Operations Research: Lecture Notes

DEPARTMENT OF MATHEMATICS, FACULTY OF SCIENCE, SOUTH VALLEY UNIVERSITY

BY DR. HILLAL M. ELSHEHABEY

Version release, 2024

Contents

1. Operations Research Overview

The main core of the current chapter is to give an answer to the following questions; what is the operations research? and how it is useful to different branches of science, especially to the computer science researchers. Then we provide the definition of optimization problems and see its mathematical statement. Finally, examples for establishing the mathematical formulation of practical problems as well as some benchmark problems are given.

1.1 Introduction

1.1.1 Definition of Operations research

Operations research, for short OR, is the act of obtaining the best result under given circumstances. Thus, we may have several solutions for a given problem and our aim is to find the best solution among those solutions taking into account some certain constrains that many exist.

- Another definition of OR is the interdisciplinary branch of applied mathematics and formal science that uses methods like mathematical modeling, statistics, and algorithms by applying advanced analytical/numerical methods to arrive at optimal or near optimal solutions to complex problems hence, make **better decisions**.

1.1.2 impacts and applications of OR

The impacts and applications of operations research: "Operations research has had an impressive impact on improving the efficiency of numerous organizations around the world. In the process, OR has made a significant contribution to increasing the productivity of the economies of various countries. There now are a few dozen member countries in the International Federation of Operational Research Societies (IFORS), with each country having a national OR society. Both Europe and Asia have federations of OR societies to coordinate holding international conferences and publishing international journals in those continents". The following are the abbreviated set of typical operations research applications to show how widely these techniques are used today:

- Accounting:
  - Assigning audit teams effectively
  - Credit policy analysis
  - Cash flow planning
  - Developing standard costs
  - Establishing costs for byproducts
  - Planning of delinquent account strategy
- Construction:
  - Project scheduling, monitoring and control
  - Determination of proper work force
  - Deployment of work force
  - Allocation of resources to projects
- Facilities Planning:
  - Factory location and size decision
  - Estimation of number of facilities required
  - Hospital planning
  - International logistic system design
  - Transportation loading and unloading
  - Warehouse location decision
- Finance:

  - Building cash management models
  - Allocating capital among various alternatives
  - Building financial planning models
  - Investment analysis
  - Portfolio analysis
  - Dividend policy making

- Manufacturing:
  - Inventory control
  - Marketing balance projection
  - Production scheduling
  - Production smoothing

- Marketing:
  - Advertising budget allocation
  - Product introduction timing
  - Selection of Product mix
  - Deciding most effective packaging alternative

- Organizational Behavior / Human Resources:
  - Personnel planning
  - Recruitment of employees
  - Skill balancing
  - Training program scheduling
  - Designing organizational structure more effectively

- Purchasing:
  - Optimal buying
  - Optimal reordering
  - Materials transfer

- Research and Development:
  - R & D Projects control
  - R & D Budget allocation
  - Planning of Product introduction

1.1.3 Historical Background of OR

- **Pre-World war II:** The roots of OR are as old as science and society. Though the roots of OR extend to even early 1800s, it was in 1885 when Ferderick W. Taylor emphasized the application of scientific analysis to methods of production, that the real start took place.
- Henry L. Gantt, emerged during a time when job scheduling methods were largely disorganized. During this period, it was common for a job to smoothly proceed on one machine but then experience lengthy waiting times before being accepted by the next machine. Gantt introduced a systematic approach by meticulously charting the path of each job through various machines, thus reducing delays significantly. Thanks to the Gantt procedure, it became feasible to forecast machine workloads several months in advance while maintaining precision in quoting delivery dates.
- In 1917, A.K.Erlang, a Danish mathematician, published his work on the problem of congestion of telephone traffic. The difficulty was that during busy periods, telephone operators were many, resulting in delayed calls. A few years after its appearance, his work was accepted by the British Post Office as the basis for calculating circuit facilities.
- The well known economic order quantity model is attributed to F.W. Harris, who published his work on the area of inventory control in 1915.
- In the 1930s, H.C. Levinson, an American astronomer, utilized scientific methods to address issues related to merchandising. His research encompassed the systematic examination of customer purchasing patterns, the impact of advertising on consumer behavior, and the correlation between the selling environment and the nature of products being sold.
- However, it was the First Industrial Revolution which contributed mainly towards the development of OR. Before this revolution, most of the industries were small scale, employing only a handful of men.
- The advent of machine tools-the replacement of man by machine as a source of power and improved means of transportation and communication resulted in fast

flourishing industry. It became increasingly difficult for a single man to perform all the managerial functions (of planning, sale, purchase, production, etc.). Consequently, a division of management function took place. Managers of production, marketing, finance, personnel, research and development etc., began to appear. With further industrial growth, further subdivisions of management functions took place. For example ,production department was sub-divided into sections like maintenance, quality control, procurement, production planning, etc.

- **World War II:** During World War II, the British military leadership enlisted a team of scientists to analyze strategic and tactical challenges related to both air and land defense. This team was led by Professor P.M.S. Blackett from the University of Manchester, a former naval officer. Known as the "Blackett circus," this group consisted of three physiologists, two mathematical physicists, an astrophysicist, an army officer, a surveyor, a general physicist, and two mathematicians. Many of the problems they tackled were of an administrative nature. Their primary goal was to determine the most efficient allocation of scarce military resources for various military operations and the tasks within each operation. Their work encompassed optimizing the use of newly developed radar technology, assigning British Air Force planes to specific missions, and devising effective patterns for locating submarines. This assembly of scientists marked the inception of the first operational research (OR) team.
- The name operations research (or operational research) was apparently coined because the team was carrying out research on (military)operation.the encouraging results of these effort led to the information of more such teams in British armed services and the use of scientific teams soon spread to western allies-the united states, Canada and France. thus through this scince of operation research originated in England,the united states soon took the lead.in united state these OR teams helped in developing stategiesfrom mining operations, inventing new flight patterns and planning of sea mines.
- **Post-world war II** : Right after the war, the achievements of military teams garnered

the interest of industrial managers in search of solutions for their challenges. The field of industrial operational research took distinct paths in the United Kingdom and the United States. In the UK, the urgent economic circumstances demanded a significant boost in production efficiency and the establishment of new markets. The nationalization of select key industries further expanded the scope for operational research. Consequently, operational research quickly extended its reach beyond the military domain to encompass government, industrial, social, and economic planning.

- In USA the situation was different. Impressed by its dramatic success in U.K., defence operations research in U.S.A was increased. Most of the war experienced OR workers remained in military service. Industrial executives did not call for much help because they were returning to the peace-time situation and many of them believed that it was merely a new application of an old technique. Operation research by a variety of names in that country such as operational analysis, operation evaluation, systems analysis, system evaluation, system research and management science.
- The scenario in the USA was distinct. The remarkable success of operational research in the UK left an impression, and as a result, defense-related operational research efforts in the USA were expanded. A majority of the operational research experts with wartime experience continued to serve in the military. Industrial executives, on the other hand, did not seek as much assistance because they were transitioning back to peacetime conditions, and some of them perceived it as a mere adaptation of existing techniques. In the United States, operational research was referred to by various names, including operational analysis, operation evaluation, systems analysis, system evaluation, system research, and management science.

1.1.4 Basic facts about OR

The following are basic facts about Operations Research:

- It is a science-based approach to analyzing problems and decision situations to aid solving such problems and decision-making. It is therefore a practical activity, although based on the theoretical construction and analysis.

- It is an approach and an aid to problem-solving and decision-making.
- Its distinctive approach is facts-finding and modeling.
- It examines functional relations (i.e. functions of a system and their related components) from a system overview.
- It utilizes interdisciplinary mixed-team approach to solving management problems.
- It adopts the planned approach (updated scientific method which reflects technological advancement as the computer) to management problems.
- It helps to discover new problems as one problem is being solved.

1.1.5 Limitations of Operations Research

Operations Research has number of applications; similarly it also has certain limitations. These limitations are mostly related to the model building and money and time factors problems involved in its application. Some of them are as given below:

- Distance between OR specialist and Manager

  Operations Researchers job needs a mathematician or statistician, who might not be aware of the business problems. Similarly, a manager is unable to understand the complex nature of Operations Research. Thus there is a big gap between the two personnel.
- Magnitude of Calculations

  The aim of the OR is to find out optimal solution taking into consideration all the factors. In this modern world these factors are enormous and expressing them in quantitative model and establishing relationships among these require voluminous calculations, which can be handled only by machines.
- Money and Time Costs

  The basic data are subjected to frequent changes, incorporating these changes into the operations research models is very expensive. However, a fairly good solution at present may be more desirable than a perfect operations research solution available in future or after some time.
- Non-quantifiable Factors:

When all the factors related to a problem can be quantifiable only then operations research provides solution otherwise not. The non-quantifiable factors are not incorporated in OR models. Importantly OR models do not take into account emotional factors or qualitative factors.

- Implementation:

  Once the decision has been taken it should be implemented. The implementation of decisions is a delicate task. This task must take into account the complexities of human relations and behavior and in some times only the psychological factors.

1.2 Steps of OR analysis

The seven steps to a good OR analysis

1. Identify the problem or opportunity

   During this step one has to identify the objectives and to determine if the proposed problem is too narrow or if it is too broad.

2. Observe and understand the system under consideration

   Within this step, we are seeking answers to the following questions:

   (a) what data should be collected?

   (b) How will data be collected?

   (c) How do different components of the system interact with each other ?

3. Formulate a mathematical model

   (a) What kind of models should be used?

   (b) Is the model accurate?

   (c) Is the model too complex?

4. Verify the model and use it for prediction

   (a) Do outputs match current observations for current inputs ?

   (b) Are outputs reasonable?

   (c) Could the model be erroneous?

5. Select the best alternative

   Given a model and a set of alternatives, the analyst now chooses the alternative that best meets the organization's objectives. Sometimes there are many best alternatives, in which case the OR analyst should present them all to the organization's decision-makers, or ask for more objectives or restrictions.Since this is the most difficult step, we could seek if there are software tools that could help us!!
6. Present the results of the analysis

   (a) The outputs of the model do fit within the output from the system itself ?

   (b) Are the outputs of the model reasonable ?

   (c) Is the model correct ? Note that the model may be wrong.
7. Implement and evaluate the obtained solution

   This step is considered with the following two items:

   (a) Users must be trained on the new system

   (b) System must be observed over time to ensure it works properly.

1.3 Problem Formulation

1. Determine **decision variables** that are going to be used mathematically to define the problem
2. Define the quantity to be maximize or minimize.

   This quantity is called **objective function**.
3. Define the **constraints**

   Those are the restriction under which we have to solve our problem.
4. Define the **non-negative constraints**

   We have to be sure that all the variables are of non-negative type. If this is not the case, then we have to modify them as we will see later on in our study.

1.4 Statement of an optimization problem

Definition 1.4.1 An optimization or a mathematical programming problem can be stated as follows. Find

$$\boldsymbol{x} = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}, \text{ which minimize } f(\boldsymbol{x})$$

Subject to the following constrains

$$g_j(\boldsymbol{x}) \leq 0, \qquad j = 1, 2, \cdots, m$$

$$l_j(\boldsymbol{x}) = 0, \qquad j = 1, 2, \cdots, p$$

where, $\boldsymbol{x}$ is an n-dimensional vector called the design vector, $f(\boldsymbol{x})$ is the objective function, and $g_j(\boldsymbol{x})$ and $l_j(\boldsymbol{x})$are known as inequality and equality constraints, respectively. The problem stated in Definition 1.4.1 is called a constrained optimization problem.

OR models can be classified into:

- linear programming: models with linear objective and constraint functions.
- Integer programming: the variables assume integer values
- Dynamic programming: the original model can be decomposed into smaller sub problems
- Network programming: the problem can be modeled as a network
- nonlinear programming: functions of the model are nonlinear.

R Many practical problems can be formulated as an optimization problem in which the objective function and the constrains are are linear i.e. any term is either a constant or a constant multiplied by an unknown. This problem is called **Liner Programming (LP)** problem.

Now, we give some examples for how a practical problem can be formulated as an LP problem.

Example 1.4.1 The Haty shop makes its sandwiches from a combination beef and goat meat. The beef contains 80% meat and 20% fat, and it costs 24 pounds per kilo. The goat meat contains 68% meat and 32% fat, and it costs 18 pounds per kilo. What is the amount of meat from each type must be used in each kilo of meat if it wants to minimize its costs and keep the ratio of fat so that no more than 25%? ■

Solution 1.4.1 Let x_1 be weight of beef meat and x_2 be weight of goat meat

Objective function is

minimize $z = 24x_1 + 18x_2$

The constrains

(1) Rate of fat

$$0.20x_1 + 0.32x_2 \leq 0.25$$

(2) Per kilo

$$x_1 + x_2 = 1$$

Non-negative condition

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

Thus, the final formula for the linear programming problem is

$$\text{Minimize } z = 24x_1 + 18x_2$$

Subject to

$$0.20x_1 + 0.32x_2 \leq 0.25$$

$$x_1 + x_2 = 1$$

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

■

Example 1.4.2 Tela Inc. manufactures two product: #1 and #2. To manufacture one unit of product #1 costs €40 and to manufacture one unit of product #2 costs €60. The profit from product #1 is €30, and the profit from product #2 is €20.

The company wants to maximize its profit. How many products #1 and #2 should it manufacture? ■

■ **Solution 1.4.2** The solution is trivial: There is no bound on the amount of units the company can manufacture. So it should manufacture infinite number of either product #1 or #2, or both. If there is a constraint on the number of units manufactured then the company should manufacture only product #1, and not product #2. This constrained case is still rather trivial. ■

■ **Example 1.4.3 — The transportation problem.**

A certain product is to be shipped in amounts $u_1, u_2, \dots, u_n$ from n service points to m destinations, where it is to be received in amounts $v_1, v_2, \dots, v_m$. See Figure 1.4-1. If the cost of sending one unit product from origin i to destination j is known to be c_{ij}, determine the quantity x_{ij} to be sent from origin i to destination j so that the total transportation cost is minimum.

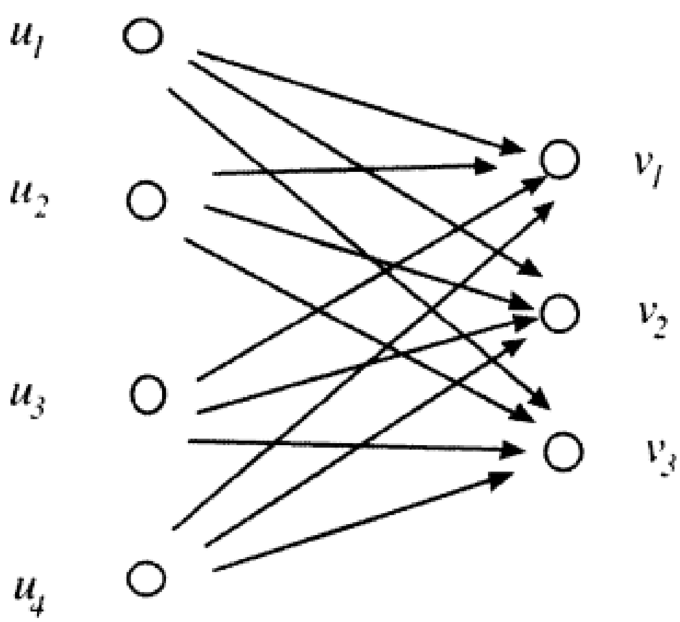


Figure 1.4-1: Sketch of the transportation problem

■

■ Solution 1.4.3

If x_{ij} is the amount of the product sent from initial location i to destination j, then the total cost will be

$$\sum_{i,j} c_{ij} x_{ij}$$

if c_{ij} is the unit cost of sending the product from i to j.

What are the restrictions we must respect? For a fixed service point i, u_i is the quantity to be shipped, so that

$$\sum_j x_{ij} = u_i, \quad i = 1, 2, \ldots, n$$

likewise, for every fixed destination, the amount v_j should be received, and this enforces

$$\sum_i x_{ij} = v_j, \quad j = 1, 2, \ldots, m.$$

Notice that these two sets of equalities are compatible if

$$\sum_i u_i = \sum_j v_j$$

which is a restriction that the data of the problem must satisfy for the problem to be well posed. Moreover, if we accept that the feature of being a service point or a destination cannot be reversed, then we must ask for

$$x_{ij} \geq 0, \text{ for all } i, j.$$

Hence, we are seeking to

$$\text{Minimize } \sum_{i,j} c_{ij} x_{ij}$$

with the following restrictions

$$\sum_j x_{ij} = u_i, \quad i = 1, 2, \ldots, n$$

$$\sum_i x_{ij} = v_j, \quad j = 1, 2, \ldots, m.$$

$$x_{ij} \geq 0 \text{ for all } i, j$$

■

■ **Example 1.4.4** A factory wants in the production of 2 models. The first one needs 3 units of wood; and 3 units of iron; 5 units of aluminum, models II need a single unit of wood; 8 units of iron; 4 units of aluminum. If you know that the maximum available of wood is 53 units, Steel 127 and 100 for aluminum. Form the mathematical model in the following cases:

(A) If the first model is given a profit of unit and the second 2 units.

(B) If the first model gives a profit of two units and the second gives a single unit. ■

■ **Solution 1.4.4** Let the factory produce x unit of 1^{st} model and y from the 2^{nd} one, then Objective function reads

$$\text{(a) Max} Z = x + 2y \qquad \text{(b) Max } Z = 2x + y$$

and the constraints are

For wood;

$$3x + y \leq 53$$

For iron;

$$3x + 8y \leq 127$$

For Aluminum;

$$5x + 4y \leq 100$$

Non-negative condition

$$x \geq 0,\ y \geq 0$$

■

■ Example 1.4.5 — Giapetto Example.

Giapetto's wooden soldiers and trains. Each soldier sells for \$27, uses \$10 of raw materials and takes \$14 of labor & overhead costs. Each train sells for \$21, uses \$9 of raw materials, and takes \$10 of overhead costs. Each soldier needs 2 hours finishing and 1 hour carpentry; each train needs 1 hour finishing and 1 hour carpentry. Raw materials are unlimited, but only 100 hours of finishing and 80 hours of carpentry are available each week. Demand for trains is unlimited; but at most 40 soldiers can be sold each week. How many of each toy should be made each week to maximize profits? ■

■ Solution 1.4.5

Decision variables completely describe the decisions to be made (in this case, by Giapetto). Giapetto must decide how many soldiers and trains should be manufactured each week. With this in mind, we define:

x_1 = the number of soldiers produced per week

x_2 =the number of trains produced per week

Objective function is the function of the decision variables that the decision maker wants to maximize (revenue or profit) or minimize (costs). Giapetto can concentrate on maximizing the total weekly profit (z)

Here profit equals to

(weekly revenues) – (raw material purchase cost) – (other variable costs).

Hence Giapetto's objective function is:

$$\max z = 3x_1 + 2x_2$$

Constraints show the restrictions on the values of the decision variables. Without constraints Giapetto could make a large profit by choosing decision variables to be very large. Here there are three constraints:

(a) Finishing time per week

(b) Carpentry time per week

(c) Weekly demand for soldiers

Sign restrictions are added if the decision variables can only assume nonnegative values (Giapetto can not manufacture negative number of soldiers or trains!)

All these characteristics explored above give the following Linear Programming (LP) model

$$
\begin{array}{lll}
\max z = 3x_1 + 2x_2 & \text{(The Objective function)} & \\
\text{s.t.} & 2x_1 + x_2 \le 100 & \text{(Finishing constraint)} \\
 & x_1 + x_2 \le 80 & \text{(Carpentry constraint)} \\
 & x_1 \quad \le 40 & \text{(Constraint on demand for soldiers)} \\
x_1, x_2 \ge 0 & \text{(Sign restrictions)} &
\end{array}
$$

A value of (x_1, x_2) is in the **feasible region** if it satisfies all the constraints and sign restrictions.

Graphically and computationally we see the solution is $(x_1, x_2) = (20, 60)$ at which $z = 180$. (**Optimal solution**)

Report

The maximum profit is \$180 by making 20 soldiers and 60 trains each week. Profit is limited by the carpentry and finishing labor available. Profit could be increased by buying more labor. ■

■ Example 1.4.6 — Advertisement Example.

Dorian makes luxury cars and jeeps for high-income men and women. It wishes to advertise with 1 minute spots in comedy shows and football games. Each comedy spot costs \$50 K and is seen by 7M high-income women and 2M high-income men. Each football spot costs \$100 K and is seen by 2M high-income women and 12M high-income men. How can Dorian reach 28M high-income women and 24M high-income men at the least cost? ■

Solution 1.4.6 The decision variables are

$$x_1 = \text{ the number of comedy spots}$$
$$x_2 = \text{ the number of football spots}$$

The model of the problem:

$$\min z = 50x_1 + 100x_2$$
$$\text{s.t.} \quad \begin{aligned} 7x_1 + 2x_2 &\geq 28 \\ 2x_1 + 12x_2 &\geq 24 \end{aligned}$$
$$x_1, x_2 \geq 0$$

The graphical solution is $z = 320$ when $(x_1, x_2) = (3.6, 1.4)$. From the graph, in this problem rounding up to $(x_1, x_2) = (4, 2)$ gives the best integer solution.

Report

The minimum cost of reaching the target audience is $400K, with 4 comedy spots and 2 football slots. The model is dubious as it does not allow for saturation after repeated viewings.

Example 1.4.7 — Diet.

Ms. Fidan's diet requires that all the food she eats come from one of the four "basic food groups". At present, the following four foods are available for consumption: brownies, chocolate ice cream, cola, and pineapple cheesecake. Each brownie costs 0.5$, each scoop of chocolate ice cream costs 0.2$, each bottle of cola costs 0.3$, and each pineapple cheesecake costs 0.8$. Each day, she must ingest at least 500 calories, 6oz of chocolate, 10oz of sugar, and 8oz of fat. The nutritional content per unit of each food is shown in Table. Formulate an LP model that can be used to satisfy her daily nutritional requirements at minimum cost.

	Calories	Chocolate (ounces)	Sugar (ounces)	Fat (ounces)
Brownie	400	3	2	2
Choc. ice cream (1 scoop)	200	2	2	4
Cola (1 bottle)	150	0	4	1
Pineapple cheesecake (1 piece)	500	0	4	5

■

Solution 1.4.7 The decision variables:

x_1 : number of brownies eaten daily

x_2 : number of scoops of chocolate ice cream eaten daily

x_3 : bottles of cola drunk daily

x_4 : pieces of pineapple cheesecake eaten daily

The objective function (the total cost of the diet in cents):

$$\min w = 50x_1 + 20x_2 + 30x_3 + 80x_4$$

Constraints:

$$400x_1 + 200x_2 + 150x_3 + 500x_4 \geq 500 \quad \text{(daily calorie intake)}$$

$$3x_1 + 2x_2 \geq 6 \quad \text{(daily chocolate intake)}$$

$$2x_1 + 2x_2 + 4x_3 + 4x_4 \geq 10 \quad \text{(daily sugar intake)}$$

$$2x_1 + 4x_2 + x_3 + 5x_4 \geq 8 \quad \text{(daily fat intake)}$$

$$x_i \geq 0, i = 1, 2, 3, 4 \quad \text{(Sign restrictions!)}$$

Report

The minimum cost diet incurs a daily cost of 90 cents by eating 3 scoops of chocolate and drinking 1 bottle of cola $(w = 90, x_2 = 3, x_3 = 1)$ ■

Example 1.4.8 — Post Office.

A PO requires different numbers of employees on different days of the week. Uni rules state each employee must work 5 consecutive days and then receive two da off. Find the minimum number of employees needed.

