Operations Research

History of Operations Research

The term Operation Research has its origin during the Second World War. The military management of England called a team of scientists to study the strategic and tactical problems which could raise in air and land defence of the country. As the resources were limited and those need to be fully but properly utilized. The team did not involve actually in military operations like fight or attending war but the team kept off the war but studying and suggesting various operations related to war.

What is Operations Research?

Several definitions have been given

- Operations research (abbreviated as OR hereafter) is a scientific method of providing executive departments with a quantitative basis for decisions regarding the operations under their control: Morse and Kimbal (1944)
- OR is an analytical method of problem-solving and decision-making that is useful in the management of organizations. In operations research, problems are broken down into basic components and then solved in defined steps by mathematical analysis.
- Operational Research (OR) is the use of advanced analytical techniques to improve decision making. It is sometimes known as Operations Research, Management Science or Industrial Engineering. People with skills in OR hold jobs in decision support, business analytics, marketing analysis and logistics planning – as well as jobs with OR in the title.
- As such a number of definitions can be found in literature, you can express the term OR with the spirit mentioned in the literature.

1.1 Introduction to Operations Research (OR)

The term Operations Research (to be termed OR hereafter) describes the discipline that is focused on the application of information technology for informed decision-making. In other words, OR represents the study of optimal resource allocation. The goal of OR is to provide rational bases for decision making by seeking to understand and structure complex situations, and to utilise this understanding to predict system behaviour and improve system performance. Much of the actual work is conducted by using analytical and numerical techniques to develop and manipulate mathematical models of organisational systems that are composed of people, machines, and procedures.

1.1.1 Brief History of Operations Research

The term operations research (O.R.) was coined during World War II, when the British military management called upon a group of scientists together to apply a scientific approach in the study of military operations to win the battle. The main Objectives was to allocate scarce resources in an effective manner to various military operations and to the activities within each operation. The effectiveness of operations research in military, inspired other government departments and industries.

1.1.2 Indian Context of Operations Research

India used the techniques of operations research in the year of 1949 at Hyderabad. In Hyderabad, a unit for operations research was set up names Regional Research Institute. Later on, in 1953, another operations research unit was established at Calcutta for national planning and survey names Indian Statistical Institute. Various other Indian companies are using operations research techniques for solving their problems of advertising, quality control, transportations, planning and sales promotions.

1.1.3 Definitions of Operations Research

OR has been defined in various ways, however given below are a few opinions about the definition of OR which have been changed along-with the development of the subject.

In 1946 Morse & Kimbel has defined O. R. as:

“OR is a scientific method of providing executive departments with a quantitative basis for decision regarding the operations under their control.”
In 1957, Churchmen Ackoff and Arnoff defined:

“OR is the application of scientific methods, techniques and tools to problems involving the operations of systems so as to provide those in control of the operations with optimum solutions to the problem.”

The operational research society of U.K. defines OR as:

“Operational Research is the application of the methods of science to complex problems arising in the direction and management of large systems of men, machines, materials and money in industry, business, government and defence. The distinctive approach is to develop a scientific model of the system, incorporating measurements of factors, such as chance and risk, with which to compare the outcome of alternative decisions, strategies and controls. The purpose is to help management determine its policies and actions scientifically.”

1.2 Phases of Operations Research

Formulate the problem: This is the most important process; it is generally lengthy and time consuming. The activities that constitute this step are visits, observations, research, etc. With the help of such activities, the O.R. scientist gets sufficient information and support to proceed and is better prepared to formulate the problem. This process starts with understanding of the organisational climate, its Objectivess and expectations. Further, the alternative courses of action are discovered in this step.

Models of Operations Research

A model is a representation of the reality. It is an idealised representation or abstraction of a real life system. A model is helpful in decision making as it provides a simplified description of complexities and uncertainties of a problem in logical structure.

Physical model:• It includes all forms of diagrams, graphs and charts. They are designed to deal with specific problems. They bring out significant factors and inter-relationship in pictorial firm so as to facilitate analysis.

Mathematical model:• It is known as symbolic models also. It employs a set of mathematical symbols to represent the decision variable of the system.

By nature of environment:• Deterministic model in which every thing is defined and the results are certain, for instance: EOQ model; and probabilistic models in which the input and output variables follow a probability distribution, for instance: Games Theory.

By the extent of generality: • The two models belonging to this class are: general models can be applied in general and does not pertain to one problem only, for instance: Linear programming; and specific model is applicable under specific condition only, for instance: Sales response curve or equation as a function of advertising is applicable in the marketing function alone.

Steps of Operations Research

Following are the steps which are involved in solving any operations research model:

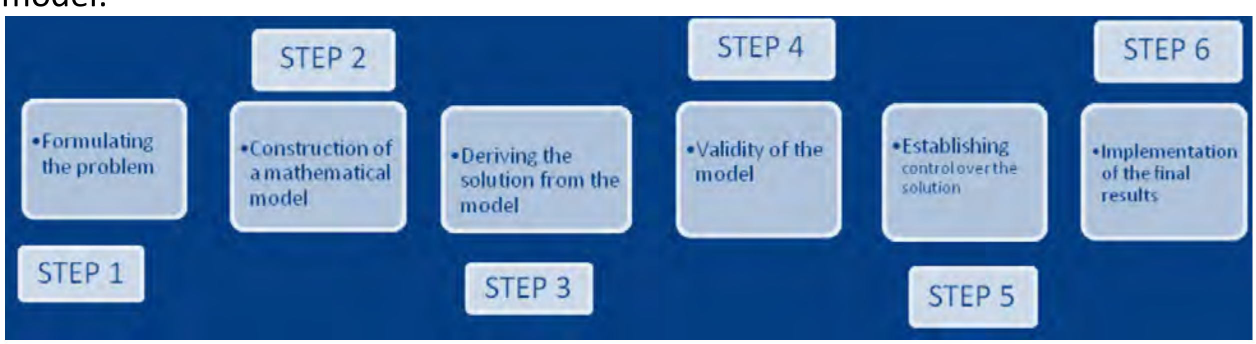


Fig. 1.1 Steps involved in operations research

Management Applications of Operations Research?

- Finance budgeting and investment

- Purchase, procure and exploration
- Production management
- Marketing
- Personal management
- Research and development

Scopes of OR

Agriculture: optimum allocation of land, crops, irrigation etc.

Finance: maximize income, profit, minimize cost etc.

In industries: Allocation of resources, assignment of problems to worthy employees etc.

Personal management: To appoint best candidate, decide minimum employees to complete job etc.

Production management: Determine number of units to produce

to maximize profit, etc.

Linear Programming

Introduction to Linear programming (LP)

Linear programming is designed by George B. Dantzig in 1947 to solve optimisation problem where all the constraints and Objectives functions are in the form of linear function. Linear programming is a technique of making decisions under the conditions of certainty that is all Objectivess and constraints are quantified.

Linear programming is the analysis of problems in which a linear function of a number of variables is to be optimised (maximise or minimised) when those variables are subject to a number of constraints in the mathematical linear inequalities.

Requirements of linear programming problems (LPP)

The common requirements of a LPP are as follows:

- Decision Variables And Their Relationship
- Well-Defined Objectives Function
- Existence Of Alternative Courses Of Action
- Non-Negative Conditions On Decision Variables

Basic assumptions of LPP

- **Linearity:** You need to express both the Objectives function and constraints as linear inequalities.
- **Deterministic:** All co-efficient of decision variables in the Objectives and constraints expressions are known and finite.
- **Additive:** The value of the Objectives function and the total sum of resources used must be equal to the sum of the contributions earned from each decision variable and the sum of resources used by decision variables respectively.
- **Divisibility:** The solution of decision variables and resources can be non-negative values including fractions.

General Form of Linear Programming

The LPP is a class of mathematical programming where the functions representing the Objectivess and the constraints are linear. Optimisation refers to the maximisation or minimisation of the Objectives functions.

The general form of linear programming

Maximise or Minimise: $Z = c_1 x_1 + c_2 x_2 + + c_n x_n$

Subject to the constraints,

$a_{11} x_1 + a_{12} x_2 + \text{-----} + a_{1n} x_n \sim b_1$

$a_{21} x_1 + a_{22} x_2 + \text{-----} + a_{2n} x_n \sim b_2$

$a_{m1} x_1 + a_{m2} x_2 + \text{-----} + a_{mn} x_n \sim b_m$

and $x_1 \geq 0, x_2 \geq 0, \text{--------------------} n \geq 0$

Where cj, bi and aij (i = 1, 2, 3, ….. m, j = 1, 2, 3 --- n) are constants determined from the technology of the problem and xj (j = 1, 2, 3 n) are the decision variables. Here ~ is either ≤ (less than), ≥ (greater than) or = (equal). Note that, in terms of the above formulation the coefficients cj, bi aij are interpreted physically as follows. If bi is the available amount of resources i, where aij is the amount of resource i that must be allocated to each unit of activity j, the “worth” per unit of activity is equal to cj.

For instance: A milk distributor supplies milk in bottles to houses in three areas A, B, C in a city. His delivery charge per bottle is 30 paisa in area A; 40 paisa in area B and 50 paisa in area C. He has to spend on an average, 1 minute to supply one bottle in area A; 2 minutes per bottle in area B and 3 minutes per bottle in area C. He can spare only 2 hours 30 minutes for this milk distribution but not more than one hour 30 minutes for area A and B together. The maximum number of bottles he can deliver is 120.

Find the number of bottles that he has to supply in each area so as to earn the maximum. Construct a mathematical model.

Solution: The decision variables of the model can be defined as follows:

x1 : Number of bottles of milk which the distributor supplies in Area A. x2 : Number of bottles of milk which the distributor supplies in Area B. x3 : Number of bottles of milk which the distributor supplies in Area C.

The Objectives:

$$\text{Maximise } Z = \frac{30}{100} x_1 + \frac{40}{100} x_2 + \frac{50}{100} x_3 \text{ in rupees}$$

Constraints:

Maximum number of milk bottles is 120 that is x1+x2+x3 ≤120.

Since he requires one minute per bottle in area A, 2 minutes per bottle in area B and 3 minutes per bottle in area C

and he cannot spend more than 150 minutes for the work, 1.x1 + 2.x2 + 3.x3 ≤150

Further, since he cannot spend more than 90 minutes for areas A and B. 1.x1+2.x2 ≤ 90.

Non-negativity $x1 \geq 0$, $x2 \geq 0$.

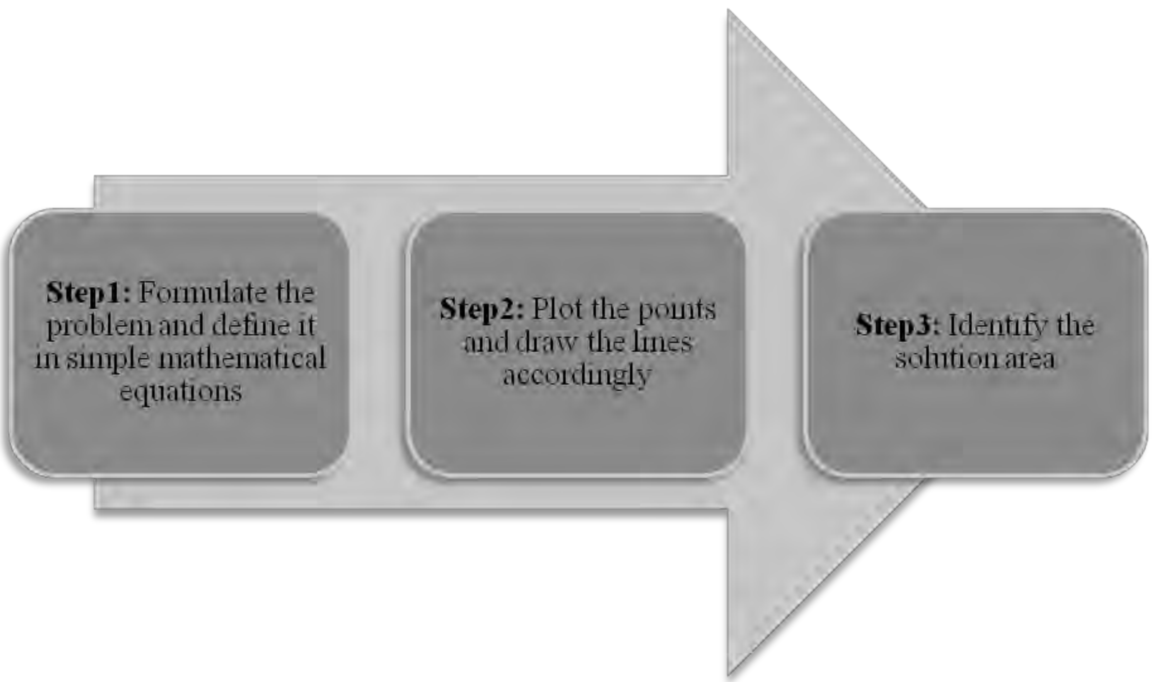
The problem can now be stated in the standard L.P. form as

