

Subject to,

$x_1 + x_2 \le 6; \ 2x_1 + x_2 \le 6;$

$x_1, x_2 \ge 0$

2. Using Simplex method solve the following LP problem

Maximize, $Z = 2x_1 + 3x_2$

Subject to,

$x_1 + x_2 \le 4\,;\ -x_1 + x_2 \le 1\,; x_1 + 2x_2 \le 5$

$x_1, x_2 \ge 0$

3. Using Simplex method solve the following LP problem

Maximize, $Z = 5x_1 + 3x_2$

Subject to,

$x_1 + x_2 \le 2\,;\ 5x_1 + 2x_2 \le 10\,; 3x_1 + 8x_2 \le 12$

$x_1, x_2 \ge 0$

4. Using Simplex method solve the following LP problem

Maximize, $Z = 5x_1 + 3x_2$

Subject to,

$x_1 \le 4\,;\ x_2 \le 3\,; x_1 + 2x_2 \le 18\,;\ x_1 + x_2 \le 9$

$x_1, x_2 \ge 0$

5. Using Simplex method solve the following LP problem

Maximize, $Z = x_1 + 2x_2 + 3x_3$

Subject to,

$x_1 + 2x_2 + 3x_3 \le 10\,;\ x_1 + x_2 \le 5$

$x_1, x_2 \ge 0$

Answer:

Fill in the blanks:

1: A variable which is added to the LHS of a "≤" type constraint to convert the constraint into an equality is called …………………….

2: A variable which is subtracted from the LHS of a "≥" type constraint to convert the constraint into an equality is called ………………

3: The necessary and sufficient condition for the existence and non-degeneracy of all the basic solutions of $Ax = b$ is that every set of m columns of the augmented matrix $[A, b]$ is ……..………..

2.10 *ANSWERS*

Answer of long answer type question

1: $x_1 = 0, x_2 = 100, x_3 = 230; Maximum(Z) = 1350$

2: $x_1 = 1, x_2 = 1, x_3 = 1/2, x_4 = 0; Maximum(Z) = 13/2$

3: $x_1 = 280/13, x_2 = 0, x_3 = 20/13, x_4 = 180/13; Maximum(Z) = 2280/13$

4: $x_1 = 50/7, x_2 = 0, x_3 = 55/7, x_4 = 0; Maximum(Z) = 695/7$

5: Unbounded solution.

Answer of short answer type question

1: $x_1 = 0, x_2 = 6, Maximum(Z) = 12$

2: $x_1 = 4 \; and \; x_2 = 0 \; or \; x_4 = 1 \; and \; x_2 = 2; Maximum(Z) = 8$

3: $x_1 = 2, x_2 = 0, Max\,Z = 10$

4: $x_1 = 4, x_2 = 3, Max\,Z = 29$

5: $x_1 = 1, x_2 = 2, x_3 = 1.67;\; Max\,Z = 10$

$x_1 = 1, x_2 = 0, x_3 = 3;\; Max\,Z = 25$

Answer of fill in the blank question

1: Slack Variable

2: Surplus variable

3: Linearly independent

UNIT-3: Big-*M* METHOD

CONTENTS:

3.1 *INTRODUCTION*

The Big-M method is a technique used in linear programming to solve problems with constraints that cannot be directly incorporated into the standard form. In linear programming, problems are typically formulated to maximize or minimize a linear objective function subject to linear equality and inequality constraints.

The Big-M method involves introducing a large positive constant (M) into the objective function for each constraint that needs to be converted from inequality to equality form. This constant ensures that the original problem's solution remains feasible even after converting the inequality constraints into equality constraints.

Since, both methods are used for solving linear programming problems, the simplex method focuses on iteratively improving a feasible solution to reach optimality, while the Big M method specifically addresses inequality constraints by introducing artificial variables and a large penalty constant to guide the optimization process.

Generally, solution of linear programming problem having artificial variables are evaluated by these two method:

1. Big-M Method or Method of Penalties.
2. Two-Phase Method.

3.2 *OBJECTIVE*

After reading this unit learners will be able to

- Understand the basic concept of Big-M method or Method of Penalty
- Implement the Big-M method for the solution of LPP.

3.3 *Big-M METHOD*

A non-negative variable is added to the left side of each equation in the usual form of an LPP when the choice variables, slack variables, and surplus variables are unable to pay for the original basic variables. We refer to this variable as an artificial variable.

One technique for solving LPP using artificial variables is the Big M approach. This strategy assigns a very big negative price (−M) (where M is positive) to every fake variable in the maximization type objective function. The issue can be resolved using the standard simplex approach once the artificial variable or variables have been introduced. Nevertheless, the following inferences are made from the final table when solving in simplex.

1: Providing the optimality requirement is met and there are no artificial variables left in the basis, the current solution is an optimal BFS.

2: If the optimality requirement is met and at least one artificial variable occurs in the basis at zero level, the present solution is an optimal degenerate BFS.

3: If at least one artificial variable exists in the basis at a positive level and the optimality criterion is met, then the issue has no feasible solution.

3.3.1 *Algorithm for Big-M Method*

The steps involved in applying the Big-M method generally include:

1. Convert inequality constraints to equality constraints by introducing slack variables.
2. Introduce artificial variables for any inequality constraints that have a ">=" or "=" sign, but not for those with "<=".

3. Introduce a term in the objective function for each artificial variable multiplied by a large positive constant (M). This term penalizes the objective function for the presence of artificial variables.
4. Solve the modified linear programming problem using standard techniques, such as the simplex method.
5. If any artificial variables remain positive in the optimal solution, it indicates that the original problem is infeasible.
6. If the problem is feasible, eliminate the artificial variables from the solution to obtain the optimal solution to the original problem.

The Big-M method is a widely used approach in linear programming, especially in introductory courses and textbooks, as it provides a systematic way to handle constraints of different types. However, care must be taken in choosing the value of M to avoid numerical instability or other computational issues.

Example 1: Using method of penalty (or Big M) solve the following LP problem

Maximize, $Z = 6x_1 + 4x_2$

Subject to,

$2x_1 + 3x_2 \le 30$; $3x_1 + 2x_2 \le 24$; $x_1 + x_2 \ge 3$

$x_1, x_2 \ge 0$

Is the solution unique? If it is not, then find two different solutions.

Solution: Introducing the slack variable.

Maximize, $Z = 6x_1 + 4x_2 + 0s_1 + 0s_2 + 0s_3$

Subject to,

$2x_1 + 3x_2 + s_1 = 30$

$3x_1 + 2x_2 + s_2 = 24$

$x_1 + 3x_2 - s_3 = 3$

$x_1, x_2, s_1, s_2, s_3 \ge 0$

Here, we can easily see that there is no initial basic feasible solution. So, we introduce an artificial variable $A_1 \ge 0$ in the third constraints. Then the initial basic feasible solution are,

$s_1 = 30, s_2 = 24$ and $A_1 = 3$.

Now, corresponding to the basic variables s_1, s_2 and A_1, the matrix $Y = B^{-1}A$ (where $B = I$, the identity matrix) and the net evaluation $z_j - c_j \ (j = 1,2,3,4,5,6)$ are computed, Where $c_B = (0, 0, -M)$.

So, the table will be

		c_j	6	4	0	0	0	-M
c_B	y_B	x_B	y_1	y_2	y_3	y_4	y_5	y_6
0	y_3	30	2	3	1	0	0	0
0	y_4	24	3	2	0	1	0	0
-M	y_6	3	**1**	1	0	0	-1	1
		z (= -3M)	-M-6	-M-4	0	0	M	0

In the above table we can easily see that $z_1 - c_1$ and $z_2 - c_2$ are negative. Among these two $z_1 - c_1$ has most negative value (Since M is very large), Therefore, y_1 enters the basis. Since, $Min\left\{\frac{x_{Bi}}{y_{i1}}; y_{i1} > 0\right\} = \frac{3}{1}$. This indicates y_6 leaves the basis and y_{31} becomes the leading element. Since corresponding y_6, A_1 is the artificial variable. So, we drop it from the objective function.

		c_j	6	4	0	0	0
c_B	y_B	x_B	y_1	y_2	y_3	y_4	y_5
0	y_3	24	0	1	1	0	2
0	y_4	15	0	-1	0	1	**3**
6	y_1	3	1	1	0	0	-1
		z (= 18)	0	2	0	0	-6

Since $z_5 - c_5 < 0$, y_5 enters the basis, Further, $Min\left\{\frac{x_{Bi}}{y_{i5}}; y_{i5} > 0\right\} = \frac{15}{3}$.

$\therefore y_4$ leaves the basis and y_{25} becomes the leading element.

		c_j	6	4	0	0	0
c_B	y_B	x_B	y_1	y_2	y_3	y_4	y_5
0	y_3	14	0	5/3	1	-2/3	0
0	y_5	5	0	-1/3	0	1/3	1
6	y_1	8	1	2/3	0	1/3	1
		z (= 48)	0	0	0	2	0

Since all $z_j - c_j \geq 0$. Thus, the optimal BFS of the given LPP is,

$x_1 = 8, x_2 = 0$ with max. Since $Z = 48$.

Example 2: Using method of penalty (or Big M) solve the following LP problem

Maximize, $Z = x_1 + 2x_2 + 3x_3 - x_4$

Subject to,

$x_1 + 2x_2 + 3x_3 = 15$; $2x_1 + x_2 + 5x_3 = 20$; $x_1 + 2x_2 + x_3 + x_4 = 10$

$x_1, x_2, x_3, x_4 \geq 0$

Solution: We can see from the problem's requirements that the starting **B** does not have the necessary identity column to make an identity matrix. So, we introduce artificial variables $A_1 \geq 0$ and $A_2 \geq 0$ in the first and second constraints respectively. An initial basic feasible solution, then, is

$A_1 = 15, A_2 = 20$ and $x_4 = 10$.

Now corresponding to basic variables, A_1, A_2 and x_4, the basis matrix $Y = B^{-1}A$ and the net evaluations $z_j - c_j$ $(j = 1, 2, 3, 5, 6)$ are computed, where $c_B = (-M \quad -M \quad -1)$. So the simplex is,

		c_j	6	4	0	0	0	-M
c_B	y_B	x_B	y_1	y_2	y_3	y_4	y_5	y_6
-M	y_6	15	1	2	3	0	0	1
-M	y_5	20	2	1	**5**	0	1	0
-1	y_4	10	1	2	1	1	0	0
	z	-35M-10	-3M-2	-3M-4	-8M-4	0	0	0

Since the most negative $(z_3 - c_3)$ corresponds to y_3, it enters the basis. Further, $Min\left\{\frac{x_{Bi}}{y_{i3}}; y_{i3} > 0\right\} = \frac{20}{5}$, the current basis vector y_5 leaves the basis and y_{23} becomes the leading element. As y_5 corresponds to an artificial variable A_2, we drop y_5 column from subsequent simplex tables.

c_B	y_B	x_B	y_1	y_2	y_3	y_4	y_6
-M	y_6	3	-1/5	**7/5**	0	0	1
3	y_3	4	2/5	1/5	1	0	0
-1	y_4	6	3/5	9/5	0	1	0
	z	-3M+6	$\frac{M}{5} - \frac{2}{5}$	$\frac{-7M}{5} - \frac{16}{5}$	0	0	0

Clearly, $(z_2 - c_2)$ is the only negative and hence y_5 enters the basis. Further $Min\left\{\frac{x_{Bi}}{y_{i2}}; y_{i2} > 0\right\}$ correspond to y_6. So, y_6 leaves the basis and y_{12} becomes the leading element. Again, y_6 corresponds to the artificial variable A_1 and therefore we drop the artificial column y_6 in the subsequent tables.

c_B	y_B	x_B	y_1	y_2	y_3	y_4
2	y_2	15/7	-1/7	1	0	0

3	y_3	25/7	3/7	0	1	0
-1	y_4	15/7	**6/7**	0	0	1
	z	90/7	$-6/7$	0	0	0

Clearly, $z_2 - c_2 < 0$ and, therefore, y_1 enters the basis. Further, $Min\left\{\frac{x_{Bi}}{y_{i1}}; y_{i1} > 0\right\}$ corresponds to y_4. So, y_4 leaves the basis and y_{31} becomes the leading element.

c_B	y_B	x_B	y_1	y_2	y_3	y_4
2	y_2	15/6	0	1	0	1/6
3	y_3	15/6	0	0	1	-3/6
1	y_1	15/6	1	0	0	7/6
	z	15	0	0	0	1

Since, all $z_j - c_j$ are positive, therefore, an optimum basic feasible solution has been attained. Hence, for the given LPP the optimal solution is,

Maximize, $z = 15$; $x_1 = x_2 = x_3 = 5/2$ and $x_4 = 0$

Example 3: Using method of penalty (or Big M) solve the following LP problem

Minimize, $Z = 2x_1 + 3x_2$

Subject to,

$x_1 + x_2 \geq 15$; $x_1 + 2x_2 \geq 6$

$x_1, x_2 \geq 0$

Proof: Introducing artificial and surplus variable, the given problem written in the standard form as,

Maximize $(Z^{'}) = \text{Minimize}(-Z) = -2x_1 - 3x_2 + 0x_3 + 0x_4 - Mx_5 - Mx_6$

Subject to,

$x_1 + x_2 - x_3 + x_5 = 5;\ x_1 + 2x_2 - x_4 + x_6 = 6;\ x_1, x_2, x_3, ..., x_6 \geq 0$

			$c_j \rightarrow$	-2	-3	0	0	-M	-M	Mini
c_B	B	x_B	b	a_1	a_2	a_3	a_4	a_5	a_6	Ratio
-M	a_5	x_5	5	1	1	-1	0	1	0	5/1=5
-M	a_6	x_6	6	1	2	0	-1	0	1	6/2=2
		$z_j - c_j$		-2M+2	-3M +3	M	M	0	0	
-M	a_5	x_5	2	1/2	0	-1	1/2	1	×	2/(1/2)=4
-3	a_2	x_2	3	1/2	1	0	-1/2	0	×	3/(1/2)=6
		$z_j - c_j$		(-M+1)/2	0	M	(-M+3)/2	0	×	
-2	a_1	x_1	4	1	0	-2	1	×	×	
-3	a_2	x_2	1	0	1	1	-1	×	×	
		$z_j - c_j$		0	0	1	1	×	×	

From the last iteration in above mentioned table we see that $z_j - c_j \geq 0$ for all j. Hence, the optimality is satisfied. So, the optimal solution of given LPP is $x_1 = 4, x_2 = 1$ and the corresponding $Z_{\min} = 11$

Check your progress

Problem 1: Using penalty method to solve the following LP problem

Maximize, $Z = 2x_1 + 3x_2$

Subject to,

$x_1 + 2x_2 \leq 4$

$x_1 + x_2 = 3$

$x_1, x_2 \geq 0$

Answer: $x_1 = 2, x_2 = 1$, Maximum $Z = 7$

Problem 2: Using Big-*M* method to solve the following LP problem

Minimize, $Z = 12x_1 + 20x_2$

Subject to,

$6x_1 + 8x_2 \geq 100$

$7x_1 + 12x_2 \geq 120$

$x_1, x_2 \geq 0$

Answer: $x_1 = 15, x_2 = 5/4$, Minimum $Z = 205$

3.4 *SUMMARY*

The Big-M method is a technique used in linear programming to solve problems involving artificial variables, typically in cases where constraints cannot be easily transformed into standard form. It involves adding artificial variables with a very large positive or negative coefficient, represented by "*M*," to the objective function. The purpose of these artificial variables is to facilitate finding an initial feasible solution. The algorithm then proceeds to minimize the impact of these artificial variables by driving their coefficients to zero, effectively removing them from the solution. If any artificial variables remain in the final solution with non-zero values, it indicates that the original problem has no feasible solution. The Big-*M* method is particularly useful in dealing with complex constraints and ensures that the artificial variables do not influence the optimal solution, unless they are necessary to indicate infeasibility.

3.5 *GLOSSARY*

- Big-*M* method or Method of penalty

3.6 *REFERENCES*

- Mokhtar S. Bazaraa, John J. Jarvis and Hanif D. Sherali: *Linear Programming and Network Flows* (4th edition). John Wiley and Sons, 2010.

- Hamdy A. Taha: *Operations Research: An Introduction (10th edition). Pearson*, 2017.
- Paul R. Thie and Gerard E. Keough: *An Introduction to Linear Programming and Game Theory* (3rd edition), Wiley India Pvt. Ltd, 2014.
- Swarup, K., Gupta, P. K., & Mohan, M. (2017). An introduction to management science operation research. *Sultan Chand & Sons educational publishers, New Delhi*.
- OpenAI. (2024). *ChatGPT (August 2024 version) [Large language model]*. OpenAI. https://www.openai.com/chatgpt

3.7 *SUGGESTED READING*

- G. Hadley, *Linear Programming*, Narosa Publishing House, 2002.
- Frederick S. Hillierand Gerald J. Lieberman: *Introduction to Operations Research* (10th edition). McGraw-Hill Education, 2015.
- https://epgp.inflibnet.ac.in/Home/ViewSubject?catid=ZLCHeZEhCZ8yCri36nSF3A==

3.8 *TERMINAL QUESTION*

Long Answer Type Question:

1. Using Big-*M* method to solve the following LP problem

Minimize, $Z = 5x_1 - 6x_2 - 7x_3$

Subject to,

$x_1 + 5x_2 - 3x_3 \geq 15$; $5x_1 - 6x_2 + 10x_3 \geq 0$; $x_1 + x_2 + x_3 = 5$

$x_1, x_2, x_3 \geq 0$

2. Using Big-*M* method to solve the following LP problem

Maximize, $Z = 2x_1 + x_2 + 3x_3$

Subject to,

$x_1 + x_2 + 2x_3 \leq 5$; $2x_1 + 3x_2 + 4x_3 = 12$;

$x_1, x_2, x_3 \geq 0$

3. Using Big-*M* method (Penalty method) to solve the following LP problem

Maximize, $Z = 8x_2$

Subject to the constraints;

$x_1 - x_2 \geq 0; 2x_1 + 3x_2 \leq 6; \ 3x_1 + 3x_2 + x_3 + x_4 \leq 80$

x_1, x_2 are unrestricted

Short answer type question:

1. Using Big-*M* method (Penalty method) to solve the following LP problem

Maximize, $Z = 2x_1 + x_2$

Subject to,

$3x_1 + x_2 = 3; 4x_1 + 3x_2 \geq 6; x_1 + 2x_2 \leq 3;$

$x_1, x_2 \geq 0$

2. Using Big-*M* method (Penalty method) to solve the following LP problem

Maximize, $Z = 3x_1 + 2x_2 + x_3$

Subject to,

$2x_1 + x_2 + x_3 = 12; \ 3x_1 + 4x_3 = 11;$

$x_2, x_3 \geq 0$ and x_1 is unrestricted.

Fill in the blanks:

1: Linear programming is a technique of finding the …………………….

2: Any solution to a linear programming problem which also satisfies the non-negative notification of the problem has ………………

3.9 *ANSWERS*

Answer of long answer type question

1: No feasible solution.

2: $x_1 = 3, x_2 = 2, x_3 = 0; Maximum(Z) = 8$

3: $x_1 = -6/5, x_2 = -6/5; Maximum(Z) = -48/5$

Answer of short answer type question

1: $x_1 = 3/5, x_2 = 6/5, Minimum(Z) = 12/5$

2: $x_1 = 11/3, x_2 = 0, x_3 = 14/3, Maximize(Z) = 47/3$

Answer of fill in the blank question

1: Optimal value

2: Feasible solution

UNIT-4: TWO PHASE METHOD AND DEGENERACY

CONTENTS:

4.1 *INTRODUCTION*

The Two-Phase Method and the Simplex Method are both techniques used to solve linear programming problems, but they differ in their approach and application:

1. **Objective**:
   a. **Two-Phase Method**: It is designed to handle linear programming problems that are not initially in the standard form. Its primary objective is to convert the original problem into an equivalent problem that can be solved using the simplex method.
   b. **Simplex Method**: This method is used to solve linear programming problems that are already in the standard form. Its objective is to iteratively improve a feasible solution until an optimal solution is reached.
2. **Phases**:

a. **Two-Phase Method**: It consists of two phases. Phase I involves introducing artificial variables to convert the original problem into a form suitable for the simplex method. Phase II eliminates these artificial variables and applies the simplex method to solve the original linear programming problem.

b. **Simplex Method**: It doesn't have distinct phases like the Two-Phase Method. Instead, it directly starts with a feasible solution and iteratively improves it until an optimal solution is found.

3. Artificial Variables:

a. **Two-Phase Method**: Artificial variables are introduced in Phase I to help convert the problem into a standard form. These variables are eliminated in Phase II after determining feasibility.

b. **Simplex Method**: It does not involve artificial variables. Instead, it starts with an initial feasible solution and iteratively improves it by moving along the edges of the feasible region.

4. Application:

a. **Two-Phase Method**: It is used when the given linear programming problem is not initially in the standard form, requiring the introduction of artificial variables and subsequent conversion.

b. **Simplex Method**: It is applied when the linear programming problem is already in the standard form and does not require additional transformations.

5. Computational Complexity:

a. **Two-Phase Method**: It typically involves more computational steps compared to the Simplex Method, especially due to the two-phase approach and the introduction and elimination of artificial variables.

b. **Simplex Method**: It is generally more computationally efficient, especially for problems already in the standard form, as it directly optimizes the objective function without additional transformations.

Degeneracy is an important concept in linear programming, and understanding its causes and effects is crucial for effectively solving and interpreting optimization problems. Proper handling of degeneracy can improve the efficiency and accuracy of linear programming algorithms.

4.2 *OBJECTIVE*

After reading this unit learners will be able to

- Understand the basic concept of Two-phase method in LPP
- Understand the concept of Degeneracy in LPP.

4.3 *TWO PHASE METHOD*

The Two-Phase Method is an algorithm used to solve linear programming problems, particularly those that are not initially in standard form. It's a systematic approach to convert such problems into standard form and then solve them using the simplex method. Here's a detailed explanation of the Two-Phase Method:

1. **Problem Setup**:

- Begin with a linear programming problem that may not be expressed in the standard form, i.e., it may contain inequalities, non-negativity constraints, or objective functions that are not in the form of maximization or minimization.

2. **Phase I**:

- **Objective**: The objective of Phase I is to convert the original problem into an equivalent problem that can be solved using the simplex method. This involves introducing artificial variables to transform the problem into standard form.
- **Artificial Variables**: Artificial variables are introduced for each inequality constraint in the problem. These artificial variables help create an initial basic feasible solution. The objective function in Phase I aims to minimize the sum of these artificial variables.
- **Initial Solution**: The simplex method is then applied to this modified problem to find a basic feasible solution.
- **Feasibility Check**: If the minimum value of the artificial variables is zero, indicating that the original problem is feasible, the method proceeds to Phase II. If the minimum value is positive, it suggests that the original problem is infeasible.

3. **Phase II**:

- **Objective**: In Phase II, the artificial variables introduced in Phase I are eliminated, and the original objective function is reintroduced.
- **Optimization**: The simplex method is applied to optimize the original objective function while maintaining feasibility. The basic feasible solution obtained from Phase I serves as the starting point for Phase II.
- **Optimal Solution**: The optimal solution found in Phase II is the solution to the original linear programming problem.

4. Conclusion:

- Once Phase II is completed, the optimal solution provides the values of decision variables that maximize or minimize the objective function, subject to the given constraints.
- The Two-Phase Method ensures that linear programming problems can be solved even if they are not initially presented in standard form, providing a systematic approach to conversion and solution.

4.3.1 *PROBLEM SOLVBING OF TWO PHASE METHOD*

To obtain a basic feasible solution to the original L.P.P., the first part of this method involves minimizing the sum of the artificial variable, subject to the stated constraints (known as the auxiliary L.P.P.). Beginning with the fundamentally feasible solution found at the conclusion of phase 1, the second step optimizes the original objective function.
The algorithm's iterative process can be summed up as follows:
Step 1: Put the provided L.P.P. into standard form and see if there is a feasible, basic solution already in place.
(a) Proceed to phase 2 if a fundamental, feasible solution is available at the present time.
(b) Proceed to the following step if a ready, basic, and feasible solution is not available.

Phase I

Step 2: Add the artificial variable on the left side of every equation in which the initial basic variables are missing. Construct an auxiliary objective function with the goal of minimizing the overall sum of artificial variables.

Thus, the new objective is to

Minimize $z = A_1 + A_2 + ... + A_n$

i.e., Maximize $z^* = -A_1 - A_2 - ... - A_n$

where, $A_i\,(i = 1,2,...,m)$ are the non-negative artificial variables.

Step 3: Use the specially created L.P.P. and the simplex method. The least possible interaction could result in either of the following three cases:

a. $Max(Z^*) < 0$ and atleast one artificial vector appear in the optimum basis at a positive level. In this instance, there is no feasible solution for the provided problem.

b. $Max(Z^*) = 0$ and at least one artificial vector appears in the optimum basis at a zero level. In this case proceed to phase-II.

c. $Max(Z^*) = 0$ and no one artificial vector appears in the optimum basis. In this case also proceed to phase-II.

Phase II

Step 4: At this point, give each artificial variable that shows up in the basis at the zero level a zero cost, and assign the actual cost to each variable in the objective function. Now, with the specified restrictions, the simplex approach maximizes this new objective function. The modified simplex table that was created at the end of phase I is subjected to the simplex method until an optimal basic feasible solution is reached. At the conclusion of phase, I, the artificial variables that are not basic are eliminated.

Note: It is possible to completely remove artificial variables from the simplex table that do not occur in the fundamental solution.

Example 1: Using two-phase method solve the following LP problem

Maximize, $Z = 5x_1 + 3x_2$

Subject to,

$2x_1 + x_2 \leq 1$; $x_1 + 4x_2 \geq 6$;

$x_1, x_2 \geq 0$

Solution: Introducing the slack variable $s_1 \geq 0$, a surplus variables $s_2 \geq 0$ and an artificial variables $A_1 \geq 0$ in the constraints of the linear programming problem.

So, the initial basic feasible solution is; $s_1 = 1$ and $A_1 = 6$ with l_2 as the basis matrix.

Phase 1: The objective function of the auxiliary L.P.P. is to maximize $Z^* = -A_1$. Using now simplex algorithm to the auxiliary linear programming problem, the simplex table is,

Initial iteration: Dropping of y_3 and introducing of y_2.

		c_j	0	0	0	0	-1
c_B	y_B	x_B	y_1	y_2	y_3	y_4	y_5

0	y_3	1	2	**1**	1	0	0
-1	y_5	6	1	4	0	-1	1
	z_j	-	-1	-4	0	1	-1
	z_j - c_j	z (= -6)	-1	-4	0	1	0

Since $z_1 - c_1$ and $z_2 - c_2$ are negative, we choose the most negative of these, *viz.*, -4. The corresponding column vector y_2 enters the basis, Therefore, y_1 enters the basis. Further, since, $Min\left\{\frac{x_{Bi}}{y_{i2}}; y_{i2} > 0\right\} = 1$, which occurs for element y_{12}, y_3 leaves the basis.

Final iteration: Optimal solution.

c_B	y_B	x_B	y_1	y_2	y_3	y_4	y_5
0	y_2	1	2	1	1	0	0
-1	y_5	2	-7	0	-4	-1	1
		z (= -2)	7	0	4	1	0

Since all $(z_j - c_j) \geq 0$, an optimum basic feasible solution to the auxiliary L.P.P. is obtained . But max. $z^* < 0$ and an artificial variables is in the basis at a positive level. Thus, there isn't a feasible option in the original L.P.P.

Example 2: Using two-phase method solve the following LP problem

Maximize, $Z = 5x_1 - 4x_2 + 3x_3$

Subject to the constraints,

$2x_1 + x_2 - 6x_3 = 20; 6x_1 + 5x_2 + 10x_3 \leq 76;\ 8x_1 - 3x_2 + 6x_3 \leq 50;$

$x_1, x_2, x_3 \geq 0$

Solution: Introducing the slack variables $s_1 \geq 0$ and $s_2 \geq 0$, the given L.P.P. in the standard form is:

Maximize, $Z = 5x_1 - 4x_2 + 3x_3$, subject to the constraints:

$2x_1 + x_2 - 6x_3 = 20; 6x_1 + 5x_2 + 10x_3 + s_1 = 76;\ 8x_1 - 3x_2 + 6x_3 + s_2 = 50;$

$x_1, x_2, x_3, s_1, s_2 \geq 0$

In the matrix form the set of constraints is,

$$\begin{pmatrix} 2 & 1 & -6 & 0 & 0 \\ 6 & 5 & 10 & 1 & 0 \\ 8 & -3 & 6 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ s_1 \\ s_2 \end{pmatrix} = \begin{bmatrix} 20 \\ 76 \\ 50 \end{bmatrix}$$

Given that there are no identity matrix columns that can be used as the initial basis matrix. To complete the identity basis matrix, we add the necessary identity column or columns. Put in the identity column [1 0 0] as the new column y_6, in other words. Clearly, this amounts to the adding an artificial variable $A_1 \geq 0$ in the 1st constraints.

Now, an initial basic feasible solution is $A_1 = 20$, $s_1 = 76$ and $s_2 = 50$.

Phase 1: The objective function of the auxiliary L.P.P. is $z^* = -A_1$. The iterative simplex tables for the auxiliary L.P.P. are;

Initial iteration: Introduce y_1 and drop y_5.

		c_j	0	0	0	0	0	-1
c_B	y_B	x_B	y_1	y_2	y_3	y_4	y_5	y_6
-1	y_6	20	2	1	-6	0	0	1
0	y_5	76	6	5	10	1	0	0
0	y_4	50	**8**	-3	6	0	1	0
		z^* (= -20)	-2	-1	6	0	0	0

Since, $z_1 - c_1$ is negative, the column vector y_1 enters the basis. Further, since $Min\left\{\frac{x_{Bi}}{y_{i1}}; y_{i1} > 0\right\} = \frac{50}{8}$; y_5 leaves the basis. The element y_{31} (=8) becomes the leading element.

First iteration: Introduce y_2 and drop y_6.

c_B	y_B	x_B	y_1	y_2	y_3	y_4	y_5	y_6
-1	y_6	15/2	0	**7/4**	-15/2	0	-1/4	1
0	y_4	77/2	0	29/4	11/2	1	-3/4	0
0	y_1	25/4	1	-3/8	3/4	0	1/8	0
		z^* (= -15/2)	0	-7/4	15/2	0	1/4	0

Here, $z_2 - c_2$ is the only negative $z_j - c_j$. This indicates that y_2 enters the basis. Also $Min\left\{\frac{x_{B2}}{y_{i2}}; y_{i2} > 0\right\} = \frac{(15/2)}{(7/4)}$ suggested that y_6 must leave the basis, thereby y_{12} $(-7/4)$ becomes the leading element.

Final iteration: Optimum solution.

c_B	y_B	x_B	y_1	y_2	y_3	y_4	y_5	y_6
0	y_2	30/7	0	1	-30/7	0	-1/7	4/7
0	y_4	52/7	0	0	256/7	1	2/7	-29/7
0	y_1	55/7	1	0	-6/7	0	1/14	3/14
		z^* (= 0)	0	0	0	0	0	1

Since all $z_j - c_j \geq 0$ an optimum solution to the auxiliary L.P.P has been reached. Moreover, the table makes it clear that there are no artificial variables in the base.

Phase 2: Now, we consider the actual costs associated with the original variables. So, the objective function is,

$$Z = 5x_1 - 4x_2 + 3x_3 + 0.s_1 + 0.s_2$$

The iterative simplex table for this phase is:

c_B	y_B	x_B	y_1	y_2	y_3	y_4	y_5
-4	y_2	30/7	0	1	-30/7	0	-1/7
0	y_4	52/7	0	0	256/7	1	2/7
5	y_1	55/7	1	0	-6/7	0	1/14
		z^* (= 155/7)	0	0	69/7	0	13/4

Since all $z_j - c_j \geq 0$ an optimum basic feasible solution has been reached. Hence an optimum basic feasible solution to the given L.P.P. is,

$x_1 = 55/7, x_2 = 30/7, x_3 = 0;$ maximum $z = 155/7,$

Example 3: Maximize, $Z = 3x_1 - x_2$

Subject to the constraints,

$2x_1 + x_2 \geq 2; x_1 + 3x_2 \leq 2; x_2 \leq 4$

$x_1, x_2 \geq 0$

Solution: Maximize, $Z = 3x_1 - x_2$

Subject to the constraints,

$2x_1 + x_2 - s_1 + a_1 = 2$

$x_1 + 3x_2 + s_2 = 2$

$x_2 + s_3 = 4$

$x_1, x_2, s_1, s_2, s_3, a_1 \geq 0$

So, the auxiliary LPP will become

Maximize, $Z^* = 0x_1 - 0x_2 + 0s_1 + 0s_2 + 0s_3 - 1a_1$

Subject to,

$$2x_1 + x_2 - s_1 + a_1 = 2$$

$$x_1 + 3x_2 + s_2 = 2$$

$$x_2 + s_3 = 4$$

$$x_1, x_2, s_1, s_2, s_3, a_1 \geq 0$$

Phase I

		$C_j \rightarrow$	0	0	0	0	0	-1	
Basic Variables	C_B	X_B	X_1	X_2	S_1	S_2	S_3	A_1	Min ratio X_B/X_k
a_1	-1	2	[2]	1	-1	0	0	1	$1\rightarrow$
s_2	0	2	1	3	0	1	0	0	2
s_3	0	4	0	1	0	0	1	0	-
	$Z^* = -2$		$\uparrow$ -2	-1	1	0	0	0	$\leftarrow\Delta_j$
x_1	0	1	1	1/2	-1/2	0	0	x	
s_2	0	1	0	5/2	1/2	1	0	x	
s_3	0	4	0	1	0	0	1	x	
	$Z^* = 0$		0	0	0	0	0	x	$\leftarrow\Delta_j$

Since all $\Delta_j \geq 0, Max Z^* = 0$ and no artificial vector appears in the basis, we proceed to phase II.

Phase II

		$C_j \rightarrow$	3	-1	0	0	0

Basic Variables	C_B	X_B	X_1	X_2	S_1	S_2	S_3	Min ratio X_B/X_k
x_1	3	1	1	1/2	-1/2	0	0	-
s_2	0	1	0	5/2	[1/2]	1	0	2→
s_3	0	4	0	1	0	0	1	-
	Z = 3		0	5/2	↑ -3/2	0	0	←Δ_j
x_1	3	2	1	3	0	1	0	
s_1	0	2	0	5	1	2	0	
s_3	0	4	0	1	0	0	1	
	Z = 6		0	10	0	3	0	←Δ_j

Since all $\Delta_j \geq 0,$ optimal basic feasible solution is obtained.

Hence, the solution is $Max\, Z = 6, x_1 = 2, x_2 = 0$.

Example 4: Maximize, $Z = 5x_1 + 8x_2$

Subject to the constraints,

$3x_1 + 2x_2 \geq 3; x_1 + 4x_2 \geq 4; x_1 + x_2 \leq 5$

$x_1, x_2 \geq 0$

Solution: Maximize, $Z = 5x_1 + 8x_2$

Subject to the constraints,

$3x_1 + 2x_2 - s_1 + a_1 = 3$

$x_1 + 4x_2 - s_2 + a_2 = 4$

$x_1 + x_2 + s_3 = 5$

$x_1, x_2, s_1, s_2, s_3, a_1, a_2 \geq 0$

So, the auxiliary LPP will become

Maximize, $Z^* = 0x_1 + 0x_2 + 0s_1 + 0s_2 + 0s_3 - 1a_1 - 1a_2$

Subject to,

$$3x_1 + 2x_2 - s_1 + a_1 = 3$$

$$x_1 + 4x_2 - s_2 + a_2 = 4$$

$$x_1 + x_2 + s_3 = 5$$

$$x_1, x_2, s_1, s_2, s_3, a_1, a_2 \geq 0$$

Phase I

	$C_j \rightarrow$		0	0	0	0	0	-1	-1	
Basic Variables	C_B	X_B	X_1	X_2	S_1	S_2	S_3	A_1	A_2	Min ratio X_B / X_k
a_1	-1	3	3	2	-1	0	0	1	0	3/2
a_2	-1	4	1	[4]	0	-1	0	0	1	1→
s_3	0	5	1	1	0	0	1	0	0	5
	$Z^* = -7$		-4	↑ -6	1	1	0	0	0	←Δ_j
a_1	-1	1	[5/2]	0	-1	1/2	0	1	x	2/5→
x_2	0	1	1/4	1	0	-1/4	0	0	x	4
s_3	0	4	3/4	0	0	1/4	1	0	x	16/3
	$Z^* = -1$		↑ -5/2	0	1	-1/2	0	0	x	←Δ_j
x_1	0	2/5	1	0	-2/5	1/5	0	x	x	
x_2	0	9/10	0	1	1/10	-3/10	0	x	x	
s_3	0	37/10	0	0	3/10	1/10	1	x	x	
	$Z^* = 0$		0	0	0	0	0	x	x	←Δ_j

Since all $\Delta_j \geq 0$, $Max Z^* = 0$ and no artificial vector appears in the basis, we proceed to phase II.

Phase II

	$C_j \rightarrow$		5	8	0	0	0	
Basic Variables	C_B	X_B	X_1	X_2	S_1	S_2	S_3	Min ratio X_B / X_k
x_1	5	2/5	1	0	-2/5	[1/5]	0	2→
x_2	8	9/10	0	1	1/10	-3/10	0	-
s_3	0	37/10	0	0	3/10	1/10	1	37

	Z = 46/5		0	0	-6/5	↑ -7/5	0	←Δ_j
s_2	0	2	5	0	-2	1	0	-
x_2	8	3/2	3/2	1	-1/2	0	0	-
s_3	0	7/2	-1/2	0	$\boxed{1/2}$	0	1	7→
	Z = 12		7	0	↑ -4	0	0	←Δ_j
s_2	0	16	3	0	0	1	2	
x_2	8	5	1	1	0	0	1/2	
s_1	0	7	-1	0	1	0	2	
	Z = 40		3	0	0	0	4	

Since all $\Delta_j \geq 0,$ optimal basic feasible solution is obtained. Therefore, the solution is,

$Max\,Z = 40, x_1 = 0, x_2 = 5$

4.4 *DEGENRACY IN LINEAR PROGRAMMING*

In this section of linear programming, degeneracy occurs when a feasible solution has more than one way to be optimal, or more technically, when the number of basic variables is less than the number of constraints at a basic feasible solution. Here's a detailed look at degeneracy in linear programming:

1. Understanding Degeneracy

Definition:

- **Degeneracy at a Vertex:** In a linear programming problem, a vertex of the feasible region is degenerate if there are more constraints (hyperplanes) intersecting at that vertex than the number of dimensions (basic variables).
- **Degeneracy in the Simplex Method:** A basic feasible solution (BFS) is degenerate if one or more of the basic variables are zero.

2. Causes of Degeneracy

- **Redundant Constraints:** Extra constraints that do not change the feasible region but increase the number of intersections.
- **Multiple Optimal Solutions:** When the objective function is parallel to a constraint boundary, leading to multiple solutions along that boundary.

3. Implications of Degeneracy

- **Cycling:** In the simplex method, degeneracy can cause the algorithm to revisit the same BFS repeatedly, potentially leading to an infinite loop (cycling).
- **Stalling:** The simplex method might make a pivot that does not improve the objective function, causing the algorithm to "stall" and take longer to find the optimal solution.

4. Handling Degeneracy

Anti-Cycling Rules:

- **Bland's Rule:** Choose the entering and leaving variables using a fixed order to prevent cycling.
- **Lexicographic Ordering:** Maintain a lexicographic ordering of the variables to ensure progress in each step.

Perturbation Techniques:

- Slightly modify the right-hand side of the constraints to break ties and remove degeneracy artificially.

Interior-Point Methods:

- These methods approach the optimal solution from within the feasible region rather than along the edges, avoiding degeneracy issues inherent in vertex-based methods like the simplex algorithm.

Degeneracy: Generativity is the property of obtaining a degenerate fundamental feasible solution in a linear programming problem.

In an L.P.P., degeneracy can occur (i) at the beginning and (ii) at any point during the subsequence iteration.

In case (i), every basic variable in the first basic feasible solution is zero. In case (ii), however, multiple vectors are allowed to exit the basis at any time during a simplex method iteration. As a result, the subsequent simplex iteration yields a degenerate solution where every basic variable is zero. This implies that the objective function's value might not increase in the ensuing iterations. Therefore, without enhancing the answer, the same simplex iteration subsequence can be repeated indefinitely. We call this idea "cycling."

Generally speaking, degeneracy is not problematic—that is, unless cycling happens. If there is a tie in the replacement ratios, it usually suffices to choose a row at random. However, by following these guidelines, the number of iterations needed to reach the optimal can be reduced.

(*i*) Using the matching positive elements of the entering column vector, go from left to right to divide the coefficients of basic variables (element of the column vector of the basic matrix) in the simplex table where degeneracy is identified.

(*ii*) The corresponding current basis vector departs the basis and the row with the least ratio, measured from left to right column wise, becomes the pivot row.

Example 4: Maximize, $Z = 22x_1 + 30x_2 + 25x_3$

Subject to the constraints,

$2x_1 + 2x_2 \leq 100;\ 2x_1 + x_2 + x_3 \leq 100,\ x_1 + 2x_2 + 2x_3 \leq 100;$

$x_1, x_2, x_3 \geq 0$

Solution: By introducing slack variables $s_1 \geq 0, s_2 \geq 0$ and $s_3 \geq 0$ in the respective inequalities, the set of constraints can be written as $Ax = b$, where

$$A = \begin{bmatrix} 2 & 2 & 0 & 1 & 0 & 0 \\ 2 & 1 & 1 & 0 & 1 & 0 \\ 1 & 2 & 2 & 0 & 0 & 1 \end{bmatrix}, b = \begin{bmatrix} 100 \\ 100 \\ 100 \end{bmatrix} \text{ and } X = \begin{bmatrix} x_1 & x_2 & x_3 & s_1 & s_2 & s_3 \end{bmatrix}$$

An obvious initial (starting) basic feasible solution in $x_B = B^{-1}b$, where $x_B = \begin{bmatrix} s_1 & s_2 & s_3 \end{bmatrix}$, $B = I_3$ and $b = [100\ 100\ 100]$.

$$x_B = I^{-1}b = Ib \text{ gives } \begin{bmatrix} s_1 & s_2 & s_3 \end{bmatrix} = [100\ 100\ 100]$$

Using now simplex method, the iterative simplex table are:

Initial iteration: Introduce y_2 and drop y_6

		c_j	22	30	25	0	0	-0
c_B	y_B	x_B	y_1	y_2	y_3	y_4	y_5	y_6

0	y_4	100	2	2	0	1	0	0
0	y_5	100	2	1	1	0	1	0
0	y_6	100	1	**2**	2	0	0	1
	z	0	-22	-30	-25	0	0	0

Since, $z_2 - c_2$ is the most negative $z_j - c_j$, y_2 enters the basis. Further, $Min\left\{\frac{x_{Bi}}{y_{i2}}; y_{i2} > 0\right\} = 50$ occurs for the element y_{12} and y_{32}. Thus, there is tie among the ratios in the first and third rows, *i.e*. among the basis vectors y_4 and y_6. To obtain the unique current basis vector that will leave the basis, we compute the ratios $\left\{\frac{y_{ij}}{y_{i2}}; y_{i2} > 0\right\}$ instead of $\left\{\frac{x_{Bi}}{y_{i2}}; y_{i2} > 0\right\}$ for those column vector which are in the basis. Here, since y_4, y_5 and y_6 are in the basis and there is a tie among y_4 and y_6 for leaving the basis, we write the coefficients (elements) from above table:

	y_4	y_5	y_6
y_4	1	0	0
y_6	0	0	1

Dividing these coefficients by the corresponding element of the entering column, i.e., of y_2, we obtain the following ratios:

	y_4	y_5	y_6
y_4	1/2	0/2	0/2
y_6	0/2	0/2	1/2

On comparing of ratios in the first column y_6 – row yields the smallest ratio and hence y_6 leaves the basis.

First iteration: Introduce y_1 and drop y_4

c_B	y_B	x_B	y_1	y_2	y_3	y_4	y_5	y_6
0	y_4	0	**1**	0	-2	1	0	-1
0	y_5	50	3/2	0	0	0	1	-1/2
30	y_2	50	1/2	1	1	0	0	1/2
	z	1500	-7	0	5	0	0	15

It is apparent from the above table that, $(z_1 - c_1) < 0$ and therefore y_1 enters the basis. Further, since $Min\left\{\frac{x_{Bi}}{y_{i1}}; y_{i1} > 0\right\} = \frac{0}{1} = 0$, on the current basis vector y_4 leaves the basis and y_{11} becomes the leading element.

Second iteration: Introduce y_3 and drop y_5.

c_B	y_B	x_B	y_1	y_2	y_3	y_4	y_5	y_6
22	y_1	0	1	0	-2	1	0	-1
0	y_5	50	0	0	**3**	-3/2	1	-1
30	y_2	50	0	1	2	-1/2	0	1
	z	1,500	0	0	-9	7	0	8

Clearly, the solution is still not optimum, since $(z_3 - c_3) < 0$. So, y_3 enters the basis. Further, since $Min\left\{\frac{x_{Bi}}{y_{i3}}; y_{i3} > 0\right\} = \frac{50}{3}$, the current basis vector y_5 leaves the basis and y_{23} becomes the leading element.

Final iteration:

c_B	y_B	x_B	y_1	y_2	y_3	y_4	y_5	y_6
22	y_1	100/3	1	0	0	0	2/3	-1/3
25	y_3	50/3	0	0	1	-1/2	1/3	1/3
30	y_2	50/3	0	1	0	1/2	-2/3	1/3
	z	1,650	0	0	0	5/2	3	11

Since, all $(z_j - c_j) \geq 0$, an optimum basic feasible solution is,

$x_1 = 100/3, x_1 = 100/3, x_3 = 50/3$ and maximum $z = 1{,}650$.

Check your progress

Problem 1: Using two phase method to solve the following LP problem

Maximize, $Z = 10x_1 + 20x_2$

Subject to the constraint,

$2x_1 + x_2 = 1$

$x_1 + 2x_2 = 5$

$x_1, x_2 \geq 0$

Answer: $x_1 = 0, x_2 = 3$, Maximum $z = 60$

Problem 2: Using two phase method to solve the following LP problem

Minimize, $Z = 2x_1 + 4x_2$

Subject to, $2x_1 + x_2 \geq 14$; $x_1 + 3x_2 \geq 18$; $x_1 + x_2 \geq 12$; $x_1, x_2 \geq 0$

Answer: $x_1 = 18, x_2 = 0$, Minimum $z = 36$

4.5 SUMMARY

In summary, while both methods are used for solving linear programming problems, the Two-Phase Method is suited for problems not initially in the standard form, requiring additional transformations, while the Simplex Method is more efficient for problems already in the standard form.

In summary, the Two-Phase Method is a powerful tool for solving linear programming problems by first converting them into standard form using artificial variables and then applying the simplex method to find the optimal solution.

Degeneracy in linear programming is a common occurrence, especially in large and complex problems. While it can complicate the solution process by causing cycling or stalling, several strategies like Bland's Rule, perturbation techniques, and the use of interior-point methods effectively address these issues. Understanding and handling degeneracy is crucial for efficient and accurate linear programming solutions.

4.6 GLOSSARY

- Two-phase method
- Degeneracy in linear programming

4.7 REFERENCES

- Mokhtar S. Bazaraa, John J. Jarvis and Hanif D. Sherali: *Linear Programming and Network Flows* (4th edition). John Wiley and Sons, 2010.
- Hamdy A. Taha: *Operations Research: An Introduction (10th edition). Pearson*, 2017.
- Paul R. Thie and Gerard E. Keough: *An Introduction to Linear Programming and Game Theory* (3rd edition), Wiley India Pvt. Ltd, 2014.
- Swarup, K., Gupta, P. K., & Mohan, M. (2017). An introduction to management science operation research. *Sultan Chand & Sons educational publishers, New Delhi*.
- OpenAI. (2024). *ChatGPT (August 2024 version) [Large language model]*. OpenAI. https://www.openai.com/chatgpt

4.8 SUGGESTED READING

- G. Hadley, *Linear Programming*, Narosa Publishing House, 2002.
- Frederick S. Hillierand Gerald J. Lieberman: *Introduction to Operations Research* (10th edition). McGraw-Hill Education, 2015.

- https://epgp.inflibnet.ac.in/Home/ViewSubject?catid=ZLCHeZEhCZ8yCri36nSF3A==

4.9 *TERMINAL QUESTION*

Long Answer Type Question:

1. Solve the L.P.P.

Maximize, $z = 5x_1 - 2x_2 + 3x_3$

Subject to,

$2x_1 + 2x_2 - x_3 \geq 2$; $3x_1 - 4x_2 \leq 3$; $x_2 + 3x_3 \leq 5$;

$x_1, x_2, x_3 \geq 0$

2. Solve the L.P.P.

Maximize, $z = x_1 + 1.5x_2 + 2x_3 + 5x_4$

Subject to,

$3x_1 + 2x_2 + 4x_3 + x_4 \leq 6$; $2x_1 + x_2 + x_3 + 5x_4 \leq 4$; $2x_1 + 6x_2 - 8x_3 + 4x_4 \leq 5$;

$x_1 + 3x_2 - 4x_3 + 3x_4 = 0$; $x_1, x_2, x_3 \geq 0$

3. Using Two-phase method

Maximize, $z = x_1 + 2x_2 + 3x_3$

Subject to,

$x_1 - x_2 + x_3 \geq 4$; $x_1 + x_2 + 2x_3 \leq 8$; $x_1 - x_3 \geq 2$

$x_1, x_2, x_3 \geq 0$

4. Using Two-phase method

Maximize, $z = 12x_1 + 15x_2 + 9x_3$

Subject to,

$8x_1 + 16x_2 + 12x_3 \le 250$; $4x_1 + 8x_2 + 10x_3 \ge 80$; $7x_1 + 9x_2 + 8x_3 = 105$

$x_1, x_2, x_3 \ge 0$

Short answer type question:

1: Solve the L.P.P.

Maximize, $z = 3x_1 + 2x_2 + 3x_3$

Subject to,

$2x_1 + x_2 + x_3 \le 2$; $3x_1 + 4x_2 + 2x_3 \ge 8$;

$x_1, x_2, x_3 \ge 0$

2: Using two-phase method solve the following L.P.P.

Minimize, $z = 2x_1 + 4x_2$

Subject to,

$2x_1 + x_2 \ge 14$; $x_1 + 3x_2 \ge 18$; $x_1 + x_2 \ge 12$; $x_1, x_2 \ge 0$

3: Using two-phase method solve the following L.P.P.

Minimize, $z = 3x_1 - x_2$

Subject to, $2x_1 + x_2 \ge 2$; $x_1 + 3x_2 \le 2$; $x_2 \le 4$; $x_1, x_2 \ge 0$

4: Using two-phase method solve the following L.P.P.

Maximize, $z = 5x_1 + 8x_2$

Subject to,

$3x_1 + 2x_2 \ge 3$; $x_1 + 4x_2 \ge 4$; $x_1 + x_2 \le 5$; $x_1, x_2 \ge 0$

5. Using two-phase method solve the following L.P.P.

Minimize, $z = x_1 + x_2 + x_3$

Subject to, $x_1 - 3x_2 + 4x_3 = 5;\ x_1 - 2x_2 \leq 3;\ 2x_2 + x_3 \geq 4;\ x_1, x_2 \geq 0$

Objective type question:

1: What is the purpose of the Two-Phase Method in linear programming?

a) To find the optimal solution directly

b) To handle problems where the initial basic feasible solution is not readily apparent

c) To maximize the objective function

d) To minimize the objective function

2: In the first phase of the Two-Phase Method, the objective function is:

a) The original objective function

b) An artificial objective function, usually the sum of artificial variables

c) A constant value

d) Unchanged

3: Which of the following is introduced in the first phase of the Two-Phase Method?

a) Slack variables

b) Surplus variables

c) Artificial variables

d) All of the above

4: If the minimum value of the artificial objective function at the end of the first phase is zero, this indicates:

a) The original problem has no feasible solution

b) The original problem is unbounded

c) A feasible solution to the original problem has been found

d) The problem needs to be reformulated

5: What happens if the artificial variables are still in the basis at the end of Phase 1?

a) The original problem has multiple optimal solutions

b) The original problem is infeasible

c) The original problem is unbounded

d) The artificial variables are ignored in Phase 2

6: In Phase 2 of the Two-Phase Method, what is done after removing the artificial variables?

a) The original objective function is optimized using the feasible basis found in Phase 1

b) The process is restarted from Phase 1

c) New artificial variables are introduced

d) The solution is checked for optimality and feasibility

7: Why are artificial variables introduced in the Two-Phase Method?

a) To convert inequalities into equalities

b) To provide an initial basic feasible solution when one is not apparent

c) To increase the complexity of the problem

d) To ensure the problem is bounded

8: Which of the following statements is true regarding the Two-Phase Method?

a) It guarantees an optimal solution in all cases

b) It is used when the primal problem has a readily available basic feasible solution

c) The second phase deals with the original linear programming problem after feasibility is ensured in the first phase

d) It is only applicable to problems with all constraints as equalities

4.10 *ANSWERS*

Answer of long answer type question

Answer 1: $x_1 = 23/3, x_2 = 0, x_3 = 5$; Maximum $z = 85/3$

2: $x_1 = 0; x_2 = 1.2; x_3 = 0.9; x_4 = 0;$; Maximum $z = 3.6$

3: $x_1 = 18/5, x_2 = 6/5, x_3 = 8/5$; Maximum $z = 108$

4: $x_1 = 6, x_2 = 7, x_3 = 0$; Maximum $z = 177$

Answer of short answer type question

Answer 1: $x_1 = 0, x_2 = 2, x_3 = 0$; Maximum $z = 4$

2: $x_1 = 18, x_2 = 0,$; Minimum $z = 36$

3: $x_1 = 3, x_2 = 0,$; Maximum $z = 9$

4: $x_1 = 0, x_2 = 5,$; Maximum $z = 40$

5: $x_1 = 0, x_2 = 5,$; Maximum $z = 40$

Answer of objective type question

Answer 1:	b)	**2:**	b)	**3:**	c)
4:	c)	**5:**	b)	**6:**	a)
7:	b)	**8:**	c)		

UNIT-5: DUALITY

CONTENTS:

5.1 *INTRODUCTION*

Duality is a fundamental concept in linear programming (LP) that connects a given optimization problem (the "primal" problem) with another related optimization problem (the "dual" problem). The solutions to these problems provide important insights into the structure of the original problem, and duality theory has many applications in economics, engineering, and operations research.

Duality in linear programming refers to a situation where every linear programming problem (the **primal problem**) has a corresponding dual problem. The solutions to these problems provide insights into each other, and solving one can give valuable information about the other. Duality is central to the theoretical foundation of linear programming and has practical implications for sensitivity analysis and economic interpretation.

Here is a detailed explanation of the duality concept in linear programming:

5.2 *OBJECTIVE*

After reading this unit learners will be able to

- Understand the basic concept of Duality.
- Solve the LPP by using duality.

5.3 *PRIMAL PROBLEM*

Consider the standard form of a linear programming problem:

Primal LP: Minimize $c^T x$ ……………. (1)

Subject to $Ax \geq b$

$x \geq 0$

Where:

- x is the vector of decision variables.
- c is the vector of coefficients for the objective function.
- A is the matrix of coefficients for the constraints.
- b is the vector of constants on the right-hand side of the constraints.

5.4 *DUAL PROBLEM*

The concept of duality in Linear Programming Problems (LPP) is a fundamental aspect of optimization theory. The dual problem provides deep insights into the structure of the original (or primal) problem and can often be used to derive bounds on the optimal value of the objective function.

Here are some key points about the dual problem in LPP:

1. Duality Principle:

- For every linear programming problem, known as the primal problem, there exists a corresponding dual problem.
- The solutions to the dual problem provide valuable information about the primal problem and vice versa.

2. Fomulation:

Given a primal problem in the standard form:

Maximize $c^T x$

Subject to, $Ax \le b,\ x \ge 0$

The corresponding dual problem is:

Minimize $b^T y$

Subject to, $A^T y \ge c,\ y \ge 0$

Here, *A* is the matrix of coefficients, *c* and *b* are vectors, *x* and *y* are the variables for the primal and dual problems, respectively.

The dual of the above primal problem (1) is formulated as follows:

Dual LP: Maximize $b^T y$

Subject to $A^T y \le c$

$y \ge 0$

Where:

- y is the vector of decision variables for the dual problem.
- A^T is the transpose of matrix A.
- *b* is the same vector as in the primal problem.
- *c* is the same vector as in the primal problem.

Remarks: One can readily detect the following from the definitions above:

(a) There is a dual variable for each primal constraint.

(b) There is a dual constraint for each primal variable.

(c) The primal and dual variable coefficients in the constraints are same except that they are transposed; i.e., the columns in the primal coefficient matrix becomes the rows in the dual coefficient matrix.

(d) While the number of primal variables and the number of dual constraints are exactly equal, whereas the number of dual variables is exactly equal to the number of primal constraints.