	Mon	Tue	Wed	Thur	Fri	Sat	Sun
Staff Needed	17	13	15	19	14	16	11

■

■ **Solution 1.4.8** The decision variables are x_i (# of employees starting on day i) Mathematically we must

$$
\begin{array}{lllll}
[left]\min z = & x_1+x_2 & +x_3+x_4 & +x_5+x_6 & +x_7 \\
\text{s.t.} & x_1 & \quad +x_4 & +x_5+x_6 & +x_7 \geq 17 \\
& x_1+x_2 & & +x_5+x_6 & +x_7 \geq 13 \\
& x_1+x_2 & +x_3 & \quad +x_6 & +x_7 \geq 15 \\
& x_1+x_2 & +x_3+x_4 & & +x_7 \geq 19 \\
& x_1+x_2 & +x_3+x_4 & +x_5 & \geq 14 \\
& \quad +x_2 & +x_3+x_4 & +x_5+x_6 & \geq 16 \\
& & +x_3+x_4 & +x_5+x_6 & +x_7 \geq 11
\end{array}
$$

$$x_t \geq 0, \forall t$$

The solution is $(x_i) = (4/3, 10/3, 2, 22/3, 0, 10/3, 5)$ giving $z = 67/3$. We could round this up to $(x_i) = (2,4,2,8,0,4,5)$ giving $z = 25$ (may be wrong!). However restricting the decision var.s to be integers and using Lindo again gives $(x_i) = (4,4,2,6,0,4,3)$ giving $z = 23$. ■

1.5 Exercises

Exercise 1.5.1

Define the following items:

Operation research, Optimization problem, Steps of OR ■

Exercise 1.5.2 Tela Inc. in Example 1.4.2 can invest €40,000 in production and use 85 hours of labor. To manufacture one unit of product #1 requires 15 minutes of labor, and to manufacture one unit of product #2 requires 9 minutes of labor. The company wants to maximize its profit. How many units of product #1 and product #2 should it manufacture? What is the maximized profit? ■

Exercise 1.5.3

Sailco must determine how many sailboats to produce in the next 4 quarters. The demand is known to be 40, 60, 75, and 25 boats. Sailco must meet its demands. At the beginning of the 1st quarter Sailco starts with 10 boats in inventory. Sailco can produce up to 40 boats with regular time labor at \$400 per boat, or additional boats at \$450 with overtime labor. Boats made in a quarter can be used to meet that quarter's demand or held in inventory for the next quarter at an extra cost of \$20.00 per boat. Formulate the LP problem? ■

Exercise 1.5.4

CSL services computers. Its demand (hours) for the time of skilled technicians in the next 5 months is

t	Jan	Feb	Mar	Apr	May
d_t	6000	7000	8000	9500	11000

It starts with 50 skilled technicians at the beginning of January. Each technician can work 160hrs/ month. To train a new technician they must be supervised for 50hrs by an experienced technician for a period of one month time. Each experienced technician is paid 2K/mth and a trainee is paid 1K/mth. Each month 5% of the skilled technicians leave. CSL needs to meet demand and minimize costs. ■

Exercise 1.5.5

Reddy Mikks produces both interior and exterior paints from two raw materials, M1 and M2. The following table provides the basic data of the problem

	Tons of raw material per ton of		
	Exterior paint	Interior paint	Maximum daily availability (tons)
Raw material, $M1$	6	4	24
Raw material, $M2$	1	2	6
Profit per ton ($1000)	5	4	

■

Exercise 1.5.6

A company wants to produce a certain alloy containing 30% lead, 30% zinc, and 40% tin. This is to be done by mixing certain amounts of existing alloys that can be purchased at certain prices. The company wishes to minimize the cost. There are 9 available alloys with the following composition and prices.

Alloy	1	2	3	4	5	6	7	8	9	Blend
Lead (%)	20	50	30	30	30	60	40	10	10	30
Zinc (%)	30	40	20	40	30	30	50	30	10	30
Tin (%)	50	10	50	30	40	10	10	60	80	40
Cost ($\$/1b$)	7.3	6.9	7.3	7.5	7.6	6.0	5.8	4.3	4.1	minimize

■

Exercise 1.5.7 Suppose an industry is manufacturing tow types of products P1 and P2. The profits per Kg of the two products are €30 and €40 respectively. These two products require processing in three types of machines. The following table shows the available machine hours per day and the time required on each machine to produce one Kg of P1 and P2. Formulate the problem in the form of linear programming model.

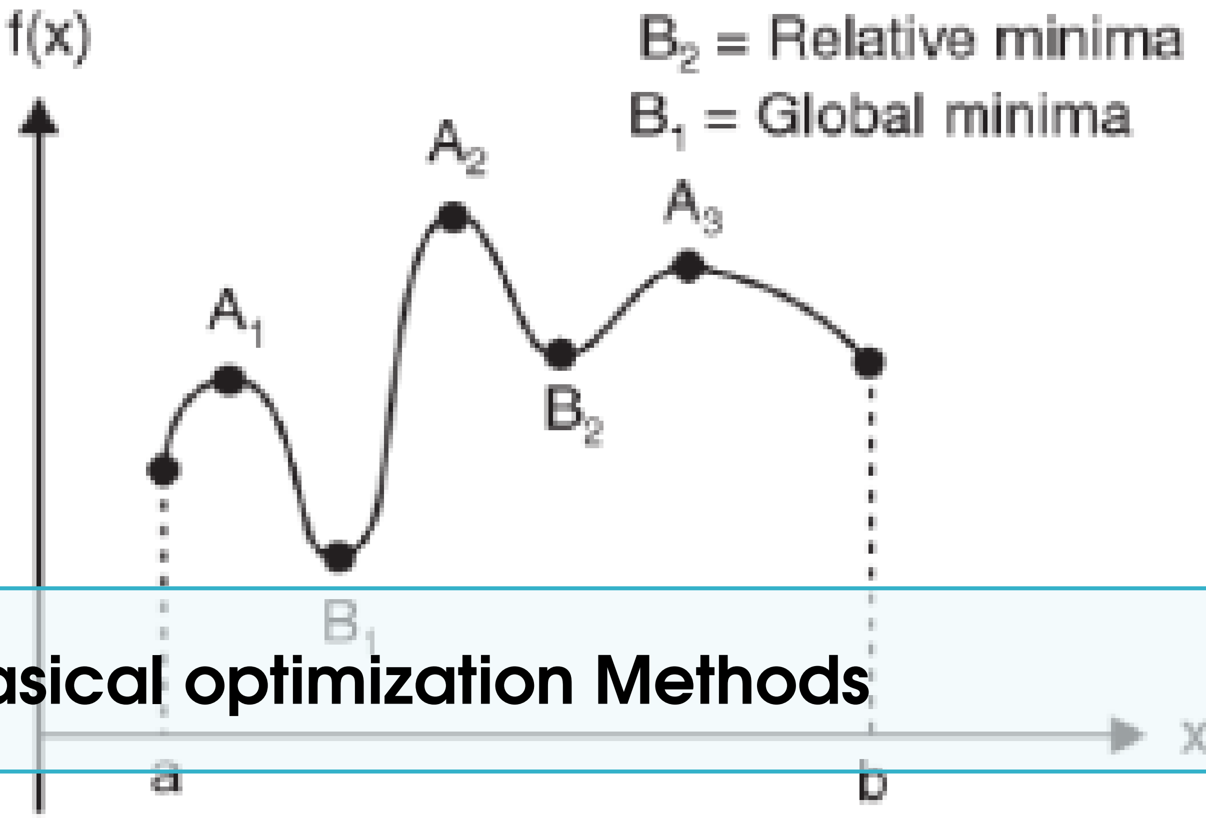
Profit/Kg	P 1 (€30)	P2 (€40)	Total available Machine (hours/day)
Machine 1	3	2	600
Machine 2	3	5	800
Machine 3	5	6	1100

■

Exercise 1.5.8 Furniture company manufactures four models of chairs. Each chair requires certain amount of raw materials (wood/steel)to make. The company wants to decide on a production that maximizes profit (assuming all produced chair are sold).The required and available amounts of materials are as follows

	Chair 1	Chair 2	Chair 3	Chair 4	Total available
Steel	1	1	3	9	4,4000(lbs)
Wood	4	9	7	2	6,000(lbs)
Profit	\$12	\$20	\$18	\$40	maximize

■



2. Clasical optimization Methods

This chapter deals with the classical analytical and numerical techniques for One-Dimensional unconstrained minimization problem as well as the classical methods for the multivariable optimization problems with no constrains and with equality consitrain.

2.1 1D unconstrained minimization problem

2.1.1 Introduction

As know from the previous chapter optimization problems consistent of an amount to be minimized or maximized that is the objective function and the constrains under which the problems is going to be solved. In the most practical problems, the optimum solution is known to lie within restricted ranges of the design variables. In some cases this range is not known, and hence the reach has to be made with no restrictions on the values of the variables.

Definition 2.1.1 **A unimodal function** is one that has only one peak (maximum) or valley (minimum) in a given interval as in figure2.1-1. Thus a function of one variable is said to be unimodal if, given that two values of the variable are on the same side of the optimum, the one nearer the optimum gives the better functional value (i.e., the smaller value in the case of a minimization problem). This can be stated mathematically as follows:
A function $f(x)$ is unimodal if $(i)x_1 < x_2 < x^*$ implies that $f(x_2) < f(x_1)$, and (ii) $x_2 > x_1 > x^*$ implies that $f(x_1) < f(x_2)$, where x^* is the minimum point.

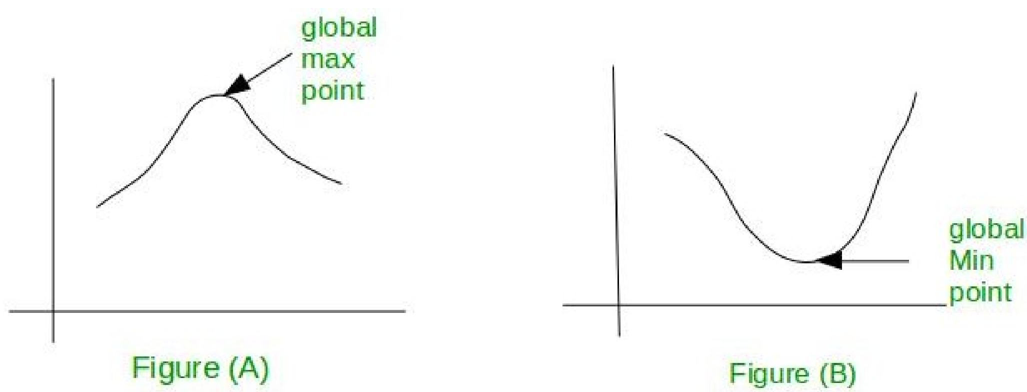


Figure 2.1-1: Sketch of unimodal function

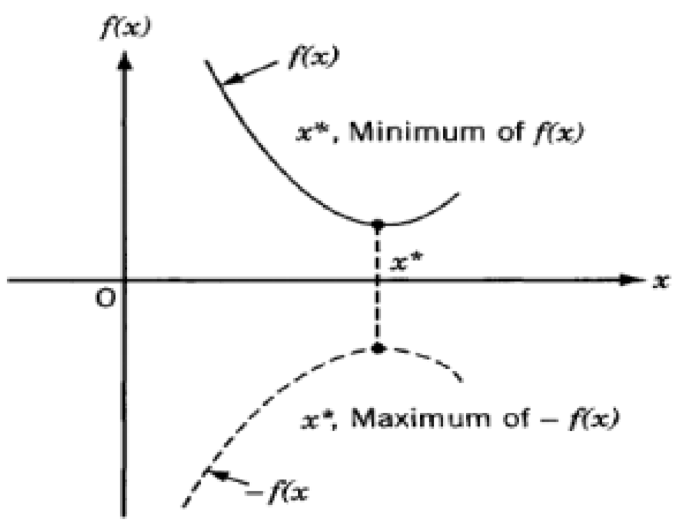


Figure 2.1-2: Minimum of $f(x)$ corresponds to maximum of $-f(x)$.

> It can be seen from the figure 2.1-2 that if a point x^* corresponds to the minimum value of a function $f(x)$, the same point also corresponds to the maximum value of the negative of the function $-f(x)$,

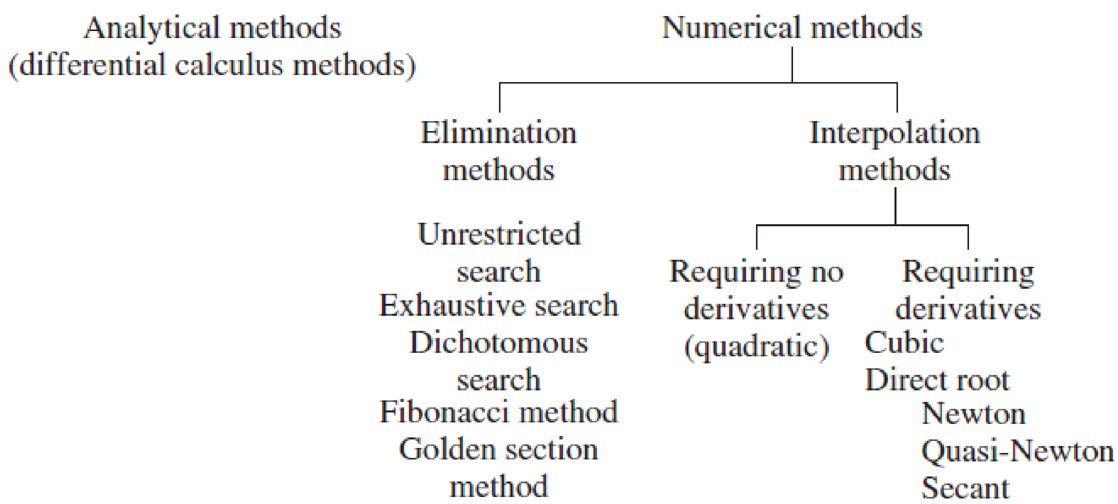


Figure 2.1-3: Flowchart of the various techniques for One-Dimensional unconstrained minimization problem

Referring to figure 2.1-3, there exists various techniques to treat a one-Dimensional unconstrained minimization problem as we see in the following sections.

2.1.2 Analytical approach for 1D unconstrained problem

Theorem 2.1.1 Necessary Condition

If a function $f(x)$ is defined in the interval $a < x < bf(x)$ and has a relative minimum at $x = x^*$, where $a < x^* < b$, and if the derivative $fracdf(x)dx = f\prime(x^*)$ exists as a finite number at x^*, then $f\prime(x^*) = 0$.

Theorem 2.1.2 Sufficient Condition

Let $f'(x^*) = f''(x^*) = \ldots = f^{(n-1)}(x^*) = 0$, but $f^{(n)}(x^*) \neq 0$. Then, $f(x^*)$ is

(i) A minimum value of $f(x^*)$ if $f^{(n)}(x^*) > 0$ and n is even;

(ii) A maximum value of $f(x^*)$ if $f^{(n)}(x^*) < 0$ and n is even;

(iii) Neither a maximum nor a minimum if n is odd.

Example 2.1.1 Using the Necessary and Sufficient Condition theorems, find the optimum values od the following function

$$f(x) = 12x^5 - 45x^4 + 40x^3 + 5$$

Solution 2.1.1

$$\begin{aligned} f'(x) &= 60x^4 - 3*60x^3 + 60*2*x^2 \\ &= 60x^2\left(x^2 - 3x + 2\right) \\ &= 60x^2(x-1)(x-2) = 0 \end{aligned}$$

The extreme points are $x = 0, x = 1$ and $x = 2$

$x = 0$	$x = 1$	$x = 2$
$f''(x) = 240x^3 - 540x^2 + 240x$ $f''(0) = 0$ We evaluate the next derivative $f'''(x) = 3*240x^2 - 2*540x$ 240 $f'''(0) = +240,$ Order of derivative is odd. So this point is neithe maximum nor minimum	$f''(1) = -60$ this point is relative maximum $f_{\text{Max}} = 12(1) - 45(1) +$ $40(1) + 5$ $= 12$	$f''(2) = 240$ this point is relative $f_{\text{Min}} = -11$

General algorithm to treat an optimization programming problem.

1. Start with an initial trial point $\mathbf{X}_1$.
2. Find a suitable direction $\mathbf{S}_i (i = 1$ to start with) which points in the general direction of the optimum.
3. Find an appropriate step length λ_i^* for movement along the direction $\mathbf{S}_i$.

4. Obtain the new approximation $\mathbf{X}_{i+1}$ as

$$\mathbf{x}_{i+1} = \mathbf{X}_i + \lambda_i^* \mathbf{S}_i$$

5. Test whether $\mathbf{X}_{i+1}$ is optimum. If $\mathbf{X}_{i+1}$ is optimum, stop the procedure. Otherwise, set a new $i = i + 1$ and repeat step (2) onward.

From this algorithm, we conclude that finding a minimum of single variable objective function is an important step (step3) in solving unconstrained multivariable optimization problem. So we start with studying unconstrained single optimization problem

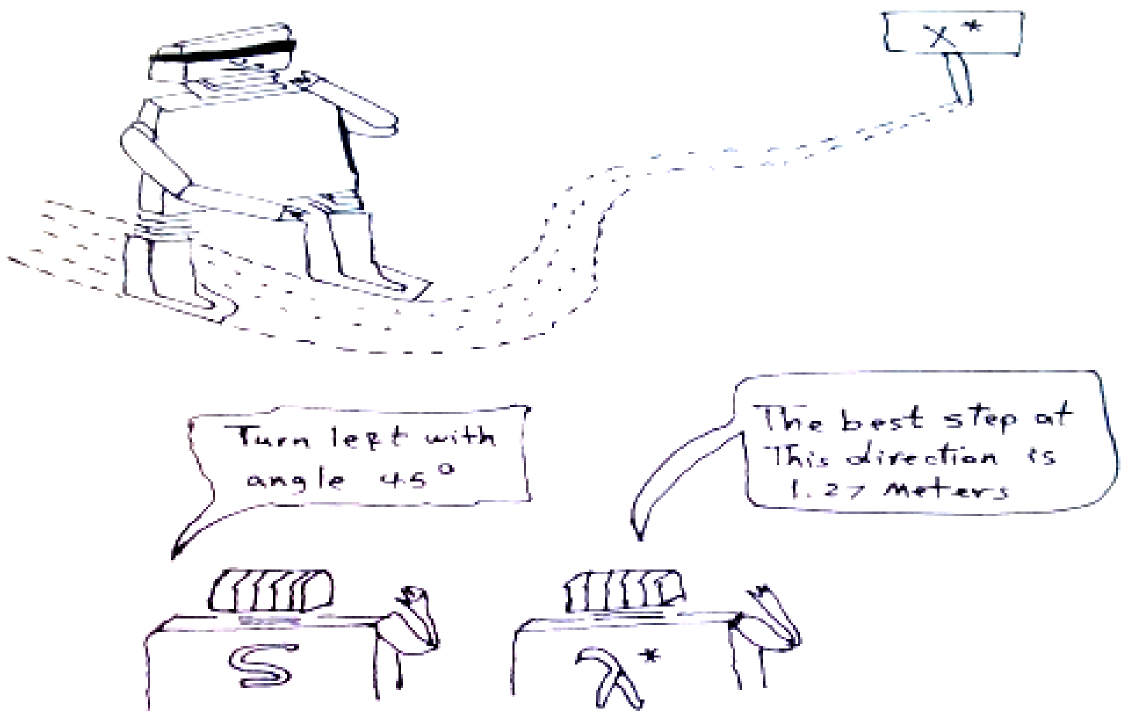


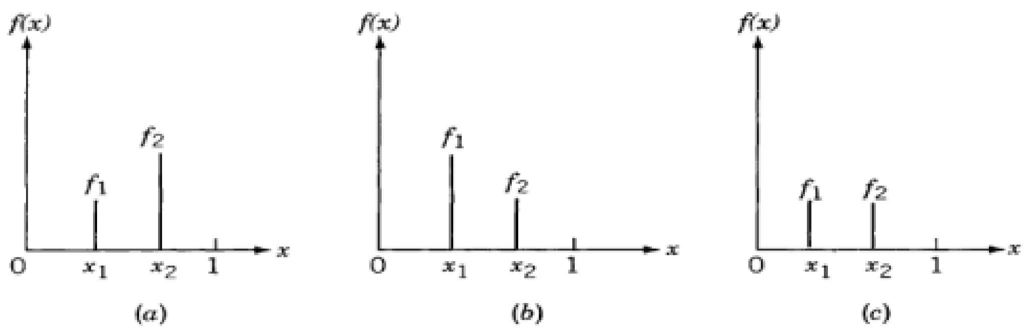
Figure 2.1-4: General algorithm to treat an optimization programming problem

2.1.3 Elimination methods

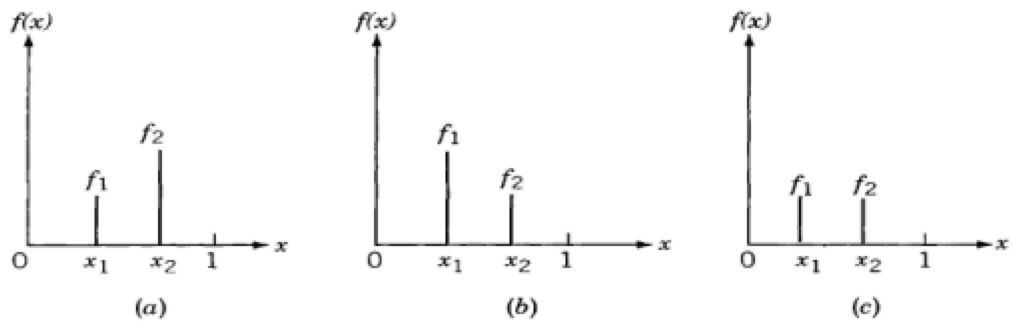
Search with fixed step size

The most elementary approach for such a problem is to use a fixed step size and move from an initial guess point in a favorable direction (positive or negative). The step size used must be small in the relation to the final accuracy desired. Although this method is very simple to implement, it is not efficient in many cases. This method is described in the following steps:

1. Start with an initial guess point, say, x_1.
2. Find $f_1 = f(x_1)$.
3. Assuming a step size s, find $x_2 = x_1 + s$.
4. Find $f_2 = f(x_2)$.
5. If $f_2 < f_1$, and if the problem is one of minimization, the assumption of unimodality indicates that the desired minimum cannot lie at $x < x_1$.



Hence the search can be continued further along points $x_3, x_4, \ldots$ using the unimodality assumption while testing each pair of experiments. This procedure is continued until a point, $x_i = x_1 + (i-1)s$, shows an increase in the function value.
6. The search is terminated at x_i, and either x_{i-1} or x_i can be taken as the optimum point.
7. Originally, if $f_2 > f_1$, the search should be carried in the reverse direction at points $x_{-2}, x_{-3}, \ldots$, where $x_{-j} = x_1 - (j-1)s$.
8. If $f_2 = f_1$, the desired minimum lies in between x_1 and x_2, and the minimum point can be taken as either x_1 or x_2.
9. If it happens that both f_2 and f_{-2} are greater than f_1, it implies that the desired minimum will lie in the double interval $x_{-2} < x < x_2$.



Example 2.1.2 Use unrestricted search with fixed step size to find the maximum of

$$f(x) = \begin{cases} \frac{1}{2}x, & x \leq 2 \\ 3-x, & x > 2 \end{cases}$$

by starting from $x_1 = 0$ with an initial step size of 0.4 . ■

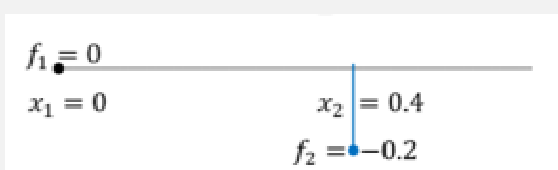
Solution 2.1.2 This problem corresponds to Find the minimum of

$$f(x) = \begin{cases} -0.5x; & x \leq 2 \\ x-3; & x > 2 \end{cases}$$

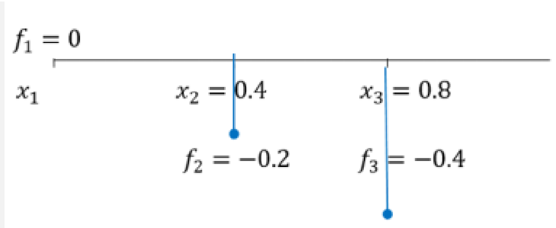
$$x_1 = 0, \quad f(x_1) = f(0) = 0, \; S = 0.4$$

$$x_2 = x_1 + S = 0.4 \qquad f(x) = \begin{cases} -\frac{1}{2}x, & x \leq 2 \\ 3-x, & x > 2 \end{cases}$$

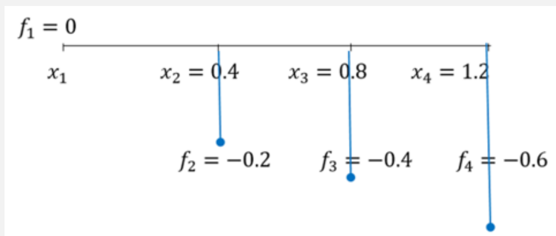
$$f(x_2) = f(0.4) = -\tfrac{1}{2}(0.4) = -0.2$$



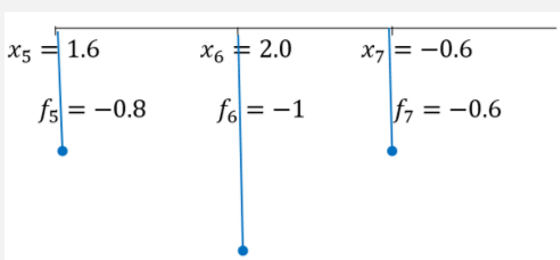
$$x_3 = x_2 + S = 0.4 + 0.4 = 0.8, \qquad f(x_3) = f(0.8) = -\frac{1}{2}(0.8) = -0.4$$



$$x_4 = x_3 + S = 0.8 + 0.4 = 1.2, \qquad f(x_4) = f(1.2) = -\frac{1}{2}(1.2) = -0.6$$



$$x_5 = x_4 + S = 1.2 + 0.4 = 1.6 \qquad f(x_5) = f(1.6) = -\frac{1}{2}(1.6) = -0.8$$
$$x_6 = x_5 + S = 1.6 + 0.4 = 2.0 \qquad f(x_6) = f(2.0) = -\frac{1}{2}(2.0) = -1$$
$$x_7 = x_6 + S = 2.0 + 0.4 = 2.4 \qquad f(x_7) = f(2.4) = 2.4 - 3 = -0.6$$



Thus, $x_6 = 2.0$ is the minimum point and $f(2.0) = -1$ ■

Fibonacci method

The Fibonacci method can be used to find the minimum of a function of one variable even if the function is not continuous. This method, like many other elimination methods, has the following limitations:

1. The initial interval of uncertainty, in which the optimum lies, has to be known.
2. The function being optimized has to be unimodal in the initial interval of uncertainty.
3. The exact optimum cannot be located in this method. Only an interval known as the final interval of uncertainty will be known. The final interval of uncertainty can be made as small as desired by using more computations.
4. The number of function evaluations to be used in the search or the resolution required has to be specified beforehand.