Maximise $Z = 0.3x1 + 0.4x2 + 0.5x3$ Subject to

$x1 + x2 + x3 \leq 120$, $x1 + 2x2 + 3x3 \leq 150$, $x1 + 2x2 \leq 90$, and $x1 \geq 0$, $x2 \geq 0$

Graphical method

If we have clear Objectives function as well as associated constraints by formulating the linear programming model, our next step is to solve the problem and achieve the best possible optimal solution. Graphical method is one of the methods for solving linear programming problems. It includes the following steps:



Important Terms

- **Solution:** Value of decision variable of linear programming model are called solutions
- **Basic solution:** A basic solution of a system of m equations and n

variables ($m < n$) is a solution where at least n-m variables are zero

- **Feasible region:** Any non-negative value of (x_1, x_2) (i.e.: $x_1 \geq 0$, $x_2 \geq 0$) is a feasible solution of the LPP if it satisfies all the constraints. The collection of all feasible solutions is known as the feasible region.
- **Feasible solution:** The solution which satisfies all the constraints of linear programming problems is called a **feasible solution.**
- **Basic feasible solution:** A basic feasible solution of a system of m equations and n variables ($m < n$) is a solution **where m variables are non-negative (≥0) and n-m variables are zero.**
- **Optimal feasible solution:** Any feasible solution that optimises the Objectives function is called an optimal feasible solution.
- **Degenerate solution:** A basic solution is said to be degenerate if one or more basic variables become zero.
- **Infeasible solution:** The solution which do not satisfy all the constraints of linear programming problem.
    - **Convex:** A set X is convex if for any points x_1, x_2 in X, the line segment joining these points is also in X. That is, $x_1, x_2 \in X, 0 \leq \lambda \leq 1 \rightarrow \lambda x_2 + (1-\lambda)x_1 \in X$

      By convention, a set containing only a single point is also a convex set.

      $\lambda x_2 + (1-\lambda)x_1$ (where $0 \leq \lambda \leq 1$) is called a convex combination of x_1 and x_2.

      A point x of a convex set X is said to be an extreme point if there does not exist $x_1, x_2 \in X$ ($x_1 \neq x_2$) such that $x = \lambda x_2 + (1-\lambda)x_1$ for some λ with $0 < \lambda < 1$

- **Half plane:** A linear inequality in two variables is known as a half plane. The corresponding equality or the line is known as the boundary of the half- plane.
- **Convex polygon:** A convex polygon is a convex set formed by the intersection of finite number of closed **half-planes.**
- **Redundant constraint:** A redundant constraint is a constraint which does not affect the feasible region.

For instance: Solve the given LPP using the graphical method. Maximise $Z = 50x_1 + 80x_2$ Subject to the constraints

$1.0x_1 + 1.5x_2 \leq 600$

$0.2x_1 + 0.2x_2 \leq 100$

$0.0x_1 + 0.1x_2 \leq 30$

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

Solution: The horizontal axis represents x1 and the vertical axis x2. Plot the constraint lines and mark the feasibility

region as shown in the figure below:

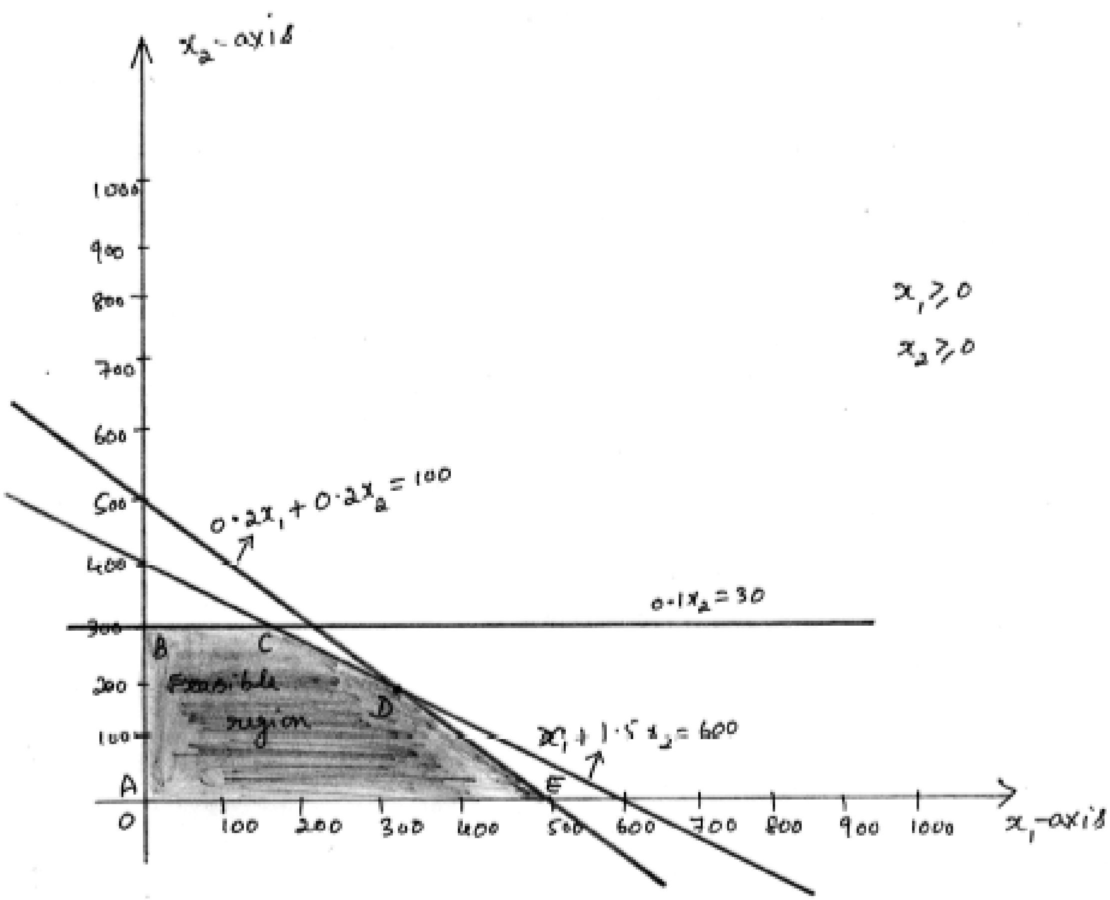


Fig. 2.1 Feasible region of two dimensional LPP

Any point on the thick line or inside the shaded portion will satisfy all the restrictions of the problem. The ABCDE is the feasibility region carried out by the constraints operating on the Objectives function. This depicts the limits within which the values of the decision variables are permissible.

The inter-section points C and D can be solved by the linear equations $x_2 = 30$; $x_1 + 1.5 x_2 = 600$, and $0.2x_1 + 0.2x_2 = 100$ and $x_1 + 1.5x_2 = 600$ That is C (150, 300) and D (300, 180).

The next step is to maximise revenues subject to the above shaded area.

At point	Feasible solution of the product-mix		Corresponding revenue		Total revenue
	x_1	x_2	From x_1	From x_2	
A	0	0	0	0	0
B	0	300	0	2400	24000
C	150	300	7500	24000	31500
D	300	180	15000	14,400	29400
E	500	0	25000	0	25,000

From the above table we find that maximum revenue is at Rs. 31,500 when 150 units of x_1 and 300 units of x_2 are produced.

Table 2.1 Revenue at different corner points

Mixed Constraint LP Problem

For instance: By using graphical method, find the maximum and minimum values of the function $Z = x – 3y$ where x and y are non-negative and subject to the following conditions:

$3x + 4y \geq 19$,
$2x – y \leq 9$
$2x + y \leq 15$
$x – y \geq – 3$

Solution: You can start by writing the constraints (conditions) to be satisfied by x, y in the following standard (less than or equal) form:

$– 3x – 4y \leq – 19$ $2x – y \leq 9$
$2x + y \leq 15$
$– x + y \leq 3$

Consider the equations:

$– 3x – 4y = – 19$, $2x – y = 9$, $2x + y = 15$, $– x + y = 3$, Both the above equations represent straight lines in the xy – plane. Denote the straight lines by L1, L2, L3 and L4 respectively.

You can see the lines L1, L2, L3 and L4 form a quadrilateral ABCD lying in the first quadrant of the xy – plane. You can see that the region bounded by this quadrilateral is convex.

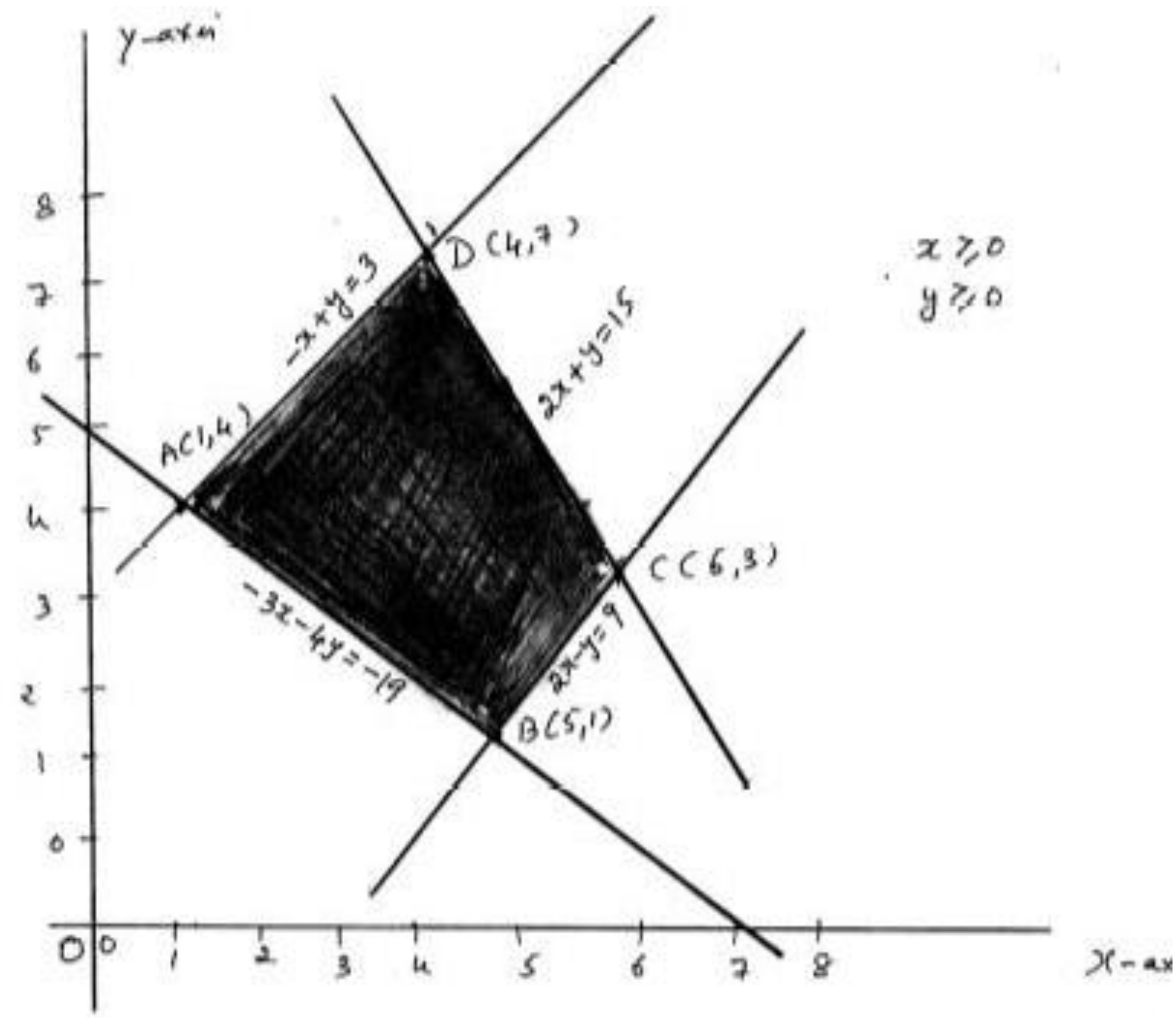


Fig. 2.2 Feasible region of two dimensional LPP

As such, the points (x, y) that lie within or on the boundary lines of this quadrilateral satisfy the inequalities $x \geq 0$, $y \geq 0$ and the constraints. The co-ordinates of the vertices A, B, C, D of the quadrilateral are obtained by solving equations taking two at a time, you will find that A (1, 4), B (5, 1), C (6, 3), D (4, 7), hence the solution is