(e) The right-hand side constants of the dual constraints become the objective coefficients of the primal problem, whereas the objective coefficients of the primal variables become the right-hand side constants of the dual constraints.

The following table can be used to summarize information about the dual variables' signs, the type of restrictions, and the primal-dual objective:

Standard primal objective	Dual		
	Objective	Constraints	Variables
Maximization	Minimization	$\geq$	Unrestricted
Minimization	Maximization	$\leq$	Unrestricted

5.5 *STEP-WISE PROCEDURE FOR FORMULATING DUAL PROBLEM*

The process of formulating a prime-dual pair involves several steps:

Step 1: In standard form, solve the given linear programming problem. Think of it as the primal problem.

Step 2: Determine the factors that will be applied to the dual problem. These variables have the same number as the constraint equations in the primal.

Step 3: Using the constants on the right side of the primal restrictions, write out the objective function of the dual.

The dual will be a minimization problem if the primal problem is of the maximization type, and vice versa.

Step 4: Write the constraints for the dual problem using the dual variable found in Step 2.

(a) If the primal is a maximization problem, the dual constraints must be all of $\geq$ type. If the primal is a minimization problem, the dual constraints must be all of $\leq$ type.

(b) The dual constraints' row coefficients are derived from the primal constraints' column coefficients.

(c) The dual constraints' constants on the right side are the fundamental objective function's coefficients.

(d) It is defined that the dual variables have an unrestricted sign.

Step 5: Using steps 3 and 4, write down the dual of the given L.P.P.

Note: It is never required to take into account the dual constraints related to an artificial variable since, in the standard form of the primal, the dual constraint relating to an artificial variable is always redundant.

Remark 1: Primal-dual pairs are symmetric if the given linear programming problem is in its canonical form.

2: The primal-dual pair is considered unsymmetric if the provided linear programming problem is in its standard form.

Solved Example

Example 2: Find the dual of the following linear programming problem.

Maximize $z = 5x_1 + 3x_2$, subject to the constraints

$3x_1 + 5x_2 \le 15,\ 5x_1 + 2x_2 \le 10,\ x_1 \ge 0$ and $x_2 \ge 0$

Solution: Standard primal: Introducing slack variables $s_1, s_2 \ge 0$, the standard linear programming problem is:

Maximize $z = 5x_1 + 3x_2 + 0.s_1 + 0.s_2$, subject to the constraints

$3x_1 + 5x_2 + s_1 + 0.s_2 = 15, 5x_1 + 2x_2 + 0.s_1 + s_2 = 10, x_1, x_2, s_1, s_2 \ge 0$

Dual: Let w_1 and w_2 be the dual variables corresponding to the primal constraints. Then, the dual problem will be:

Minimize $z^* = 5w_1 + 10w_2$, subject to the constraints:

$3w_1 + 5w_2 \ge 5, 5w_1 + 2w_2 \ge 3$

$$\left.\begin{aligned} w_1 + 0.w_2 \ge 0 \\ 0.w_1 + w_2 \ge 0 \end{aligned}\right\} \Rightarrow w_1 \ge 0 \ and \ w_2 \ge 0 \text{ unrestricted (redundant)}$$

Here w_1 *and* w_2 unrestricted (redundant).

The dual variables w_1 *and* w_2 unrestricted" are dominated by $w_1 \ge 0$ *and* $w_2 \ge 0$. Eliminating redundancy, the restricted variables are $w_1 \ge 0$ *and* $w_2 \ge 0$.

Example 3: Find the dual of the following linear programming problem.

Minimize $z = 4x_1 + 6x_2 + 18x_3$, subject to the constraints

$x_1 + 3x_2 \geq 3,\ \ x_2 + 2x_3 \geq 5,\ \ x_1, x_2, x_3 \geq 0$.

Solution: Standard primal: Introducing slack variables $s_1, s_2 \geq 0$, the standard linear programming problem is:

Minimize $z = 4x_1 + 6x_2 + 18x_3 + 0.s_1 + 0.s_2$, subject to the constraints

Dual: Let w_1 and w_2 be the dual variables corresponding to the primal constraints. Then, the dual problem will be:

Minimize $z^* = 3w_1 + 5w_2$, subject to the constraints:

$w_1 + 0.w_2 \leq 4, 3w_1 + w_2 \leq 6,\ 0.w_1 + 2w_2 \leq 18$

$$\left.\begin{array}{l} -w_1 + 0.w_2 \leq 0 \\ 0.w_1 - w_2 \leq 0 \end{array}\right\} \Rightarrow w_1 \geq 0 \ and \ w_2 \geq 0 \text{ unrestricted (redundant)}$$

Eliminating redundancy, the dual problem is:

Maximize $z* = 3w_1 + 5w_2$ subject to the constraints:

$w_1 \leq 4, 3w_1 + w_2 \leq 6, 2w_2 \leq 18;\ w_1 \geq 0 \ and \ w_2 \geq 0.$

Example 4: Find the dual of the following linear programming problem.

Minimize $z = 3x_1 - 2x_2 + 4x_3$, subject to the constraints

$$3x_1 + 5x_2 + 4x_3 \geq 7,\ \ 6x_1 + x_2 + 3x_3 \geq 4,$$
$$7x_1 - 2x_2 - x_3 \leq 10,\ \ x_1 - 2x_2 + 5x_3 \geq 3,\ \ 4x_1 + 7x_2 - 2x_3 \geq 2, .$$
$$x_1, x_2, x_3 \geq 0$$

Solution: Introducing the slack variable $s_3 \geq 0$ surplus variables $s_1 \geq 0, s_2 \geq 0, s_4 \geq 0, s_5 \geq 0$.

Minimize: $z = 3x_1 - 2x_2 + 4x_3 + 0.s_1 + 0.s_2 + 0.s_4 + 0.s_5$

Subject to constraint, $3x_1 - 2x_2 + 4x_3 - s_1 = 7$

$6x_1 + x_2 + 3x_3 - s_2 = 4$

$7x_1 - 2x_2 - x_3 + s_3 = 10$

$x_1 - 2x_2 + 5x_3 - s_4 = 3$

$4x_1 + 7x_2 - 2x_3 - s_5 = 2$

$x_1, x_2, x_3, s_1, s_2, s_3, s_4, s_5 \geq 0$

Dual: If $w_j\,(j = 1, 2, 3, 4, 5)$ are the dual variables corresponding to mentioned five primal constraints, So, the dual of the given L.P.P. will be;

Maximize $z^* = 7w_1 + 4w_2 + 10w_3 + 3w_4 + 2w_5$, subject to the constraints:

$3w_1 + 6w_2 + 7w_3 + w_4 + 4w_5 \leq 3$

$5w_1 + w_2 - 2w_3 - 2w_4 + 7w_5 \leq -2$

$4w_1 + 3w_2 - w_3 + 5w_4 - 2w_5 \leq 4$

$-w_1 \leq 0, -w_2 \leq 0, w_3 \leq 0, -w_4 \leq 0, -w_5 \leq 0$

$w_j\ (j = 1,2,3,4,5)$ are unrestricted in sign.

Hence, after eliminating the redundancy, the dual variables are:
$w_1 \geq 0, w_2 \geq 0, w_3 \leq 0, w_4 \geq 0,$ *and* $w_5 \geq 0$

Example 5: Find the dual of the following linear programming problem.

Minimize $z = x_1 - 3x_2 - 2x_3$, subject to the constraints

$3x_1 - x_2 + 2x_3 \leq 7,\ 2x_1 - 4x_2 \geq 12, -4x_1 + 3x_2 + 8x_3 = 10\ x_1, x_2 \geq 0$ and x_3 is unrestricted.

Solution: Initially, we introduced the slack and surplus variable $s_1 \geq 0$ and $s_2 \geq 0$ respectively, the primal problem is restated as,

Minimize $z = cx$; subject to the constraints: $Ax = b, x \geq 0$

Where $x = [x_1, x_2, x_3', x_3'', s_1, s_2]$, $c = [1, -3, -2, 2, 0, 0], b = [7, 12, 10]$ and

$$A = \begin{pmatrix} 3 & -1 & 2 & -2 & 1 & 0 \\ 2 & -4 & 0 & 0 & 0 & -1 \\ -4 & 3 & 8 & -8 & 0 & 0 \end{pmatrix}, \text{ when } x_3 = x_3' - x_3''$$

Dual: If $w = (w_1, w_2, w_3)$ are the dual variables , then the dual of the given prima is

Maximize $z^* = 7w_1 + 12w_2 + 10w_3$ subject to the

$3w_1 + 2w_2 - 4w_3 \leq 1$

$-w_1 - 4w_2 + 3w_3 \leq -3 \Rightarrow w_1 + 4w_2 - 3w_3 \geq 3$

$$\left.\begin{array}{l} 2w_1 + 8w_3 \leq -2 \\ -2w_1 - 8w_3 \leq 2 \end{array}\right\} \Rightarrow -2w_1 - 8w_2 = 2$$

$$\left.\begin{array}{l} w_1 \leq 0 \\ -w_2 \leq 0 \end{array}\right\} \Rightarrow w_1 \leq 0 \ \ and \ \ w_2 \geq 0$$

Where w_1, w_2 *and* w_3 unrestricted.

Eliminating redundancy, dual variables are $w_1 \leq 0, w_2 \geq 0$ *and* w_3 unrestricted. So, this is re-written as follows:

Maximize $z^* = 7w_1 + 12w_2 + 10w_3$ subject to the constraints:

$3w_1 + 2w_2 - 4w_3 \leq 1$; $w_1 + 4w_2 - 3w_3 \geq 3$; $-2w_1 - 8w_2 = 2$;

$w_1 \leq 0$ *and* $w_2 \geq 0$, w_3 unrestricted.

Some important statement of theorem related to duality:

1. The dual of the dual is the primal.

2. **(Weak- Duality Theorem)** Let x_0 be a feasible solution to the primal problem,

Maximize $f(x) = cx$ subject to: $Ax \leq b, x \leq 0$

Where x^T and $c \in R^n, b^T \in R^m$ and A is $m \times n$ real matrix. If w_0 be a feasible solution to the dual of the primal, namely

Minimize $g(w) = b^T w$, subject to: $A^T w \geq c^T, w \geq 0$

Where $w^T \in R^m$, then $cx_0 \in b^T w_0$

3. **(Basic duality theorem)** Let a primal problem be

Maximize $f(x) = cx$ subject to: $Ax \leq b, x \geq 0, x^T, c \in R^n$

And the associated dual be

Minimize $g(w) = b^T w$ subject to: $A^T w \geq c^T, w \geq 0, w^T, b^T \in R^m$

5.6 *USING SIMPLEX METHOD WITH DUALITY*

- **Formulate the Dual Problem**: Given a primal problem, formulate its dual.
- **Solve the Dual Problem Using Simplex Method**: Sometimes, it is easier to solve the dual problem than the primal problem. The solution to the dual provides information about the primal solution.
- **Interpreting the Dual Solution**: The values of the dual variables provide the shadow prices or the marginal values of the resources in the primal problem.
- **Complementary Slackness**: This principle helps in validating the solutions. For each pair of primal and dual variables, at least one in the pair must be zero in the optimal solution.

OR

If the primal problem is a maximization problem, the following set of rules govern the derivation of the optimal solution:

Rule 1: The corresponding net evaluations of the initial primal variables are equal to the difference between the left and right sides of the dual constraints associated with these initial primal variables.

Rule 2: The negative of the corresponding net evaluations of the initial dual variables is equal to the difference between the left and right sides of the primal constraints associated with these initial dual variables.

Rule 3: If the primal (dual) problem is unbounded, then the dual (primal) problem has no feasible solution.

Note: In rule 2, solve the dual problem by changing its objective from minimization to maximization.

Solved Examples

Example 6: Using duality solve the following L.P.P

Maximize $z = 2x_1 + x_2$, subject to the constraints

$$x_1 + 2x_2 \le 10;\ \ x_1 + x_2 \le 6;\ \ x_1 - x_2 \le 2;\ \ x_1 - 2x_2 \le 1;\ \ x_1, x_2 \ge 0$$

Solution: The dual problem for the given problem is as follows:

Minimize $z^* = 10w_1 + 6w_2 + 2w_3 + w_4$, subject to the constraints

$$w_1 + w_2 + w_3 + w_4 \ge 2;\ \ 2w_1 + w_2 - w_3 - 2w_4 \ge 1;\ \ w_1, w_2, w_3, w_4 \ge 0$$

Introducing surplus variables $s_1 \geq 0, s_2 \geq 0$ and artificial variables $A_1 \geq 0, A_2 \geq 0$, an initial basic feasible solution is $A_1 = 2, A_2 = 1$. (The primal constraints associated with s_1, s_2, A_1, A_2 are: $-x_1 \leq 0, -x_2 \leq 0, x_1 \leq M$ *and* $x_2 \leq M$).

The iterative simplex table are:

Initial Iteration: Introduce y_1 and y_8

c_B	y_B	w_B	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8
-*M*	y_7	2	1	1	1	1	-1	0	1	0
-*M*	y_8	1	**2**	1	-1	-2	0	-1	0	1
	*z**	-3M	-3M+10	-2M+6	2	M+1	M	M	0	0

First Iteration: Introduce y_3 and drop y_7

c_B	y_B	w_B	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8
-*M*	y_7	3/2	0	1/2	**3/2**	2	-1	1/2	1	-1/2
-10	y_1	1/2	1	1/2	-1/2	-1	0	-1/2	0	1/2
	*z**	-5-3M/2	0	1-M/2	7-3M/2	11-2M	M	5-M/2	0	-5+3M/2

Second Iteration: Introduce y_2 and drop y_1

c_B	y_B	w_B	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8
-2	y_3	1	0	1/3	1	4/3	-2/3	1/3	2/3	-1/3
-10	y_1	1	1	**2/3**	0	-1/3	-1/3	-1/3	1/3	13
	*z**	-12	0	-4/3	0	5/3	14/3	8/3	M-14/3	M-8/3

Final Iteration: Optimal Solution.

c_B	y_B	w_B	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8
-2	y_3	1/2	-1/2	0	1	3/2	-1/2	1/2	1/2	-1/2
-6	y_2	3/2	3/2	1	0	-1/2	-1/2	-1/2	1/2	1/2
	z^*	-10	2	0	0	1	4	2	M-4	M-2

Thus, an optimum feasible solution to the dual problem is.

$w_1 = 0, w_2 = 3/2$ *and* $w_3 = 1/2;\ \min(Z^*) = -(-10) = 10$.

Also the primal constraints associated with the dual variables A_1, A_2 are $x_1 \le M$ and $x_2 \le M$. Thus, by applying duality rules, the optimal solution to the primal problem is derived as follows:

Starting dual variables	A_1	A_2
Corresponding $\{-(z_j - c_j)\}$	$-(M-4)$	-(M-2)
The difference between the left and right sides of the primal constraints associated with the initial dual variables	$x_1 - M$	$x_2 - M$

Making use of Rule 2, we get

$x_1 - M = -M + 4$ and $x_2 - M = -M + 2$

$x_1 = 4$ *and* $x_2 = 2$

Hence , Maximum $z = Minimum\ \ z^* = 10$

Example 7: Consider the linear programming

Maximize $z = 3x_1 + 2x_2 + 5x_3$, subject to the constraints

$x_1 + 2x_2 + x_3 \le a_1;\ 3x_1 + 2x_2 \le a_2;\ x_1 + 4x_2 \le a_3;$

Where a_1, a_2, a_3 are constant. For specific values of a_1, a_2, a_3 the optimal solution is

Basic	x_1	x_2	x_3	x_4	x_5	x_6	Solution
Z	4	0	0	c_1	c_2	0	1350
x_2	b_1	1	0	1/2	-1/4	0	100
x_3	b_2	0	1	0	1/2	0	c_3
x_6	b_3	0	0	-2	1	1	20

Where b_i's and c_i's are constant. Determine:

(i) The values of a_1, a_2 and a_3 that yield the given optimal solution.
(ii) The values of b_1, b_2 , b_3 and c_1, c_2 , c_3 in the optimal tableau.
(iii) The optimal dual solution.

Solution: The optimal table indicates that slack variables x_4, x_5, x_6 are introduced in the three primal constraints. They happen to be the starting primal basic variables also. Thus the optimal basis inverse is given by $B^{-1} = [y_4 \;\; y_5 \;\; y_6]$ from the optimal table.

(i) We have $B^{-1}b = x_B$

$$\begin{bmatrix} 1/2 & -1/4 & 0 \\ 0 & 1/2 & 0 \\ -2 & 1 & 1 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} 100 \\ c_3 \\ 20 \end{bmatrix}$$

$$\Rightarrow \frac{1}{2}a_1 - \frac{1}{4}a_2 = 100, \frac{1}{2}a_2 = c_3, -2a_1 + a_2 + a_3 = 20$$

Also, $z = c_B x_B \Rightarrow 1350 = 200 + 5c_3 \Rightarrow c_3 = 230$, where $c_B = [2 \; 5 \; 0]$.

Thus, we get $a_1 = 430, a_2 = 460$ and $a_3 = 480$

(ii) The *z*-row gives:

$$4 = c_B y_1 - c_1 = 2b_1 + 5b_2 - 3 \Rightarrow 2b_1 + 5b_2 = 7$$

$$c_1 = c_B y_4 - c_4 = 1 - 0 = 1$$

$$c_2 = c_B y_5 - c_5 = -1/2 + 5/2 - 0 = 2$$

To obtain the value of b_1, b_2 and b_3, we perform iteration on the starting primal table:

Initial Iteration: Introduce y_3 and drop y_5

			3	2	5	0	0	0
c_B	y_B	x_B	y_1	y_2	y_3	y_4	y_5	y_6
0	y_4	$a_1 = 430$	1	2	1	1	0	0
0	y_5	$a_2 = 460$	3	0	**2**	0	1	0
0	y_6	$a_3 = 480$	1	4	0	0	0	1
		0	-3	-2	-5	0	0	0

First Iteration: Introduce y_2 and drop y_4

c_B	y_B	x_B	y_1	y_2	y_3	y_4	y_5	y_6
0	y_4	200	-1/2	2	0	1	-1/2	0
0	y_5	230	3/2	0	1	0	1/2	0
0	y_6	480	1	4	0	0	0	1
		1150	9/2	-2	0	0	5/2	0

Second Iteration: Optimum Solution.

c_B	y_B	x_B	y_1	y_2	y_3	y_4	y_5	y_6
2	y_2	100	-1/4	1	0	1/2	-1/4	0
5	y_3	230	3/2	0	1	0	1/2	0
0	y_6	80	-4	0	0	-2	1/2	1
		1350	4	0	0	1	2	0

Comparing it with the given optimal table, we get

$b_1 = -1/4, b_2 = 3/2$ and $b_3 = -4$

(Note that the values of c_1, c_2 are also readily available.)

(iii) The dual problem is,

Minimize $z^* = a_1 w_1 + a_2 w_1 + a_3 w_3$, subject to the constraints:

$w_1 + 3w_2 + w_3 \geq 3,\ 2w_1 + w_2 + 4w_3 \geq 2,\ w_1 + 2w_2 + 0w_3 \geq 5$

$w_1 \geq 0,\ w_2 \geq 0\ and\ w_3 \geq 0$

The dual constraints associated with the starting primal variables x_4, x_5 and x_6 and $w_1 \geq 0, w_2 \geq 0$ and $w_3 \geq 0$

Thus we have the following information:

Starting primal variables	x_4	x_5	x_6
Left minus right sides of the associated dual constraint Net evaluation primal optimal table	$w_1 - 0$ c_1	$w_2 - 0$ c_2	$w_3 - 0$ 0

This using Rule 1 we get

$c_1 = w_1 - 0 \Rightarrow w_1^* = c_1 = 1$

$c_2 = w_2 - 0 \Rightarrow w_2^* = c_2 = 2$

$0 = w_3 - 0 \Rightarrow w_3^* = 0$

The optimal dual objective is Min. $z^* = 1350 = a_1 w_1^* + a_2 w_2^*$

Check your progress

Problem 1: Obtain the dual of the problem

Maximize $Z = 2x_1 + 3x_2 + x_3$ subject to the constraints:

$4x_1 + 3x_2 + x_3 = 6,\ x_1 + 2x_2 + 5x_3 = 4;\ x_1, x_2, x_3 \geq 0$

Answer: Maximize $Z^* = 7w_1 + 12w_2 + 10w_3$ subject to the constraints:

$3w_1 + 2w_2 - 4w_3 \leq 1, = 6,\ w_1 + 4w_2 - 3w_3 \geq 3, - 2w_1 - 8w_3 = 2$

$w_1 \leq 0$ *and* $w_2 \geq 0,\ w_3$ unrestricted.

Problem 2: Find the dual of the following problem.

Maximize, $Z = 4x_1 + 2x_2$

Subject to,

$x_1 + x_2 \geq 3;\ x_1 - x_2 \geq 2;\ x_1, x_2 \geq 0$

Answer: Minimize $z^* = 3w_1 + 2w_2$

Subject to the constraints:

$w_1 + w_2 \geq 4;\ w_1 - w_2 \geq 2;\ w_1 \leq 0$ and $w_2 \leq 0$

Problem 3: Solve the following LPP using dual

Maximize, $Z = 8x_1 + 4x_2$

Subject to,

$4x_1 + 2x_2 \leq 30;\ 2x_1 + 4x_2 \leq 24;\ x_1, x_2 \geq 0$

Answer: Minimize $z^* = 30w_1 + 24w_2$

Subject to the constraints:

$4w_1 + 2w_2 \geq 8;\ 2w_1 + 4w_2 \geq 4;\ w_1 \geq 0, w_2 \geq 0$ and $w_2 \leq 0$

And optimal solution is $x_1 = 6,\ x_2 = 3$ maximum $z = 60$

5.7 SUMMARY

The summary of this unit are as follows:

- Every linear programming problem (primal) has a corresponding dual problem.
- The primal problem involves maximizing or minimizing an objective function subject to constraints and non-negativity restrictions.
- The dual problem is derived from the primal problem and involves a different set of variables and constraints, effectively reversing the roles of the constraints and the objective function.
- For any feasible solutions to the primal and dual problems, the value of the objective function in the primal problem is less than or equal to the value in the dual problem (for maximization) or greater than or equal to the value in the dual problem (for minimization).
- If the primal problem has an optimal solution, the dual problem also has an optimal solution, and the optimal values of their objective functions are equal.
- Sometimes, solving the dual problem is easier than solving the primal problem. This can be especially true when the dual problem has fewer constraints or variables.
- Duality theory is also used in sensitivity analysis to understand how changes in the coefficients of the primal problem affect the optimal solution.

Understanding duality is essential for grasping the deeper structure of linear programming problems, providing insights that can be leveraged in both theoretical analyses and practical applications.

5.8 GLOSSARY

- Duality

5.9 REFERENCES

- Mokhtar S. Bazaraa, John J. Jarvis and Hanif D. Sherali: *Linear Programming and Network Flows* (4th edition). John Wiley and Sons, 2010.
- Hamdy A. Taha: *Operations Research: An Introduction (10th edition). Pearson*, 2017.
- Paul R. Thie and Gerard E. Keough: *An Introduction to Linear Programming and Game Theory* (3rd edition), Wiley India Pvt. Ltd, 2014.
- Swarup, K., Gupta, P. K., & Mohan, M. (2017). An introduction to management science operation research. *Sultan Chand & Sons educational publishers, New Delhi*.

- OpenAI. (2024). *ChatGPT (August 2024 version) [Large language model]*. OpenAI. https://www.openai.com/chatgpt

5.10 *SUGGESTED READING*

- G. Hadley, *Linear Programming*, Narosa Publishing House, 2002.
- Frederick S. Hillierand Gerald J. Lieberman: *Introduction to Operations Research* (10th edition). McGraw-Hill Education, 2015.
- https://epgp.inflibnet.ac.in/Home/ViewSubject?catid=ZLCHeZEhCZ8yCri36nSF3A==

5.11 *TERMINAL QUESTION*

Short Answer Type Question:

1. Find the dual of the following problem

Maximize, $z = 2000x_1 + 3000x_2$

Subject to the constraints,

$6x_1 + 9x_2 \leq 100;\ 2x_1 + x_2 \leq 20;\ x_1, x_2 \geq 0$

Answer: $x_1 = 23/3, x_2 = 0, x_3 = 5$; Maximum $z = 85/3$

2. Find the dual of the following problem

Maximize, $z = 10x + 8y$

Subject to,

$x + 2y \geq 5;\ 2x - y \geq 12;\ x + 3y \geq 4$

$x \geq 0$ and y is unrestricted.

Answer: $x_1 = 0; x_2 = 1.2; x_3 = 0.9; x_4 = 0;$; Maximum $z = 3.6$

3. Find the dual of the following problem

Maximize, $z = 2x + 5y + 6z$

Subject to,

$5x+6y-z\le 3;\ -2x+y+4z\le 4;\ x-5y+3z\le 1;\ -3x-3y+7z\le 6;$

$x, y, z \ge 0$

Answer: $x_1 = 18/5, x_2 = 6/5, x_3 = 8/5$; Maximum $z = 108$

4. Find the dual of the following problem

Minimize, $z = x_1 + 2y$

Subject to,

$2x+4y\le 160;\ x-y=30;\ x\ge 10;\ x\ge 0$ and $y\ge 0$

Answer: $x_1 = 6, x_2 = 7, x_3 = 0$; Maximum $z = 177$

5. Find the dual of the following problem

Maximize, $z = 2x+3y+z$

Subject to,

$4x+3y+z=6;\ x+2y+5z=4;$

$x, y, z \ge 0$

6. Find the dual of the following problem

Maximize, $z = 3x+5y+7z$

Subject to,

$x+y+3z\le 10;\ 4x-y+2z\ge 15$

$x, y \ge 0$ and z is unrestricted.

7. Find the dual of the following problem

Minimize, $z = x+y+z$

Subject to,

$x-3y+4z=5;\ x-2y\le 3; 2y-z\ge 4$

$x, y\ge 0$ and z is unrestricted.

8. Find the dual of the following problem

Minimize, $z=2x+3y+4z$

Subject to,

$2x+3y+5z\ge 2;\ 3x+y+7z=3; x+4y+6z\le 5$

$x, y\ge 0$ and z is unrestricted.

9. Find the dual of the following problem

Maximize, $z=6x+6y+z+7w+5s$

Subject to,

$3x+7y+8z+5w+s=2;\ 2x+y+3z+2w+9s=6$

$x, y, z, w\ge 0$ and t is unrestricted.

Long answer type question:

1. Solve the following LPP by using dual of the following problem

Maximize, $z=8x+4y$

Subject to,

$4x+2y\le 30;\ 2x+4y\le 24$

$x, y\ge 0$

2. Solve the following LPP by using dual of the following problem

Minimize, $z=15x+10y$

Subject to,

$3x + 5y \geq 5;\ 5x + 2y \geq 3$

$x, y \geq 0$

3. Solve the following LPP by using dual of the following problem

Maximize, $z = 5x + 2y$

Subject to,

$6x + y \geq 6;\ 4x + 3y \geq 12;\ x + 2y \geq 4$ and $x, y \geq 0$

4. Solve the following LPP by using dual of the following problem

Maximize, $z = 2x + 9y + z$

Subject to,

$x + 4y + 2z \geq 5;\ 3x + y + 2z \geq 4;$ and $x, y, z \geq 0$

5. Solve the following LPP by using dual of the following problem

Maximize, $z = x + 5y + 3z$

Subject to,

$x + 2y + z = 3;\ 2x - y = 4;$ and $x, y, z \geq 0$

6. Solve the following LPP by using dual of the following problem

Minimize, $z = 10x + 4y + 5z + w$

Subject to,

$5x - 7y + 3z + 0.5w \geq 150;$ and $x, y, z, w \geq 0$

7. Solve the following LPP by using dual of the following problem

Maximize, $z = x - y + 3z + 2w$

Subject to,

$x + y \geq -1;\ x - 3y - z \leq 7;\ x + z - 3w = -2$ and $x, y, z, w \geq 0$

8. Solve the following LPP by using dual of the following problem

Maximize, $z = 2y - 5z$

Subject to,

$x + z \geq 2;\ 2x + y + 6z \leq 6; x - y + 3z = 0$ and $x, y, z, w \geq 0$

Fill in the blanks:

1: The dual of the dual is …………………….

2: If either the primal or the dual problem has an unbounded objective function value, then the other problem has ………………

5.12 *ANSWERS*

Answer of short answer type question

1: Min $(z^*) = 100w_1 + 20w_2$

Subject to, $6w_1 + 2w_2 \geq 2000;\ 9w_1 + w_2 \geq 3000;\ w_1 \geq 0$ *and* $w_2 \geq 0$

2: Min $(z^*) = 5w_1 + 12w_2 + 4w_3$

Subject to, $w_1 + 2w_2 + w_3 \geq 10;\ 2w_1 - w_2 + 3w_3 = 8;\ w_1 \leq 0,\ w_2 \leq 0$ *and* $w_3 \leq 0$

3: Min $(z^*) = 3x_1 + 4x_2 + x_3$

Subject to, $5x_1 - 2x_2 + x_3 \geq 2;\ 6x_1 + x_2 - 5x_3 \geq 5;\ -x_1 + 4x_2 + 3x_3 \geq 6; x_1, x_2, x_3, x_4 \geq 0$

4: Max $(z^*) = 160x_1 + 30x_2 + 10x_3$

Subject to, $2x_1 + x_2 + x_3 \leq 1;\ 4x_1 - x_2 \leq 2;; x_1 \leq 0, x_3 \geq 0$ and x_2 is unrestricted

5: Min $(z^*) = 6x_1 + 4x_2$

Subject to, $4x_1 + 2x_2 \geq 2;\ 3x_1 + 2x_2 \leq 3;\ x_1 + 5x_2 \geq 1;\ x_1$ and x_2 is unrestricted

6: Min $(z^*) = 10x_1 + 15x_2$

Subject to, $x_1 + 4x_2 \geq 3;\ x_1 - x_2 \geq 5;\ 3x_1 + 2x_2 = 7;\ \ x_1 \geq 0\ \textit{and}\ x_2 \leq 0$

7: Min $(z^*) = 5x_1 + 4x_2$

Subject to, $x_1 + x_2 \leq 1;\ -3x_1 - 2x_2 + 2x_3 \leq 1;\ 4x_1 - x_3 = 1;\ \ x_2 \leq 0$, $x_3 \geq 0$ and x_1 is unrestricted

8: Max $(z^*) = 2x_1 + 3x_2 + 5x_3$

Subject to, $2x_1 + 3x_2 + x_3 \leq 2;\ 3x_1 + x_2 + 4x_3 \leq 3;$

$5x_1 + 7x_2 + 6x_3 = 4; x_1 \geq 0, x_3 \leq 0\ \textit{and}\ x_2$ is unrestricted

9: Min $(z^*) = 2x_1 + 6x_2$

Subject to, $3x_1 + 2x_2 \geq 6;\ 7x_1 + x_2 \geq 6; 8x_1 + 3x_2 \geq 1; 5x_1 + 2x_2 \geq 7;$

$x_1 + 9x_2 = 5$; x_1 and x_2 is unrestricted

Answer of long answer type question

1: Min $(z^*) = 30x_1 + 24x_2$

Subject to, $4x_1 + 2x_2 \geq 8;\ 2x_1 + 4x_2 \geq 4; x_1 \geq 0; x_2 \geq 0$

The optimal solution is $w_1 = 6$ and $w_2 = 3$, max $z = 60$

2: The optimal solution is $w_1 = 5/19$ and $w_2 = 16/19$, min $z = 235/19$

3: Unbounded solution

4: $w_1 = 0$, $w_2 = 0$ and $w_3 = 5/2$; min $z = 5/2$

5: Minimize $(z^*) = 3x_1 + 4x_2$

Subject to, $x_1 + 2x_2 \geq 1;\ 2x_1 - x_2 \geq 5; x_1 \geq 3$ and x_2 is unrestricted.

The optimal solution is $x_1 = 3$ and $x_2 = -1$, min $z^* = 5$.

6: The optimal solution is $x_1 = 0$; $x_2 = 0$; $x_3 = 50$; $x_4 = 0$; min $z^* = 250$.

7: Unbounded solution.

8: Minimize $z^* = 2x_1 + 6x_2$

Subject to the constraint, $x_1 + 2x_2 + x_3 \geq 0;\ x_2 - x_3 \geq 2;\ x_1 + 6x_2 + 3x_3 \geq -5$ and $x_1 \leq 0,\ x_2 \geq 0$ and x_3 is unrestricted in sign.

The optimal solution is $x_1 = 0$; $x_2 = 2/3$; $x_3 = -4/3$; min $z^* = 4$.

Answer of fill in the blank question

1: Primal

2: No Feasible solution

BLOCK- II

REAL LIFE PROBLEM

UNIT-6: SENSITIVITY ANALYSIS

CONTENTS:

6.1 *INTRODUCTION*

Post-optimal analysis, also known as sensitivity analysis, is a crucial phase in decision-making processes, particularly in the context of optimization and operations research. This type of analysis is performed after an optimal solution has been found, aiming to understand the robustness of the solution and how changes in the input parameters can affect the outcome. Sensitivity analysis is an essential instrument for decision-makers to comprehend how modifications to a mathematical model's parameters affect the best possible outcome. It can assist in determining which factors have the biggest effects on the outcome and what adjustments are required to get at a different ideal outcome.

6.2 *OBJECTIVE*

After reading this unit learners will be able to

- Understand the basic concept of sensitivity analysis or Post-optimal analysis.
- Learn about the Structural changes from adding or removing variables.
- Visualized about the Structural changes from adding or removing linear constraints.

6.3 *SENSITIVITY ANALYSIS* (*POST-OPTIMAL ANALYSIS*)

Post-optimal analysis evaluates the effects of varying input parameters on the optimal solution of an optimization problem. This analysis helps decision-makers to understand the sensitivity of the optimal solution to changes in parameters and to assess the potential risks and uncertainties associated with these changes.

Objectives of Post-Optimal Analysis

1. Robustness Check: Determine how stable the optimal solution is with respect to changes in input parameters.
2. What-If Scenarios: Explore different scenarios by varying parameters and observe their impacts on the optimal solution.
3. Parameter Ranges: Identify the range of parameter values for which the current optimal solution remains valid.
4. Decision Making: Provide additional insights that help in making informed decisions under uncertainty.

Types of Post-Optimal Analysis

1. Sensitivity Analysis: Examines how small changes in the parameters affect the optimal solution. This includes:

- **Objective Function Coefficients**: How changes in the coefficients of the objective function impact the optimal solution.
- **Right-Hand Side (RHS) Values**: How changes in the constraints' RHS values influence the solution.

- **Constraint Coefficients**: How variations in the coefficients of the constraints affect the optimal solution.

2. **Shadow Prices**: These are the rates at which the optimal value of the objective function improves as the right-hand side of a constraint is increased by one unit, indicating the marginal worth of resources.

3. **Scenario Analysis**: Evaluates the outcomes of different potential scenarios by altering multiple parameters simultaneously to see how the solution space and optimal solution change.

Importance of Post-Optimal Analysis

- **Decision Support**: Provides deeper insights into the problem, helping decision-makers understand the implications of their choices.
- **Risk Management**: Identifies parameters that are critical to the solution, allowing for better management of risks and uncertainties.
- **Resource Allocation**: Helps in efficient allocation of resources by understanding the value and impact of changes in resource availability.

Steps in Conducting Post-Optimal Analysis

1. **Identify Key Parameters**: Determine which parameters are most likely to change or have the greatest impact on the solution.

2. **Perform Sensitivity Analysis**: Analyze how changes in these parameters affect the optimal solution.

3. **Interpret Results**: Assess the implications of the analysis for decision-making.

4. **Make Recommendations**: Provide actionable insights based on the analysis to support robust and informed decision-making.

Note: The variations in LPP that are typically examined using sensitivity analysis consist of:

1. Variations in the profit per unit connected to the choice variables, or the objective function coefficients, or $c_j^{'}s$.

2. Variations in the bi value, or the right-hand side constants of the constraints, which represent the resources that are accessible.

3. Variations in the decision variable's coefficients on the left side of the restrictions a_{ij}, or the amount of resources used per unit of the decision variables x_j.

4. Modifications in structure brought about by the inclusion and removal of certain variables.

5. Modifications in structure brought about by the inclusion and removal of certain linear constraints.

Remarks: Following the aforementioned modifications in an L.P.P., we could encounter any of the following scenarios:
(i) The optimum solution stays the same, meaning that the fundamental variables and their values essentially stay the same.
(ii) The basic variables don't change, but their values do.
(iii) The basic solution is completely altered.

6.4 *CHANGES IN OBJECTIVE FUNCTION COEFFICIENTS, c_j's*

Changes in the objective function coefficients $c_j^{'}s$ in an optimization problem can significantly impact the optimal solution. Sensitivity analysis related to these coefficients helps in understanding how robust the optimal s1olution is when the objective function changes. Here's a detailed look at this aspect of post-optimal analysis:

Objective Function Coefficients $c_j^{'}s$

In a linear programming problem, the objective function is typically expressed as:

Maximize or Minimize $Z = c_1x_1 + c_2x_2 + ... + c_nx_n$

where $c_j^{'}s$ are the coefficients of the decision variables x_j. These coefficients represent the contribution of each decision variable to the objective function.

Sensitivity Analysis of $c_j^{'}s$

Sensitivity analysis of the objective function coefficients involves examining how changes in c_j affect the optimal solution. This analysis provides insights into how the optimal value of the objective function and the optimal values of decision variables change when c_j is varied.

Key Concepts

1. **Range of Optimality**: This is the range of values that an objective function coefficient c_j can take without changing the optimal solution. Within this range, the current set of basic variables remains optimal.
2. **Reduced Costs**: For a non-basic variable in the optimal solution, the reduced cost indicates how much the objective function coefficient would need to improve before that variable would enter the basis and become part of the optimal solution.

Steps in Sensitivity Analysis of c_j

1. **Identify the Current Optimal Solution**: Determine the optimal values of the decision variables and the value of the objective function for the current coefficients c_j.
2. **Calculate Reduced Costs**: For each non-basic variable, compute the reduced cost to understand how changes in c_j will affect the variable's status (whether it enters the basis or not).
3. **Determine the Range of Optimality**: For each c_j, identify the range within which the current solution remains optimal. This involves checking when the relative cost of entering the basis changes.

Example 1: Consider a simple linear programming problem:

Maximize $Z = 3x_1 + 5x_2$

Subject to: $x_1 + 2x_2 \le 4$; $2x_1 + x_2 \le 5$; $x_1 \ge 0, x_2 \ge 0$

Let's assume the optimal solution is $x_1 = 1, x_2 = 1.5$ and $Z = 10.5$

To perform sensitivity analysis on c_1 and c_2

1. **Identify the Range of Optimality for c_1:**
- Determine the reduced costs and see how much c_1 can change while keeping x_1 and x_2 as the basic variables.
2. **Identify the Range of Optimality for c_2:**
- Similarly, find out the range for c_2 where the current solution remains optimal.

Interpretation of Results

- **If c_1 changes within its range of optimality**, the optimal solution (values of x_1 and x_2) remains the same, but the value of the objective function Z will change accordingly.
- **If c_2 changes within its range of optimality**, the same holds true for x_2 and the objective function Z.
- **If c_j changes outside its range of optimality**, the optimal solution may change, meaning the values of the decision variables and potentially the basis of the solution will be different.

Analyzing changes in the objective function coefficients c_j helps in understanding the sensitivity and stability of the optimal solution. It aids decision-makers in evaluating the impact of variations in the coefficients, thus enabling more robust and informed decision-making in the presence of uncertainties or changes in the problem parameters.

Example 2: Examine the impact on the solution's optimality when the objective function changes to, $Z = 3x_1 + x_2$

The LPP problem is,

Maximize $Z = 3x_1 + 5x_2$, subject to the constraints:

$x_1 \le 4, x_2 \le 6;\ 3x_1 + 2x_2 \le 18;\ x_1, x_2 \ge 0$

Solution: After introducing the slack variables $s_1 \ge 0, s_2 \ge 0$ and $s_3 \ge 0$ in the given LPP by simplex method, the optimum simplex table is:

c_B	y_B	x_B	y_1	y_2	y_3	y_4	y_5
0	y_3	2	0	0	1	2/3	-1/3
5	y_2	6	0	1	0	1	0
3	y_1	2	1	0	0	-2/3	1/3
	z	36	0	0	0	3	1

Since, the objective function is changed to $3x_1 + x_2; c_2$ has been changed to 1 keeping c_1 fixed. So, we find the variation in c_2.

Variation in c_2: Since, $c_2 \in c_B$, the range of Δc_2 is given by.

$$\underset{y_{2j}>0}{Max}\left\{\frac{-(z_j - c_j)}{y_{2j}}\right\} \le \Delta c_2 \le \underset{y_{2j}>0}{Min}\left\{\frac{-(z_j - c_j)}{y_{2j}}\right\} \text{ i.e., } Max\left\{\frac{-3}{1}\right\} \le \Delta c_2 < \infty \text{ or } -3 \le \Delta c_2 < \infty$$

$$\therefore 5 - 3 \le c_2 < 5 + \infty \quad \text{or} \quad 2 \le c_2 < \infty$$

This indicate that if c_2 is changed 1, The above-mentioned optimal solution is no longer optimal.

To find the new optimum solution:

When the new objective function $3x_1 + 5x_2$ is changed to $z^* = 3x_1 + 5x_2$:

$z_1 - c_1 = z_2 - c_2 = z_3 - c_3 = 0; z_4 - c_4 = c_B y_4 - c_4 = -1$; and $z_5 - c_5 = c_B y_5 - c_5 = 1$

This shows that y_4 enters the basis and y_3 leaves the basis in the next iteration. As a result, we have the ideal simplex table shown below:

c_B	y_B	x_B	y_1	y_2	y_3	y_4	y_5
0	y_4	3	0	0	3/2	1	-1/2
1	y_2	3	0	1	-3/2	0	1/2
3	y_1	4	1	0	1	0	0
	Z^*	15	0	0	3/2	0	1/2

$\therefore$ The optimum solution to the revised LPP is,

$x_1 = 4,\ x_2 = 3,$ and Maximum $z^* = 15$.

6.5 *CHANGE IN THE b_i VALUES*

Changes in the right-hand side (RHS) values b_i in an optimization problem can also significantly impact the optimal solution. Sensitivity analysis related to these values helps in understanding how variations in the constraints affect the optimal solution. Here's a detailed examination of this aspect of post-optimal analysis:

Right-Hand Side (RHS) Values b_i

In a linear programming problem, constraints are typically expressed as:

$$a_{11}x_1 + a_{12}x_2 + ... + a_{1n}x_n \leq b_1$$

$$a_{21}x_1 + a_{22}x_2 + ... + a_{2n}x_n \leq b_2$$

$\vdots$

$a_{m1}x_1 + a_{m2}x_2 + ... + a_{mn}x_n \le b_m$

where b_i are the RHS values of the constraints. These values represent the availability of resources or limits within the problem.

Sensitivity Analysis of b_i

Sensitivity analysis of the RHS values involves examining how changes in b_i affect the optimal solution. This analysis provides insights into how the optimal value of the objective function and the optimal values of decision variables change when b_i is varied.

Key Concepts

1. **Shadow Prices**: The shadow price (or dual value) of a constraint represents the rate at which the objective function value improves as the RHS value of the constraint is increased by one unit. It indicates the marginal worth of an additional unit of the resource.
2. **Range of Feasibility**: This is the range of values that an RHS value b_i can take without changing the optimal basis. Within this range, the current set of basic variables remains optimal, though their values may change.

Steps in Sensitivity Analysis of b_i

1. **Identify the Current Optimal Solution**: Determine the optimal values of the decision variables and the value of the objective function for the current RHS values b_i.
2. **Calculate Shadow Prices**: For each constraint, compute the shadow price to understand the impact of changes in b_i on the objective function.
3. **Determine the Range of Feasibility**: For each b_i, identify the range within which the current basis remains optimal. This involves analyzing when adding or removing units of the resource will change the basis of the solution.

Example 3: Consider a simple linear programming problem:

Maximize, $Z = 3x_1 + 5x_2$

Subject to: $x_1 + 2x_2 \le 4$

$2x_1 + x_2 \le 5; \quad x_1, x_2 \ge 0$

Let's assume the optimal solution is $x_1 = 1, x_2 = 1$ and $Z = 10.5$

To perform sensitivity analysis on b_1 and b_2 :

1. **Identify the Shadow Prices**:
- Calculate the shadow price for each constraint to determine the rate at which Z changes with b_1 and b_2
2. **Determine the Range of Feasibility for** b_1:
- Identify the range within which b_1 can vary while keeping the current basis optimal.
3. **Determine the Range of Feasibility for** b_2 :
- Similarly, find out the range for b_2 where the current basis remains optimal.

Interpretation of Results

- **If** b_1 **changes within its range of feasibility**, the optimal solution (values of x_1 and x_2) remains optimal, but the values of the decision variables may change, and the objective function Z will be adjusted according to the shadow price.
- **If** b_2 **changes within its range of feasibility**, the same holds true for x_2 and the objective function Z.
- **If** b_i **changes outside its range of feasibility**, the optimal basis may change, meaning the set of basic variables and potentially the solution itself will be different.

Analyzing changes in the RHS values b_i helps in understanding the sensitivity and stability of the optimal solution with respect to variations in resource availability or constraint limits. It aids decision-makers in evaluating the impact of these variations, enabling more robust and informed decision-making in the presence of uncertainties or changes in the problem parameters.

Example 4: In the given LPP

Maximize, $Z = -x_1 + 2x_2 - x_3$

Subject to: $3x_1 + x_2 - x_3 \le 10$; $-x_1 + 4x_2 + x_3 \ge 6$; $x_2 + x_3 \ge 4$; $x_j \ge 0$ for $j = 1,2,3$

Establish the ranges for discrete modifications to the requirement vector's components b_2 and b_3 in order to preserve the feasibility of the present optimal solution.

Solution: After introducing the slack variables and surplus variable $s_1 \geq 0, s_3 \geq 0$ and $s_2 \geq 0$ respectively and also an artificial variable $A_1 \geq 0$ in the given constraint of problem, the optimum simplex table is:

c_B	y_B	x_B	y_1	y_2	y_3	y_4	y_5	y_6	y_7
0	y_4	6	3	0	-2	1	0	-1	0
2	y_2	4	0	1	1	0	0	1	0
0	y_5	10	1	0	3	0	1	4	-1
	Z	8	1	0	3	0	0	2	M

From the above table we observe that

x_B = [6 4 10] and $B^{-1} = [y_4\ y_7\ \ y_6] = \begin{pmatrix} 1 & 0 & -1 \\ 0 & 0 & 1 \\ 0 & -1 & 4 \end{pmatrix}$

Column of the matrix B^{-1} correspond to the column of the starting basis matrix B.

The individual effects of changes in b_2 and b_3 where b = [b_1 b_2 b_3] such that the optimality of the basic feasible solution is not violated, are given by

$$Max\left\{\frac{-x_{Bi}}{\beta_{ik} > 0}\right\} \leq \Delta b_k \leq Min\left\{\frac{-x_{Bi}}{\beta_{ik} < 0}\right\}$$

i.e., $\Delta b_k \leq \frac{-10}{-1}$ and Max. $\left\{\frac{-4}{1}, -\frac{-10}{4}\right\} \leq \Delta b_3 \leq \frac{-6}{-1}$

i.e., $\Delta b_2 \leq 10$ and $-5/2 \leq \Delta b_3 \leq 6$

Now, since $b_2 = 6$ and $b_3 = 4$, then variation required range is

$b_2 \leq 10 + 6$ and $4 - 5/2 \leq b_3 \leq 4 + 6$

i.e., $b_2 \leq 16$ and $3/2 \leq b_3 \leq 10$

6.6 *CHANGE IN THE COEFFICIENTS a_{ij}'s VALUES*

Changes in the coefficients a_{ij} of the constraints in an optimization problem can affect the feasible region and the optimal solution. Sensitivity analysis related to these coefficients helps in understanding how variations in the constraint coefficients impact the optimal solution. Here's a detailed examination of this aspect of post-optimal analysis:

Constraint Coefficients a_{ij}

In a linear programming problem, constraints are typically expressed as:

$$a_{11}x_1 + a_{12}x_2 + ... + a_{1n}x_n \leq b_1$$

$$a_{21}x_1 + a_{22}x_2 + ... + a_{2n}x_n \leq b_2$$

$$\vdots$$

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

where a_{ij} are the coefficients of the decision variables x_j in the i^{th} constraint. These coefficients represent the contribution of each decision variable to the constraint.

Sensitivity Analysis of a_{ij}

Sensitivity analysis of the constraint coefficients involves examining how changes in a_{ij} affect the optimal solution. This analysis provides insights into how the optimal value of the objective function and the optimal values of decision variables change when a_{ij} is varied.

Key Concepts

1. **Range of Feasibility**: This is the range of values that a constraint coefficient a_{ij} can take without changing the optimal basis. Within this range, the current set of basic variables remains optimal, though their values may change.
2. **Allowable Increase and Decrease**: These are the amounts by which a coefficient a_{ij} can be increased or decreased without changing the optimal basis.

Steps in Sensitivity Analysis of a_{ij}

1. **Identify the Current Optimal Solution**: Determine the optimal values of the decision variables and the value of the objective function for the current coefficients a_{ij}.
2. **Calculate Allowable Changes**: For each a_{ij}, compute the allowable increase and decrease to understand the range within which the current basis remains optimal.
3. **Analyze the Impact on the Optimal Solution**: Evaluate how changes within the allowable range affect the values of the decision variables and the objective function.

Example 5: Consider a simple linear programming problem:

Maximize, $Z = 3x_1 + 5x_2$

Subject to: $x_1 + 2x_2 \leq 4$

$2x_1 + x_2 \leq 5; \quad x_1, x_2 \geq 0$

Let's assume the optimal solution is $x_1 = 1, x_2 = 1$ and $Z = 10.5$

To perform sensitivity analysis on a_{11} and a_{21}:

1. **Identify the Current Optimal Solution**:
   - Determine the optimal values of x_1 and x_2 and the value of Z for the given constraints.
2. **Calculate Allowable Changes for a_{11}:**
   - Find the range within which a_{11} can vary while keeping the current basis optimal.
3. **Calculate Allowable Changes for a_{21}:**
   - Similarly, determine the range for a_{21}.

Interpretation of Results

- **If a_{11} changes within its allowable range**, the optimal solution (values of x_1 and x_2) remains optimal, but the values of the decision variables and the objective function Z may change.
- **If a_{21} changes within its allowable range**, the same holds true for x_2 and the objective function Z.
- **If a_{ij} changes outside its allowable range**, the optimal basis may change, meaning the set of basic variables and potentially the solution itself will be different.

Analyzing changes in the constraint coefficients a_{ij} helps in understanding the sensitivity and stability of the optimal solution with respect to variations in the contributions of decision variables to the constraints. It aids decision-makers in evaluating the impact of these variations, enabling more robust and informed decision-making in the presence of uncertainties or changes in the problem parameters.

Example 6: In the given LPP

Maximize, $Z = 3x_1 + 4x_2 + x_3 + 7x_4$

Subject to: $8x_1 + 3x_2 + 4x_3 + x_4 \le 7$; $2x_1 + 6x_2 + x_3 + 5x_4 \le 3$; $x_1 + 4x_2 + 5x_3 + 2x_4 \le 8$

$x_1, x_2, x_3, x_4 \ge 0$

The given table provides the optimal solution

c_B	y_B	x_B	y_1	y_2	y_3	y_4	y_5	y_6	y_7
3	y_1	16/19	1	9/38	1/2	0	5/38	-1/38	0
7	y_4	5/19	0	21/19	0	1	-1/19	8/38	0
0	y_7	166/19	0	59/38	9/2	0	-1/38	-15/38	1
	Z	83/19	0	169/38	1/2	0	1/38	53/38	0

Discuss how the current optimum basic feasible solution to the provided LPP is affected by discrete changes in the activity co-efficient a_{ij} of A.

Solution: Using above table we construct the B^{-1} as

$$B^{-1} = \begin{pmatrix} \overset{\beta_1}{5/38} & \overset{\beta_2}{-1/38} & \overset{\beta_3}{0} \\ -1/19 & 8/38 & 0 \\ -1/38 & -15/38 & 1 \end{pmatrix}$$, Here B represents the matrix form of the basis of given problem.

Thus, $c_B\beta_1 = 3(5/38) + 7(-1/19) + 0(-1/38) = 1/38$

$c_B\beta_2 = 3(-1/38) + 7(8/19) + 0(-15/38) = 53/38$

$c_B\beta_3 = 3(0) + 7(0) + 0(1) = 0$

Let the element a_{rk} of A be changed to $a_{rk}* = a_{rk} + \Delta a_{rk}$

Case I: Let the element a_{rk} is in the basis.

Since y_2, y_3, y_5 and y_6 are not in the basis in the optimal solution table, the following inequalities provide the ranges for discrete changes in the coefficients a_{ij} corresponding to these non-basis vectors.

$$-\frac{169/38}{1/38} \le \Delta a_{12} \Rightarrow \Delta a_{12} \ge -169$$

$$-\frac{169/38}{53/38} \le \Delta a_2 \Rightarrow \Delta a_{22} \ge -169/53$$

$$-\frac{169/38}{0} \le \Delta a_{32} < \infty \Rightarrow -\infty < \Delta a_{32} < \infty$$

$$\frac{-1/2}{1/38} \le \Delta a_{13} \Rightarrow \Delta a_{13} \ge -19$$

$$\frac{-1/2}{53/38} \le \Delta a_{23} \Rightarrow \Delta a_{23} \ge -19/53$$

$$\frac{-1/2}{0} \le \Delta a_{33} < \infty \Rightarrow -\infty \le \Delta a_{33} < \infty$$

$$a_{12} \ge 6 - 169 \;\; or \;\; a_{12} \ge -163;$$

$$a_{22} \ge \frac{-169}{53} + 6 \;\; or \;\; a_{22} \ge \frac{149}{53}; -\infty \le a_{32} < \infty$$

$$a_{13} \ge 4 - 19 \;\; or \;\; a_{13} \ge 15;$$

$$a_{23} \ge 1 - \frac{19}{53} \;\; or \;\; a_{23} \ge \frac{34}{53} \text{ and } -\infty \le a_{33} < \infty$$

Case II: Let the element a_{rk} is in the basis.

Given that y_1, y_4, and y_7 form the basis of the optimal solution table, any discontinuous alteration in ark associated with any of these vectors may have an impact on both the feasibility and optimality of the first optimum basic solution, x_B.

Examining the distinct modifications in a_{rk} that correspond to $y_4 = \beta_2$, we have

$$\beta_{22}(z_2 - c_2) - y_{22}c_B\beta_2 = \frac{8}{38} \times \frac{169}{38} - \frac{21}{19} \times \frac{53}{38} = -\frac{23}{38}$$

$$\beta_{22}(z_3 - c_3) - y_{23}c_B\beta_2 = \frac{8}{38}\times\frac{1}{2} - 0\times\frac{53}{38} = \frac{2}{19}$$

$$\beta_{22}(z_5 - c_5) - y_{25}c_B\beta_2 = \frac{8}{38}\times\frac{1}{38} - \frac{-1}{19}\times\frac{53}{38} = \frac{3}{38}$$

$$\beta_{22}(z_6 - c_6) - y_{26}c_B\beta_2 = \frac{8}{38}\times\frac{53}{38} - \frac{8}{38}\times\frac{53}{38} = 0$$

As a result, the range for the discrete change in element a_{24} that will preserve the solution's optimality is provided by

$$Max.\left\{\frac{-1/2}{2/19}, \frac{-1/38}{3/38}\right\} \le \Delta a_{24} \le Min\left\{\frac{-169/38}{-23/38}\right\}$$

i.e., $\frac{-1}{3} \le \Delta a_{24} \le \frac{169}{23}$

Further, since for the given element a_{24}, $r = 2$ and $k = 4$, therefore, we have

$$x_{B1}\beta_{22} - x_{B2}\beta_{31} = \frac{16}{19}\times\frac{8}{38} - \frac{5}{19}\times\frac{-1}{38} = \frac{7}{38}$$

$$x_{B3}\beta_{22} - x_{B2}\beta_{32} = \frac{126}{19}\times\frac{8}{38} - \frac{5}{19}\times\frac{-15}{38} = \frac{3}{2}$$

Thus, the feasible range for the discrete change in the element a_{24} is given by

$$Max.\left\{\frac{-16/19}{7/38}, \frac{-129/19}{3/2}\right\} \le \Delta a_{24}$$

i.e., $\frac{-84}{19} \le \Delta a_{24}$.

$$\therefore 5 - \frac{84}{19} \le a_{24} < \infty \text{ or } \frac{11}{19} \le a_{24} < \infty \qquad [\because a_{24} = 5]$$

Similarly the ranges for changes in the elements a_{14}, a_{34} etc. can be obtained.