The Fibonacci sequence $\{F_n\}$ are defined as

$$F_0 = F_1 = 1$$

$$F_n = F_{n-1} + F_{n-2}, \qquad n = 2, 3, 4, \cdots$$

that yields the explicit sequence as$1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, \cdots$

Procedure

- Let L_0 be the initial interval of uncertainty defined by $a \leq x \leq b$ and n be the total number of experiments to be conducted.
- Define

$$L_2^* = \frac{F_{n-2}}{F_n} L_0$$

  and place the first two experiments at points x_1 and x_2, which are located at a distance of L_2^* from each end of L_0.
- This gives

$$x_1 = a + L_2^* = a + \frac{F_{n-2}}{F_n} L_0$$

$$x_2 = b - L_2^* = b - \frac{F_{n-2}}{F_n} L_0 = a + \frac{F_{n-1}}{F_n} L_0$$

- Discard part of the interval by using the unimodality assumption.
- Then there remains a smaller interval of uncertainty L_2 given by

$$L_2 = L_0 - L_2^* = L_0 \left(1 - \frac{F_{n-2}}{F_n}\right) = \frac{F_{n-1}}{F_n} L_0$$

and with one experiment left in it. This experiment will be at a distance of

$$L_2^* = \frac{F_{n-2}}{F_n} L_0 = \frac{F_{n-2}}{F_{n-1}} L_2$$

from one end and

$$L_2 - L_2^* = \frac{F_{n-3}}{F_n} L_0 = \frac{F_{n-3}}{F_{n-1}} L_2$$

from the other end.

- Now place the third experiment in the interval L_2 so that the current two experiments are located at a distance of

$$L_3^* = \frac{F_{n-3}}{F_n} L_0 = \frac{F_{n-3}}{F_{n-1}} L_2$$

from each end of the interval L_2.

- Again the unimodality property will allow us to reduce the interval of uncertainty to L_3 given by

$$L_3 = L_2 - L_3^* = L_2 - \frac{F_{n-3}}{F_{n-1}} L_2 = \frac{F_{n-2}}{F_{n-1}} L_2 = \frac{F_{n-2}}{F_n} L_0$$

This process of discarding a certain interval and placing a new experiment in the remaining interval can be continued, so that the location of the j th experiment and the interval of uncertainty at the end of j experiments are, respectively, given by

$$L_j^* = \frac{F_{n-j}}{F_{n-(j-2)}} L_{j-1}$$
$$L_j = \frac{F_{n-(j-1)}}{F_n} L_0$$

- The ratio of the interval of uncertainty remaining after conducting j of the n predetermined experiments to the initial interval of uncertainty becomes

$$\frac{L_j}{L_0} = \frac{F_{n-(j-1)}}{F_n}$$

and for $j = n$, we obtain

$$\frac{L_n}{L_0} = \frac{F_1}{F_n} = \frac{1}{F_n}$$

- The ratio L_n/L_0 will permit us to determine n, the required number of experiments, to achieve any desired accuracy in locating the optimum point.

Golden Section Method

The golden section method is same as the Fibonacci method except that in the Fibonacci method the total number of experiments to be conducted has to be specified before beginning the calculation, whereas this is not required in the golden section method. In the Fibonacci method, the location of the first two experiments is determined by the total number of experiments, N. In the golden section method we start with the assumption that we are going to conduct a large number of experiments. Of course, the total number of experiments can be decided during the computation.

Example 2.1.3 Deduce the best value for the eliminating part of the interval in Fibonacci method assuming we conduct a large number of iterations.

n	3	4	5	6	7
$\frac{F_{n-2}}{F_n}$	$\frac{f_1}{F_3}=\frac{1}{3}=0.33$	$\frac{F_2}{F_4}=\frac{2}{5}=0.4$	$\frac{F_3}{F_5}=\frac{3}{8}=0.37$	$\frac{F_4}{F_6}=\frac{5}{13}=0.382$	$\frac{F_5}{F_7}=\frac{8}{21}=0.382$

The Fibonacci sequence reads $1,1,2,3,5,8,13,21,\cdots$ The intervals of uncertainty remaining at the end of different number of experiments can be computed as follows:

$$L^* = \frac{F_{n-2}}{F_n} L_0$$

$$\lim_{n\to\infty} \frac{F_{n-2}}{F_n} = 0.382$$

■

Procedure

The procedure is same as the Fibonacci method except that the location of the first two experiments is defined by$L^* = 0.382L_0$ thus, 1- Let L_0 be the initial interval:$L_0 = [a,b]$

2- Define $L^* = 0.382L_0$

3- Put points of test to be $x_1 = a + L^*, x_2 = b - L^*$

4- Eliminate the non-desired part of the interval depending on the unimodality property

5- Define the new interval $L_o = [a,b]$, repeat steps 2-5 until a desired accuracy is obtained.

> In step 5, we can use one of the following accuracy formula: $|f(x_1) - f(x_2)| \leq \varepsilon$ Or $|L_o| \leq \varepsilon$ Where ε is small chosen value (such as 0.1).

2.1.4 MATLAB solution of one-dimensional minimization problems

The solution of one-dimensional minimization problems, using the MATLAB program *optimset*, is illustrated by the following example.

Example 2.1.4 Find the minimum of the following function:

$$f(x) = 0.65 - \frac{0.75}{1+x^2} - 0.65\tan^{-1}(\frac{1}{x})$$

Solution 2.1.3

Step 1: Write an M-file "objfun.m" for the objective function as

```
function f= objfun(x)
f= 0.65-(0.75/(1+x^2))-0.65*x*atan(1/x);
```

Step 2 : Invoke unconstrained optimization program (write this in new MATLAB file).

```
clc
clear all
warning off
options = optimset('LargeScale','off');
[x,fval] = fminbnd(@objfun,0,0.5,options)
```

This produces the solution or ouput as follows:

```
x=
0.4809
fval =
-0.3100
```

2.2 Multivariable Optimization

Here, we present the classical techniques for multivariable optimization. The necessary and sufficient conditions for the minimum or maximum of an unconstrained function of several variables are given. Finally, we present one way of solving multivariable optimization with equality constrained.

2.2.1 Multivariable optimization with no constraints

Theorem 2.2.1 Necessary Condition

If $f(\mathbf{X})$ has an extreme point (maximum or minimum) at $\mathbf{X} = \mathbf{X}^*$ and if the first partial derivatives of $f(\mathbf{X})$ exist at $\mathbf{X}^*$, then

$$\frac{\partial f}{\partial x_1}(\mathbf{X}^*) = \frac{\partial f}{\partial x_2}(\mathbf{X}^*) = \cdots = \frac{\partial f}{\partial x_n}(\mathbf{X}^*) = 0$$

Theorem 2.2.2 Sufficient Condition

A sufficient condition for a stationary point $\mathbf{X}^*$ to be an extreme point is that the matrix of second partial derivatives (Hessian matrix) of $f(\mathbf{X})$ evaluated at $\mathbf{X}^*$ is (i) positive definite when $\mathbf{X}^*$ is a relative minimum point, and (ii) negative definite when $\mathbf{X}^*$ is a relative maximum point.

Definition 2.2.1 A matrix A will be positive definite if all its eigenvalues are positive; that is, all the values of λ that satisfy the determinantal equation

$$|\mathbf{A} - \lambda \mathbf{I}| = 0$$

should be positive. Similarly, the matrix $[\mathbf{A}]$ will be negative definite if its eigenvalues are negative.

Definition 2.2.2 Saddle Point In the case of a function of two variables, $f(x,y)$, the Hessian matrix may be neither positive nor negative definite at a point (x^*, y^*) at which

$$\frac{\partial f}{\partial x} = \frac{\partial f}{\partial y} = 0$$

In such a case, the point (x^*, y^*) is called a saddle point.

Example 2.2.1 Find the extreme points of the function

$$f(x_1,x_2) = x_1^3 + x_2^3 + 2x_1^2 + 4x_2^2 + 6$$

■

Solution 2.2.1 The necessary conditions for the existence of an extreme point are

$$\frac{\partial f}{\partial x_1} = 3x_1^2 + 4x_1 = x_1(3x_1 + 4) = 0$$

$$\frac{\partial f}{\partial x_2 1} = 3x_2^2 + 8x_2 = x_2(3x_2 + 8) = 0$$

These equations are satisfied at the points

$$(0,0), \quad \left(0, -\frac{8}{3}\right), \quad \left(-\frac{4}{3}, 0\right), \quad \text{and} \quad \left(-\frac{4}{3}, -\frac{8}{3}\right)$$

■

2.2.2 Multivariable Optimization With Equality Constraints

In this section we consider the optimization of continuous functions subjected to equality constraints:

$$\text{Minimize } f = f(\mathbf{X})$$

$$\text{subject to}$$

$$g_j(\mathbf{X}) = 0, \quad j = 1, 2, \ldots, m$$

where

$$\mathbf{X} = [x_x, x, 2 \cdots, x_n]^\mathsf{T}$$

Lagrange multiplier method

The basic features of the Lagrange multiplier method is given initially for a simple problem of two variables with one constraint. The extension of the method to a general problem of n variables with m constraints is given later.

Problem with Two Variables and One Constraint. Consider the problem:

Minimize $f(x_1, x_2)$ subject to $g(x_1, x_2) = 0$ The necessary conditions generated by constructing a function L, known as the Lagrange function, as

$$L(x_1, x_2, \lambda) = f(x_1, x_2) + \lambda g(x_1, x_2)$$

by treating L as a function of the three variables x_1, x_2, and λ, the necessary conditions for its extremum are given by

$$\frac{\partial L}{\partial x_1}(x_1, x_2, \lambda) = \frac{\partial f}{\partial x_1}(x_1, x_2) + \lambda \frac{\partial g}{\partial x_1}(x_1, x_2) = 0$$
$$\frac{\partial L}{\partial x_2}(x_1, x_2, \lambda) = \frac{\partial f}{\partial x_2}(x_1, x_2) + \lambda \frac{\partial g}{\partial x_2}(x_1, x_2) = 0$$
$$\frac{\partial L}{\partial \lambda}(x_1, x_2, \lambda) = g(x_1, x_2) = 0$$

■ **Example 2.2.2** Using the Lagrange multiplier method, find the solution for:

$$\text{Minimize } f(x, y) = kx^{-1}y^{-2}$$

subject to

$$g(x, y) = x^2 + y^2 - a^2 = 0$$

■

■ **Solution 2.2.2** The Lagrange function is

$$L(x, y, \lambda) = f(x, y) + \lambda g(x, y) = kx^{-1}y^{-2} + \lambda\left(x^2 + y^2 - a^2\right)$$

The necessary conditions for the minimum of $f(x, y)$ give

$$\frac{\partial L}{\partial x} = -kx^{-2}y^{-2} + 2x\lambda = 0$$
$$\frac{\partial L}{\partial y} = -2kx^{-1}y^{-3} + 2y\lambda = 0$$
$$\frac{\partial L}{\partial \lambda} = x^2 + y^2 - a^2 = 0$$

From the first two equations, we have

$$2\lambda = \frac{k}{x^3y^2} = \frac{2k}{xy^4}$$

from which the relation $x^* = (1/\sqrt{2})y^*$ can be obtained. Hence,

$$x^* = \frac{a}{\sqrt{3}} \text{ and } y^* = \sqrt{2}\frac{a}{\sqrt{3}}$$

■

2.3 Exercises

Exercise 2.3.1 What are the limitations of classical methods in solving a one-dimensional minimization problem? ■

Exercise 2.3.2 Explain with graph the meaning of unimodal function and give its mathematical representation ■

Exercise 2.3.3 Explain the main core difference between the Fibonacci and golden section method to obtain the optimum value of unconstrained problem in one dimensional ■

Exercise 2.3.4 Use Fibonacci method and golden section method to find the maximum of

$$f(x) = \begin{cases} \frac{1}{2}x, & x \leq 2 \\ 3 - x, & x > 2 \end{cases}$$

by starting from $[0, 3]$ with $n = 6$.

■

Exercise 2.3.5 Find the minimum of the function $f(x) = x^3 + x^2 - x - 2$ in the interval -4 and 4 using MATLAB ■

Exercise 2.3.6 Using MATLAB, find the minimum of $f(x) = x(x - 1.5)$ in the interval $(0, 1)$ ■

3. Linear programming problem

3.1 Standard or canonical form of a linear programming problem

The general linear programming problem can be stated in the following standard forms:

Scalar Form

$$\text{Minimize } f(x_1, x_2, \ldots, x_n) = c_1x_1 + c_2x_2 + \cdots + c_nx_n$$

subject to the constraints

$$\begin{aligned}
a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n &= b_1 \\
a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n &= b_2 \\
&\vdots \\
a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n &= b_m \\
x_1 &\geq 0 \\
x_2 &\geq 0 \\
&\vdots \\
x_n &\geq 0
\end{aligned}$$

where c_j, b_j, and $a_{ij} (i = 1,2,\ldots,m; j = 1,2,\ldots,n)$ are known constants, and x_j are the decision variables. **Matrix Form**

$$\text{Minimize } f(\mathbf{X}) = \mathbf{c}^{\mathrm{T}}\mathbf{X}$$

subject to the constraints

$$\mathbf{aX} = \mathbf{b}$$

$$\mathbf{X} \geq \mathbf{0}$$

where,

$$\mathbf{x} = \begin{Bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{Bmatrix}, \quad \mathbf{b} = \begin{Bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{Bmatrix}, \quad \mathbf{c} = \begin{Bmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{Bmatrix},$$

$$\mathbf{a} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & & & \end{bmatrix}$$

The characteristics of a linear programming problem, stated in standard form, are

1. The objective function is of the minimization type.
2. All the constraints are of the equality type.
3. All the decision variables are nonnegative.

Thus, it is now shown that any linear programming problem can be expressed in standard form by using the following transformations.

1. **Objective function:** The maximization of a function $f(x_1, x_2, \ldots, x_n)$ is equivalent to the minimization of the negative of the same function. For example, the objective function minimize $f = c_1x_1 + c_2x_2 + \cdots + c_nx_n$ is equivalent to

$$\text{maximize } f' = -f = -c_1x_1 - c_2x_2 - \cdots - c_nx_n$$

Consequently, the objective function can be stated in the minimization form in any linear programming problem.

2. **Decision variables:** In most engineering optimization problems, the decision variables represent some physical dimensions, and hence the variables x_j will be nonnegative. However, a variable may be unrestricted in sign in some problems. In such cases, an unrestricted variable (which can take a positive, negative, or zero value) can be written as the difference of two nonnegative variables. Thus if x_j is unrestricted in sign, it can be written as $x_j = x'_j - x''_j$, where

$$x'_j \geq 0 \text{ and } x''_j \geq 0$$

It can be seen that x_f will be negative, zero, or positive, depending on whether x_j^N is greater than, equal to, or less than x_j^y.

3. **Constrains:** If a constraint appears in the form of a "less than or equal to" type of inequality as

$$a_{k1}x_1 + a_{k2}x_2 + \cdots + a_{ke}x_n \leq b_k$$

it can be converted into the equality form by adding a nonnegative **slack variable** x_{n+1} as follows:

$$a_{k1}x_1 + a_{k2}x_2 + \cdots + a_{ke}x_k + x_{n+1} = b_k$$

Similarly, if the constraint is in the form of a "greater than or equal to" type of inequality as

$$a_{k1}x_1 + a_{k2}x_2 + \cdots + a_{kn}x_n \geq b_k$$

it can be converted into the equality form by subtracting a variable as

$$a_{k1}x_1 + a_{k2}x_2 + \cdots + a_{kn}x_n - x_{n+1} = b_k$$

where x_{n+1} is a nonnegative variable known as a **surplus variable**.

Now, it can be seen that

- There are m equations in n decision variables in a linear programming problem.
- We can assume that $m < n$; for if $m > n$, there would be $m - n$ redundant equations that could be eliminated.

- The case $n = m$ is of no interest, for then there is either a unique solution $\mathbf{X}$ that satisfies the constraints and sign equations given above(in which case there can be no optimization) or no solution, in which case the constraints are inconsistent.
- The case $m < n$ corresponds to an underdetermined set of linear equations, which, if they have one solution, have an infinite number of solutions.

Hence, we have the following theorem :

Theorem 3.1.1 Every linear program has either

1. a unique optimal solution, or
2. multiple (infinity) optimal solutions, or
3. is infeasible (has no feasible solution), or
4. is unbounded (no feasible solution is maximal).

3.2 Geometrical solution of linear programming problems

Before we start with the geometrical solution of the LP problem we give some mathematical distentions that is needed for completely drawing the full picture.

3.3 Relevant Definitions

Point in n-Dimensional Space:

$$(x_1, x_2, \ldots, x_n).$$

Line Segment in n**-Dimensions (L):** If the coordinates of two points A and B are given by $x_j^{(1)}$ and $x_j^{(2)}$ $(j = 1, 2, \ldots, n)$, the line segment (L) joining these points is the collection of points $\mathbf{X}(\lambda)$ whose coordinates are given by $x_j = \lambda x_j^{(1)} + (1-\lambda)x_j^{(2)}$, $j = 1, 2, \ldots, n$, with $0 \leq \lambda \leq 1$. Thus

$$L = \left\{ \mathbf{X} \mid \mathbf{X} = \lambda \mathbf{X}^{(1)} + (1-\lambda)\mathbf{X}^{(2)} \right\}$$

In one dimension, for example, it is easy to see that the definition is in accordance with our experience (Fig. 3.3-1):

$$x^{(2)} - x(\lambda) = \lambda \left[x^{(2)} - x^{(1)} \right], \quad 0 \leq \lambda \leq 1$$

Figure Line segment. whence

$$x(\lambda) = \lambda x^{(1)} + (1-\lambda)x^{(2)}, \quad 0 \le \lambda \le 1$$

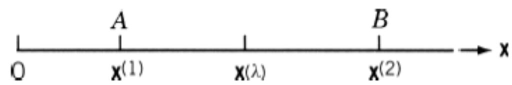


Figure 3.3-1: Line segment in n-dimensional

Convex set: A convex set is a collection of points such that if $\mathbf{X}^{(1)}$ and $\mathbf{X}^{(2)}$ are any two points in the collection, the line segment joining them is also in the collection. A convex set, S, can be defined mathematically as follows: If $\mathbf{X}^{(1)}, \mathbf{X}^{(2)} \in S$, then $\mathbf{X} \in S$ where

$$\mathbf{X} = \lambda \mathbf{X}^{(1)} + (1-\lambda)\mathbf{X}^{(2)}, \quad 0 \le \lambda \le 1$$

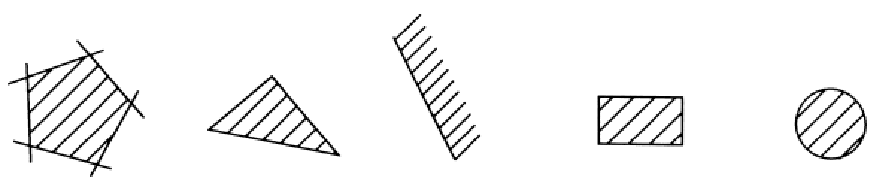


Figure 3.3-2: Convex Sets

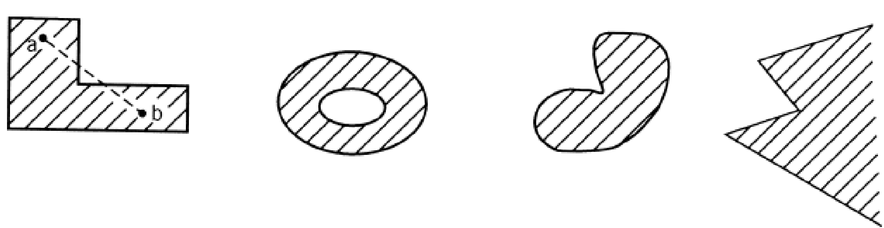


Figure 3.3-3: NonConvex Sets

Vertex, corner or Extreme Point This is a point in the convex set that does not lie on a line segment joining two other points of the set. For example, every point on the circumference of a circle and each corner point of a polygon can be called a vertex or extreme point.

Feasible solution: In a linear programming problem, any solution that satisfies the constraints

$$a_{ij}x_j = b_j, \qquad x_j \geq 0$$

is called a feasible solution.

Basic solution: A basic solution is one in which $n-m$ variables are set equal to zero. A basic solution can be obtained by setting $n-m$ variables to zero and solving the constraint Eqs. (3.2) simultaneously.

Basis: The collection of variables not set equal to zero to obtain the basic solution is called the basis.

Basic Feasible Solution This is a basic solution that satisfies the non negativity conditions of the problem

$$a_{ij}x_j = b_j, \qquad x_j \geq 0$$

Optimal Solution: A feasible solution that optimizes the objective function is called an optimal solution.

Optimal Basic Solution This is a basic feasible solution for which the objective function is optimal.

3.4 Graphical method

- Any LP with only two variables can be solved graphically (with possible solution as in theorem 3.1.1)
- The following characteristics can be noted from the graphical solution:

Theorem 3.4.1

(a) The feasible region is a convex polygon.

(b) If a linear program has an optimal solution, then it also has an optimal solution that is a corner point of the feasible region.

■ Example 3.4.1

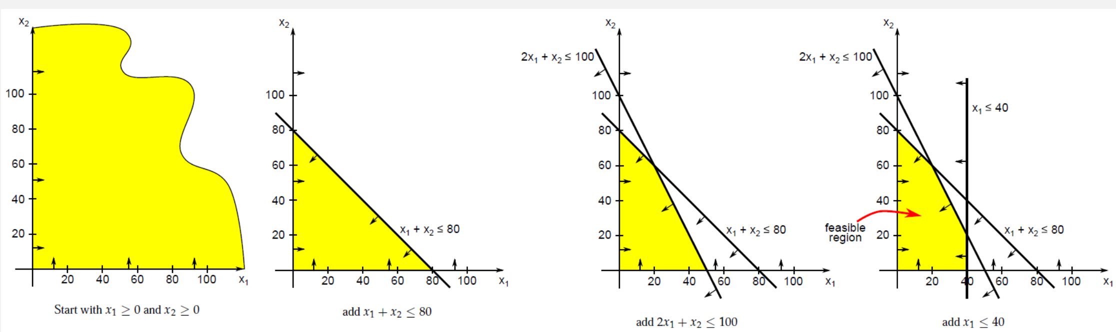
Find the optimum solution of the following LP problem using graphical method

$$\text{Max } 3x_1 + 2x_2$$
$$x_1 + x_2 \leq 80$$
$$2x_1 + x_2 \leq 100$$
$$x_1 \leq 40$$
$$x_1,\ x_2 \geq 0$$

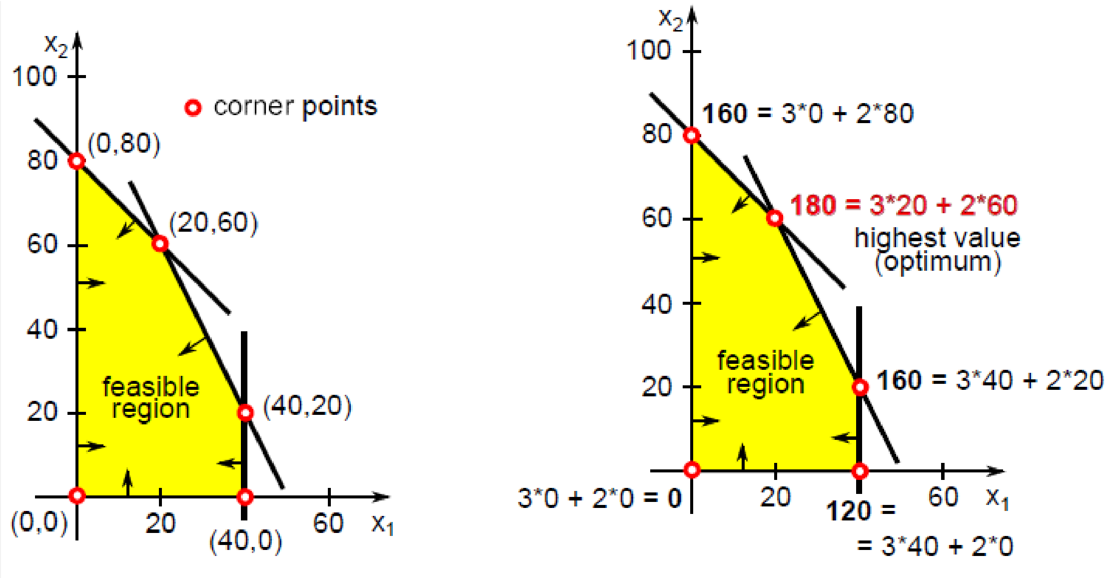
■

■ Solution 3.4.1

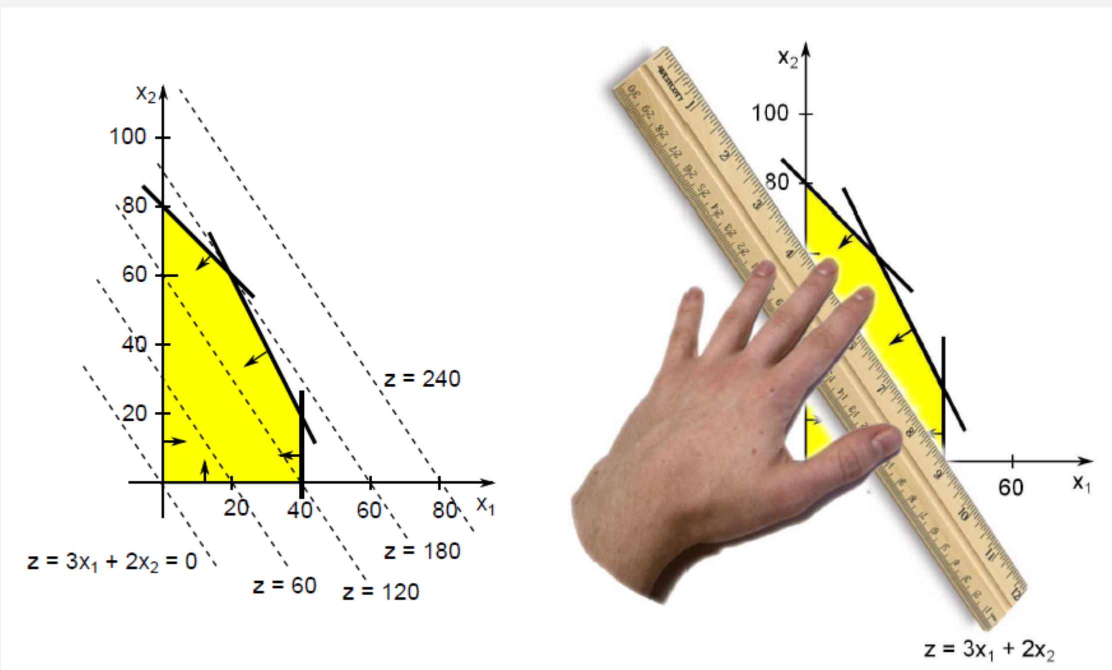
1. First, we have to find the feasible region
   - Plot each constraint as an equation $\equiv$ line in the plane
   - Feasible points on one side of the line – plug in $(0,0)$ to find out which



2. Find all corner points. Evaluate the objective function at those point since the optimum solution exists at those points according to theorem 3.4.1.