Zat A(1, 4) = 1 – 3X4 = – 11
Zat B(5, 1) = 5 – 3X1 = 2
Zat C(6, 3) = 6 – 3X3 = – 3
Zat D(4, 7) = 4 – 3X7 = – 17

Z is highest at the vertex B and minimum at the vertex D. The maximum value of Z is Zat B (5, 1) = 2, which corresponds to x = 5, y = 1, and the minimum values of Z is –17 at D (4, 7), which corresponds to x = 4, y = 7.
Solved problem for unbound problem

For instance: Solve the given LPP in the graphical method. Maximise $Z = 2x_1+3x_2$

Subject to

$x_1 - x_2 \leq 2$

$x_1 + x_2 \geq 4$ and $x_1, x_2 \geq 0$

Solution:

The intersection point A of the straight lines $x_1 - x_2 = 2$ and $x_1+x_2 = 4$ is A (3, 1). Here the solution space is unbounded. The vertices of the feasible region are A (3, 1) and B (0, 4). Values of Objectives at these vertices

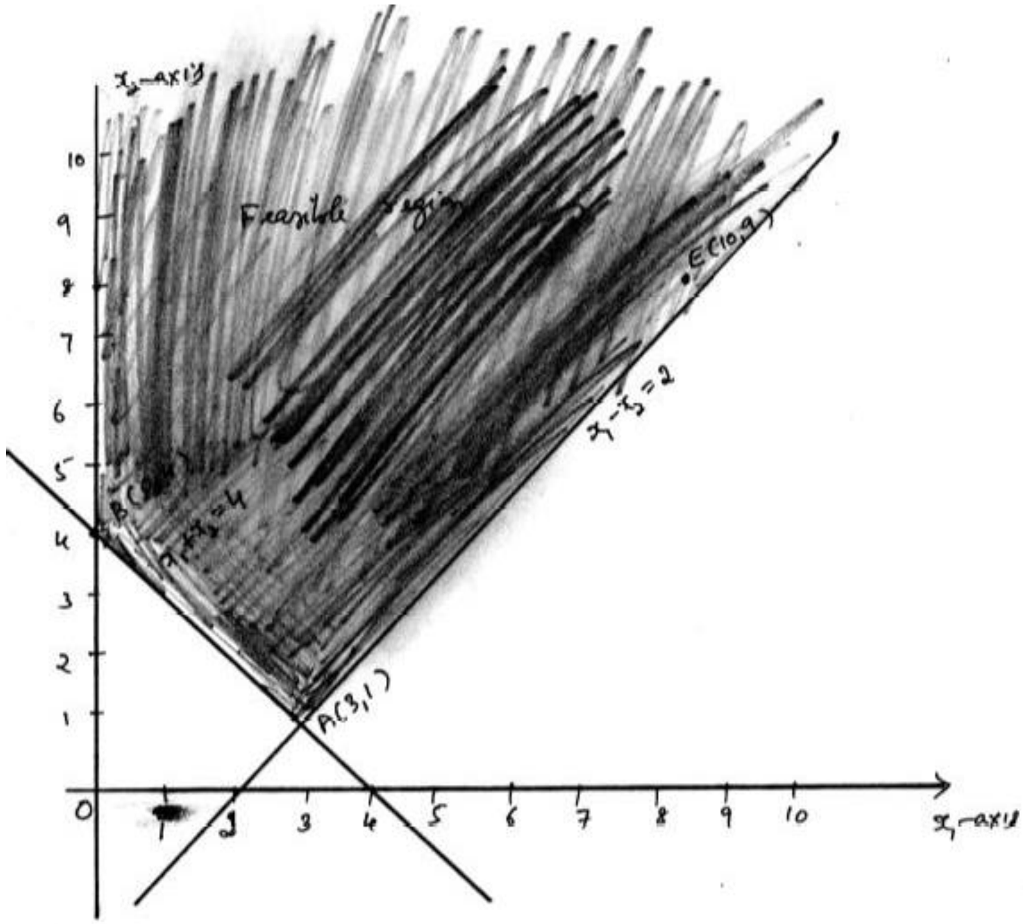


Fig. 2.3 Feasible region of two dimensional LPP

$\mathbf{Z_{\text{at A }(3,1)} = 2X3+3X1 = 9}$
$\mathbf{Z_{\text{at B }(0,\ 4)} = 2X0+4X3 = 12}$

But there are points in the convex region for which Z will have much higher values. For example, E (10, 9) lies in the shaded region and the value of Z there is at 47. In fact, the maximum value of Z occurs at infinity. Thus the problem has unbounded solutions.

Simplex Method

The simplex method focuses on solving LPP of any enormity involving two or more decision variables. Simplex method is the steps of algorithm until an optimal solution is reached. It is also known as iterative method. The simplex method simply selects the optimal solution amongst the set of feasible solutions of the problem.

To solve a problem by the simplex method, follow the steps below:

1. Introduce stack variables (Si's) for "≤" type of constraint.
2. Introduce surplus variables (Si's) and artificial variables (Ai) for "≥" type of constraint.
3. Introduce only Artificial variable for "=" type of constraint.
4. Cost (Cj) of slack and surplus variables will be zero and that of artificial variable will be "M"
5. Find Zj - Cj for each variable.
6. Slack and artificial variables will form basic variable for the first simplex table. Surplus variable will never become basic variable for the first simplex table.
7. Zj = sum of [cost of variable x its coefficients in the constraints – Profit or cost coefficient of the variable].
8. Select the most negative value of Zj - Cj. That column is called key column. The variable corresponding to the column will become basic variable for the next table.
9. Divide the quantities by the corresponding values of the key column to get ratios; select the minimum ratio. This becomes the key row. The basic variable corresponding to this row will be replaced by the variable found in step 6.
10. The element that lies both on key column and key row is called Pivotal element.
11. Ratios with negative and "α" value are not considered for determining key row.
12. Once an artificial variable is removed as basic variable, its column will be deleted from next iteration.
13. For maximisation problems, decision variables coefficient will be same as in the Objectives function. For minimisation

problems, decision variables coefficients will have opposite signs as compared to Objectives function.

14. Values of artificial variables will always be – M for both maximisation and minimisation problems.
15. The process is continued till all Zj - Cj ≥ 0.

For instance: Solve the following linear programming problem using simplex method.

Maximise $z = x_1 + 9x_2 + x_3$, Subject to $x_1 + 2x_2 + 3x_3 \leq 9$, $3x_1 + 2x_2 + 2x_3 \leq 15$

$x_1, x_2, x_3 \geq 0$.

Rewriting in the standard form

Maximise $z = x_1 + 9x_2 + x_3 + 0.S_1 + 0.S_2$ Subject to the conditions

$x_1 + 2x_2 + 3x_3 + S_1 = 9$

$3x_1 + 2x_2 + 2x_3 + S_2 = 15$

$x_1, x_2, x_3, S_1, S_2 \geq 0$.

Where S1 and S2 are the slack variables.

Solution: The initial basic solution is $S_1 = 9$, $S_2 = 15$

$$\therefore \mathbf{Xo} = \begin{pmatrix} S_1 \\ S_2 \end{pmatrix}, \mathbf{Co} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

The initial simplex table is given below.

	x_1 1	x_2 9	x_3 1	S_1 0	S_2 0		Ratio
S_1 0 S_2 0	1 3	2* 2	3 2	1 0	0 1	9 15	$\frac{9}{2} = 4.5 \leftarrow$ $\frac{15}{2} = 7.5$
$Z_j - c_j$	–1	–9 ↑	–1	0	0		

Work column* – pivot element.

S_1 – outgoing variable, x_2 incoming variable.

Since the three Zj – Cj are negative, the solution is not optimal. Choose the maximum negative value that is – 9. The

corresponding column vector x_2 enters the basis replacing S_1, since ratio is at minimum. You can use the elementary row operations to reduce the pivot element to 1 and other elements of work column to zero.

First iteration – The variable x1 becomes a basic variable replacing S1 and you obtain the following table.

	x_1	x_2	x_3	S_1	S_2	
	1	9	1	0	0	
X_2 9	$\frac{1}{2}$	1	$\frac{3}{2}$	$\frac{1}{2}$	0	$\frac{9}{2}$
S_2 0	2	0	– 1	– 1	1	6
$Z_j - C_j$	$\frac{9}{2}$	0	$\frac{25}{2}$	$\frac{9}{2}$	0	$\frac{81}{2}$

Table 2.3 First iteration table 1

Since all elements of the last row are non-negative, the optimal solution is obtained. The maximum value of the Objectives function Z is 81/2, achieved for x_2 = 9/2, S_2 = 6, are the basic variables. All other variables are non- basic.

Project Scheduling and PERT-CPM

Introduction to Project Scheduling

Some projects can be defined as a collection of inter-related activities which must be completed in a specified time according to a specified sequence and require resources, such as personnel, money, materials, facilities and so on. For instance like the projects of construction of a bridge, a highway, a power plant, repair and maintenance of an oil refinery and so on.

The growing complexities of today's projects had demanded more systematic and more effective planning techniques with the Objectives of optimising the efficiency of executing the project. Efficiency here refers to effecting the utmost reduction in the time required to complete a project while ensuring optimum utilisation of the available resources. Project management has evolved as a new field with the development of two analytic techniques for planning, scheduling and controlling projects. These are the Critical Path Method (CPM) and the Project Evaluation and Review Technique (PERT). PERT and CPM are basically time-oriented methods in the sense that they both lead to the determination of a time schedule.

PERT

A PERT chart is a project management tool used to schedule, organise, and co-ordinate tasks within a project. PERT stands for (Program Evaluation Review Technique), a methodology developed by the U.S. Navy in the 1950s to manage the Polaris submarine missile program.

Some key points about PERT are as follows:

- PERT was developed in connection with an R&D work. Therefore, it had to cope with the uncertainties that are associated with R&D activities. In PERT, the total project duration is regarded as a random variable. Therefore, associated probabilities are calculated so as to characterise it.
- It is an event-oriented network because in the analysis of a network, emphasis is given on the important stages of completion of a task rather than the activities required to be performed to reach a particular event or task.
- PERT is normally used for projects involving activities of non-repetitive nature in which time estimates are uncertain.
- It helps in pinpointing critical areas in a project so that necessary adjustment can be made to meet the scheduled completion date of the project.

CPM

The Critical Path Method (CPM) is one of several related techniques for doing project planning. CPM is for projects that are made up of a number of individual "activities." If some of the activities require other activities to finish before they can start, then the project becomes a complex web of activities. Some key points about PERT are as follows:

- CPM was developed in connection with a construction project, which consisted of routine tasks whose resource requirements and duration were known with certainty. Therefore, it is basically deterministic.
- CPM is suitable for establishing a trade-off for optimum balancing between schedule time and cost of the project.
- CPM is used for projects involving activities of repetitive nature.

Project Scheduling by PERT-CPM

It consists of three basic phases namely:

- planning
- scheduling
- controlling

Project Planning: In this phase following activities are performed:

- Identify various tasks or work elements to be performed in the project.
- Determine requirement of resources, such as men, materials, and machines, for carrying out activities listed above
- Estimate costs and time for various activities
- Specify the inter-relationship among various activities
- Develop a network diagram showing the sequential inter-relationships between the various activities

Project Scheduling: Once the planning phase is over, scheduling of the project starts where each of the activities required to be performed, is taken up. The various steps involved during this phase are listed below:

- Estimate the durations of activities. Take into account the resources required for these execution in the most economic manner
- Based on the above time estimates, a time chart showing the start and finish times for each activity is prepared.Use the time chart for the following exercises:
    a. To calculate the total project duration by applying network analysis techniques, such as forward (backward) pass and floats calculation.
    b. To identify the critical path
    c. To carry out resource smoothing (or levelling) exercises for critical or scarce resources including re-costing of the schedule taking into account resource constraints

Project Control: Project control refers to comparing the actual progress against the estimated schedule. If significant differences are observed then you need to re-schedule the project to update or revise the uncompleted part of the project.

PERT/CPM Network Components and Precedence Relationship

PERT/CPM Network Components and Precedence Relationship PERT/CPM networks consist of two major components as discussed below:

Events: An event represents a point in time that signifies the completion of some activities and the beginning of new ones. The beginning and end points of an activity are thus described by 2 events usually known as the tail and head events. Events are commonly represented by circles (nodes) in the network diagram. They do not consume time and resource.