6.7 *STRUCTURAL CHANGES*

Structural changes in an optimization problem refer to modifications in the formulation of the problem itself, which can alter its structure and potentially change the optimal solution. These changes are more significant than simple adjustments to parameters and can include adding or removing constraints or variables, changing the relationships between variables, or altering the objective function. Here's an in-depth look at structural changes and their implications:

Types of Structural Changes

1. **Adding or Removing Constraints**: Introducing new constraints or eliminating existing ones can change the feasible region of the problem.
2. **Adding or Removing Variables**: Adding new decision variables or removing existing ones can impact the solution space and the optimal solution.
3. **Changing Constraint Relationships**: Modifying the relationships between variables within constraints, such as changing from an inequality to an equality constraint.
4. **Modifying the Objective Function**: Altering the objective function to reflect new goals or priorities.

Impact of Structural Changes

Structural changes can lead to significant shifts in the feasible region, the optimal solution, and the overall problem complexity. The impact of these changes needs to be carefully analyzed to understand their implications.

Example Scenarios

1. Adding a Constraint:

- Consider an optimization problem where a new constraint is introduced to reflect a new resource limitation or requirement.
- This new constraint may reduce the feasible region, potentially eliminating the current optimal solution and necessitating a search for a new optimal solution.

2. Removing a Constraint:

- Removing an existing constraint can enlarge the feasible region, possibly leading to a different optimal solution that better utilizes the available resources.

3. Adding a Variable:

- Introducing a new decision variable to represent a new activity or option can change the optimization landscape. The new variable might provide additional flexibility and lead to a different optimal solution.

4. Changing the Objective Function:

- Modifying the objective function, such as changing from maximizing profit to minimizing cost, can fundamentally alter the optimization approach and result in a different solution.

Analysis of Structural Changes

To analyze structural changes, the following steps are typically undertaken:

1. **Reformulate the Problem**: Update the problem formulation to reflect the structural changes, ensuring all new constraints and variables are properly incorporated.
2. **Solve the Modified Problem**: Use appropriate optimization techniques to solve the newly structured problem.
3. **Compare Solutions**: Compare the new solution with the original one to understand the impact of the structural changes.
4. **Perform Sensitivity Analysis**: Conduct sensitivity analysis on the new problem formulation to understand how robust the new solution is to changes in the parameters.

Example 7: Consider a linear programming problem:

Maximize, $Z = 3x_1 + 5x_2$

Subject to: $x_1 + 2x_2 \leq 4$

$2x_1 + x_2 \leq 5$; $x_1, x_2 \geq 0$

Let's introduce a new constraint:

$x_1 + x_2 \leq 3$

The new problem becomes:

Maximize, $Z = 3x_1 + 5x_2$

Subject to: $x_1 + 2x_2 \leq 4$; $2x_1 + x_2 \leq 5$; $x_1 + x_2 \leq 3$; $x_1, x_2 \geq 0$

Solving this new problem may yield a different optimal solution, as the feasible region has been reduced by the new constraint.

Structural changes in an optimization problem can have significant impacts, necessitating a thorough re-evaluation of the problem formulation and solution. By systematically analyzing these changes, decision-makers can understand their implications, ensure that the new solution is optimal, and make informed decisions in light of the modified problem structure.

Example 8: In a LPP problem

Maximize, $Z = 3x_1 + 5x_2$

Subject to: $x_1 \le 4$; $3x_1 + 2x_2 \le 18$ and $x_1, x_2 \ge 0$

If a new variable, x_5, is added to the equation with $c_5 = 7$ and $a_5 = [1, 2]$, discuss about the impact of the addition and, if necessary, find the updated answer.

Solution: Introducing the slack variables $s_1 \ge 0, s_2 \ge 0$ and then solving by simplex method; the optimum solution is given in the following table.

c_B	y_B	x_B	y_1	y_2	y_3	y_4
0	y_3	4	1	0	1	0
5	y_2	9	3/2	1	0	1/2
		Z =45	9/2	0	0	5/2

We can see from the above table that

$$x_B = [4, 9] \text{ and } B^{-1} = \begin{pmatrix} 1 & 0 \\ 0 & 1/2 \end{pmatrix}$$

Since a new variable $x_5 = B^{-1}a_5 = \begin{pmatrix} 1 & 0 \\ 0 & 1/2 \end{pmatrix}[1, 2] = [1, 1]$

And $z_5 - c_5 = c_B{}^T y_5 - c_5 = (0, 5)[1, 1] - 7 = -2$

This suggests that there is a violation of the optimality criterion. Thus, by adding y_5 to the basis, a new optimal solution can be found with ease:

c_B	y_B	x_B	y_1	y_2	y_3	y_4	y_5
0	y_3	4	1	0	1	0	**1**
5	y_2	9	3/2	1	0	1/2	1
		Z =45	9/2	0	0	5/2	2

Final iteration: Optimal solution

c_B	y_B	x_B	y_1	y_2	y_3	y_4	y_5
0	y_3	4	1	0	1	0	**1**
5	y_2	9	3/2	1	0	1/2	1
		Z =45	9/2	0	0	5/2	2

Optimum solution is, $x_1 = 0, x_2 = 5$ and $x_5 = 4$ with maximum $z = 53$

Remarks: It is interesting to note that $z_5 - c_5 = 2 > 0$ would result if instead of 7. Therefore, the post-optimal addition of x_5 has no effect on the given problem's optimality.

Example 9: Take a look at the table below, which shows an optimal solution to a particular linear programming issue.

		$c_j \longrightarrow$	2	4	1	3	2	0	0	0
c_B	Vectors in basis y_B	x_B	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8
2	y_1	3	1	0	0	-1	0	0.5	0.2	-1
4	y_2	1	0	1	0	2	1	-1	0	0.5
1	y_3	7	0	0	1	-1	-2	5	-0.3	2
	Z	17	0	0	0	2	0	2	0.1	2

If the additional constraint $2x_1 + 3x_2 - x_3 + 2x_4 + 4x_5 \leq 5$ were annexed to the system, would the ideal solution alter in any way? Explain your response.

Solution: The Optimum solution from the table is, $x_1 = 3,\ x_2 = 1$ and $x_3 = 7$ (basic)

$x_4 = x_5 = x_6 = x_7 = x_8 = 0$ (non-basic)

This also meets the recently added restrictions.

$2x_1 + 3x_2 - x_3 + 2x_4 + 4x_5 \leq 5$

Therefore, adding the aforementioned limitation to the original problem does not affect its optimal basic feasible solution.
Because of this, the extra constraints are unnecessary, and the best solution for the existing LPP likewise works best for the new LPP.

Example 10: In the LPP of example 9, let's remove the variable x_2. Find the best answer for the resultant LPP.

Solution: We set a cost of -M to x_2 because the variable that needs to be eliminated is a basic variable, and we use the best simplex table as the starting simplex table for the new LPP. Consequently, we have

Initial iteration: Introduce y_4 and drop y_2.

		$c_j \longrightarrow$	2	-M	1	3	2	0	0	0
c_B	Vectors in basis y_B	x_B	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8
2	y_1	3	1	0	0	-1	0	0.5	0.2	-1
-M	y_2	1	0	1	0	**2**	1	-1	0	0.5
1	y_3	7	0	0	1	-1	-2	5	-0.3	2
	Z	-M+13	0	0	0	-2M-6	-M-4	M+6	0.1	-0.5M+1

First iteration: Introduce y_5 and drop y_4.

c_B	Vectors in basis y_B	x_B	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8
2	y_1	7/2	1	1/2	0	0	1/2	0	0.2	-3/4
3	y_4	1/2	0	1/2	0	**1**	**1/2**	-1/2	0	1/4
1	y_3	15/2	0	1/2	1	0	-3/2	9/2	-0.4	9/4

	Z	16	0	M+3	0	0	-1	3	0.1	3/2

Final iteration: optimal solution

c_B	Vectors in basis y_B	x_B	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8
2	y_1	3	1	0	0	-1	0	0.5	0.2	-1
2	y_5	1	0	1	0	**2**	1	-1	0	0.5
1	y_3	9	0	2	1	3	0	3	-0.3	3
	Z	17	0	M+4	0	2	0	2	0.1	2

Optimal solution obtained above is:

$x_1 = 3, x_2 = 0, x_3 = 9, x_4 = 0 \quad and \quad x_5 = 1$; Max $z = 17$

Check your progress

Problem 1: Obtain the optimum solution of the LPP

Maximize $Z = 15x_1 + 45x_2$ subject to the constraints:

$x_1 + 6x_2 \le 240;\ 5x_1 + 2x_2 \le 162;\ x_2 \le 50; x_1, x_2 \ge 0$

If maximum $z = \sum c_j x_j,\ j = 1, 2$ and c_2 is kept fixed at 45, determine how much can c_1 be changed without affecting the above optimal solution.

Answer: Optimal solution: $x_1 = 27.1, x_2 = 13.3$; Maximum $z = 1005$; $2.8 \le c_1 \le 112.5$

6.8 *SUMMARY*

Post-optimal analysis is a vital tool in optimization that enhances the understanding of how optimal solutions behave under different conditions. By conducting this analysis, decision-makers can gain confidence in their solutions, prepare for uncertainties, and make more robust and informed decisions. In this unit we have learned the following:

1. **Range Analysis**: Determining the range over which objective function coefficients or right-hand side values of constraints can vary without changing the optimal solution.
2. **Impact Assessment**: Evaluating the effect of these changes on the objective function value and the optimal solution.
3. **Shadow Prices**: Identifying the value of additional resources and understanding which constraints are binding and which have slack.
4. **Scenario Analysis**: Considering different scenarios and their potential impacts on the solution, such as changes in resource availability or costs.

By conducting post-optimal analysis, decision-makers can better understand the robustness of the optimal solution and make informed adjustments based on possible future changes.

6.9 *GLOSSARY*

- Sensitivity analysis
- Post-Optimal Analysis

6.10 *REFERENCES*

- Mokhtar S. Bazaraa, John J. Jarvis and Hanif D. Sherali: *Linear Programming and Network Flows* (4th edition). John Wiley and Sons, 2010.
- Hamdy A. Taha: *Operations Research: An Introduction (10th edition). Pearson*, 2017.
- Paul R. Thie and Gerard E. Keough: *An Introduction to Linear Programming and Game Theory* (3rd edition), Wiley India Pvt. Ltd, 2014.
- Swarup, K., Gupta, P. K., & Mohan, M. (2017). An introduction to management science operation research. *Sultan Chand & Sons educational publishers, New Delhi*.
- OpenAI. (2024). *ChatGPT (August 2024 version) [Large language model]*. OpenAI. https://www.openai.com/chatgpt

6.11 *SUGGESTED READING*

- G. Hadley, *Linear Programming*, Narosa Publishing House, 2002.

- Frederick S. Hillierand Gerald J. Lieberman: *Introduction to Operations Research* (10th edition). McGraw-Hill Education, 2015.

- https://epgp.inflibnet.ac.in/Home/ViewSubject?catid=ZLCHeZEhCZ8yCri36nSF3A==

6.12 *TERMINAL QUESTION*

Short Answer Type Question:

1. Examine how the optimal solution is influenced by discrete alterations in the requirement vector for the given Linear Programming Problem (LPP)

Maximize, $z = 2x + y$

Subject to the constraints,

$3x + 5y \le 15;\ 6x + 2y \le 24;\ x, y \ge 0;$

Answer: $x_1 = 23/3, x_2 = 0, x_3 = 5$; Maximum $z = 85/3$

2: Determine the range within which c_3, c_4 and b_2 can be varied while preserving the optimality of the current solution in the specified Linear Programming Problem (LPP).

Maximize, $z = 3x + 5y + 4z$

Subject to the constraints,

$2x + 3y \le 8;\ 2x + 5y \le 10;\ 3x + 2y + 4z \le 15; x, y, z \ge 0;$

3: In the given LPP

Minimize, $z = 3x + 6y + z$

Subject to the constraints, $x + y + z \ge 6;\ x + 5y - z \ge 4; x + 5y + z \ge 24; x, y, z \ge 0;$

Solve the Linear Programming Problem (LPP) and analyze the impact of modifying the requirement vector from [6, 4, 24] to [6, 2, 12] on the optimal solution.

4: In the given LPP

Minimize, $z = x_2 - 3x_3 + 2x_5$

Subject to the constraints, $3x_2 - x_3 + 2x_5 \le 7$; $-2x_2 + 4x_3 \le 12$; $-4x_2 + 3x_3 + 8x_5 \le 10$; $x_2, x_3, x_5 \ge 0$

The optimal solution table is given by

c_B	y_B	x_B	y_1	y_2	y_3	y_4	y_5	y_6
-1	y_2	4	2/5	1	0	1/10	4/5	0
3	y_3	5	1/5	0	1	3/10	2/5	0
0	y_6	11	1	0	0	-1/2	10	1
	Z	11	1/5	0	0	4/5	12/5	0

(a) Determine how much c_5 can be decreased before y_5 enters the basis

(b) Determine how much the requirement of 7 in the first constraint can be increased before the basis changes.

5: In the given LPP

Maximize, $z = 4x_1 + 3x_2 + 4x_3 + 6x_4$

Subject to, $x_1 + 2x_2 + 2x_3 + 4x_4 \le 8; 2x_1 + 2x_3 + x_4 \le 6$;

$3x_1 + 3x_2 + x_3 + x_4 \le 8; x_1, x_2, x_3, x_4 \ge 0$

(a) Identify the individual ranges for discrete changes in a_{12}, a_{22} and a_{23} that are consistent with maintaining the optimal solution of the given LPP.

(b) If a_{11} is changed to $a_{11} + \Delta a_{11}$, determine the allowable limit for the discrete change Δa_{11} in order to preserve the optimality of the current solution.

6: Let the optimum simplex table for a maximization problem be

c_B	y_B	x_B	y_1	y_2	y_3	y_4	y_5
12	y_2	8/5	0	1	-1/5	2/5	-1/5
5	y_1	9/5	1	0	7/5	1/5	2/5
	Z	141/5	0	0	3/5	29/5	M-2/5

Where x_4 is the slack variable and x_5 is an artificial variable, introduce a new variable x_6 into the problem with a cost of 30 in the objective function. The coefficients of x_5 in the two constraints are 5 and 7, respectively. Discuss the impact of adding the new variable and obtain the revised solution if applicable.

Long answer type question:

1. A manufacturer produces four products A, B, C, and D which are processed on three machines: X, Y, and Z. The time needed to produce one unit of each product and the capacity of each machine are detailed in the following table.

Product	Processing time (in hrs)		
	Machine X	Machine Y	Machine Z
A	1.5	4	2
B	2	1	3
C	4	2	1
D	3	1	2
Capacity (hours)	550	700	200

The profit contribution per unit of the four products A, B, C and D is Rs. 4, 6, 3 and 1 respectively. The manufacturer wants to determine its optimal product-mix.

(a) Formulate the problem as a linear programming model.

(b) Solve it using the simplex method.

(c) Determine the optimal product mix and the total maximum profit contribution.

(d) Identify which constraints are binding.

(e) Determine which constraints have excess capacity and by how much.

(f) Assess whether an increase of Rs. 2 per unit in the profit contribution from product Y will affect the optimal product mix.

(g) Evaluate if shutting down machine X for 50 hours for repairs will alter the product mix.

(h) Calculate the shadow prices of machine-hours for the three machines.

Objective type question:

1: Choose the correct correct option for the statement "Post-optimal analysis is a technique to"

(a) Analyze how the optimal solution to a Linear Programming Problem (LPP) is affected by changes in the problem inputs.

(b) Distribute resources in the most effective way.

(c) Minimize the operational costs.

(d) Describe the relationship between the dual problem and its primal.

2: Addition of a new constraints in the existing constraints will ensure a

(a) Change in the coefficient a_{ij}.

(b) Change in the objective function coefficient c_j.

(c) Both (a) and (b).

(d) Neither (a) nor (b)

3: To achieve the maximum marginal increase in the objective function value, it is advisable to increase the value of a resource with the highest shadow price

(a) Smaller.

(b) larger.

(c) Both (a) and (b).

(d) Neither (a) nor (b)

Fill in the blanks:

1: Post-optimality analysis study only the continuous changes in the parameter of …………………….

2: Optimum solution to an LPP is not very sensitive to the changes in the RHS values of the ………………

3: The optimality of the current solution may be affected if right hand side of the constraints is …………

6.13 *ANSWERS*

Answer of short answer type question

1: $12 \le b_1 \le 60$ and $6 \le b_2 \le 30$

2: $1.25 \le c_3 \le 11.5$; $c_4 \le 1.10$; $3.75 \le b_2 \le 17.42$

3: (a) $x_1 = 14; x_2 = 0; x_3 = 10;$ Minimum $z = 52$

(b) $x_1 = 7; x_2 = 0; x_3 = 5;$ Minimum = 26

4: (a) $c_5 \le -2/5$; (b) $-3 \le b_1 < \infty$

5: (a) $(15/16) \le a_{12}$; $(-17/6) \le a_{22}$ and $(-1/8) \le a_{23}$

(b) $-3 \le \Delta a_{11} \le 17/21$

6: $x_1 = 0; x_2 = 25/19; x_5 = 9/19;$ Maximum $z = 30$

Answer of long answer type question

1: Maximize $z = 4x_1 + 6x_2 + 3x_3 + x_4$

Subject to the constraints

$1.5x_1 + 2x_2 + 4x_3 + 3x_4 \le 550$; $4x_1 + x_2 + 2x_3 + x_4 \le 700$

$2x_1 + 3x_2 + x_3 + 2x_4 \le 200$; $x_j \ge 0\,(j = 1,2,3,4)$

(c) $x_1 = 0; x_2 = 25; x_3 = 125;$ and $x_4 = 0$; Maximum profit= Rs. 525

(d) First and third constraints (e) Machine Y have excess capacity, 425 hours

(f) Product-mix will not change (g) Product-mix will not change

(h) Machine X: Re. 0.3, Machine Y: Nil; Machine Z= Re. 0.18

Answer of Objective type question

Answer: **1:** (a) **2:** (c) **3:** (b)

Answer of fill in the blank question

Answer: 1: LPP **2:** Constraints **3:** Changed

UNIT-7: INTEGER PROGRAMMING

CONTENTS:

7.1 *INTRODUCTION*

Integer programming is a type of optimization where some or all of the variables are required to be integers. It is commonly used for problems where solutions must be whole numbers, such as scheduling, allocation, and logistics. Integer programming can be more complex and computationally intensive than linear programming due to the discrete nature of the variables.

7.2 *OBJECTIVE*

After reading this unit learners will be able to

- Understand the basic concept of Integer programming problem.
- Learn about the Gomory's method.
- Visualized about fractional cut method.
- Learn about the branch and bound method.

7.3 *INTEGER LINEAR PROGRAMMING*

Integer programming is a specialized branch of mathematical optimization that focuses on problems where some or all of the decision variables are required to be integers. This is particularly important in situations where fractional solutions are not practical or possible, such as in scheduling, allocation, or logistics where quantities must be whole numbers.

Key Concepts

1. Variables: Unlike linear programming, integer programming mandates that decision variables take integer values. This can be further categorized into:

- **Pure Integer Programming**: All decision variables must be integers.
- **Mixed-Integer Programming (MIP)**: Some decision variables are integers, while others can be non-integers.
- **Binary Integer Programming**: Variables are restricted to binary values (0 or 1).

2. Formulation: Integer programming problems are formulated similarly to linear programming problems but with the added integer constraint on the variables. A standard integer programming problem can be expressed as:

Maximize or Minimize $c^T x$

Subject to $Ax \leq b$

$x_i \in Z \ \ \forall i$

Where, x is the vector of decision variables, c and b are vectors of coefficients, A is a matrix of coefficients, and x_i are required to be integers.

Applications

Integer programming is used in various fields, including:

- **Operations Research**: Optimizing resource allocation, production scheduling, and supply chain management.
- **Finance**: Portfolio optimization and capital budgeting.
- **Logistics**: Vehicle routing and facility location planning.
- **Manufacturing**: Production planning and job scheduling.

Challenges

Integer programming is generally more complex than linear programming because the solution space is discrete rather than continuous, making it harder to solve. The problem complexity can increase exponentially with the number of variables and constraints, leading to longer computation times.

Solution Techniques

Several methods are used to solve integer programming problems, including:

- **Exact Algorithms**: Such as Branch and Bound, Branch and Cut, and Cutting Plane methods.
- **Heuristics**: Including Genetic Algorithms, Simulated Annealing, and Tabu Search, which provide approximate solutions more quickly for large or complex problems.

Integer programming is a powerful tool for optimizing problems that require discrete decisions. Despite its complexity, advancements in computational techniques and algorithms continue to enhance its applicability and efficiency in solving real-world problems.

7.4 *PURE AND MIXED INTEGER PROGRAMMING PROBLEMS*

Integer programming problems can be categorized into two main types: pure integer programming and mixed integer programming. Both types involve optimization where some or all of the decision variables are required to be integers. Here's a closer look at each type:

Pure Integer Programming

In pure integer programming, all the decision variables are constrained to be integers. This type of problem is used when all variables in the optimization model must take on whole number values.

Example 1: Consider a factory that produces two types of products (Product 1 and Product 2). The objective is to maximize the profit given the constraints on resources (e.g., labor, materials).

Maximize: $Z = 4x_1 + 7x_2$
Subject to: $3x_1 + 2x_2 \le 12$

$5x_1 + 3x_2 \leq 15$

$x_1 \geq 0, x_2 \geq 0\,;\ x_1, x_2 \in Z$

In this example, x_1 and x_2 represent the quantities of Product 1 and Product 2, respectively, and both must be integers.

Mixed Integer Programming (MIP)

In mixed integer programming, only some of the decision variables are required to be integers, while others can be continuous. This type of problem is useful when some decisions are inherently discrete (e.g., number of units produced), while others can vary continuously (e.g., amounts of resources used).

Example 2: Consider a company that wants to determine the optimal production quantities for two products, where one of the products can be produced in fractional quantities (e.g., a liquid), and the other must be in whole units.

Maximize: $Z = 4x_1 + 7x_2$
Subject to: $3x_1 + 2x_2 \leq 12\,;\ 5x_1 + 3x_2 \leq 15\,;\ x_1 \geq 0\,; x_2 \geq 0\,;\ x_1 \in Z$

Here, x_1 (the integer variable) might represent the number of whole units of a product, while x_2 (the continuous variable) represents a product that can be produced in any amount.

Applications

Pure Integer Programming:

- **Scheduling**: Assigning whole shifts to employees.
- **Knapsack Problem**: Determining the number of items to fit into a limited space.
- **Vehicle Routing**: Deciding the number of trucks needed for deliveries.

Mixed Integer Programming:

- **Production Planning**: Optimizing production quantities with both discrete and continuous resources.
- **Capital Budgeting**: Making investment decisions where some investments can be scaled (continuous) and others are fixed (discrete).
- **Network Design**: Determining the number of facilities (discrete) and the flow of goods (continuous).

Pure and mixed integer programming problems provide a framework for solving a wide range of real-world optimization problems. By incorporating integer constraints, these methods enable more realistic modeling of scenarios where decisions are not continuous, thereby offering more practical and applicable solutions.

7.5 *GOMORY's ALL I.P.P. METHOD*

Gomory's All-Integer Programming Problem (All-I.P.P.) method is a technique used to solve integer programming problems. Developed by Ralph Gomory in the 1950s, this method is a cutting-plane algorithm specifically designed to handle integer constraints. The general approach involves solving a series of linear programming relaxations and iteratively adding cuts (constraints) to eliminate non-integer solutions, gradually converging to an integer solution.

Steps in Gomory's All-I.P.P. Method

1. **Solve the Linear Programming Relaxation**:
- Solve the original integer programming problem without the integer constraints, treating it as a standard linear programming problem.
- This step provides an optimal solution to the relaxed problem, which may not be an integer solution.

2. **Identify the Fractional Variables**:
- Examine the optimal solution from the LP relaxation. Identify any decision variables that have non-integer values.

3. **Generate Gomory Cuts**:

- Write the equation corresponding to the chosen fractional basic variable from the simplex tableau: $x_i + \sum_j a_{ij} x_j = b_i$
- Isolate the fractional parts of the coefficients and the right-hand side: $f_i + \sum_j a_{ij} x_j = b_i$
- The Gomory cut is derived as: $\sum_j \left(a_{ij} - \lfloor a_{ij} \rfloor\right) x_j \geq b_i - \lfloor b_i \rfloor$
- Here, $\lfloor . \rfloor$ denotes the floor function, which returns the greatest integer less than or equal to the given number.

4. **Add the Cut to the LP**:
- Add the newly generated Gomory cut to the original set of constraints.
- This modifies the feasible region of the LP by cutting off the current non-integer solution.

5. **Re-Solve the LP**:
- Solve the modified linear programming problem with the added cut.
- Repeat the process of generating cuts and re-solving until an integer solution is found.

6. **Check for Optimality**:

- Once an integer solution is obtained, check if it is optimal.
- If it is not optimal, further cuts might be necessary, or another branch-and-bound approach may be combined to refine the solution.

Example 3: Consider a simple integer programming problem

- **Objective Function**: Maximize $Z = 3x_1 + 2x_2$
- **Constraints**:
- $2x_1 + x_2 \leq 4$
- $x_1 + 2x_2 \leq 3$
- $x_1, x_2 \geq 0$
- $x_1, x_2 \in Z$

Step-by-Step Illustration:

1. **Solve LP Relaxation**:
- Relax the integer constraints and solve the LP problem.
- Suppose the optimal solution to the LP relaxation is $x_1 = 1.5, x_2 = 1.0$
2. **Identify Fractional Variables**:
- The current solution is $x_1 = 1.5$, which is fractional.
3. **Write the Equation for the Fractional Basic Variable**:
- Suppose the optimal tableau provides the following equation for x_1: $x_1 + 0.5x_2 = 1.5$
4. **Isolate the Fractional Parts**:
- The fractional part of $x_1 = 1.5$ is 0.5
- The equation can be written as: $0.5 + 0.5x_2 = 1.5$
5. **Generate the Gomory Cut**:
- The Gomory cut is derived as: $0.5x_2 \geq 1.5 - 1$
- Simplifying this, we get: $0.5x_2 \geq 0.5 \Rightarrow x_2 \geq 1$
6. **Add the Gomory Cut to the Original Constraints**:
- The new constraint $x_2 \geq 1$ is added to the original set of constraints.
- This modifies the feasible region of the LP to exclude the current non-integer solution.
7. **Re-Solve the LP**:
- Solve the modified LP problem with the added cut.
- Repeat the process until an integer solution is found.

Gomory's constraints are a powerful tool in integer programming, helping to iteratively eliminate non-integer solutions and converge to an optimal integer solution. The process involves generating cuts from the fractional parts of the basic variables in the optimal simplex tableau and adding these cuts to the set of constraints. This method is systematic and can be combined with other techniques like branch-and-bound to solve complex integer programming problems efficiently.

7.6 *FRACTIONAL CUT METHOD-ALL INTEGER LPP*

The Fractional Cut Method is a technique used specifically for solving all-integer linear programming problems. It involves iteratively adding constraints (cuts) to eliminate non-integer solutions while preserving feasible integer solutions. Here's a detailed breakdown of the method:

Overview

The Fractional Cut Method is used to handle integer programming problems by leveraging the results of linear programming relaxations and generating constraints to eliminate fractional solutions. It is particularly effective in conjunction with other integer programming techniques.

Steps in the Fractional Cut Method

1. **Solve the Linear Programming Relaxation**:
- Solve the integer programming problem (IPP) by first relaxing the integer constraints, treating it as a linear programming problem (LPP). This provides an optimal solution to the relaxed problem.

2. **Identify Fractional Solutions**:
- Examine the optimal solution of the LP relaxation. Identify which variables have fractional values. If the solution is entirely integer, then it is already optimal.

3. **Generate Fractional Cuts**:
- For each fractional solution, generate a cutting plane (cut) that eliminates the current fractional solution while keeping all feasible integer solutions. The cut is derived from the simplex tableau.

4. **Add the Cut to the LP**:
- Incorporate the new cut into the existing constraints of the LP. This effectively narrows the feasible region to exclude the fractional solution.

5. **Re-Solve the LP**:
- Solve the modified LP problem with the added cut. Repeat the process of identifying fractional solutions and generating cuts until an integer solution is found.

6. **Check for Integer Solutions**:

- After adding each cut and solving, check if the resulting solution is an integer. If it is, then this is the optimal integer solution. If not, continue with the process of generating and adding cuts.

Example 4: Consider the following integer programming problem:

- **Objective Function**: Maximize $Z = 2x_1 + 3x_2$
- **Constraints**:
  - $x_1 + 2x_2 \leq 5$
  - $2x_1 + x_2 \leq 6$
  - $x_1, x_2 \geq 0$
  - $x_1, x_2 \in Z$

Step-by-Step Illustration:

1. **Solve the LP Relaxation**:
- Relax the integer constraints and solve the LP problem. Suppose the optimal solution is $x_1 = 2.5, x_2 = 1.5$.
2. **Identify Fractional Variables**:
- The optimal solution $x_1 = 2.5, x_2 = 1.5$ are both fractional.
3. **Generate Fractional Cuts**:
- From the simplex tableau, we might find a cut for a fractional basic variable. Assume the cut derived is: $x_1 + x_2 \leq 3$
4. **Add the Cut to the LP**:
- Add the constraint $x_1 + x_2 \leq 3$ to the LP constraints.
5. **Re-Solve the LP**:
- Solve the modified LP problem. Suppose the new optimal solution is $x_1 = 2, x_2 = 1$, which is an integer solution.
6. **Check for Integer Solutions**:
- The new solution $x_1 = 2, x_2 = 1$ is an integer and satisfies the constraints.

Key Points

- **Generating Cuts**: The cuts are generated based on the fractional parts of the variables in the optimal solution. These cuts are typically derived from the simplex tableau and are designed to exclude the fractional solution.
- **Iterative Process**: The method involves iteratively adding cuts and solving the modified LP until an integer solution is found.

- **Integration with Other Methods**: The Fractional Cut Method is often used in conjunction with other techniques, such as branch-and-bound, to handle more complex integer programming problems efficiently.

Benefits and Challenges

Benefits:

- Provides a systematic way to eliminate fractional solutions.
- Can be combined with other optimization techniques for improved performance.

Challenges:

- May require a large number of iterations and cuts for complex problems.
- Computationally intensive for large-scale problems.

The Fractional Cut Method is a powerful tool in integer programming, providing a means to solve integer problems by systematically refining the feasible region of the LP relaxation.

Example 5: In the given LPP evaluate the optimum integer solution

Maximize $Z = x_1 + 4x_2$

Subject to; $2x_1 + 4x_2 \le 7$; $5x_1 + 3x_2 \le 15$; $x_1, x_2 \ge 0$ such that $x_1, x_2 \in Z$

Solution: Initially, we add the slack variable $s_1 \ge 0$ and $s_2 \ge 0$, an initial basic feasible solution is $s_1 = 7$ and $s_2 = 15$. Using the simplex method, an optimal non-integer solution is achieved, and it is presented in the following simplex table:

Initial iteration: Non-integer optimum solution

c_B	y_B	x_B	y_1	y_2	y_3	y_4
4	y_2	7/4	1/2	1	1/4	0
0	y_4	39/4	7/2	0	-3/4	1
	z	7	1	0	1	0

Step 2: Because of the optimal solution is not an integer, we focus solely on the fractional parts of $x_{B1} = \frac{7}{4}\left(=1+\frac{3}{4}\right)$ and $x_{B2} = \frac{39}{4}\left(=9+\frac{3}{4}\right)$.

Step 3: Max $\{f_1, f_2\} = Max\left\{\frac{3}{4}, \frac{3}{4}\right\} = \frac{3}{4}$ and $x_{B2} = \frac{39}{4}\left(=9+\frac{3}{4}\right)$ i.e., both f_1 and f_2 are equal. Therefore, we arbitrarily select one of these fractional parts. For instance, let's choose f_2.

Step 4: In the second row, since $y_{23} = -3/4$, we write $y_{23} = -1+1/4$

Step 5: Let G_1 denote the first Gomory slack. We can then express it as follows:

$$G_1 = -f_{20} + f_{21} + f_{22}x_2 + f_{23}x_3 + f_{24}x_4 = -\frac{3}{4} + \frac{1}{2}x_1 + 0.x_2 + \frac{1}{4}x_3 + 0.x_4$$

Step 6: By adding this additional constraint to the optimal simplex table, we obtain:

First Iteration: Drop G_1 and introduce y_1

c_B	y_B	x_B	y_1	y_2	y_3	y_4	G_1
4	y_2	7/4	1/2	1	1/4	0	0
0	y_4	39/4	7/2	0	-3/4	1	0
0	G_1	-3/4	**-1/2**	0	-1/4	0	1
	z	7	1	0	1	0	0

Since, the optimum solution is still non-integral, we introduce the second Gomorian constraints.

Now, $x_{B3} = -3/4$ only is negative, this basic variable leaves the basis. Further, since Max. $Max\left\{\frac{(z_j - c_j)}{y_{3j}}, y_{3j} < 0\right\} = \max\left\{\frac{1}{-1/2}, \frac{1}{-1/4}\right\} = -2, y_1$ enters the basis, i.e., x_1 becomes basic variable in place of G_1.

Second iteration: Non-integer optimal solution.

c_B	y_B	x_B	y_1	y_2	y_3	y_4	G_1
4	y_2	1	0	1	0	0	1
0	y_4	9/2	0	0	-5/2	1	7
1	y_1	3/2	1	0	1/2	0	-2
	z	11/2	0	0	1/2	0	2

Since the optimal solution is still non-integral, we introduce the second Gomory constraint. Now,

$$X_{B2} = \frac{9}{2}\left(= 4 + \frac{1}{2}\right) \text{ and } X_{B3} = \frac{3}{2}\left(= 1 + \frac{1}{2}\right)$$

Since, Max $\{f_2, f_3\} = Max\left\{\frac{1}{2}, \frac{1}{2}\right\} = \frac{1}{2}$ i.e., both f_2 and f_3 are equal. So, let us choose $f_2 = 1/2$ and write $y_{23} = -6 + 1/2$.

$$\therefore G_2 = -f_{20} + f_{21}x_1 + f_{22}x_2 + f_{23}x_3 + f_{21}x_4 = -1/2 + 0.x_1 + 0.x_2 + (1/2)x_3 + 0.x_4$$

Adding these additional constraints in the second iterative table, we have

Third iteration: Drop G_2 and introduce y_3.

c_B	y_B	x_B	y_1	y_2	y_3	y_4	G_1	G_2
4	y_2	1	0	1	0	0	1	0
0	y_4	9/2	0	0	-5/2	1	7	0
1	y_1	3/2	1	0	1/2	0	-2	0
0	G_2	-1/2	0	0	$-1/2*$	0	0	1
	z	11/2	0	0	1/2	0	2	0

Final iteration: Optimal Solution in the integers.

c_B	y_B	x_B	y_1	y_2	y_3	y_4	G_1	G_2
4	y_2	1	0	1	0	0	1	0
0	y_4	7	0	0	0	1	7	-5
1	y_1	1	1	0	0	0	-2	1
0	y_3	1	0	0	1	0	0	-2
	z	5	0	0	0	0	2	1

The table indicates that the optimal basic feasible solution has been achieved. Therefore, the optimal solution is

$x_1 = 1, x_2 = 1$ and Maximum $z = 5$.

7.7 FRACTIONAL CUT METHOD FOR MIXED INTEGER LPP

The Fractional Cut Method, also known as Gomory's Cut Method, can be applied to Mixed Integer Linear Programming Problems (MILPP). These problems involve both integer and non-integer (continuous) variables. The method iteratively adds constraints to the linear programming relaxation to eliminate non-integer solutions for the integer-constrained variables.

Steps in the Fractional Cut Method for Mixed Integer LPP

1. **Solve the Linear Programming Relaxation**:
- Solve the MILP problem by ignoring the integer constraints on the integer variables. This gives an optimal solution to the relaxed problem.
2. **Identify Fractional Solutions**:
- Examine the solution obtained from the LP relaxation. Identify which of the integer-constrained variables have fractional values.
3. **Generate Fractional Cuts**:
- For each fractional value of an integer-constrained variable, generate a cutting plane (Gomory cut). This cut will eliminate the current fractional solution while keeping all feasible integer solutions intact.
- Gomory cuts are derived from the simplex tableau. For a basic variable x_i with a fractional part f_i, the cut can be formulated to exclude this fractional part.
4. **Add the Cut to the LP**:

- Incorporate the new cut into the existing constraints of the LP. This restricts the feasible region, excluding the fractional solution.

5. **Re-Solve the LP**:
- Solve the modified LP problem with the added cut. Repeat the process of identifying fractional solutions and generating cuts until all integer-constrained variables are integers.

6. **Check for Integer Solutions**:
- After each iteration, check if the resulting solution satisfies the integer constraints. If it does, the optimal solution is found. If not, continue generating and adding cuts.

Example 6: Consider the following mixed integer linear programming problem:

- **Objective Function**: Maximize $z = 5x_1 + 3x_2$
- **Constraints**:

$2x_1 + 3x_2 \le 8$; $x_1 + x_2 \le 3$; $x_1 \ge 0, x_2 \ge 0$; $x_1 \in Z$, x_2 is continuous

Step-by-Step Illustration:

1. **Solve the LP Relaxation**:
- Relax the integer constraint on x_1 and solve the LP problem. Suppose the optimal solution is $x_1 = 2.5, x_2 = 0.5$.

2. **Identify Fractional Variables**:
- The optimal solution $x_1 = 2.5$ is fractional.

3. **Generate Fractional Cuts**:
- Suppose the equation from the simplex tableau for x_1 is $x_1 + 0.5x_2 = 2.5$
- The fractional part of x_1 is 0.5.
- Generate a cut: $x_1 + 0.5x_2 \le 2$

4. **Add the Cut to the LP**:
- Add the constraint $x_1 + 0.5x_2 \le 2$ to the LP.

5. **Re-Solve the LP**:
- Solve the modified LP problem. Suppose the new optimal solution is $x_1 = 2, x_2 = 1$, which satisfies the integer constraint on x_1.

6. **Check for Integer Solutions**:
- The new solution $x_1 = 2, x_2 = 1$ is valid since x_1 is an integer.

Key Points

- **Generating Cuts**: Cuts are derived based on the fractional parts of the integer-constrained variables in the optimal solution of the LP relaxation. These cuts eliminate the fractional solutions.
- **Iterative Process**: The method involves iteratively refining the feasible region by adding cuts and solving the modified LP until all integer-constrained variables are integers.
- **Mixed Variables**: This method can handle problems with both integer and continuous variables, making it versatile for mixed integer programming problems.

Benefits and Challenges

Benefits:

- Systematically eliminates fractional solutions.
- Can be combined with other optimization techniques for enhanced performance.

Challenges:

- May require multiple iterations and cuts for complex problems.
- Computationally intensive for large-scale problems.

The Fractional Cut Method is a robust approach for solving mixed integer linear programming problems, providing a systematic way to refine the feasible region and find optimal integer solutions.

Example 7: Solve the following mixed integer programming problem.

Maximize $Z = 4x_1 + 6x_2 + 2x_3$

Subject to; $4x_1 - 4x_2 \le 5$; $-x_1 + 6x_2 \le 5$; $-x_1 + x_2 + x_3 \le 5$ $x_1, x_2, x_3 \ge 0$ such that $x_1, x_3 \in Z$

Solution: Introducing slack variable $s_1 \ge 0, s_2 \ge 0$. So, the initial basic feasible solution is $s_1 = 5, s_2 = 5$ and $s_3 = 5$.

Disregarding the integer constraints, the optimal solution of the given linear programming problem is shown in the table below.

c_B	y_B	x_B	y_1	y_2	y_3	y_4	y_5	y_6
4	y_1	5/2	1	0	0	3/10	1/5	0

6	y_2	5/4	0	1	0	1/20	1/5	0
2	y_3	25/4	0	0	1	1/4	0	1
	z	30	0	0	0	2	2	2

Since both x_1, x_3 are not integer and $Max.\{f_1, f_3\} = Max..\left\{\frac{1}{2}, \frac{1}{4}\right\} = \frac{1}{2}$; from the first row, we have

$$\left(2 + \frac{1}{2}\right) = x_1 + \frac{3}{10}s_1 + \frac{1}{5}s_2$$

Here, both s_1 and s_2 have positive coefficient, therefore the mixed fractional cut is:

$$G_1 = -\frac{1}{2} + \frac{3}{10}s_1 + \frac{1}{5}s_2$$

First iteration: Drop G_1 and enter y_4

c_B	y_B	x_B	y_1	y_2	y_3	y_4	y_5	y_6	G_1
4	y_1	5/2	1	0	0	3/10	1/5	0	0
6	y_2	5/4	0	1	0	1/20	1/5	0	0
2	y_3	25/4	0	0	1	1/4	0	1	0
0	G_1	-1/2	0	0	0	**-3/10**	-1/5	0	1
	z	30	0	0	0	2	2	2	0

Here, since $x_{B4} < 0, G_1$ leaves the basis since $Max.\left\{\frac{2}{-3/10}, \frac{2}{-1/5}\right\} = Max\left\{-\frac{20}{3}, -10\right\}$ i.e., $\frac{20}{-3}$; y_4 enters the basis.

Second iteration: Improved solution

c_B	y_B	x_B	y_1	y_2	y_3	y_4	y_5	y_6	G_1
4	y_1	2	1	0	0	0	0	0	1
6	y_2	7/6	0	1	0	0	1/6	0	1/6
2	y_3	35/6	0	0	1	0	-1/6	1	5/6
0	y_4	5/3	0	0	0	1	2/3	0	-10/3
	z	80/3	0	0	0	0	2/3	2	20/3

Since x_3 is still not an integer, we refer to the third row of the second iteration.

$$\left(5+\frac{5}{6}\right)=x_3-\frac{1}{6}s_2+s_3+\frac{5}{6}G_1$$

Here, since s_2 has negative coefficient, the mixed fractional cut is:

$$G_1=-\frac{5}{6}+\left(\frac{\frac{5}{6}}{-1+\frac{5}{6}}\right)\left(-\frac{1}{6}\right)s_2+\frac{5}{6}G_1=-\frac{5}{6}+\frac{5}{6}s_2+\frac{5}{6}G_1$$

Third iteration: Drop G_2 and enter y_5

c_B	y_B	x_B	y_1	y_2	y_3	y_4	y_5	y_6	G_1	G_2
4	y_1	2	1	0	0	0	0	0	1	0
6	y_2	7/6	0	1	0	0	1/6	0	1/6	0
2	y_3	35/6	0	0	1	0	-1/6	1	5/6	0
0	y_4	5/3	0	0	0	1	2/3	0	-10/3	0
0	G_2	-5/6	0	0	0	0	**-5/6**	0	-5/6	1
	z	80/3	0	0	0	0	2/3	2	20/3	0

Here, $x_{B5} < 0$ implies G_2 leaves the basis and since y_5 is the only negative in G_2-row, y_5 enters the basis.

Final iteration: Optimum integer solution

c_B	y_B	x_B	y_1	y_2	y_3	y_4	y_5	y_6	G_1	G_2
4	y_1	2	1	0	0	0	0	0	1	0
6	y_2	1	0	1	0	0	0	0	1/6	1/5
2	y_3	6	0	0	1	0	0	1	5/6	-1/5
0	y_4	1	0	0	0	0	0	0	-10/3	4/5
0	y_5	1	0	0	0	1	1	0	1	-6/5
	z	26	0	0	0	0	0	2	20/3	4/5

Thus, the optimal solution which satisfies the condition $x_1, x_3 \in Z$. Hence the required solution is: $x_1 = 2, x_2 = 1, x_3 = 6$ and Max. $Z = 26$

7.8 *BRANCH AND BOUND METHOD*

The Branch and Bound method is a widely used algorithm for solving integer programming problems, including both pure integer and mixed-integer problems. It systematically explores all potential solutions to find the optimal one while efficiently pruning suboptimal solutions to reduce computational effort.

Steps in the Branch and Bound Method

1. Initialization:

- Solve the linear programming relaxation of the integer programming problem (i.e., ignore the integer constraints) to obtain an initial solution. This gives an upper bound for maximization problems and a lower bound for minimization problems.

2. Branching:

- Identify a variable that has a fractional value in the current solution. Create two new subproblems (branches) by adding constraints to this variable to take its floor and ceiling values.
- For example, if $x_i = 3.5$in the current solution, create two new problems: one with $x_i \leq 3$ and the other with $x_i \geq 4$.

3. Bounding:

- Solve the LP relaxation of each new subproblem to obtain new bounds.
- If a subproblem yields an integer solution, compare it with the current best solution and update the best solution if this one is better.
- If the subproblem's bound is worse than the current best solution or infeasible, discard (prune) that branch.

4. Pruning:

- Eliminate branches that cannot yield a better solution than the current best solution. This is done by comparing the bounds of the subproblems to the current best known integer solution.
- Discard infeasible branches or those that lead to worse solutions than the current best solution.

5. Repeat:

- Continue the branching, bounding, and pruning process until all branches have been either explored or pruned. The best solution found during this process is the optimal integer solution.

Example 8: Consider the following integer programming problem:

Objective Function: Maximize $z = 3x_1 + 2x_2$

Constraints:

- $x_1 + x_2 \leq 4$
- $x_1 - x_2 \leq 1$
- $x_1, x_2 \geq 0$
- $x_1, x_2 \in Z$

Step-by-Step Illustration:

1. Initialization:

Solve the LP relaxation:

Maximization $z = 3x_1 + 2x_2$

Subject to, $x_1 + x_2 \leq 4$; $x_1 - x_2 \leq 1$; $x_1, x_2 \geq 0$

- Suppose the optimal solution is $x_1 = 2.5, x_2 = 1.5$ with $z = 10.5$.

2. Branching:

- Create two new subproblems by branching on x_1:
- Subproblem 1: $x_1 \leq 2$
- Subproblem 2: $x_1 \geq 3$

3. Bounding:

- Solve the LP relaxation for Subproblem 1:

Maximize, $Z = 3x_1 + 2x_2$

Subject to, $x_1 + x_2 \leq 4$; $x_1 - x_2 \leq 1$; $x_1 \leq 2$; $x_1, x_2 \geq 0$

- Suppose the optimal solution is $x_1 = 2, x_2 = 2,$ with $z = 10$ (an integer solution).

Solve the LP relaxation for Subproblem 2:

Maximize, $Z = 3x_1 + 2x_2$

Subject to, $x_1 + x_2 \leq 4$; $x_1 - x_2 \leq 1$; $x_1 \geq 3$; $x_1, x_2 \geq 0$

- Suppose this subproblem is infeasible.

4. Pruning:

- Subproblem 2 is pruned because it is infeasible.
- The solution from Subproblem 1 is an integer solution with $z = 10$, so we update our best solution.

5. Repeat:

- Continue the process for any remaining branches (if applicable). In this example, Subproblem 1 has provided an integer solution that is feasible and maximizes z.

The Branch and Bound method is an effective algorithm for solving integer programming problems by exploring and eliminating suboptimal branches systematically. It is widely used in operations research, scheduling, logistics, and other fields where optimization of discrete decisions is crucial.

Example 9: Solve the following LPP using branch and bound method.

Maximize, $Z = 7x_1 + 9x_2$

Subject to the constraints, $-x_1 + 3x_2 \leq 6; 7x_1 + x_2 \leq 35; x_2 \leq 7$; $x_1, x_2 \geq 0$ and $x_1, x_2 \in Z$

Solution: Step 1: Disregarding the integer constraints, the optimal solution to the given linear programming problem can be readily obtained as:

$x_1 = 9/2, x_2 = 7/2$ and Maximum $Z = 63$

Step 2: Since the solution is not an integer, let's select x_1 i.e., $x_1^* = 9/2$ as the variable with the largest fractional value.

Step 3: Taking the value of z as the initial upper bound, i.e., $z = 63$; the lower bound is found by rounding off the values of x_1 and x_2 to the nearest integers, i.e., $x_1 = 4$ and $x_2 = 3$. Thus, the lower bound is $z_1 = 55$.

Step 4: Since $[x^*{}_1] = [9/2] = 4;$ where, $[.]$ denote the greatest integer.

Sub-problem 1: Maximize $z = 7x_1 + 9x_2$

Subject to constraints, $-x_1 + 3x_2 \leq 6;\ 7x_1 + x_2 \leq 35;\ 0 \leq x_1 \leq 4$ *and* $0 \leq x_2 \leq 7;$

Sub-problem 2: Maximize $z = 7x_1 + 9x_2$

Subject to constraints, $-x_1 + 3x_2 \leq 6;\ 7x_1 + x_2 \leq 35;\ 0 \leq x_1 \geq 5$ *and* $0 \leq x_2 \leq 7;$

Step 5 Optimal solutions to the sub-problem are determined as follows

Sub-problem 1: Maximize $x_1 = 4, x_2 = 10/3$ and Maximum $z = 58$

Sub-problem 2: Maximize $x_1 = 5, x_2 = 0$ and Maximum $z = 35$

Since the solution to sub-problem 1 is not in integers, we further divide it into the following two sub-problems:

Sub-problem 3: Maximize $z = 7x_1 + 9x_2$

Subject to, $-x_1 + 3x_2 \leq 6;\ 7x_1 + x_2 \leq 35;\ 0 \leq x_1 \leq 4$ *and* $0 \leq x_2 \leq 3$

Sub-problem 4: Maximize $z = 7x_1 + 9x_2$

Subject to, $-x_1 + 3x_2 \leq 6;\ 7x_1 + x_2 \leq 35;\ 0 \leq x_1 \leq 4$ *and* $0 \leq x_2 \geq 4$

Step 6: The optimal solutions to the sub-problem 3 and 4 are:

Sub-problem 3: $x_1 = 4, x_2 = 3$ and maximum $z = 55$.

Sub-problem 3: No feasible solution.

Step 7: Among the recorded integer-valued solutions, the highest value of z is 55; therefore, the required optimal solution is:

$x_1 = 4, x_2 = 3$ and maximum $z = 55$.

The entire branch and bound procedure for the given problem is shown below:

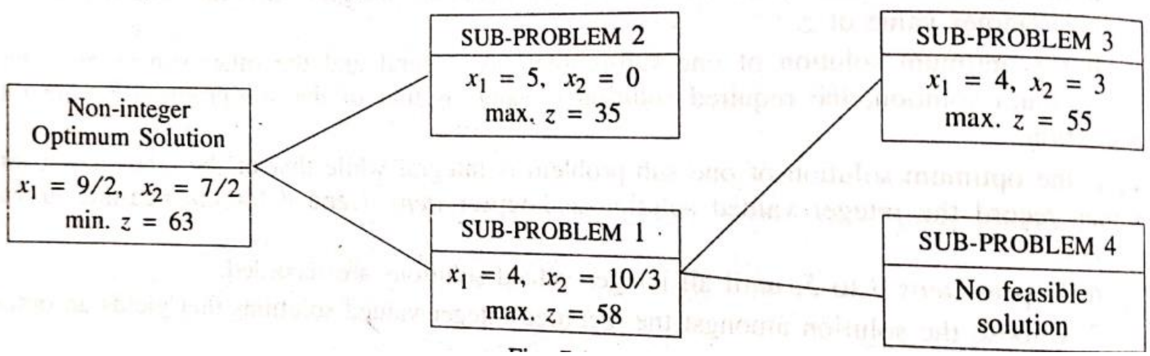


Figure 1

Check your progress

Problem 1: Obtain the optimum solution of the IPP

Maximize $Z = x_1 - 2x_2$ subject to the constraints:

$4x_1 + 2x_2 \le 15;\ x_1, x_2 \ge 0$ and $x_1, x_2 \in Z$

Answer: $x_1 = 3, x_2 = 0$**;** Maximum $z = 3$

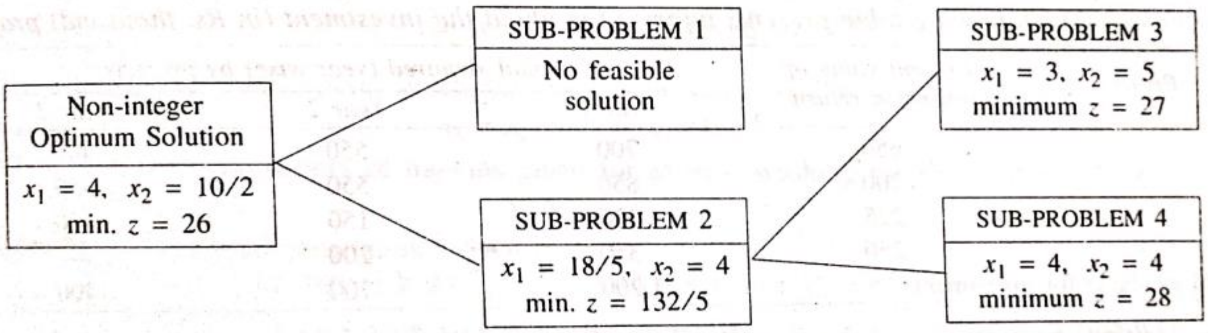
Problem 2: Solve the following LPP by using branch and bound method

Minimize $Z = 4x + 3y$ subject to the constraints:

$5x + 3y \ge 30;\ x \le 4, y \le 6, x, y \ge 0$ and $x_1, x_2 \in Z$

Answer: $x_1 = 3, y = 5$**;** Minimum $z = 27$

Hint: The complete branch and bound procedure for the given LPP is shown below:



7.9 *SUMMARY*

Integer programming is a powerful tool for solving discrete optimization problems, but it requires sophisticated methods to handle its computational challenges. Its applications are vast and impactful in various fields such as operations research, logistics, finance, and more. Integer programming is used in various fields, including:

- **Operations Research:** Scheduling, resource allocation, production planning.
- **Logistics:** Vehicle routing, supply chain optimization.
- **Finance:** Portfolio optimization, capital budgeting.
- **Telecommunications:** Network design, bandwidth allocation.

7.10 *GLOSSARY*

- Integer Programming
- Gomory's method
- Fractional cut method
- Branch and bound method

7.11 *REFERENCES*

- Mokhtar S. Bazaraa, John J. Jarvis and Hanif D. Sherali: *Linear Programming and Network Flows* (4th edition). John Wiley and Sons, 2010.
- Hamdy A. Taha: *Operations Research: An Introduction (10th edition). Pearson*, 2017.
- Paul R. Thie and Gerard E. Keough: *An Introduction to Linear Programming and Game Theory* (3rd edition), Wiley India Pvt. Ltd, 2014.
- Swarup, K., Gupta, P. K., & Mohan, M. (2017). An introduction to management science operation research. *Sultan Chand & Sons educational publishers, New Delhi*.

- OpenAI. (2024). *ChatGPT (August 2024 version) [Large language model]*. OpenAI. https://www.openai.com/chatgpt

7.12 *SUGGESTED READING*

- G. Hadley, *Linear Programming*, Narosa Publishing House, 2002.
- Frederick S. Hillierand Gerald J. Lieberman: *Introduction to Operations Research* (10th edition). McGraw-Hill Education, 2015.
- https://epgp.inflibnet.ac.in/Home/ViewSubject?catid=ZLCHeZEhCZ8yCri36nSF3A==

7.13 *TERMINAL QUESTION*

Short Answer Type Question:

1: Solve the following IPP

Maximize, $z = 3y$

Subject to the constraints,

$3x + 2y \leq 7;\ -x + y \leq 2;\ x, y \geq 0$; and $x, y \in Z$

2: Find the optimal solution of the following IPP

Maximize, $z = x - y$

Subject to the constraints,

$x + 2y \leq 4;\ 6x + 2y \leq 9;\ x, y \geq 0$; and $x, y \in Z$

3: Find the optimal solution of the following IPP

Maximize, $z = 2x + 3y$

Subject to the constraints,

$-3x + 7y \leq 14;\ 7x - 3y \leq 14;\ x, y \geq 0$; and $x, y \in Z$

4: Explain a method for solving an integer programming problem and then apply it to solve the problem.

Maximize, $z = 2x + 2y$

Subject to the constraints,

$5x + 3y \le 8;\ x + 2y \le 4;\ x, y \ge 0$; and $x, y \in Z^+$

5: Solve the following IPP:

Minimize, $z = 9x + 10y$

Subject to the constraints,

$x \le 9;\ y \le 8; 4x + 3y \ge 40;\ x, y \ge 0$; and $x, y \in Z^+$

6: Solve the following IPP:

Maximize, $z = 11x + 4y$

Subject to the constraints,

$-x + 2y \le 4;\ 5x + 2y \le 16; 2x - y \le 4$ and $x, y \in Z^+$

7: Solve the following IPP:

Maximize, $z = x + 2y$

Subject to the constraints,

$x + y \le 7;\ 2x \le 11; 2y \le 7; x, y \ge 0$ and $x, y \in Z^+$

8: Solve the given mixed-integer programming problem using Gomory's cutting plane method.

Maximize, $z = x + y$

Subject to the constraints,

$3x + 2y \le 5;\ y \le 2; x, y \ge 0$ and $x \in Z$

9: Solve the given mixed-integer programming problem using Gomory's cutting plane method.

Maximize, $z = x - 3y$

Subject to the constraints,

$x + y \leq 5; -2x + 4y \leq 11;\ x, y \geq 0$ and $y \in Z$

10: Solve the given mixed-integer programming problem using Gomory's cutting plane method.