3. **Iso-value** line ≡ all points on this line the objective function has the same value: For our objective $3x_1 + 2x_2$ an iso-value line consists of points satisfying $3x_1 + 2x_2 = z$ where z is some number.



Optimal solution is $(x_1, x_2) = (20, 60)$.

Observe that this point is the intersection of two lines forming the boundary of the feasible region. Recall that lines we use to construct the feasible region come from inequalities (the points on the line satisfy the particular inequality with equality).

Binding constraint $\equiv$ constraint satisfied with equality

■

From this example, the **Main steps of Graphical Method** are

1. Find the feasible region.
2. Plot an iso-value (isoprofit, isocost) line for some value.
3. Slide the line in the direction of increasing value until it only touches the region.
4. Read-off an optimal solution.

■ Example 3.4.2 Find the solution of the following LP problem graphically:

Maximize $f(x,\ y) = 3x+y+2$,

Subject to $2x+y+9 \geq 0, 3y-x+6 \geq 0, x+2y \leq 3, y \leq x+3$ ■

■ Solution 3.4.2 1. Find the feasible region.

$$2x+y+9=0$$

$$2x+y=-9 \Longrightarrow x=0,\ y=-9$$

$$y=0,\ x=-4.5$$

$(0,0)$ Satisfies it, so the proposed area is up right the line

$$3y-x+6 \geq 0$$

$$3y-x=-6 \Longrightarrow x=0,\ y=-2$$

$$y=0,\ x=6$$

$(0,0)$ Satisfies it, so the proposed area is up Left the line

$$x+2y=3 \Longrightarrow x=0, y=1.5$$

$$y = 0,\ x = 3$$

$(0,0)$satisfies it, so the proposed area is Down Left the line

$$y - x = 3 \Longrightarrow x = 0,\ y = 3$$

$$y = 0,\ x = -3$$

$(0,0)$ satisfies it, so the proposed area is Down Wright the line

2. Corner or vertex

- The intersection of

$$2x + y = -9, 3y - x = -6$$

is obtained by solving these two eqs. to obtain

$$x = -3,\ y = -3$$

$f(at\ A) = 3(-3) + (-3) + 2 = -10$

- The intersection of

$$3y - x = -6, x + 2y = 3$$

is obtained by solving these two eqs. To obtain

$$x = 4.5,\ y = -0.6$$

$f(at\ B) = 3(4.5) + (-0.6) + 2 = 14$

- The intersection of

$$y - x = 3, x + 2y = 3$$

is obtained by solving these two eqs. to obtain

$$x = -1,\ y = 2$$

$f(at\ C) = 3(-1) + (2) + 2 = 1$

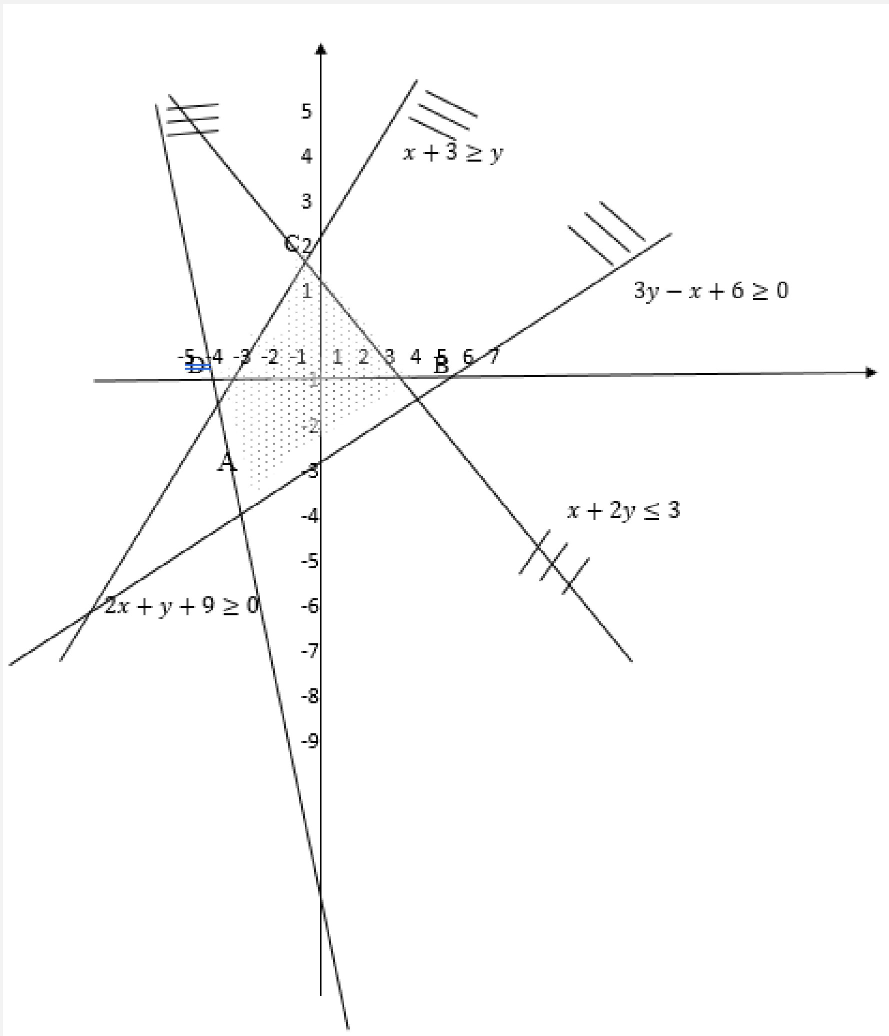
- The intersection of

$$y + x = 3, 2x + y = -9$$

is obtained by solving these two eqs. to obtain

$$x = -4,\ y = -12$$

$f(x,\ y) = 3x + y + 2$



Thus,

$f_A = -10 \quad at\ A(-3,-3)$

$f_C = 1 \quad at\ C(-1,2)$

$f_B = 14 \quad at\ B(4.2,-0.6)$

$f_D = -11 \quad at\ D(-4,-1)$

Hence the Maximum value is $f_B = 14$ at $B(4.2,-0.6)$

And the Minimum value is $f_D = -11$ at $D(-4,-1)$.

■

3.5 Simplex method

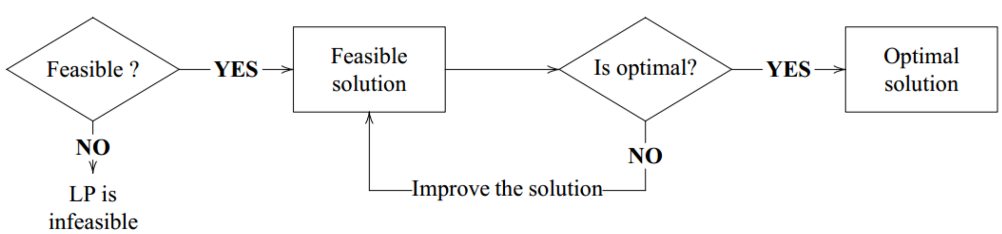
The Linear Programming with two variables can be solved graphically. The graphical method of solving linear programming problem is of limited application in the business problems as the number of variables is substantially large. If the linear programming problem has larger number of variables, the suitable method for solving is Simplex Method. The simplex method is an iterative process, through which it reaches ultimately to the minimum or maximum value of the objective function.

The simplex algorithm for solving linear programs (LP's) was developed by Dantzig in the late 1940's and since then a number of different versions of the algorithm have been developed. One of these later versions, called the revised simplex algorithm (sometimes known as the "product form of the inverse" simplex algorithm) forms the basis of most modern computer packages for solving LP's.

In General, the process consists of two steps

1- Find a feasible solution (or determine that none exists).

2- Improve the feasible solution to an optimal solution



Steps or procedures

1. Convert the LP to standard form
2. Obtain a basic feasible solution (bfs) from the standard form
3. Determine whether the current bfs is optimal. If it is optimal, stop.
4. If the current bfs is not optimal, determine which nonbasic variable should become a basic variable and which basic variable should become a nonbasic variable to find a new bfs with a better objective function value
5. Go back to Step 3.

Related concepts:

Standard form: all constraints are equations and all variables are nonnegative

Bfs: any basic solution where all variables are nonnegative

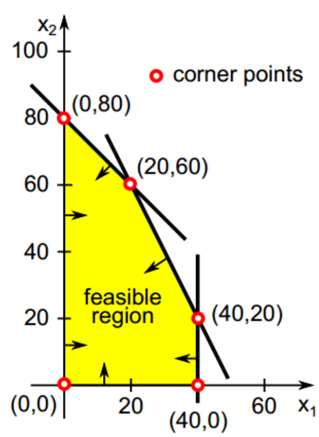
Nonbasic variable: a chosen set of variables where variables equal to 0

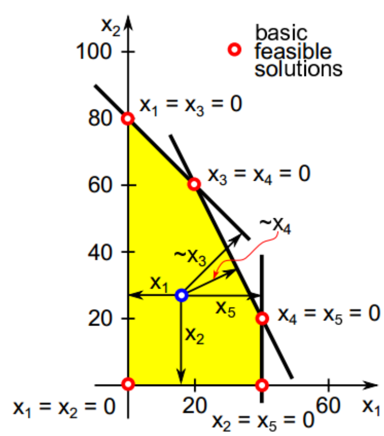
Basic variable: the remaining variables that satisfy the system of equations at the standard form.

3.5.1 Simplex method by example

Consider the toyshop example from earlier lectures. Convert to equalities by adding slack variables

$$\begin{array}{lrcl} \max & 3x_1 + 2x_2 & & \\ & x_1 + x_2 & \leq & 80 \\ & 2x_1 + x_2 & \leq & 100 \\ & x_1 & \leq & 40 \\ & x_1, x_2 & \geq & 0 \end{array} \quad \longrightarrow \quad \begin{array}{lrcl} \max & 3x_1 + 2x_2 & & \\ & x_1 + x_2 + x_3 & = & 80 \\ & 2x_1 + x_2 + x_4 & = & 100 \\ & x_1 + x_5 & = & 40 \\ & x_1, x_2, x_3, x_4, x_5 & \geq & 0 \end{array}$$





- **Starting feasible solution**

  - Set variables x_1, x_2 to zero and set slack variables to the values on the right-hand side. $\Longrightarrow$ yields a feasible solution

$$x_1 = x_2 = 0,\; x_3 = 80,\; x_4 = 100,\; x_5 = 40$$

  - Recall that the solution is feasible because all variables are **non-negative** and **satisfy** all equations.
  (we get a feasible solution right away because the right-hand side is non-negative; this may not always work)
  - Note something **interesting**: in this feasible solution two variables (namely x_1, x_2) are zero. Such a solution is called a **basic solution** of this problem, because the value of at least two variables is zero.

  > In a problem with n variables and m constraints, a solution where at least $(n-m)$ variables are zero is a **basic solution**

  A basic solution that is also feasible is called a **basic feasible solution (BFS).** The importance of basic solutions is revealed by the following observation.

  > **Basic solutions** are precisely the **corner points** of the **feasible region**

  - Recall that we have discussed that to find an optimal solution to an LP, it suffices to find a **best solution** among all **corner points**. The above tells us how to compute them - they are the **basic feasible solutions**.

  > A variable in a **basic solution** is called a non-basic variable if it is chosen to be zero. Otherwise, the variable is **basic**.

- **Dictionary**

  To conveniently deal with basic solutions, we use the so-called dictionary. A dictionary lists values of basic variables as a function of non-basic variables. The correspondence is obtained by expressing the basic variables from the initial set of equations. (We shall come back to this later; for now, have a look below.)

Express the slack variables from the individual equations

$$\begin{aligned} \max\ & 3x_1 + 2x_2 \\ & x_1 + x_2 + x_3 = 80 \\ & 2x_1 + x_2 + x_4 = 100 \\ & x_1 + x_5 = 40 \\ & x_1, x_2, x_3, x_4, x_5 \geq 0 \end{aligned} \implies$$

This is called a **dictionary**

$$\begin{aligned} x_3 &= 80 - x_1 - x_2 \\ x_4 &= 100 - 2x_1 - x_2 \\ x_5 &= 40 - x_1 \\ \hline z &= 0 + 3x_1 + 2x_2 \end{aligned}$$

- x_1, x_2 independent (**non-basic**) variables
- x_3, x_4, x_5 dependent (**basic**) variables
- $\{x_3, x_4, x_5\}$ is a **basis**

set $x_1 = x_2 = 0 \implies$ the corresponding (feasible) solution is $x_3 = 80, x_4 = 100, x_5 = 40$ with value $z = 0$

- **Improving the solution**

  -Try to increase x_1 from its current value 0 in hopes of improving the value of z

  - Try $x_1 = 20, x_2 = 0$ and substitute into the dictionary to obtain the values of x_3, x_4, x_5 and $z \implies x_3 = 60, x_4 = 60, x_5 = 20$ with value $z = 60 \to$ feasible

  -Try again $x_1 = 40, x_2 = 0 \implies x_3 = 40, x_4 = 20, x_5 = 0$ with value $z = 120 \to$ feasible

  - Now try $x_1 = 50, x_2 = 0 \implies x_3 = 30, x_4 = 0, x_5 = -10 \to$ not feasible

  How much we can increase x_1 before a (dependent) variable becomes negative?

  If $x_1 = t$ and $x_2 = 0$, then the solution is feasible if

$$\begin{aligned} x_3 &= 80 - t - 0 \geq 0 & & t \leq 80 \\ x_4 &= 100 - 2t - 0 \geq 0 & \implies & t \leq 50 \\ x_5 &= 40 - t \geq 0 & & t \leq 40 \end{aligned} \Bigg\} \Rightarrow t \leq 40$$

  Maximal value is $x_1 = 40$ at which point the variable x_5 becomes zero

  x_1 is incoming variable and x_5 is outgoing variable

  (we say that x_1 **enters** the dictionary/basis, and x_5 **leaves** the dictionary/basis)

- **Ratio test**

  The above analysis can be streamlined into the following simple "ratio" test.

$$x_3 : \frac{80}{1} = 80 \qquad x_4 : \frac{100}{2} = 50 \qquad x_5 : \frac{40}{1} = 40$$

$$\boxed{\begin{array}{rcl} x_3 & = & 80 - x_1 - x_2 \\ x_4 & = & 100 - 2x_1 - x_2 \\ x_5 & = & 40 - x_1 \\ \hline z & = & 0 + 3x_1 + 2x_2 \end{array}}$$

ratio for x_4:

$$\frac{100}{2} = 50$$

(**watch-out:** we only consider this ratio because the coefficient of x_1 is negative (-2)... more on that in the later steps)

Minimum achieved with $x_5 \Longrightarrow$ outgoing variable

Express x_1 from the equation for x_5

$$x_5 = 40 - x_1 \quad \longrightarrow \quad x_1 = 40 - x_5$$

Substitute x_1 to all other equations $\longrightarrow$ new feasible dictionary

$$\begin{array}{rcl} x_1 & = & (40 - x_5) \\ x_3 & = & 80 - (40 - x_5) - x_2 \\ x_4 & = & 100 - 2(40 - x_5) - x_2 \\ \hline z & = & 0 + 3(40 - x_5) + 2x_2 \end{array} \quad \longrightarrow \quad \begin{array}{rcl} x_1 & = & 40 - x_5 \\ x_3 & = & 40 - x_2 + x_5 \\ x_4 & = & 20 - x_2 + 2x_5 \\ \hline z & = & 120 + 2x_2 - 3x_5 \end{array}$$

now x_2, x_5 are independent variables and x_1, x_3, x_4 are dependent

$\rightarrow \{x_1, x_3, x_4\}$ is a basis

we repeat: we increase $x_2 \rightarrow$ incoming variable, ratio test:

x_1 : does not contain $x_2 \rightarrow$ no constraint

$$x_2 : \frac{40}{1} = 40$$

$$x_4 : \frac{20}{1} = 20$$

minimum achieved for $x_4 \rightarrow$ outgoing variable

$$x_4 = 20 - x_2 + 2x_5 \quad \longrightarrow \quad x_2 = 20 - x_4 + 2x_5$$

$$\begin{array}{rcl} x_1 & = & 40 - x_5 \\ x_2 & = & (20 - x_4 + 2x_5) \\ x_3 & = & 40 - (20 - x_4 + 2x_5) + x_5 \\ \hline z & = & 120 + 2(20 - x_4 + 2x_5) - 3x_5 \end{array} \quad \longrightarrow \quad \begin{array}{rcl} x_1 & = & 40 - x_5 \\ x_2 & = & 20 - x_4 + 2x_5 \\ x_3 & = & 20 + x_4 - x_5 \\ \hline z & = & 160 - 2x_4 + x_5 \end{array}$$

x_5 incoming variable, ratio test:

$$x_1 : \frac{40}{1} = 40$$

x_2 : positive coefficient $\rightarrow$ no constraint

$$x_3 : \frac{20}{1} = 20$$

minimum achieved for $x_3 \rightarrow$ outgoing variable

$$x_3 = 20 + x_4 - x_5 \quad \longrightarrow \quad x_5 = 20 + x_4 - x_3$$

$$\begin{array}{rcl} x_1 & = & 40 - (20 + x_4 - x_3) \\ x_2 & = & 20 - x_4 + 2(20 + x_4 - x_3) \\ x_5 & = & (20 + x_4 - x_3) \\ \hline z & = & 160 - 2x_4 + (20 + x_4 - x_3) \end{array} \quad \longrightarrow \quad \begin{array}{rcl} x_1 & = & 20 + x_3 - x_4 \\ x_2 & = & 60 - 2x_3 + x_4 \\ x_5 & = & 20 - x_3 + x_4 \\ \hline z & = & 180 - x_3 - x_4 \end{array}$$

no more improvement possible $\longrightarrow$ **optimal solution**

$$x_1 = 20, x_2 = 60, x_3 = 0, x_4 = 0, x_5 = 20 \text{ of value } z = 180$$

Why? setting x_3, x_4 to any non-zero values results in a smaller value of z

Each dictionary is **equivalent** to the original system (the two have the same set of solutions)

Simplex algorithm

Preparation: find a starting feasible solution/dictionary

1. Convert to the canonical form (constraints are equalities) by adding slack variables $x_{n+1}, \ldots, x_{n+m}$
2. Construct a starting dictionary - express slack variables and objective function z
3. If the resulting dictionary is feasible, then we are done with preparation
   If not, try to find a feasible dictionary using the **Phase I. method** (next lecture).

Simplex step (maximization LP): try to improve the solution

1. **(Optimality test)**: If **no variable** appears with a **positive** coefficient in the equation for z
   $\rightarrow$ STOP, current solution is **optimal**
   - set non-basic variables to zero
   - read off the values of the basic variables and the objective function z
     $\rightarrow$ Hint: the values are the constant terms in respective equations
   - report this (optimal) solution
2. Else pick a variable x_i having positive coefficient in the equation for z
   $x_i \equiv$ *incoming* variable
3. Ratio test: in the dictionary, find an equation for a variable x_j in which
   - x_i appears with a negative coefficient $-a$
   - the ratio $\frac{b}{a}$ is smallest possible
     (where b is the constant term in the equation for x_j)
4. If no such such x_j exists $\rightarrow$ stop, no optimal solution, report that **LP is unbounded**
5. Else $x_j \equiv$ outgoing variable $\rightarrow$ construct a new dictionary by *pivoting*:
   - express x_i from the equation for x_j,
   - add this as a new equation,
   - remove the equation for x_j,
   - substitute x_i to all other equations (including the one for z)
6. Repeat from 1.

Questions:

- which variable to choose as incoming, which as outgoing
- is this guaranteed to terminate in a finite number of steps
- how to convert other LP formulations to the standard form
- how to find a starting dictionary
- how do we find alternative optimal solutions

Example 3.5.1

Let x_1, X_2, x_3 be the number of desks, tables and chairs produced. Let the weekly profit be \$$z$. Then, we must

$$\max z = 60x_1 + 30x_2 + 20x_3$$
$$\text{s.t. } 8x_1 + 6x_2 + x_3 \leq 48$$
$$4x_1 + 2x_2 + 1.5x_3 \leq 20$$
$$2x_1 + 1.5x_2 + .5x_3 \leq 8$$
$$x_2 \leq 5$$
$$x_1, x_2, x_3 \geq 0$$

■

Solution 3.5.1

R_0	z	$-60x_1$	$-30x_2$	$-20x_3$					$=0$
R_1		$8x_1$	$+6x_2$	$+x_3$	$+s_1$				$=48$
R_2		$4x_1$	$+2x_2$	$+1.5x_3$		$+s_2$			$=20$
R_3		$2x_1$	$+1.5x_2$	$+.5x_3$			$+s_3$		$=8$
R_4			x_2					$+s_4$	$=5$

$x_1, x_2, x_3, s_1, s_2, s_3, s_4 \geq 0$

Obtain a starting bfs.

As $(x_1, x_2, x_3) = 0$ is feasible for the original problem, the below given point where three of the variables equal 0 (the **non-basic variables**) and the four other variables (the **basic variables**) are determined by the four equalities is an obvious bfs:

$$x_1 = x_2 = x_3 = 0, s_1 = 48, s_2 = 20, s_3 = 8, s_4 = 5.$$

Determine whether the current bfs is optimal.

Determine whether there is any way that z can be increased by increasing some nonbasic variable.

If each nonbasic variable has a nonnegative coefficient in the objective function row (**row**

0), current bfs is optimal.

However, here all nonbasic variables have negative coefficients: It is not optimal.

Find a new bfs

- z increases most rapidly when x_1 is made non-zero; i.e. x_1 is the **entering variable**.
- Examining R_1, x_1 can be increased only to 6 . More than 6 makes $s_1 < 0$.
  Similarly R_2, R_3, and R_4, give limits of 5,4 , and no limit for x_1 (**ratio test**). The smallest ratio is the largest value of the entering variable that will keep all the current basic variables nonnegative. Thus by R_3, x_1 can only increase to $x_1 = 4$ when s3 becomes 0 . We say s_3 is the **leaving variable** and R_3 is the **pivot equatio**n.
- Now we must rewrite the system so the values of the basic variables can be read off.
  The new pivot equation $(\mathrm{R}_3/2)$ is

$$\mathrm{R}_3' : x_1 + .75x_2 + .25x_3 + \quad .5\ \mathrm{s}_3 = 4$$

Then use R_3' to eliminate x_1 in all the other rows.

$$\mathrm{R}_0' = \mathrm{R}_0 + 60\mathrm{R}_3', \quad \mathrm{R}_1' = \mathrm{R}_1 - 8\mathrm{R}_3', \quad \mathrm{R}_2' = \mathrm{R}_2 - 4\mathrm{R}_3', \quad \mathrm{R}_4' = \mathrm{R}_4$$

R_0'	z		$+15x_2$	$-5x_3$			$+30s_3$		$= 240$	$\Longrightarrow z = 240$
R_1'				$-x_3$	$+s_1$		$-4s_3$		$= 16$	$\Longrightarrow s_1 = 16$
R_2'			$-x_2$	$+0.5x_3$		$+s_2$	$-2s_3$		$= 4$	$\Longrightarrow s_2 = 4$
R_3'		x_1	$+.75x_2$	$+.25x_3$			$+.5s_3$		$= 4$	$\Longrightarrow x_1 = 4$
R_4'			x_2					$+s_4$	$= 5$	$\Longrightarrow s_4 = 5$

The new bfs is $x_2 = x_3 = s_3 = 0, x_1 = 4, s_1 = 16, s_2 = 4, s_4 = 5$ making $z = 240$.