Activities: Activities of the network represent project operations or tasks to be conducted. An arrow is commonly used to represent an activity, with its head indicating the direction of progress in the project. Activities originating from a certain event cannot start until the activities terminating at the same event have been completed. They consume time and resource.

Events in the network diagram are identified by numbers. Numbers are given to events such that the arrow head number is greater than the arrow tail number.

Activities are identified by the numbers of their starting (tail) event and ending (head) event.

In Fig. 7.1 the arrow (P.Q) extended between two events represents the activity. The tail event P represents the start of the activity and the head event Q represents the completion of the activity.

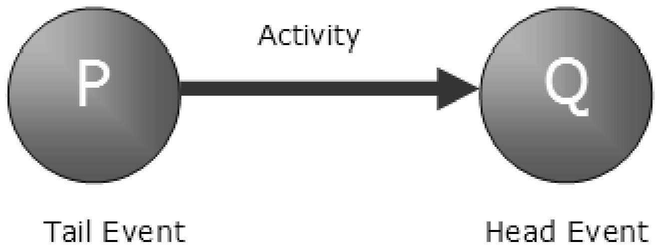


Fig. 7.1 Basic PERT-CPM network

Fig. 7.2 is example of another PERT-CPM network with activities (1, 3), (2, 3) and (3, 4). As the figure indicates,

activities (1, 3) and (2, 3) need to be completed before activity (3, 4) starts.

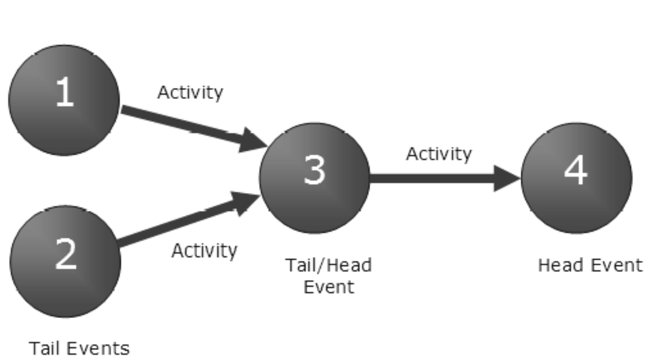


Fig. 7.2 A PERT-CPM network

The rules for constructing the arrow diagram are as follows:

- Each activity is represented by one and only one arrow in the network
- No two activities can be identified by the same head and tail events
- To ensure the correct precedence relationship in the arrow diagram, we need to answer the following points as we add every activity to the network:

② What activities must be completed immediately before these activity can start

② What activities must follow this activity

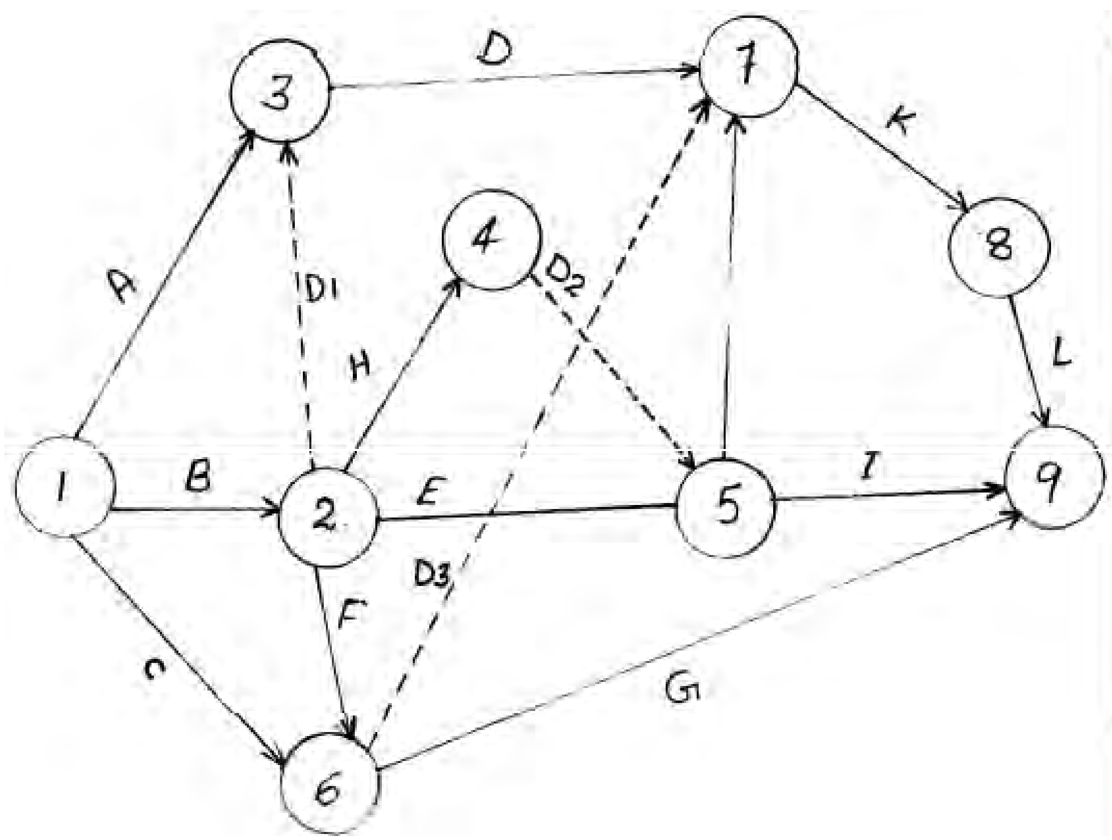
② What activity must occur concurrently with this activity

This rule is self-explanatory. It actually allows for checking (and rechecking) the precedence relationships as one progresses in the development of the network.

For instance: Construct the arrow diagram comprising activities A, B, C ……..and L such that the following relationships are satisfied:

1) A, B and C the first activities of the project, can start simultaneously.
2) A and B precede D.
3) B precedes E, F and H.
4) F and C precede G.
5) E and H precede I and J.
6) C, D, F and J precede K.
7) K precedes L.
8) I, G and L are the terminal activities of the project.

Solution:



Note: A dummy activity in a project network analysis has zero duration.

Critical Path Calculations

The application of PERT/CPM should ultimately yield a schedule specifying the start and completion time of each activity. The arrow diagram is the first step towards achieving that goal. The start and completion timings are calculated directly on the arrow diagrams using simple arithmetic. The end result is to classify the activities as critical or non-critical.

An activity is said to be critical if a delay in the start of the course makes a delay in the completion time of the entire project.

A non-critical activity is such that the time between its earliest start and its latest completion time is longer than its actual duration. A non-critical activity is said to have a slack or float time.

Determination of the Critical Path

A critical path defines a chain of critical activities that connects the start and end events of the arrow diagram. In other words, the critical path identifies all the critical activities of a project.

The critical path calculations are done in two phases:

1. The first phase is called the Forward Pass. In this phase all calculations begin from the start node and move to the end node. At each node a number is computed representing the earliest occurrence time of the corresponding event. These numbers are shown in squares. Here we note the number of heads joining the event. We take the maximum earliest timing through these heads.
2. The second phase is called the Backwards Pass. It begins calculations from the "end" node and moves to the "start" node. The number computed at each node is shown in a triangle near the end point, which represents the latest occurrence time of the corresponding event. In backward pass, we see the number of tails and take minimum value through these tails.

Let ES_i be the earliest start time of all the activities emanating from event i. Then ES_i represents the earliest occurrence time of event i.

If i = 1 is the "start" event then conventionally for the critical path calculations, ES_i = 0. Let D_{ij} be the duration of the activity (i, j).

Then the forward pass calculations for all defined (i, j) activities with ES_i=0 is given by the formula:

$$\mathbf{ES_j = maxi\{ES_i + D_{ij}\}}$$

Therefore, to compute ES_j for event j, we need to first compute ES_i for the tail events of all the incoming activities (i, j).

With the computation of all ES_j, the forward pass calculations are completed. The backward pass starts from the "end" event. The Objectives of the backward pass phase is to calculate LC_i, the latest completion time for all the activities coming into the event i.

Thus, if i = n is the end event, $LC_n = ES_n$ initiates the backward pass.

In general for any node i, we can calculate the backward pass for all defined activities using the formula:

$$\mathbf{LC_i = min\{LC_j - D_{ij}\}}$$

We can now identify the critical path activities using the results of the forward and backward passes. An activity (i,j) lies on the critical path if it satisfies the following conditions:

A. $ES_i = LC_i$
B. $ES_j = LC_j$
C. $ES_j - ES_i = LC_j - LC_i = D_{ij}$

These conditions actually indicate that there is no float or slack time between the earliest stand and the latest start of the activity. Thus, the activity must be critical.

Thus, the activity must be critical. In the arrow diagram these are characterised by same numbers within rectangles and triangles at each of the head and tail events. The difference between the numbers in rectangles or triangles at the head event and the number within rectangles or triangles at the tail event is equal to the duration of the activity. Thus, we will get a critical path, which is a chain of connected activities, spanning the network from start to end. For instance: Consider a network which stands from node 1 and terminates at node 6, the time required to perform each activity is indicated on the arrows.

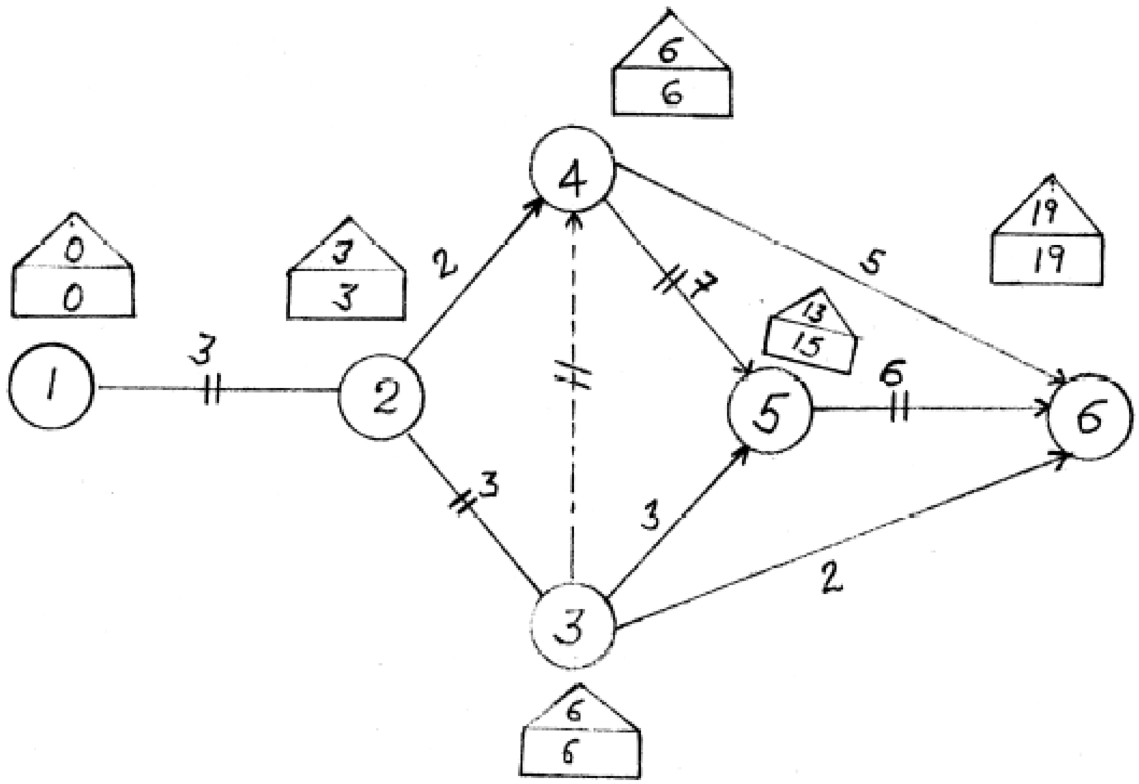


Fig. 7.3 Analysis work

Solution: Let us start with forward pass with ESi = 0.

Since there is only one incoming activity (1, 2) to event 2 with D_{12} = 3.

$ES_2 = ES_1 + DS_2 = 0+3=3$.

Let us consider the end 3, since there is only one incoming activity (2, 3) to event 3, with D_{23} = 3.

$ES_3 = ES_2 + D_{23} = 3+3 = 6$.

To obtain ES_4, since there are two activities A (3, 4) and (2, 4) to the event 4 with $D_{24} = 2$ and $D_{34} = 0$.