Maximize, $z = 7x + 9y$

Subject to the constraints,

$-x + 3y \leq 6; 7x + y \leq 35;\ x, y \geq 0$ and $x \in Z$

Long answer type question:

1: Solve the following integer linear programming problems using the branch and bound method.

Maximize, $z = 2x + 3y$

Subject to the constraints,

$5x + 7y \leq 35; 4x + 9y \leq 36;\ x, y \geq 0$ and $x, y \in Z$

2: Solve the following integer linear programming problems using the branch and bound method.

Maximize, $z = 2x + 3y$

Subject to the constraints,

$x + y \leq 7; 0 \leq x \leq 5;\ 0 \leq y \leq 4,$ and $x, y \in Z$

3: Solve the following integer linear programming problems using the branch and bound method.

Maximize, $z = x + 2y$

Subject to the constraints,

$x+y\le 12; 4x+3y\le 14;\ x,y\ge 0$ and $x,y\in Z$

4: Solve the following integer linear programming problems using the branch and bound method.

Maximize, $z=2x+3y$

Subject to the constraints,

$6x+5y\le 25; x+3y\le 10;\ x,y\ge 0$ and $x,y\in Z$

5: Solve the following integer linear programming problems using the branch and bound method.

Maximize, $z=2x+y$

Subject to the constraints,

$x\le 3/2; y\le 3/2;\ x,y\ge 0$ and $x,y\in Z$

6: Solve the following integer linear programming problems using the branch and bound method.

Maximize, $3x+3y+13z$

Subject to the constraints, $-3x+6y+7z\le 8; 6x-3y+7z\le 8; 0\le x,y,z\le 5$ and $x,y,z\in Z$

Objective type question:

1: In integer programming, which type of decision variables are used?

A) Only continuous variables

B) Only binary variables

C) Only integer variables

D) Both integer and continuous variables

2: Which method is commonly used to solve integer programming problems?

A) Gradient Descent

B) Branch and Bound

C) Newton's Method

D) Least Squares

3: What type of optimization problem is an integer programming problem classified as?

- A) Linear
- B) Non-linear
- C) NP-hard
- D) Polynomial-time

4: In a mixed integer programming problem, some of the decision variables are:

A) Real numbers

B) Integer numbers

C) Binary numbers

D) Both A and B

5: Which of the following is NOT a method used for integer programming?

A) Simplex method

B) Cutting Plane method

C) Branch and Bound method

D) Genetic Algorithm

6: What is a Gomory cut?

A) A technique for dividing problems into subproblems

B) A type of cutting plane used to eliminate fractional solutions

C) A method for rounding solutions to the nearest integer

D) A type of constraint that ensures non-negativity

Fill in the blanks:

1: In integer programming, decision variables are restricted to __________ values.

2: The __________ method is a common algorithm used to solve integer programming problems by dividing them into smaller subproblems.

3: Mixed integer programming problems contain both __________ and __________ variables.

4: A __________ cut is a type of cutting plane used to eliminate fractional solutions in integer programming.

5: Integer programming problems are classified as __________, indicating they are computationally intensive and difficult to solve.

6: The __________ method iteratively adds linear constraints to exclude fractional solutions in integer programming.

7: When solving an integer programming problem, a feasible solution must satisfy all __________.

8: In binary integer programming, decision variables can take on the values __________ and __________.

9: __________ programming problems involve optimization where some or all decision variables must be integers.

10: The process of eliminating suboptimal branches in the Branch and Bound method is known as __________.

7.14 *ANSWERS*

Answer of short answer type question

Answer 1: $x_1 = 0, x_2 = 2,$; Maximum $z = 6$

2: $x = 1, y = 0$; Maximum $z = 1$

3: $x = 3, y = 3$; Maximum $z = 15$

4: $x = y = 1$; Maximum $z = 4$

5: $x = 9, y = 2$; Minimum $z = 101$

6: $x = 2, y = 3$; Maximum $z = 34$

7: $x = 4, y = 3$; Maximum $z = 10$

8: $x = 0, y = 2$; Maximum $z = 2$

9: $x = 17/2, y = 2$; Minimum $z = 5/2$

10: $x = 4, y = 10/3$; Maximum $z = 58$

Answer of long answer type question

Answer 1: $x_1 = 4, x_2 = 2,$ and Maximum $z = 14$

2: $x_1 = 3, x_2 = 4,$ and Maximum $z = 18$

3: $x_1 = 0, x_2 = 4,$ and Maximum $z = 8$

4: $x_1 = 1, x_2 = 3,$ and Maximum $z = 11$

5: $x_1 = 1, x_2 = 1,$ and Maximum $z = 3$

6: $x_1 = 0, x_2 = 0, x_3 = 1$ and Maximum $z = 13$

Answer of Multiple choice question

Answer:1:	D	**2:**	B	**3:**	C
4:	D	**5:**	A	**6:**	B

Answer of fill in the blank question

Answer:1:	Integer	**2:**	Branch and Bound	**3:**	Integer, continuous
4:	Gomory	**5:**	NP-hard	**6:**	Cutting Plane
7:	Constraints	**8:**	0, 1	**9:**	Integer
10:	Pruning				

UNIT-8: TRANSPORTATION

CONTENTS:

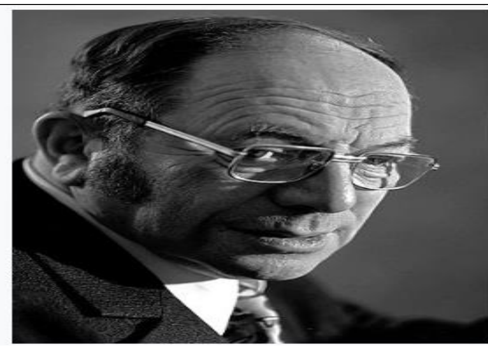
8.19 Answers

8.1 *INTRODUCTION*

The theory of transportation problems in operational research was first formulated by the Russian mathematician and economist **Leonid Kantorovich** in 1939. Kantorovich's work laid the foundation for linear programming and included the mathematical formulation of the transportation problem.

The theory of transportation problems in operational research has its roots in the pioneering work of Leonid Kantorovich, who laid the groundwork for linear programming and optimal resource allocation. This was further developed by other mathematicians and economists such as George B. Dantzig and F.L. Hitchcock, making the transportation problem a fundamental topic in the field of operational research.

Russian-born Leonid Vitalyevich Kantorovich (19 January 1912 – 7 April 1986) was a Soviet economist and mathematician who is renowned for his theory and invention of methods for the most efficient use of resources. He is recognised as linear programming's creator. Both the 1949 Stalin Prize and the 1975 Nobel Memorial Prize in Economic Sciences went to him.



Leonid Kantorovich

19 January 1912 - 7 April 1986

8.2 *OBJECTIVE*

After reading this unit learners will be able to

- Understand the basic concept of Transportation problem in linear programming.
- Visualized about the test of optimality.
- Learn about the concept of transhipment problem.

8.3 *TRANSPORTATION PROBLEM IN OPERATIONAL RESEARCH*

1. Definition: The **Transportation Problem** is a type of linear programming problem where the objective is to determine the most cost-effective way to transport goods from multiple sources (such as factories) to multiple destinations (such as warehouses) while meeting supply and demand constraints.

2. Components:

- **Supply Nodes:** Locations where goods are produced or stored, each with a specified supply quantity.
- **Demand Nodes:** Locations where goods are required, each with a specified demand quantity.
- **Cost Matrix:** The cost of transporting one unit of goods from each supply node to each demand node.

3. Formulation:

Objective Function: Minimize the total transportation cost. Minimize $z = \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij}$ In this context, c_{ij} represents the cost of transporting a single unit of goods from supply node i to demand node j, while x_{ij} denotes the quantity of goods being transported from node i to node j.

Constraints:

- **Supply Constraints:** Total goods shipped from each supply node must not exceed its supply. $\sum_{j=1}^{n} x_{ij} \le s_i \ \forall \ i = 1, 2, ..., m$
- **Demand Constraints:** Total goods received at each demand node must meet its demand. $\sum_{j=1}^{n} x_{ij} = d_i \ \forall \ i = 1, 2, ..., n$
- **Non-negativity Constraints:** $x_{ij} \ge 0 \ \ \forall i, j$

4. Methods for Solving:

Initial Solution Methods:

- **Northwest Corner Method:** Starts at the top-left corner of the cost matrix and allocates as much as possible, moving systematically.

- **Least Cost Method:** Selects the least cost cell first and allocates as much as possible, moving to the next least cost cell.
- **Vogel's Approximation Method (VAM):** Provides a more efficient starting solution by considering penalty costs for not using the second least cost route.
- **Optimal Solution Methods:**
- **Stepping Stone Method:** Evaluates the effect of moving goods from one allocation to another to reduce total cost.
- **MODI (Modified Distribution) Method:** Uses dual variables to find an optimal solution by calculating potential adjustments to the current solution.

5. Steps in Solving the Transportation Problem:

1. **Formulate the Problem:** Define the cost matrix, supply, and demand.
2. **Obtain an Initial Feasible Solution:** Use methods like the Northwest Corner, Least Cost, or VAM to get a starting point.
3. **Test for Optimality:** Apply the Stepping Stone or MODI method to check if the current solution can be improved.
4. **Iterate:** If the solution is not optimal, adjust the allocations to reduce costs and repeat the optimality test.
5. **Finalize the Solution:** Continue iterating until the optimal solution is reached.

6. Example: Given,

- Supply: Factory 1 (20 units), Factory 2 (30 units)
- Demand: Warehouse 1 (25 units), Warehouse 2 (25 units)
- Cost Matrix:

	Warehouse 1	Warehouse 2
Factory 1	8	6
Factory 2	7	9

Formulation:

Minimize $z = 8x_{11} + 8 = 6x_{12} + 7x_{21} + 9x_{22}$

Subject to:

$x_{11} + x_{12} \le 20$; $x_{21} + x_{22} \le 30$; $x_{11} + x_{21} = 25$; $x_{12} + x_{22} = 25$; $x_{ij} \ge 0$

Solution Approach:

- **Step 1:** Use the Northwest Corner Method to find an initial feasible solution.
- **Step 2:** Apply the MODI method to check and improve the solution.
- **Step 3:** Iterate until the optimal solution is found.

7. Applications:

- **Logistics and Supply Chain Management:** Optimizing transportation costs for goods distribution.
- **Manufacturing:** Allocating raw materials to production plants cost-effectively.
- **Military Operations:** Efficient allocation of supplies to different bases.
- **Public Transportation:** Planning routes and schedules to minimize costs.

8.4 *EXISTENCE OF SOLUTION IN TRANSPORTING PROBLEM*

The existence of a feasible solution in the transportation problem is guaranteed if the problem is balanced, meaning that the total supply equals the total demand. If the problem is unbalanced, it can be adjusted by adding dummy supply or demand nodes to ensure balance. Once balanced, various methods can be used to find and optimize a feasible transportation plan that meets all supply and demand constraints while minimizing the total transportation cost.

Theorem 1: Necessary and Sufficient Condition for the Existence of a Feasible Solution to the General Transportation Problem

OR

Statement: A necessary and sufficient condition for the existence of a feasible solution to the general transportation problem is that the total supply equals the total demand.

Proof: Definitions and Setup

- **Supply Nodes:** We have m supply nodes, each with a supply s_i for $i = 1, 2, ..., m$.
- **Demand Nodes:** We have n demand nodes, each with a demand d_j for $j = 1, 2, ..., n$
- **Transportation Cost:** The cost to transport one unit from supply node i to demand node j is denoted as c_{ij}.
- **Decision Variables:** x_{ij} represents the quantity transported from supply node i to demand node j.

The goal is to minimize the total transportation cost:

Minimize $z = \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij}$

subject to: $\sum_{j=1}^{n} x_{ij} = s_i \ \forall \ i = 1, 2, ..., m$

$$\sum_{j=1}^{n} x_{ij} = d_j \ \forall \ j = 1, 2, ..., n$$

$$x_{ij} \geq 0 \ \forall i, j$$

1. **Necessary Condition**

To show: If a feasible solution exists, then the total supply must equal the total demand.

Proof:

- Suppose a feasible solution exists. This implies there exists a set of values $x_{ij} \geq 0$ that satisfy all the constraints.
- For the supply constraints, we have:

$$\sum_{j=1}^{n} x_{ij} = s_i \ \forall \ i$$

Summing these equations over all supply nodes:

$$\sum_{i=1}^{n} \sum_{j=1}^{n} x_{ij} = \sum_{i=1}^{m} s_i$$

- For the demand constraints, we have: $\sum_{i=1}^{m} x_{ij} = d_j \ \forall \ j$

Summing these equations over all demand nodes:

$$\sum_{j=1}^{m} \sum_{i=1}^{m} x_{ij} = \sum_{j=1}^{n} d_j$$

- By comparing the two results: $\sum_{i=1}^{m} s_i = \sum_{j=1}^{n} d_j$

This equality must hold, indicating that if a feasible solution exists, then the total supply must equal the total demand.

3. **Sufficient Condition**

To show: If the total supply equals the total demand, then a feasible solution exists.

Proof:

- Suppose the total supply equals the total demand: $\sum_{i=1}^{m} s_i = \sum_{j=1}^{n} d_j$
- We can balance the problem by setting up a feasible solution:

Initial Solution: Use methods like the Northwest Corner Method, Least Cost Method, or Vogel's Approximation Method to allocate the quantities x_{ij} such that all supply and demand constraints are satisfied.

Verification: Ensure that each supply constraint $\sum_{j=1}^{n} x_{ij} = s_i$ is satisfied.

- Ensure that each demand constraint $\sum_{j=1}^{n} x_{ij} = d_j$ is satisfied.
- Since the total supply equals total demand, it is possible to find non-negative values x_{ij} that satisfy all constraints.

Example of Balancing:

1. **Balanced Case:**
    - Given:
        1. Supply: $s_1 = 20$, $s_2 = 30$
        2. Demand: $d_1 = 25, d_1 = 20, d_3 = 5$
    - Total Supply = Total Demand = 50
    - Use initial solution methods to find a feasible distribution of goods.
2. **Unbalanced Case:**

- If total supply ≠ total demand, introduce dummy nodes to balance the problem:
- **Excess Supply:** Add a dummy demand node with demand equal to the excess supply.
- **Excess Demand:** Add a dummy supply node with supply equal to the excess demand.
- Ensure the new total supply equals the total demand and proceed with the initial solution methods.

Theorem 2: In a general transportation problem with m supply nodes and n demand nodes, the number of basic variables in any feasible solution is $m+n-1$.

Proof: Definitions and Setup:

- **Supply Nodes:** S_i with supply s_i for $i=1,2,...,m$
- **Demand Nodes:** D_j with demand d_j for $j=1,2,...,n$
- **Decision Variables:** x_{ij} represents the quantity transported from supply node i to demand node j
- **Basic Variables:** The variables that correspond to the non-zero entries in the solution and satisfy all constraints

Proof:

1. **Constraints and Equations:**

- **Supply Constraints:** For each supply node S_i : $\sum_{j=1}^{n} x_{ij} = s_i \; \forall \, i = 1 \, to \, m$
- **Demand Constraints:** For each demand node D_j : $\sum_{i=1}^{m} x_{ij} = d_j \; \forall \, j = 1 \, to \, n$
- **Total Supply Equals Total Demand:** This ensures a balanced transportation problem: $\sum_{i=1}^{m} s_j = \sum_{i=1}^{m} d_j \forall \, j = 1 \, to \, n$

2. **Total Number of Variables:**

- The total number of decision variables x_{ij} is $m \times n$

3. **System of Linear Equations:**

- There are m supply constraints and n demand constraints.
- This gives us a system of $m+n$ linear equations.

4. **Redundancy in Constraints:**

- The equation $\sum_{i=1}^{m} s_i = \sum_{j=1}^{n} d_j \forall \, j = 1 \, to \, n$ is implied by the supply and demand constraints and introduces a dependency among them.

- Thus, only $m+n-1$ of these constraints are linearly independent.

5. Feasibility and Basic Feasible Solution:

- In linear programming, a basic feasible solution is found at the intersection of $m+n-1$ hyperplanes (constraints).
- For a system of linear equations with $m+n-1$ independent constraints, there can be at most $m+n-1$ basic variables.
- The basic variables are those that form a basis for the solution space, meaning they are sufficient to describe the solution entirely while satisfying the constraints.

6. Non-Basic Variables:

- The remaining variables, which are $(m \times n)-(m+n-1)$ in number, are non-basic and are set to zero in a basic feasible solution.

7. Example for Clarity:

- Consider a transportation problem with 3 supply nodes $(m=3)$ and 4 demand nodes $(n=4)$.
- According to the theorem, the number of basic variables should be 3+4−1=6.
- These 6 basic variables will satisfy the 6 independent constraints derived from the original 7 constraints (3 supply + 4 demand - 1 dependent equation).

The proof shows that for a feasible solution to the general transportation problem, the number of basic variables must indeed be $m+n-1$. This ensures that the solution satisfies all the supply and demand constraints and provides a complete description of the transportation plan. This theorem is fundamental to understanding and solving the transportation problem using methods like the simplex method, where the concept of basic and non-basic variables is crucial.

8.5 *DUALITY IN TRANSPORTATION PROBLEM*

Duality is a fundamental concept in linear programming that provides a way to analyze and solve optimization problems. In the context of the transportation problem, duality helps in understanding the relationship between the original problem (primal problem) and its dual problem. The primal problem typically focuses on minimizing the transportation cost, while the dual problem deals with maximizing the potential or shadow prices associated with supply and demand constraints.

1. Primal Problem (Transportation Problem): The primal transportation problem can be formulated as follows:

- **Objective Function:** Minimize the total transportation cost.

$$z = \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij}$$

Where, c_{ij} is the cost to transport one unit from supply node *i* to demand node *j*, and x_{ij} is the quantity transported.

- **Supply Constraints:**

$$z = \sum_{j=1}^{n} x_{ij} = s_i \;\; \forall \; i = 1, 2, ..., m$$

Where, s_i is the supply at node *i*.

- **Demand Constraints:**

$$\sum_{j=1}^{n} x_{ij} = d_j \;\; \forall \;\; j = 1, 2, ..., n$$

Where, d_j is the demand at node *j*.

- **Non-negativity Constraints:**

$$x_{ij} \geq 0 \,\forall\, i, j$$

2. Dual Problem

The dual of the transportation problem can be derived by associating a dual variable with each constraint in the primal problem:

- Let u_i be the dual variable associated with the supply constraint at node *i*.
- Let v_j be the dual variable associated with the demand constraint at node *j*.

The dual problem can be formulated as:

- **Objective Function:** Maximize the total potential associated with the supplies and demands.

Maximize $w = \sum_{j=1}^{n} u_i s_i + \sum_{j=1}^{n} v_j d_j$

- **Constraints:**

  $u_i + v_j \le c_{ij} \forall\ i, j$

where c_{ij} is the transportation cost from supply node *i* to demand node *j*.

- **Non-negativity Constraints:**

  u_i and v_j are unrestricted in sign

3. Complementary Slackness Conditions

The complementary slackness conditions link the primal and dual solutions and provide a way to check optimality. For the transportation problem, the conditions are:

- If $x_{ij} > 0$ in the primal solution, then $u_i + v_j = c_{ij}$ in the dual solution.
- If $x_{ij} = 0$ in the primal solution, then $u_i + v_j \le c_{ij}$ in the dual solution..

These conditions ensure that if both the primal and dual solutions satisfy them, then both solutions are optimal.

4. Economic Interpretation

- **Primal Solution Interpretation:** The primal variables x_{ij} represent the quantities transported from supply nodes to demand nodes to minimize the total transportation cost.
- **Dual Solution Interpretation:** The dual variables $u_i = 0$ and $v_j = 0$ represent the shadow prices or potentials associated with the supply and demand constraints. They provide insights into the cost structure and the marginal values of increasing supply or demand at specific nodes.

5. Example

Consider a simple transportation problem with 2 supply nodes and 3 demand nodes. The cost matrix, supply, and demand are given by:

	D_1	D_2	D_3	
S_1	4	6	8	Supply
S_2	5	7	9	Supply
Demand	Demand	Demand	Demand	

- **Supply:** $s_1 = 20, s_2 = 30,$
- **Demand:** $d_1 = 15, d_2 = 25, d_3 = 10,$

Primal Problem:

Minimize $z = 4x_{11} + 6x_{12} + 8x_{13} + 5x_{21} + 7x_{22} + 9x_{23}$

Subject to, $x_{11} + x_{12} + x_{13} = 20$; $x_{21} + 7x_{22} + 9x_{23} = 30$; $x_{11} + x_{21} = 15$; $x_{12} + x_{22} = 25$

$x_{13} + x_{23} = 10$; $x_{ij} \geq 0$

Dual Problem:

Minimize $w = 20u_1 + 30u_2 + 15v_1 + 25v_2 + 10v_3$

Subject to, $u_1 + v_1 \leq 4$; $u_1 + v_2 \leq 6$; $u_1 + v_3 \leq 8$; $u_2 + v_1 \leq 5$; $u_2 + v_2 \leq 7$; $u_2 + v_3 \leq 9$

u_i, v_j unrestricted in sign

Duality in the transportation problem provides a framework to analyze and solve the problem from two perspectives: minimizing transportation costs and maximizing potentials associated with supply and demand. The dual problem complements the primal problem, and the solutions to both are interrelated through the complementary slackness conditions. This dual perspective helps in gaining deeper insights into the cost structure and the marginal values in the transportation network.

8.6 *TRANSPORTATION TABLE*

The transportation table is a tabular representation used to solve the transportation problem. It helps visualize the allocation of resources from supply nodes to demand nodes while minimizing transportation costs. The table format aids in systematically applying solution methods such as the Northwest Corner Method, Least Cost Method, and Vogel's Approximation Method, and it is also useful in checking for feasibility and optimality.

Structure of the Transportation Table

The transportation table consists of the following components:

1. **Rows and Columns:** Represent supply nodes and demand nodes, respectively.
2. **Cells:** Represent the possible routes from supply to demand nodes, where each cell contains the transportation cost and the quantity to be transported.
3. **Supply and Demand Margins:** Indicate the available supply at each source and the required demand at each destination.

Example 1: Consider a transportation problem with 3 supply nodes and 4 demand nodes. The table structure would look like this:

	D_1	D_2	D_3	D_4	Supply
S_1	c_{11}	c_{12}	c_{13}	c_{14}	s_1
S_2	c_{21}	c_{22}	c_{23}	c_{24}	s_2
S_3	c_{31}	c_{32}	c_{33}	c_{34}	s_3
Demand	d_1	d_2	d_3	d_4	

Components:

- c_{ij} : Cost to transport one unit from supply node i to demand node j .
- s_i : Supply available at supply node i .
- d_j : Demand required at demand node j .
- x_{ij} : Quantity to be transported from supply node i to demand node j .

Example 2: Given the following transportation problem:

Supply Nodes:

- S_1 with supply $s_1 = 20$
- S_2 with supply $s_2 = 30$
- S_3 with supply $s_3 = 25$

Demand Nodes:

- D_1 with demand $d_1 = 10$
- D_2 with demand $d_2 = 35$
- D_3 with demand $d_3 = 20$
- D_4 with demand $d_4 = 10$

Transportation Costs:

	D_1	D_2	D_3	D_4	Supply
S_1	4	6	8	10	20
S_2	5	8	7	9	30
S_3	9	4	3	2	25
Demand	10	35	20	10	

8.7 *INITIAL SOLUTION METHODS*

Northwest Corner Method:

- Start at the northwest (top-left) corner of the table and allocate as much as possible to the cell until either the supply or demand is exhausted.
- Move to the next cell to the right or downward as appropriate and repeat the process.

Least Cost Method:

- Identify the cell with the lowest cost and allocate as much as possible to that cell.
- Adjust the supply and demand, then repeat the process with the remaining cells.

Vogel's Approximation Method:

- Calculate penalties for not using the cheapest routes and use these penalties to determine the allocation.

Optimality Check

- Once an initial feasible solution is obtained using one of the methods above, check for optimality using methods such as the **Stepping Stone Method** or the **MODI Method** (Modified Distribution Method).
- Adjust the allocations to minimize the total transportation cost further.

Example 2: Calculation using Northwest Corner Method

1: Start with S_1 and D_1:

- Allocate $x_1 = \min(s_1, d_1) = \min(20,10) = 10$
- Update supply and demand: $s_1 = 20 - 10 = 10, d_1 = 10 - 10 = 0$

2: Move to S_1 and D_2:

- Allocate $x_{12} = \min(s_1, d_2) = \min(10, 35) = 10$
- Update supply and demand: $s_1 = 10 - 10 = 0, d_2 = 35 - 10 = 25$

3: Move to S_2 and D_2:

- Allocate $x_{22} = \min(s_2, d_2) = \min(30, 25) = 25$
- Update supply and demand: $s_2 = 30 - 25 = 5, d_2 = 25 - 25 = 0$

4: Move to S_2 and D_3:

- Allocate $x_{23} = \min(s_2, d_3) = \min(5, 20) = 5$
- Update supply and demand: $s_2 = 5 - 5 = 0, d_3 = 20 - 5 = 15$

5: Move to S_3 and D_3:

- Allocate $x_{33} = \min(s_3, d_3) = \min(25,15) = 15$
- Update supply and demand: $s_3 = 25 - 15 = 10, d_3 = 15 - 15 = 0$

6: Move to S_3 and D_4:

- Allocate $x_{34} = \min(s_3, d_4) = \min(10,10) = 10$
- Update supply and demand: $s_3 = 10 - 10 = 0, d_4 = 10 - 10 = 0$

So, the resulting allocation table:

	D_1	D_2	D_3	D_4	Supply
S_1	10	10	0	0	20
S_2	0	25	5	0	30
S_3	0	0	15	10	25
Demand	10	35	20	10	

This allocation satisfies all supply and demand constraints, and now further steps would involve checking the optimality and potentially redistributing the allocations to minimize costs further.

8.8 *NORTH-WEST CORNER METHOD*

The North-West Corner Method is a technique used to find an initial feasible solution to the transportation problem. This method starts at the top-left (north-west) cell of the transportation table and allocates as much as possible to this cell, then moves to adjacent cells to continue the allocation process.

Steps of the North-West Corner Method:

1. **Start at the North-West Corner:** Begin at the top-left cell of the transportation table.
2. **Allocate the Minimum Supply or Demand:** Allocate the minimum of the supply available or the demand required in the current cell.
3. **Adjust Supply and Demand:** After allocation, adjust the supply and demand for the respective row and column.
4. **Move to the Next Cell:** Move to the next cell to the right if the demand is met or move down if the supply is exhausted. Repeat the process until all supplies and demands are met.

Example 3: Let's examine a transportation problem characterized by the given supply, demand, and cost matrix:

	D_1	D_2	D_3	Supply
S_1	20	10	0	30
S_2	0	40	0	40
S_3	0	0	20	20
Demand	20	50	20	

Solution Steps:

Start at the North-West Corner (Cell (S_1, D_1)):

- Allocate $x_1 = \min(s_1, d_1) = \min(30,20) = 20$
- Update supply and demand: $s_1 = 30 - 20 = 10, d_1 = 20 - 20 = 0$

Move to the Next Cell (Cell (S_1, D_2)):

- Allocate $x_{12} = \min(s_1, d_2) = \min(10,50) = 10$
- Update supply and demand: $s_1 = 10 - 10 = 0, d_2 = 50 - 10 = 40$

Move to the Next Cell (Cell (S_2, D_2)):

- Allocate $x_{22} = \min(s_2, d_2) = \min(40,40) = 40$
- Update supply and demand: $s_2 = 40 - 40 = 0, d_2 = 40 - 40 = 0$

Move to the Next Cell (Cell ((S_3, D_3)):

- Allocate $x_{33} = \min(s_3, d_3) = \min(20,20) = 20$
- Update supply and demand: $s_3 = 20 - 20 = 0, d_3 = 20 - 20 = 0$

Resulting allocation table:

	D_1	D_2	D_3	Supply
S_1	20	10	0	30
S_2	0	40	0	40
S_3	0	0	20	20
Demand	20	50	20	

Transportation cost calculation:

The total transportation cost can be calculated by summing the product of allocated quantities and their respective costs:

$$Total\cos t = (20\times 2)+(10\times 3)+(40\times 4)+(20\times 8) = 40+30+160+160 = 390$$

The North-West Corner Method provides an initial feasible solution to the transportation problem. The method is straightforward but does not guarantee an optimal solution. To find the optimal solution, other methods such as the Stepping Stone Method or the MODI Method (Modified Distribution Method) can be applied to this initial solution.

8.9 *VOGEL'S APPROXIMATION METHOD (OR PENALTY METHOD)*

Vogel's Approximation Method (VAM) is an efficient technique to find an initial feasible solution for the transportation problem. It typically results in a better starting solution compared to other methods like the Northwest Corner Method or the Least Cost Method.

Steps of Vogel's Approximation Method:

1. **Calculate Penalties:** For each row and each column, compute the penalty by finding the difference between the smallest and the second smallest cost in that row or column.
2. **Select the Highest Penalty:** Identify the row or column with the highest penalty. This indicates the potential cost increase for not choosing the cheapest route.
3. **Allocate as Much as Possible:**

- Within the selected row or column, allocate as much as possible to the cell with the lowest cost.

- Adjust the supply and demand for the respective row and column.

4. Update the Table:

- If a row or column is satisfied (supply or demand becomes zero), exclude that row or column from further consideration.
- Recompute penalties for the remaining rows and columns.

5. Repeat:

- Repeat the process until all supplies and demands are satisfied.

Example: Consider the following transportation problem:

- **Supply Nodes:**
  - S_1 with supply $s_1 = 30$
  - S_2 with supply $s_2 = 40$
  - S_3 with supply $s_3 = 20$
- **Demand Nodes:**
  - D_1 with demand $d_1 = 20$
  - D_2 with demand $d_2 = 50$
  - D_3 with demand $d_3 = 20$
- **Transportation Costs:**

	D_1	D_2	D_3	Supply
S_1	2	3	1	30
S_2	5	4	8	40
S_3	5	6	8	20
Demand	20	50	20	

Solution Steps Using VAM:

1. Calculate Initial Penalties:

- Row Penalties:
  - $S_1 : 3-1=2$
  - $S_2 : 5-4=1$
  - $S_3 : 6-5=1$
- Column Penalties:
  - $D_1 : 5-2=3$
  - $D_2 : 4-3=1$
  - $D_3 : 8-1=7$

2. **Select the Highest Penalty (Column D_3, Penalty = 7):**

➢ Allocate to the cell with the lowest cost in column D_3, which is S_1D_3:

▪ Allocate $x_{13} = \min(s_1, d_3) = \min(30,20) = 20$

▪ Update supply and demand: $s_1 = 30 - 20 = 10, d_3 = 20 - 20 = 0$

3. **Update the Table:**

	D_1	D_2	Removed	Supply
S_1	2	3	–	10
S_2	5	4	–	40
S_3	5	6	–	20
Demand	20	50	0	

4. **Recalculate Penalties:**

- Row Penalties:
  - $S_1 : 3 - 2 = 1$
  - $S_2 : 5 - 4 = 1$
  - $S_3 : 6 - 5 = 1$
- Column Penalties:
  - $D_1 : 5 - 2 = 3$
  - $D_2 : 4 - 3 = 1$

5. **Select the Highest Penalty (Column D_1, Penalty = 3):**

- Allocate to the cell with the lowest cost in column D_1, which is S_1D_1:
  - Allocate $x_{11} = \min(s_1, d_1) = \min(10, 20) = 10$
  - Update supply and demand: $s_1 = 10 - 10 = 0, d_1 = 20 - 10 = 10$

6. **Update the Table:**

	D_1	D_2	Supply
S_2	5	4	40
S_3	5	6	20
Demand	10	50	

7. **Recalculate Penalties:**

- Row Penalties:

- $S_2 : 5 - 4 = 1$
- $S_3 : 6 - 5 = 1$

- Column Penalties:
  - $D_1 : 5 - 5 = 0$
  - $D_1 : 4 - 6 = -2$ (we ignore negative values for penalties)

8. Select the Highest Penalty (Row S_2, Penalty = 1):

- Allocate to the cell with the lowest cost in row S_2, which is S_2D_2:
  - Allocate $x_{22} = \min(s_2, d_2) = \min(40,50) = 40$
  - Update supply and demand: $s_2 = 40 - 40 = 0, d_2 = 50 - 40 = 10$

9. Update the Table:

	D_1	D_2	Supply
S_3	5	6	20
Demand	10	10	

10. Allocate the Remaining Supply and Demand:

- Allocate $x_{31} = \min(s_3, d_1) = \min(20,10) = 10$
- Allocate $x_{32} = \min(s_3, d_2) = \min(10,10) = 10$

Resulting Allocation Table:

	D_1	D_2	D_3	**Supply**
S_1	10	0	20	30
S_2	0	40	0	40
S_3	10	10	0	20
Demand	20	50	20	

Transportation Cost Calculation:

The total transportation cost can be calculated by summing the product of allocated quantities and their respective costs:

Total Cost= $(10\times 2)+(20\times 1)+(40\times 4)+(10\times 5)+(10\times 6)$

$$=20+20+160+50+60=310$$

Vogel's Approximation Method (VAM) provides a systematic approach to find an initial feasible solution to the transportation problem, often resulting in a solution closer to the optimal compared to simpler methods. Further optimization techniques can be applied to this initial solution to achieve the optimal transportation cost.

8.10 *TESTS FOR OPTIMALITY*

Once an initial feasible solution is found for the transportation problem, the next step is to test if this solution is optimal. The most common methods to check for optimality are the **Stepping Stone Method** and the **Modified Distribution Method (MODI)**.

Steps for Testing Optimality Using the MODI Method:

1. **Calculate the Initial Basic Feasible Solution (BFS):**
- Use any initial solution method like the Northwest Corner Method, Least Cost Method, or Vogel's Approximation Method.

2. **Compute the Potentials (u and v):**
- Assign potential u_i to each row and potential v_j to each column such that for each occupied cell (i, j): $u_i + v_j = c_{ij}$
- Set $u_1 = 0$ (you can set any u or v to 0, but u_1 is a common choice).

3. **Calculate Opportunity Costs:**
- For each unoccupied cell (i, j), calculate the opportunity cost:

$$\Delta_{ij} = c_{ij} - (u_i + v_j)$$

- If all $\Delta_{ij} \geq 0$, the current solution is optimal.

4. **Check for Negative Opportunity Costs:**
- If any $\Delta_{ij} < 0$, , the current solution is not optimal. Identify the cell with the most negative opportunity cost for improvement.

5. **Improve the Solution:**
- Form a closed loop (cycle) starting and ending at the selected cell with negative opportunity cost.
- Allocate as much as possible to the positive cells of the loop and adjust the allocations alternately subtracting and adding the same amount along the loop.

- Update the basic feasible solution and repeat the process until no negative opportunity cost remains.

Example: Consider the following initial feasible solution obtained by VAM or any other method:

	D_1	D_2	D_3	Supply
S_1	10 (2)	0 (3)	20 (1)	30
S_2	0 (5)	40 (4)	0 (8)	40
S_3	10 (5)	10 (6)	0 (8)	20
Demand	20	50	20	

The numbers in parentheses are the costs c_{ij}

1. **Calculate Potentials u and v:**
   - Start with $u_1 = 0$
   - For $x_{11} = 10 : v_1 = 2u_1 + v_1 = 2 \Rightarrow v_1 = 2$
   - For $x_{13} = 20 : u_1 + v_3 = 1 \Rightarrow v_3 = 1$
   - For $x_{22} = 40 : u_2 + v_2 = 4 \Rightarrow u_2 = 4 - v_2$
   - For $x_{31} = 10 : u_3 + v_1 = 5 \Rightarrow u_3 = 5 - v_1$
   - For $x_{32} = 10 : u_3 + v_2 = 6 \Rightarrow u_3 = 6 - v_2$
2. **Solving these equations:**
   - $v_1 = 2$
   - $v_3 = 1$
   - $u_3 = 3$
   - From $x_{22} : 4 = u_2 + v_2$
   - From $x_{32} : 3 + v_2 = 6 \Rightarrow v_2 = 3$
   - $u_2 = 1$
3. **Calculate Opportunity Costs Δ_{ij}:**
   - $\Delta_{12} = 3-(0+3) = 0$
   - $\Delta_{21} = 5-(1+2) = 2$
   - $\Delta_{23} = 8-(1+1) = 6$

- $\Delta_{33} = 8-(3+1) = 4$

Since all $\Delta_{ij} \geq 0$, the current solution is optimal.

By applying the MODI method, we can verify the optimality of a given initial feasible solution to the transportation problem. If the solution is not optimal, we can iteratively improve it until an optimal solution is reached.

8.11 *DEGENERACY IN THE TRANSPORATAION PROBLEM*

Degeneracy in the transportation problem occurs when the number of basic variables in a basic feasible solution is less than $m+n-1$, where mmm is the number of rows (sources) and n is the number of columns (destinations) in the transportation table.

Why Degeneracy Occurs:

Degeneracy can happen if:

1. There are too many zero supplies or demands.
2. The initial allocation method leads to fewer allocations than $m+n-1$.

Handling Degeneracy:

To handle degeneracy, we introduce artificial allocations (usually very small quantities, denoted by ϵ\epsilon) in such a way that the number of allocations becomes $m+n-1$. This helps in maintaining the feasibility of the solution and allows the application of optimization methods.

Example of degeneracy and its resolution:

Consider a transportation problem with the following supply, demand, and cost matrix:

	D_1	D_2	D_3	Supply
S_1	2	3	1	30
S_2	5	4	8	40
S_3	5	6	8	20
Demand	20	50	20	

Step-by-Step Solution Using North-West Corner Method:

1. **Start at the North-West Corner (Cell S_1D_1):**
- Allocate $x_{11} = \min(30, 20) = 20$
- Update supply and demand: $s_1 = 30 - 20 = 10, d_1 = 20 - 20 = 0$
2. **Move to the Next Cell (Cell S_1D_2):**
- Allocate $x_{12} = \min(10, 50) = 10$
- Update supply and demand: $s_1 = 10 - 10 = 0, d_2 = 50 - 10 = 40$
3. **Move to the Next Cell (Cell S_2D_2):**
- Allocate $x_{22} = \min(40, 40) = 40$
- Update supply and demand: $s_2 = 40 - 40 = 0, d_2 = 40 - 40 = 0$
4. **Move to the Next Cell (Cell S_3D_3):**
- Allocate $x_{33} = \min(20, 20) = 20$
- Update supply and demand: $s_3 = 20 - 20 = 0, d_3 = 20 - 20 = 0$

Resulting Allocation Table:

	D_1	D_2	D_3	Supply
S_1	20	10	0	30
S_2	0	40	0	40
S_3	0	0	20	20
Demand	20	50	20	

Checking for Degeneracy:

The number of basic variables is 4 (cells with allocations: $x_{11}, x_{12}, x_{22}, x_{33}$), but $m + n - 1 = 3 + 3 - 1 = 5$. Since 4 < 5, this solution is degenerate.

Resolving Degeneracy:

To resolve degeneracy, we introduce an artificial allocation of a very small quantity ϵ\epsilon in an empty cell to make the number of basic variables equal to $m + n - 1$.

We can place ϵ in any empty cell that does not form a closed loop with the existing allocations. Let's place ϵ in cell x_{23}:

Adjusted Allocation Table:

	D_1	D_2	D_3	Supply
S_1	20	10	0	30
S_2	0	40	ϵ	40
S_3	0	0	20	20
Demand	20	50	20	

Now, the number of basic variables is 5 (cells with allocations: $x_{11}, x_{12}, x_{22}, x_{33}, x_{23}$), which equals $m+n-1$. This resolves the degeneracy.

Degeneracy in the transportation problem occurs when there are fewer basic variables than required. It is resolved by introducing artificial allocations (usually ϵ) to ensure the number of allocations is equal to $m+n-1$. This allows the application of optimization techniques to find the optimal solution.

8.12 *BALANCED AND UNBALANCED TRANPORTATION PROBLEM*

Balanced Transportation Problem: A transportation problem is considered balanced when the total supply equals the total demand. In such cases, it is possible to allocate all supplies to meet all demands without any surplus or shortage.

Unbalanced Transportation Problem: A transportation problem is unbalanced when the total supply does not equal the total demand. This scenario requires adjustments to convert it into a balanced problem before applying standard solution methods.

Converting Unbalanced Transportation Problem to Balanced

To balance an unbalanced transportation problem, you can introduce a dummy source or dummy destination:

- **If total supply > total demand**: Add a dummy destination with a demand equal to the excess supply.
- **If total demand > total supply**: Add a dummy source with a supply equal to the excess demand.

The cost associated with transportation to or from a dummy location is usually set to zero or some large value to indicate that no real transportation is happening.

Example: (Balanced and Unbalanced Problems): Consider the following balanced transportation problem where total supply equals total demand

	D_1	D_2	D_3	Supply
S_1	2	3	1	30
S_2	5	4	8	40
S_3	5	6	8	20
Demand	20	50	20	

Total supply = 30 + 40 + 20 = 90 Total demand = 20 + 50 + 20 = 90

Since the total supply equals the total demand, this problem is balanced.

Unbalanced Transportation Problem Example: Consider the following unbalanced transportation problem where total supply does not equal total demand:

	D_1	D_2	D_3	Supply
S_1	2	3	1	30
S_2	5	4	8	50
S_3	5	6	8	20
Demand	20	50	40	

Total supply = 30 + 50 + 20 = 100 Total demand = 20 + 50 + 40 = 110

Since the total supply (100) is less than the total demand (110), this problem is unbalanced.

Balancing the Unbalanced Problem: To balance the problem, add a dummy source with a supply equal to the excess demand (110 - 100 = 10).

	D_1	D_2	D_3	Dummy	Supply
S_1	2	3	1	0	30
S_2	5	4	8	0	50
S_3	5	6	8	0	20
Dummy	0	0	0	0	10
Demand	20	50	40	10	

Now, the total supply equals the total demand (100 + 10 = 110), making the problem balanced.

Solving the Balanced Problem

Once the problem is balanced, you can use any transportation problem-solving method (e.g., Northwest Corner Method, Least Cost Method, Vogel's Approximation Method) to find an initial feasible solution and then optimize it.

Balanced transportation problems have equal total supply and demand, making them straightforward to solve using standard methods. Unbalanced problems require adjustments by adding dummy sources or destinations to balance them before applying solution techniques. Balancing ensures that the problem is well-posed and can be tackled using classical optimization methods.

8.13 *TRANSHIPMENT PROBLEM*

The transshipment problem is a generalization of the transportation problem. It involves not only direct transportation from sources to destinations but also includes intermediate transfer points called transshipment nodes. The goal is to determine the most cost-effective way to transport goods from sources to destinations, possibly via these transshipment nodes.

Characteristics of Transshipment Problems:

1. **Multiple Nodes**: Includes sources, destinations, and transshipment nodes.
2. **Intermediate Transfers**: Goods can be transferred through one or more transshipment points before reaching their final destinations.
3. **Increased Complexity**: Due to the additional nodes and possible routes, the problem becomes more complex compared to a basic transportation problem.

Formulating a Transshipment Problem

To formulate a transshipment problem, follow these steps:

1. **Define the Nodes**:
- **Sources**: Nodes where goods originate.
- **Transshipment Points**: Intermediate nodes where goods can be transferred.
- **Destinations**: Nodes where goods are required.
2. **Define the Decision Variables**:
- Let x_{ij} represent the amount of goods transported from node *i* to node *j*.
3. **Objective Function**:
- Minimize the total transportation cost:

  Minimize $Z = \sum_i \sum_j c_{ij} x_{ij}$, where $Z = c_{ij}$ is the cost of transporting goods from node *i* to node *j*

4. **Constraints**:
- **Supply Constraints**: For each source node, the total outgoing shipment should not exceed its supply. $Z = \sum_j x_{ij} \le s_i$ for each source *i*

  where s_i is the supply at node *i*.

- **Demand Constraints**: For each destination node, the total incoming shipment should meet its demand. $\sum_i x_{ij} \ge d_j$ for each destination j

  where d_j is the demand at node *j*.

- **Flow Balance Constraints**: For each transshipment node, the total incoming shipment should equal the total outgoing shipment.

  $\sum_i x_{ij} = \sum_k x_{jk}$ for each transhipment node *j*

- **Non-Negativity Constraints**: $x_{ij} \ge 0 \ \forall \ i, j$

Example of a Transshipment Problem

Consider a problem with the following nodes:

- **Sources**: S_1 and S_2
- **Transshipment Points**: T_1 and T_2

- **Destinations**: D_1 and D_2

Data

From \ To	T_1	T_2	D_1	D_2	Supply
S_1	2	3	-	-	50
S_2	4	1	-	-	40
T_1	-	-	5	2	0
T_2	-	-	3	4	0

To \ From	S_1	S_2	T_1	T_2	Demand
D_1	-	-	5	3	60
D_2	-	-	2	4	30

Formulation

1. **Decision Variables**:

- x_{S_1,T_1}, x_{S_1,T_2}, x_{S_2,T_1}, x_{S_2,T_2}
- x_{T_1,D_1}, x_{T_1,D_2}, x_{T_2,D_1}, x_{T_2,D_2}

2. **Objective Function**:

- Minimize the total transportation cost:

Minimize $Z = 2x_{S_1,T_1} + 3x_{S_1,T_2} + 4x_{S_2,T_1} + 1x_{S_2,T_2} + 5x_{T_1,D_1} + 2x_{T_1,D_2} + 3x_{T_2,D_1} + 4x_{T_2,D_2}$

3. **Constraints**:

- Supply constraints for sources: $x_{S_1,T_1} + x_{S_1,T_2} \leq 50$; $x_{S_2,T_1} + x_{S_2,T_2} \leq 40$
- Demand constraints for destinations: $x_{T_1,D_1} + x_{T_2,D_1} \geq 60$; $x_{T_1,D_2} + x_{T_2,D_2} \geq 30$
- Flow balance constraints for transshipment points:

$$x_{S_1,T_1} + x_{S_2,T_1} = x_{T_1,D_1} + x_{T_1,D_2}$$

$$x_{S_1,T_2} + x_{S_2,T_2} = x_{T_2,D_1} + x_{T_2,D_2}$$

Non-negativity constraints: $x_{ij} \geq 0 \ \forall i, j$

Note: You can solve this transshipment problem using linear programming methods. Tools like Excel Solver, MATLAB, or optimization software such as Gurobi, CPLEX, or LINGO are suitable for solving such problems.

The transshipment problem adds complexity to the basic transportation problem by introducing intermediate transfer points. It can be formulated as a linear programming problem with additional constraints for transshipment points. By carefully defining the decision variables, objective function, and constraints, the problem can be solved using standard optimization techniques. This type of problem is commonly encountered in logistics and supply chain management, where goods often pass through multiple stages before reaching their final destinations.

Check your progress

Problem 1: Obtain the initial basic feasible problem of the transportation problem using Vogel's approximation method

	D	E	F	G	Available
A	11	13	17	14	250
B	16	18	14	10	300
C	21	24	13	10	400
Demand	200	225	275	250	

Answer: $z = 200\times 11 + 50\times 13 + 175\times 18 + 125\times 10 + 275\times 13 + 250\times 10 = 12{,}075$

Problem 2: Obtain the initial basic feasible problem of the transportation problem using North-west corner method

	D	E	F	G	Available
A	11	13	17	14	250
B	16	18	14	10	300
C	21	24	13	10	400
Requirement	200	225	275	250	

Answer: $z = 200\times 11 + 50\times 13 + 175\times 18 + 125\times 14 + 150\times 13 + 250\times 10 = 12{,}200$

Problem 3: Solve the following Transportation problem

From	A	B	C	Available
I	50	30	220	1
II	90	45	170	3
III	250	200	50	4
Requirement	4	2	2	

Answer: $x_{11} = 1, x_{21} = 3, x_{32} = 2, x_{33} = 2;$ Maximum total cost = 820

8.14 *SUMMARY*

Transportation problems are widely used in logistics, supply chain management, and distribution planning. They help in optimizing costs in scenarios involving shipping goods, distributing products, and allocating resources.

In summary, the transportation problem provides a structured approach to minimize transportation costs while satisfying supply and demand constraints. It involves finding an initial feasible solution and then iterating to reach the optimal solution using linear programming techniques.

8.15 *GLOSSARY*

- Transportation
- Test of optimality
- Balanced and Unbalanced transportation method
- Transhipment problem
- North West Corner method
- Vogel's approximation

8.16 *REFERENCES*

- Mokhtar S. Bazaraa, John J. Jarvis and Hanif D. Sherali: *Linear Programming and Network Flows* (4th edition). John Wiley and Sons, 2010.
- Hamdy A. Taha: *Operations Research: An Introduction (10th edition). Pearson*, 2017.

- Paul R. Thie and Gerard E. Keough: *An Introduction to Linear Programming and Game Theory* (3rd edition), Wiley India Pvt. Ltd, 2014.
- Swarup, K., Gupta, P. K., & Mohan, M. (2017). An introduction to management science operation research. *Sultan Chand & Sons educational publishers, New Delhi*.
- OpenAI. (2024). *ChatGPT (August 2024 version) [Large language model]*. OpenAI. https://www.openai.com/chatgpt

8.17 *SUGGESTED READING*

- G. Hadley, *Linear Programming*, Narosa Publishing House, 2002.
- Frederick S. Hillierand Gerald J. Lieberman: *Introduction to Operations Research* (10th edition). McGraw-Hill Education, 2015.
- https://epgp.inflibnet.ac.in/Home/ViewSubject?catid=ZLCHeZEhCZ8yCri36nSF3A==

8.18 *TERMINAL QUESTION*

Short Answer Type Question:

1: Obtain the initial basic feasible problem of the transportation problem using Vogel's approximation method

	D_1	D_2	D_3	D_4	Supply
S_1	20	25	28	31	200
S_2	32	28	32	41	180
S_3	18	35	24	32	110

2: Obtain the initial basic feasible problem of the transportation problem using North-west corner method

From	D1	D2	D3	D4	Availablity
O1	5	3	6	2	19
O2	4	7	9	1	37

O3	3	4	7	5	34
Requirement	16	18	31	25	

3: Obtain the initial basic feasible problem of the transportation problem using Vogel's approximation method

From	Destination				Availablity
	1	2	3	4	
1	20	22	17	4	120
2	24	37	9	7	70
3	32	37	20	15	50
Requirement	60	40	30	110	240

4: Solve the following Transportation problem

From	To			Availablity
	A	B	C	
I	6	8	4	14
II	4	9	8	12
III	1	2	6	5
Requirement	6	10	15	

5: Define Transportation problem?

6: Describe an unbalanced transportation problem?

7: Describe balanced transportation problem? Discuss about its applications?

8: Describe transshipment problem?

Long answer type question:

1: Solve the following Transportation problem

source	Destination				Availablity
	1	2	3	4	
I	21	16	25	13	11
II	17	18	14	23	13
III	32	27	18	41	19
Requirement	6	10	12	15	43

2: Solve the following Transportation problem

origin	Destination				Availablity
	D_1	D_2	D_3	D_4	
O_1	1	2	1	4	30
O_2	3	3	2	1	50
O_3	4	2	5	9	20
Requirement	20	40	30	10	

3: Solve the following Transportation problem

origin	Destination				Availablity
	D_1	D_2	D_3	D_4	
O_1	23	27	16	18	30
O_2	12	17	20	51	40
O_3	22	28	12	32	53
Requirement	22	35	25	41	123

The cell entries are unit transportation problem

4: Determine the initial solution for the given transportation problem using Vogel's Approximation Method. Then, proceed to find the optimal solution.

origin	Destination				Supply
	D_1	D_2	D_3	D_4	
S_1	3	7	6	4	5
S_2	2	4	3	2	2
S_3	4	3	8	5	3
Demand	3	3	2	2	

Objective type question:

1: What is the primary objective of a transportation problem?

A) Maximize total supply

B) Minimize total cost

C) Maximize total demand

D) Minimize total transportation time

2: Which method is often used to find an initial feasible solution to a transportation problem?

A) Simplex Method

B) Vogel's Approximation Method

C) Branch and Bound Method

D) Dual Simplex Method

3: In a balanced transportation problem, the total supply is:

A) Greater than total demand

B) Less than total demand

C) Equal to total demand

D) Unrelated to total demand

4: If the transportation problem is unbalanced, which of the following steps is necessary?

A) Introduce a dummy row or column

B) Apply the Simplex method directly

C) Reduce the supply or demand to balance

D) Ignore the excess supply or demand

5: Which of the following is not a method to solve the transportation problem?

A) North-West Corner Method

B) Least Cost Method

C) Vogel's Approximation Method

D) Hungarian Method

6: In the Vogel's Approximation Method (VAM), penalties are calculated based on:

A) The difference between the largest and smallest costs in each row and column

B) The difference between supply and demand

C) The sum of the costs in each row and column

D) The total cost of the initial solution

7: In a transportation problem, the number of basic variables in a feasible solution is:

A) Equal to the number of sources plus the number of destinations

B) Equal to the number of sources minus the number of destinations

C) One less than the sum of the number of sources and destinations

D) The product of the number of sources and destinations

True and False:

1: In a balanced transportation problem, a feasible solution always exists.

2: Degeneracy occurs in a transportation problem when the number of occupied cells is less than m+n-1, where m is the number of rows and n is the number of columns.

Fill in the blanks:

1: In a balanced transportation problem, the total supply is equal to the total ________.

2: To solve an unbalanced transportation problem, we introduce a ________ row or column to balance the problem.

3: The primary objective of the transportation problem is to minimize the total ________ of transporting goods.

4: The ________ Method is one of the techniques used to find an initial feasible solution for a transportation problem.

5: In Vogel's Approximation Method, penalties are calculated as the difference between the smallest and second smallest ________ in each row and column.

6: A transportation problem is said to be ________ when the number of basic variables in a feasible solution is less than m+n-1.

7: The ________ Method is used to check the optimality of a feasible solution in a transportation problem.

8: In the Least Cost Method, the allocation is made in the cell with the ________ cost in the cost matrix.

9: The ________ method calculates the opportunity cost for unallocated cells to determine whether the current solution is optimal.

10: In a transportation problem, each row in the cost matrix represents a ________ and each column represents a ________.

8.19 *ANSWERS*

Answer of short answer type question

Answer 1: Total cost $= 40\times20+40\times28+120\times31+40\times28+140\times32+110\times18+50\times0=13220$

2: $x_{11}=16, x_{12}=3, x_{22}=15, x_{23}=22, x_{33}=9$ and $x_{34}=25$

3: $x_{12}=40, x_{14}=80, x_{21}=10, x_{23}=30, x_{24}=30, x_{31}=50$ and $x_{34}=25$

4: $x_{13}=14, x_{21}=6, x_{22}=5, x_{23}=1, x_{32}=5$ Minimum total cost = 143

Answer of long answer type question

Answer 1: $x_{14}=11, x_{21}=6, x_{22}=3, x_{24}=4, x_{32}=7, x_{33}=12$; Minimum total cost = 796

2: $x_{11}=20, x_{13}=10, x_{22}=20, x_{23}=20, x_{24}=10, x_{32}=20$; Minimum total cost = 180

3: **Answer:** $x_{14}=30, x_{21}=5, x_{22}=35, x_{31}=17, x_{33}=25, x_{34}=11$; Minimum total cost = 2221

4: The optimal solution is $x_{11}=3, x_{14}=2, x_{23}=2, x_{24}=\in_1, x_{32}=3, x_{34}=\in_2$;

The transportation cost associated with the optimum schedule is:

$3\times3+2\times4+2\times3+2\times\in_1+3\times3+5\times\in_2;\ 32+2\in_1+5\in_2\ \textit{as}\ \in_1\to0\ \textit{and}\ \in_2\to0$

Answer of Multiple choice question

Answer 1: B **2:** B **3:** C **4:** A

5: D **6:** A **7:** C

True and False

Answer 1: True **2:** True

Answer of fill in the blank question

Answer 1: demand **2:** dummy **3:** cost

4: North-West Corner **5:** costs **6:** degenerate

7: MODI (Modified Distribution) **8:** lowest

9: stepping-stone **10:** source, destination

BLOCK- III

NON-LINEAR PROGRAMMING

UNIT-9: NON-LINEAR PROGRAMMING

CONTENTS:

9.1 *INTRODUCTION*

The Karush–Kuhn–Tucker (KKT) conditions, also referred to as the Kuhn–Tucker conditions in mathematical optimisation, are first derivative tests (also called first-order necessary conditions) that determine whether a nonlinear programming solution is optimal, given that certain regularity conditions are met. The KKT technique of nonlinear programming permits inequality constraints, whereas the Lagrange multiplier method only allows equality constraints. The restricted maximisation (minimisation) issue is reformulated as a Lagrange function, whose optimal point is a global minimum (maximum) over the multipliers and a global maximum (minimum) over the domain of the choice variables, much like the Lagrange technique. The saddle-point theorem is another name for the Karush–Kuhn–Tucker theorem.

The technique of solving an optimisation problem in mathematics where part of the constraints are nonlinear equalities or the objective function is nonlinear is known as nonlinear programming, or NLP. Calculating the extrema (maxima, minima, or stationary points) of an objective function over a collection of unknown real variables, conditional on the fulfilment of a system of equalities and inequalities collectively referred to as constraints is the definition of an optimisation problem. It is the branch of mathematical optimisation that handles nonlinear issues.