<u>Check optimality of current bfs. Repeat steps until an optimal solution is reached</u>

- We increase z fastest by making x_3 non-zero (i.e. x_3 enters).

- x_3 can be increased to at most $x_3 = 8$, when $s_2 = 0$ (i.e. s_2 leaves.)

Rearranging the pivot equation gives

$$\mathrm{R}_2'' \qquad -2x_2 + x_3 + 2s_2 - 4s_3 = 8 \quad \left(\mathrm{R}_2' \times 2\right).$$

Row operations with R_2'' eliminate x_3 to give the new system

$$R_0'' = R_0' + 5R_2'', \quad R_1'' = R_1' + R_2'', \quad R_3'' = R_3' - .5R_2'', \quad R_4'' = R_4'$$

The bfs is now $x_2 = s_2 = s_3 = 0, x_1 = 2, x_3 = 8, s_1 = 24, s_4 = 5$ making $z = 280$. Each nonbasic variable has a nonnegative coefficient in row $0\,(5x_2, 10s_2, 10s_3)$.

THE CURRENT SOLUTION IS OPTIMAL

Report: Dakota furniture's optimum weekly profit would be 280$ if they produce 2 desks and 8 chairs.

tableau format This was once written as a tableau.

$$\begin{aligned} \max z &= 60x_1 + 30x_2 + 20x_3 \\ \text{s.t.} \quad 8x_1 + 6x_2 + x_3 &\le 48 \\ 4x_1 + 2x_2 + 1.5x_3 &\le 20 \\ 2x_1 + 1.5x_2 + .5x_3 &\le 8 \\ x_2 &\le 5 \\ x_1, x_2, x_3 &\ge 0 \end{aligned}$$

Initial tableau:

z	x_1	x_2	x_3	s_1	s_2	s_3	s_4	RHS	BV	Ratio
1	-60	-30	-20	0	0	0	0	0	$z = 0$	
0	8	6	1	1	0	0	0	48	$s_1 = 48$	6
0	4	2	1.5	0	1	0	0	20	$s_2 = 20$	5
0	2	1.5	0.5	0	0	1	0	8	$s_3 = 8$	4
0	0	1	0	0	0	0	1	5	$s_4 = 5$	-

First tableau:

z	x_1	x_2	x_3	s_1	s_2	s_3	s_4	RHS	BV	Ratio
1	0	15	-5	0	0	30	0	240	$z = 240$	
0	0	0	-1	1	0	-4	0	16	$s_1 = 16$	-
0	0	-1	0.5	0	1	-2	0	4	$s_2 = 4$	8
0	1	0.75	0.25	0	0	0.5	0	4	$x_1 = 4$	16
0	0	1	0	0	0	0	1	5	$s_4 = 5$	-

Second and optimal tableau:

z	x_1	x_2	x_3	s_1	s_2	s_3	s_4	RHS	BV	Ratio
1	0	5	0	0	10	10	0	280	$z = 280$	
0	0	-2	0	1	2	-8	0	24	$s_1 = 24$	
0	0	-2	1	0	2	-4	0	8	$x_3 = 8$	
0	1	1.25	0	0	-0.5	1.5	0	2	$x_1 = 2$	
0	0	1	0	0	0	0	1	5	$s_4 = 5$	

■

■ **Example 3.5.2 Modified Dakota Furniture**

If we modify the objective function in the previous example to be

$\max z = 60x_1 + 35x_2 + 20x_3$

■

■ **Solution 3.5.2**

Second and optimal tableau for the modified problem:

z	x_1	x_2 ⇓	x_3	s_1	s_2	s_3	s_4	RHS	BV	Ratio
1	0	**0**	0	0	10	10	0	280	$z = 280$	
0	0	-2	0	1	2	-8	0	24	$s_1 = 24$	-
0	0	-2	1	0	2	-4	0	8	$x_3 = 8$	-
0	1	1.25	0	0	-0.5	1.5	0	2	$x_1 = 2$	2/1.25 $\Longrightarrow$
0	0	1	0	0	0	0	1	5	$s_4 = 5$	5/1

Another optimal tableau for the modified problem:

z	x_1	x_2	x_3	s_1	s_2	s_3	s_4	RHS	BV	Ratio
1	0	0	0	0	10	10	0	280	$z = 280$	
0	1.6	0	0	1	1.2	-5.6	0	27.2	$s_1 = 27$	
0	1.6	0	1	0	1.2	-1.6	0	11.2	$x_3 = 11.2$	
0	0.8	1	0	0	-0.4	1.2	0	1.6	$x_2 = 1.6$	
0	-0.8	0	0	0	0.4	-1.2	1	3.4	$s_4 = 3.4$	

Therefore the optimal solution is as follows: $z = 280$ and for $0 \leq c \leq 1$

$$\begin{vmatrix} x_1 \\ x_2 \\ x_3 \end{vmatrix} = c \begin{vmatrix} 2 \\ 0 \\ 8 \end{vmatrix} + (1-c) \begin{vmatrix} 0 \\ 1.6 \\ 11.2 \end{vmatrix} = \begin{vmatrix} 2c \\ 1.6 - 1.6c \\ 11.2 - 3.2c \end{vmatrix}$$

■

3.6 MATLAB solution of LP problems

The solution of linear programming problems, using simplex method, can be found as illustrated by the following example.

■ **Example 3.6.1** Find the solution of the following linear programming problem using MATLAB (simplex method):

$$
\begin{aligned}
\min z &= -x_1 - 2x_2 - x_3 \\
\text{s.t.} \quad 2x_1 + x_2 - x_3 &\leq 2 \\
2x_1 - 2x_2 + 5x_3 &\leq 6 \\
4x_1 + x_2 + x_3 &\leq 6 \\
x_1, x_2, x_3 &\geq 0
\end{aligned}
$$

■

■ Solution 3.6.1

Step 1 Express the objective function in the form $f(x) = f^T x$ and identify the vectors x and f as

$$x = \begin{Bmatrix} x_1 \\ x_2 \\ x_3 \end{Bmatrix} \text{ and } f = \begin{Bmatrix} -1 \\ -2 \\ -1 \end{Bmatrix}$$

Express the constraints in the form $Ax \leq b$ and identify the matrix A and the vector b as

$$A = \begin{bmatrix} 2 & 1 & -1 \\ 2 & -1 & 5 \\ 4 & 1 & 1 \end{bmatrix} \text{ and } b = \begin{Bmatrix} 2 \\ 6 \\ 6 \end{Bmatrix}$$

Step 2 Use the command for executing linear programming program using simplex method as indicated below:

```
clc
clear all
f=[-1;-2;-1];
A=[2 1 - 1;
2 -1 5;
4 1 1];
b=[2;6;6];
lb=zeros(3,1);
```

```
Aeq=[];
beq=[];
options = optimset('LargeScale', 'off', 'Simplex', 'on')
   ;
[x,fval,exitflag,output] = linprog(f,A,b,Aeq,beq,lb
   ,[],[],...
optimset('Display','iter'))
```

This produces the solution or output as follows:

```
Optimization terminated.
x=
0
4
2
fval =
-10
exitflag =
1
output =
iterations:3
algorithm: 'medium scale: simplex'
cgiterations: []
message: 'Optimization terminated.'
```

■

3.7 Exercises

Exercise 3.7.1 Define the following items

- surplus variable - slack variable ■

Exercise 3.7.2 what are the conditions that Linear Programming (LP) problem has to satisfy to be in the canonical form? ■

Exercise 3.7.3 Put the following LP problem in its stander matrix form

$$\begin{aligned} &\max z = 4x_1 + 7x_2 \\ &\text{s.t.} \qquad 2x_1 + 2x_2 \leq 100 \\ &\qquad\qquad x_1 + x_2 \leq 80 \\ &\qquad\qquad x_1 \quad \leq 40 \\ &\qquad\qquad x_2 \quad \geq 1 \\ &x_1 \geq 0 \end{aligned}$$

(Explain all the details and names of the addition variables if needed) ■

Exercise 3.7.4 Put the following LP problem in its stander scalar form

$$\begin{aligned} &\min z = 50x_1 + 100x_2 \\ &\text{s.t.} \\ &7x_1 + 2x_2 \geq 28 \\ &2x_1 + 12x_2 \geq 24 \\ &x_1, x_2 \geq 0 \end{aligned}$$

(Explain all the details and names of the addition variables if needed) ■

Exercise 3.7.5 Detect which of the following Mathematical statements is true and which is false.

1. The Matrix form of the standard form of linear programming problem is Minimize $f(X) = c^T X$, where c is unknown constant.

Exercise 3.7.6 Find the solution of the following LP problem graphically:

Maximize $f(x,\ y) = 50x + 100y$

subject to

$$10x + 5y \leq 2500$$

$$4x + 10y \leq 2000$$

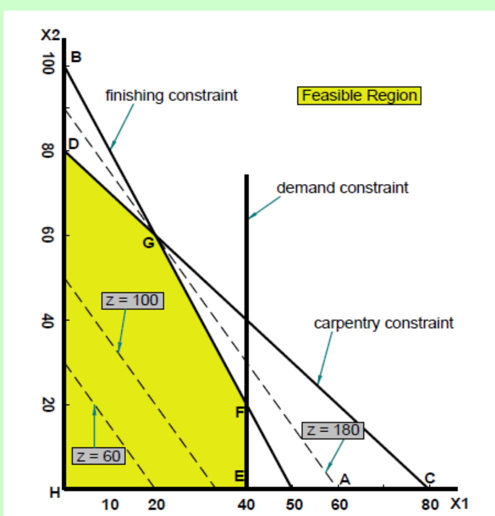
$$x + 1.5y \leq 450$$

$$x \geq 0,\ y \geq 0$$

Exercise 3.7.7 Find the solution of the following LP problem graphically (Giapetto LP):

$$\begin{array}{lll} \max z = 3x_1 + 2x_2 & \text{(The Objective function)} & \\ \text{s.t.} & 2x_1 + x_2 \leq 100 & \text{(Finishing constraint)} \\ & x_1 + x_2 \leq 80 & \text{(Carpentry constraint)} \\ & x_1 \quad \leq 40 & \text{(Constraint on demand for soldiers)} \\ x_1, x_2 \geq 0 & \text{(Sign restrictions)} & \end{array}$$

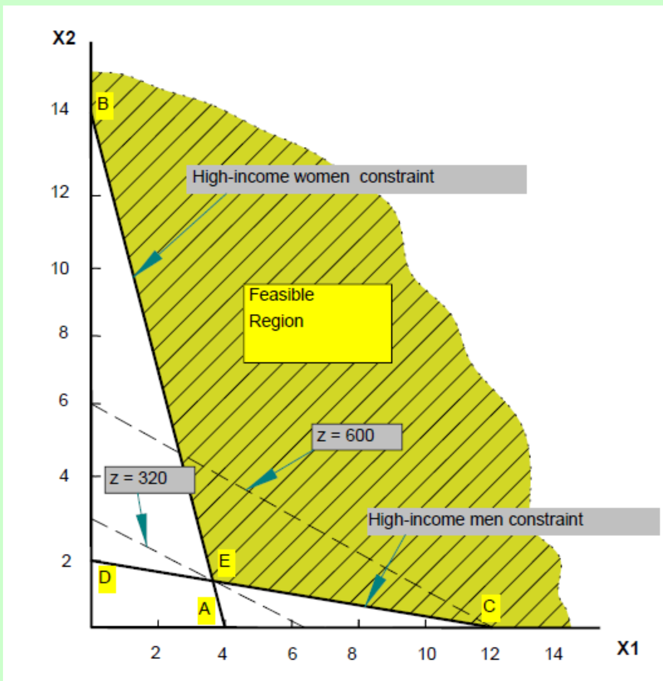
Hint:



Exercise 3.7.8 Find the solution of the following LP problem graphically (Advertisement LP):

$$\min z = 50x_1 + 100x_2$$
$$\text{s.t.}$$
$$7x_1 + 2x_2 \geq 28$$
$$2x_1 + 12x_2 \geq 24$$
$$x_1, x_2 \geq 0$$

Hint:



Exercise 3.7.9 Find the solution of the following LP problem graphically (Modified Giapetto LP):

$$\max z = 4x_1 + 2x_2 \quad \text{(The Objective function)}$$
$$\text{s.t.} \quad 2x_1 + x_2 \leq 100 \quad \text{(Finishing constraint)}$$
$$x_1 + x_2 \leq 80 \quad \text{(Carpentry constraint)}$$
$$x_1 \leq 40 \quad \text{(Constraint on demand for soldiers)}$$
$$x_1, x_2 \geq 0 \quad \text{(Sign restrictions)}$$

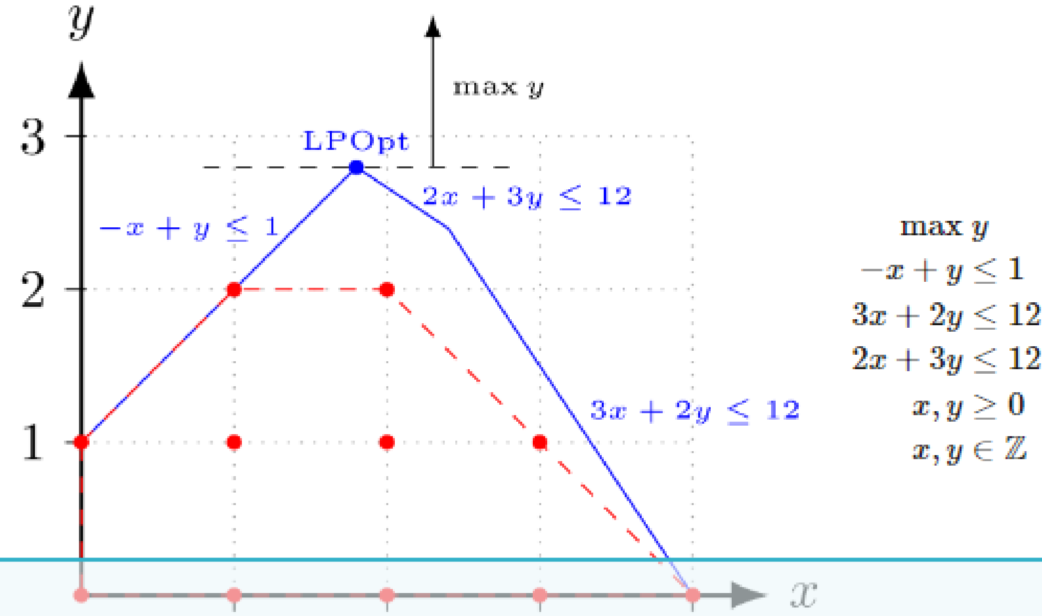
Exercise 3.7.10 Using SIMPLEX method, find

$$\max z = 4x_1 + 7x_2$$
$$\text{s.t.} \quad 2x_1 + 2x_2 \leq 100$$
$$x_1 + x_2 \leq 80$$
$$x_1 \leq 40$$
$$x_2 \geq 1$$
$$x_1 \geq 0$$

Exercise 3.7.11 Using SIMPLEX method, find

$$\max z = 10x_1 + 20x_2$$
$$\text{s.t.} \quad 5x_1 + 3x_2 \leq 30$$
$$3x_1 + 6x_2 \leq 36$$
$$2x_1 + 5x_2 \leq 20$$
$$x_1, x_2 \geq 0$$

Exercise 3.7.12 Using MATLAB solve the above LP problems based on SIMPLEX method



4. Integer Programming Problem

4.1 Introduction

In all the optimization techniques considered so far, the design variables are assumed to be continuous, which can take any real value. Other cases could be considered as follows:

- When all the variables are constrained to take *only integer values* in an optimization problem, it is called an **(all)-integer programming problem**.
- When the variables are restricted to take *only discrete values*, the problem is called a **discrete programming problem**.
- When *some variables* only are restricted to take *integer (discrete) values*, the optimization problem is called a **mixed-integer (discrete) programming problem** .
- When all the design variables of an optimization problem are allowed to take on values of either *zero or 1*, the problem is called **a zero-one programming problem**.

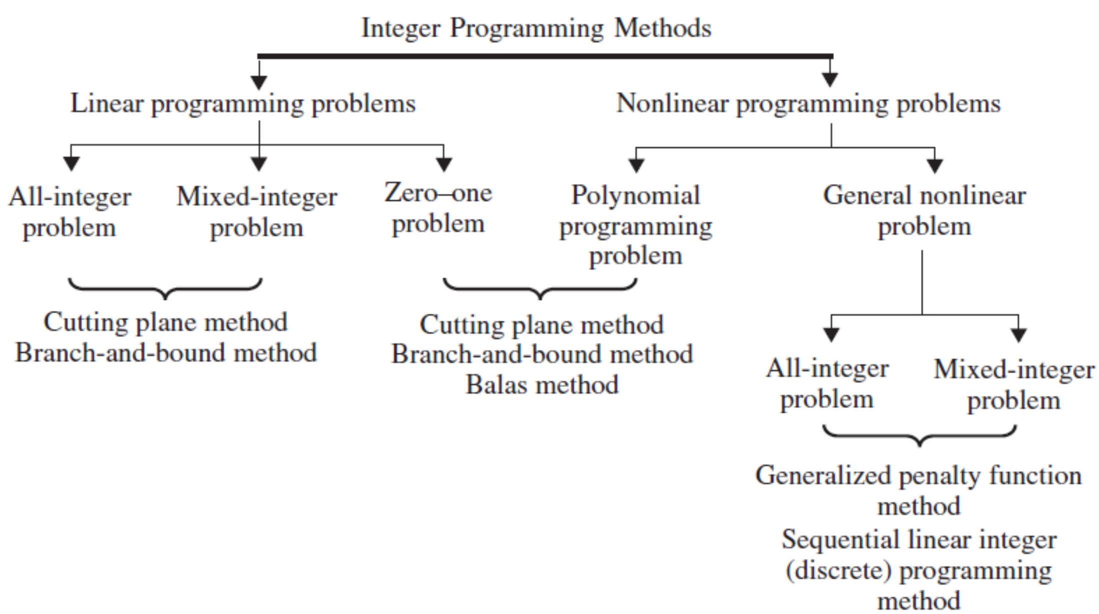
Solving various type of integral programming

- Among the several techniques available for solving the all-integer and mixed-integer linear programming problems, **the cutting plane algorithm of Geometry**
- The branch-and-bound algorithm of Land and Doig have been quite popular.
- The zero–one linear programming problems can be solved by the general cutting

plane or the branch-and-bound algorithm.

- Balas developed an efficient enumerative algorithm for solving the zero–one linear programming problems.
- Very little work has been done in the field of integer nonlinear programming. The generalized penalty function method and the sequential linear integer (discrete) programming method can be used tomethod and the sequential linear integer (discrete) programming method can be used to solve all integer and mixed-integer nonlinear programming problems.

The various solution techniques of solving integer programming problems are summarized in the following figure



4.2 Integer Linear Programming

4.2.1 Graphical representation

Consider the following integer programming problem:

$$\text{Maximize } f(\mathbf{X}) = 3x_1 + 4x_2$$

subject to

$$3x_1 - x_2 \leq 12$$

$$3x_1 + 11x_2 \leq 66$$

$$x_1 \geq 0$$

$$x_2 \geq 0$$

$$x_1 \text{ and } x_2 \text{ are integers}$$

The graphical solution of this problem, by ignoring the integer requirements, is shown in Fig. 4.2-1. It can be seen that the solution is $x_1 = 5\frac{1}{2}, x_2 = 4\frac{1}{2}$ with a value of $f = 34\frac{1}{2}$.

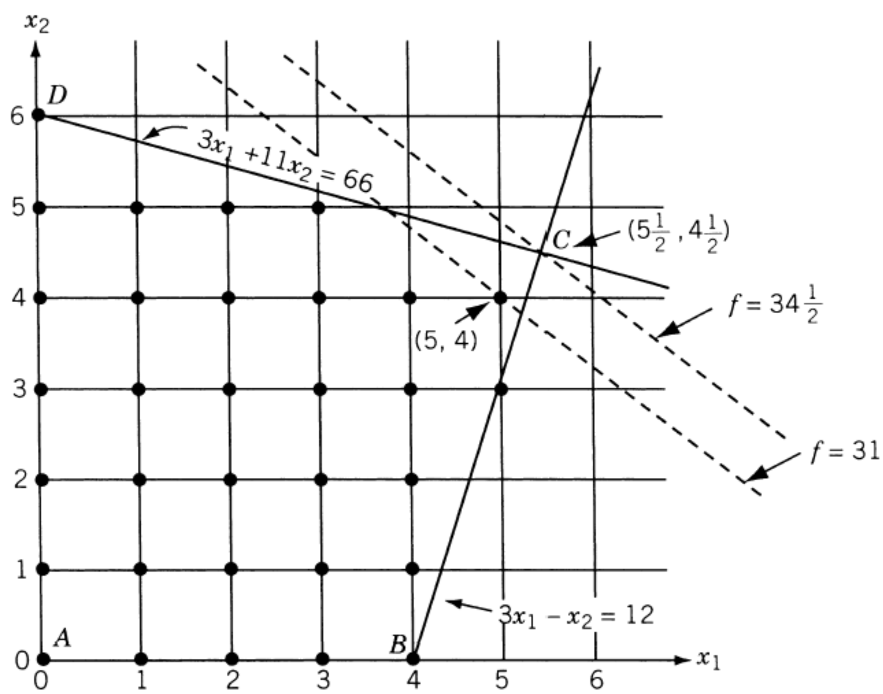


Figure 4.2-1: Graphical solution of the problem

Now, since this is a noninteger solution, we truncate the fractional parts and obtain the new solution as $x_1 = 5$, $x_2 = 4$, and $f = 31$. By comparing this solution with all other integer feasible solutions (shown by dots in Fig. 4.2-1), we find that this solution is optimum for the integer LP problem.

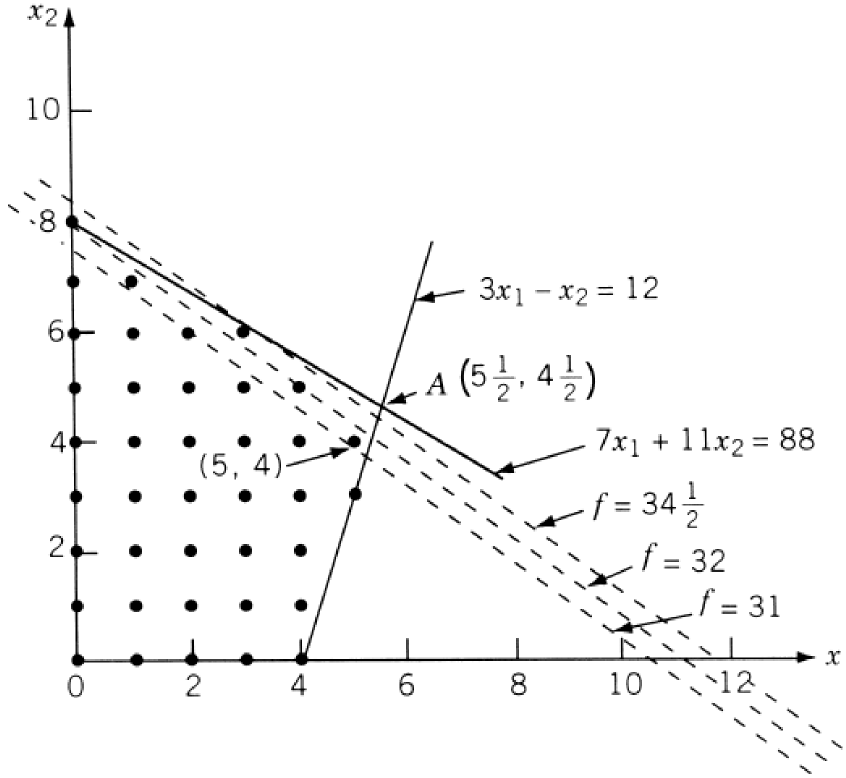


Figure 4.2-2: Graphical solution with modified constraint

It is to be noted that truncation of the fractional part of a LP problem will not always give the solution of the corresponding integer LP problem. This can be illustrated by changing the constraint

$$3x_1 + 11x_2 \leq 66 \text{ to } 7x_1 + 11x_2 \leq 88$$

With this altered constraint, the feasible region and the solution of the LP problem, without considering the integer requirement, are shown in Fig. 4.2-2. The optimum solution of this problem is identical with that of the preceding problem: namely, $x_1 = 5\frac{1}{2}, x_2 = 4\frac{1}{2}$, and $f = 34\frac{1}{2}$. The truncation of the fractional part of this solution gives $x1 = 5, x2 = 4$, and $f = 31$. Although this truncated solution happened to be optimum to the corresponding integer problem in the earlier case, it is not so in the present case. In this case the optimum solution of the integer programming problem is given by $x_1^* = 0, x_2^* = 8$, and $f^* = 32$.