$ES_4 = \max_{i=2, 3} \{ES_i + D_{e4}\}$

$= \max \{ES_2 + D_{24}, ES_3 + D_{34}\}$

$= \max \{3+2, 6+0\} = 6$

Similarly, $ES_5 = 13$ and $ES_6 = 19$. This completes the first phase. In the second phase we have $LC_6 = 19 = ES_6$

$LC_5 = 19-6 = 13$

$LC_4 = \min_{J = 5, 6} \{LCj - D4_j\} = 6$

$LC_s = 6$, $LC_2 = 3$ and $LC_1 = 0$

- Therefore, activities (1, 2), (2, 3) (3, 4) (4, 5) (5, 6) are critical and (2, 4) (4, 6), (3, 6), are non-critical.
- Thus, the activities (1, 2), (2, 3), (3, 4), (4, 5) and (5, 6) define the critical path which is the shortest possible time to complete the project.

Determination of Floats

Following the determination of the critical path, we need to compute the floats for the non-critical activities. For the critical activities this float is zero. Before showing how floats are determined, it is necessary to define two new times that are associated with each activity. These are as follows:

- **Latest Start** (LS) time and
- **Earliest Completion** (EC) time

We can define activity (i, j) for these two types of time by

$LS_{ij} = LC_j - D_{ij}$

$EC_{ij} = ES_i + D_{ij}$

There are two important types of floats namely:

- Total Float (TF)
- Free Float (FF)

The total float TF_{ij} for activity (i, j) is the difference between the maximum time available to perform the activity (= $LC_j – ES_i$) and its duration (= D_{ij})

$TF_{ij} = LC_j – ES_i – D_{ij} = LC_j – EC_{ij} = LS_{ij} – ES_i$

The free float is defined by assuming that all the activities start as early as possible. In this case FF_{ij} for activity (i,j) is the excess of available time (= $ES_i – ES_i$) over its deviation (= D_{ij}); that is, $FF_{ij} = ES_i – ES_i = D_{ij}$

Note

For critical activities float is zero. Therefore, the free float must be zero when the total float is zero. However, the converse is not true, that is, a non-critical activity may have zero free floats.

Let us consider the example taken before the critical path calculations. The floats for the non-critical activities can be summarised as shown in the following table:

Activity (i j)	Duration D_{ij}	Earliest		Latest		Table Float TF_{ij}	Free Float FF_{ij}
		Start ES_i	Completion EC_{ij}	Start LS_{ij}	Completion ΔLC_j		
(1, 2)	3	0	3	0	3	0*	0
(2, 3)	3	3	6	3	6	0*	0
(2, 4)	2	3	5	4	6	1	1
(3, 4)	0	6	6	6	6	0*	0
(3, 5)	3	6	9	10	13	4	4
(3, 6)	2	6	8	17	19	11	11
(4, 5)	7	6	13	6	13	0*	0
(4, 6)	5	6	11	14	19	8	8
(5, 6)	6	13	19	13	19	0*	0

Table 7.1 Float for non-critical activities

Total float = ES_{ij} = LF_{ij} - ES_{ij}

Free float = Total float - - Head slack

For instance: A project consists of a series of tasks A, B, C, – D, – E, F, G, H, I with the following relationships:

- W < X,Y means X and Y cannot starts until W is completed
- X,Y < W means W cannot start until both X and Y are completed

With this notation construct the network diagram having the following constraints A < D, E; B, D < F; C < G, B < H; F, G < I.

Also find the minimum time of completion of the project, the critical path, and the total floats of each task, when the time (in days) of completion of each task is as follows:

Task	A	B	C	D	E	F	G	H	I
Time	23	8	20	16	24	18	19	4	10

Solution:

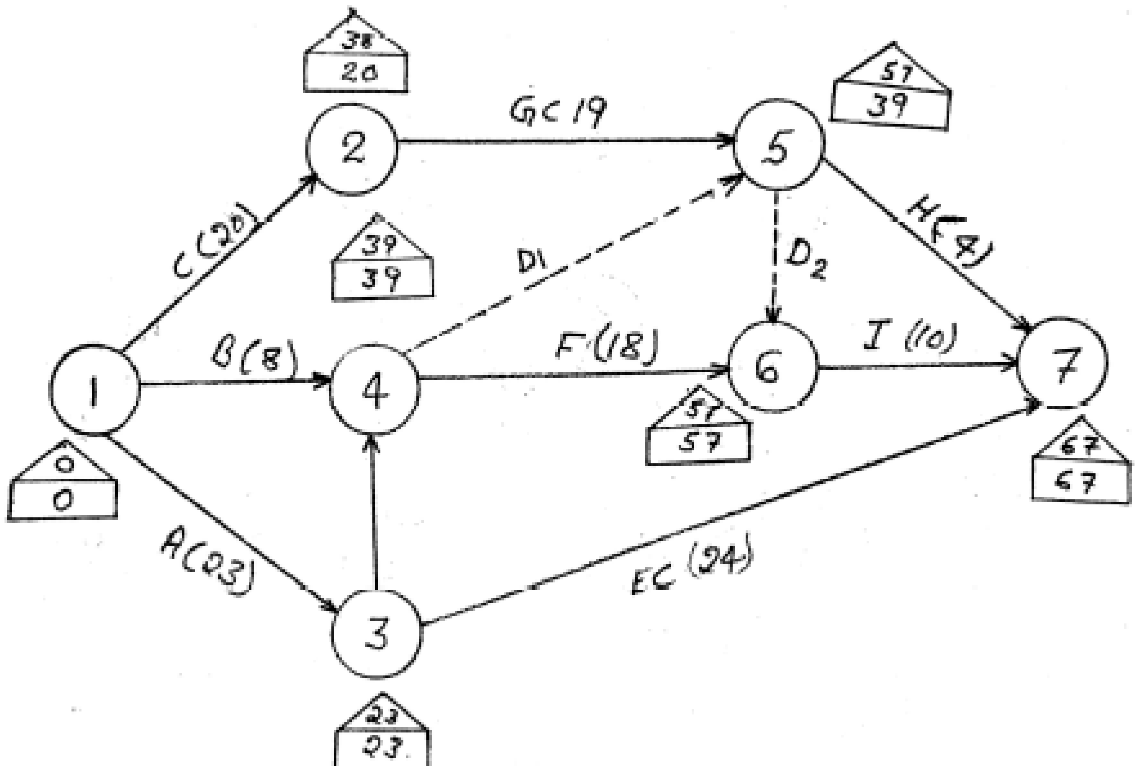


Fig. Analysis table

$ES_1 = 0$, $ES_2 = 20$, $ES_3 = 23$, $ES_4 = 59$, $ES_5 = 39$, $ES_6 = 57$, $ES_7 = 67$

Activity (i, j)	Duration D_{ij}	Earliest		Latest		Table Float TF_{ij}	Free Float FF_{ij}
		Start ES_e	Finish Ee_{ij}	Start L_j-D_{ij}	Finish L_j		
(1, 2)	20	0	20	18	38	18	0
(1, 3)	23	0	23	0	23	0*	0
(1, 4)	8	0	8	31	39	31	31
(2, 5)	19	20	39	38	57	18	0
(3, 4)	16	23	39	23	39	0*	0
(3, 7)	24	23	47	43	67	20	20
(4, 5)	0	39	39	57	57	10	0
(4, 6)	18	39	57	39	57	0*	0
(5, 6)	0	39	39	57	57	18	18
(5, 7)	4	39	43	63	67	24	24
(6, 7)	10	37	67	57	67	0*	0

Table 7.2 Activity table

Critical path is 1 – 3 – 4 – 6 – 7.

Project Management – PERT

The analysis in CPM does not take in the cases where time estimates for the different activities are probabilistic. It

also does not consider explicitly the cost of schedules. Here we will consider both probability and cost aspects in project scheduling.

Probability considerations are incorporated in project scheduling by assuming that the time estimate for each activity is based on 3 different values. They are as follows:

a = the optimistic time, which will be required if the execution of the project goes extremely well. b = the pessimistic time, which will be required if everything goes bad.

m = the most likely time, which will be required if execution is normal.

The most likely estimate m need not coincide with the mid-point $\frac{a+b}{2}$ of a and b.

Then the expected duration of each activity D can

be obtained as the mean $\frac{a+b}{2}$ and 2m. Therefore,

$$\bar{D} = \frac{\frac{a+b}{2} + 2m}{3} = \frac{a+b+4m}{6}$$

We can use this estimate to study the single estimate D in the critical path calculation.
The variance of each activity denoted by V is defined by,

$$\text{Variance } V = \left(\frac{b-a}{6}\right)^2$$

The earliest expected times for the node i is denoted by E(μi). For each node i, E(μi) is obtained by taking the sum of expected times of all activities leading to the node i, when more than one activity leads to a node i, then the greatest of all E(μi) is chosen. Let μi be the earliest occurrence time of the event i, we can consider μi as a random variable. Assuming that all activities of the network are statistically independent, we can calculate the mean and the variance of μi as follows:

$$E\{\mu i\} = ES_i \text{ and } Var\{\mu i\} = \sum_k V_k$$

Where, k defines the activities along the largest path leading to i.
For the latest expected time, we consider the last node. Now for each path move backwards and substitute the Dij for each activity (i, j).
Thus we have,

$$E(L_j) = E(\mu a)$$
$$E(\mu i) = L(L_j) – D_{ij}$$

if only one path events from j to i or if it is the minimum of {E[Lj – Dij] for all j for which the activities (i, j) is
defined.

Note: The probability distribution of times for completing an event can be approximated by the normal distribution due to central limit theorem.

Since μi represents the earliest occurrence time, event will meet a certain schedule time STi (specified by an analyst)
with probability

$$Pr\ (\mu_i \leq ST_i) = Pr\left(\frac{\mu_i - E(\mu_i)}{\sqrt{V(\mu_i)}} \leq \frac{ST_i - E(\mu_i)}{\sqrt{V(\mu_i)}}\right)$$
$$= Pr\ (Z \leq K_i)$$

$$Z \sim N(01) \text{ and } K_i = \frac{ST_i - E(\mu_i)}{V(\mu_i)}$$

Where,

It is a common practice to compute the probability that event i will occur no later than its LC_e. Such probability will represent the chance that the succeeding events will occur within the (ES_e, LC_e) duration.

For instance: A project is represented by the network shown below and has the following data.

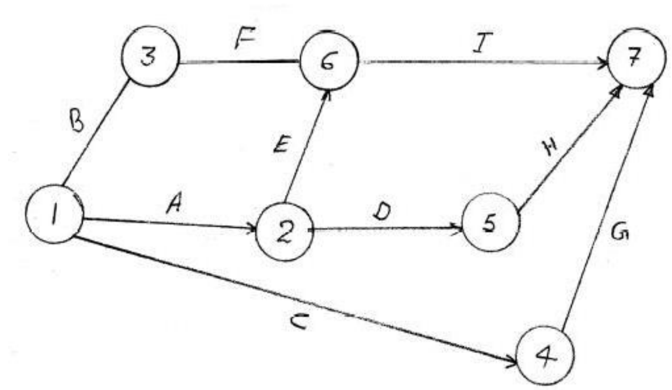


Fig. 7.5 Analysis network

Task	A	B	C	D	E	F	G	H	I
Optimistic Time	5	18	26	16	15	6	7	7	3
Pessimistic Time	10	22	40	20	25	12	12	9	5
Most Likely Time	8	20	33	18	20	9	10	8	4

Table 7.3 Data table

Determine the following:

a) Expected task time and their variance

b) The earliest and latest expected times to reach each event

c) The critical path

d) The probability of an event occurring at the proposed completion data if the original contract time of completing the project is 41.5 weeks.

e) The duration of the project that will have 96% chances of being completed.