9.2 *OBJECTIVE*

After reading this unit learners will be able to

- Understand the basic concept of Non-linear programming.
- Visualized the neceaasry and sufficient condition for general NLPP.
- Learn about the concept of Kuhn-Tucker conditions for general NLPP.

9.3 *NON-LINEAR PROGRAMMING IN LPP*

Non-linear programming (NLP) extends the concepts of Linear Programming (LPP) to situations where the objective function or constraints are non-linear. In Linear Programming, all relationships objective function and constraints are linear. However, many real-world problems involve non-linear relationships, necessitating the use of non-linear programming.

Key Differences Between LPP and NLP

1. **Objective Function**:
- **LPP**: The objective function is a linear function of the decision variables. For example, $Z = c_1x_1 + c_2x_2$.
- **NLP**: The objective function is non-linear. For example, $Z = x_1^2 + 3x_2$ or $Z = e^{x_1} + \log(x_2)$

2. **Constraints**:
- **LPP**: Constraints are linear inequalities or equalities. For example, $a_1x_1 + a_2x_2 \le b$
- **NLP**: Constraints can be non-linear. For example, $x_1^2 + x_2^2 \le 1$ or $\sqrt{x_1} + x_2 \ge 4$

3. **Feasibility Region**:
- **LPP**: The feasible region, formed by linear constraints, is a convex polyhedron.
   1. **NLP**: The feasible region can be more complex and non-convex, depending on the nature of the non-linear constraints.

4. **Solution Techniques**:

- **LPP**: Problems can be solved using the Simplex method, the Interior Point method, or other linear programming algorithms.
- **NLP**: Requires more complex algorithms such as Gradient Descent, Newton's Method, Sequential Quadratic Programming (SQP), and others.

Example 1: Consider the following non-linear programming problem:

Objective Function:

Maximize $Z = x_1 x_2$

Subject to:

$x_1^2 + x_2^2 \le 1$ (Non-linear constraint)

$x_1 + x_2 \le 1$ (Linear constraint)

$x_1 + x_2 \ge 0$

In this example, the objective function $Z = x_1 x_2$ is non-linear, and one of the constraints $x_1^2 + x_2^2 \le 1$ is also non-linear, making this an NLP problem.

Solving NLP Problems in the Context of LPP

In some cases, non-linear programming can be approached by linearization techniques, particularly if the non-linearity is mild or can be approximated by linear functions within a certain range. However, if the non-linearities are significant, specialized NLP techniques must be used.

Linearization Techniques:

- **Piecewise Linear Approximation**: The non-linear function is approximated by several linear segments.
- **Taylor Series Expansion**: Non-linear functions are approximated by their first-order (or higher) Taylor series around a certain point, making them linear.

These methods can sometimes transform a non-linear programming problem into a linear programming problem (or a series of linear problems), allowing the use of LPP methods.

Applications of NLP in LPP Contexts

NLP techniques are applied in situations where linear models are insufficient to capture the complexity of real-world problems, such as:

- **Economics**: Maximizing utility functions, which are often non-linear.
- **Engineering**: Designing systems with non-linear behavior, such as stress-strain relationships in materials.
- **Finance**: Portfolio optimization with non-linear risk measures.
- **Operations Research**: Resource allocation and production planning with non-linear cost functions.

Non-linear programming builds on the foundation of linear programming but expands it to more complex scenarios where relationships between variables are non-linear. While LPP provides efficient solutions to linear problems, NLP is essential for dealing with the non-linearities inherent in many real-world problems. Understanding both frameworks allows for more comprehensive modeling and solution strategies in optimization tasks.

9.4 *GENERAL NON-LINEAR PROGRAMMING PROBLEM*

A general Non-Linear Programming Problem (NLPP) can be expressed in a standard form, where the objective is to either maximize or minimize a non-linear objective function subject to a set of constraints. These constraints can be both equality and inequality constraints, and they may also be non-linear.

General Form of a Non-Linear Programming Problem

Objective Function:

- **Minimize** or **Maximize** $f(x_1, x_2, ..., x_n)$

  Where $f(x_1, x_2, ..., x_n)$ is a non-linear function of the decision variables $x_1, x_2, ..., x_n$.

Subject to Constraints:

- **Inequality Constraints**:

  $g_i(x_1, x_2, ..., x_n) \le 0$, for $i = 1, 2, ..., m$

  Where $g_i(x_1, x_2, ..., x_n)$ are non-linear functions representing the constraints.

- **Equality Constraints**:

  $h_j(x_1, x_2, ..., x_n) = 0$, for $i = 1, 2, ..., p$

  Where $h_j(x_1, x_2, ..., x_n)$ are non-linear functions representing the equality constraints.

- **Bounds on Variables**:

  $x_k^{(lower)} \leq x_k \leq x_k^{(upper)}$, for $k = 1, 2, ..., n$

  Where $x_k^{(lower)}$ and $x_k^{(upper)}$ are the lower and upper bounds on the decision variable x_k.

Example of a General Non-Linear Programming Problem

Consider a problem where a company wants to minimize its operational cost, which is a non-linear function of the amount of resources x_1 and x_2 used.

Objective Function:

- **Minimize** $f(x_1, x_2) = x_1^2 + 2x_2^2 + 3x_1x_2$

Subject to:

- **Inequality Constraint**: $g_1(x_1, x_2) = x_1^2 + x_2^2 - 10 \leq 0$
- **Equality Constraint**: $h_1(x_1, x_2) = x_1^2 + x_2^2 - 25 = 0$
- **Bounds**: $0 \leq x_1 \leq 5, 0 \leq x_2 \leq 5$

Interpretation

- **Objective Function**: The function $f(x_1, x_2) = x_1^2 + x_2^2 + 3x_1x_2$ represents the cost the company wants to minimize. It's non-linear because of the squared terms and the product term $3x_1x_2$.
- **Inequality Constraint**: The constraint $x_1^2 + x_2^2 - 10 \leq 0$ might represent a resource limitation or a performance criterion that must not be exceeded.
- **Equality Constraint**: The equality $x_1^2 + x_2^2 = 25$ could represent a constraint where the total combined use of x_1 and x_2 must be exactly 25 units, perhaps due to a strict budget or resource allocation.
- **Bounds**: The variables x_1 and x_2 are bounded between 0 and 5, which might represent physical limits on the amount of resources that can be utilized.

This general form of an NLPP illustrates the flexibility of non-linear programming in modeling complex real-world optimization problems. It allows for the inclusion of non-linear relationships, making it suitable for more complex scenarios than linear programming. Solving such problems typically requires specialized algorithms such as Gradient Descent, Newton's Method, or more advanced techniques like Sequential Quadratic Programming (SQP)

9.5 *CONSTRAINED OPTIMIZATION WITH EQUALITY CONSTRAINTS*

Constrained optimization with equality constraints is a type of optimization problem where the objective is to maximize or minimize a function subject to one or more equality constraints. These constraints specify that certain functions of the decision variables must be exactly equal to a given value.

Problem Formulation

Suppose we want to minimize or maximize an objective function $f(x_1, x_2, x_3, ..., x_n)$ subject to equality constraints. The general form of such a problem is:

Minimize (or Maximize) $f(x_1, x_2, x_3, ..., x_n)$

Subject to:

$h_j(x_1, x_2, x_3, ..., x_n) = 0 \ for \ j = 1, 2, ..., p$

Where:

- $f(x_1, x_2, x_3, ..., x_n)$ is the objective function.
- $h_j(x_1, x_2, x_3, ..., x_n) = 0$ are the equality constraints.
- $x_1, x_2, x_3, ..., x_n$ are the decision variables.

Method of Solution: Lagrange Multipliers

The most common method for solving constrained optimization problems with equality constraints is the **method of Lagrange multipliers**. This technique converts the constrained problem into an unconstrained problem by introducing additional variables called Lagrange multipliers.

Steps to Solve Using Lagrange Multipliers

1. **Formulate the Lagrangian**: Construct the Lagrangian function L by incorporating the objective function and the equality constraints:

$$L(x_1, x_2, x_3, ..., x_n, \lambda_1, \lambda_2, ..., \lambda_p) = f(x_1, x_2, x_3, ..., x_n) + \sum_{j=1}^{p} \lambda_j h_j(x_1, x_2, ..., x_n)$$

Where λ_j are the Lagrange multipliers associated with the equality constraints $h_j(x_1, x_2, ..., x_n) = 0$.

2. **Take Partial Derivatives**: Find the partial derivatives of the Lagrangian with respect to each decision variable x_i and each Lagrange multiplier λ_j, and set them equal to zero:

$$\frac{\partial L}{\partial x_i} = 0, \text{ for all } i = 1, 2, ..., n$$

$$\frac{\partial L}{\partial \lambda_j} = 0, \text{ for all } j = 1, 2, ..., p$$

3. **Solve the System of Equations**: The partial derivatives yield a system of equations. Solve this system simultaneously to find the values of the decision variables $x_1, x_2, ..., x_n$ and the Lagrange multipliers $\lambda_1, \lambda_2, ..., \lambda_p$.
4. **Verify Solutions**: Once a solution is found, verify that it satisfies all the original equality constraints and check the nature (maximum or minimum) of the objective function at that point.

Example 2: Minimize the function $f(x, y) = x^2 + y^2$ subject to the constraint $x + y = 1$

Solution: Using Lagrange Multipliers:

1. **Formulate the Lagrangian**: $L(x, y, \lambda) = x^2 + y^2 + \lambda(x + y - 1)$
2. **Take Partial Derivatives**: $\frac{\partial L}{\partial x} = 2x + \lambda = 0$ … (1)

$$\frac{\partial L}{\partial y} = 2y + \lambda = 0 \quad \ldots (2)$$

$$\frac{\partial L}{\partial \lambda} = 2y + \lambda = 0 \quad \ldots (3)$$

3. **Solve the System of Equations**: From equations (1) and (2):

$2x + \lambda = 0$ and $2y + \lambda = 0 \Rightarrow x = y$

Substituting $x = y$ into the constraint equation $x + y = 1$:

$$x + x = 1 \Rightarrow 2x = 1 \Rightarrow x = \frac{1}{2}, y = \frac{1}{2},$$

4. **Verify the Solution**: The solution $x = \frac{1}{2}, y = \frac{1}{2}$ satisfies the constraints $x + y = 1$. The objective function value at this point is $f\left(\frac{1}{2}, \frac{1}{2}\right) = \left(\frac{1}{2}\right)^2 + \left(\frac{1}{2}\right)^2 = \frac{1}{4} + \frac{1}{4} = \frac{1}{2}$

Constrained optimization with equality constraints using Lagrange multipliers is a powerful method for finding the optimum of a function subject to strict equality conditions. It transforms the problem into a system of equations that can be solved to find the optimal solution, incorporating both the objective function and the constraints.

9.6 *NECESSARY AND SUFFICIENT CONDITION FOR A GENERAL NLPP*

The necessary conditions for a general Non-Linear Programming Problem (NLPP) using the Lagrange multiplier method are derived when the problem involves only equality constraints. These conditions help determine the potential optimal points of the problem.

Problem Formulation: Consider the following general NLPP:

Minimize (or Maximize): $f(x_1, x_2, ..., x_n)$

Subject to:

$$h_j(x_1, x_2, ..., x_n) = 0, \; j = 1, 2, ..., p$$

Where:

- $f(x_1, x_2, ..., x_n)$ is the objective function.
- $h_j(x_1, x_2, ..., x_n) = 0$ are the equality constraints.
- $x = (x_1, x_2, ..., x_n)$ are the decision variables.

Lagrangian Function

To incorporate the constraints into the optimization problem, we define the **Lagrangian function**:

$$L(x_1, x_2, ..., x_n, \lambda_1, \lambda_2, ..., \lambda_p) = f(x_1, x_2, ..., x_n) + \sum_{j=1}^{p} \lambda_j h_j(x_1, x_2, ..., x_n)$$

Here:

- λ_j are the Lagrange multipliers associated with each constraint $h_j(x) = 0$.

Necessary Conditions (First-Order Conditions)

For a point $x^* = (x_1^*, x_2^*, ..., x_n^*)$ to be a candidate for optimality, the following conditions must be satisfied:

1. **Stationarity**:

The partial derivatives of the Lagrangian function with respect to each decision variable x_i must equal zero at the point x^*:

$$\frac{\partial L}{\partial x_i} = \frac{\partial f(x^*)}{\partial x_i} + \sum_{j=1}^{p} \lambda_j^* \frac{\partial h_j(x^*)}{\partial x_i} = 0, \text{ for all } i = 1, 2, ..., n$$

This condition ensures that the gradient of the objective function $f(x)$ is a linear combination of the gradients of the constraints at the point x^*.

2. **Feasibility**:

- The point x^* must satisfy the original constraints of the problem:

$$h_j(x^*) = 0, \text{ for all } j = 1, 2, ..., p$$

Interpretation of the Conditions

- The **stationarity condition** implies that at the optimal point, the objective function cannot be further improved in any direction without violating at least one of the constraints.
- The **feasibility condition** ensures that the point satisfies all the equality constraints.

Example 3: Suppose you want to minimize $f(x, y) = x^2 + y^2$ subject to the constraint $x + y - 1 = 0$.

1. **Lagrangian Function**: $L(x, y, \lambda) = x^2 + y^2 + \lambda(x + y - 1)$
2. **Stationarity**:

- Partial derivative with respect to *x*: $\frac{\partial L}{\partial x} = 2x + \lambda = 0 \Rightarrow \lambda = -2x$
- Partial derivative with respect to *y*: $\frac{\partial L}{\partial y} = 2y + \lambda = 0 \Rightarrow \lambda = -2y$

Equating λ from both equations gives $2x = 2y$, so $x = y$

3. **Feasibility**:

- Substitute $x = y$ into the constraint $x + y - 1 = 0$:

$$x + x = 1 \Rightarrow 2x = 1 \Rightarrow x = \frac{1}{2}, y = \frac{1}{2}$$

4. The candidate solution $(x^*, y^*) = \left(\frac{1}{2}, \frac{1}{2}\right)$ satisfies both the stationarity and feasibility conditions.

The necessary conditions for optimality using the Lagrangian method involve finding the stationary points of the Lagrangian function and ensuring that the solution satisfies the equality constraints. These conditions are the first step in identifying potential optimal solutions to non-linear programming problems.

Sufficient Conditions (Second-Order Conditions)

Let x^* be a point that satisfies the first-order necessary conditions (stationarity and feasibility). The second-order sufficient conditions can be stated as follows:

1. **Hessian of the Lagrangian**:

- Define the Hessian matrix $H(x, \lambda)$ of the Lagrangian function with respect to the decision variables *x*:

$$H(x, \lambda) = \nabla_{xx}^2 L(x, \lambda)$$

where $\nabla_{xx}^2 L(x, \lambda)$ is the matrix of second-order partial derivatives of L with respect to the decision variables *x*.

2. **Constraint Qualification**:

- The gradients of the active constraints $\nabla h_j(x^*)$ must be linearly independent at x^*.

3. Sufficient Condition for a Local Minimum:

- The Hessian matrix of the Lagrangian evaluated at x^* and the corresponding Lagrange multipliers λ^* must be **positive definite** on the tangent space to the constraints:

$$z^T H(x^*, \lambda^*) z > 0 \text{ for all } z \neq 0 \text{ such that } \nabla h_j(x^*)^T z = 0 \text{ for all } j$$

This condition ensures that the point x^* is a local minimum.

4. Sufficient Condition for a Local Maximum:

- The Hessian matrix of the Lagrangian evaluated at x^* and the corresponding Lagrange multipliers λ^* must be **negative definite** on the tangent space to the constraints:

$$z^T H(x^*, \lambda^*) z < 0 \text{ for all } z \neq 0 \text{ such that } \nabla h_j(x^*)^T z = 0 \text{ for all } j$$

This condition ensures that the point x^* is a local maximum.

- **Positive Definite Hessian**: If the Hessian matrix of the Lagrangian is positive definite on the tangent space to the constraints at x^*, then x^* is a **local minimum**.
- **Negative Definite Hessian**: If the Hessian matrix of the Lagrangian is negative definite on the tangent space to the constraints at x^*, then x^* is a **local maximum**.

These second-order sufficient conditions, together with the first-order necessary conditions, help in verifying whether a candidate solution is optimal for the NLPP.

Example 3: Derive the necessary conditions for solving the non-linear programming problem.

Minimize $z = kx^{-1}y^{-2}$ subject to the constraint

$$x^2 + y^2 - a^2 = 0 \text{ with } x \geq 0, y \geq 0;$$

And hence find the value of z.

Solution: The Lagrange's function is,

$$L(x, y, \lambda) = kx^{-1}y^{-2} + \lambda(x^2 + y^2 - a^2);$$

where $f(x, y) = kx^{-1}y^{-2}$ and $h(x, y) = g(x, y) - C = (x^2 + y^2) - a^2$

The necessary conditions for the minimum of $f(x, y)$ gives

$$\frac{\partial L}{\partial x} = 0 \Rightarrow -kx^{-2}y^{-2} + 2x\lambda = 0$$

$$\frac{\partial L}{\partial y} = 0 \Rightarrow -2kx^{-1}y^{-3} + 2\lambda y = 0$$

And $\frac{\partial L}{\partial \lambda} = 0 \Rightarrow x^2 + y^2 - a^2 = 0$

From the first two equations, we get

$2\lambda = kx^{-3}y^{-2} = 2kx^{-1}y^{-4}$. This yields $x = y/\sqrt{2}$

Using this value of x in the third equation,

$y = a\sqrt{2/3}$ and therefore $x = \frac{1}{\sqrt{2}} \times a\sqrt{\frac{2}{3}} = \frac{a}{\sqrt{3}}$

Minimum, $z = k \times (a\sqrt{3})^{-1}(a\sqrt{2/3})^{-2} = 3\sqrt{3}k/2a^3$.

Example 4: Derive the necessary conditions for the non-linear programming problem.

Maximize, $z = x_1^{\ 2} + 3x_2^{\ 2} + 5x_3^{\ 2}$ subject to the constraint:

$x_1 + x_2 + 3x_2 = 2; 5x_1 + 2x_2 + x_3 = 5; x_1, x_2, x_3 \geq 0$

Answer: Since, we have $x = (x_1, x_2, x_3)$, $f(X) = x_1^{\ 2} + 3x_2^{\ 2} + 5x_3^{\ 2}$, $g^1(x) = x_1 + x_2 + 3x_2 = 5$,

$g^2(x) = 5x_1 + 2x_2 + x_3$ and $c_1 = 2, c_2 = 5$. Defining, $h^i(x) = g^i(x) - c_i, i = 1, 2$, we have the constraints: $h^i(x) = 0, i = 1, 2$.

For necessary conditions for maximizing $f(x)$, we construct the Lagrangian function

$L(x, \lambda) = f(x) - \lambda_1 h^1(x) - \lambda_2 h^2(x), \lambda = (\lambda_1, \lambda_2)$

This yields the following necessary conditions:

$$\frac{\partial L}{\partial x_1} = 2x_1 - \lambda_1 - 5\lambda_2 = 0 \qquad \frac{\partial L}{\partial x_2} = 6x_2 - \lambda_1 - 2\lambda_2 = 0$$

$$\frac{\partial L}{\partial x_3} = 10x_3 - 3\lambda_1 - \lambda_2 = 0 \qquad \frac{\partial L}{\partial \lambda_1} = -(x_1 + x_2 + 3x_3 - 2) = 0$$

$$\frac{\partial L}{\partial \lambda_2} = -(5x_1 + 2x_2 + x_3 - 5) = 0$$

Example 5: Derive the necessary conditions for the non-linear programming problem.

Minimize, $z = 2x_1^{\ 2} - 24x_1 + 2x_2^{\ 2} - 8x_2 + 2x_3^{\ 2} - 12x_3 + 200$

subject to the constraint: $x_1 + x_2 + x_3 = 11; x_1, x_2, x_3 \geq 0$

Answer: We formulate the Lagrangian function as

$$L(x_1, x_2, \lambda) = 2x_1^{\ 2} - 24x_1 + 2x_2^{\ 2} - 8x_2 + 2x_3^{\ 2} - 12x_3 + 200 - \lambda(x_1 + x_2 + x_3 - 11)$$

For necessary conditions for stationary points are

$$\frac{\partial L}{\partial x_1} = 4x_2 - 24 - \lambda = 0 \qquad \frac{\partial L}{\partial x_3} = 4x_3 - 8 - \lambda = 0$$

$$\frac{\partial L}{\partial x_3} = 4x_3 - 12 - \lambda = 0 \qquad \frac{\partial L}{\partial \lambda} = -(x_1 + x_2 + x_3 - 11) = 0$$

The solution of the simultaneous equation yields the stationary point

$x_0 = (x_1, x_2, x_3) = (6, 2, 3); \lambda = 0$.

It is sufficient for both minors Δ_3 and Δ_4 to be negative in order for the stationary point to be a minimum. Currently, we have

$$\Delta_3 = \begin{vmatrix} 0 & 1 & 1 \\ 1 & 4 & 0 \\ 1 & 0 & 4 \end{vmatrix} = -8 \quad \text{and} \quad \Delta_4 = \begin{vmatrix} 0 & 1 & 1 & 1 \\ 1 & 4 & 0 & 0 \\ 1 & 0 & 4 & 0 \\ 1 & 0 & 0 & 4 \end{vmatrix} = -48$$

Since Δ_3 and Δ_4 both are negative, $x_0 = (6, 2, 3)$ provides the solution to the NLPP.

Hence, the stationary point is a local minimum. Thus $x_0 = (6, 2, 3)$ provides the solution to the NLPP.

Example 6: Derive the necessary conditions for the non-linear programming problem.

Minimize, $z = f(x_1, x_2) = 3e^{2x_1+1} + 2e^{2x_2+5}$

subject to the constraint: $x_1 + x_2 = 7;\ x_1, x_2 \geq 0$

Solution: Let us introduce a new differential Lagrangian function $L(x_1, x_2, \lambda) = f(x_1, x_2) - \lambda(x_1 + x_2 - 7) = 3e^{2x_1+1} + 2e^{x_2+5} - \lambda(x_1 + x_2 - 7)$

Where λ is an Lagrangian multiplier.

Since the objective function $z = f(x_1, x_2)$ is convex and constraint an equality, the necessary and sufficient conditions for the minimum of $f(x_1, x_2)$ are given by,

$$\frac{\partial L}{\partial x_1} = 6e^{2x_1+1} - \lambda = 0 \text{ or } \lambda = 6e^{2x_1+1}$$

$$\frac{\partial L}{\partial x_2} = 2e^{2x_2+5} - \lambda = 0 \text{ or } \lambda = 2e^{x_2+5}$$

$$\frac{\partial L}{\partial \lambda} = -(x_1 + x_2 - 7) = 0 \text{ or } x_1 + x_2 = 7$$

Using these three, we get

$$6e^{2x_1+1} = 2e^{x_2+5} \text{ or } 3e^{2x_1+1} = e^{x_2+5} = e^{7-x_1+5}$$

Or $\log 3 + (2x_1 + 1) = 7 - x_1 + 5$ or $x_1 = \frac{1}{3}(11 - \log 3)$

Thus, $x_2 = 7 - \frac{1}{3}(11 - \log 3) = (10 + \log 3)/3$

Example 7: Derive the necessary conditions for the non-linear programming problem.

Minimize, $z = f(x_1, x_2) = 4x_1{}^2 + 2x_2{}^2 + x_3{}^2 - 4x_1x_2$

subject to the constraint: $x_1 + x_2 + x_3 = 15$; $2x_1 - x_2 + 2x_3 = 20$

Solution: Here, we have

$f(x) = 4x_1^2 + 2x_2^2 + x_3^2 - 4x_1x_2$; $h^1(x) = x_1 + x_2 + x_3 - 15$

$h^2(x) = 2x_1 - x_2 + 2x_3 - 20$

Construct the Lagrangian function,

$$L(x, \lambda) = f(x) - \lambda_1 h^1(x) - \lambda_2 h^2(x)$$

$$= (4x_1^2 + 2x_2^2 + x_3^2 - 4x_1x_2) - \lambda_1((x_1 + x_2 + x_3 - 15) - \lambda_2((2x_1 - x_2 + 2x_3 - 20)$$

The stationary point (x_0, λ_0) has thus given the following necessary consitions:

$$\frac{\partial L}{\partial x_1} = 8x_1 - 4x_2 - \lambda_1 - 2\lambda_2 = 0 \qquad \frac{\partial L}{\partial x_1} = 4x_2 - 4x_1 - \lambda_1 + \lambda_2 = 0$$

$$\frac{\partial L}{\partial x_3} = 2x_3 - \lambda_1 - 2\lambda_2 = 0 \qquad \frac{\partial L}{\partial \lambda_1} = -(x_1 + x_2 + x_3 - 15) = 0$$

$$\frac{\partial L}{\partial \lambda_2} = -(2x_1 - x_2 + 2x_3 - 20) = 0$$

So, solution of these simultaneous linear equations yields

$x_0 = (x_1, x_2, x_3) = (33/9, 10/3, 8)$ and $\lambda_0 = (\lambda_1, \lambda_2) = (40/9,\ 52/9)$

The board Hessian matrix at this solution (x_0, λ_0) is given by

$$H(x_0, \lambda_0) = \begin{bmatrix} 0 & 0 & \vdots & 1 & 1 & 1 \\ 0 & 0 & \vdots & 2 & -1 & 2 \\ \cdots & \cdots & \vdots & \cdots & \cdots & \cdots \\ 1 & 2 & \vdots & 8 & -4 & 0 \\ 1 & -1 & \vdots & -4 & 4 & 0 \\ 1 & 2 & \vdots & 0 & 0 & 2 \end{bmatrix}$$

Here since $n = 3$ and $m = 2$, therefore $n - m = 1$, $(2m + 1 = 5)$. This means that one needs to check the determinant of $H(x_0, \lambda_0)$ only and it must have the sign of $(-1)^2$.

Now, since $\det(H(x_0, \lambda_0)) = 90 > 0, x_0$ is a minimum point.

9.7 *CONSTRAINED OPTIMIZATION WITH INEQUALITY CONSTRAINTS*

Constrained optimization with inequality constraints involves finding the optimal solution (either a maximum or minimum) of an objective function subject to certain inequality constraints. The Karush-Kuhn-Tucker (KKT) conditions are a set of necessary (and under certain conditions, sufficient) conditions used to solve such problems.

Problem Formulation

Consider the following general constrained optimization problem:

Minimize (or Maximize) $f(x)$

Subject to:

$$g_i(x) \le 0,\ i = 1, 2, ..., m$$
$$h_j(x) \le 0,\ j = 1, 2, ..., p$$

Where:

- $f(x)$ is the objective function.
- $g_i(x) \le 0$ are the inequality constraints.
- $h_j(x) = 0$ are the equality constraints.
- $x = (x_1, x_2, ..., x_n)$ are the decision variables.

Lagrangian Function

The Lagrangian function for the problem is defined as:

$$L(x, \lambda, \mu) = f(x) + \sum_{j=1}^{p} \lambda_j h_j(x) + \sum_{i=1}^{m} \mu_i g_i(x)$$

Where:

- λ_j are the Lagrange multipliers associated with the equality constraints.
- μ_i are the Lagrange multipliers associated with the inequality constraints.

KKT Conditions (Karush-Kuhn-Tucker Conditions)

The KKT conditions are a set of necessary conditions for a solution x^* to be optimal:

1. **Stationarity**:

$$\nabla f(x^*) + \sum_{j=1}^{p} \lambda_j^* h_j(x^*) + \sum_{i=1}^{m} \mu_i^* g_i(x^*) = 0$$

This condition ensures that the gradient of the objective function can be expressed as a linear combination of the gradients of the constraints.

2. **Primal Feasibility**:

$h_j(x^*) = 0,$ for all $j = 1, 2, ..., p$

$g_i(x^*) \leq 0,$ for all $i = 1, 2, ..., m$

The solution x^* must satisfy the original constraints.

3. **Dual Feasibility**:

$\mu_i^* \geq 0$, for all $i = 1, 2, ..., m$

The Lagrange multipliers associated with the inequality constraints must be non-negative.

4. **Complementary Slackness**:

$\mu_i^* g_i(x^*) = 0,$ for all $i = 1, 2, ..., m$

This condition ensures that for any active constraint (where $g_i(x^*) = 0$), the corresponding multiplier μ_i^* is positive, and for any inactive constraint (where $g_i(x^*) < 0)$), the corresponding multiplier μ_i^* is zero.

Sufficient Conditions

If the objective function $f(x)$ is convex, and the inequality constraints $g_i(x) = 0)$ are convex (or concave for a maximization problem), the KKT conditions are not only necessary but also sufficient for optimality.

Example 8: Suppose you want to minimize $f(x, y) = x^2 + y^2$ subject to the inequality constraint $g(x, y) = x + y - 1 \leq 0$

1. **Lagrangian Function**: $L(x, y, \mu) = x^2 + y^2 + \mu(x + y - 1)$
2. **Stationarity:** $\frac{\partial L}{\partial x} = 2x + \mu = 0, \frac{\partial L}{\partial y} = 2y + \mu = 0$

   Solving gives $x = -\mu/2$ and $y = -\mu/2$

3. **Primal Feasibility**: $x + y - 1 \leq 0$

   Substituting $x = -\mu/2$ and $y = -\mu/2$ gives $\mu \geq 2$

4. **Dual Feasibility**: $\mu \geq 0$
5. **Complementary Slackness**: $\mu(x + y - 1) = 0$

   Since $\mu \geq 2$, the constraint is active and must hold as an equality: $x + y = 1$.

6. Substituting $x = y$ and $x + y = 1$ gives $x = y = 1/2$

Thus, the solution $(x^*, y^*) = (1/2, 1/2)$ satisfies the KKT conditions, indicating it's an optimal solution.

The KKT conditions provide a powerful framework for solving constrained optimization problems with inequality constraints. By satisfying these conditions, one can determine whether a solution is optimal under given constraints.

Note: The following set of Kuhn-Tucker conditions is obtained:

$$\begin{array}{|ll|} \hline f_j - \lambda h_j = 0, \ j = 0, 1, 2, \ldots, n & \\ \lambda h = 0, & \text{minimize } f \\ h \geq 0, & \textit{subject to}: \\ \lambda \geq 0, & h \geq 0 \\ \hline \end{array}$$

9.8 *KUHN-TUCKER CONDITIONS FOR GENERAL NLPP WITH m(<n) CONSTRAINTS*

The KKT conditions are a set of necessary conditions for a solution x^* to be optimal, particularly in the context where $m < n$:

1. **Stationarity**:
    - The gradient of the Lagrangian with respect to the decision variables *x* must be zero:

$$\nabla f(x^*) + \sum_{j=1}^{p} \lambda_j^* h_j(x^*) + \sum_{i=1}^{m} \mu_i^* g_i(x^*) = 0$$

2. **Primal Feasibility**:
    - The point x^* must satisfy the original constraints:

$h_j(x^*) = 0$, for all $j = 1, 2, ..., p$

$g_i(x^*) \le 0$, for all $i = 1, 2, ..., m$

3. **Dual Feasibility**:
    - The Lagrange multipliers associated with the inequality constraints must be non-negative:

    $\mu_i^* \ge 0$, for all $i = 1, 2, ..., m$

4. **Complementary Slackness**:
    - For each inequality constraint, the product of the Lagrange multiplier and the constraint function must be zero:

    $\mu_i^* g_i(x^*) = 0$, for all $i = 1, 2, ..., m$

    This condition implies that if $g_i(x^*) < 0$ (constraint is inactive), then $\mu_i^* = 0$, and if $g_i(x^*) = 0$ (constraint is active), then $\mu_i^* \ge 0$.

Implications of m<n

- **Degrees of Freedom**: When *m* < *n*, there are more decision variables than constraints, which means the feasible region may have some degrees of freedom even at the optimum. This can lead to a solution space that is not unique unless additional conditions (e.g., convexity) are imposed.

- **Feasibility and Redundancy**: The problem may have multiple optimal solutions or a region of optimal solutions rather than a single point, depending on the structure of the objective function and constraints.
- **Lagrange Multipliers Interpretation**: The values of μ_i^* indicate the sensitivity of the objective function to the constraints. If $\mu_i^* > 0$, the corresponding constraint is active and binds the solution, while if $\mu_i^* = 0$, the constraint is not binding at the optimal solution.

Example 9: Suppose we want to minimize $f(x_1, x_2) = x_1^2 + x_2^2$ subject to a single constraint:

$g(x_1, x_2) = x_1 + x_2 - 1 \le 0$

Here, $m = 1$ and $n = 2$

1. **Lagrangian Function**:

$$f(x_1, x_2, \mu_1) = x_1^2 + x_2^2 + \mu_1(x_1 + x_2 - 1)$$

2. **Stationarity**:
   - Partial derivatives with respect to x_1 and x_2:

$$\frac{\partial L}{\partial x_1} = 2x_1 + \mu_1 = 0$$

$$\frac{\partial L}{\partial x_2} = 2x_2 + \mu_1 = 0$$

Solving these gives $x_1 = x_2 = -\frac{\mu_1}{2}$

3. **Primal Feasibility**: $x_1 + x_2 - 1 \le 0$. Substituting $x_1 = x_2$ gives $2x_1 \le 1, x_1 \le 1/2$
4. **Dual Feasibility**: $\mu_1 \ge 0$
5. **Complementary Slackness**: $\mu_1(x_1 + x_2 - 1) = 0$

Substituting $x_1 = x_2 = 1/2$ satisfies this condition with $\mu_1 = 1$

Thus, the solution $(x_1^*, x_2^*) = (1/2, 1/2)$ with $\mu_1^* = 1$ satisfies the KKT conditions, indicating it is an optimal solution.

The KKT conditions provide a powerful framework for solving NLPPs with inequality constraints, even when the number of constraints is less than the number of variables. These conditions involve stationarity, primal and dual feasibility, and complementary slackness. When $m < n$, the solution may not be unique, and understanding the role of Lagrange multipliers becomes critical in interpreting the optimality conditions.

Example 10: Maximize $z = 3.6x_1 - 0.4x_1^2 + 1.6x_2 - 0.2x_2^2$; subject to the constraints:

$2x_1 + x_2 \le 10$ and $x_1, x_2 \ge 0$

Solution: $f(x) = 3.6x_1 - 0.4x_1^2 + 1.6x_2 - 0.2x_2^2$

$g(x) = 2x_1 + x_2$; $c = 10$

$h(x) = g(x) - c = 2x_1 + x_2 - 10$

The Kuhn-Tucker conditions are:

$$\frac{\partial f(x)}{\partial x_1} - \lambda \frac{\partial h(x)}{\partial x_1} = 0; \frac{\partial f(x)}{\partial x_2} - \lambda \frac{\partial h(x)}{\partial x_2} = 0; \lambda h(x) = 0; h(x) \le 0; \lambda \ge 0$$

Where λ is as usual the Lagrangian multiplier.

i.e., $3.6 - 0.8x_1 = 2\lambda$ … (1)

$1.6 - 0.4x_2 = \lambda$ … (2)

$\lambda[2x_1 + x_2 - 10] = 0$ … (3)

$2x_1 + x_2 - 10 = 0$ … (4)

$\lambda \ge 0$ … (5)

From equation (3) either $\lambda = 0$ or $2x_1 + x_2 - 10 = 0$.

Let $\lambda = 0$, then (2) and (1) yield $x_1 = 4.5$ and $x_3 = 4$. With theses value of x_1 and x_2 however (4) cannot be satisfied. Thus, optimal solution cannot be obtained here for $\lambda = 0$. Let then $\lambda \ne 0$

, which implies [from (3)] that $2x_1 + x_2 - 10 = 0$. This together with (1) and (2) yields the stationary value

$$x_0 = (x_1, x_2) = (3.5, 3)$$

Now, it is easy to observe that $h(x)$ is convex in *x*, and $f(x)$ is concave in *x*. Kuhn-Tuker requirements are therefore the necessary conditions for the minimum. Hence $x_0 = (3.5, 3)$ is the solution to the given NLPP. The maximum value of z (corresponding to x_0) is given by,

$$z_0 = 10.7$$

Example 11: Determine x_1, x_2, x_3 so as to,

Maximize $z = -x_1^2 - x_2^2 - x_3^2 + 4x_1 + 6x_2$

Subject to the constraints: $x_1 + x_2 \le 2; 2x_1 + 3x_2 \le 12; x_1, x_2 \ge 0$

Solution: 1 Here, $f(x) = -x_1^2 - x_2^2 - x_3^2 + 4x_1 + 6x_2$

$h^1(x) = x_1 + x_2 - 2; h^2(x) = 2x_1 + 3x_2 - 12$

Clearly, $f(x)$ is concave and $h^1(x), h(x)$ are convex in *x*. Thus the Kuhn-Tucker conditions will be the necessary and sufficient conditions for a maximum. These conditions are obtained by the partial differentiation of the Lagrangian function.

$$L(x, S, \lambda) = f(x) - \lambda_1[h^1(x) + S_1^2] - \lambda_2[h^2(x) + S_2^2]$$

Where, $S = (S_1, S_2), \lambda = (\lambda_1, \lambda_2), S_1, S_2$ being slack variable and λ_1, λ_2 are the Lagrange multipliers. Now, the Kuhn-Tucker conditions are given by,

- $f_j = \sum_{i=1}^{m} \lambda_i h_j^i \qquad (j = 1, 2, 3)$
- $\lambda_i h_j^i = 0 \qquad (j = 1, 2)$
- $h^i \le 0 \qquad (i = 1, 2)$
- $\lambda_i \ge 0 \qquad (i = 1, 2)$

Thus, in problem these are

1 (i) $-2x_1+4=\lambda_1+2\lambda_2$ (ii) $-2x_1+6=\lambda_1+3\lambda_2$ (iii) $-2x_3=0$

2 (i) $\lambda_1(x_1+x_2-2)=0$ (ii) $\lambda_2(2x_1+3x_2-12)=0$

3 (i) $x_1+x_2-2\le 0$ (ii) $2x_1+3x_2-12\le 0$

4 (i) $\lambda_1\ge 0, \lambda_2\ge 0$

Now, there are 4 cases will be arises:

Case I: $\lambda_1=0$ and $\lambda_2=0$, (i), (ii) and (iii) yield $x_1=2, x_2=3, x_3=0$

However, this solution violates (3) [(i) and (ii) both], and it must therefore be discarded.

Case II: $\lambda_1=0$ and $\lambda_2\ne 0$, (2) yield $2x_1+3x_2=12$ and (1) (i) and (ii) yield $-2x_1+4=2\lambda_2$, $-2x_2+6=3\lambda_2$. The solution of these simultaneous equation yields $x_1=2/13$, $x_2=3/13$, $\lambda_2=24/13>0$; also (1) (iii) gives $x_3=0$. However, this solution violates (3) (i). This solution is also thus discarded.

Case III: $\lambda_1\ne 0$ and $\lambda_2\ne 0$, (2) (i) and (ii) yield $x_1+x_2=2$ and $2x_1+3x_2=12$. These together yield $x_1=-6$, $x_2=8$. Thus (1) (i), (ii) and (iii) give $x_3=0$, $\lambda_1=68$, $\lambda_2=-26$. However, this solution is to be discarded since $\lambda_2=-26$ violates (4).

Case IV: $\lambda_1\ne 0$ and $\lambda_2=0$, (2) (i) and (ii) yield $x_1+x_2=0$ and $2x_1+3x_2=12$. This together with (1) (i) and (ii) gives $x_1=1/2$, $x_2=3/2, \lambda_1=3>0$. Further from (1) (iii) give $x_3=0$. We observe that this solution does not violates any of the Kuhn-Tucker conditions.

Hence, the optimal solution to the given NLPP is

$x_1=1/2$, $x_2=3/2, x_3=0; \lambda_1=3, \lambda_2=0$.

Hence the maximum value of objective function is $z_0=17/2$

Check your progress

Problem 1: Optimize $2x_1 + 3x_2 - (x_1{}^2 + x_2{}^2 + x_3{}^2)$

Subject to, $x_1 + x_2 \le 1; 2x_1 + 3x_2 \le 6; x_1 \ge 0, x_2 \ge 0$

Answer: The optimal solution is, $x_1^0 = 1/4, x_2^0 = 3/4, x_3^0 = 0$ Maximum total cost z = 17/8

Problem 2: Using the Kuhn-Tucker condition solve the given NLPP.

Minimize, $z = 2x_1{}^2 + 12x_1x_2 - 7x_2{}^2$

Subject to the constraints, $2x_1 + 5x_2 \le 98; x_1 \ge 0, x_2 \ge 0$

Answer: $x_1 = 44$, $x_2 = 2$; minimum $z = 4900$.

Problem 3: Using Lagrangian multiplier solve the following NLPP.

Minimize, $z = 6x_1{}^2 + 5x_2{}^2$

Subject to the constraints, $x_1 + 5x_2 = 3; x_1 \ge 0, x_2 \ge 0$

Answer: $x_1 = 3/31$, $x_2 = 18/31$; minimum $z = 54/31$.

9.9 *SUMMARY*

Nonlinear Programming Problems (NLPPs) extend the principles of optimization to more complex scenarios where linear assumptions are inadequate. By introducing powerful tools like the Lagrange multiplier method and the Kuhn-Tucker conditions, NLPPs can be tackled despite their complexity. Understanding the nature of the objective function, constraints, and feasible region is key to solving these problems effectively. While algorithms for NLPPs are more sophisticated and computationally demanding than their linear counterparts, they are essential for optimizing real-world systems where nonlinearity is the norm.

9.10 *GLOSSARY*

- Non-linear programming
- Kuhn-Tucker condition for general NLPP

9.11 *REFERENCES*

- Mokhtar S. Bazaraa, John J. Jarvis and Hanif D. Sherali: *Linear Programming and Network Flows* (4th edition). John Wiley and Sons, 2010.
- Hamdy A. Taha: *Operations Research: An Introduction (10th edition). Pearson*, 2017.
- Paul R. Thie and Gerard E. Keough: *An Introduction to Linear Programming and Game Theory* (3rd edition), Wiley India Pvt. Ltd, 2014.
- Swarup, K., Gupta, P. K., & Mohan, M. (2017). An introduction to management science operation research. *Sultan Chand & Sons educational publishers, New Delhi*.
- OpenAI. (2024). *ChatGPT (August 2024 version) [Large language model]*. OpenAI. https://www.openai.com/chatgpt

9.12 *SUGGESTED READING*

- G. Hadley, *Linear Programming*, Narosa Publishing House, 2002.
- Frederick S. Hillierand Gerald J. Lieberman: *Introduction to Operations Research* (10th edition). McGraw-Hill Education, 2015.
- https://epgp.inflibnet.ac.in/Home/ViewSubject?catid=ZLCHeZEhCZ8yCri36nSF3A==

9.13 *TERMINAL QUESTION*

Short Answer Type Question:

1: Using Lagrangian multiplier solve the following NLPP.

Minimize, $f(x_1, x_2) = 3x_1^2 + x_2^2 + 2x_1x_2 + 6x_1 + 2x_2$

Subject to the constraints, $2x_1 - x_2 = 4; x_1 \geq 0, x_2 \geq 0$

2: Using Lagrangian multiplier solve the following NLPP.

Maximize, $z = 5x_1 + x_2 - (x_1 - x_2)^2$

Subject to the constraints, $x_1 + x_2 = 4; x_1 \geq 0, x_2 \geq 0$

3: Using Lagrangian multiplier solve the following NLPP.

Maximize, $z = 4x_1 + 6x_2 - 2x_1^2 - 2x_1x_2 - 2x_2^2$

Subject to the constraints, $x_1 + 2x_2 = 2; x_1 \geq 0, x_2 \geq 0$

4: Using Lagrangian multiplier solve the following NLPP.

Minimize, $z = 2x_1^2 + 2x_2^2 + 3x_3^2 + 10x_1 + 8x_2 + 6x_3 - 100$

Subject to the constraints, $x_1 + x_2 + x_3 = 20; x_1 \geq 0, x_2 \geq 0, x_3 \geq 0$

5: Using Lagrangian multiplier solve the following NLPP.

Minimize, $z = x_1^2 + x_2^2 + x_3^2$

Subject to the constraints, $4x_1 + x_2^2 + 2x_3 = 14; x_1 \geq 0, x_2 \geq 0, x_3 \geq 0$

Long answer type question:

1: Find the optimal solution maximises or minimises the objective function for the given NLPP.

Minimize, $z = x_1^2 - 10x_1 + x_2^2 - 6x_2 + x_3^2 - 4x_3$

Subject to the constraints, $4x_1 + x_2 + x_3 = 7; x_1 \geq 0, x_2 \geq 0, x_3 \geq 0$

2: Solve the following NLPP

Optimize, $z = 4x_1 + 9x_1 - x_1^2 - x_2^2$

Subject to the constraints, $4x_1 + 3x_2 = 15; 3x_1 + 5x_2 = 14; x_1 \geq 0, x_2 \geq 0$

3: Solve the following NLPP

Optimize, $z = x_1^2 + x_2^2 + x_3^2$

Subject to the constraints, $x_1 + x_2 + 3x_3 = 2; 5x_1 + 2x_2 + x_3 = 5; x_1 \geq 0, x_2 \geq 0, x_3 \geq 0$

4: Using the Kuhn-Tucker condition solve the given NLPP

Maximize, $z = 8x_1 + 10x_2 - x_1^2 - x_2^2$

Subject to the constraints, $3x_1 + 2x_2 \leq 6; x_1 \geq 0, x_2 \geq 0, x_3 \geq 0$

5: Using the Kuhn-Tucker condition solve the given NLPP

Minimize, $z = x_1^2 + x_2^2 + x_3^2$

Subject to the constraints, $2x_1 + x_2 \le 5; x_1 + x_2 \le 2, x_1 \ge 1, x_2 \ge 2, x_3 \ge 0$

6: Using the Kuhn-Tucker condition solve the given NLPP

Minimize, $z = 0.3x_1^2 - 2x_1 + 0.4x_2^2 - 2.4x_2 + 0.6x_1x_2 + 100$

Subject to the constraints, $2x_1 + x_2 \ge 4; x_1 \ge 0, x_2 \ge 0$.

Objective type question:

1: In a general nonlinear programming problem, the objective function and constraints:

A) Must be linear.
B) Must be quadratic.
C) Can be any nonlinear functions.
D) Must be polynomial functions.

2: The Kuhn-Tucker conditions are used in nonlinear programming problems to:

A) Find the global minimum.
B) Check the feasibility of the problem.
C) Provide necessary conditions for optimality in the presence of inequality constraints.
D) Ensure that the solution is unique.

3: In the Kuhn-Tucker conditions, complementary slackness implies:

A) The constraint is always active.
B) The Lagrange multiplier must be non-negative.
C) If the constraint is not binding, the corresponding multiplier is zero.
D) The objective function has a unique solution.

4: For a point to be a local minimum in a nonlinear programming problem with equality constraints, the gradient of the Lagrangian function with respect to the decision variables must:

A) Be positive.
B) Be negative.
C) Equal zero.
D) Equal the gradient of the objective function.

5: In the context of NLPP, a problem is considered convex if:

A) The objective function is quadratic.
B) The Hessian matrix is positive semi-definite and constraints form a convex set.
C) The constraints are all linear.
D) The solution is unique.

6: The Lagrangian function for a constrained optimization problem is defined as:

A) The sum of the objective function and the constraints.
B) The objective function minus the constraints.
C) The objective function plus a weighted sum of the constraints.
D) The gradient of the objective function.

7: In nonlinear programming, if the problem has more variables than constraints (m<n), what can we generally say about the feasible region?

A) It is always empty.
B) It is likely to be a single point.
C) It might have degrees of freedom, meaning there could be multiple optimal solutions.
D) It is always a bounded region.

8: In nonlinear programming, if the problem has more variables than constraints (m<nm < nm<n), what can we generally say about the feasible region?

A) It is always empty.
B) It is likely to be a single point.
C) It might have degrees of freedom, meaning there could be multiple optimal solutions.
D) It is always a bounded region.

True and False:

1: In nonlinear programming, all constraints must be linear functions.

2: The Kuhn-Tucker conditions are necessary conditions for optimality in nonlinear programming problems with inequality constraints.

3: If the Hessian matrix of the Lagrangian function is positive definite, the solution is guaranteed to be a global minimum.

4: The Lagrange multiplier method can only be applied to optimization problems with equality constraints.

5: In a nonlinear programming problem, the objective function can be any non-linear function, including exponential and logarithmic functions.

6: The feasible region of a nonlinear programming problem is always convex.

Fill in the blanks:

1: The method of Lagrange multipliers is used to find the _______________ of a function subject to equality constraints.

2: In a nonlinear programming problem, the ____________ function is formed by adding the objective function to a linear combination of the constraints multiplied by their corresponding Lagrange multipliers.

3: The ____________ conditions are a generalization of the method of Lagrange multipliers to include inequality constraints.

4: A necessary condition for a solution to be optimal in a nonlinear programming problem is that the gradient of the Lagrangian function with respect to the decision variables must be equal to ____________.

5: In the Kuhn-Tucker conditions, if a constraint is not binding (inactive), the corresponding Lagrange multiplier must be ____________.

6: For a nonlinear programming problem to be convex, the objective function must be a convex function, and the feasible region must also be a ____________ set.

7: The ____________ matrix of a function is used to determine the nature (convexity or concavity) of a nonlinear function in optimization problems.

8: In a nonlinear programming problem, the ____________ region is defined by the set of all points that satisfy the constraints.

9: When the number of constraints is less than the number of variables ($m<n$), the feasible region may have some degrees of ____________, leading to a potential family of solutions.

10: In constrained optimization, the solution that satisfies all constraints and optimizes the objective function is known as the ____________ feasible solution.

9.14 *ANSWERS*

Answer of short answer type question

Answer 1: $x_1 = 1\ and\ x_2 = -2; Maximum\ f(x_1, x_2) = 5$

2: $x_1 = 5/2\ and\ x_2 = 3/2; Maximum\ z = 13$

3: $x_1 = 1/3\ and\ x_2 = 5/6; Maximum\ z = 4.16$

4: $x_1 = 5, x_2 = 11\ and\ x_3 = 4; Minimum\ z = 281$

5: $x_1 = 2, x_2 = 2\ and\ x_3 = 1; Minimum\ z = 9$

Answer of long answer type question

Answer 1: $x_1 = 4, x_2 = 2\ and\ x_3 = 1; Minimum\ z = -35$

2: $x_1 = 3, x_2 = 1; Minimum\ z = 11$

3: $x_1 = 0.81, x_2 = 0.35\ and\ x_3 = 0.28; Minimum\ z = 0.857$

4: $x_1 = 4/13, x_2 = 33/13; Maximum\ z = 21.3$

5: $x_1 = 1, x_2 = 2, x_3 = 0; Minimum\ z = 5$

6: $x_1 = 1.3, x_2 = 2; Minimum\ z = 96.3$ **7:** $x_1 = 1, x_2 = 3/4; Minimum\ z = 17/8$

Answer of Multiple choice question

Answer 1: C **2:** C **3:** C **4:** C

5: B **6:** C **7:** D **8:** C

True and False

Answer 1: False **2:** True **3:** False

4: False **5:** True **6:** False

Answer of fill in the blank question

Answer 1:	extremum	**2:**	Lagrangian	**3:**	Kuhn-Tucker
4:	Zero	**5:**	zero	**6:**	convex
7:	Hessian	**8:**	feasible	**9:**	freedom
10:	optimal				

UNIT-10: QUADRATIC PROGRAMMING

CONTENTS:

10.1 *INTRODUCTION*

In non-linear programming (NLPP), optimization problems are characterized by objective functions or constraints that are non-linear. Two important approaches within NLPP are the **graphical method** and **quadratic programming**.

The **graphical method** is a visual technique primarily used for solving small-scale optimization problems involving two variables. By plotting the objective function and constraints on a graph, the feasible region is identified, and the optimal solution is determined by analyzing where the objective function achieves its best value within this region. This method is intuitive and useful for providing a clear, visual understanding of the problem, but it is limited to problems with only two decision variables.

Quadratic programming (QP) is a specialized type of NLPP where the objective function is quadratic, typically involving squared terms of the variables, while the constraints are linear. QP problems are prevalent in many practical applications, such as portfolio optimization in finance, where returns and risks (modeled quadratically) need to be optimized under certain constraints. The quadratic nature of the objective function allows for modeling complex relationships, and the solution methods for QP, such as the Karush-Kuhn-Tucker (KKT) conditions, are well-established and powerful for finding optimal solutions in these non-linear environments.

Both the graphical method and quadratic programming play vital roles in the broader scope of non-linear programming, offering tools for solving different types of optimization problems that are non-linear in nature.

10.2 *OBJECTIVE*

After reading this unit learners will be able to

- Understand to solve the non-linear programming problem by graphical method.
- Learn to solve the NLPP by quadratic programming.

10.3 *NON-LINEAR PROGRAMMING METHODS*

Below is an overview of some commonly used nonlinear programming methods:

1. Gradient-Based Methods

- **Gradient Descent**: This method iteratively moves in the direction of the steepest descent (negative gradient) to find a local minimum. It is simple and widely used but can be slow and may get stuck in local minima.
- **Newton's Method**: An iterative method that uses the second-order derivative (Hessian matrix) to find the optimal solution. It has a faster convergence rate compared to gradient descent but requires the computation of the Hessian matrix, which can be computationally expensive.
- **Quasi-Newton Methods**: These methods approximate the Hessian matrix to reduce computational complexity. Examples include the Broyden–Fletcher–Goldfarb–Shanno (BFGS) algorithm and its limited-memory version (L-BFGS), which are popular in practice.

2. Conjugate Gradient Methods: These methods are used to solve large-scale optimization problems. They improve upon the basic gradient descent method by using information from previous iterations to converge more rapidly. The method is effective when the objective function is quadratic or nearly quadratic.

3. Interior Point Methods: These methods are effective for large-scale nonlinear optimization problems. They work by maintaining feasibility with respect to the inequality constraints while approaching the optimal solution from within the feasible region. Interior point methods are particularly useful for problems with a large number of constraints.

4. Penalty and Barrier Methods

- **Penalty Methods**: These methods convert a constrained optimization problem into an unconstrained one by adding a penalty term to the objective function for any violation of the constraints. The penalty term increases as the solution approaches the boundary of the feasible region.
- **Barrier Methods**: Similar to penalty methods, but they add a barrier term that prevents the solution from leaving the feasible region. The barrier term becomes infinite at the boundary, forcing the solution to stay within the feasible region.

5. Sequential Quadratic Programming (SQP): SQP is a powerful and widely used method for solving NLP problems. It iteratively solves a quadratic programming subproblem that approximates the original problem. SQP is effective for problems with smooth nonlinear functions and constraints.

6. Augmented Lagrangian Methods: This method combines the penalty function approach with the Lagrange multiplier method. It adds a penalty for constraint violation and adjusts Lagrange multipliers to enforce constraints more accurately. The method is suitable for problems with both equality and inequality constraints.

7. Genetic Algorithms (GA): GA is a stochastic, population-based optimization technique inspired by natural selection. It is used for solving complex NLP problems where traditional gradient-based methods are not effective. GA explores a broader search space and is less likely to get stuck in local optima.

8. Simulated Annealing: This is a probabilistic technique that mimics the annealing process in metallurgy. It is used to find an approximate global optimum by allowing occasional uphill moves, which helps in escaping local minima. The method gradually reduces the probability of such moves, "cooling down" the search as it converges.

9. Trust Region Methods: These methods define a region around the current solution within which the model is trusted to be an accurate approximation of the objective function. The method iteratively adjusts the size of this region based on the accuracy of the model.

10. Direct Search Methods

- **Nelder-Mead (Simplex) Method**: This method does not require derivative information and is used for problems where the objective function is not smooth or differentiable. It is based on the concept of a simplex (a polytope of $n+1$ vertices in n-dimensional space).
- **Pattern Search**: A derivative-free optimization method that explores the search space by evaluating the objective function at a set of points forming a pattern around the current solution.

11. Kuhn-Tucker (Karush-Kuhn-Tucker) Conditions: These are first-order necessary conditions for optimality in NLP problems with inequality constraints. The KKT conditions generalize the method of Lagrange multipliers and are fundamental in the theory of nonlinear optimization.

12. Dynamic Programming: Although not exclusively a nonlinear programming method, dynamic programming can be applied to multi-stage decision problems where the objective function and constraints are nonlinear. It involves breaking down the problem into simpler subproblems and solving them recursively.

10.4 *GRAPHICAL METHOD TO SOLVE NLPP*

Graphical solutions are typically used for understanding and solving optimization problems with two decision variables. While graphical methods are commonly applied to linear programming problems, they can also be extended to simple nonlinear programming problems to gain visual insights.

Steps for Graphical Solution of NLPP:

1. **Formulate the Problem**:
   - Define the objective function, which can be nonlinear (e.g., quadratic, exponential).
   - Specify the constraints, which can be linear or nonlinear, forming the feasible region.
2. **Plot the Constraints**:
   - Represent each constraint as an equation on a two-dimensional graph.