4.3 Cutting Plane

Theorem 4.3.1 The optimal solution of the ILP problem lies at one corner of the closed convex polyhedron of the feasible region, or at the nearest integer point of the best corner.

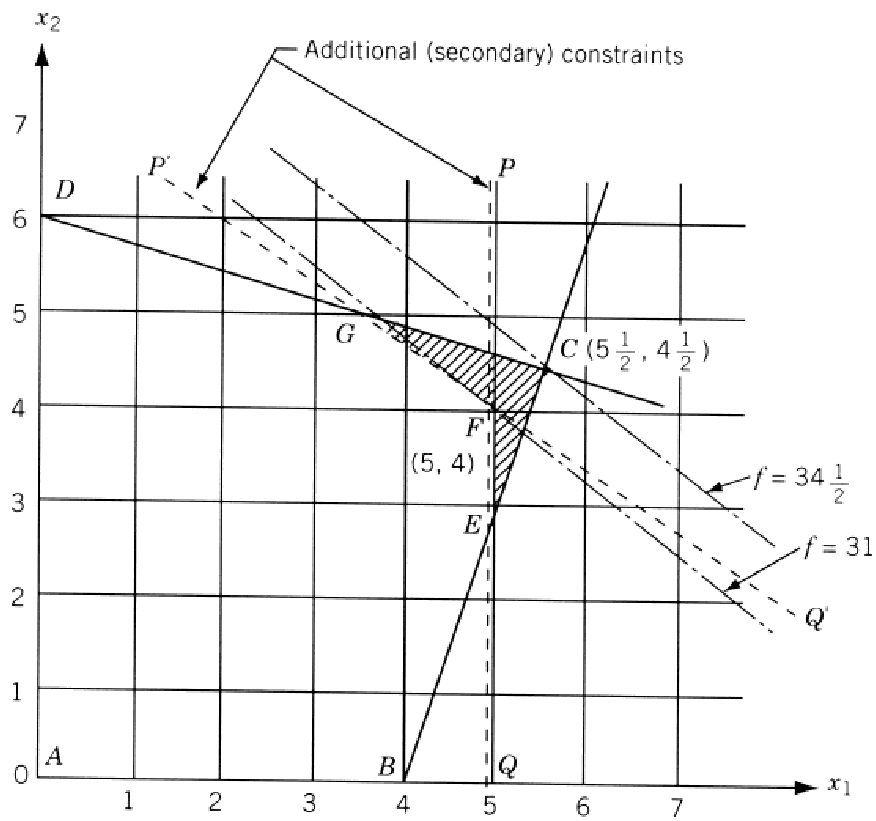


Figure 4.3-3: Graphical solution with modified constraint

- For the same problem stated in the previous section, the feasible region of the problem is denoted by ABCD in Fig. 4.2-1.
- The optimal solution of the problem, without considering the integer requirement, is given by point C. This point corresponds to $x_1 = 5\frac{1}{2}, x_2 = 4\frac{1}{2}$, and $f = 34\frac{1}{2}$, which is not optimal to the integer programming problem since the values of x_1 and x_2 are not integers.
- The feasible integer solutions of the problem are denoted by dots in Fig. 4.2-1. These points are called the integer **lattice** points.

- In Fig. 4.3-3, the original feasible region is reduced to a new feasible region ABEFGD by including the additional (arbitrarily selected) constraints.
- The idea behind adding these additional constraints is to reduce the original feasible convex region ABCD to a new feasible convex region (such as ABEFGD) such that an extreme point of the new feasible region becomes an integer optimal solution to the integer programming problem.
- There are two main considerations to be taken while selecting the additional constraints:

  (1) The new feasible region should also be a convex set, and

  (2) The part of the original feasible region that is sliced off because of the additional constraints should not include any feasible integer solutions of the original problem.

4.4 Balas' algorithm for zero-one programming problems

- When all the variables of a LP problem are constrained to take values of 0 or 1 only, we have a zero-one (or binary) LP problem. A study of the various techniques available for solving zero-one programming problems is important because of the following reasons:
  (a) Certain class of integer nonlinear programming problems can be converted into equivalent zero-one LP problems.
  (b) A wide variety of industrial, management, and engineering problems can be formulated as zero-one problems. For example, in structural control, the problem of selecting optimal locations of actuators (or dampers) can be formulated as a zero-one problem. In this case, if a variable is zero or 1, it indicates the absence or presence of the actuator, respectively, at a particular location.
- The zero-one LP problems can be solved by using any of the general integer LP techniques like Gomory's cutting plane method and Land and Doig's branch-and-bound method by introducing the additional constraint that all the variables must be less than or equal to 1.

- This additional constraint will restrict each of the variables to take a value of either zero (0) or one (1). Since the cutting plane and the branch-and-bound algorithms were developed primarily to solve a general integer LP problem, they do not take advantage of the special features of zero-one LP problems.
- Thus several methods have been proposed to solve zero-one LP problems more efficiently. In this section we present an algorithm developed by Balas (in 1965) for solving LP problems with binary variables only.
- If there are n binary variables in a problem, an explicit enumeration process will involve testing 2^n possible solutions against the stated constraints and the objective function. In Balas method, all the 2^n possible solutions are enumerated, explicitly or implicitly.
- The efficiency of the method arises out of the clever strategy it adopts in selecting only a few solutions for explicit enumeration.
- The method starts by setting all the n variables equal to zero and consists of a systematic procedure of successively assigning to certain variables the value 1, in such a way that after trying a (small) part of all the 2^n possible combinations, one obtains either an optimal solution or evidence of the fact that no feasible solution exists. The only operations required in the computation are additions and subtractions, and hence the round-off errors will not be there. For this reason the method is sometimes referred to as additive algorithm.

Standard Form of the Problem:

To describe the algorithm, consider the following form of the LP problem with zero-one variables:

$$\text{Find } \mathbf{X} = \begin{Bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{Bmatrix} \text{ such that } f(\mathbf{X}) = \mathbf{C}^T\mathbf{X} \to \text{ minimum}$$

subject to

$$\begin{aligned} \mathbf{AX} + \mathbf{Y} &= \mathbf{B} \\ x_i &= 0 \text{ or } 1 \\ \mathbf{Y} &\geq \mathbf{0} \end{aligned}$$

where

$$\mathbf{C} = \begin{Bmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{Bmatrix} \geq \mathbf{0}, \quad \mathbf{Y} = \begin{Bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{Bmatrix}, \quad \mathbf{B} = \begin{Bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{Bmatrix}$$

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & & & \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}$$

where $\mathbf{Y}$ is the vector of slack variables and c_i and a_{ij} need not be integers.

Initial Solution:

- An initial solution for the problem stated in Eqs. (10.28) can be taken as

$$\begin{aligned} f_0 &= 0 \\ x_i &= 0, \quad i = 1, 2, \ldots, n \\ \mathbf{Y}^{(0)} &= \mathbf{B} \end{aligned}$$

- If $\mathbf{B} \geq \mathbf{0}$, this solution will be feasible and optimal since $\mathbf{C} \geq \mathbf{0}$. In this case there is nothing more to be done as the starting solution itself happens to be optimal.
- On the other hand, if some of the components b_j are negative, the solution given by the above equation will be optimal (since $\mathbf{C} \geq \mathbf{0}$) but infeasible. Thus the method starts with an optimal (actually better than optimal) and infeasible solution. The algorithm forces this solution toward feasibility while keeping it optimal all the time.
- This is the reason why Balas called his method the pseudo dual simplex method. The word pseudo has been used since the method is similar to the dual simplex method only as far as the starting solution is concerned and the subsequent procedure has no similarity at all with the dual simplex method.

4.5 Solution of binary programming problems using MATLAB

The MATLAB function *bintprog* can be used to solve a binary (or zero–one) programming problem. The following example illustrates the procedure.

■ **Example 4.5.1** Example 10.7 Find the solution of the following binary programming problem using the MATLAB function bintprog:

$$\text{Minimize } f(\mathbf{X}) = -5x_1 - 5x_2 - 8x_3 + 4x_4 + 4x_5$$

subject to

$$3x_1 - 6x_2 + 7x_3 - 9x_4 - 9x_5 \leq -10, \quad x_1 + 2x_2 - x_4 - 3x_5 \leq 0$$

$$x_i \text{ binary } ; i = 1,2,3,4,5$$

■

■ **Solution 4.5.1** **Step 1:** State the problem in the form required by the program bintprog:

$$\text{Minimize } f(x) = f^{\mathrm{T}} x \text{ subject to } Ax \leq b \text{ and } A_{eq}x = b_{eq}$$

Here

$$f^{\mathrm{T}} = \left\{ \begin{array}{ccccc} -5 & -5 & -8 & 2 & 4 \end{array} \right\}, x = \left\{ \begin{array}{ccccc} x_1 & x_2 & x_3 & x_4 & x_5 \end{array} \right\}^{\mathrm{T}}$$

$$A = \begin{bmatrix} 3 & -6 & 7 & -9 & -9 \\ 1 & 2 & 0 & -1 & -3 \end{bmatrix}, b = \left\{ \begin{array}{c} -10 \\ 0 \end{array} \right\}$$

Step 2: The input is directly typed on the MATLAB command window and the program *bintprog* is called as indicated below:

```
% using bintprog MATLAB function
clear; clc;
f = [-5 -5 -8 2 4]';
A = [3 -6 7 -9 -9; 1 2 0 -1 -3];
b = [-10 0]';
x = bintprog (f, A, b,[])
```

Step 3: The output of the program is shown below:

```
Optimization terminated.
x =
  1
  1
  1
  1
  1
```

■

4.6 Exercises

Exercise 4.6.1 Solve the following programming problem using a graphical procedure

$$\text{Minimize } f = 4x_1 + 5x_2$$

subject to

$$3x_1 + x_2 \geq 2$$

$$x_1 + 4x_2 \geq 5$$

$$3x_1 + 2x_2 \geq 7$$

$$x_1, x_2 \geq 0, \text{ integers}$$

■

Exercise 4.6.2 Solve the following programming problem using a graphical procedure

$$\text{Maximize } f = 4x_1 + 8x_2$$

subject to

$$4x_1 + 5x_2 \leq 40$$

$$x_1 + 2x_2 \leq 12$$

$$x_1, x_2 \geq 0, \text{ integers}$$

■

Exercise 4.6.3 Solve the following programming problem using a graphical procedure

$$\text{Maximize } f = 4x_1 + 3x_2$$

subject to

$$3x_1 + 2x_2 \leq 18$$

$$x_1, x_2 \geq 0, \text{ integers}$$

■

Exercise 4.6.4 Solve the following programming problem using a graphical procedure

$$\text{Maximize } f = 3x_1 - x_2$$

subject to

$$3x_1 - 2x_2 \leq 3$$
$$-5x_1 - 4x_2 \leq -10 x_1, x_2 \geq 0, \text{ integers}$$

Exercise 4.6.5 Solve the following zero-one programming problem using MATLAB

$$\text{Maximize } f = -10x_1 - 5x_2 - 3x_3$$

subject to

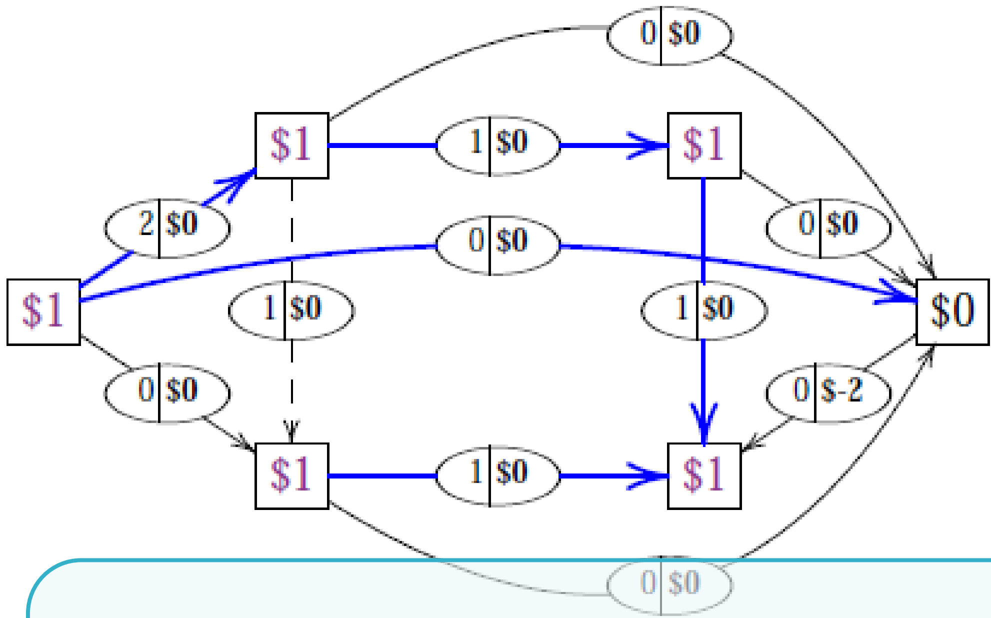
$$x_1 + 2x_2 + x_3 \geq 4$$
$$2x_1 + x_2 + x_3 \leq 6$$
$$x_i = 0 \text{ or } 1, \quad i = 1, 2, 3$$

Exercise 4.6.6 Solve the following zero-one programming problem using MATLAB

$$\text{Minimize } f = -5x_1 + 7x_2 + 10x_3 - 3x_4 + x_5$$

subject to

$$x_1 + 3x_2 - 5x_3 + x_4 + 4x_5 \leq 0$$
$$2x_1 + 6x_2 - 3x_3 + 2x_4 + 2x_5 \geq 4$$
$$x_2 - 2x_3 - x_4 + x_5 \leq -2$$
$$x_i = 0 \text{ or } 1, \quad i = 1, 2, \ldots, 5$$



5. Network problems

5.1 Introduction

Network routing problems commonly arise in communication and transportation systems. Delays that occur at the nodes (e.g., railroad classification yards or telephone switchboards) may be a function of the loads placed on them and their capacities. Breakdowns may occur in either links or nodes. Much studied is the "traveling salesman problem," which consists of starting a route from a designated node that goes through each node (e.g., city) only once and returns to the origin in the least time, cost, or distance. This problem arises in selecting an order for processing a set of production jobs when the cost of setting up each job depends on which job has preceded it. In this case the jobs can be thought of as nodes, each of which is connected to all of the others, with setup costs as the analogue of distances between them. The order that yields the least total setup cost is therefore equivalent to a solution to the traveling salesman problem.

A network may be defined by a set of points, or "nodes," that are connected by lines, or "links." A way of going from one node (the "origin") to another (the "destination") is called a "route" or "path." Links, which may be one-way or two-way, are usually characterized by the time, cost, or distance required to traverse them. The time or cost of traveling in different directions on the same link may differ.

Thus, G =**graph or network** consists of

– a set V of vertices (nodes, points) and

– a set E of edges (arcs, lines) which are connections between vertices.

write $G = (V, E)$; write $V(G)$ for vertices of G, and $E(G)$ for edges of G.

(vertices are usually denoted u or v with subscripts; edges we usually denote e) edges may have **direction**: an edge e between u and v may go from u to v, we write $e = (u, v)$, or from v to u, we write $e = (v, u)$

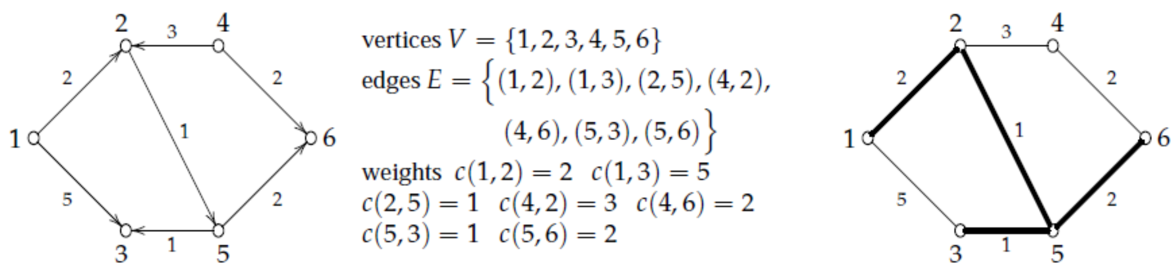


Figure 5.1-1: Network components

(if an edge e does not have a direction, we treat it the same way as having both directions) if all edges do not have a direction (are undirected), we say that the network is **undirected** edges may have weight: a weight of edge $e = (u, v)$ is a real number denoted $c(e)$ or $c(u, v), c_e, c_{uv}$ a sequence of nodes and edges $v_1, e_1, v_2, e_2, \ldots v_{k-1}, e_k, v_k$ is

- a path (directed path) if each e_i goes from v_i to v_{i+1}
- a chain (undirected path) if each e_i connects v_i and v_{i+1} (in some direction)

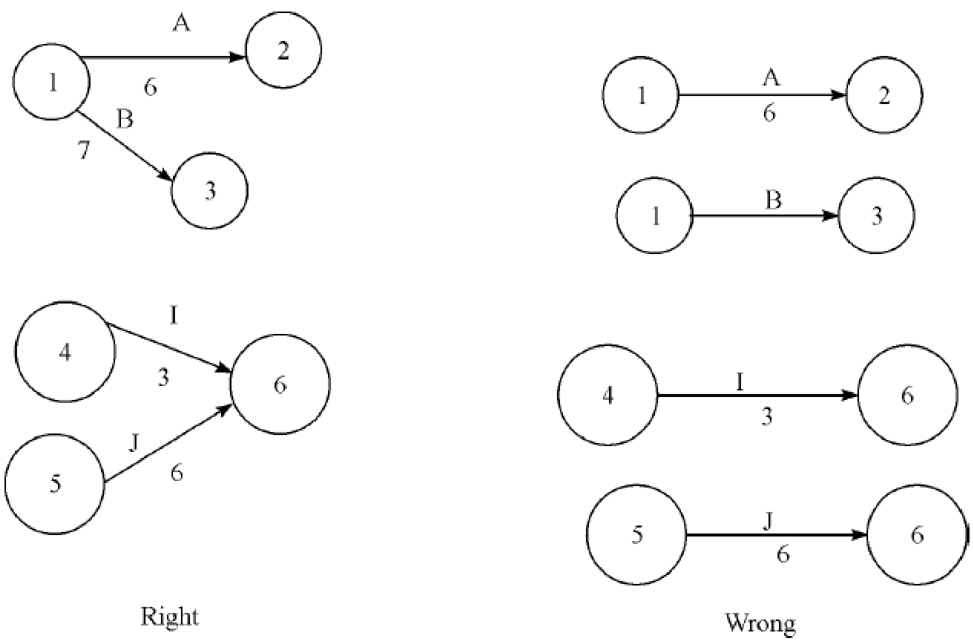
(often we write: $e_1, e_2, \ldots, e_k$ is a path (we omit vertices) or write: $v_1, v_2, \ldots, v_k$ is a path (we omit edges))

a network is connected if for every two nodes there is a path connecting them; otherwise it is disconnected a cycle (loop, circuit) is a path starting and ending in the same node, never repeating any node or edge a forest (acyclic graph) is an undirected graph that contains no cycles a tree is a connected forest

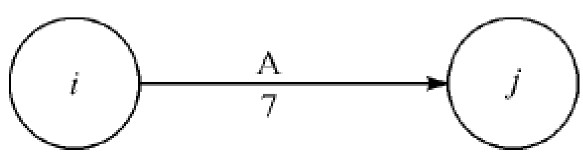
Claim: A tree with n nodes contains exactly $n - 1$ edges. Adding any edge to a tree creates a cycle. Removing any edge from a tree creates a disconnected forest.

Some of the points to be remembered while drawing the network are

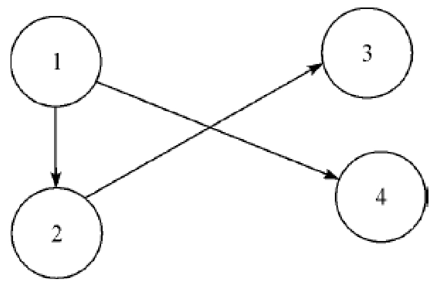
- There must be only one beginning and one end for the network as in this figure



- Event number should be written inside the circle or node (or triangle/square/rectangle etc). Activity name should be capital alphabetical letters and would be written above the arrow. The time required for the activity should be written below the arrow as

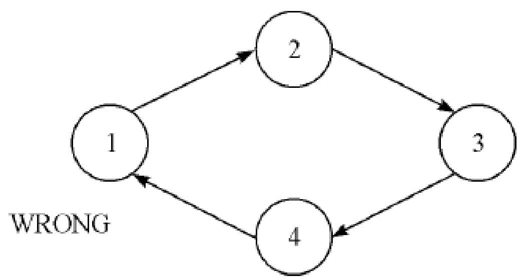


- While writing network, see that activities should not cross each other. And arcs or loops as in the following figure should not join Activities (Crossing of activities not allowed).

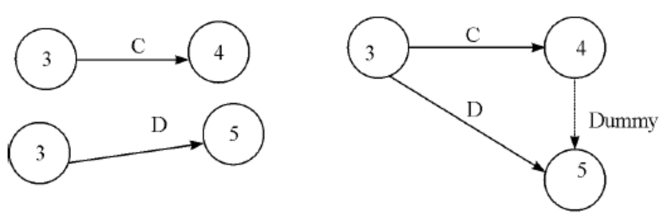


WRONG

- While writing network, looping should be avoided. This is to say that the network arrows should move in one direction, i.e. starting from the beginning should move towards the end, as in the following figure (Looping is not allowed.)

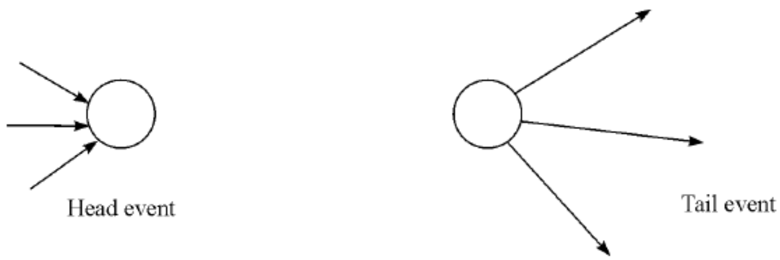


- When two activities start at the same event and end at the same event, they should be shown by means of a dummy activity as in the following figure. **Dummy activity** is an activity, which simply shows the logical relationship and does not consume any resource. It should be represented by a dotted line as shown. In the figure, activities C and D start at the event 3 and end at event 4. C and D are shown in full lines, whereas the dummy activity is shown in dotted line.

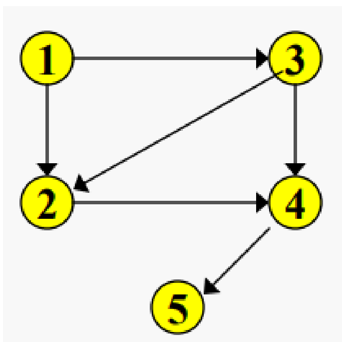


Wrong Right

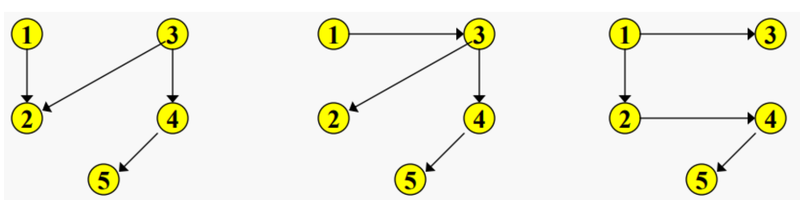
- When the event is written at the tail end of an arrow, it is known as **tail event**. If event is written on the head side of the arrow it is known as **head event**. A tail event may have any number of arrows (activities) emerging from it. This is to say that an event may be a tail event to any number of activities. Similarly, a head event may be a head event for any number of activities. This is to say that many activities may conclude at one event.



- A network is **connected** if every node can be reached from every other node by a path



- A **spanning tree** is a connected subset of a network including all nodes, but containing no cycles.