Solution: Using the formula we can calculate expected activity times and variance in the following table

$$\bar{D} = \frac{1}{6}(a + b + 4m) \quad V = \left(\frac{b - a}{6}\right)^2$$

A)

Activity	A	B	m		v
1-2	5	10	8	7-8	0.696
1-3	18	22	20	20-00	0.444
1-4	26	40	33	33-0	5.429
2-5	16	20	18	18-0	0.443
2-6	15	25	20	20-0	2.780
3-6	6	12	9	9-0	1.000
4 7	7	12	10	9 8	0.694
5-7	7	9	8	8-0	0.111
6-7	3	5	4	4-0	0.111

Earliest and latest expected time for event

Forward Pass:

E1 = 0 E2 = 7.8 E3 = 20 E4 = 33 E5 = 25-8 E6 = 29 E7 = 42.8

Backward Pass:

L7 = 42.8 L6 = 38.8 L5 = 34.8 L4 = 33.0 L3 = 29.8 L2 = 16.8 L1 = 0.
The E-values and L-values are shown in figure 7.6

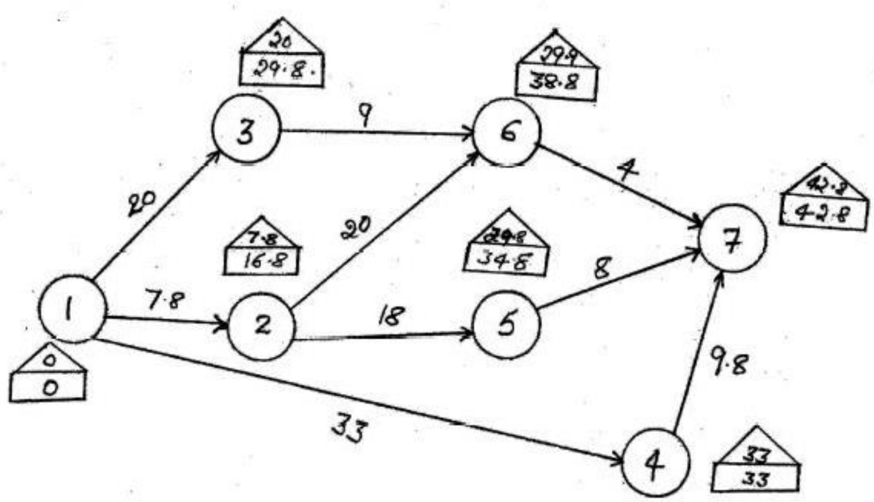


Fig. Network analysis

B) The E-values and L-values are shown in figure. The critical path is shown by thick line in the figure. The critical path is 1-4-7 and the earliest completion time for the project is 42.8 weeks.

C) The last event 7 will occur only after 42.8 weeks. For this we require only the duration of critical activities. This will help us in calculating the standard duration of the last event.

Expected length of critical path = 33 + 9.8 = 42.8

Variance of article path length = 5.429 + 0.694 = 6.123

Probability of meeting the schedule time is given by (From normal distribution table)

$P_i (Z \leq K_i) = P_i (Z - 0.52) = 0.30$

Thus, the probability that the project can be completed in less than or equal to 41.5 weeks is 0.30. In other words probability that the project will get delayed beyond 41.5 weeks is 0.70.

D) Given that P (Z ≤ Ki) = 0.95. But Z0.9S = 1.6 u, from normal distribution table

Then 1.6 u

$$= \frac{ST_i - E(\mu_i)}{\sqrt{V(\mu_i)}} \text{ is } 1.6\text{ u} = \frac{ST_i - 42.8}{2.47} \text{ or}$$

S_{ji} = 1.64 X 2.47 + 42.8 = 46.85 weeks.

It has been discovered that every LPP has been associated with another LPP. One of these LPP is called Prime while the other LPP will be called Dual. Sometimes, the solution to a dual is easier than the primal so it is better to convert at that time primal into its dual.

Primal LPP

Suppose following LPP is given Max $Z_x = 3x_1 + 5x_2$ *subject to*

$x_1 <= 4;\ x_2 <= 6;\ 3x_1 + 2x_2 <= 18;$ *and* $x_1, x_2 >= 0$

Its corresponding dual will be as follows

Dual LPP

Min $Z_w = 4w_1 + 6w_2 + 18w_3$ *subject to*

$$w_1 + 3w_3 >= 3;\ w_2 + 2w_3 >= 5;\ \text{and } w_1, w_2, w_3 >= 0$$

Matrix Form of Primal and Dual Suppose the matrix for LPP is

Min $Zx = Cx$ *(Primal objective function),* $C \in R^n$

Subject to $AX <= b$, $b \in R^n$ Where A is an mXn real matrix

Dual of above LPP will be

Minimize $z_w = b^T w$, $b \in R^n$ Subject to

$$A^t w >= c^T,\ C \in R^n$$

Where $w = (w_1, w_2, ..w_m)$ *and* A^T, b^T, c^T *are the transpose of A, B and C*

General rules to convert primal into dual

i.Convert objective function to max form (min z = -*min z')*

ii.Bring all inequalities to <= *(* >= *can be written as* -<=*)*

iii.Equality signs can be written as >= *and* <= *so a*=*4 means a*>=*4 and a*<=*4; ie. –a*<=*-4 and a* <=*4*

iv.Write unrestricted variables c as c' – c'' where c', c'' >=*0*

v.Transpose the rows and columns of constraint coefficients

vi.Transpose the objective function coefficients (c's) and right hand constants (b's)

vii.Change the inequalities from <= *to* >=

viii.Minimize the objective function from maximize

Duality is fairly simple, Try few numeric exercises

CHAPTER – II
SEQUENCING MODELS

The basic models in scheduling due to Johnson (1957) and owing to Maggu & Das (1977) T.P. Singh (1985, 86, 2005, 2006) are explained one by one which form a basis of scheduling problems dealt in this dissertation. Johnson's (1954) considered the very simple case of n jobs to be processed on two machines A and B, each job requiring the same sequence of operations and no passing allowed. Whichsoever jobs is processed first on machine A must also be processed on machine B and whichsoever job is processed second on machine A also be processed on machine B i.e. the flow shop model.

2.1 Notations, Terminology and Assumptions

NOTATIONS

t_{ij} = Processing time (time required) for job I on machine j.

T = Total elapsed time for processing all the jobs. This includes idle time, if any.

T_{ij} = Idle time on machine j from the end of job (i-1) to the start of job i.

TERMINOLOGY

- Number of Machines: The number of machines refer to number of service facilities through which a job must pass before it is assumed to be completed.
- Processing time: The time required by a job on

each machine.

- Processing Order: It is the order (sequence) in which machines are required for completing the job.
- Idle Time on a Machine It is the time for which a machine does not have a job to process, i.e. idle time from the end of job (i-1) to the start of job i.
- Total Elapsed Time: It is the time interval between starting the first job and completing the last job including the idle time (if any) in a particular order by the given set of machines.
- No Passing Rule: It refers to the rule of maintaining the order in which jobs are to be processed on given machines.

Assumptions:

1. The processing time on different machine are machines are exactly known and are independent of the order of the jobs in which they are to be processed.
2. The time taken by the job in moving from one machine is negligible.
3. Once a job has begun on a machine, it must be completed before another job can begin on the same machine.

4. All jobs are known and are ready for processing before the period under consideration begins.
5. Only one job can be processed on a given machine at a time.
6. Machines to be used are of different types.
7. The orders of completion of jobs are independent of he sequence of jobs.

2.2 Processing n Jobs through two machines

Let there be n jobs, each of which is to be processed through two machines. M_1 and M_2 in the order M_1 M_2 i.e. each job has to be pass through the same sequence of operations. In other words a job is assigned to machine M_1 first and after it has been completely processed on machine M_1, it is assigned to machine M_2. If the machine M_2 is not free at the moment for processing the same job, them the job has to wait in a waiting line for its turn on machine M_2, so passing is not allowing.

Since passing is not allowed, therefore machine M_1 will remain busy in processing all the n jobs one-by-one while machine M_2 may remain idle time of the second machine. This can be achieved only by determining sequence of n jobs which are to be processed on two machines M_1 and M_2. The procedure suggested by Johnson for determining the optimal sequence can be summarized as follows.

The Algorithm

Step 1 List the jobs along with their processing times in a table as shown below:

Processing Time on Machine	Job Number				
	1	2	3	….	N
M_1	t_{11}	t_{12}	t_{13}	….	t_{1n}
M_2	T_{21}	T_{22}	T_{23}	….	T_{2n}

Step 2 Examine the columns for processing times on machines M_1 and M_2, and find the smallest processing time in each column, i.e. find out, min(t_{1j}, t_{2j}) for all j.

Step 3(a) If the smallest processing time is for the first machine M_1, then place the corresponding job in the first available position in the sequence. If it is for the second machine, the place the corresponding job in the last available position in the sequence.

(b) If there is a tie in selecting the minimum of all the processing times, then there may be three situations:

(i) Minimum among all processing times is same for the machine, i.e. min $(t_{1j}, t_{2j}) = t_{1k} = t_{2r}$, then process the kth job first and the rth job last.

(ii) If the tie for the minimum occurs among processing times t_{1j} on machine M_1 only, then select the job corresponding to the smallest job subscript first.

(iii) If the tie for the minimum occurs among processing time t_{2j} on machine M_2, then select the job corresponding to the largest job corresponding to the largest job subscript last.

Step 4 Remove the assigned jobs from the table. If the table is empty, stop and go to step 5. Otherwise, go to Step 2.

Step 5 Calculate idle time for machines M_1 and M_2:

(a) Idle time for machine M_1 = (Total elapsed time)-(Time when the last job in a sequence finishes on machine M_1)

(b) Idle time for machine M_2 = Time at which the first job in a sequence finishes on machine M_1

$+\sum_{j=2}^{n}$ {Time when the job in a sequence starts on Machine M_2}

-{Time when the (j-1)th job in a sequence finishes on Machine M_2}

Step 6 The total elapsed time to process all jobs through two machines is given by

Total elapsed time = Time when the nth job in a sequence finishes on machine M_2.

$$= \sum_{j=1}^{n} M_{2j} + \sum_{j=1}^{n} I_{2j}$$

Where M_{2j} = Time required for processing jth job on machine M_2.

I_{2j} = Time for which machine M_2 remains idle after processing (j-1)th job and before starting work in jth job.

2.3 Processing n Jobs through three machines

Johnson provides an extension of his procedure to the case in which there are three instead of two machines, each job is to be processed through three machines M_1, M_2 and M_3 in the order M_1, M_2 M_3. The list of jobs with their processing times is given below. An optimal solution to this problem can be obtained if either or both of the following conditions hold good:

Processing Time on Machine	Job Number 1	2	3	….	N
M_1	t_{11}	t_{12}	t_{13}	….	t_{1n}
M_2	T_{21}	T_{22}	T_{23}	….	T_{2n}
M_3	T_{31}	T_{32}	T_{33}	….	T_{3n}

1. The minimum processing time on machine M_1 is at least as great as the maximum processing time on machine M_2, that is,

$$\min t_{1j} \geq \max t_{2j}, \qquad j = 1, 2, \ldots\ldots, n$$

2. The minimum processing time on machine M_3 is at least as great as the maximum processing time on machine M_2, that is,

$$\min t_{3j} \geq \max t_{2j}, \qquad j = 1, 2, \ldots\ldots, n$$

If either or both the above conditions hold good, then the steps of the algorithm can be summarized in the following steps.

The Algorithm

Step 1 Examine processing time of given jobs on all three machines and if either one or both the above conditions hold, then go to Step 2, otherwise the algorithm fails.

Step 2 Introduce two fictitious machines, say G and H with corresponding processing times given by

G_i and H_i are defined by $G_i - A_i + B_i$, $H_i = B_i + C_i$

Then applying the Johnson's job 2 machine algorithm

2.4 Processing n jobs through m machines

Let there be n jobs, each of which is to be processed through m machines, say M_1, M_2, ... M_m in the order M_1 M_2 ... M_m. The optimal solution to this problem can be obtained if either or both of the following condition hold good.

(a) Min $\{t_{1j}\} \geq$ Max $\{t_{ij}\}$; j = 2, 3, ..., m-1

and/or (b) Min $\{t_{mj}\} \geq$ Max $\{t_{ij}\}$; j = 2, 3, ..., m-1

that is, the minimum processing time on machines M_1 and M_m is as great as the maximum processing time on any of the remaining (m-1) machines.

If either or both these conditions hold good, then the steps of the algorithm can be summarized in the following steps.

Step 1 Find, Min $\{t_{1j}\}$, Min$\{t_{mj}\}$ and Max$\{t_{ij}\}$ and verify above conditions. If either or both the conditions mentioned above hold, then go to Step 2. Otherwise the algorithm fails.

Step 2 Convert m-machine problem into 2-machine problem by introducing two fictitious machines, say G and H with corresponding processing times given by

(i) $t_{Gj} = t_{ij} + t_{2j} + \ldots + t_{m-1j}$; j= 1, 2, ..., n

i.e. processing time of n-jobs on machine G is the sum of the processing times on machines M_1, M_2, ...$M_{m-1,j}$

(ii) $t_{Hj} = t_{2j} + t_{3j} + \ldots + t_{mj}$; j= 1, 2, ..., n

i.e. processing time of n-jobs on machine H is the sum of the processing times on machines M_1, M_2, ...$M_{m-1,j}$

Step 3 The new processing times so obtained can now be used for solving n-job, two-machine equivalent sequencing problem with the prescribed ordering HG in the same way as discussed earlier.

2.5 Concept of equivalent job in flow shop

The most common optimizer in flow shop requiring has been to minimize the total elapsed time (make span or maximum flow time) for a set of n independent jobs to be processed over m ordered machines. Now consider a situation in which some sets of specified jobs are required to be processed together as a block in a sequence either by virtue of technological constraint or some externally imposed restriction. This type of situation is known as a **Group Technology** which has very wide applications to a variety of production systems for the purpose of improving the productivity. The problem of determining an optimal sequence under the stated restriction is difficult to be stored with the help of available method.