- Identify the feasible region by shading the area that satisfies all the constraints.

3. **Plot the Objective Function**:
   - The objective function is plotted as contour lines (iso-profit or iso-cost lines) on the same graph. Each contour line represents points with the same objective function value.
   - Since the objective function is nonlinear, these contour lines may be curves instead of straight lines.
4. **Identify the Feasible Region**:
   - The feasible region is the area on the graph where all the constraints overlap. For a nonlinear programming problem, this region may not be a simple polygon and can be curved or bounded by nonlinear constraints.
5. **Optimize the Objective Function**:
   - Move the contour lines of the objective function parallel to themselves across the feasible region.
   - Depending on whether you are minimizing or maximizing the function, look for the contour line that touches the feasible region at the optimal point(s).
6. **Determine the Optimal Solution**:
   - The optimal solution is the point (or points) in the feasible region where the contour line of the objective function either reaches its maximum or minimum value.
   - In the case of nonlinear constraints, this might occur at the boundary of the feasible region or at a point where the gradient of the objective function is perpendicular to the constraint boundary.

Example 1: Maximize $f(x, y) = xy$ (a nonlinear objective function).
Subject to, $x + y \leq 6$ (linear constraint); $x \geq 0$; $y \geq 0$

Graphical Solution:

1. **Plot the Constraint** $x + y \leq 6$ on a graph:
   - Draw the line $x + y = 6$.
   - Shade the area below and to the right of the line, where the inequality holds.
   - Also, plot the $x \geq 0$ and $y \geq 0$ constraints, which restrict the feasible region to the first quadrant.
2. **Plot the Objective Function** $f(x, y) = xy$:
   - Contour lines for $f(x, y) = xy$ will be hyperbolas of the form $xy = c$ where c is a constant.
   - Plot several contour lines, for example, $xy = 2, 4, 6, 8$ etc.

3. **Optimize**:
    - Move the hyperbolas upwards to find the highest one that still touches the feasible region.
    - The optimal solution will be at the point where the highest contour line of xy touches the boundary of the feasible region, which in this case could be at a corner point or along the boundary.

Interpretation:

- The point of contact between the contour lines and the feasible region boundary represents the optimal solution.
- In more complex NLPPs, the feasible region might not be a simple polygon, and the objective function might have multiple local optima.

Graphical methods for solving nonlinear programming problems are limited to problems with two variables. They provide visual insights into how the feasible region and objective function interact. By plotting both the constraints and the nonlinear objective function, one can visually identify the optimal solution within the feasible region. While this method is not suitable for higher-dimensional problems, it is useful for educational purposes and for simple cases where a visual solution is feasible.

Example 2: Reduce the origin's distance from the convex area that is enclosed by the constraints.

$x_1 + x_2 \geq 4; 2x_1 + x_2 \geq 5;\ x_1, x_2 \geq 0$

At the point of minimal distance, confirm that the Kuhn-Tucker required requirements are met.
Solution: The issue of minimising the radius of a circle with the origin at its centre, let's say $r^2 = x_1^2 + x_2^2$, such that it contacts (passes through) the convex area enclosed by the specified constraints is equal to the problem of minimising the distance between the origin and the covex region. As a result, the issue is stated as,

Minimize $z(= r^2) = x_1^2 + x_2^2$

Subject to the constraints:

$x_1 + x_2 \geq 4;\ 2x_1 + x_2 \geq 5;\ \ x_1, x_2 \geq 0$

Graphical solution: In the plane, consider a set of rectangular Cartesian axes OX_1X_2. Every point has type (x_1, x_2) coordinates, and every ordered pair (x_1, x_2) of real numbers, on the other hand, determines a point on the plane.

It is obvious that any point that meets condition $x_1 \geq 0, x_2 \geq 0$ belongs in the first quadrant, and vice versa for any point (x_1, x_2) in the first quadrant that meets condition $x_1 \geq 0, x_2 \geq 0$. This means that the first quadrant point is the sole place where we may look for the number pair (x_1, x_2). The target point must thus be located in the unbounded convex area ABC (shown shaded in fig. 1) since $x_1 + x_2 \geq 4$ and $2x_1 + x_2 \geq 5$. The target point will be the location in the region where a side of the convex region is tangent to the circle, as our search is for such a pair (x_1, x_2) that produces a minimal value of $x_1^2 + x_2^2$ and lies in the convex region. Next, we move forward as follows:

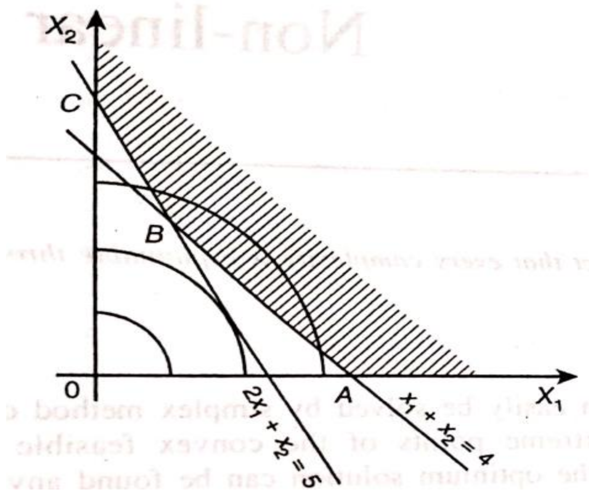


Fig. 1

After differentiating the equation of circle, we have $2x_1 dx_1 + 2x_2 dx_2 = 0$, yielding

$$\frac{dx_2}{dx_1} = -\frac{x_1}{x_2}$$

Considering the equalities $2x_1 + x_2 = 5$ and $x_1 + x_2 = 4$, we have on differentiation

$$2dx_1 + dx_2 = 0 \text{ and } dx_1 + dx_2 = 0$$

These yield, $\frac{dx_2}{dx_1} = -2$ and $\frac{dx_2}{dx_1} = -2$ repectively.

Thus, from (1) and (2), we obtain

$$-\frac{x_1}{x_2} = -1 \quad \text{or} \quad x_1 = x_2 \quad \text{and} \quad -\frac{x_1}{x_2} = -2 \quad \text{or} \quad x_1 = 2x_2$$

This shows that the circle $r^2 = x_1^2 + x_2^2$ has a tangent to

(i) The line $x_1 + x_2 = 4$ at the point (2, 2)
(ii) The line $2x_1 + x_2 = 5$ at the point (2, 1)

Since the point (2, 1) is outside of the convex zone, it should be ignored. As a result, the shortest path between the origin and the convex region enclosed by the constraints is

$$z_0{}^2 = 2^2 + 2^2 = 8$$, and is given by the point (2,2).

10.5 *KUHN-TUCKER CONDITIONS WITH NON-NEGATIVE CONSTRAINTS*

The Kuhn-Tucker conditions, also known as the Karush-Kuhn-Tucker (KKT) conditions, are necessary conditions for a solution to be optimal in a nonlinear programming problem with inequality constraints, including non-negativity constraints.

Problem Formulation: Consider a general nonlinear programming problem of the form:

Minimize $f(x)$

subject to:

$g_i(x) \le 0, i = 1, 2, ..., m$; $x \ge 0$

where:

- $f(x)$ is the objective function,
- $g_i(x)$ are the inequality constraint functions,
- $x \ge 0$, represents the non-negativity constraints on the decision variables.

Kuhn-Tucker Conditions

For a solution x^* to be optimal, there must exist Lagrange multipliers $\lambda_i \ge 0$ for each inequality constraint $g_i(x) \ge 0$ and $\mu_j \ge 0$ for each non-negativity constraint $x_j \ge 0$ such that the following conditions hold:

1. **Stationarity**:

- The gradient of the Lagrangian function with respect to x must be zero at x^*:

$$\nabla f(x^*) + \sum_{i=1}^{m} \lambda_i \nabla g_i(x^*) - \sum_{j=1}^{n} \mu_j e_j = 0$$

where e_j is a unit vector with 1 in the j^{th} position and 0 elsewhere.

2. **Primal Feasibility**:

- The point x^* must satisfy all the original constraints:

$$g_i(x^*) \le 0, i = 1, 2, ..., m \text{ ; } x^* \ge 0$$

3. **Dual Feasibility**:

- The Lagrange multipliers associated with the inequality constraints and non-negativity constraints must be non-negative:

$$\lambda_i \ge 0, i = 1, 2, ..., m$$

$$\mu_j \ge 0, \ j = 1, 2, ..., n$$

4. **Complementary Slackness**:

- The product of each Lagrange multiplier and its corresponding constraint must be zero:

$$\lambda_i g_i(x^*) = 0, i = 1, 2, ..., m$$

$$\mu_j x_j^* = 0, \ j = 1, 2, ..., n$$

This condition ensures that either the constraint is active (equality holds) and the corresponding Lagrange multiplier is positive, or the constraint is inactive and the multiplier is zero.

Lagrangian Function

The Lagrangian function for this problem is:

$$L(x,\lambda,\mu) = f(x) + \sum_{i=1}^{m} \lambda_i g_i(x) - \sum_{i=1}^{m} \mu_j x_j$$

Interpretation of Kuhn-Tucker Conditions

- **Stationarity**: Ensures that the gradient of the Lagrangian (a combination of the objective function and the constraints) is zero, indicating a potential optimal point.
- **Primal Feasibility**: The solution must satisfy the original problem's constraints.
- **Dual Feasibility**: The Lagrange multipliers, which represent the shadow prices of the constraints, must be non-negative.
- **Complementary Slackness**: Ensures that for each constraint, either the constraint is binding and the corresponding multiplier is positive, or the constraint is non-binding and the multiplier is zero.

Example 3: Consider a simple NLP problem

Minimize $f(x, y) = x^2 + y^2$

subject to:

$g_1(x, y) = -x + 1 \le 0$

$g_2(x, y) = -y + 2 \le 0$

$x + 1 \le 0$

1. **Lagrangian Function**:

$$L(x, y, \lambda_1, \lambda_2, \mu_1, \mu_2) = x^2 + y^2 + \lambda_1(-x+1) + \lambda_2(-y+2) - \mu_1 x - \mu_2 y$$

2. **Stationarity**:

$$\frac{\partial L}{\partial x} = 2x - \lambda_1 - \mu_1 = 0$$

$$\frac{\partial L}{\partial y} = 2y - \lambda_2 - \mu_2 = 0$$

3. **Primal Feasibility**:

$$-x+1 \le 0; -y+2 \le 0; x \ge 0, y \ge 0$$

4. **Dual Feasibility**:

$$\lambda_1 \ge 0; \lambda_2 \ge 0; \mu_1 \ge 0, \mu_2 \ge 0$$

5. **Complementary Slackness**:

$$\lambda_1(-x+1) = 0; \lambda_2(-y+2) = 0; \mu_1 x = 0, \mu_2 y = 0$$

The optimal solution (x^*, y^*) must satisfy all these conditions.

The Kuhn-Tucker conditions provide a powerful framework for solving nonlinear programming problems with inequality constraints, including non-negativity constraints. These conditions generalize the method of Lagrange multipliers to handle inequality constraints and are essential for understanding the optimality of solutions in NLP problems.

10.6 *QUADRATIC PROGRAMMING*

Quadratic Programming (QP) is a special type of nonlinear programming where the objective function is quadratic, and the constraints are linear. QP problems are commonly encountered in various fields such as finance (portfolio optimization), engineering (control systems), and machine learning (support vector machines).

Problem Formulation

A standard Quadratic Programming problem can be formulated as:

Minimize, $f(x) = \frac{1}{2} x^T Q x + c^T x$

subject to: $Ax \le b$; $x \ge 0$

where:

- $x \in R^n$ is the vector of decision variables.
- $Q \in R^{n \times n}$ is a symmetric positive semi-definite matrix (this ensures the objective function is convex).

- $c \in R^n$ is a vector of coefficients.
- $A \in R^{m \times n}$ is a matrix representing the coefficients of the linear constraints.
- $b \in R^m$ is a vector representing the right-hand side of the constraints.

Characteristics

1. **Objective Function**:
- The objective function $f(x)$ is quadratic, meaning it includes terms of the form x_i^2 and $x_i x_j$.
- The matrix Q determines the curvature of the objective function. If Q is positive semi-definite, the problem is convex, and any local minimum is also a global minimum.
2. **Constraints**:
- The constraints are linear inequalities, which define a convex feasible region.
3. **Solution**:
- Due to the convex nature of the problem (if Q is positive semi-definite), quadratic programming problems can be efficiently solved using various optimization algorithms.

Solution Methods

1. Interior Point Methods:
- These methods are highly effective for solving large-scale QP problems. They approach the optimal solution from within the feasible region while ensuring that the iterates remain feasible with respect to the constraints.

2. Active Set Methods:
- Active set methods work by iteratively solving a sequence of equality-constrained subproblems. At each iteration, some constraints are considered active (treated as equalities), and the method optimizes over this reduced set of constraints.

3. Simplex-Based Methods:
- Extensions of the simplex method for linear programming, such as the Wolfe or the revised simplex method, can also solve QP problems by incorporating the quadratic nature of the objective function.

4. Gradient-Based Methods:
- Methods like projected gradient descent can be used for simple QP problems, especially when the feasible region is a simple set like a box constraint.

5. Dual Methods:
- The dual problem in QP often has a simpler structure and can be solved using dual decomposition or dual ascent methods.

Example 4: Consider the following QP problem:

Minimize, $f(x_1, x_2) = \frac{1}{2}(x_1^2 + 2x_2^2) + x_1 + 2x_2$

subject to: $x_1 + 2x_2 \le 1$

$x_1 \ge 0, x_2 \ge 0$

Solution:

1. **Objective Function**:

$$f(x) = \frac{1}{2} x^T Qx + c^T x$$

Where, $Q = \begin{pmatrix} 1 & 0 \\ 0 & 2 \end{pmatrix}$ and $c = \begin{pmatrix} 1 \\ 2 \end{pmatrix}$.

2. **Constraints**:

$A = (1 \quad 1),\ b = 1$

This problem can be solved using any of the QP methods mentioned above, resulting in the optimal values of x_1 and x_2 that minimize the objective function while satisfying the constraints.

Applications of QP

- **Portfolio Optimization**: Optimizing the trade-off between risk and return in a financial portfolio.
- **Support Vector Machines (SVMs)**: Training SVMs in machine learning involves solving a QP problem.
- **Control Systems**: QP is used in optimal control and model predictive control for systems with quadratic cost functions.
- **Structural Optimization**: Designing structures with minimum weight subject to stress and displacement constraints.

Quadratic Programming is a powerful optimization tool for problems where the objective function is quadratic, and the constraints are linear. The convex nature of QP problems (under certain conditions) ensures that they can be solved efficiently and reliably using various optimization methods. These problems are widely used in diverse fields, from finance to engineering to machine learning.

Check your progress

Problem 1: Using the Graphical method solve the following NLPP.

Maximize, $z = 8x_1 - x_1^2 + 8x_2 - x_2^2$

Subject to the constraints: $x_1 + x_2 \leq 12$; $x_1 - x_2 \geq 4$ and $x_1, x_2 \geq 0$

Answer: $x_1 = 6, x_2 = 2$; Maximum $z = 24$

10.7 *SUMMARY*

In non-linear programming (NLPP), the graphical method is a visual approach used to solve optimization problems with two variables. It involves plotting the objective function and constraints on a graph to identify the feasible region, which is the set of all points that satisfy the constraints. The optimal solution is typically found on the boundary of this region, where the objective function reaches its maximum or minimum value. This method is limited to problems with two variables and is most effective for simpler, visually interpretable problems.

Quadratic programming is a specific type of NLPP where the objective function is quadratic (i.e., involves terms with squares of variables) and the constraints are linear. The quadratic function may represent various scenarios, such as minimizing costs or maximizing profits under given constraints. The key characteristic of quadratic programming is the form of the objective function, which can either be convex (leading to a unique global minimum) or concave (leading to a unique global maximum). The solution to a quadratic programming problem can be found using various methods, including the Karush-Kuhn-Tucker (KKT) conditions or numerical optimization techniques. Quadratic programming is widely used in finance, operations research, and other fields where quadratic relationships are important. In upcoming units we can learn how to solve quadratic programming by different methods.

10.8 *GLOSSARY*

- Graphical method to solve NLPP
- Quadratic programming

10.9 *REFERENCES*

- Mokhtar S. Bazaraa, John J. Jarvis and Hanif D. Sherali: *Linear Programming and Network Flows* (4th edition). John Wiley and Sons, 2010.
- Hamdy A. Taha: *Operations Research: An Introduction (10th edition). Pearson*, 2017.
- Paul R. Thie and Gerard E. Keough: *An Introduction to Linear Programming and Game Theory* (3rd edition), Wiley India Pvt. Ltd, 2014.

- Swarup, K., Gupta, P. K., & Mohan, M. (2017). An introduction to management science operation research. *Sultan Chand & Sons educational publishers, New Delhi*.
- OpenAI. (2024). *ChatGPT (August 2024 version) [Large language model]*. OpenAI. https://www.openai.com/chatgpt

10.11 *SUGGESTED READING*

- G. Hadley, *Linear Programming*, Narosa Publishing House, 2002.
- Frederick S. Hillierand Gerald J. Lieberman: *Introduction to Operations Research* (10th edition). McGraw-Hill Education, 2015.
- https://epgp.inflibnet.ac.in/Home/ViewSubject?catid=ZLCHeZEhCZ8yCri36nSF3A==

10.12 *TERMINAL QUESTION*

Short Answer Type Question:

1: Using the Graphical method solve the following NLPP.

Maximize, $z = x_1 + 2x_2$

Subject to the constraints: $x_1{}^2 + x_2{}^2 \le 1;\ 2x_1 + x_2 \le 2$ and $x_1, x_2 \ge 0$

2: Using the Graphical method solve the following NLPP.

Maximize, $z = -(2x_1 - 5)^2 + 2x_2$

Subject to the constraints: $x_1{}^2 + x_2{}^2 \le 1;\ 2x_1 + x_2 \le 2$ and $x_1, x_2 \ge 0$

3: Using the Graphical method solve the following NLPP.

Maximize, $z = (x_1 - 1)^2 + (x_2 - 2)^2$

Subject to the constraints: $x_1 \le 2, x_2 \le 1; x_1 \ge 0$ and $x_2 \ge 0$.

4: Using the Graphical method solve the following NLPP.

Maximize, $z = (x_1 - 2)^2 + (x_2 - 1)^2$

Subject to the constraints: $-x_1 + x_2 \geq 0, x_1 + x_2 \leq 0;\ x_1 \geq 0$ and $x_2 \geq 0$.

Long answer type question:

1: Using the Graphical method solve the following NLPP.

Maximize, $z = x_1^2 + x_2^2$

Subject to the constraints: $x_1 + x_2 \geq 8, x_1 + x_2 \leq 0;\ x_1 \geq 0$ and $x_2 \geq 0$.

2: Using the Graphical method solve the following NLPP.

Maximize, $z = x_1^2 + x_2^2$

Subject to the constraints: $x_1 + x_2 \leq 8, x_1 + 2x_2 \leq 10;\ 2x_1 + x_2 \leq 10; x_1 \geq 0$ and $x_2 \geq 0$.

3: Using the Graphical method solve the following NLPP.

Maximize, $z = x_1$

Subject to the constraints: $(1 - x_1)^3 - x_2 \geq 0, x_1, x_2 \geq 0$.

Objective type question:

1: In the graphical method of solving a non-linear programming problem, the objective function is typically:

a) Linear

b) Quadratic

c) Polynomial of any degree

d) Any non-linear function

2: When solving a non-linear programming problem graphically, the optimal solution is found at:

a) Any random point in the feasible region
b) The intersection of the constraints
c) The boundary of the feasible region

d) The center of the feasible region

3: In the graphical method, which of the following is true about the feasible region?

a) It is always a convex region.
b) It is defined by the intersection of linear constraints only.
c) It can be a convex or non-convex region depending on the constraints.

d) It is always an unbounded region.

4: In non-linear programming, the feasible region for a graphical solution is often

a) A straight line
b) A curve
c) A triangle

d) A point

5: If the objective function in a non-linear programming problem is concave and the feasible region is convex, the global maximum is:

a) At the center of the feasible region
b) At a corner point of the feasible region
c) At a point where the objective function is tangent to the feasible region

d) Anywhere within the feasible region

6: In the graphical solution of a non-linear programming problem, when the constraints are non-linear, the feasible region is typically:

a) A straight line
b) A polygon
c) A curved shape

d) A circle

7: In the context of graphical solutions to non-linear programming problems, if the objective function is non-linear and constraints are linear, then:

a) The feasible region is linear
b) The objective function can be optimized at the corner points

c) The solution is always at the intersection of the constraints

d) The optimal solution may occur at any point on the boundary or inside the feasible region

True and False:

1: The graphical method of solving non-linear programming problems can handle any number of variables.

2: In a non-linear programming problem, the optimal solution is always at the intersection of the constraints.

3: The feasible region in a non-linear programming problem solved graphically is always a convex shape.

4: The graphical method for non-linear programming problems is effective for visualizing the relationships between the objective function and constraints.

5: In non-linear programming, the objective function can only be quadratic for the graphical method to be applicable.

6: If the objective function is concave and the feasible region is convex, the graphical method will help find the global maximum.

7: The graphical method of solving non-linear programming problems can provide an exact solution if the feasible region and objective function are simple enough.

8: In a graphical non-linear programming problem, contour lines of the objective function can be used to find the optimal solution.

9: The graphical method is useful for solving large-scale non-linear programming problems with many constraints and variables.

10: The graphical method requires that all constraints in a non-linear programming problem be linear.

Fill in the blanks:

1: The graphical method is typically used to solve non-linear programming problems with ____ variables.

2: In the graphical method, the feasible region is the set of all points that satisfy the ____ of the problem.

3: For a non-linear objective function, the optimal solution is usually found on the ____ of the feasible region.

4: The feasible region in non-linear programming may be ____ or ____ depending on the nature of the constraints.

5: When solving a non-linear programming problem graphically, the objective function can be represented by ____ curves, which show different levels of the objective function.

6: In the graphical method, if the feasible region is bounded, the optimal solution will occur either at a ____ point or along a boundary where the objective function is ____ to the constraint.

7: In non-linear programming, if the objective function is convex and the feasible region is convex, then any local minimum is also a ____ minimum.

8: If the objective function in a graphical non-linear programming problem is concave, the optimal solution will typically be at the ____ of the feasible region.

9: The graphical method for non-linear programming is limited to problems with two variables because it allows for visual representation of the ____ and ____ functions.

10: In the graphical solution of non-linear programming problems, the objective function can be linear or non-linear, but the method becomes more complex when both the objective function and the constraints are ____.

10.13 *ANSWERS*

Answer of short answer type question

Answer 1: Total cost $x_1 = 0.6, x_2 = 0.8$; Maximum $z = 2.20$

2: $x_1 = 0, x_2 = 1$; Maximum $z = -25$

3: $x_1 = 0, x_2 = 1$; Maximum $z = 2$

4: $x_1 = 1, x_2 = 1$; Maximum $z = 1$

Answer of long answer type question

Answer 1: $x_1 = 4, x_2 = 4$; Maximum $z = 32$

2: $x_1 = 4, x_2 = 2$ or $x_1 = 2, x_2 = 4$; Maximum $z = 20$

3: $x_1 = 1, x_2 = 0$; Maximum $z = 1$. Constraint qualification is not satisfied.

Answer of Multiple choice question

Answer 1:	d	**2:**	c	**3:**	c	**4:**	a
5:	c	**6:**	c	**7:**	d		

True and False

Answer 1:	False	**2:**	False	**3:**	False
4:	True	**5:**	False	**6:**	True
7:	True	**8:**	True		
9:	False	**10:**	False		

Answer of fill in the blank question

Answer 1:	Two	**2:**	constraints	**3:**	boundary
4:	Convex, Non-convex	**5:**	Contour	**6:**	corner, tangent
7:	global	**8:**	extremity		
9:	objective, constraint	**10:**	non-linear		

BLOCK- IV

DYNAMICAL PROGRAMMING

UNIT-11: ASSIGNMENT PROBLEM

CONTENTS:

11.1 *INTRODUCTION*

The Assignment Problem is a fundamental concept in Linear Programming and Operations Research, where the goal is to assign a set of tasks or jobs to a set of agents or machines in the most efficient way possible. The objective is to minimize the total cost or maximize the overall effectiveness of the assignments, subject to the constraint that each task is assigned to exactly one agent and each agent is assigned exactly one task.

11.2 *OBJECTIVE*

After reading this unit learners will be able to

- Understand the concept of assignment problem.

- Understand the concept of dual of the assignment problem.
- Understand the concept of Hungarian assignment method.

11.3 *MATHEMATICAL FORMULATION OF ASSIGNMENT PROBLEM*

Key Features of the Assignment Problem:

1. **One-to-One Assignment:** Each task must be assigned to one agent, and each agent can handle only one task.
2. **Cost Matrix:** The problem is typically represented by a cost matrix, where each element represents the cost associated with assigning a particular agent to a specific task.
3. **Optimization Objective:** The goal is to find the assignment that results in the minimum total cost (or maximum benefit) while satisfying the assignment constraints.
4. **Balanced Problem:** In a standard assignment problem, the number of tasks equals the number of agents, ensuring a one-to-one match.

Applications:

The Assignment Problem has wide-ranging applications, including job assignment, resource allocation, scheduling, and matching problems in various industries. The problem can be solved efficiently using specialized algorithms such as the Hungarian Method, which guarantees finding the optimal solution in polynomial time. By formulating the Assignment Problem as a Linear Programming Problem (LPP), it becomes easier to solve and analyze, making it a crucial tool in decision-making processes.

The Assignment Problem can be mathematically formulated as a Linear Programming Problem (LPP). Below is the general formulation:

Problem Statement:

- **Given:**
  - n tasks and nnn agents (or jobs and machines).
  - A cost matrix $C = [c_{ij}]$ where c_{ij} represents the cost of assigning task i to agent j.
- **Objective:**
  - Assign each task to exactly one agent and each agent to exactly one task such that the total assignment cost is minimized.

Decision Variables:

Let x_{ij} be a binary decision variable defined as:

$$x_{ij} = \begin{cases} 1 \ if \ task \ i \ is \ assigned \ to \ agent \ j \\ 0 \ otherwise \end{cases}$$

Objective Function:

Minimize the total cost of assignment:

Minimize $Z = \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij}$

Constraints:

1. **Each task is assigned to exactly one agent:**

$$Z = \sum_{j=1}^{n} x_{ij} = 1 \text{ for each } i = 1, 2, ..., n$$

2. **Each agent is assigned to exactly one task:**

$$Z = \sum_{j=1}^{n} x_{j} = 1 \text{ for each } i = 1, 2, ..., n$$

3. **Binary constraint:** $x_{ij} \in \{0,1\} \ \forall i, j$

Interpretation:

- The objective function minimizes the total assignment cost across all possible assignments.
- The first set of constraints ensures that each task is assigned to one and only one agent.
- The second set of constraints ensures that each agent is assigned exactly one task.
- The binary constraint ensures that the decision variable x_{ij} takes on a value of either 0 or 1, indicating whether a task is assigned to an agent or not.

This formulation ensures that the assignment problem is structured as a linear programming problem, making it solvable by standard optimization techniques such as the Hungarian method, simplex method, or specialized integer programming solvers.

Example 1: Given below is an assignment problem, write it as Transportation problem:

	A_1	A_2	A_3
R_1	1	2	3
R_2	4	5	1
R_3	2	1	4

Solution: Let x_{ij} denote the assignment of R_1, R_2, R_3 to A_1, A_2, A_3, such that

$$x_{ij} = \begin{cases} 1 \; if \; R_i \; assigned \; to \; A_j \\ 0 \; otherwise \end{cases}$$

Then the transportation problem is:

Minimize: $1x_{11} + 2x_{12} + 3x_{13} + 4x_{21} + 5x_{22} + x_{23} + 2x_{31} + x_{32} + 4x_{33}$

Subject to the constraints: $\left.\begin{matrix} x_{11} + x_{12} + x_{13} = 1 \\ x_{21} + x_{22} + x_{23} = 1 \\ x_{31} + x_{32} + x_{33} = 1 \end{matrix}\right\}, \quad \left.\begin{matrix} x_{11} + x_{21} + x_{31} = 1 \\ x_{12} + x_{22} + x_{32} = 1 \\ x_{13} + x_{23} + x_{33} = 1 \end{matrix}\right\},$

$x_{ij} = 0 \; or \; 1$ for $i = 1, 2, 3$ and $j = 1, 2, 3$

Theorem 1: (Reduction Theorem) An assignment that minimises the total cost on one matrix also minimises the total cost of the other matrix if we add or subtract a constant from each element of every row (or column) of the cost matrix [a_{ij}]. In other word, $x_{ij} = x_{ij}^*$ minimizes,

$$z = \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij} \text{ with } \sum_{i=1}^{n} x_{ij} = 1, \sum_{j=1}^{n} x_{ij} = 1; x_{ij} = 0 \; or \; 1$$

Then x_{ij}^* also minimizes $z^* = \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij}^* x_{ij}$, where $c_{ij}^* = c_{ij} - u_i - v_j \;\; \forall \, i, j = 1 \; to \; n$ and u_i, v_j are some real number.

Proof: We wijrite, $z^* = \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij}^* x_{ij} = \sum_{i=1}^{n} \sum_{j=1}^{n} (c_{ij} - u_i - v_j) x_{ij}$

$$= \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij} - \sum_{i=1}^{n} u_i \sum_{j=1}^{n} x_{ij} - \sum_{i=1}^{n} x_{ij} \sum_{j=1}^{n} v_j$$

$$= z - \sum_{i=1}^{n} u_i - \sum_{j=1}^{n} v_j; \qquad \left[\text{Since } \sum_{i=1}^{n} x_{ij} = \sum_{j=1}^{n} x_{ij} = 1\right]$$

This demonstrates that since $\sum u_i$ and $\sum v_j$ are independent on, the minimization of the new objective function z^* produces the same solution as the minimisation of the original objective function *z*.

Note: If $c_{ij} \geq 0$, such that minimum $\sum_{i=1}^{n}\sum_{j=1}^{n} c_{ij} x_{ij} = 0$, then the feasible solution x_{ij} provides an optimum assignment.

11.4 *HUNGARIAN ASSIGNMENT METHOD*

The Hungarian Assignment Method, also known as the Hungarian Algorithm, is an efficient combinatorial optimization algorithm that solves the assignment problem in polynomial time. It finds the optimal assignment of tasks to agents (or jobs to machines) to minimize the total cost or maximize the total profit.

Steps of the Hungarian Method:

1. **Construct the Cost Matrix:** Start with a cost matrix $C = [c_{ij}]$, where c_{ij} represents the cost of assigning task i to agent j.
2. **Subtract the Row Minimum:** For each row of the cost matrix, subtract the smallest value in that row from all the entries in the row. This step creates at least one zero in each row.
3. **Subtract the Column Minimum:** After the row reduction, perform column reduction by subtracting the smallest value in each column from all entries in that column. This step creates at least one zero in each column.
4. **Cover All Zeros with a Minimum Number of Lines:** Cover all zeros in the matrix using the minimum number of horizontal and vertical lines. This can be done by:

- Marking each row and column that contains a zero.
- Drawing lines through marked rows and columns.

5. **Test for Optimality:** If the minimum number of lines used to cover all zeros is equal to the number of rows (or columns), an optimal assignment can be made from the matrix. Proceed to step 7.

- If not, continue to step 6.

6. Adjust the Matrix:

- Find the smallest entry not covered by any line.
- Subtract this smallest value from all uncovered elements and add it to the elements at the intersection of the lines.
- Return to step 4 and repeat the process until the number of lines used to cover all zeros is equal to the number of rows (or columns).

7. **Make the Optimal Assignment:** Find the zeros in the matrix and make assignments one by one, ensuring that each task is assigned to one agent, and each agent is assigned to one task. Ensure no two assignments are in the same row or column.
8. **Interpret the Solution:** The resulting assignments indicate the optimal pairings that minimize the total cost (or maximize profit).

Example: Consider a cost matrix for assigning 4 tasks to 4 agents:

$$C = \begin{bmatrix} 9 & 2 & 7 & 8 \\ 6 & 4 & 3 & 7 \\ 5 & 8 & 1 & 8 \\ 7 & 6 & 9 & 4 \end{bmatrix}$$

Following the steps of the Hungarian Method:

1. **Subtract the row minimum:** $C' = \begin{bmatrix} 7 & 0 & 5 & 6 \\ 3 & 1 & 0 & 4 \\ 4 & 7 & 0 & 7 \\ 3 & 2 & 5 & 0 \end{bmatrix}$

2. **Subtract the column minimum:** $C'' = \begin{bmatrix} 4 & 0 & 5 & 6 \\ 0 & 1 & 0 & 4 \\ 1 & 7 & 0 & 7 \\ 0 & 2 & 5 & 0 \end{bmatrix}$

3. **Cover zeros with the minimum number of lines:** Cover zeros using lines. Here, three lines can cover all zeros.
4. **Adjust the matrix (if necessary):** The smallest uncovered value is 1. Subtract 1 from all uncovered elements and add it to elements covered twice.

5. **Repeat until optimality is achieved:** Repeat steps 4 and 5 until four lines cover all zeros.
6. **Make assignments based on zeros:** Assign tasks to agents where zeros appear in the matrix, ensuring one per row and one per column. The assignments will provide the optimal solution, minimizing the total assignment cost.

Advantages:

- **Efficiency:** The Hungarian Method is efficient, with a time complexity of $O(n^3)$ for an $n \times n$ matrix.
- **Optimality:** It guarantees finding the optimal solution for the assignment problem.

The Hungarian Method is widely used in operations research, economics, and computer science to solve various matching and assignment problems.

Example 2: A department head has four responsibilities to complete and four subordinates. Both the fundamental complexity of the duties and the effectiveness of the subordinates vary. The matrix below shows his estimation of how long each man would take to complete each task:

Tasks	*Men*			
	E	*F*	*G*	*H*
A	18	26	17	11
B	13	28	14	26
C	38	19	18	15
D	19	26	24	10

In order to reduce the overall number of man-hours, how should the duties be distributed, one to a man?

Solution: Step 1: Since there are four tasks and four subordinates in this instance, the problem is balanced, and we may go to step 3.

Step 2: Subtracting the smallest element of each row from every element of the corresponding row, we get the reduced matrix:

7	15	6	0
0	15	1	13
23	4	3	0
9	16	14	0

Step 4: Subtracting the smallest element of each column of the reduced matrix from every element of the corresponding column, we get the following reduced matrix:

7	11	5	0
0	11	0	13
23	0	2	0

9	12	13	0

Step 5: We enrectangle ($\square$)(i.e., assign) the first row's zero, if any, and cross ($\times$) every other zero in the designated column. Consequently, we obtain

7	11	5	$\boxed{0}$
$\boxed{0}$	11	$\not{0}$	13
23	$\boxed{0}$	2	$\not{0}$
9	12	13	$\not{0}$

In the above matrix, we arbitrarily enrectangled a zero in column 1, because row 2 had two zeros. It may be noted that column 3 and row 4 do not have any assignment. So, we move on to the next step.

Step 7 (i): Since row 4 does not have any assignment, we mark this row (✓).

(ii) Now there is a zero in the fourth column of the marked row. So, we mark fourth column (✓).

(iii) Further there is an assignment in the first row of the marked column. So we mark first row (✓).

(iv) Draw a line through each of the marked column and unmarked row. This gives us

7	11	5	$\boxed{0}$	√
$\boxed{0}$	11	$\not{0}$	13	
23	$\boxed{0}$	2	$\not{0}$	
9	12	13	$\not{0}$	√
			√	

Step 8: Step 7 showed us that the present assignment is not optimal since the minimal number of lines created in step 3 is fewer than the order of the cost matrix. We create additional zeroes in the altered matrix in order to raise the minimal number of lines. Five is the smallest piece that the lines do not cover. We get the following new reduced cost matrix by subtracting this element from all of the uncovered elements and adding it to all of the elements located at the junction of the lines:

2	6	0	0
0	11	0	18
23	0	2	5
4	7	8	0

Step 9: Repeating step 5 on the reduced matrix, we get

2	6	[0]	~~0~~
[0]	11	~~0~~	18
23	[0]	2	5
4	7	8	[0]

The optimal answer has now been achieved because there is just one assignment in each row and each column. The optimum task is:

$$A \rightarrow G, B \rightarrow E, C \rightarrow F \text{ and } D \rightarrow H$$

The minimum total time for this assignment scheduled is $17+13+19+10$ or 59 man-hours.

Example 3: A pharmaceutical firm manufactures one product, which it distributes through five agencies spread throughout several locations. Suddenly, five more cities without a firm agency are seeing a demand for the product. The organisation must determine how to allocate the current agencies to distribute the goods to underserved cities in a way that minimises the trip distance. The following table provides the distance (in km) between the surplus and deficit cities:

Deficit cities

		a	*b*	*c*	*d*	*e*
	A	85	75	65	125	75
	B	90	78	66	132	78
Surplus cities	*C*	75	66	57	114	69
	D	80	72	60	120	72

E	76	64	56	112	68

Determine the optimum assignment schedule.

Solution: By deducting the smallest element from each row's element and the smallest element from each column's element, we may obtain the reduced distance table as follows:

	a	*b*	*c*	*d*	*e*
A	2	2	0	4	0
B	6	4	0	10	2
C	0	1	0	1	2
D	2	4	0	4	2
E	2	0	0	0	2

Table: 1

In the reduced distance table, we make assignments in rows and coloumns having single zeros and cross off all other zeros is those rows and columns, where assignments have been made. Now draw the minimum number of lines to cover all the zeros. This is done in the following steps :

(*i*) Mark (√) row '*D*' since it has no assignment.

(*ii*) Mark (√) coloumn '*C*' since row '*D*' has zero in this column.

(*iii*) mark (√) row *B* since column '*C*' has an assignment in row '*B*'.

(iv) Since no other rows or column can be marked, draw straight lines through the unmarked row 'A', 'C' and 'E' and marked column 'C' as shown in Table 2.

	a	*b*	*c*	*d*	*e*	
A	2	2	~~0~~	4	[0]	
B	6	4	[0]	10	2	√
C	[0]	1	~~0~~	1	2	
D	2	4	~~0~~	4	2	√
E	2	[0]	~~0~~	~~0~~	2	
			√			

Table: 2

	a	*b*	*c*	*d*	*e*
A	2	2	2	4	0
B	4	2	0	8	0
C	0	1	2	1	2
D	0	2	0	2	0
E	2	0	2	0	2

Table: 3

Modify the reduced distance table (Table 4) by subtracting the smallest element not covered by lines from all the uncovered elements and add the same at the intersection elements of the lines. The modified distance table so obtain is shown in Table 3.

Repeat the above procedure to find the new assignment in Table 4.

	a	b	c	d	e	
A	2	2	2	4	[0]	√
B	4	2	[0]	8	×	√
C	[0]	1	2	1	2	√
D	×	2	×	2	×	√
E	2	[0]	2	×	2	
	√		√		√	

Table: 4

	a	b	c	d	e
A	2	1	2	3	[0]
B	4	1	[0]	7	×
C	×	[0]	2	×	2
D	[0]	1	×	1	×
E	3	×	3	[0]	3

Table: 5

It is also evident that the assignment in Table 4 is not ideal. given that there are only four assignments. We sketch the fewest horizontal and vertical lines necessary to completely enclose each zero in Table 4 in order to arrive at the next solution. Table 5 is obtained by deducting the least uncovered element (i.e., 1) from all uncovered elements and adding the same to the element that represents the junction of two lines.

Table 5 is the updated assignment schedule. Since there are two zeros in rows "*C*" and "*E*" choosing any cell at random from any of these two rows will result in an alternate solution with the same total distance.

An optimal solution is obtained, i.e., the number of assignments equals the order of the supplied matrix.

$A \to e, B \to c, C \to b,\ D \to a,\ E \to d$; or $A \to e, B \to c, C \to d,\ D \to a,\ E \to d$

In both scenarios, the minimum total distance will be 399 km.

11.5 *DUAL OF THE ASSIGNMENT PROBLEM*

The dual of the assignment problem is derived from the primal linear programming formulation of the assignment problem. The assignment problem aims to find the optimal way to assign a set of tasks to a set of agents (or other similar bipartite entities) such that the total cost is minimized (or total profit is maximized).

Primal Problem Formulation

The primal problem is typically formulated as follows:

Minimize: $\sum_{i=1}^{n}\sum_{j=1}^{n} c_{ij} x_{ij}$

where:

- c_{ij} is the cost of assigning task *j* to agent *i*,
- x_{ij} is a binary variable that equals 1 if task *j* is assigned to agent *i*, and 0 otherwise.

Subject to:

1. **Each task is assigned to exactly one agent:** $\sum_{i=1}^{n} x_{ij} = 1 \forall j$
2. **Each agent is assigned to exactly one task:** $\sum_{j=1}^{n} x_{ij} = 1 \forall i$
3. **Non-negativity constraint:** $x_{ij} \geq 0 \forall i, j$

Dual Problem Formulation

To find the dual problem, we introduce dual variables corresponding to the constraints in the primal problem:

- Let u_i be the dual variable associated with the constraint that ensures each task *j* is assigned to exactly one agent.
- Let v_j be the dual variable associated with the constraint that ensures each agent *i* is assigned to exactly one task.

The dual problem is then:i

Maximize:

$$\sum_{i=1}^{n} u_i + \sum_{j=1}^{n} v_j$$

Subject to:

1. $u_i + v_j \leq c_{ij} \forall i, j$

2. No constraint on the signs of u_i and v_j, meaning they can take any real values.

Interpretation of the Dual Problem: The dual variables u_i and v_j can be interpreted as the "prices" or "values" associated with the agents and tasks, respectively. The dual problem seeks to maximize the total value of these prices while ensuring that the sum of the prices for any agent-task pair does not exceed the cost of assigning that task to that agent.

In practical terms, solving the dual problem provides a lower bound on the cost of the optimal assignment in the primal problem. If the primal and dual problems have the same optimal value (which happens under certain conditions, like when the primal has a feasible solution), this optimal valu2e is the minimum cost of the assignment.

Example 4: Write the dual of the following assignment problem and obtain the optimal values of the dual variables:

Programers				
Programers		*A*	*B*	*C*
	1	120	100	80
	2	80	90	110
	3	110	140	120

Solution: The given problem can be written as a transportation problem:

Programmes	Programmers			
	A	B	C	Supply
1	120	100	80	1
2	80	90	110	1
3	110	140	120	1
Demand	1	1	1	3

This T.P. is expressed mathematically as follows:

Minimize $Z = (120x_{11} + 100x_{12} + 80x_{13}) + (80x_{21} + 90x_{22} + 110x_{23}) + (110x_{31} + 140x_{32} + 120x_{33})$

subject to the constraints :

$120x_{11} + 100x_{12} + 80x_{13} = 1,\ 80x_{21} + 90x_{22} + 110x_{23} = 1,\ 110x_{31} + 140x_{32} + 120x_{33} = 1$

$120x_{11} + 80x_{21} + 110x_{31} = 1,\ 100x_{12} + 90x_{22} + 140x_{32} = 1,\ 80x_{13} + 110x_{23} + 120x_{33} = 1$

$$x_{ij} = 1 \text{ or } 0, \text{ for all } i = 1, 2, 3 \text{ and } j = 1, 2, 3.$$

Let $u_i\ (i = 1, 2, 3)$ and $v_j\ (j = 1, 2, 3)$ are the dual variables corresponding to the constraints of the above T.P. Then the dual problem is :

Maximize $z^* = (u_1 + u_2 + u_3) + (v_1 + v_2 + v_3)$ subject to the constraints :

$u_1 + v_1 \le 120,\ u_1 + v_2 \le 100,\ u_1 + v_3 \le 80$

$u_2 + v_1 \le 80,\ u_2 + v_2 \le 90,\ u_2 + v_3 \le 110$

$u_3 + v_1 \le 110,\ u_3 + v_2 \le 140,\ u_3 + v_3 \le 120.$

$u_i \ge 0\ (i = 1, 2, 3)$ and $v_j \ge 0\ (j = 1, 2, 3)$

For the optimum values of the dual variables, an optimum solution to the given problem is to be obtained by transportation method.

Therefore, using VAM for initial basic feasible solution and MODI method for optimality, the optimum solution to the given T.P. is obtained, as displayed in the following transportation table :

				u_i
	120	100	[1] 80	0
	[ϵ] 80	[1] 90	110	10
	[1] 110	140	[ϵ] 120	40
v_j	70	80	80	

We observe that the optimum values of the dual variables are: $u_1 = 0, u_2 = 10, u_3 = 40, v_1 = 70, v_2 = 80$ and $v_3 = 80$. The total of this is 280

Also, Minimum $z = 1\times 80 + \varepsilon \times 80 + 1\times 90 + 1\times 110 + \varepsilon \times 120 = 280 + 200\varepsilon$ as $\varepsilon \to 0$.

This show that minimum $z = \text{maximum}\, z^* = 280$.

Furthermore, it can be seen that the optimal dual variable values are unrestricted in sign and meet all restrictions.

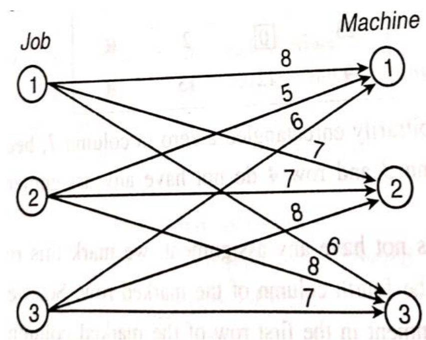
Check your progress

Problem 1: A company wishes to assign 3 jobs to 3 machines in such a way that each job is assigned to some machine and no machine works on more than the job. The cost of assigning job i to machine j is given by the matrix below (ij th) entry:

$$\text{Cost matrix: } \begin{bmatrix} 8 & 7 & 6 \\ 5 & 7 & 8 \\ 6 & 8 & 7 \end{bmatrix}$$

Draw the associated network. Formulate the network LPP and find the minimum cost of making the assignment.

Answer: Total minimum cost will be 19 and the network formulation is:



11.6 *SUMMARY*

The assignment problem is a fundamental optimization problem where the objective is to assign a set of tasks to a set of agents in a way that minimizes the total cost or maximizes the total profit. Each agent can be assigned to exactly one task, and each task must be assigned to exactly one agent, making it a special case of the transportation problem. The problem is often represented by a cost matrix, where the elements denote the cost of assigning a particular task to a specific agent. The Hungarian method is a widely used algorithm for solving the assignment problem efficiently. The problem also has a dual formulation, which provides insights into the pricing structure and allows for the determination of bounds on the optimal solution.

11.7 *GLOSSARY*

- Assignment problem
- Dual of the assignment problem

- Hungarian assignment method

11.8 *REFERENCES*

- Mokhtar S. Bazaraa, John J. Jarvis and Hanif D. Sherali: *Linear Programming and Network Flows* (4th edition). John Wiley and Sons, 2010.
- Hamdy A. Taha: *Operations Research: An Introduction (10th edition). Pearson*, 2017.
- Paul R. Thie and Gerard E. Keough: *An Introduction to Linear Programming and Game Theory* (3rd edition), Wiley India Pvt. Ltd, 2014.
- Swarup, K., Gupta, P. K., & Mohan, M. (2017). An introduction to management science operation research. *Sultan Chand & Sons educational publishers, New Delhi*.
- OpenAI. (2024). *ChatGPT (August 2024 version) [Large language model]*. OpenAI. https://www.openai.com/chatgpt

11.9 *SUGGESTED READING*

- G. Hadley, *Linear Programming*, Narosa Publishing House, 2002.
- Frederick S. Hillierand Gerald J. Lieberman: *Introduction to Operations Research* (10th edition). McGraw-Hill Education, 2015.
- https://epgp.inflibnet.ac.in/Home/ViewSubject?catid=ZLCHeZEhCZ8yCri36nSF3A==

11.10 *TERMINAL QUESTION*

Short Answer Type Question:

1: Consider the problem of assigning five operators to five machines. The assignment costs are given below:

Operators	Machine				
	A	B	C	D	E
I	10	3	10	7	7
II	5	9	7	11	9
III	13	18	2	9	10
IV	15	3	2	7	4
V	16	6	2	12	12

To reduce the overall cost, assign the operators to various equipment.

2: Think about the issue of giving five people five different jobs. The following are the assignment costs:

Jobs

		1	2	3	4	5
	A	8	4	2	6	1
Person	B	0	9	5	5	4
	C	3	8	9	2	6
	D	4	3	1	0	3
	E	9	5	8	9	5

Determine the optimum assignment schedule.

3: There are five persons available to perform five distinct tasks. The following table shows the amount of time (in hours) that each man needs to complete each task based on historical records:

Jobs

		I	II	III	IV	V
	A	2	9	2	7	1
Men	B	6	8	7	6	1
	C	4	6	5	3	1
	D	4	2	7	3	1
	E	5	3	9	5	1

Long answer type question:

1: MCS Inc. is a software business that is working on three Y2K projects with the Maharashtra government's departments of housing, education, and health. The project leaders' performance varies across different projects, depending on their expertise and background. Below is the performance score matrix:

Project Leader	Projects

	Health	Education	Housing
P_1	20	26	42
P_2	24	32	50
P_3	32	34	44

Objective type question:

1: Which of the following is true about the assignment problem?

A) It is a special case of the transportation problem.

B) It is a problem where the objective is to minimize the total assignment cost.

C) Each agent can be assigned to multiple tasks.

D) It always has a feasible solution.

2: In the assignment problem, if there are n tasks and n agents, how many decision variabl x_{ij} will there be?

A) n

B) n^2

C) $2n$

D) $n/2$

3: The Hungarian method is used to solve which of the following types of problems?

A) Transportation problem

B) Assignment problem

C) Linear programming problem

D) Traveling salesman problem

4: In the assignment problem, each agent is assigned to:

A) At least one task

B) At most one task

C) Exactly one task

D) None of the above

Answer: C) Exactly one task

5: Which of the following statements is true about the dual of the assignment problem?

A) The dual problem seeks to minimize costs.

B) The dual problem is always infeasible.

C) The dual problem provides a lower bound on the primal problem's objective value.

D) The dual problem introduces non-negativity constraints on the dual variables.

Fill in the blanks:

1: The assignment problem is a special case of the __________ problem.

2: The objective of the assignment problem is to __________ the total cost of assignments.

3: In the assignment problem, the decision variable x_{ij} is equal to 1 if task *j* is assigned to agent *i*, and __________ otherwise.

4: The __________ method is a commonly used algorithm to solve the assignment problem.

5: In the assignment problem, the total number of assignments is equal to the number of __________.

6: The dual variables in the assignment problem can be interpreted as the __________ associated with the agents and tasks.

7: In the assignment problem, each agent is assigned to exactly __________ task(s).

8: If there are *n* agents and *n* tasks, the assignment problem has __________ decision variables.

9: The dual problem of the assignment problem seeks to __________ the total value of the dual variables.

10: In the assignment problem, the sum of the decision variables for each task must equal __________.

True and False:

1: The assignment problem can be solved using the Hungarian method.

2: In the assignment problem, each agent can be assigned to multiple tasks.

3: The assignment problem is a special case of the linear programming problem.

4: In the assignment problem, the cost matrix must always be square.

5: The objective of the assignment problem is to maximize the total profit of the assignments.

6: In the dual of the assignment problem, the dual variables represent the costs associated with the assignments.

7: The assignment problem can be used to solve matching problems in bipartite graphs.

8: If there are n tasks and n agents, the assignment problem will have n constraints.

9: In the assignment problem, the decision variable x_{ij} is always binary.

10: The dual of the assignment problem always has a higher objective value than the primal problem.

11.11 *ANSWERS*

Answer of short answer type question

Answer 1: $A \to II, B \to I, C \to V, D \to III, E \to IV$ **,** minimum total cost=23

2: $A \to 5, B \to 1, C \to 4, D \to 3, E \to 2$ **,** minimum total cost=9

3: $A \to III, B \to V, C \to IV, D \to I, E \to II, OR\ A \to III, B \to V, C \to I, D \to IV, E \to II$ **,** minimum total time= 55 hours.

Answer of long answer type question

Answer 1: $P_1 \rightarrow Education, P_2 \rightarrow Hou\sin g, P_3 \rightarrow Health;$ Maximum performance=108

Answer of Multiple choice question

Answer 1:	B)	**2:**	B)	**3:**	B)
4:	C)	**5:**	C)		

Answer of fill in the blank question

Answer 1:	transportation	**2:**	minimize	**3:**	0
4:	Hungarian	**5:**	tasks (or agents)	**6:**	prices (or values)
7:	One	**8:**	n^2	**9:**	maximize
10:	1				

Answer of True and False questions

Answer 1:	True	**2:**	False	**3:**	True
4:	True	**5:**	False	**6:**	False
7:	True	**8:**	False	**9:**	True
10:	False				

UNIT-12: DYNAMIC PROGRAMMING

CONTENTS:

12.1 *INTRODUCTION*

American applied mathematician Richard Ernest Bellman (August 26, 1920 – March 19, 1984) is credited with developing dynamic programming in 1953 and making significant advances in biomathematics and other areas of mathematics. He established the Journal of Mathematical Analysis and Applications and the prestigious biomathematical journal Mathematical Biosciences.

Richard Ernest Bellman
26 August-1920 – 19-March-1984

Dynamic programming is a powerful optimization technique used to solve complex problems by breaking them down into simpler sub-problems. Unlike traditional methods that solve a problem in one step, dynamic programming works by solving each sub-problem only once and storing its solution, thus avoiding the need to re-compute the answer every time the sub-problem reappears. This approach is particularly effective for problems with overlapping sub-problems and optimal substructure, meaning the optimal solution to the overall problem can be constructed from the optimal solutions of its sub-problems.

Dynamic programming is widely applicable across various fields, including operations research, computer science, and economics. It is used to solve a range of problems, from shortest path and knapsack problems to more complex scenarios like resource allocation and inventory management. The method works by either adopting a **bottom-up approach**, where solutions are built from the simplest sub-problems up to the main problem, or a **top-down approach**, often implemented with recursion and memoization.

The strength of dynamic programming lies in its ability to significantly reduce the computational effort required for problems that would otherwise be infeasible to solve using straightforward brute-force methods. It is a versatile tool that provides both exact and approximate solutions to optimization problems, making it a fundamental technique in both theoretical and applied contexts.

12.2 *OBJECTIVE*

After reading this unit learners will be able to

- Understand to solve LPP by using dynamic programming.
- Learn the concept of decision-tree analysis.

12.3 *BELLMAN's PRINCIPAL OF OPTIMALITY*

Principle of Optimality

The **Principle of Optimality** is a foundational concept in dynamic programming, introduced by Richard Bellman in the 1950s. It states that:

"An optimal solution to a problem can be constructed from optimal solutions to its sub-problems."

In other words, regardless of how you arrive at a particular stage in the decision-making process, the remaining decisions must constitute an optimal policy with regard to the state resulting from those decisions. This principle underlies the recursive nature of dynamic programming, where a problem is broken down into smaller, overlapping sub-problems that are solved independently.

Key Aspects of the Principle of Optimality:

1. **Optimal Substructure**: The problem can be broken down into sub-problems, and the optimal solution to the problem can be built from the optimal solutions of these sub-problems. This property is crucial for dynamic programming to be applicable.
2. **Overlapping Sub-problems**: Sub-problems recur in the process of solving the main problem. Dynamic programming efficiently solves these sub-problems once and reuses their solutions.
3. **Recursive Solution**: The principle of optimality allows for a recursive approach to solving problems. The problem is solved by solving smaller versions of itself, using the results of these smaller problems to construct the solution to the original problem.

Example 1: Consider the shortest path problem, where you want to find the shortest route from one point to another in a network. According to the principle of optimality, if you know the shortest path from point A to point B passes through point C, then the path from A to C and from C to B must themselves be the shortest paths for those respective segments. This recursive structure allows dynamic programming to solve such problems efficiently.

Importance: The principle of optimality is essential because it justifies the use of dynamic programming in problem-solving. Without this principle, breaking down a problem into smaller sub-problems might not lead to an optimal solution. It ensures that by solving sub-problems optimally, we can guarantee an optimal solution for the entire problem.

12.4 *THE RECURSIVE EQUATION APPROACH*

The recursive equation approach is a fundamental method in dynamic programming used to solve optimization problems by expressing the solution as a recurrence relation. This approach involves breaking down a problem into smaller sub-problems, solving each sub-problem recursively, and then combining these solutions to solve the overall problem.

Key Steps in the Recursive Equation Approach:

1. **Define the Sub-problems:** Identify the smaller sub-problems that the original problem can be decomposed into. These sub-problems should overlap, meaning that the same sub-problems are solved multiple times within different parts of the main problem.
2. **Establish the Recursive Relation:** Formulate a recurrence relation (or recursive equation) that expresses the solution to the original problem in terms of solutions to its sub-problems. This relation is based on the principle of optimality, which ensures that the solution to the main problem can be constructed from optimal solutions to its sub-problems.
3. **Base Case Identification:** Identify the base case(s) that represent the simplest instance(s) of the problem, which can be solved directly without recursion. These base cases terminate the recursion.
4. **Solve Recursively:** Use the recursive relation to solve the problem by working from the base cases up to the original problem. This often involves either a top-down approach with memoization (storing solutions to sub-problems to avoid redundant calculations) or a bottom-up approach, where solutions to the smallest sub-problems are used to build up the solution to the larger problem.
5. **Combine Results:** Combine the solutions to the sub-problems according to the recursive relation to get the solution to the original problem.