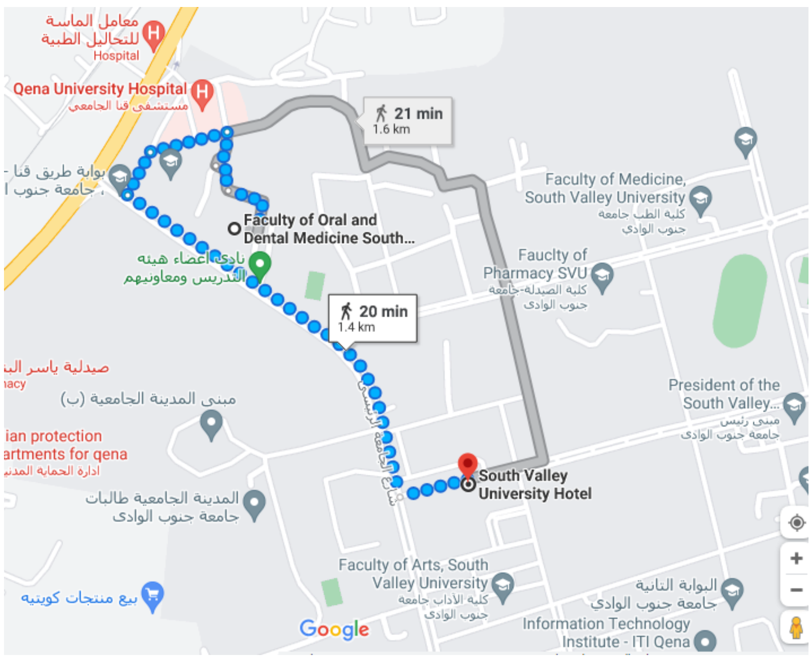
- An out-tree is a spanning tree in which every node has exactly one incoming arc except for the root.

5.2 Shortest Path Problem

Where does it arise in practice?

Common applications

- shortest paths in a vehicle (Navigator)
- shortest paths in internet routing
- shortest paths around any university camps for instance; SVU.

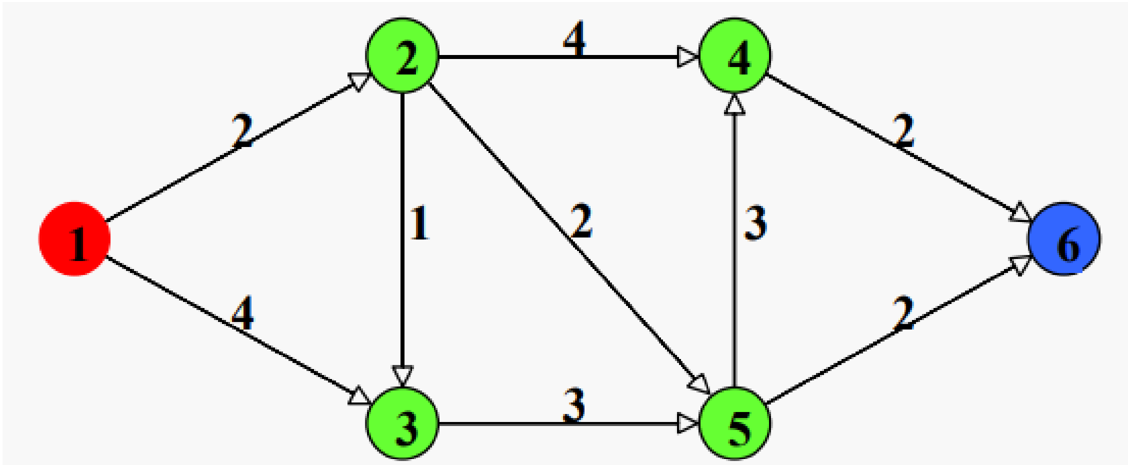


How will we solve the shortest path problem?

– Dijkstra's algorithm

What is the shortest path from a source node (often denoted as s) to a sink node, (often denoted as t)?

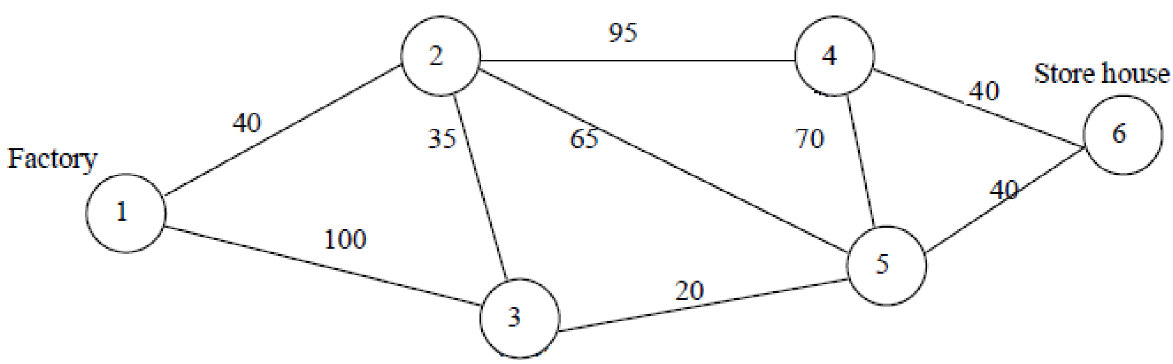
What is the shortest path from node 1 to node 6?



- Given a network $G = (V, E)$ with two distinguished vertices $s, t \in V$, find a shortest path from s to t?

Example 5.2.1 *A real life situation involving a shortest route problem.*

A leather manufacturing company has to transport the finished goods from the factory to the store house. The path from the factory to the store house is through certain intermediate stations as indicated in the following diagram. The company executive wants to identify the path with the shortest distance so as to minimize the transportation cost. The problem is to achieve this objective. ■



The shortest route technique can be used to minimize the total distance from a node designated as the starting node or origin to another node designated as the final node.

■ Solution 5.2.1

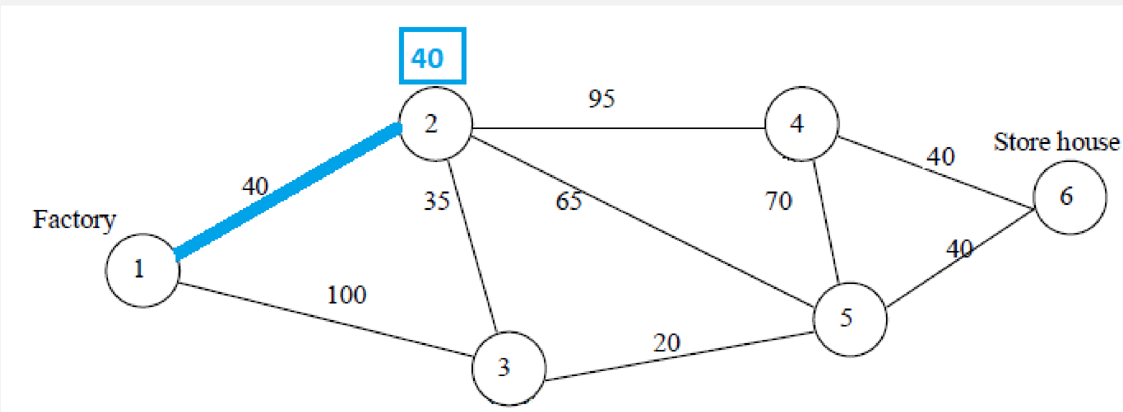
Step 1

- Looking at the diagram, we see that node 1 is the origin and the nodes 2 and 3 are neighbours to the origin.

- Among the two nodes, we see that node 2 is at a distance of 40 units from node 1 whereas node 3 is at a distance of 100 units from node 1.

- The minimum of $\{40, 100\}$ is 40. Thus, the node nearest to the origin is node 2, with a distance of 40 units. So, out of the two nodes 2and 3, we select node 2.

-We form a set of nodes $\{1, 2\}$ and construct a path connecting the node 2 with node 1 by a thick line and mark the distance of 40 in a box by the side of node 2. This first iteration is shown in the following diagram.



Step 2: Now we search for the next node nearest to the set of nodes $\{1, 2\}$. For this purpose, consider those nodes which are neighbours of either node 1 or node 2. The nodes $3, 4$ and 5 fulfill this condition. We calculate the following distances.

The distance between nodes 1 and $3 = 100$.

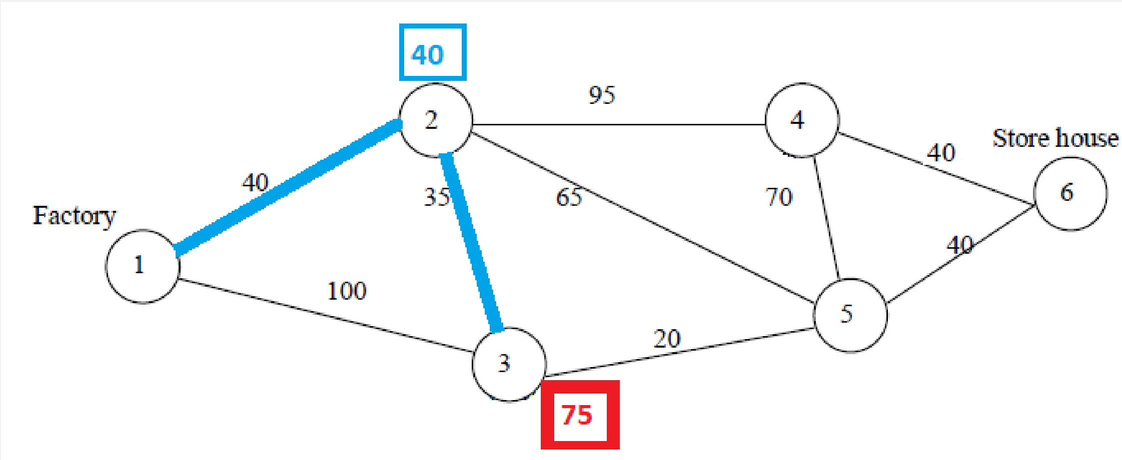
The distance between nodes 2 and $3 = 35$.

The distance between nodes 2 and $4 = 95$.

The distance between nodes 2 and $5 = 65$.

Minimum of $\{100, 35, 95, 65\} = 35$.

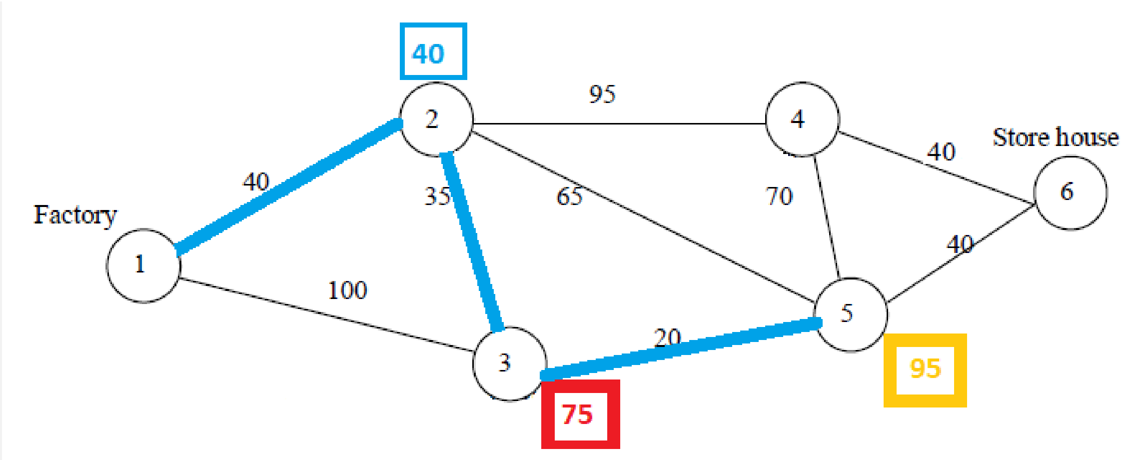
Therefore, node 3 is the nearest one to the set $\{1,2\}$. In view of this observation, the set of nodes is enlarged from $\{1,2\}$ to $\{1,2,3\}$. For the set $\{1,2,3\}$, there are two possible paths, viz. Path $1 \to 2 \to 3$ and Path $1 \to 3 \to 2$. The Path $1 \to 2 \to 3$ has a distance of $40+35 = 75$ units while the Path $1 \to 3 \to 2$ has a distance of $100+35 = 135$ units. Minimum of$\{75, 135\} = 75$. Hence we select the path $1 \to 2 \to 3$ and display this path by thick edges. The distance 75 is marked in a box by the side of node 3. We obtain the following diagram at the end of Iteration No. 2



Repeating the process: We repeat the process. The next node nearest to the set $\{1,2,3\}$ is either node 4 or node5.

Node 4 is at a distance of 95 units from node 2 while node 2 is at a distance of 40 units from node 1. Thus, node 4 is at a distance of$95+40 = 135$ units from the origin.

As regards node 5, there are two paths viz. $2 \to 5$ and $3 \to 5$, providing a link to the origin. We already know the shortest routes from nodes 2 and 3 to the origin. The minimum distances have been indicated in boxes near these nodes. The path $3 \to 5$ involves the shortest distance. Thus, the distance between nodes 1 and 5 is 95 units (20 units between nodes 5 and $3+75$ units between node 3 and the origin). Therefore, we select node 5 and enlarge the set from $\{1,2,3\}$ to $\{1,2,3,5\}$. The distance 95 is marked in a box by the side of node 5. The following diagram is obtained at the end of Iteration No. 3.



Now 2 nodes remain, viz., nodes 4 and 6.

- Among them, node 4 is at a distance of 135 units from the origin (95 units from node 4 to node $2 + 40$ units from node 2 to the origin).

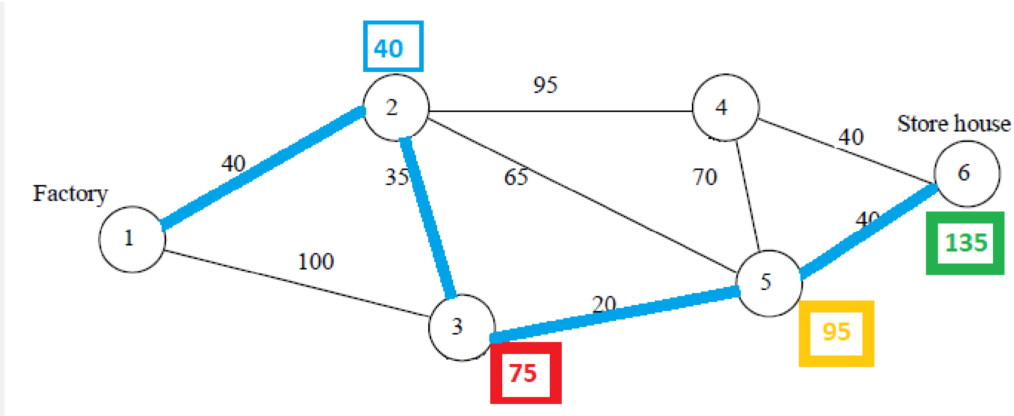
- Node6 is at a distance of 135 units from the origin ($40 + 95$ units). Therefore, nodes 4 and 6 are at equal distances from the origin.

- If we choose node 4, then travelling from node 4 to node 6 will involve an additional distance of 40 units. However, node 6 is the ending node.

- Therefore, we select node 6 instead of node 4. Thus the set is enlarged from $\{1,2,3,5\}$ to $\{1,2,3,5,6\}$.

-The distance 135 is marked in a box by the side of node 6. Since we have got a path beginning from the start node and terminating with the stop node, we see that the solution to the given problem has been obtained.

We have the following diagram at the end of Iteration No. 4.



Minimum distance: Referring to the above diagram, we see that the shortest route is provided by the path $1 \to 2 \to 3 \to 5 \to 6$ with a minimum distance of135 units ■

5.3 Minimum spanning tree problem

Tree: A minimally connected network is called a tree. If there are n nodes in a network, it will be a tree if the number of edges $= n-1$.

5.3.1 Minimum spanning tree algorithm

Problem : Given a connected network with weights assigned to the edges, it is required to find out a tree whose nodes are the same as those of the network.

The weight assigned to an edge may be regarded as the distance between the two nodes with which the edge is incident. **Algorithm:**

The problem can be solved with the help of the following algorithm.

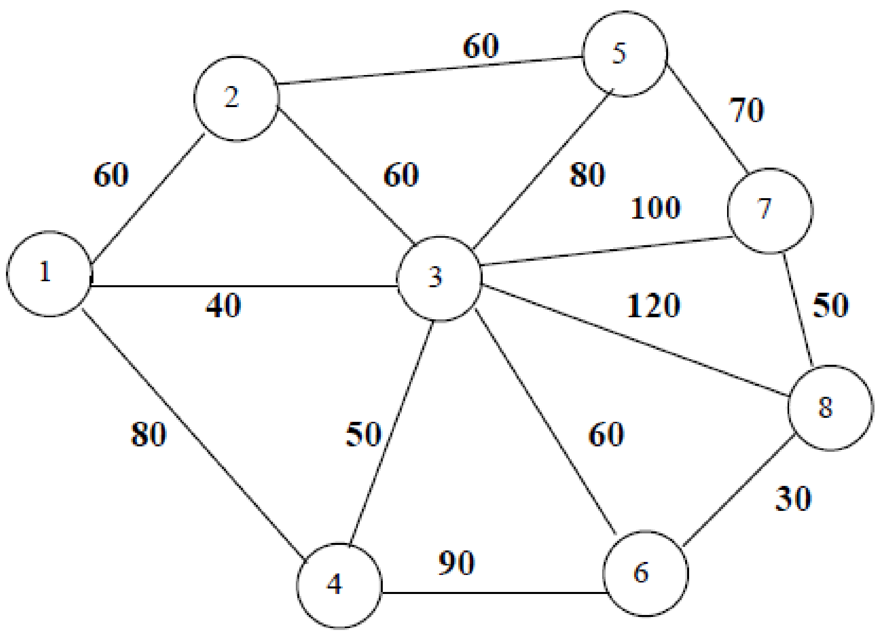
The procedure consists of selection of a node at each step.

- **Step 1:** First select any node in the network. This can be done arbitrarily. We will start with this node.
- **Step 2:** Connect the selected node to the nearest node.
- **Step 3:** Consider the nodes that are now connected. Consider the remaining nodes. If there is no node remaining, then stop. On the other hand, if some nodes remain, among them find out which one is nearest to the nodes that are already connected. Select this node and go to Step 2.

Thus the method involves the repeated application of Steps 2 and 3. Since the number of nodes in the given network is finite, the process will end after a finite number of steps. The algorithm will terminate with step 3.

R While applying the above algorithm, if some nodes remain in step 3 and if there is a tie in the nearest node, then the tie can be broken arbitrarily.
As a consequence of tie, we may end up with more than one optimal solution.

Example 5.3.1 Determine the minimum spanning tree for the following network.

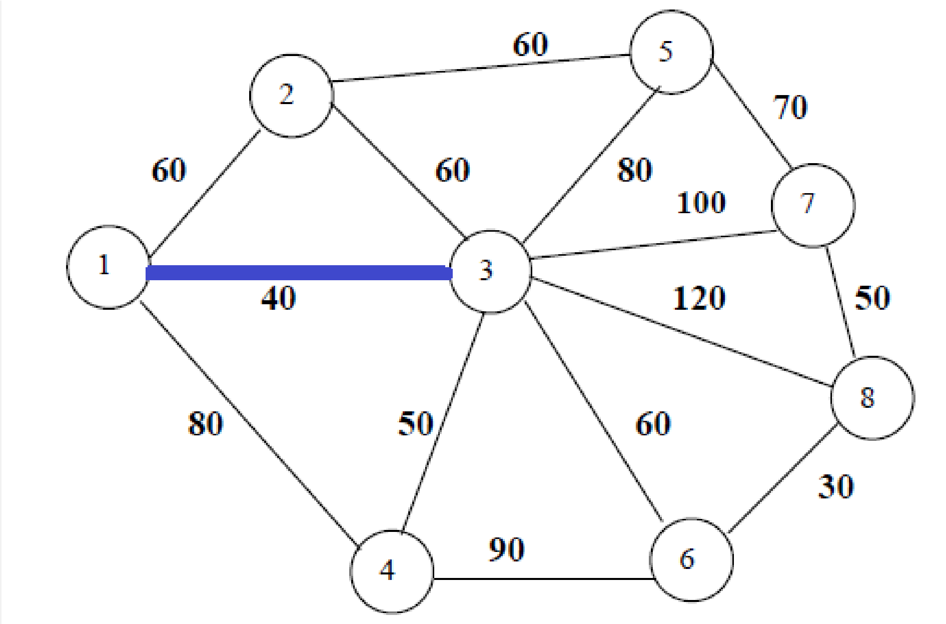


■

Solution 5.3.1 **Step 1:** First select node 1. (This is done arbitrarily)

Step 2: We have to connect node 1 to the nearest node. Nodes $2,3$ and 4 are adjacent to node 1. They are at distances of $60,40$ and 80 units from node 1. Minimum of $\{60,40,80\} = 40$.

Hence the shortest distance is40. This corresponds to node 3. So we connect node 1to node 3 by a thick line. This is Iteration No. 1.

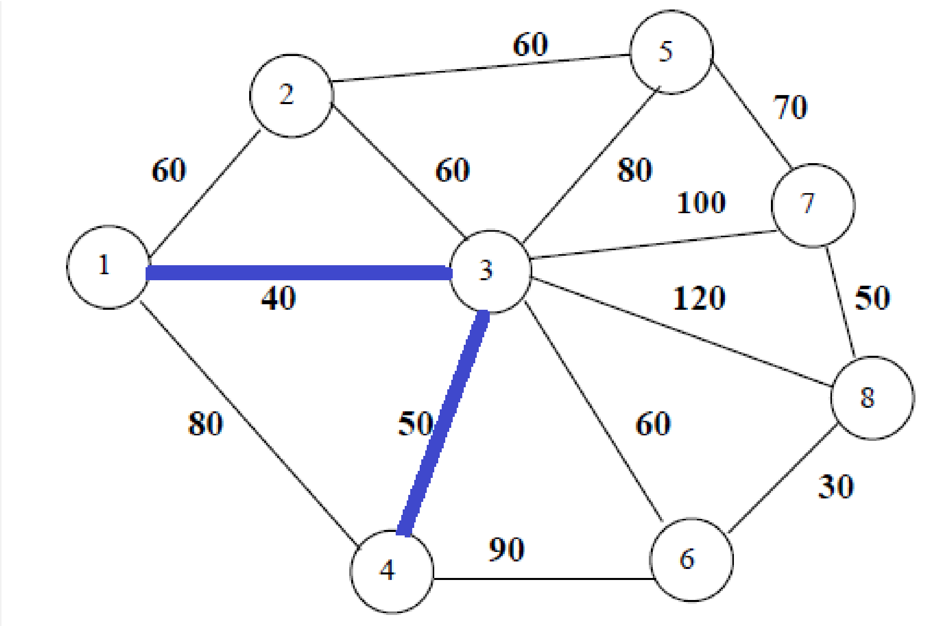


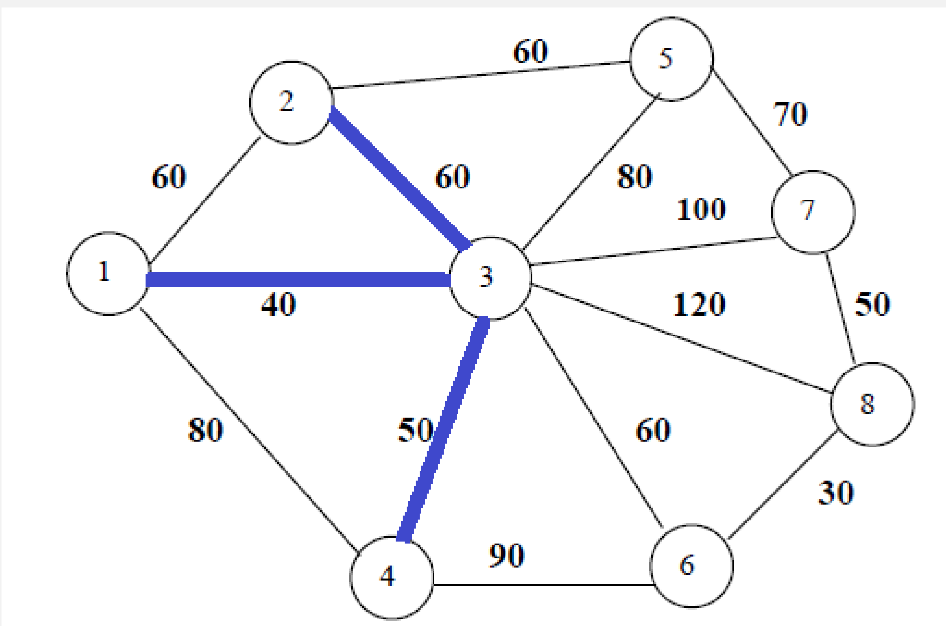
Step 3: Now the connected nodes are 1 and 3. The remaining nodes are $2, 4, 5, 6, 7$ and 8. Among them, nodes 2 and 4 are connected to node 1. They are at distances of 60 and 80 from node 1. Minimum of $\{60, 80\} = 60$. So the shortest distance is 60. Next, among the nodes $2, 4, 5, 6, 7$ and 8, find out which nodes are connected to node 3. We find that all of them are connected to node 3. They are at distances of $60, 50, 80, 60, 100$ and 120 from node 3.

Minimum of $\{60, 50, 80, 60, 100, 120\} = 50$. Hence the shortest distance is 50.

Among these nodes, it is seen that node 4 is nearest to node 3.

Now we go to Step 2. We connect node 3 to node 4 by a thick line. This is Iteration No.2.