2.6 Equivalent job block theorem given by Maggu & Das

Let there be two jobs i and j is a sequence S to be processed on two machines A and B in the order A B. Let the equivalent job of i and j b denoted by a'

Then $A_a = A_i + A_j - \min(A_j, B_i)$

$B_a = B_i + B_j - \min(A_j, B_i)$

Where A_a and B_a denote the processing time of equivalent job 'a' on machine A and B respectively.

2.7 Concept of Transportation time in flow shop:

In many practical situations of scheduling it is seen that machines are distantly situated and therefore, definite finite time is taken in transporting the job from one machine to another if the form of.

i) Loading time of jobs.
ii) Moving time of jobs.
iii) Unloading time of jobs.

The sum of all the above times has been designated by various researches as transportation time of a job. This transportation time is the amount of time required to dispatch the job i after it has been completed on machine A, to the next succeeding machine B for its onward processing. It is denoted by t_i for job i.

2.8 Flow Shop problem with break down of machines

it has been assumed so for that no machine fails and hence no disturbance occurs in the processing of the jobs. Many a times it is practically possible that machine may not work:

i) Due to failure of electric supply from mains. Or
ii) Machines stop working due to failure of one or more components suddenly.
iii) Machines are required to stop for certain interval of time due to excessive leading or some other external course. Therefore now we are considered to take into account the effect of break down interval of machines on the completion time of the set of jobs. Now we presses a heuristic method for providing an optimal or near optimal solution for a n x 2 flow shop problem.

Step of the Algorithm

Step 1: Determine the optimal sequence so for the given n x 2 flow shop problem by Johnson's algorithm.

Step 2: Find the optimal total elapsed time for the sequence so

Step 3: Locate the time intervals for jobs r.t the break down intervals overlaps with these. Two cases are possible.

i) Either the break down interval is not present practically or fully in any time interval of the completion of jobs on any machine. In this cases effect of the break-down on the total elapsed time, and sequence so obtained in step 1 is optimal stop the process.

ii) If the break down interval is contained practically or fully in some of the completion time intervals of jobs. Go to step 4.

Step 4: Formulate a new problem with processing time A'_i and B'_i for job i on machine A and B respectively s.t

$$A'_i = A_i + (b-a)$$
$$B'_i = B_i + (b-a)$$

Where (b-a) represents the break down time interval of machines.

Step 5: Determine the optimal sequence by Johnson's algorithm for the modified problem obtained in step 4. This new sequence is either new optimal or near optimal for the original problem. Also determine the total elapsed time.

2.9 n-jobs two machines flow shop with equivalent job for a job block transportation time for a job and break down

This topic gives a heuristic approach to study n-job, 2-machine flow-shop problem in which equivalent job for job-block; transportation time for a job from one machine to another machine and break down machine time was involved. The objective of the chapter is to find optimal or near optimal solution technique according to which processing of sequence gives the minimum or near to minimum total completion times for n jobs in the sequence.

Johnson (1954) basically studied a two machine n-job flow shop problem and he gave an optimal algorithm to find a sequence giving minimum completion time for all jobs when processed through two machines. He, further extended the optimal algorithm for a special type of three machine flow shop problem. After this study, a good deal of efforts has been made of three machine flow shop problem. After this study, a good deal of efforts has been made to find an optimal solution for the general n job m machine flow shop problem. However, various research cal attempts have been made to find near optimal solution for a general type of flow shop problem through heuristic approaches or numerical techniques. To mention a few of the attempts made by the researcher will refer to studies by Mitten, Maggu, G. Das etc. Further mention in the above studies it was assumed that the transportation time of a job after completion on one machine and then going to the successive machine is negligible. Here we have taken into account the concept of transportation time.

PROBLEM:

n jobs are processed through two machines A and B in the order (A, B) with processing times for job i on machines A and B being defined as A_i and B_i respectively. Let t_i be transportation time of job i from machine A to B. Let (a, b) be the break down interval with length I = b-a. Let the jobs be performed in a job block β= (α_1, α_2) where α_1, α_2 are, out of n jobs 1, 2, 3, ……., n.

Then problem is to find an optimal or near optimal sequence to minimize the total elapsed time in completion of all jobs on the two machines.

Step [1]: Find a reduced problem with new processing time A_i' and B_i' defined as

$A_i' = A_i + t_i$

$B_i' = B_i + t_i$

On the lines of Step [1] in Maggu and Das [1980] problem.

Step [2]: Find processing times for the equivalent job β following Maggu and Das [1977] for the problem in Step [2] as follows:

$A'_\beta = A'_\alpha + A'\alpha_{k+1} – \min (\beta'\alpha_k, A'\alpha_{k+1})$

$B'_\beta = B'_\alpha + B'\alpha_{k+1} – \min (B'\alpha_k, A'\alpha_{k+1})$

Step [3]: Now form a reduced problem replacing the given job-block in the theorem (as per Step [1]) by their equivalent job.

Step [4]: Use Johnson's method to obtain the optimal sequence of reduced problem as per steps and read effect of break down interval of (a, b) on different jobs.

Step [5]: Find a reduced problem in Step [3] with processing times A_i'' B_i''

Where

$$\left.\begin{array}{l} A_i'' = A_i' + 1 \\ B_i'' = B_i + 1 \end{array}\right\} \quad \text{if (a, b) has effected on job i}$$

Where

$$\left.\begin{array}{l} A_i'' = A_i' \\ B_i'' = B_i \end{array}\right\} \quad \text{if (a, b) has no effect on job i}$$

Step [6]: Now repeat the procedure to get optimal sequence of reduced problem as Step [5] using step [2], [3], [4] on the similar lines as done for getting the optimal sequence of reduced problem as per Step [1].

Then one of the optimal sequences is optimal or near optimal for the original problem.

Numerical Example:

Let the problem be defined as, job processing times of job

(i)	A (Ai)	B (Bi)	Transportation Times (ti)
1	6	2	3
2	7	8	4
3	11	6	6
4	5	11	2

with equivalent job β= (1, 4) and with break down interval (a, b) defined by (a, b) = (13, 20) with l= 20-13= 7

Problem is to find optimal or near optimal sequence summing the total elapsed time.

Solution: Reduced problem as per step [1] is,

Job (i)	A': $(A_i + t_i)$	B': $(B_i + t_i)$
1	9	5
2	11	12
3	17	12
4	7	13

Processing time for equivalent job β= (1, 4) as per step [2] is

$$
\begin{aligned}
A'\beta &= A'1 + A'4 - \min(B'1, A'4) \\
&= 9 + 7 - \min(5, 7) \\
&= 16 - 5 \\
&= 11 \\
B'\beta &= B'1 + B'4 - \min(B'1, A'4) \\
&= 5 + 13 - \min(5, 7) \\
&= 18 - 5 \\
&= 13
\end{aligned}
$$

Reduced problem as per step [3] is as follows:

Job	A';	B';
B	11	13
2	11	12
3	17	12

Using Johnson's method optimal sequence of problem as per step is either (β, 2, 3) or (2, β, 3) replacing equivalent job β for job-block (1, 4) optimal sequences are (, 4, 2, 3) or (2, 1, 4, 3) now two cases arises as follows:

Case[1]: Effect of break-down interval (13, 20) on sequence (1, 4, 2, 3) is read as follows:

Job	A	ti	B
	in-out		in-out
1	0 – 6	3	9 – 11
4	6 – 11	2	$\boxed{13 – 24}$
2	11 – 18	4	24 – 35
3	18 – 29	6	35 - 41

Reduced problem as per step [5]

Job	A';	B';
1	6	2
2	7	8
3	11	6
4	5	18

$$
\begin{aligned}
A''\beta &= A''1 + A''4 - \min(B''1, A''4) \\
&= 6 + 5 - \min(2, 5) \\
&= 11 - 2 \\
&= 9
\end{aligned}
$$

$$
\begin{aligned}
B''\beta &= B''1 + B''4 - \min(B''1, A''4) \\
&= 2 + 18 - \min(2, 5) \\
&= 20 - 2 \\
&= 18
\end{aligned}
$$

Reduced problem is

Job (i)	A'i	B'i
β	9	18
2	7	8
3	11	6

Using Johnson' technique the optimal sequence is (β, 2, 3) or (2, 1, 4, 3)

Job(i)	A	ti	B
2	0 – 7	4	11 – 19
1	7 – 13	3	19 – 21
4	13 – 18	2	21 – 32
3	18 – 29	6	35 - 41

Reduced problem in this case is as per step [5]

Job (i)	A'i	B'i
1	6	9
2	7	8
3	18	6
4	12	11

$$\begin{aligned} A''\beta &= A''1 + A''4 - \min(B''1, A''4) \\ &= 6 + 12 - \min(9, 12) \\ &= 18 - 9 \\ &= 9 \end{aligned}$$

B”β = B”1 + B”4 – min(B”1, A”4)

= 9 + 11 – min (9, 12)

= 20 – 9

= 11

The further reduced problem is

Job	A’i	B’i
β	9	11
2	7	8
3	18	6

Using Johnson’s technique, the optimal sequence is (2, β, 3) or (2, 1, 4, 3)

The optima or near optimal sequence is either (1, 4, 2, 3) or (2, 1, 4, 3). Since in each case the total minimum elapsed time is 47 hours therefore, each of the two sequences is optimal.

2.10 On Equivalent job for job block in a three machine flow shop problem with break down machine times included

This chapter has the objective to give heuristic algorithm for finding an optimal or near optimal solution minimizing the total elapsed time in nx3 Johnson’s flow shop for block is taken into account when the machines operating the jobs are allowed to got break downs for certain times of intervals. The efficiency of the heuristic algorithm (found on the basis of experimental work) is verified by means of an illustrative example.

Heuristic Algorithm:

The heuristic algorithm scanning optimal or near optimal schedule minimizing the total elapsed time may consist of the following steps:

Step [1]: Find the reduced problem by replacing the block job processing times by their respective equivalent jobs processing times with the aid of Das [1978] or Maggu, Das and Sishpal [1982] methods.

Step [2]: Find the effect of break down interval on the flow time intervals of the different jobs for the reduced problem in step [1].

Step [3]: Find a second reduced problem in which processing times are defined by A_i', B_i', C_i' as

$$\left.\begin{array}{l} A_i'' = A_i' + 1 \\ B_i'' = B_i + 1 \\ C_i'' = C_i + 1 \end{array}\right\} \quad \text{if job i is effected by break down interval (a, b)}$$

$$\left.\begin{array}{l} A_i'' = A_i' \\ B_i'' = B_i \\ C_i'' = C_i \end{array}\right\} \quad \text{if job i is effected by break down interval (a, b)}$$

Step [4]: Find the equivalent job processing time for the given job block in reduced as per Step [3].

Step [5]: Find a third reduced problem replacing the given set of jobs by their equivalent jobs with processing times as calculated in the above Step [4].

Step [6]: Use Johnson's technique to find optimal sequence the problem in Step [5].

Step [7]: One of the optimal sequences obtained in step [6] is now optimal or ner optimal sequence for the original problem.

Illustrative Example:

To illustrate the procedure of the heuristic principle, we now take up one numerical problem in the tableau form as follows:

Jobs	**Machines**		
(i)	A	B	C
	(t_i1)	(t_i2)	(t_i3)
1	15	8	8
2	10	9	7
3	11	10	7
4	14	9	9
5	12	10	6
6	13	7	5
7	11	6	4

The problem is to obtain optimal sequence in which job β is equivalent job for an order pair job block (1, 5) of job 1 and job 5 in this order with breakdown intervals (35, 47) with l=12.