Example 2: Consider the **Fibonacci sequence**, where each term is the sum of the two preceding ones, usually starting with 0 and 1. The Fibonacci sequence can be defined recursively as: $F(n) = F(n-1) + F(n-2)$ with base cases: $F(0) = 0;\ F(1) = 1$

This recursive relation allows us to compute the nth Fibonacci number by building up from the known base cases.

Application in Dynamic Programming

In dynamic programming, the recursive equation approach is used to solve more complex problems such as the knapsack problem, shortest path problem, or matrix chain multiplication. By systematically applying the recursive relation, dynamic programming ensures that each sub-problem is solved once, and its result is reused, leading to efficient solutions.

Advantages

- **Efficiency:** By solving each sub-problem only once and storing its solution, the recursive equation approach avoids the exponential time complexity associated with naive recursive methods.

- **Clarity:** The approach provides a clear and structured way to solve complex problems by breaking them down into manageable sub-problems.

Challenges

- **Memory Usage:** Storing solutions to sub-problems can consume significant memory, especially in large-scale problems.
- **Complexity in Formulation:** Deriving the recursive relation and ensuring it correctly captures the problem's structure can be challenging.

Overall, the recursive equation approach is a powerful technique in dynamic programming, enabling the efficient solution of complex optimization problems by leveraging the principle of optimality and the reuse of sub-problem solutions.

12.5 *CHARATERISTICS OF DYNAMIC PROGRAMMING*

For a problem to be suitable for dynamic programming, it typically exhibits the following key characteristics:

Characteristics of Dynamic Programming Problems

1. Optimal Substructure: A problem has an optimal substructure if an optimal solution to the problem can be constructed from optimal solutions to its sub-problems. This characteristic allows the problem to be solved by solving its sub-problems and combining their solutions.

Example 3: In the shortest path problem, the shortest path from one node to another can be found by combining the shortest paths from intermediate nodes.

2. Overlapping Sub-problems: Dynamic programming is particularly effective for problems that involve overlapping sub-problems, meaning the same sub-problems are solved multiple times during the course of solving the problem. By storing the results of these sub-problems (memoization), dynamic programming avoids redundant computations, which leads to a significant reduction in time complexity.

Example 4: In the Fibonacci sequence, the same Fibonacci number is calculated multiple times if computed recursively without memoization.

3. Recursion and Memoization: Dynamic programming problems are often solved using a recursive approach. Memoization involves storing the results of sub-problems so that they do not need to be recomputed when they are encountered again. This is often combined with recursion to build up solutions.

Example 5: In the knapsack problem, the maximum value for a given weight limit is computed recursively, and memoization is used to store the results of sub-problems with specific weight capacities.

4. **State and Decision Variables:** A dynamic programming problem is typically defined by state variables that represent the sub-problem's parameters and decision variables that indicate the choices or actions to be taken at each step. The goal is to find a policy that optimizes the objective function based on these variables.

Example 6: In a dynamic inventory management problem, the state variables could include the current inventory level, while the decision variables could represent the quantity to order.

5. **Stages and Transitions:** Dynamic programming problems can often be broken down into stages, with each stage representing a decision point or step in the problem. Transitions describe how moving from one stage to the next affects the state of the problem.

Example 7: In the stages of the matrix chain multiplication problem, each stage represents multiplying matrices in a specific order, and transitions describe the multiplication process.

6. **Bottom-Up or Top-Down Approach:** Problems can be solved using a bottom-up approach (iterative) or a top-down approach (recursive with memoization). In the bottom-up approach, the solutions to sub-problems are built up from the simplest cases, while in the top-down approach, the problem is broken down recursively.

Example 8: A bottom-up approach is often used in problems like calculating the minimum number of coins needed for a given amount in a coin change problem.

7. **Polynomial-Time Complexity:** Dynamic programming problems typically have a polynomial-time complexity due to the systematic approach of solving and storing solutions to sub-problems. This makes dynamic programming much more efficient than naive methods for many problems.

Example 9: The time complexity for solving the knapsack problem using dynamic programming is $O(nW)$, where n is the number of items and W is the maximum weight capacity.

The characteristics of dynamic programming optimal substructure, overlapping sub-problems, recursion and memoization, state and decision variables, stages and transitions, and the choice between bottom-up or top-down approaches make it a versatile and powerful tool for solving a wide range of optimization problems efficiently.

Solved example

Example 10: Divide the positive quantity *c* into *n* parts in such a way that their products is a maximum.

Or

Maximize, $z = y_1.y_2....y_n$

Subject to the constraints: $y_1 + y_2 + ... + y_n = c$ and $y_i \geq 0$; $j = 1, 2, ..., n$

Solution: Let y_j be the j^{th} part of the positive quantity $c\,(j = 1, 2, ..., n)$, then each j corresponding to part y_j may be regarded as a stage. Now, since y_j may assume any non-negative value satisfying the constraints.

$$y_1 + y_2 + ... + y_n = c,$$

The alternative at each stage are infinite. This means that y_j may be considered to be continuous.

Let $f_n(c)$ denote the maximum attainable product when the quantity c is divided into *n* parts. If we regard c as a fixed quantity and *n* as the number of stages, which varies over positive integers, then a recursive equation connecting $f_n(c)$ and $f_{n-1}(c)$ is,

$$f_n(c) = \max_{0<x\leq c}\{x.f_{n-1}(c-x)\}$$

For, $n = 1$(i.e., one-stage problem), we write

$$f_1(c) = \max_{y_1=c}\{y_1\} = c \text{ (initially true)}$$

For, $n = 2$ (i.e., two-stage problem), the quantity c is divided into two parts, say $y_1 = x$ and $y_2 = c - x$. Then

$f_2(c) = \max\{y_1, y_2\}$ (initially true)

$$= \max_{0<x\leq c}\{x.(c-x)\}$$

$$= \max_{0<x\leq c}\{x.f_1(c-x)\}, \text{ since } f_1(c-x) = c-x$$

Similarly, for $n = 3$, the quantity c is divided into three parts, given the initial choices of x which leaves c-x to be further divided into two parts. Denote the maximum possible product of $(c-x)$ into two parts by $f_2(c-x).$ Then using the principle of optimality, we have

$$f_3(c) = \max_{0<x\leq c}\{x.f_2(c-x)\}$$

Continuing in a similar manner, the recursive equation for general value of x is given by

$$f_n(c) = \max_{0<x\leq c}\{x.f_{n-1}(c-x)\}$$

We now solve the recurrence equation formulated above.

For $n = 2$, the function $x.(c-x)$ attains its maximum value at $x = c/2$ satisfying the condition $0 < x \leq c$. Thus,

$$f_2(c) = \frac{c}{2}\left(c - \frac{c}{2}\right) = \left(\frac{c}{2}\right)^2$$

$\therefore$ The optimal policy is $\left(\frac{c}{2}, \frac{c}{2}\right)$ and $f_2(c) = (c/2)^2$.

For $n = 3$, $f_3(c) = \max_{0<x\leq c}\left\{x.\left(\frac{c-x}{2}\right)^2\right\}$, since $f_2(c-x) = \left(\frac{c-x}{2}\right)^2$

Now, since the maximum value of $x\left(\frac{c-x}{2}\right)^2$ is attained for $x = c/3$ satisfying the condition $0 < x \leq c$; therefore

$$f_3(c) = \left\{ \frac{c}{3} . \frac{1}{4}\left(c - \frac{c}{3} \right)^2 \right\} = \left(\frac{c}{3} \right)^3 .$$

Thus, for $n = 3$, we have

Optimal policy: $\left(\frac{c}{3}, \frac{c}{3}, \frac{c}{3} \right)$ and $f_3(c) = \left(\frac{c}{3} \right)^3$

In general, for $n -$stage problem, we assume that

Optimal policy: $\left(\frac{c}{n}, \frac{c}{n}, ..., \frac{c}{n} \right)$ and $f_n(c) = \left(\frac{c}{n} \right)^n$, for $n = 1,2,..., m$

Now, for n=m+1, the recursive equation is

$$f_{m+1}(c) = \max_{0<x\le c} \{ x . f_m (c - x) \} = \max_{0<x\le c} \left\{ x . \left(\frac{c - x}{m} \right)^m \right\} = \left(\frac{c}{m+!} \right)^{m+1}$$

As the maximum value of $x.\left(\frac{c - x}{m} \right)^m$ is attained at $x = \frac{c}{m+1}$, i.e., the result is also true for $n = m + 1$.

Hence, by mathematical induction, the optimal policy is

$\left(\frac{c}{n}, \frac{c}{n}, ..., \frac{c}{n} \right)$ and $f_n{}^0(c) = \left(\frac{c}{n} \right)^n$

Example 11: Prove with the help of dynamic programming,

$$z = p_1 \log p_1 + p_2 \log p_2 + ... + p_n \log p_n$$

Subject to the constraint,

$p_1 + p_2 + ... + p_n = 1$ and $p_j \ge 0$. $(j = 1, 2, ..., n)$

Is a minimum when $p_1 = p_2 = ... = p_n = 1/n$

Solution: The problem here is to divide unity into n parts as as to minimize the quantity $\sum_i p_i \log p_i$

Let $f_n(1)$ denote the minimum attainable sum of $p_i \log p_i \, (i=1,2,\ldots n)$.

For $n=1$ (stage 1), we have $f_1(1) = \min_{0<x\le 1}\{p_1 \log p_1\} = 1\log 1$

As unity is divided only into $p_1 = 1$ part.

For $n=2$, the unity is divided into two parts p_1 and p_2, such that $p_1 + p_2 = 1$

If $p_1 = x$ and $p_2 = 1-x$, then

$$f_2(1) = \min_{0<x\le 1}\{p_1 \log p_1 + p_2 \log p_2\}$$

$$= \min_{0<x\le 1}\{x\log x + (1-x)\log(1-x)\}$$

$$= \min_{0<x\le 1}\{x\log x + f_1(1-x)\}$$

In general, for an $n-$stage problem, the recursive equation is

$$f_n(1) = \min_{0<x\le 1}\{p_1 \log p_1 + p_2 \log p_2 + ... + p_n \log p_n\}$$

$$= \min_{0<x\le 1}\{x\log x + f_{n-1}(1-x)\}$$

Now, we will solve this recursive relation.

For $n=2$ (stage 2), the function $x\log x + (1-x)\log(1-x)$ attain its minimum value at $x = 1/2$ satisfying the condition $0 < x \le 1$. Thus,

$$f_2(1) = \frac{1}{2}\log\frac{1}{2} + (1-\frac{1}{2})\log(1-\frac{1}{2}) = 2\left(\frac{1}{2}\log\frac{1}{2}\right)$$

Similarly, for stage 3, the minimum value of the recursive equation is obtained as

$$f_3(1) = \min_{0<x\le 1}\{x\log x + f_2(1-x)\}$$

$$= \min_{0<x\le 1}\left\{x\log x + 2\left(\frac{1-x}{2}\right)\log\left(\frac{1-x}{2}\right)\right\}$$

Now, since the minimum value of $x\log x + 2\left(\frac{1-x}{2}\right)\log\left(\frac{1-x}{2}\right)$ is attained at $x = 1/3$ satisfying $x \in (0,1]$, we have

$$f_3(1) = \left\{\frac{1}{3}\log\frac{1}{3} + 2\left(\frac{1}{3}\log\frac{1}{3}\right)\right\} = 3\frac{1}{3}\log\frac{1}{3}$$

$\therefore$ Optimal policy is: $p_1 = p_2 = p_3 = 1/3$

In general, for $n-$stage problem we assume that

$\therefore$ Optimal policy is: $p_1 = p_2 = ... = p_n = 1/n$ and $f_3^{\circ}(1) = n\left\{\frac{1}{n}\log\frac{1}{n}\right\}$

This can be easily using mathematical induction.

For, $n = m+1$, the recursive equation is

$$f_{m+!}(1) = \min_{0<x\le 1}\{x\log x + f_m(1-x)\}$$

$$= \min_{0<x\le 1}\left[x\log x + m\left\{\frac{1-x}{m}\log\left(\frac{1-x}{m}\right)\right\}\right]$$

$$= \frac{1}{m+1}\left\{\frac{1}{m+1}\log\left(\frac{1}{m+1}\right)\right\}$$

Since minimum of $x\log x + \frac{1-x}{m}\log\frac{1-x}{m}$ is attained at $x = \frac{1}{m+1}$

Hence, the required optimal policy is

$\left(\frac{1}{n}, \frac{1}{n}, ..., \frac{1}{n}\right)$ with $f_{n^\circ}(1) = n\left(\frac{1}{n}\log\frac{1}{n}\right)$

12.6 *DECISION-TREE ANALYSIS*

Decision tree analysis is a visual and analytical decision support tool used in decision-making processes. It helps in making decisions by breaking down complex problems into smaller, manageable parts and evaluating the possible outcomes of different decisions. This method is particularly useful when dealing with uncertainties, risks, and rewards.

Components of a Decision Tree

1. **Decision Nodes (Squares):** Represented by squares, these nodes indicate points where a decision needs to be made. Each branch stemming from a decision node represents a possible action or choice.
2. **Chance Nodes (Circles):** Represented by circles, these nodes indicate points where the outcome is uncertain and not under the decision-maker's control. Each branch from a chance node represents a possible outcome, along with its probability of occurrence.
3. **End Nodes (Triangles or Leaves):** Represented by triangles or leaves, these nodes indicate the final outcome or payoff of a particular path through the tree.
4. **Branches:** These are the lines connecting the nodes, representing the flow from decisions to outcomes or from one decision/chance node to another. They are labeled with the possible actions, outcomes, or probabilities.
5. **Payoffs:** At the end of each path, a payoff or value is associated with the outcome. This could be in terms of profit, cost, utility, or any other measure of success.

Steps in Decision Tree Analysis

1. **Identify the Decision:** Define the primary decision to be made and the possible alternatives.
2. **Build the Tree:** Start by drawing the decision node and then branch out for each possible action. For each action, identify the chance events that may occur and their associated probabilities.
3. **Assign Probabilities:** For each chance node, assign probabilities to each possible outcome based on historical data, expert opinion, or statistical analysis.
4. **Estimate Payoffs:** Assign a payoff (or cost) to each end node, reflecting the outcome of following a particular path through the tree.
5. **Evaluate the Tree:** Work backwards from the end nodes to the decision node (this is called "folding back the tree"). For each chance node, calculate the expected value by

multiplying the payoffs by their respective probabilities and summing them. For each decision node, choose the action with the highest expected value.

6. **Make the Decision:** The decision with the highest expected value is the optimal choice based on the analysis.

Example of Decision Tree Analysis

Consider a business decision where a company must decide whether to launch a new product. The decision tree might look like this:

1. **Decision Node:** Launch Product or Not Launch Product.
2. **Chance Node (if Product is Launched):**
   - High Demand (70% probability)
   - Low Demand (30% probability)
3. **Payoffs:**
   - High Demand: \$500,000 profit
   - Low Demand: \$100,000 loss
   - Not Launching: \$0 profit/loss

Evaluation

- **High Demand Expected Value:** 0.7×500,000=350,0000.7 \times 500,000 =

  350,0000.7×500,000=350,000

- **Low Demand Expected Value:** 0.3×−100,000=−30,0000.3 \times -100,000 =

  -30,0000.3×−100,000= −30,000

- **Total Expected Value of Launching:** 350,000−30,000=320,000350,000 - 30,000

  = 320,000350,000−30,000=320,000

Since launching the product has an expected value of \$320,000, which is greater than not launching (\$0), the decision would be to launch the product.

Advantages of Decision Tree Analysis

- **Clarity:** Provides a clear and visual representation of decisions, outcomes, and associated risks.

- **Structured Approach:** Helps break down complex decisions into smaller, more manageable components.
- **Quantitative Analysis:** Allows for the calculation of expected values and probabilities, making the decision-making process more objective.
- **Flexibility:** Can be used for a wide range of decisions, from simple choices to complex, multi-stage problems.

Limitations

- **Complexity:** For decisions with many variables or stages, the decision tree can become very large and complex.
- **Assumptions:** The accuracy of the decision tree depends on the accuracy of the probabilities and payoffs assigned, which are often based on estimates.
- **Overfitting:** Decision trees may overfit the data, particularly if the tree is too detailed or specific to past events.

Decision tree analysis is a powerful tool for making informed decisions in the face of uncertainty. By visually mapping out the possible choices, outcomes, and their associated risks and rewards, it allows decision-makers to identify the best course of action systematically.

A decision-tree is a graphic display of various decision alternatives and the sequence of events as if they were branches of tree. In constructing a tree diagram, it is a convention to use the symbol "□" to indicate the decision point and "○" to denote the situation of uncertainty or event. Branches coming out of a decision point are nothing but representation of immediate mutually exclusive alternative act (alternative options) open to the decision maker. Branches emanating from the 'event' point "○" represent all possible situations (events). These events are not fully under the control of the decision-maker and may represent consumer demand, etc. The basic advantage of a tree diagram is that another act (called second act) subsequent to the happening of each event may also be represented. The resulting outcome (payoff) for each act-event combination may be indicated in the tree diagram at the outer end of each branch. A decision-tree diagram is illustrated below:

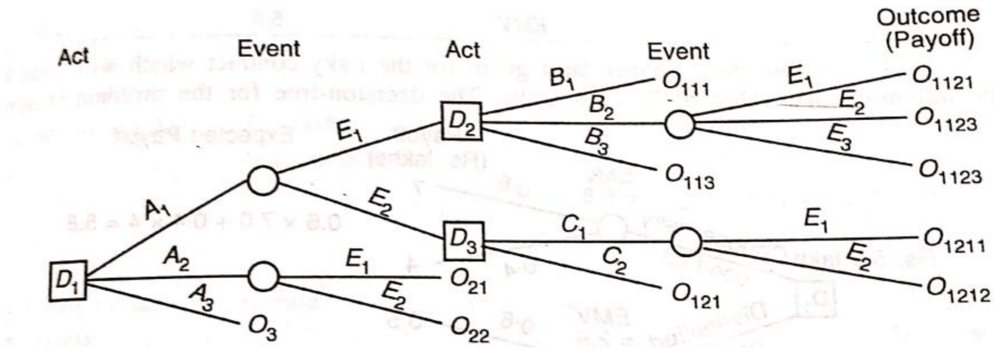


Figure 1: A decision tree diagram

12.7 *EXPECTED MONETARY VALUE (EMV) CRITERION*

The Expected Monetary Value (EMV) is a decision-making criterion used in decision analysis, particularly in scenarios involving uncertainty. It is used to determine the best course of action by calculating the average outcome when the future includes scenarios that may or may not happen. EMV is especially useful in situations where decisions involve risks and different possible outcomes with associated probabilities.

Key Concepts of EMV Criteria

1. **Monetary Value:** This is the financial value associated with each possible outcome of a decision. It can represent profit, revenue, cost, or any other quantifiable monetary outcome.
2. **Probability:** Each possible outcome is associated with a probability, which is the likelihood of that outcome occurring. The sum of the probabilities for all outcomes of a particular decision should equal 1.
3. **Expected Value:** The EMV for a decision is calculated as the sum of the products of each outcome's monetary value and its probability. In other words, EMV is the weighted average of all possible outcomes, where the weights are the probabilities.

EMV Formula

The formula to calculate the EMV for a decision with several possible outcomes is:

$$\text{EMV} = \sum (\Pr obability\ of\ outcome_i \times Monetary\ value\ of\ outcome_i)$$

Where:

- i represents each possible outcome of the decision.
- The probability of each outcome is the likelihood that the outcome will occur.

- The monetary value of each outcome is the financial impact if that outcome occurs.

Steps to Apply EMV Criteria

1. **List All Possible Outcomes:** Identify all the possible outcomes or scenarios for each decision alternative.
2. **Assign Probabilities:** Assign a probability to each outcome. Ensure that the sum of probabilities for all outcomes related to a decision equals 1.
3. **Determine Monetary Values:** Assign a monetary value to each outcome, reflecting the financial impact if that outcome occurs.
4. **Calculate EMV for Each Decision Alternative:** Multiply each outcome's monetary value by its probability, and sum these values to get the EMV for each decision alternative.
5. **Compare EMVs:** Compare the EMVs of all decision alternatives. The decision with the highest EMV is generally considered the best choice, as it maximizes the expected monetary gain.

Example 12: A riskier contract offering Rs. 7 lakhs with probability 0.6 and 4 lakhs with probability 0.4 is one option available to a manager. The other is a diversified portfolio made up of two contracts with independent outcomes, each offering Rs. 3.5 lakhs with probability 0.6 and Rs. 2 lakhs with probability 0.4.
Build a decision tree with the EMV criterion in mind. Can you use the EMV criteria to make the decision?

Solution: The problem's conditional payout table might be created like this:

Event	Prabability	Conditional payoffs (Decision)		Expected payoffs (Decision)	
E_i	E_1	Contract	Portfolio	Contract	Portfolio
	(i)	(ii)	(iii)	$(i)\times(ii)$	$(i)\times(iii)$
E_1	0.6	7	3.5	4.2	2.1
E_2	0.4	4	2	1.6	0.8
			EMV	**5.8**	2.0

Using the EMV criterion, the manager must go in for the risky contract which will yield him a higher expected monetary valur of Rs. 5.8 lakhs. The decision-tree for the problem is given if fig. 2.

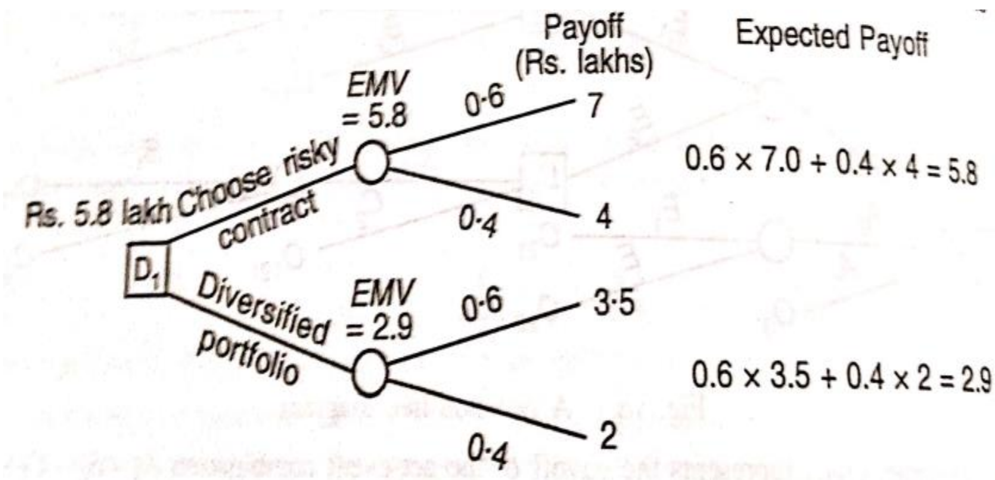


Figure2: Decision tree for the manager

Example 13: Amar company is currently working with a process which after paying for materials, labour, etc., brings a profit of Rs. 12000. The following alternatives are made available to the company:

(i) The company can conduct research (R1) which is expected to cost Rs. 10000 having 90% chances of success. If it proves a success, the company gets a gross income of Rs. 25000.

(ii) The company can conduct research (R2) which is expected to cost Rs. 8000 having a probability of 60% success, the gross income will be Rs. 25000.

(iii) The company can pay Rs. 6000 as royalty for a new process which will bring a gross income of Rs. 20,000.

(iv) The company continues the current process.

Because of limited resources, it is assumed that only one of the two types of research can be carried out at a time.

Use decision –tree analysis to locate the optimal strategy for the company.

Answer: The various act-event combination and resulting playoffs of the problem are introduced

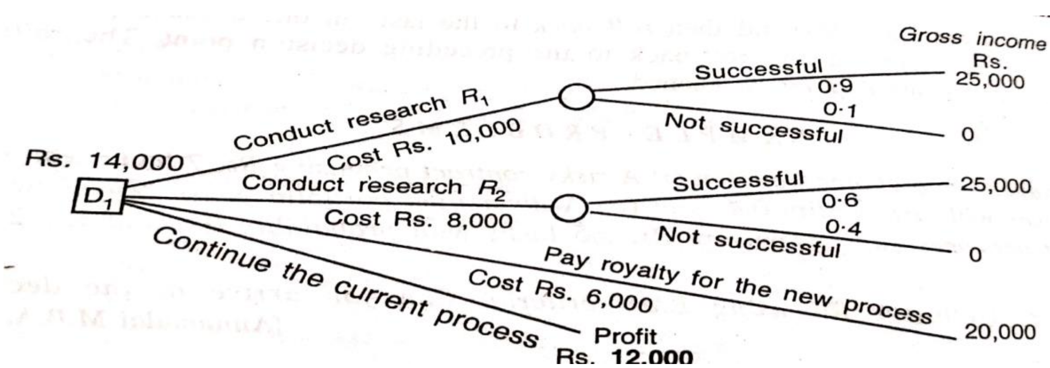


Figure 3: Amar company's decision-tree

in the decision-tree diagram (fig. 3). The net EMV corresponding to various event/decision points are indicated in bold type in fig. 3.

Decision Analysis at Point D

Decision	Event	Probability	Gross income	Expected income
1. Conduct research R_1	Successful	0.9	(Rs.) 25,000	(Rs.) 22,500
	Not successful	0.1	0	0
				Total expected income = Rs. 22,500
				Less cost Rs. 10,000
				Net *EMV* Rs. 12,500
2. Conduct research R_2	Successful	0.6	25,000	15,000
	Not successful	0.4	0	0
				Total expected income = Rs. 15,000
				Less cost Rs. 8,000
				Net *EMV* Rs. 7,000
3. Pay royalty for the new process	Certain	1	20,000	Rs. 20,000
				Less cost Rs. 6,000
				Net *EMV* Rs. 14,000
4. Continue the current process	Certain	1	12,000	*EMV* Rs. 12,000

As the net EMV is highest for the alternative 'pay royalty for the new process', the optimal decision would be to procure new process on royalty basis.

Check your progress

Problem 1: Solve the following LPP by using dynamic programming.

Minimize, $z = x_1^2 + x_2^2 + x_3^2$

Subject to the constraints: $x_1 + x_2 + x_3 \geq 15$ and $x_1, x_2, x_3 \geq 0$

Answer: The optimal solution is $(5,5,5)$ with $f_3^0(15) = 75$

12.8 *SUMMARY*

In this unit we have learned that, dynamic programming is a powerful algorithmic technique used to solve optimization problems by breaking them down into simpler subproblems. It is particularly effective for problems that exhibit overlapping subproblems and optimal substructure properties. Instead of solving the same subproblems repeatedly, dynamic programming stores the results of subproblems in a table, known as memoization or tabulation, to avoid redundant calculations. This approach can be applied using either a top-down (recursive) or bottom-up (iterative) method, and is commonly used in problems like the knapsack problem, shortest path algorithms, and sequence alignment. By efficiently reusing the solutions to subproblems, dynamic programming significantly reduces the computational complexity of these problems.

Also we have learned that, decision tree analysis is a visual and analytical tool used for decision-making under uncertainty. It involves creating a tree-like model of decisions, where each branch represents a possible choice or chance event, leading to different outcomes. The analysis begins with a root node, where a decision is made, and branches out to include decision nodes, chance nodes, and terminal nodes that represent the final outcomes. Decision trees help evaluate the possible consequences of different decisions by calculating expected values, allowing decision-makers to choose the most optimal path based on criteria like expected monetary value (EMV). This method is widely used in business, finance, and risk management to simplify complex decision-making processes.

12.9 *GLOSSARY*

- Dynamic Programming
- Decision tree analysis
- Bellman's principal of optimality
- Expected monetary value (EMV) criterion

12.10 *REFERENCES*

- Mokhtar S. Bazaraa, John J. Jarvis and Hanif D. Sherali: *Linear Programming and Network Flows* (4th edition). John Wiley and Sons, 2010.
- Hamdy A. Taha: *Operations Research: An Introduction (10th edition). Pearson*, 2017.
- Paul R. Thie and Gerard E. Keough: *An Introduction to Linear Programming and Game Theory* (3rd edition), Wiley India Pvt. Ltd, 2014.
- Swarup, K., Gupta, P. K., & Mohan, M. (2017). An introduction to management science operation research. *Sultan Chand & Sons educational publishers, New Delhi*.
- OpenAI. (2024). *ChatGPT (August 2024 version) [Large language model]*. OpenAI. https://www.openai.com/chatgpt

12.11 *SUGGESTED READING*

- G. Hadley, *Linear Programming*, Narosa Publishing House, 2002.
- Frederick S. Hillierand Gerald J. Lieberman: *Introduction to Operations Research* (10th edition). McGraw-Hill Education, 2015.

- https://epgp.inflibnet.ac.in/Home/ViewSubject?catid=ZLCHeZEhCZ8yCri36nSF3A==

12.12 *TERMINAL QUESTION*

Short Answer Type Question:

1: Solve the following LPP by using dynamic programming.

Maximum, $z = x_1 . x_2 . x_3$

Subject to the constraints: $x_1 + x_2 + x_3 = 5$ and $x_1, x_2, x_3 \geq 0$

2: Solve the following LPP by using dynamic programming.

Minimum, $z = x_1 + x_2 + x_3 + ... + x_n$

Subject to the constraints: $x_1 . x_2 . x_3 .... x_n = d$ and $x_j \geq 0$ for $j = 1$ *to* n

3: Solve the following LPP by using principal of optimality.

Maximum, $b_1 x_1 + b_2 x_2 + b_3 x_3 + ... + b_n x_n$

when: $x_1, x_2, x_3, ..., x_n \geq 0$ and $x_j \geq 0$ for $j = 1$ *to* n

4: Solve the following LPP by using dynamic programming.

Minimum, $z = x_1^2 + x_2^2 + x_3^2 + ... + x_n^2$

Subject to the constraints: $x_1 . x_2 . x_3 .... x_n = c$ and $x_j \geq 0$ for $j = 1$ *to* n

5: Solve the following LPP by using dynamic programming.

Maximize, $z = x_1^2 + x_2^2 + x_3^2 + ... + x_{10}^2$

Subject to the constraints: $x_1 . x_2 . x_3 .... x_{10} = 8$ and $x_j \geq 0$ for $j = 1$ *to* 10

6: Solve the following LPP by using dynamic programming.

Minimize, $z = x_1^3 + x_2^3 + x_3^3$

Subject to the constraints: $x_1 . x_2 . x_3 = 27$ and $x_j \geq 0$ for $j = 1$ *to* 3

7: Solve the following LPP by using dynamic programming.

Minimize, $z = x_1^2 + x_2^2 + x_3^2$

Subject to the constraints: $x_1 + x_2 + x_3 = 10$

(a) x_j, for $j = 1$ *to* 3 are non-negative

(b) x_j, for $j = 1$ *to* 3 are non-negative integers.

8: Solve the following LPP by using dynamic programming.

Minimize, $z = x_1^2 + 2x_2^2 + 4x_3$

Subject to the constraints: $x_1 + 2x_2 + x_3 \geq 8$; $x_1, x_2, x_3 \geq 0$

9: Solve the following LPP by using dynamic programming.

Maximum, $z = x_1^2 + 2x_2^2 + 4x_3$

Subject to the constraints: $x_1 + 2x_2 + x_3 \geq 8$; $x_1, x_2, x_3 \geq 0$

10: Solve the following LPP by using dynamic programming.

Maximum, $z = -x_1^2 - 2x_2^2 + 3x_2 + x_3$

Subject to the constraints: $x_1 + 2x_2 + x_3 \leq 1$; $x_1, x_2, x_3 \geq 0$

Long answer type question:

1: Recently, a particular region has been granted rights by an oil corporation to carry out surveys and test drillings, with the ultimate goal of lifting oil in the event that it is discovered in economically exploitable amounts.
It is thought that there is a strong chance of discovering oil in commercial quantities in the area. The corporation might choose to launch a drilling program right now or to carry out more geological testing at the oustest. The business estimates that there is a 70:30 possibility of successful additional testing under the current conditions.

The corporation has two options: it either continue with its drilling program or sell the license to drill in the region, depending on whether the tests indicate a chance of final success or not. But after that, when it executes the drilling program, the possibility of ultimate success or failure is thought to rely on the earlier phases. Thus,

(i) If testing have proven effective, the drilling success rate is predicted to be 80:20
(ii) In the event that the test results show failure, the drilling success rate is 20:80.
(iii) If no testing have been conducted at all, the probability of drilling success is 55:45.

The following is the net present value of each cost and revenue, which have been assessed for every scenario that might occur.

Outcome		*Net present value (Rs. million)*
Success:	With prior tests	100
	Without prior tests	120
Failure:	With prior tests	–50
	Without prior tests	–40
Sale of exploitation rights:	Prior tests show 'success'	65
	Prior tests show 'failure'	15
	Without prior tests	45

(a) Create a decision (probability) tree diagram to illustrate the data above: and
(b) Assess the tree to recommend the best course of action to the company's management.

Objective type question:

1: Which of the following problems is typically solved using dynamic programming?

A) Sorting a list of numbers

B) Finding the shortest path in a graph

C) Calculating the nth Fibonacci number

D) Searching for an element in a sorted array

2: Dynamic programming is particularly useful when the problem has which of the following properties?

A) No overlapping subproblems

B) Independent subproblems

C) Overlapping subproblems and optimal substructure

D) Only dependent subproblems

3: Which of the following is an example of a problem that can be solved using dynamic programming?

A) Merge Sort

B) Dijkstra's Algorithm

C) Knapsack Problem

D) Binary Search

4: The time complexity of dynamic programming algorithms generally depends on:

A) The number of states and the transition time between states

B) The height of the tree

C) The depth of recursion

D) The sorting order of input data

5: Which of the following best describes the concept of memoization in dynamic programming?

A) It is a technique where the solution to a problem is built from solutions to subproblems.

B) It is a method of solving problems by breaking them down into smaller problems.

C) It is the process of storing the results of expensive function calls and reusing them when the same inputs occur again.

D) It is a way of sorting data to optimize the search process.

6: In dynamic programming, what does "optimal substructure" refer to?

A) The problem can be broken down into independent subproblems.

B) The problem's solution can be recursively built from solutions to its subproblems.

C) Each subproblem has exactly one solution.

D) The subproblems do not overlap in any way.

7: Which of the following is not a step in the process of solving a problem using dynamic programming?

A) Characterize the structure of an optimal solution.

B) Define the value of an optimal solution recursively.

C) Compute the value of an optimal solution using a top-down or bottom-up approach.

D) Sort all elements to arrange them in a particular order.

8: The Bellman-Ford algorithm, which finds the shortest path in a graph, is an example of:

A) Greedy algorithm

B) Dynamic programming

C) Divide and conquer

D) Backtracking

9: Which of the following is true about a decision tree?

A) It is a linear model used for classification tasks.

B) It always results in a binary outcome.

C) It visually represents decisions and their possible consequences, including chance event outcomes.

D) It is only used for regression problems.

10: In a decision tree, what does a decision node represent?

A) The end result of a decision

B) A point where a decision must be made

C) An uncertain event with multiple possible outcomes

D) A measure of the accuracy of the decision tree

11: Which of the following components of a decision tree represents a chance event?

A) Root node

B) Decision node

C) Chance node

D) Leaf node

12: What is the primary goal of using a decision tree in decision analysis?

A) To maximize the expected utility of a decision

B) To create a simple model for linear relationships

C) To minimize the computational complexity of decision-making

D) To identify all possible outcomes of a decision

13: Which of the following is not a typical use case for decision tree analysis?

A) Risk assessment

B) Project management

C) Time series forecasting

D) Strategic planning

14: In decision tree analysis, the process of "folding back" the tree refers to:

A) Simplifying the tree by combining similar nodes

B) Calculating the expected value at each chance node by working from the end nodes backward

C) Recalculating the probabilities of outcomes

D) Adjusting the tree based on new data or outcomes

15: Which of the following best describes the concept of "expected monetary value" (EMV) in decision tree analysis?

A) The sum of all possible outcomes

B) The average value of all outcomes weighted by their probabilities

C) The highest possible payoff in the decision tree

D) The lowest possible loss in the decision tree

True and False:

1: The graphical method of solving non-linear programming problems can handle any number of variables.

2: In a non-linear programming problem, the optimal solution is always at the intersection of the constraints.

3: The feasible region in a non-linear programming problem solved graphically is always a convex shape.

4: The graphical method for non-linear programming problems is effective for visualizing the relationships between the objective function and constraints.

5: In non-linear programming, the objective function can only be quadratic for the graphical method to be applicable.

6: If the objective function is concave and the feasible region is convex, the graphical method will help find the global maximum.

7: The graphical method of solving non-linear programming problems can provide an exact solution if the feasible region and objective function are simple enough.

8: In a graphical non-linear programming problem, contour lines of the objective function can be used to find the optimal solution.

9: The graphical method is useful for solving large-scale non-linear programming problems with many constraints and variables.

10: The graphical method requires that all constraints in a non-linear programming problem be linear.

Fill in the blanks:

1: In dynamic programming, the technique of storing previously computed results to avoid redundant calculations is called ___________.

2: Dynamic programming is particularly useful for problems that exhibit ___________ and ___________ properties.

3: The __________ approach in dynamic programming builds the solution by solving smaller subproblems first and using their solutions to solve larger problems.

4: In dynamic programming, a __________ is a table used to store the results of subproblems to avoid recalculating them.

5: The Bellman equation is used to express the __________ in dynamic programming.

6: Dynamic programming is often used to solve optimization problems, such as the __________ problem, where the goal is to maximize or minimize a certain objective.

7: A key difference between dynamic programming and divide-and-conquer is that dynamic programming __________ subproblems, while divide-and-conquer __________ them.

8: The __________ method is a classic dynamic programming approach for finding the shortest path in a weighted graph with possible negative weights.

9: In the context of dynamic programming, the term __________ refers to the structure of the problem that allows a solution to be constructed efficiently from the solutions to subproblems.

10: The __________ algorithm is an example of a dynamic programming technique used to find the longest common subsequence between two sequences.

12.13 *ANSWERS*

Answer of short answer type question

Answer 1: $x_1 = 5/3, x_2 = 5/3, x_3 = 5/3$; Maximum $z = 125/27$

2: $x_1 = d^{1/n}, x_2 = d^{1/n}, x_3 = d^{1/n}$; Minimum $z = nd^{1/n}$

3: $x_1 = 0, x_2 = 0, ..., x_{n-1} = 0, x_n = c$; Minimum $z = b_n c$

4: $x_1 = c^{1/n}, x_2 = c^{1/n}, .., x_n = c^{1/n}$; Minimum $z = nc^{2/n}$

5: $x_1 = c^{1/10}, x_2 = c^{1/10}, .., x_{10} = c^{1/10}$; Minimum $z = 10c^{1/5}$

6: $x_1 = 10/3, x_2 = 10/3, x_3 = 10/3$; Minimum $z = 100/3$

7: $x_1 = 3, x_2 = 4, x_3 = 3$ or $x_1 = 3, x_2 = 3, x_3 = 4$ or $x_1 = 4, x_2 = 3, x_3 = 3$; Minimum Z=34

8: $x_1 = 2, x_2 = 2, x_3 = 2$; Minimum Z=20

9: $x_1 = 8, x_2 = 0, x_3 = 0$; Minimum Z=64

10: $x_1 = 0, x_2 = 1/2, x_3 = 1/2$; Minimum Z=3/2

Answer of Multiple choice question

Answer 1:	(C)	**2:**	(C)	**3:**	(C)
4:	(A)	**5:**	(C)	**6:**	(B)
7:	(D)	**8:**	(B)	**9:**	(C)
10:	(B)	**11:**	(C)	**12:**	(A)
13:	(C)	**14:**	(B)	**15:**	(B)

Answer of fill in the blank question

Answer 1: memorization **2:** overlapping subproblems, optimal substructure

3: bottom-up **4:** DP table (or dynamic programming table)

5: recursive relationship **6:** knapsack

7: reuses, solves independently **8:** Bellman-Ford

9: optimal substructure

10: LCS (Longest Common Subsequence)

UNIT-13: WOLFE'S MODIFIED SIMPLEX METHODS & BELL'S METHOD

CONTENTS:

13.1 *INTRODUCTION*

Wolfe's Modified Simplex Method is an extension of the traditional simplex algorithm, designed specifically to solve quadratic programming problems where the objective function is quadratic, and the constraints are linear. This method integrates quadratic programming techniques with the simplex method by transforming the quadratic problem into a series of linear programming subproblems. Wolfe's approach is particularly useful for handling problems with a convex objective function, ensuring convergence to an optimal solution. The modified simplex method systematically explores the feasible region, leveraging the structure of the quadratic function to efficiently find the global minimum while maintaining the simplicity and interpretability of the simplex algorithm.

Beale's Modified Simplex Method is a specialized algorithm developed to solve quadratic programming problems, where the objective function is quadratic and the constraints are linear.

Similar to Wolfe's method, Beale's approach extends the traditional simplex method used in linear programming by adapting it to handle the quadratic nature of the problem. The method operates by iteratively solving linear approximations of the quadratic objective function within a feasible region defined by the constraints. Beale's modification aims to maintain the efficiency and simplicity of the simplex method while effectively addressing the additional complexity introduced by the quadratic terms, making it suitable for a wide range of optimization problems in operations research and economics.

13.2 *OBJECTIVE*

After reading this unit learners will be able to

- Understand the concept of Wolfe's method to solve QPP
- Understand the concept of Beale's method to solve QPP.

13.3 *WOLFE'S MODIFIED SIMPLEX METHODS*

Wolfe's Modified Simplex Method is a specialized algorithm designed to solve quadratic programming problems, where the objective function is quadratic, and the constraints are linear. This method is an extension of the simplex method, traditionally used for linear programming, but adapted to handle the complexities introduced by a quadratic objective function.

Key Features of Wolfe's Modified Simplex Method:

1. **Quadratic Programming Context:** Wolfe's method is used when the optimization problem involves a quadratic objective function (typically of the form $\frac{1}{2}x^T Qx + c^T x$, where Q is a symmetric matrix) and linear constraints.
2. **Transformation into Linear Subproblems:** The method transforms the quadratic problem into a series of linear programming subproblems. It does this by using the Kuhn-Tucker conditions (also known as the Karush-Kuhn-Tucker, or KKT conditions), which are necessary and sufficient for the optimality of quadratic programming problems under certain conditions.
3. **Iterative Process:** The method starts by solving an initial linear approximation of the quadratic problem using the simplex method. It then iteratively refines the solution, adjusting the linear constraints to better approximate the quadratic objective function in each iteration.

4. **Handling Convexity:** Wolfe's method is particularly efficient for convex quadratic programming problems, where the matrix Q in the objective function is positive semi-definite. This ensures that the objective function has a single global minimum.
5. **Convergence:** The method guarantees convergence to an optimal solution by systematically moving through the feasible region defined by the constraints, using the structure of the quadratic function to guide the search.

Applications:

Wolfe's Modified Simplex Method is widely used in fields such as finance, engineering, and economics, where quadratic programming problems frequently arise, such as in portfolio optimization, production planning, and resource allocation.

In summary, Wolfe's Modified Simplex Method provides an effective way to solve quadratic programming problems by leveraging the simplicity of the simplex method while adapting it to handle the complexities of quadratic optimization.

The following is a summary of the iterative process used by Wolfe's modified simplex approach to solve a quadratic programming problem:

Step 1: By adding the slack variables S_i^2 in the i^{th} constraint, $i = 1, 2, ..., m$ and S_{m+j}^2 in the j^{th} non-negative constraint, $j = 1, 2, ..., n$ you may turn the inequality constraints into an equation.

Step 2: Construct the Lagrangian function

$$L(x, s, \lambda) = f(x) - \sum_{i=1}^{m} \lambda_i \left[\sum_{j=1}^{n} a_{ij} x_j - b_i + S_i^2 \right] - \sum_{j=1}^{m} \lambda_{m+j} \left(-x_j + S_{m+j}^2 \right)$$

Where $x = (x_1, x_2, ..., x_n), S = (S_1, S_2, ..., S_{m+n}), \lambda = (\lambda_1, \lambda_2, ..., \lambda_{m+n})$

Partially differentiate $L(x, s, \lambda)$ with regard to the x, s and λ components, and then set the first order partial derivatives to zero. From the resultant equations, derive the Kuhn-Tucker conditions.

Step 3: Introduce the non-negative artificial variable $A_j, j = 1, 2, ..., n$ in the Kuhn-Tucker condition

$$c_j + \sum_{k=1}^{n} d_{jk} x_k - \sum_{k=1}^{n} \lambda_i a_{ij} + \lambda_{m+j} = 0$$

for $j = 1, 2, ..., n$ and construct an objective function

$z = A_1 + A_2 + ... + A_n$

Step 4: Obtain an initial basic feasible solution to the LPP

Minimize, $z = A_1 + A_2 + ... + A_n$

Subject to the constraint:

$$\sum_{k=1}^{n} d_{jk} x_k - \sum_{k=1}^{n} \lambda_i a_{ij} + \lambda_{m+j} + A_j = -c_j \qquad (j = 1 \; to \; n)$$

$$\sum_{j=1}^{n} a_{ij} x_j + x_{n+i} = b_i \qquad (i = 1 \; to \; m)$$

$$A_j, \lambda_i, \lambda_{m+j}, x_j \geq 0 \qquad (i = 1 \; to \; m; \; j = 1, 2, ..., n)$$

Where $x_{n+1} = S_i^{\;2}, i = 1, 2, ..., m$ and satisfying the complementary slackness conditions

$$\sum_{j=1}^{n} \lambda_{m+j} x_j + \sum_{j=1}^{n} x_{n+i} \lambda_i = 0$$

Step 5: To find the best solution to the LPP of Step 4, employ the two-phase simplex approach. This solution must meet the complementary slackness criteria.

Step 6: Step 5's optimal solution also happens to be the *QPP*'s optimal solution.

Note: If the provided QPP is in the minimisation form, modify $f(x_1, x_2, ..., x_n)$ appropriately to change it to the maximisation form.

Example 1: Solve the following QPP by using Wolfe's method.

Solution: First we introduce the slack variable S_1^2 and S_2^2 to convert the inequalities into equalities. We transform $x_1 \geq 0$ and $x_2 \geq 0$ into equations by adding the slack variables S_3^2 and S_4^2 and treating them as inequality constraints as well. The problem thus becomes:

Maximize, $z = 2x_1 + 3x_2 - 2x_1^{\;2}$

Subject to the constraint: $x_1 + 4x_2 + S_1{}^2 = 4; x_1 + x_2 + S_2{}^2 = 2; -x_1 + S_3{}^2 = 0; -x_2 + S_4{}^2 = 0$

Construct the Lagrangian function,

$$L = L(x_1, x_2, S_1, S_2, S_3, S_4, \lambda_1, \lambda_2, \lambda_3, \lambda_4)$$

$$= (2x_1 + 3x_2 - 2x_1{}^2) - \lambda_1(x_1 + 4x_2 + S_1{}^2 - 4) - \lambda_2(x_1 + x_2 + S_2{}^2 - 2) - \lambda_3(-x_1 + S_3{}^2) - \lambda_4(-x_2 + S_4{}^2)$$

As $-x_1{}^2$ represents a negative semi-definite quadratic form $z = 2x_1 + 3x_2 - 2x_1{}^2$ is concave in x_1, x_2. Thus, maxima of L will be maxima of $z = 2x_1 + 3x_2 - 2x_1{}^2$ and vice versa. To derive the necessary and sufficient condition for maxima of L (and hence of z) we equate the first-order partial derivatives of L w.r.t. the variables $x_1, x_2, S_i^{'s}$ and $\lambda_i^{'s}$. Thus, we have

$$\frac{\partial L}{\partial x_1} = 2 - 4x_1 - \lambda_1 - \lambda_2 + \lambda_3 = 0 \qquad \frac{\partial L}{\partial x_2} = 3 - 4\lambda_1 - \lambda_2 + \lambda_4 = 0$$

$$\frac{\partial L}{\partial S_1} = -2\lambda_1 S_1 = 0 \qquad \frac{\partial L}{\partial \lambda_1} = x_1 + 4x_2 + S_1^2 - 4 = 0$$

$$\frac{\partial L}{\partial S_2} = -2\lambda_2 S_2 = 0 \qquad \frac{\partial L}{\partial \lambda_2} = x_1 + x_2 + S_2^2 - 2 = 0$$

$$\frac{\partial L}{\partial S_3} = -2\lambda_3 S_3 = 0 \qquad \frac{\partial L}{\partial \lambda_3} = -x_1 + S_3^2 = 0$$

$$\frac{\partial L}{\partial S_4} = -2\lambda_4 S_4 = 0 \qquad \frac{\partial L}{\partial \lambda_4} = -x_2 + S_4^2 = 0$$

Upon simplification and necessary manipulations these yield:

(1) $$\begin{cases} 4x_1 + \lambda_1 + \lambda_2 - \lambda_3 = 2,\ 4\lambda_1 + \lambda_2 - \lambda_4 = 3 \\ x_1 + 4x_2 + S_1{}^4 = 4,\ x_1 + x_2 + S_2{}^2 = 2 \end{cases}$$

(2) $\lambda_1 S_1{}^2 + \lambda_2 S_2{}^2 + x_1\lambda_3 + x_2\lambda_4 = 0,\ x_1, x_2, S_1{}^2, S_2{}^2, \lambda_i \geq 0$, $i = 1, 2, 3, 4.$

A solution x_j, $j = 1, 2$ to (1) above and satisfying (2) shall necessarily be an optimal one for maximizing L. To determine the solution to the above simultaneous equation (1), we introduce the artificial variables A_1 and A_2 (both non-negative) in the first constraints of (1) and construct the dummy objective function $z = A_1 + A_2$.

Then the problem becomes,

Maximize $z = -A_1 - A_2$

Subject to the constraints: $4x_1 + \lambda_1 + \lambda_2 - \lambda_3 + A_1 = 2$

$4\lambda_1 + \lambda_2 - \lambda_4 + A_2 = 3$

$x_1 + 4x_2 + x_3 = 4$

$x_1 + x_2 + x_4 = 2$

$x_1, x_2, x_3, x_4 \geq 0$

$A_1, A_2, \lambda_i \geq 0$, $i = 1, 2, 3, 4.$

Satisfying the complementary slackness condition $\sum \lambda_i x_i = 0$, where we have replaced S_1^2 by x_3 and S_2^2 by x_4.

The two phase simplex approach will now be used to find the LPP's optimal solution. A simple workable solution to the LPP is provided in a transparent manner by

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} A_1 \\ A_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 2 \\ 3 \\ 4 \\ 2 \end{bmatrix}$$

The simplex iterations leading to an optimal solution are:

Initial Iteration: Enter x_1 and drop A_1

c_B	y_B	x_B	x_1	x_2	x_3	x_4	λ_1	λ_2	λ_3	λ_4	A_1	A_2
1	A_1	2	**4**	0	0	0	1	1	-1	0	1	0

1	A_2	3	0	0	0	0	4	1	0	-1	0	1
0	x_3	4	1	4	1	0	0	0	0	0	0	0
0	x_4	2	1	1	0	1	0	0	0	0	0	0
	z	5	4	0	0	0	5	2	-1	-1	0	0

From the above table, we observe that x_1, λ_1 or λ_2 can enter the basis. But λ_1 and λ_2 will not enter the basis, because x_3 and x_4 are in the basis. This is in view of the complimentary slackness conditions $\lambda_1 x_3 = 0$ and $\lambda_2 x_4 = 0$.

First iteration: Enter x_2 and x_3.

c_B	y_B	x_B	x_1	x_2	x_3	x_4	λ_1	λ_2	λ_3	λ_4	A_2
0	x_1	1/2	1	0	0	0	1/4	1/4	-1/4	0	0
1	A_2	3	0	0	0	0	4	1	0	-1	1
0	x_3	7/2	0	**4**	1	0	-1/4	-1/4	1/4	0	0
0	x_4	3/2	0	1	0	1	-1/4	-1/4	1/4	0	0
	z	3	0	0	0	0	4	1	0	0	0

Here, we observe that either λ_1 or λ_2 can enter the basis. But x_3 and x_4 are still in the basis, therefore these cannot enter the basis because of the complementary slackness conditions. However, since λ_4 is not in the basis, x_2 can enter the basis (because of the condition $\lambda_4\ x_2$ =0).

Second iteration: Enter λ_1 and drop A_2

c_B	y_B	x_B	x_1	x_2	x_3	x_4	λ_1	λ_2	λ_3	λ_4	A_2
0	x_1	1/2	1	0	0	0	1/4	1/4	-1/4	0	0
1	A_2	3	0	0	0	0	**4**	1	0	-1	1

0	x_2	7/8	0	1	1/4	0	-1/16	-1/16	1/16	0	0
0	x_4	5/8	0	0	-1/4	1	-3/16	-3/16	3/16	0	0
	z	3	0	0	0	0	4	1	0	-1	0

From this table, we see that λ_1 or λ_2 can enter the basis. But since x_4 is in the basis, λ_2 can't enter the basis and hence λ_1 enters the basis.

Final Iteration: Optimum solution

c_B	y_B	x_B	x_1	x_2	x_3	x_4	λ_1	λ_2	λ_3	λ_4
0	x_1	5/16	1	0	0	0	0	3/16	-1/4	1/16
0	λ_1	3/4	0	0	0	0	1	1/4	0	-1/4
0	x_2	59/64	0	1	1/4	0	0	-3/64	1/16	-1/64
0	x_4	49/64	0	0	-1/4	1	0	-9/64	3/16	-3/64
	z	0	0	0	0	0	0	0	0	0

The optimal solution is, $x_1 = 5/16, x_1 = 59/64$ and Maximum $z = 3.19$

13.4 *BEALE'S METHOD*

Beale's algorithm is a method for solving quadratic programming problems (QPP), which are optimization problems where the objective function is quadratic and the constraints are linear. The method is an extension of the simplex algorithm used for linear programming, modified to handle the quadratic nature of the objective function.

Beale's algorithm for *QPP*

Let the general quadratic programming be to maximize $f(x) = c^T x + \frac{1}{2} x^T Qx$, subject to the constraints: $Ax\{\leq, \geq or = b\}$ and $x \geq 0$: where $x \in R^n$, A is $m \times n$, b is $m \times 1$, c is $n \times 1$ and Q is an $n \times n$ symmetric matrix.

The Beale's iterative procedure of solving this problem can be summarized as follows:

Step 1: If necessary, change the minimisation function f(x) to the maximisation function. If there is an inequality restriction, add a slack or surplus variable and write QPP in standard form.

Step 2: Any m variables can be randomly chosen to be the basic variable, making the remaining $n-m$ variables non-basic. Indicate which variables are fundamental and non-basic by

$x_B = (x_{B_1}, x_{B_2}, ..., x_{B_m})$ and $x_{NB} = (x_{NB_1}, x_{NB_2}, ..., x_{NB_{n-m}})$ respectively.

Step 3: Express each basic variables x_{B_i} entirely in terms of the non-basic variables $x_{NB_i}'s$ (and $u_i's$, if any) using the given (as well as additional, if any) constraints.

Step 4: Express the objective function f, also entirely in terms of the non-basic $x_{NB_i}'s$ $x_{NB_i}'s$ (and $u_i's$, if any).

Step 5: Examine the partial derivative of $f(x)$ formulated above w.r.to the non-basic variable at the point $x_{NB} = 0$ (and $u = 0$):

(i) If $\left(\frac{\partial f(x)}{\partial x_{NB_K}}\right)x_{\substack{NB=0 \\ u=0}} = 0$ For each $k = 1, 2, ..., n-m$

and $\left(\frac{\partial f(x)}{\partial u_i}\right)x_{\substack{NB=0 \\ u=0}} = 0$ For each i

the current basic solution is optimal. Go to step 8.

(ii) If $\left(\frac{\partial f(x)}{\partial x_{NB_K}}\right)x_{\substack{NB=0 \\ u=0}} > 0$ For at least one k,

Choose the most positive one. The corresponding non-basic variable will enter the basis.

(iii) If $\left(\frac{\partial f(x)}{\partial x_{NB_K}}\right)x_{\substack{NB=0 \\ u=0}} < 0$ For each $k = 1, 2, ..., n-m$

But $\left(\frac{\partial f(x)}{\partial u_i}\right)x_{\substack{NB=0 \\ u=0}} \neq 0$ For some $i = r$

Introduce a new non-basic variable, u_j, defined by $u_j = \frac{1}{2}\frac{\partial f}{\partial u_r}$ and treat u_r as a basic variable. Then move to step 3.

Step 6: Let $x_{NB_i} = x_k$ be the entering variable identified in Step 5 (ii). Compute the minimum of ratios

$$\left\{\frac{\alpha_{h_0}}{|\alpha_{hk}|}, \frac{\gamma_{h_0}}{|\gamma_{kk}|}\right\}$$

For all basic variable x_h, where α_{h_0} is the constant term and α_{hk} is the coefficient of x_k, in the expression of basic variable x_h when expressed in terms of the non-basic ones; and γ_{k_0} is the constant term and γ_k is the coefficient of x_k in $\frac{\partial f}{\partial x_k}$.