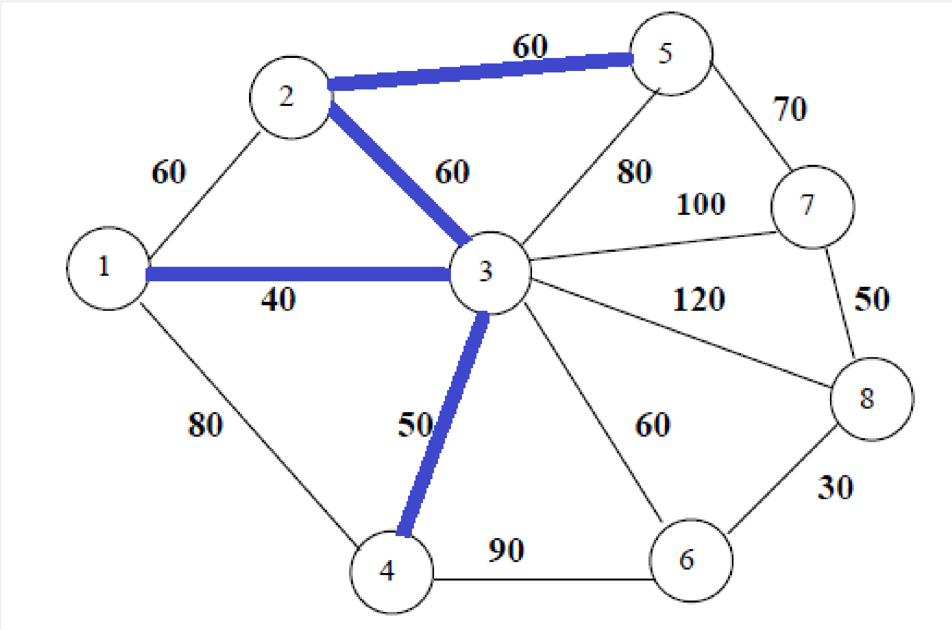


Next go to step 3. Now the connected nodes are 1, 3 and 4. The remaining nodes are 2, 5, 6, 7 and 8.

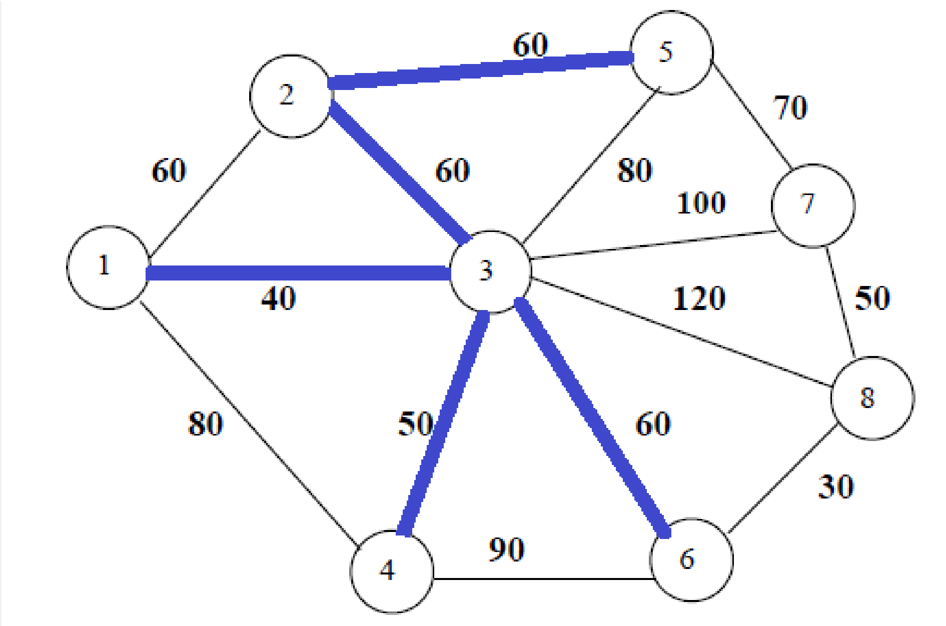
Node 2 is at a distance of 60 from node 1. Nodes 5, 6, 7 and 8 are not adjacent to node 1.

All of the nodes 2, 5, 6, 7 and 8 are adjacent to node 3. Among them, nodes 2 and 6 are nearer to node 3, with equal distance of 60. Node 6 is adjacent to node 4, at a distance of 90. Now there is a tie between nodes 2 and 6. The tie can be broken arbitrarily. So we select node 2. Connect node 3 to Node 2 by a thick line. This is Iteration No. 3.

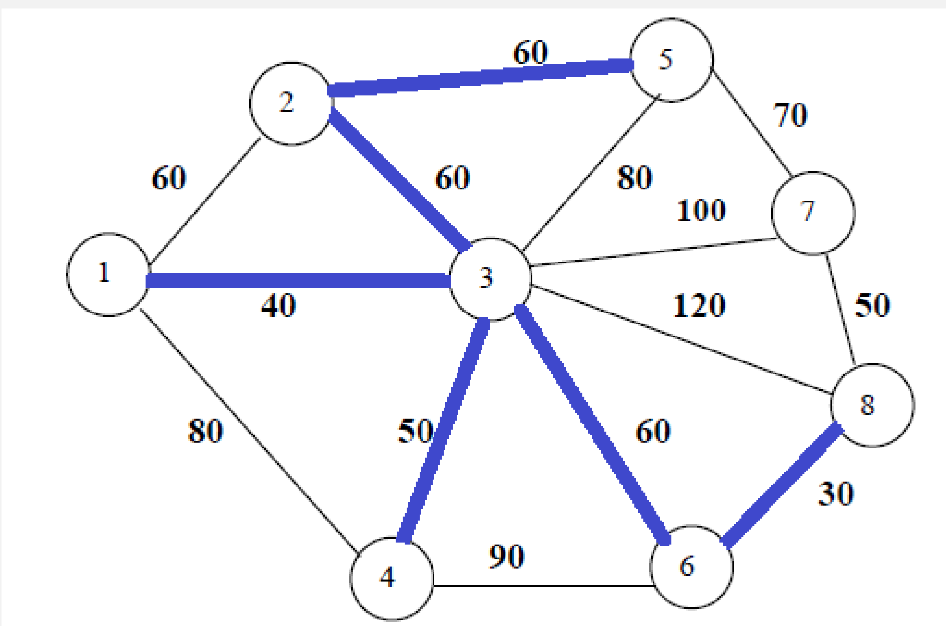
• We continue the above process. Now nodes $1,2,3$ and 4 are connected. The remaining nodes are $5,6,7$ and 8. None of them is adjacent to node 1. Node 5 is adjacent to node 2 at a distance of 60. Node 6 is at a distance of 60 from node 3. Node 6 is at a distance of 90 from node 4. There is a tie between nodes 5 and 6. We select node 5. Connect node 2 to node 5 by a thick line. This is Iteration No. 4.



Now nodes $1,2,3,4$ and 5 are connected. The remaining nodes are $6,7$ and 8.
Among them,node 6 is at the shortest distance of 60 from node 3. So, connect node 3 to node 6 by a thick line. This is Iteration No. 5.

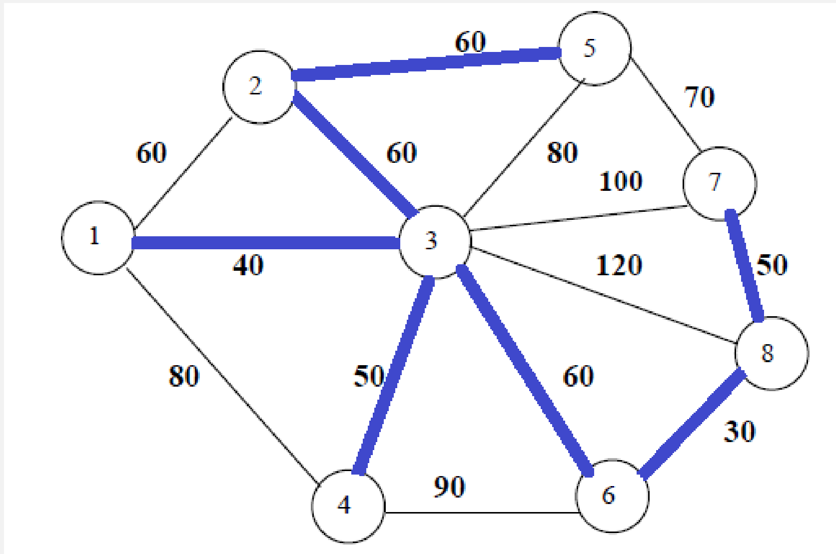


Now nodes1, 2, 3, 4, 5 and 6 are connected. The remaining nodes are 7 and 8. Among them, node 8 is at the shortest distance of 30 from node 6. Consequently we connect node 6 to node 8 by a thick line. This is Iteration No. 6.

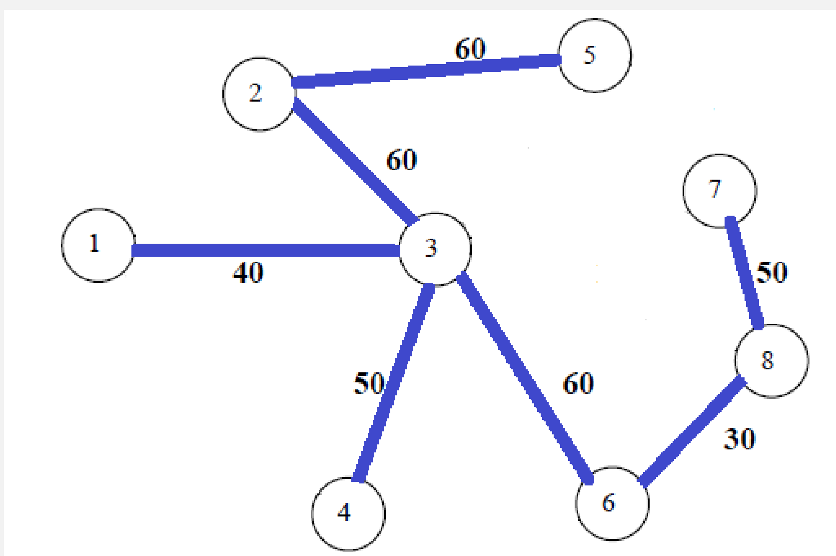


Now nodes 1, 2, 3, 4, 5, 6 and 8 are connected. The remaining node is7.

It is at the shortest distance of 50 from node 8. So, connect node 8 to node 7 by a thick line. This is Iteration No.7.



Now all the nodes 1,2,3,4,5,6,7 and 8 are connected by seven thick lines. Since no node is remaining, we have reached the stopping condition. Thus, we obtain the following minimum spanning tree for the given network.



■

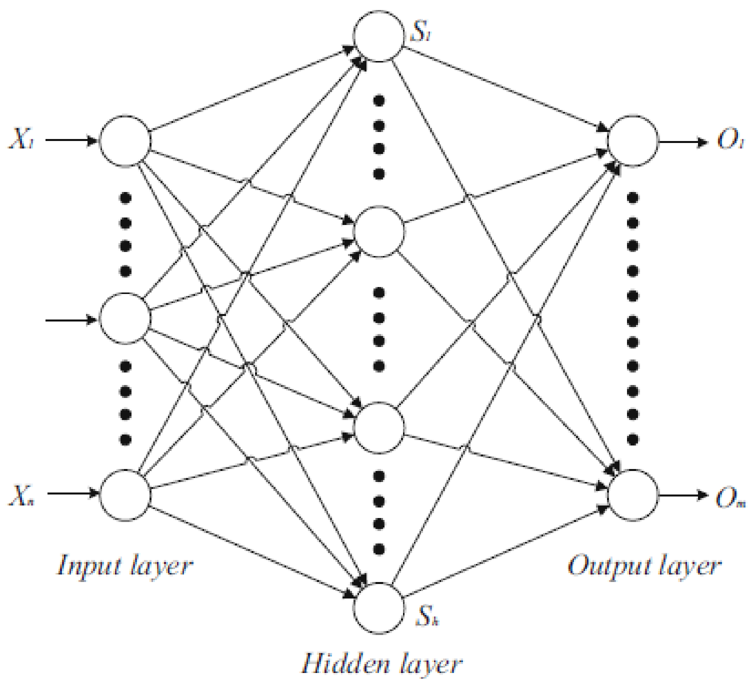
5.4 Neural-network-based optimization

- The immense computational power of nervous system to solve perceptional problems in the presence of massive amount of sensory data has been associated with its parallel processing capability.
- The neural computing strategies have been adopted to solve optimization problems in recent years
- A neural network is a massively parallel network of interconnected simple processors (neurons) in which each neuron accepts a set of inputs from other neurons and computes an output that is propagated to the output nodes. Thus a neural network can be described in terms of the individual neurons, the network connectivity, the weights associated with the interconnections between neurons, and the activation function of each neuron. The network maps an input vector from one space to another. The mapping is not specified but is learned.

5.4.1 Feedforward Neural Networks

- Feedforward Neural Networks (FNNs) have consistently ranked among the most widely adopted neural network models, showcasing their versatility across various applications. The procedure of fine-tuning the control parameters of a neural network is referred to as training and can be viewed as an optimization challenge. Within this optimization context, the primary objective revolves around discovering optimal parameter values to minimize errors in the neural network or enhance its accuracy.
- Traditionally, FNNs have been trained using a method known as Back Propagation (BP). This training algorithm relies on gradients and employs a gradient descent approach to converge towards the best solution, starting from an initial random point. However, the BP algorithm encounters a challenge in the form of local optima stagnation due to its reliance on gradients.
- Feedforward neural networks (FNNs) are characterized by a unidirectional connection between their neurons. These neural networks consist of neurons organized into distinct parallel layers. The initial layer is consistently referred to as the input layer,

while the ultimate layer is termed the output layer. Any intermediary layers positioned between the input and output layers are known as hidden layers. An FNN featuring a single hidden layer is simply referred to as an FNN, as depicted in the following figure



- Upon supplying the inputs, along with their corresponding weights and biases, the output of FNNs is determined through the subsequent steps:
  (a) The weighted sums of inputs are first calculated by

  $$s_j = \sum_{i=1}^{n} \left(W_{ij}X_i\right) - \theta_j, \quad j = 1,2,\ldots,h$$

  where n is the number of the input nodes, W_{ij} shows the connection weight from the i^{th} node in the input layer to the jth node in the hidden layer, θ_j is the bias (threshold) of the j th hidden node, and X_i indicates the i^{th} input.

  (b) The output of each hidden node is calculated as follows:

  $$S_j = \text{sigmoid}(s)j\big) = \frac{1}{1+e^{-s_j}}, \quad j = 1,2,\ldots m$$

(c) The final outputs are defined based on the calculated outputs of the hidden nodes as follows:

$$o_k = \sum_{j=1}^{h} \left(w_{jk} S_j \right) - \theta'_k, \quad k = 1, 2, \ldots, m$$

$$O_k = \text{sigmoid}\left(o_k\right) = \frac{1}{1 + e^{-o_k}}, \quad k = 1, 2, \ldots, m$$

where w_{jk} is the connection weight from the j^{th} hidden node to the k^{th} output node, and θ'_k is the bias (threshold) of the k^{th} output node.

As evident from these equations, it is apparent that the weights and biases play a crucial role in determining the ultimate output of FNNs based on provided inputs. The task of identifying suitable values for these weights and biases, with the aim of establishing a desirable relationship between inputs and outputs, precisely characterizes the process of training FNNs.

- Several neural network architectures, such as the Hopfield and Kohonen networks, have been proposed to reflect the basic characteristics of a single neuron.
  These architectures differ one from the other in terms of the number of neurons in the network, the nature of the threshold functions, the connectivities of the various neurons, and the learning procedures.
- Various neural network architectures, including models like Hopfield and Kohonen networks, have been devised to capture the fundamental attributes of an individual neuron.

- These architectures exhibit distinctions from one another in terms of factors such as the network's neuron count, the characteristics of threshold functions, the interconnections among neurons, and the methodologies employed for learning

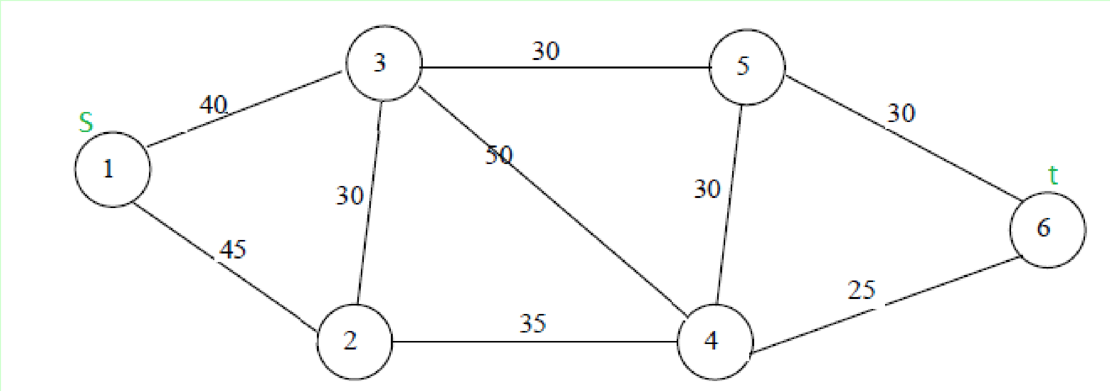
Further reading can be found in the bibliography and references therein.

5.5 Exercises

Exercise 5.5.1 Explain the shortest path problem. ■

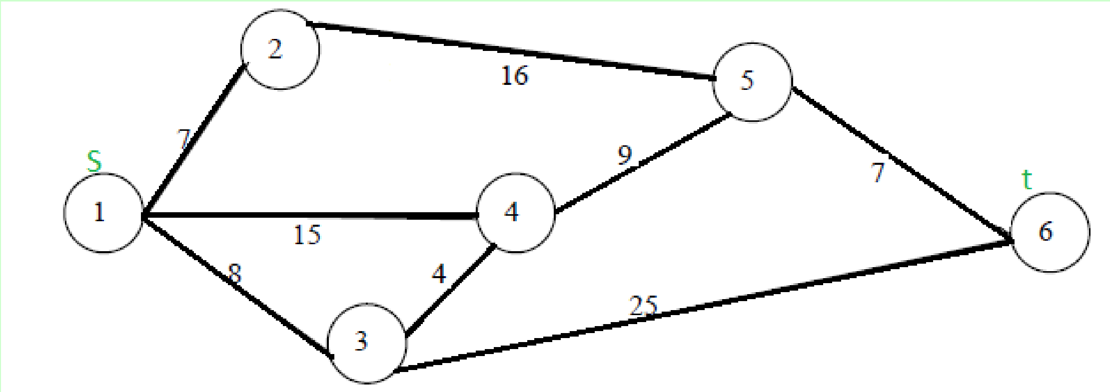
Exercise 5.5.2 Explain the algorithm for a shortest path problem ■

Exercise 5.5.3 Find the shortest path of the following network:



■

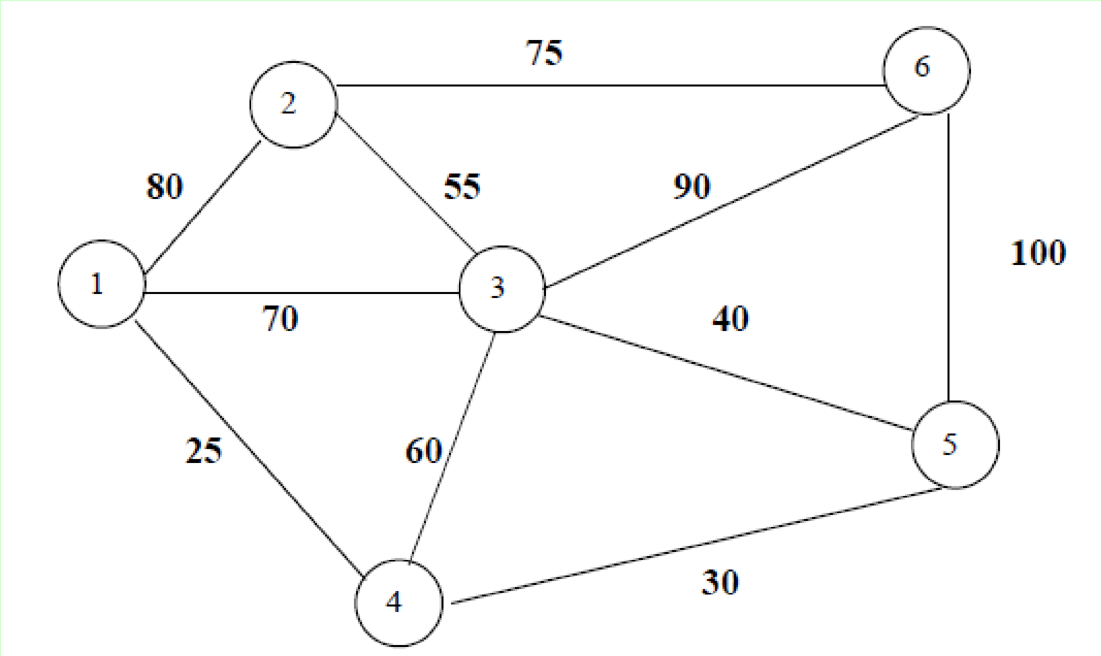
Exercise 5.5.4 Find the shortest path of the following network:



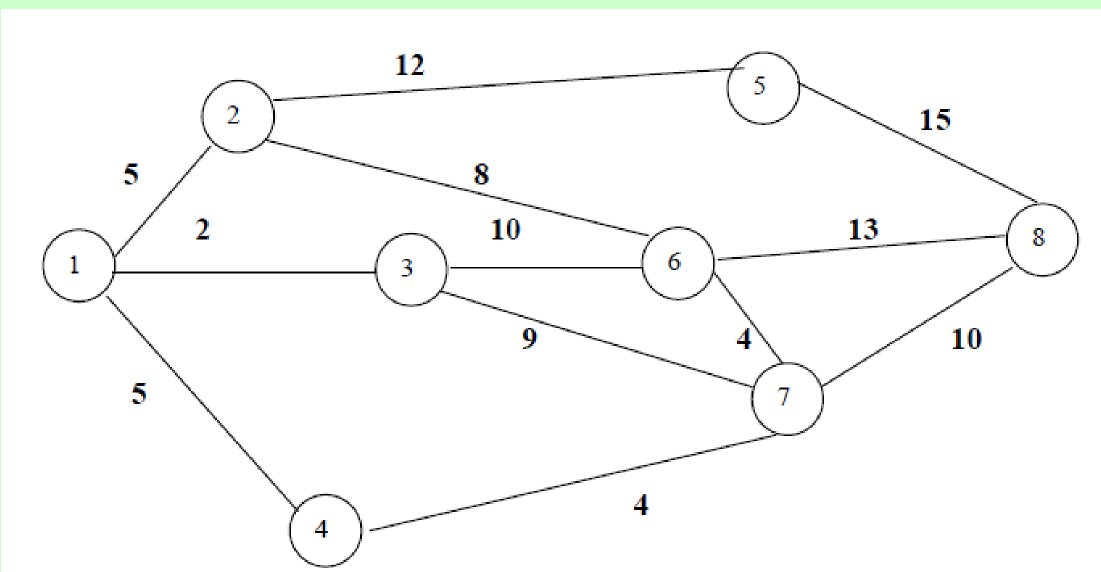
■

Exercise 5.5.5 Explain the minimum spanning tree algorithm. ■

Exercise 5.5.6 From the following network, find the minimum spanning tree.



Exercise 5.5.7 Find the minimum spanning tree of the following network:



Bibliography

[1] Maher GM Abdolrasol, SM Suhail Hussain, Taha Selim Ustun, Mahidur R Sarker, Mahammad A Hannan, Ramizi Mohamed, Jamal Abd Ali, Saad Mekhilef, and Abdalrhman Milad. Artificial neural networks based optimization techniques: A review. *Electronics*, 10(21):2689, 2021.

[2] Frederick S Hillier. *Introduction to operations research*. McGrawHill, 2001.

[3] Olvi L Mangasarian. *Nonlinear programming*. SIAM, 1994.

[4] Achille Messac. *Optimization in practice with MATLAB®: for engineering students and professionals*. Cambridge University Press, 2015.

[5] Seyedali Mirjalili. Evolutionary algorithms and neural networks. In *Studies in computational intelligence*, volume 780. Springer, 2019.

[6] P Rama Murthy. *Operations research (linear programming)*. bohem press, 2005.

[7] Pablo Pedregal. *Introduction to optimization*, volume 46. Springer, 2004.

[8] Singiresu S Rao. *Engineering optimization: theory and practice*. John Wiley & Sons, 2019.

[9] Juraj Stacho. Introduction to operations research: Deterministic models, 2014.

[10] Wenyu Sun and Ya-Xiang Yuan. *Optimization theory and methods: nonlinear programming*, volume 1. Springer Science & Business Media, 2006.

[11] Hamdy A Taha. *Operations research: an introduction*. Pearson Education India, 2013.

[12] Steven Vajda. An introduction to linear programming and the theory of games. *(No Title)*, 1960.

[13] Gary R Waissi. Network flows: Theory, algorithms, and applications, 1994.