Solution: here,

Min (t_{is}) ≥ max (t_{is+1}) (S= 1, 2, 3)

Are satisfied also here,

$\alpha_k = \alpha_{k=1} = 5$

As per step [1], the processing times of β on machine A, B, C are given by using the formal due to Maggu, Das and Shishpal [1982]

$t_{\beta A} = (t_{1A} = U_1) + (t_{5A} + U_5) - \min(t_{ic} + u_i)\ t_{5A} = U_5)$

$= 15 + 8 + 12 + 10 - \min(8 + 8, 12 + 10)$

$= 23 + 22 - 16$

$= 29$

$t_{\beta\beta} = 0$

$t_{\beta C} = (t_{1C} = U_1) + (t_{5C} + U_5) - \min(t_{ic} + u_i)\ t_{5A} = U_5)$

$= 8 + 8 + 6 + 10 - \min(8 + 8, 12 + 10)$

$= 16$

Hence, by step [2], replacing jobs (1, 5) by β reduces the problem into

Jobs		**Machines**	
(i)	A	B	C
β	29	0	16
2	10	9	7
3	11	10	7
4	14	9	9
6	13	7	5
7	11	6	4

Using Johnson's technique, the optimal sequence is

4	3	2	β	6	7

The flow- time chart for the sequence 4, 3, 2, β, 6, 7 with replacing β by (1, 5) is

Job	**A**	**B**	**C**
(i)	in-out	in-out	in-out
4	0 – 17	14 – 23	23 – 32
3	14 – 25	25 – 35	$\boxed{35 - 42}$
2	25 – 35	$\boxed{35 - 44}$	44 – 51
1	$\boxed{35 - 50}$	50 – 58	58 – 60
5	50 – 62	62 – 72	72 – 78
6	62 – 75	75 – 82	82 – 87
7	75 – 86	86 – 92	92 - 96

Now using Maggu’s technique [1982] after reading the effects of the break down interval (35, 47) we have the reduced problem as per step [3].

Job	**A**	**B**	**C**
(i)	in-out	in-out	in-out
4	14	9	9
3	11	10	19
2	10	21	7
1	27	8	8
5	12	10	6
6	13	7	5
7	11	6	4

$t_{\beta A}$ $= (t_{1A} = u_1) + (t_{5A} + u_5) - \min (t_{1c} + u_1)\ t_{5A} = u_5)$

$= 27 + 8 + 12 + 10 - \min (8 + 8,\ 12 + 10)$

$= 41$

$t_{\beta\beta} = 0$

$t_{\beta C}$ $= 8 + 8 + 12 + 10 - 16$
$= 22$

As per step [5] the reduced problem becomes

Job	**A**	**B**	**C**
(i)	(A_i)	(B_i)	(C_i)
4	14	9	9
3	11	10	19
2	10	21	7
β	41	0	22
7	11	6	4

Using Johnson's technique, optimal sequence is

3	4	2	β	6	7

Or 3, 4, 2, 1, 5, 6, 7 when β is replaced by (1, 5). This optimal sequence (3, 4, 2, 1, 5, 6, 7) is also optimal or near optimal for the original problem. The flow time chart in the sequence 3, 4, 2, 1, 5, 6, 7 job is given below

Job	**A**	**B**	**C**
(i)	in-out	in-out	in-out
3	0 – 11	11 – 21	21 – 28
4	11 – 25	25 – 34	34 – 43
2	25 – 35	35 – 44	44 – 51
1	35 – 50	50 – 58	58 – 66
5	50 – 62	62 – 72	72 – 78
6	62 – 75	75 – 82	82 – 87
7	75 – 86	86 – 92	92 - 96

With minimum total elapsed time is 96 hours.

2.11 Minimizing Rental Cost under specified rental policy in Two Stage Flow Shop

Recently an attempt has been made by T.P. Singh et al. [2006] to study the two machine general flow shop problem following some restrictive renting policy including equivalent job-block criteria. The object of the study is to find an algorithm to minimize the rental cost of the machines under specified renting policy.

2.12 PRACTICAL SITUATIONS

Various practical situations occur in real life when one has got the assignments but does not have one's own machine or does not have enough money or does not want to take risk of investing huge amount of money to purchase machine. Under such circumstances the machine has to be taken on rent in order to complete the assignments. In his starting career, we find a medical practitioner does not buy expensive machines say X-ray machine, the ultra sound machine etc. but instead takes on rent. The examination branch of a board/institute needs machines as data entry machine, computer, printer etc. on rent for computerizing and compiling examination result for secrecy point of view.

Moreover in hospitals industries concern, sometimes the priority of one job over other is preferred. It may be because of urgency or demand of its relative importance. Hence the job block criteria becomes significant.

2.13 NOTATIONS

S: Sequence of job 1, 2, 3, ……., n

M_j : Machine j, j= 1, 2, ……….

A_i: Processing time of i^{th} job on machine A.

B_i: Processing time of i^{th} job on machine B.

A'_i: Expected processing time of i^{th} job on machine A.

B'_i: Expected processing time of i^{th} job on machine B.

p_i: Probability associated to the processing time A_i of i^{th} job on machine A.

q_i: Probability associated to the processing time B_i of i^{th} job on machine B.

β: Equivalent job for job-block.

S_i: Sequence obtained from Johnson's procedure to minimize rental cost.

C_j: Rental cost per unit time of machine j.

U: Utilization time B (2^{nd} machine) for each sequence S_i.

$t_1(S_i)$: Completion time of last job of sequence S_i on machine A.

$t_1(S_i)$: Completion time of last job of sequence S_i on machine A.

$R(S_i)$: Total rental cost for sequence S_i of all machines.

$CT(S_i)$: Completion time of 1^{st} job of each sequence S_i on machine A.

2.14 ASSUMPTIONS

1. We assume rental policy that all the machines are taken on rent as and when they are required and are returned as when they are no longer required for processing. Under this policy second machine is taken on rent at time when first job completes its processing on first machine. Therefore idle time of second machine for first job is zero.

2. Jobs are independent to each other.
3. Machine break down is not considered. This simplifies the problem by ignoring the stochastic component of the problem.
4. Per-emption is not allowed i.e. jobs are not being split, clearly, once a job is started on a machine, the process on that machine can't be stopped unless the job is completed.

2.15 DEFINITIONS

Definition 1

As operation is defined as a specific job on a particular machine.

Definition 2

Sum of idle time of M_2 (for all jobs)

$$\sum_{i=1}^{n} Ii2 = \max\left[\sum_{i=1}^{n} A'_i - \sum_{i=1}^{n-1} B'_i, \sum_{i=1}^{n-1} A'_i - \sum_{i=1}^{n-2} B'_i \sum_{i=1}^{n-2} A'_i - \sum_{i=1}^{n-3} B'_i \ldots\ldots\ldots \sum_{i=1}^{2} A'_i - \sum_{i=1}^{2-1} B'_i A'_i\right]$$

$= \max [P_n, P_{n-1} \ldots\ldots\ldots\ldots\ldots\ldots\ldots P_2, P_1]$

$= \max [P_k]$

where

$$P_k = \sum_{i=1}^{k} A'_i - \sum_{i=1}^{k-1} B'_i$$

$1 \le k \le n$ $\qquad$ $i=1$

and $A'_i = A_i \times p_i$

$B'_i = B_i \times q_i$

Definitions

Total elapsed time for a given sequence.

= Sum of expected processing time on 2nd machine (M_2)+ total idle time on M_2.

$$= \sum_{i=1}^{n} B'_i + \sum_{i=1}^{n} I'_{i2}$$

$$= \sum_{i=1}^{n} B'_i + \max [P_k], \text{ where } P_k = \sum_{i=1}^{n} A'_i - \sum_{i=1}^{n} B'_i$$

Note 1:

Idle time of 1st machine is always zero i.e. $\sum_{i=1}^{n} I'_{i1} = 0$

Note 2:

Idle time of 1st job on 2nd machine i.e. I^{i2} = Expected processing time of 1st job on 1st machine.

= A'_1

Note 3:

Rental cost of machines will be minimum if idle time of machine 2 is minimum.

2.16 Now we state two theorems which are applied in our algorithm

Theorem 1:

Equivalent job block theorem due to Maggu & Das [1977]. In two machine flow shop problem "In processing a schedule S = (α_1, α_2, ... α_{k-1}, ... α_n) of n jobs on two machines A and B in the order AB with no passing allowed the job block (α_k, α_m) having processing time $A\alpha_k$, $B\alpha_k$, $A\alpha_m$, $B\alpha_m$ is equivalent to the single job β (called equivalent job β). The processing times of equivalent job β on the machines. A & B denote respectively by A_β and B_β are given by.

$A_\beta = A_{\alpha k} + A_{\alpha m} - \min(B_{\alpha k}, A_{\alpha m})$

$B_\beta = B_{\alpha k} + B_{\alpha m} - \min(B_{\alpha k}, A_{\alpha m})$

Theorem 2:

Job i precedes to job j in optimal ordering having idle time on B. If $\min(A'_i, B'_j) < \min(A'_j, B'_i)$.

Where

A'_i = Expected processing time of i^{th} job on A = $A_i \times p_i$

B'_i = Expected processing time of i^{th} job on B = $B_i \times q_i$

Algorithm

Step 1: Define expected processing time of job block β = (k, m) on machine A & B using equivalent job block Criteria given by Maggu & Das [6] i.e. find A'_β & B'_β as:

$A'_\beta = A'_k + A'_m - \min(B'_k, A'_m)$

$B'_\beta = B'_k + B'_m - \min(B'_k, A'_m)$

Step 2: Using Johnson's two machine algorithm [5] obtain the sequence S_i, which minimize the total elapsed time.

Step 3: Observe the processing time of 1^{st} job of S_1 on the first machine A: let it be α

Step 4: Obtain all the jobs having processing time on A greater than α. Put these jobs one by one in the 1^{st} position of the sequence S_1, keeping other jobs of the sequence S_1 in the same order. Let these sequence S_2, S_3, S_4, …………., S_n.

Step 5: Prepare in-out table for each sequence S_1 (i= 1, 2, ...r) and evaluate total completion time of last job of each sequence machine $t_1(S_i)$ & $t_2(S_i)$ on machine A & B respectively.

Step 6: Evaluate completion time CT (S_i) of 1st job of each sequence S_i on machine A.

Step 7: Calculate utilization time U_i of 2nd machine for each sequence S_i as

$U_i = t_2 (S_i) – CT (S_i)$ for i= 1, 2, 3,........

Step 8: Find Min $\{U_i\}$, i= 1, ...r. Let it be corresponding to i=m. then S is the optimal sequence for minimum rental cost.

Min Rental Cost = $t_1 (S_m) \times C_i + U_m \times C_2$

Where C_1 & C_2 are the rental cost per unit time of 1st & 2nd machine respectively.

Numerical Illustration

Consider 5 jobs and 2 machines problem to minimize the rental cost. The processing times are given as follows:

Jobs	Machine A (A_i)	Machine B (B_i)
1	11	7
2	15	11
3	12	13
4	17	16
5	14	17

Rental costs per unit time for machine A & B are 15 and 13 units respectively, and jobs 2, 5 are to be processed as an equivalent group job β.

Solution: Jobs by alongwith the processing times of equivalent job block β = (2, 5) are given by

A'_β = 15 + 14 – 11

= 18

B'_β = 11 + 17 – 11

= 17

Jobs	A_i'	B_i'
1	11	7
β	18	17
3	12	13
4	17	10

As per step II using Johnson's method, optimal sequence is

S_1 = 3, β, 4, 1

i.e. 3, 2, 5, 4, 1

S_2 = 1, 4, 3, 2, 5

S_3 = 2, 5, 4, 3, 1

S_4 = 2, 5, 4, 1, 3

These sequences are enumerated in following table

S_1 = 3, 2, 5, 4, 1

Jobs	A	B
	In-Out	In-Out
3	0 – 12	12 – 25
2	12 – 27	27 – 38
5	27 – 41	41 – 58
4	41 – 58	58 – 74
1	58 – 69	74 - 81

Then the total elapsed time = 81 units and utilization time for B = 81 – 12 = 69 units

S_2 = 1, 4, 3, 2, 5

Jobs	A	B
	In-Out	In-Out
3	0 – 11	11 – 18
4	11 – 28	28 – 44
3	28 – 40	44 – 57
2	40 – 55	57 – 68
5	55 – 69	69 - 86

Total elapsed time = 86 units utilization of B = 86 – 11 = 75 units

S_3 = 2, 5, 4, 3, 13

Jobs	A	B
	In-Out	In-Out
2	0 – 15	15 – 26
5	15 – 29	29 – 46
4	29 – 46	46 – 62
3	46 – 58	62 – 75
1	58 – 69	75 - 82

Total elapsed time=82 units utilization time of B=82–15=67 units

S_4 = 2, 5, 4, 3, 13

Jobs	A	B
	In-Out	In-Out
2	0 – 15	15 – 26
5	15 – 29	29 – 46
4	29 – 46	46 – 62
3	46 – 57	62 – 69
1	57 – 69	69 - 82

Total elapsed time=82 units utilization time of B=82–15=67 units

The total utilization of A machine is fixed 69 units and minimum utilization time of B machine is 67 units for two sequence S_3 and S_4. Therefore optimal sequence are S_3 2-5-4-3-1 and S_4 2-5-4-1-3 and total rental cost = 15 x 69 + 13 x 67

= 1035 + 871

= 1906 units

Conclusion

The study may further by extended if parameter like set up time, transportation time etc. are taken into consideration.