(i) If the minimum ratio occurs for some $\frac{\alpha_{h_0}}{|\alpha_{hk}|}$, the corresponding basic variable, x_h, will leave the basis.

(ii) If the minimum ratio occurs for some $\frac{\gamma_{h_0}}{|\gamma_{kk}|}$, the exit criterion correspond to non-basic variable. In this case, introduce an additional non-basic variable, called the free variable defined by

$$u_i = \frac{1}{2}\frac{\partial f}{\partial x_k} \quad (u_i \text{ is unrestricted})$$

Which relation becomes an additional constraint equation.

Step 7: Proceed to Step 3 and continue the process until the best basic answer is obtained.

Step 8: Establish $x_{NB} = 0$ in the expressions of x_B and f(x) that were derived in Steps 3 and 4 to get the best values of these variables.

Example 2: Solve the following NLPP using the Beale's method:

Minimize $z = 6 - 6x_1 + 2x_1^2 - 2x_1x_2 + 2x_2^2$

Subject to the constraints: $x_1 + x_2 \le 2$ and $x_1, x_2 \ge 0$

Solution. Introducing a slack variable x_3, the constraint equation becomes $x_1 + x_2 + x_3 = 2$. Converting the minimization objective function into a maximization one, we have

$$f(\mathbf{x}) = -6 + 6x_1 - 2x_1^2 + 2x_1x_2 - 2x_2^2.$$

Choosing arbitrarily x_3 as the basic variable, we have

$$\mathbf{x}_B = (x_3), \quad \mathbf{x}_{NB} = (x_1, x_2).$$

Expressing $\mathbf{x}_B$ and $f(\mathbf{x})$ in terms of $\mathbf{x}_{NB}$, we have

$$x_3 = 2 - x_1 - x_2$$

$$f = -6 + 6x_1 - 2x_1^2 + 2x_1x_2 - 2x_2^2.$$

The partial derivatives are :

$$\left(\frac{\partial f}{\partial x_1}\right)_{\mathbf{x}_{NB}=0} = \left(6 - 4x_1 + 2x_2\right)_{\substack{x_1=0\\x_2=0}} = 6$$

$$\left(\frac{\partial f}{\partial x_2}\right)_{\mathbf{x}_{NB}=0} = \left(2x_1 - 4x_2\right)_{\substack{x_1=0\\x_2=0}} = 0.$$

Clearly, x_1 will enter the basis.

Now,

$$\min.\left\{\frac{\alpha_{30}}{|\alpha_{31}|}, \frac{\gamma_{10}}{|\gamma_{11}|}\right\} = \min.\left\{\frac{2}{|-1|}, \frac{6}{|-4|}\right\} = \frac{6}{4}$$

which corresponds to $\dfrac{\gamma_{10}}{|\gamma_{11}|}$. Thus, we cannot remove x_3 from the basis. We introduce a new non-basic free variable u_1, defined by

$$u_1 = \frac{1}{2}\frac{\partial f}{\partial x_1} = 3 - 2x_1 + x_2$$

The current basis is $\mathbf{x}_B = (x_3, x_1)$ and $\mathbf{x}_{NB} = (x_2, u_1)$. Expressing the current $\mathbf{x}_B$ and $f(\mathbf{x})$ in terms of $\mathbf{x}_{NB}$:

$$x_1 = \frac{3}{2} - \frac{1}{2}u_1 + \frac{1}{2}x_2, \quad x_3 = \frac{1}{2} + \frac{1}{2}u_1 - \frac{3}{2}x_2$$

and
$$f = -6 + \left(\frac{3}{2} - \frac{1}{2}u_1 + \frac{1}{2}x_2\right)(6 - 3 + u_1 - x_2 + 2x_2) - 2x_2^2$$
$$= -6 + \left(\frac{3}{2} - \frac{1}{2}u_1 + \frac{1}{2}x_2\right)(3 + u_1 + x_2) - 2x_2^2$$
$$= -\frac{3}{2} - \frac{1}{2}u_1^2 - \frac{3}{2}x_2^2 + 3x_2$$

Evaluating the partial derivatives of f w.r.t. $\mathbf{x}_{NB}$:

$$\left(\frac{\partial f}{\partial x_2}\right)_{\substack{\mathbf{x}_{NB}=0\\u_1=0}} = \left(3 - 3x_2\right)_{x_2=0} = 3$$

$$\left(\frac{\partial f}{\partial u_1}\right)_{\substack{\mathbf{x}_{NB}=0\\u_1=0}} = \left(-\frac{u_1}{2}\right)_{u_1=0} = 0$$

Clearly, x_2 will enter the basis.

Again, we compute the ratios,

$$\min.\left\{\frac{\alpha_{10}}{|\alpha_{12}|}, \frac{\alpha_{30}}{|\alpha_{32}|}, \frac{\gamma_{20}}{|\gamma_{22}|}\right\} = \min.\left\{\frac{3/2}{1/2}, \frac{1/2}{|-3/2|}, \frac{3}{|-3|}\right\} = \frac{\alpha_{30}}{|\alpha_{32}|}$$

Thus, x_3 will leave the basis and the new variables are :

$$\mathbf{x}_B = (x_1, x_2) \quad \text{and} \quad \mathbf{x}_{NB} = (u_1, x_3)$$

Expressing the new basis $\mathbf{x}_B$ and $f(\mathbf{x})$ in terms of $\mathbf{x}_{NB}$:

$$x_2 = \frac{1}{3} + \frac{1}{3}u_1 - \frac{2}{3}x_3, \quad x_1 = \frac{5}{3} - \frac{1}{3}u_1 - \frac{1}{3}x_3$$

and
$$f = -6 + \left(\frac{5}{3} - \frac{1}{3}u_1 - \frac{1}{3}x_3\right)\left(\frac{10}{3} + \frac{4}{3}u_1 - \frac{2}{3}x_3\right) - 2\left(\frac{1}{3} + \frac{1}{3}u_1 - \frac{2}{3}x_3\right)^2$$
$$= -\frac{2}{3} + \frac{2}{3}u_1 - \frac{4}{3}x_3 + \frac{2}{3}x_3u_1 - \frac{2}{3}u_1^2 - \frac{2}{3}x_3^2.$$

The partial derivatives of f w.r.t. $\mathbf{x}_{NB}$ are :

$$\left(\frac{\partial f}{\partial x_3}\right)_{\substack{\mathbf{x}_{NB}=0\\u_1=0}} = \left(\frac{-4}{3} + \frac{2}{3}u_1 - \frac{4}{3}x_3\right)_{\substack{x_3=0\\u_1=0}} = -\frac{4}{3}$$

$$\left(\frac{\partial f}{\partial u_1}\right)_{\substack{x_{NB}=0\\u_1=0}} = \left(\frac{2}{3} + \frac{2}{3}x_3 - \frac{4}{3}u_1\right)_{\substack{x_3=0\\u_1=0}} = \frac{2}{3}$$

Since $\frac{\partial f}{\partial u_1} \neq 0$, current solution can be further improved. However, the entry criterion does not qualify x_3 to enter the basis. Thus, we introduce another non-basic free variable u_2, defined by

$$u_2 = \frac{1}{2}\frac{\partial f}{\partial u_1} = \frac{1}{3} + \frac{1}{3}x_3 - \frac{2}{3}u_1.$$

Treating u_1 as a basic variable, and expressing the basic variables $f(\mathbf{x})$ in terms of non-basic variables, we have

$$u_1 = \frac{1}{2} - \frac{3}{2}u_2 + \frac{1}{2}x_3,\ \ x_2 = \frac{1}{2} - \frac{1}{2}u_2 - \frac{1}{2}x_3,\ x_1 = \frac{3}{2} + \frac{1}{2}u_2 - \frac{1}{2}x_3$$

and

$$\begin{aligned} f &= -\frac{2}{3} + \frac{2}{3}u_1(1 + x_3 - u_1) - \frac{4}{3}x_3 - \frac{2}{3}x_3^2 \\ &= -\frac{2}{3} + \frac{1}{6}(1 - 3u_2 + x_3)(1 + 3u_2 - x_3) - \frac{4}{3}x_3 - \frac{2}{3}x_3^2 \\ &= -\frac{2}{3} + \frac{1}{6}(1 - 9u_2^2 - 5x_3^2 + 6u_2x_3 - 8x_1) \\ &= -\frac{1}{2} + \frac{1}{6}[-9u_2^2 - 5x_3^2 + 6u_2x_3 - 8x_3]. \end{aligned}$$

Now we see that

$$\left(\frac{\partial f}{\partial x_3}\right)_{\substack{x_{NB}=0\\u_2=0}} = \frac{1}{6}\left(-10x_3 + 6u_2 - 8\right)_{\substack{x_3=0\\u_2=0}} = -\frac{8}{6}$$

$$\left(\frac{\partial f}{\partial u_1}\right)_{\substack{x_{NB}=0\\u_2=0}} = \frac{1}{6}\left(-18u_2 + 6x_3\right)_{\substack{x_3=0\\u_2=0}} = 0.$$

Thus the current $\mathbf{x}_B$ provides an optimal solution.

Ignoring the free variable u_1, the optimal solution to the given *QPP* is :

$$x_1 = \frac{3}{2},\ \ x_2 = \frac{1}{2};\ \ z^* = -\left(-\frac{1}{2}\right) = \frac{1}{2}.$$

Check your progress

Problem 1: Solve the following QPP by Wolfe's method

Minimize, $z = 4x_1 + 6x_2 - 2x_1^2 - 2x_1x_2 - 2x_2^2$

Subject to the constraints: $x_1 + x_2 \leq 2$ and $x_1, x_2 \geq 0$

Answer: $x_1 = 1/3, x_2 = 5/6$ and Maximum $z = 25/6$

Problem 2: Solve the following QPP by Beale's method

Minimize, $z = 2x_1 + 3x_2 - x_1^2$

Subject to the constraints: $x_1 + 2x_2 \leq 4$ and $x_1, x_2 \geq 0$

Answer: $x_1 = 1/4, x_2 = 15/8$ and Maximum $z = 97/16$

13.5 *SUMMARY*

Wolfe's and Beale's methods are specialized algorithms for solving quadratic programming problems, where the objective function is quadratic and the constraints are linear. Wolfe's method extends the simplex algorithm by transforming the quadratic problem into a series of linear programming subproblems, iteratively solving them while ensuring feasibility and optimality through duality theory and the Karush-Kuhn-Tucker (KKT) conditions. Beale's method similarly modifies the simplex approach, linearizing the quadratic term around the current solution and solving successive linear approximations until convergence. Both methods are effective for convex quadratic programming, leveraging the simplicity of the simplex method while addressing the complexities introduced by the quadratic objective function.

13.6 *GLOSSARY*

- Wolfe's method
- Beale's method

13.7 *REFERENCES*

- Mokhtar S. Bazaraa, John J. Jarvis and Hanif D. Sherali: *Linear Programming and Network Flows* (4th edition). John Wiley and Sons, 2010.
- Hamdy A. Taha: *Operations Research: An Introduction (10th edition). Pearson*, 2017.
- Paul R. Thie and Gerard E. Keough: *An Introduction to Linear Programming and Game Theory* (3rd edition), Wiley India Pvt. Ltd, 2014.
- Swarup, K., Gupta, P. K., & Mohan, M. (2014). An introduction to management science operation research. *Sultan Chand & Sons educational publishers, New Delhi*.
- OpenAI. (2024). *ChatGPT (August 2024 version) [Large language model]*. OpenAI. https://www.openai.com/chatgpt

13.8 *SUGGESTED READING*

- G. Hadley, *Linear Programming*, Narosa Publishing House, 2002.
- Frederick S. Hillierand Gerald J. Lieberman: *Introduction to Operations Research* (10th edition). McGraw-Hill Education, 2015.
- https://epgp.inflibnet.ac.in/Home/ViewSubject?catid=ZLCHeZEhCZ8yCri36nSF3A==

13.9 *TERMINAL QUESTION*

Short Answer Type Question:

1: Solve the following QPP by Wolfe's method

Maximize, $z = 8x_1 + 10x_2 - x_1^2 - x_2^2$

Subject to the constraints: $3x_1 + 2x_2 \le 6$ and $x_1, x_2 \ge 0$

2: Solve the following QPP by Wolfe's method

Maximize, $z = 6x_1 + 3x_2 - 4x_1x_2 - 2x_1^2 - 3x_2^2$

Subject to the constraints: $x_1 + x_2 \le 1$; $2x_1 + 3x_2 \le 4$ and $x_1, x_2 \ge 0$

3: Solve the following QPP by Wolfe's method

Maximize, $f(x_1, x_2) = 1.8x_1 + 3x_2 - 0.001x_1^2 - 0.005x_2^2 - 100$

Subject to the constraints: $2x_1 + 3x_2 \le 2500$; $x_1 + 2x_2 \le 1500$ and $x_1, x_2 \ge 0$

4: Solve the following QPP by Wolfe's method

Maximize, $z = 2x_1 + x_2 - x_1^2$

Subject to the constraints: $2x_1 + 3x_2 \le 6$; $2x_1 + x_2 \le 4$ and $x_1, x_2 \ge 0$

5: Solve the following QPP by Wolfe's method

Maximize, $z = -x_1 - x_2 - x_3 + \frac{1}{2}\left(x_1^2 + x_2^2 + x_3^2\right)$

Subject to the constraints: $x_1 + x_2 + x_3 \le 1$; $4x_1 + 2x_3 \le 7/3$ and $x_1, x_2, x_3 \ge 0$

Long answer type question:

1: Solve the following QPP by Beale's method

Minimize, $f = 4x_1 + 6x_2 - x_1^2 - 3x_2^2$

Subject to the constraints: $x_1 + 2x_2 \le 4$ and $x_1, x_2 \ge 0$.

2: Solve the following QPP by Beale's method

Minimize, $f(x) = \frac{1}{4}(2x_3 - x_1) - \frac{1}{2}(x_1^2 + x_2^2 + x_3^2)$

Subject to the constraints: $x_1 - x_2 + x_3 = 1$ and $x_1, x_2, x_3 \ge 0$.

3: Solve the following QPP by Beale's method

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

Subject to the constraints: $x_1 + 4x_2 \le 4$ and $x_1 + x_2 \le 2$; $x_1, x_2 \ge 0$

4: Solve the following QPP by Beale's method

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

Subject to the constraints: $2x_1 + x_2 \ge 6$ and $x_1 - 4x_2 \ge 0$; $x_1, x_2 \ge 0$

Objective type question:

1: **Wolfe's Method is primarily used to solve:**

a) Linear Programming Problems

b) Non-linear Programming Problems

c) Quadratic Programming Problems

d) Dynamic Programming Problems

2: Beale's Method is an extension of which algorithm?

a) Newton's Method

b) Gradient Descent

c) Simplex Method

d) Genetic Algorithm

3: Which of the following is a key feature of Wolfe's Modified Simplex Method?

a) It uses duality theory for solving linear programming problems.

b) It converts the quadratic programming problem into a series of linear programming subproblems.

c) It is specifically designed for non-convex optimization problems.

d) It does not require any initial feasible solution.

4: In Beale's Method, the quadratic programming problem is solved by:

a) Linearizing the quadratic term and solving a series of linear programs

b) Directly solving the quadratic program using matrix inversion

c) Applying a genetic algorithm to find the global minimum

d) Using Lagrange multipliers to solve the problem

5: Which of the following statements is true about Wolfe's Modified Simplex Method?

a) It is only applicable to linear programming problems with equality constraints.

b) It requires the quadratic objective function to be convex (positive semi-definite matrix).

c) It can solve non-linear programming problems with any type of objective function.

d) It does not require any iterations and provides a direct solution.

Fill in the blanks:

1: Wolfe's method transforms a quadratic programming problem into a series of ________ subproblems.

2: Beale's method is used for solving ________ programming problems where the objective function is quadratic and the constraints are linear.

3: In Wolfe's method, the quadratic programming problem must have a ________ matrix for the quadratic term to ensure a convex objective function.

4: Beale's method involves iteratively refining the solution by updating the ________ of the quadratic objective function.

5: Wolfe's and Beale's methods are extensions of the ________ method for handling quadratic programming problems.

True and False:

1. Wolfe's Modified Simplex Method can be used to solve both convex and non-convex quadratic programming problems.
2. Beale's method directly solves the quadratic programming problem without transforming it into linear subproblems.
3. The KKT conditions are used in Wolfe's Modified Simplex Method to ensure the optimality of the solution.
4. Beale's Method requires that the matrix associated with the quadratic term in the objective function be symmetric and positive semi-definite.
5. Wolfe's Method does not require an initial feasible solution to start the optimization process.

13.10 *ANSWERS*

Answer of short answer type question

Answer 1: $x_1 = 4/13, x_2 = 33/13$; Maximum $z = 267/13$

2: $x_1 = 1, x_2 = 0$; Maximum $z = 4$

3: $x_1 = 500, x_2 = 500$; Maximum $z = 800$

4: $x_1 = 2/3, x_2 = 14/9$; Maximum $z = 22/9$

5: $x_1 = x_2 = x_3 = 1/3$; Minimum $z = -5/6$

Answer of long answer type question

Answer 1: $x_1 = 2, x_2 = 1, x_3 = 0$; Maximum $f = 7$

2: $x_1 = 1/8, x_2 = 0, x_3 = 7/8$; Maximum $f(x) = 1/64$

3: $x_1 = 0, x_2 = 3/4$; Maximum $z = 9/8$

4: $x_1 = 38/13, x_2 = 2/13$; Maximum $z = 116/13$

Answer of Multiple choice question

Answer 1: (c) **2:** (c) **3:** (b)

4: (a) **5:** (b)

Answer of fill in the blank question

Answer 1: linear programming **2:** Quadratic **3:** positive semi-definite

4: linear approximation **5:** simplex

True and False

Answer 1: False **2:** False **3:** True

4: True **5:** False

UNIT-14: GOAL PRAGRAMMING

CONTENTS:

14.1 *INTRODUCTION*

Goal programming is an extension of linear programming that is used to solve multi-objective optimization problems where multiple, often conflicting, goals need to be achieved simultaneously. Unlike traditional optimization methods that focus on optimizing a single objective, goal programming allows for the prioritization of various goals by minimizing the deviations from these target values. This method is particularly useful in decision-making scenarios where trade-offs between different objectives are necessary, providing a flexible framework for finding solutions that best satisfy the set goals within the given constraints.

14.2 *OBJECTIVE*

After reading this unit learners will be able to

- Understand the concept of Goal programming.
- Understand the concept of Graphical goal attainment method.
- Understand the concept of simplex method for goal programming problem.

14.3 *LINEAR GOAL PROGRAMMING PROBLEM*

The formulation of a linear goal programming problem involves translating multiple goals into a mathematical model where deviations from these goals are minimized. The steps for formulating a linear goal programming problem are as follows.

1. Identify Decision Variables: Define the decision variables $x_1, x_2, ..., x_n$ that will be optimized to achieve the desired goals.

2. Set Up Goal Equations:

- Express each goal as a linear equation. Each goal will typically have a target value, and can be represented as: $a_{i1}x_1 + a_{i2}x_2 + ... + a_{in}x_n = b_i$
- Here, a_{ij} are coefficients, and b_i is the target value for the i^{th} goal.

3. Introduce Deviation Variables:

- For each goal, introduce deviation variables to represent under-achievement (negative deviation d_i^-) and over-achievement (positive deviation d_i^+) from the target:
  $a_{i1}x_1 + a_{i2}x_2 + ... + a_{in}x_n + d_i^- - d_i^+ = b_i$
- d_i^- and d_i^+ are non-negative variables representing the amount by which the actual achievement falls short of or exceeds the target, respectively.

4. Establish Priority Structure: Assign priorities $(P_1, P_2, ...)$ to each goal based on their importance. Goals with higher priority should be satisfied before those with lower priority.

5. Formulate the Objective Function:

- The objective function is to minimize the weighted sum of the deviations from the goals: Minimize $Z = \sum_i \left(P_i^- d_i^- + P_i^+ d_i^+\right)$
- Here, P_i^- and P_i^+ are weights or penalty factors for under-achievement and over-achievement deviations for each goal.

6. Specify Constraints: Include the original constraints of the problem, if any, alongside the goal equations. These constraints should be linear and represent the limitations within which the solution must lie.

7. Solve the Goal Programming Model: Use linear programming techniques to solve the goal programming model, seeking to minimize the weighted deviations and achieve the goals as closely as possible.

Example 1: Suppose a company has two goals: maximize profit and minimize production costs, with more emphasis on maximizing profit. Let x_1 and x_2 be the amounts produced of two products.

1. Goal 1 (Maximize profit):

$$5x_1 + 4x_2 + d_1^- - d_1^+ = 10000$$

2. Goal 2 (Minimize costs):

$$2x_1 + 3x_2 + d_2^- - d_2^+ = 2000$$

3. Objective Function:

Minimize $Z = P_1^+ d_1^+ + P_2^- d_2^-$

where P_1^+ and P_2^- reflect the priorities of the goals.

4. Constraints:

$$x_1 + x_2 \le 5000, \; x_1, x_2 \ge 0$$

This model would then be solved to find the optimal production levels x_1 and x_2 that best satisfy the goals.

Example 2: A company makes, let's say, X and Y. Product X generates a net profit per unit of Rs. 80, but product Y generates a net profit per unit of Rs. 40. The company wants to make Rs. 900 in the upcoming week. Additionally, the management hopes to reach sales volumes of about 17 and 15, respectively, for the two goods. Create a goal programming model for this problem.

Solution:

Let x_1 and x_2 denote the number of units of product X and Y respectively. The linear programming formulation of the problem is :

$$\text{Maximize } z = 80x_1 + 40x_2 \text{ subject to the constraints :}$$
$$x_1 \leq 17, \; x_2 \leq 15 \text{ and } x_1 \geq 0, \; x_2 \geq 0.$$

Since the goal of the firm pertains to profit attainment with a target established at Rs. 900 per week, the constraints of the problem can be stated as :

$$80x_1 + 40x_2 = 900 \quad \textit{(profit target goal)}$$
$$x_1 \leq 17 \text{ and } x_2 \leq 15 \quad \textit{(sales target goal)}$$

The problem can now be formulated as goal programming model as follows :

$$\text{Minimize } z = d_1^- + d_1^+ + d_2^- + d_3^- \text{ subject to the constraints :}$$
$$80x_1 + 40x_2 + d_1^- - d_1^+ = 900, \; x_1 + d_2^- = 17, \; x_2 + d_3^- = 15,$$
$$x_1, x_2, d_1^-, d_1^+, d_2^-, d_3^- \geq 0 ;$$

where d_2^- and d_3^- represent underachievements of sales volume for products A and B respectively. Since sales target goals are given as the maximum possible sales volume, therefore d_2^+ and d_3^+ are not included in the sales target constraints.

d_1^- = under-achievement of the profit goal of Rs. 900.

d_1^+ = over-achievement of the profit goal of Rs. 900.

Example 2: There are two varieties of TV sets produced by Delton Electronics. The Supreme TV set takes four hours to assemble, whereas the Deluxe TV set just takes two hours. There is a weekly cap of 80 hours for regular assembly work. According to a marketing assessment, the weekly production of TV sets should not exceed 60 deluxe and 30 ultimate sets. The net profit for each deluxe model is Rs. 100, and for each supreme model, it is Rs. 150.

The company president has stated the following objectives in order of priority.

(1) Maximize total profit
(2) Minimize overtime operation of the assembly line
(3) Sell as many TVs as you can; note that this does not equate to maximum revenue. The president is twice as motivated to maximise sales of the supreme model as he is of the Deluxe model since the net profit from the supreme model is double that of the Deluxe model.

Formulate this as a linear Goal programming problem.

Solution: The decision variables are:

x_1 = number of Deluxe TV sets built each week,

x_2 = number of Supreme TV sets built each week.

The highest priority objective of the president is to maximise profit. Profit, in turn, can be written as a function of x_1 and x_2 as follows :

$$\text{Profit/week} = 100x_1 + 150x_2.$$

Setting profit to an arbitrarily high level of Rs. 5,000 per week, the goal is written as :

$$100x_1 + 150x_2 + d_1^- - d_1^+ = 5{,}000$$

Note that d_1^- is the negative deviation measure, *i.e.*, the amount by which we *underachieve* our objective. On the other hand, d_1^+ is the amount by which we *overachieve* our target profit level.

The second priority objective is the minimisation of assembly line overtime operation.

Since assembly line time is *normally* 80 hours per week, our goal is :

$$2x_1 + 4x_2 + d_2^- - d_2^+ = 80$$

Here, d_2^- is the amount of "slack" time on the assembly line while d_2^+ represents the amount of "overtime". Since our objective is measured by the minimisation of overtime, we achieve this goal by minimising d_2^+.

The third, and final priority is associated with a sales objective. Since marketing has indicated that the demands for Deluxe and Supreme TV sets are 60 and 30 units, respectively per week, we naturally strive to satisfy these demands and our goal is :

$$x_1 + d_3^- - d_3^+ = 60, \quad \text{and} \quad x_2 + d_4^- - d_4^+ = 30.$$

Our third goal is achieved by minimizing d_3^- and fourth goal is achieved by minimizing d_4^-. Further, since the Supreme profit is 2 times the Deluxe profit, we shall have 2 times more desire to minimize d_4^- as to minimize d_3^-.

The achievement function for the problem, then is to find x_1 and x_2 so as to minimize

$$z = \{d_3^- + d_4^-, d_1^-, d_2^+, d_3^- + 2d_4^-\}$$

The final decision model—a linear goal programming problem is :

Find x_1 and x_2 so as to

Minimize $z = \{d_1^-, d_2^+, d_3^- + d_4^-, d_3^- + 2d_4^-\}$ subject to the constraints :

$$100x_1 + 150x_2 + d_1^- - d_1^+ = 5000,\quad 2x_1 + 4x_2 + d_2^- - d_2^+ = 80,\quad x_1 + d_3^- - d_3^+ = 60,$$

$$x_2 + d_4^- - d_4^+ = 30;\quad x_1, x_2, d_1^-, d_1^+, d_2^-, d_2^+, d_3^-, d_3^+, d_4^-, d_4^+ \geq 0.$$

14.4 *GRAPHICAL GOAL ATTAINMENT METHOD*

The Graphical Goal Attainment (GGA) method is a visual tool used in multi-objective optimization, where multiple conflicting objectives need to be optimized simultaneously. It helps decision-makers find a satisfactory solution that balances different goals by visualizing the trade-offs and allowing for interactive adjustments.

Steps to Implement the GGA Method:

1. **Define Objectives and Constraints:** Identify the objectives you want to optimize. These could be cost, performance, quality, etc.
   - Set constraints that the solution must satisfy.
2. **Set Goal Levels:** Establish a goal level for each objective. This represents the desired target or aspiration level for that particular objective.
3. **Construct the Feasible Region:** Graphically represent the feasible region of solutions by plotting the objectives on a coordinate system, with each axis representing a different objective.
4. **Plot the Goal Point:** Mark the goal point on the graph, which is the point where all the set goal levels intersect.
5. **Determine Attainment Region:** Draw lines (or curves) from the origin (or another reference point) to the goal point. This represents the trade-offs between different objectives.
   - The region formed by these lines is the attainment region, where the feasible solutions are compared to the goals.
6. **Evaluate Solutions:** Solutions that fall within or near the attainment region are considered better, as they are closer to meeting the set goals.
7. **Iterative Adjustment:** Decision-makers can adjust the goals or constraints and re-plot the graph to explore different scenarios and find the most satisfactory solution.

Example: Imagine you are optimizing two objectives: cost and quality. You would:

- Plot cost on the x-axis and quality on the y-axis.
- Set a desired cost level and a desired quality level, and mark this as the goal point.
- Plot the feasible solutions based on the constraints.
- Draw the attainment region from the origin to the goal point and see which solutions fall closest to or within this region.

The GGA method is particularly useful in scenarios where decision-makers need to balance multiple objectives and want a visual tool to help understand the trade-offs.

Example 3: The Duplex Paints firm produces two kind of exterior paints: enamel point and latex. Twenty labour hours are needed for every 200 gallons of latex point, and thirty labour hours are needed for every 200 gallons of enamel paint. It is expected that the company's owner would not consider recruiting more staff or using overtime, and that there are only 80 man hours of work available each week. For every gallon, the latex and enamel paints provide a profit of two rupees. The company's managing director has communicated to the board of directors that these two paints should generate a profit of Rs. 2,000 every week. Lastly, the management has also committed to providing a sister company with 1,400 gallons of enamel paint every week, if at all feasible. Solve this as an LGPP formulation.

Solution:

The management objectives and their priorities are as follows :

P_1 : avoid the utilization of overtime—this is given the top priority because it is stated in the problem as an absolute objective. *(First goal)*

P_2 : achieve a weekly profit of at least Rs. 2,000 *(Second goal)*

P_3 : supply the sister concern at least 1,400 gallons of enamel paint every week. *(Third goal)*

Let x_1 and x_2 designate the decision variables corresponding to the number of latex and enamel paints (200 gallons taken as a unit) required to be manufactured per week satisfying all the goals. If d_i^- denotes the negative deviation from the ith goal attainment and d_i^+ denotes the positive deviation from the ith goal attainment, then the various goals of the problem (in order of priority) are :

$$G_1 : \quad 20x_1 + 20x_2 + d_1^- - d_1^+ = 80$$

$$G_2 : \quad 200x_1 + 200x_2 + d_2^- - d_2^+ = 2000$$

$$G_3 : \quad 2x_2 + d_3^- - d_3^+ = 14$$

The goal G_1 shall be completely attained if overtime is minimised, that is, the positive deviation (d_1^+) is minimised (*i.e.*, overutilisation is minimised). The second goal shall be attained if the profit is maximised which will happen if the negative deviation (d_2^-) from the goal is minimised (*i.e.*, underachievement is minimised). The third goal shall be attained when the underachievement, *i.e.*, d_3^- is minimised. The achievement function of the problem is thus

$$z = \{d_1^+, d_2^-, d_3^-\}$$

The following is the formulation of the linear goal programming problem:
Determine x_1 and x_2 so as to
Minimize, $z = \{d_1^+, d_2^-, d_3^-\} = p_1 d_1^+ + p_2 d_2^- + p_3 d_3^-$; and satisfy the goals:

$$20x_1 + 20x_1 + d_1^- - d_1^+ = 80$$
$$200x_1 + 200x_2 + d_2^- - d_2^+ = 2000$$
$$2x_2 + d_3^- - d_3^+ = 14$$

$$x_1, x_2, d_1^-, d_2^-, d_3^-, d_1^+, d_2^+, d_3^+ \geq 0$$

Now, construct the graph of goals

The graph representing the different objectives will have two axes: a horizontal one called the X_1-axis and a vertical one called the X_2-axis. This is because there are only two decision variables, x_1 and x_2. Any pair's value (x_1, x_2) may be found at any location in any of the four quadrants. However, as the variables cannot take on negative values, we are limited to the first quadrant alone. Plotting the goal equations G_1, G_2, and G_3 as though there were no deviations from goal achievement results in the three constraints being shown only as straight lines once each $d_i^- = d_i^+ = 0$ is handled. Figure 1 has the three goals shown as straight lines. Keep in mind that the graph only uses the decision variables (x_1 and x_2). The arrows ($\rightarrow$ and $\leftarrow$) across each target line, however, represent the region of an increase in any positive or negative deviation variable ($d_1^-, d_2^-, d_3^-, d_1^+, d_2^+ \, and \, d_3^+$). The accomplishment function's specific deviation variables, or those that need to be minimised, have been highlighted.

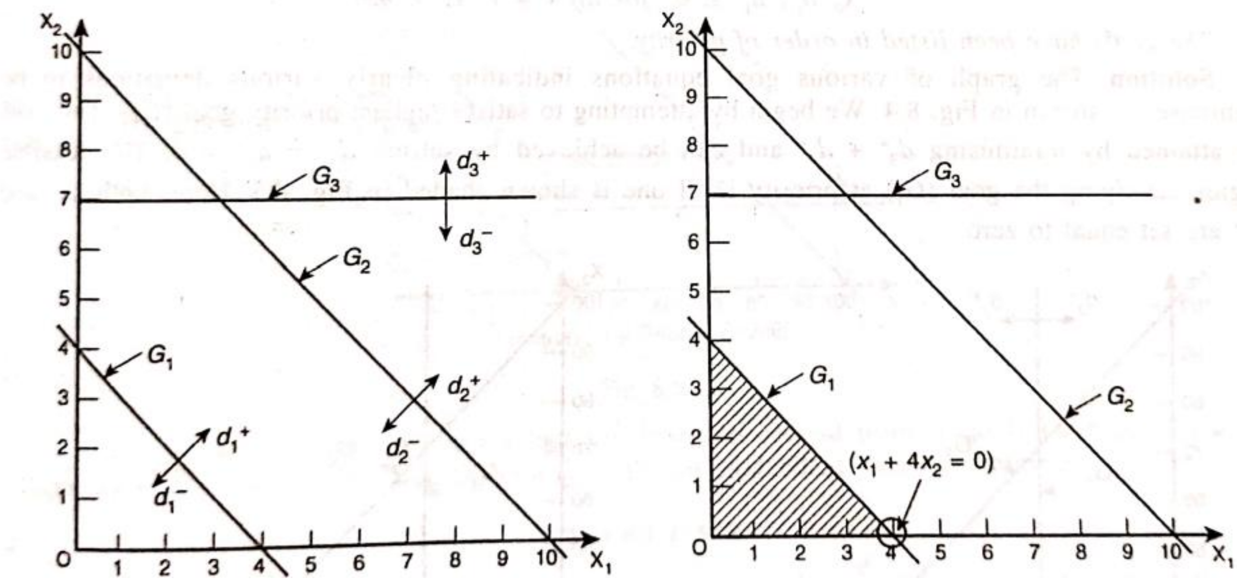


Figure: 2

Goal attainment at priority level one:

First priority (G_1) goals are taken into consideration. When d_1^+ is minimised, this objective will be accomplished. As this is the initial objective, its minimum value can be set to $d_1^+ = 0$. In Figure 1, the region that satisfies $d_1^+ = 0$ and $x_1, x_2 \geq 0$ is shaded. Any two values (x_1 and x_2) in the area must minimise d_1^+ at the first priority level while also satisfying G_1.

Goal attainment at next higher priority level:

Moving on to P_2, the next higher priority, we now take those goals into consideration. This is only G_2, and we have to minimise d_2^- in order to fulfil G_2. The reader can quickly see that d_2^- cannot be adjusted to zero, as this would deteriorate our preceding solution, which was a higher priority aim. The point shown in Figure 1 is the smallest value of d_2^- without harming the previously achieved higher priority goals, which is at $x_1 = 4, x_2 = 0$.

Moving now to the last priority level we consider the goal G_3. To satisfy G_3 we must minimise d_3^-. But we see that any minimisation of d_2^- would require a movement from the solution previously determined and thus would degrade at least one higher priority goal already attained. Consequently the optimal solution to the problem is

$$x_1^o = 4, \; x_2^o = 0; \; z^o = (0, 600, 14).$$

The optimum values of the achievement function z^o simply indicate that the highest priority goal has been completely attained while the second and third priority goals have not.

14.5 *SIMPLEX METHOD FOR GOAL PROGRAMMING PROBLEM*

Goal Programming (GP) is an extension of Linear Programming (LP) that deals with multiple, often conflicting objectives. The Simplex Method for Goal Programming modifies the standard Simplex Method to handle these multiple goals by introducing deviational variables.

Steps to Solve a Goal Programming Problem Using the Simplex Method:

1: ***Formulate the Problem*:**

- **Identify the Goals:** List the specific goals you want to achieve.
- **Define the Constraints:** Write down the constraints that limit the solution space.
- **Identify Deviational Variables:** For each goal, introduce deviational variables d_1^+ and d_1^- representing the positive and negative deviations from the target, respectively.

2. ***Objective Function:***

- **Weight the Deviations:** Assign weights to the deviational variables based on the priority or importance of each goal. The general objective is to minimize the weighted sum of deviations.
- **Formulate the Objective:** Minimize $Z = \sum_i w_i^+ d_i^+ + w_i^- d_i^-$, where w_i^+ and w_i^- the weights of positive and negative deviations.

3. ***Set Up the Simplex Table:***

- **Initial Simplex Table:** Include all decision variables, slack variables, and deviational variables in the simplex tableau.
- **Artificial Variables:** If needed, introduce artificial variables to handle equality constraints or goal constraints directly in the simplex method.

4. ***Apply the Simplex Algorithm:***

- **Iterate:** Apply the usual Simplex method, selecting the entering and leaving variables and updating the tableau iteratively until an optimal solution is found.

5. ***Interpret the Solution:***

- **Check the Deviational Variables:** Examine the deviational variables to understand how well each goal has been met.
- **Analyze the Results:** The values of the decision variables provide the optimal solution, and the deviational variables show the degree of goal attainment or shortfall.

Example Problem:

Let's go back to the Duplex Paints Company example.

Goals:

1. Achieve a profit of Rs. 2,000 per week.
2. Supply 1,400 gallons of enamel paint.

Decision Variables:

Let x_1 and x_2 be the number of gallons (in hundreds) of latex and enamel paints produced per week, respectively.

Formulate the Goals:

1. Profit Goal: $2x_1 + 2x_2 = 2000$ (profit of Rs. 2 per gallon)
   - Deviations: $2x_1 + 2x_2 + d_1^- - d_1^+ = 2000$
2. Enamel Paint Supply Goal: $x_2 = 14$ (supply 1,400 gallons)
   - Deviations: $x_2 + d_2^- - d_2^+ = 14$

Constraints:

- Labor Constraint: $10x_1 + 15x_2 \le 80$ (total labor hours available per week)
- Non-negativity: $x_1, x_2, d_1^-, d_1^+, d_2^-, d_2^+ \ge 0$

Objective Function:

Assuming equal weight for simplicity: Minimize Z= $d_1^- + d_1^+ + d_2^- + d_2^+$

Set Up and Solve Using Simplex:

- Form the initial tableau with decision variables x_1, x_2 slack variables for constraints, and deviational variables.
- Follow the Simplex procedure to find the optimal values of $x_1, x_2, d_1^-, d_1^+, d_2^-$ and d_2^+.

Solution Interpretation:

The optimal solution gives the values of x_1, x_2 that minimize the deviations from the goals, showing the best balance between conflicting objectives given the constraints.

Example 4: A company's production manager wishes to plan a week-long production run for two items, A and B, each of which needs the manpower and supplies listed below:

Product	Labour (in hours)	Material M_1 (in Kgs)	Material M_2 (in Kgs)
A	2	4	5
B	4	5	4
Available (per week)	600	1000	1200

The unit profit for A and B is Rs. 20 and Rs. 32 respectively.

The manager would like to maximise profit, but he is equally concerned with maintaining workforce of the division at nearly constant level in the interest of employee morale. The work, which consists of people engaged in production, sales, distribution and other general staff is consisted of 108 persons in all. Also, it is known that the production of one unit of A would maintain 0.3 persons in the workforce and one unit of B would maintain 0.75 persons.

Had the production manager been considering only maximising profit, without regard to maintaining the workforce, he would do so by producing 167.67 units of A and 66.67 units of B. On the basis of the available capacity, this would yield a profit of Rs. 5,486.67. However, this would maintain 100.3 people in the workforce. The manager feels that probably he could increase the workforce requirement to the desired level by accepting somewhat lower profit. So, the following two goals are to be achieved : (a) profit of Rs. 5,400 per week and (b) workforce of 108 persons.

Formulate and solve the given problem as linear goal programming problem.

Solution: Let $x_1 =$ Number of units or products A to produced every week.

Let $x_2 =$ Number of units or products B to be produced every week.

The problem's limitations and objectives may therefore be stated as follows:

$2x_1 + 4x_2 \leq 600$ (Labour Constraint)

$4x_1 + 5x_2 \leq 1000$ (Material M_1 Constraint)

$5x_1 + 4x_2 \leq 1200$ (Material M_2 Constraint)

$20x_1 + 32x_2 = 5400$ (Goal 1)

$0.3x_1 + 0.75x_2 = 108$ (Goal 2)

Since, we have to satisfy goal 1 and goal 2 simultaneously, the given problem may not have feasible solution. Further, in order to solve the given problem by simplex method, we introduce the deviational variables d_1^+ and d_1^- in goal 1 constraint and, d_2^+ and d_2^- in goal 2 constraints, where

d_1^+ = number of rupees above the goal of Rs. 5,400,
d_1^- = number of rupees below the goal of Rs. 5,400,

d_2^+ = number of people above the workforce goal of 108,

d_2^- = number of people below the workforce goal of 108.

Making use of slack variables $s_1 \geq 0$, $s_2 \geq 0$, $s_3 \geq 0$ in the first three constraints respectively, and the deviational variables in the fourth and fifth constraints; the goal linear programming problem is:

Minimize $z = d_1^- + d_2^- + 0.s_1 + 0.s_2 + 0.s_3 + 0.d_1^+ + 0.d_2^+$ subject to the constraints:

$$2x_1 + 4x_2 + s_1 = 600, \ 4x_1 + 5x_2 + s_2' = 1{,}000$$

$$5x_1 + 4x_2 + s_3 = 1{,}200, \ 20x_1 + 32x_2 + d_1^- - d_1^+ = 5{,}400$$

$$0.3x_1 + 0.75x_2 + d_2^- - d_2^+ = 108,$$

$$x_1, x_2, s_1, s_2, s_3 \geq 0 \text{ and } d_1^-, d_1^+, d_2^-, d_2^+ \geq 0.$$

Solution by Simplex method

An initial, basic, and feasible solution is found by applying the simplex approach.

$s_1 = 600, s_2 = 1{,}000, s_3 = 1200, d_1^- = 5400$ and $d_2^- = 108$

With I_5 as the initial basis matrix.

Initial Iteration: Introduce y_2 and drop d_2^-.

			0	0	0	0	0	1	0	1	0
c_B	y_B	x_B	y_1	y_2	y_3	y_4	y_5	d_1^-	d_1^+	d_2^-	d_2^+
0	y_3	600	2	4	1	0	0	0	0	0	0
0	y_4	1000	4	5	0	1	0	0	0	0	0
0	y_5	1200	5	4	0	0	1	0	0	0	0
1	d_1^-	5400	20	32	0	0	0	1	-1	0	0
1	d_2^-	108	0.3	**0.75**	0	0	0	0	0	1	-1
	z	5508	20.3	32.75	0	0	0	0	-1	0	-1

Since, $z_1 - c_1 > 0$ and $z_1 - c_2 > 0$, current solution is not optimum. As the largest of these two positive quantities is 32.75 corresponding to $z_2 - c_2$, y_2 enters the basis. Further,

c_B	y_B	x_B	y_1	y_2	y_3	y_4	y_5	d_1^-	d_1^+	d_2^-	d_2^+
0	y_3	24	2/5	0	1	0	0	0	0	-16/3	**16/3**
0	y_4	280	2	0	0	1	0	0	0	-20/3	20/3
0	y_5	624	17/5	0	0	0	1	0	0	-16/3	16/3
1	d_1^-	792	36/5	0	0	0	0	1	-1	-128/3	128/3
1	y_2	144	2/5	1	0	0	0	0	0	4/3	-4/3
	z	792	36/5	0	0	0	0	0	-1	-131/3	128/3

Since, $z_9 - c_9 = \frac{128}{3}$ is largest positive net evaluation, d_2^+ enters the basis. Further, min. $\left\{\frac{x_{Bi}}{y_{i9}}, y_{i9} > 0\right\}$ is $\frac{24}{16/3}$. This implies y_3 leaves the basis.

Second iteration: Introduce y_1 and drop d_2^+.

c_B	y_B	x_B	y_1	y_2	y_3	y_4	y_5	d_1^-	d_1^+	d_2^-	d_2^+
0	d_2^+	9/2	**3/40**	0	3/16	0	0	0	0	-1	1
0	y_4	250	3/2	0	-5/4	1	0	0	0	0	0
0	y_5	600	3	0	-1	0	1	0	0	0	0
01	d_1^-	600	4	0	-8	0	0	1	-1	0	0
1	y_2	150	1/2	1	1/4	0	0	0	0	0	0
	z	600	4	0	-8	0	0	0	-1	0	0

Clearly, y_1 enters the basis, since $z_1 - c_1 > 0$. Also, Min. $\left\{\frac{x_{Bi}}{y_{i1}}, y_{i1} > 0\right\} = \frac{9/2}{3/40}$ indicates d_2^+ leaves the basis.

Third iteration: Introduce d_2^- and drop d_1^-

c_B	y_B	x_B	y_1	y_2	y_3	y_4	y_5	d_1^-	d_1^+	d_2^-	d_2^+
0	y_1	60	1	0	5/2	0	0	0	0	-40/3	40/3
0	y_4	160	0	0	-5	1	0	0	0	20	-20
0	y_5	420	0	0	-17/2	0	1	0	0	40	-40

0	d_1^-	360	0	0	-18	0	0	1	-1	**160/3**	-160/3
1	y_2	120	0	1	-1	0	0	0	0	20/3	-20/3
	z		0	0	-18	0	0	0	-1	157/3	-160/3

Clearly, d_2^- enters the basis because $z_8 - c_8 > 0$, and d_1^- leaves the basis.

Final iteration: Optimum solution.

c_B	y_B	x_B	y_1	y_2	y_3	y_4	y_5	d_1^-	d_1^+	d_2^-	d_2^+
0	y_1	150	1	0	-2	0	0	¼	-1/4	0	0
0	y_4	25	0	0	7/4	1	0	-3/8	3/8	0	0
0	y_5	150	0	0	5	0	1	-3/4	¾	0	0
0	d_2^-	27/4	0	0	-27/80	0	0	-3/160	-3/160	1	-1
1	y_2	75	0	1	5/4	0	0	1/8	1/8	0	0
	z	27/4	0	0	-27/80	0	0	-3/160	-3/160	0	-1

Since, all $z_j - c_j \le 0$, an optimum solution is obtained. Hence, the optimum solution is :

$x_1 = 150,\ x_2 = 75,\ d_2^- = \frac{27}{4} = 6.75,\ s_2 = 25$ and $s_3 = 150$ with the minimum of $z = 6.75$.

This implies that the workforce shall be 108 – 6.75 (= 101.25), with the employment goal being under-achieved to the extent of 6.75 people ; while 25 kg. of material M_1 and 150 kg. of material M_2 would remain unutilised. The other variables are non-basic so that all the available labour hours shall be used and the profit goal be met exactly.

Check your progress

Problem 1: A manufacturing company creates two different kinds of products: A and B. Product A has a profit per unit of Rs. 70, while product B has a profit per unit of Rs. 30. The company wants to make precisely Rs. 800 in profit over the course of the upcoming week. Create a goal programming model for this problem.

Answer: Minimize, $z = \{d_1^-, d_1^+\}$

Subject to the constraints: $70x_1 + 30x_2 + d_1^- - d_1^+ = 800$ and $x_1, x_2, d_1^-, d_1^+ \ge 0$

14.6 *SUMMARY*

Goal Programming is a mathematical optimization technique used to handle decision-making problems involving multiple, often conflicting objectives. Unlike traditional linear programming, which focuses on optimizing a single objective, Goal Programming seeks to minimize the deviations from pre-set target levels for each objective. By introducing deviational variables to represent overachievement and underachievement of goals, and assigning weights based on the importance of each goal, Goal Programming provides a structured approach to finding a balanced solution that best meets all objectives within the given constraints. This method is widely used in scenarios where decision-makers must consider and prioritize multiple goals simultaneously.

14.7 *GLOSSARY*

- Goal programming problem
- Graphical goal attainment method
- Simplex method for goal programming problem

14.8 *REFERENCES*

- Mokhtar S. Bazaraa, John J. Jarvis and Hanif D. Sherali: *Linear Programming and Network Flows* (4th edition). John Wiley and Sons, 2010.
- Hamdy A. Taha: *Operations Research: An Introduction (10th edition). Pearson*, 2017.
- Paul R. Thie and Gerard E. Keough: *An Introduction to Linear Programming and Game Theory* (3rd edition), Wiley India Pvt. Ltd, 2014.
- Swarup, K., Gupta, P. K., & Mohan, M. (2014). An introduction to management science operation research. *Sultan Chand & Sons educational publishers, New Delhi*.
- OpenAI. (2024). *ChatGPT (August 2024 version) [Large language model]*. OpenAI. https://www.openai.com/chatgpt

14.9 *SUGGESTED READING*

- G. Hadley, *Linear Programming*, Narosa Publishing House, 2002.

- Frederick S. Hillierand Gerald J. Lieberman: *Introduction to Operations Research* (10th edition). McGraw-Hill Education, 2015.

- https://epgp.inflibnet.ac.in/Home/ViewSubject?catid=ZLCHeZEhCZ8yCri36nSF3A==

14.10 *TERMINAL QUESTION*

Short Answer Type Question:

1: Two successive machines are used to make two items. The machining times for the two products, expressed in minutes per unit, are shown in the accompanying table.

Machine	Product 1	Product 2
X	7	5
Y	8	4

Use a goal programming model to formulate the problem. The two goods have a daily production quota of 70 and 50 units, respectively. Each machine operates for eight hours a day. Although overtime is not ideal, it may be essential to reach the production quota.

2: Two types of items are produced by a producer of office equipment: chairs and lighting. The plant needs an hour of manufacturing capacity to produce a chair or a light. The plant can only produce for a maximum of 50 hours every week. The maximum number of chairs and lamps that may be sold each week are six and eight, respectively, due to the limited sales capacity. For a chair, the gross margin is Rs. 90, and for a light, it is Rs. 60.

The plant manager wants to calculate how many units of each product should be produced each week while taking the following objectives into account:

(i) The maximum manufacturing capacity that is available should be used, but not exceeded.
(ii) Sales of two products should be as much as possible
(iii) Overtime should not exceed 20 per cent of available production time.

Create a Goal programming model for this issue so that the plant manager may get as near to his objectives as feasible.

3: Solve the following linear goal programming problem graphically, Find x_1 and x_2 so as to:

Minimise, $z = G_1(d_3^+ + d_4^+) + G_2 d_1^+ + G_3 d_2^- + G_4\left(d_3^- + \frac{3}{2} d_4^-\right)$ and satisfy the goals:

G_1 : $x_1 + x_2 + d_1^- + d_1^+ = 40$

G_2 : $x_1 + x_2 + d_2^- - d_2^+ = 100$

G_3 : $x_1 + d_3^- - d_3^+ = 30$

G_4 : $x_2 + d_4^- - d_4^+ = 15$

$x_i, d_i^-, d_i^+ \geq 0$ for all $i = 1, 2, 3, 4$

The goals have been listed in order of priority.

Long answer type question:

1: Solve the following goal programming problems using the simplex method:

Minimize, $f = P_1 d_1^- + P_2 d_2^- + P_3 d_1^+$

Subject to the constraints: $10x_1 + 10x_2 + d_1^- - d_1^+ = 400$; $x_1 + d_2^- = 40$; $x_2 + d_3^- = 30$;
$x_1, x_2, d_1^+, d_1^-, d_2^-, d_3^- \geq 0$

2: Solve the following goal programming problems using the simplex method:

Minimize, $f = P_1 d_1^- + P_2 (2d_2^- + d_3^-) + P_3 d_1^+$

Subject to the constraints: $x_1 + x_2 + d_1^- - d_1^+ = 400$; $x_1 + d_2^- = 240$; $x_2 + d_3^- = 300$;
$x_1, x_2, d_1^-, d_1^+, d_2^-, d_3^- \geq 0$

Objective type question:

1: What is the primary objective of Goal Programming?

a) Maximizing profit

b) Minimizing cost

c) Minimizing deviations from goals

d) Maximizing resource utilization

2: In Goal Programming, what do the deviational variables d_i^+ and d_i^- represent?

a) Surplus and shortage in resources

b) Overachievement and underachievement of goals

c) Positive and negative profits

d) Slack and surplus variables

3: Which of the following statements is true about Goal Programming?

a) It optimizes a single objective.

b) It cannot handle multiple conflicting objectives.

c) It uses weighted deviations to prioritize different goals.

d) It is only applicable to linear problems.

4: When using Goal Programming, if a goal is not achievable, which of the following is a possible solution approach?

a) Increase the resource availability

b) Adjust the goal level or priority

c) Ignore the goal

d) Add more constraints

5: In the context of Goal Programming, what does it mean if a deviational variable d_i^+ has a value of zero?

a) The goal has been exactly met.

b) The goal has been underachieved.

c) The goal has been overachieved.

d) The goal is unattainable.

6: Which of the following is not typically a feature of Goal Programming?

a) Incorporating multiple goals with different priorities

b) Minimizing the total cost only

c) Handling conflicting objectives

d) Using a weighted objective function to balance different goals

7: In a Goal Programming problem, which of the following is a correct approach to model a goal?

a) Use constraints to restrict all deviations to zero.

b) Introduce artificial variables for each goal.

c) Introduce deviational variables and include them in the objective function.

d) Ignore deviations and focus only on maximizing or minimizing the objective function.

Fill in the blanks:

1: In Goal Programming, the primary objective is to minimize the ___________ from the specified goals.

2: The variables d_i^+ and d_i^- in Goal Programming represent the ___________ and ___________ deviations from the goal, respectively.

3: The ___________ method is commonly used to solve Goal Programming problems by iteratively improving the solution.

4: When prioritizing goals in Goal Programming, the goals with higher ___________ are considered more important and are optimized first.

5: In Goal Programming, if the objective is to minimize the deviation from a goal, the objective function can be formulated as minimizing the sum of ___________ deviations for all goals.

6: Goal Programming allows for multiple objectives to be achieved simultaneously by introducing ___________ variables into the model to capture the extent of goal achievement.

7: A Goal Programming problem is a type of ___________ optimization problem that extends the concept of Linear Programming to handle multiple objectives.

8: In Goal Programming, a ___________ solution is one that best satisfies the goals while adhering to the constraints of the problem.

9: The ___________ region in a Goal Programming problem represents all possible solutions that satisfy the given constraints.

10: When the positive and negative deviations from a goal are equal to zero in Goal Programming, it indicates that the goal has been ___________.

True and False:

1: Goal Programming is used to optimize a single objective function.

2: In Goal Programming, the objective is to minimize the deviations from the desired goals.

3: Positive and negative deviational variables in Goal Programming measure how much a goal has been overachieved or underachieved.

4: In Goal Programming, all goals must have equal importance and weight in the objective function.

5: Goal Programming can handle both linear and non-linear constraints.

6: If the deviation variables in Goal Programming are all zero, it indicates that all goals have been exactly met.

7: The Simplex method can be adapted to solve Goal Programming problems.

8: In Goal Programming, it is possible to ignore less important goals if they conflict with higher-priority goals.

9: In Goal Programming, slack and surplus variables are used to measure deviations from goals.

10: Goal Programming is not suitable for problems with multiple conflicting objectives.

14.11 *ANSWERS*

Answer of short answer type question

Answer 1: Minimize, $z = \{d_1^+, d_2^+, d_3^-, d_4^+\}$

Subject to the constraints: $x_1 - d_1^+ + d_1^- = 70; x_2 - d_2^+ + d_2^- = 50$;
$7x_1 + 5x_2 - d_3^+ + d_3^- = 480; 8x_1 + 4x_2 - d_4^+ + d_4^- = 480$ and all variables are non-negative.

2: Minimize, $z = \{d_1^+, d_2^-, d_3^-, d_4^+\}$

Subject to the constraints: $x_1 + x_2 + d_1^- - d_1^+ = 50; x_1 + d_2^- = 6; x_2 + d_3^- = 8$;
$d_1^+ + d_4^- - d_4^+ = 10;\ x_1, x_2, d_1^-, d_1^+, d_2^-, d_3^-, d_4^-, d_4^+ \geq 0$.

3: The final solution is $x_1 = 25, x_2 = 15, z^0 = (0, 0, 60, 5)$

Answer of long answer type question

Answer 1: $x_1 = 70, x_2 = 30, d_1^+ = 300; d_1^- = d_2^- = d_3^- = 0$

2: $x_1 = 240, x_2 = 300, d_1^+ = 140; d_1^- = d_2^- = d_3^- = 0$

Answer of Multiple choice question

Answer 1: **c)** **2:** **b)** **3:** **c)**

4: **b)** **5:** **a)** **6:** **b)**

7: **c)**

Answer of fill in the blank question

Answer 1: deviations **2:** positive, negative **3:** Simplex

4: weights (or priority) **5:** weighted **6:** deviational

7: multi-objective **8:** feasible **9:** feasible

10: Exactly met

True and False

Answer 1: False **2:** True **3:** True

4: False **5:** False **6:** True

7: True **8:** True **9:** False

10: False

Teen Pani Bypass Road, Transport Nagar
Uttarakhand Open University,
Haldwani, Nainital-263139
Phone No. 05946-261122, 261123
Toll free No. 18001804025
Fax No. 05946-264232,

E-mail:info@uou.ac.in
Website: https://www.uou.ac.in/