Chapter

15 Network Scheduling by PERT/CPM

15.1 INTRODUCTION

Network scheduling is a technique used for planning and scheduling large projects, in the fields of construction, maintenance, fabrication and purchasing of computer systems, etc. It is a method of minimizing the trouble spots such as production, delays and interruptions, by determining critical factors and co-ordinating various parts of the overall job.

There are two basic planning and control techniques that utilize a network to complete a predetermined project or schedule. These are Programme Evaluation Review Technique (PERT) and Critical Path Method (CPM).

A project is defined as a combination of interrelated activities, all of which must be executed in a certain order for its completion.

The work involved in a project can be divided into three phases, corresponding to the management functions of planning, scheduling and controlling.

Planning This phase involves setting the objectives of the project as well as the assumptions to be made. It also involves the listing of tasks or jobs that must be performed in order to complete a project under consideration. In this phase, in addition to the estimates of costs and duration of the various activities, the manpower, machines and materials required for the project are also determined.

Scheduling This consists of laying the activities according to their order of precedence and determining the following:

(*i*) The start and finish times for each activity
(*ii*) The critical path on which the activities require special attention.
(*iii*) The slack and float for the non-critical paths.

Controlling This phase is exercised after the planning and scheduling. It involves the following:

(*i*) Making periodical progress reports
(*ii*) Reviewing the progress
(*iii*) Analyzing the status of the project
(*iv*) Making management decisions regarding updating, crashing and resource allocation, etc.

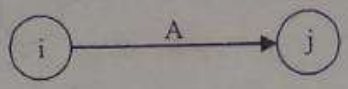
15.2 BASIC TERMS

To understand the network techniques, one should be familiar with a few basic terms of which both CPM and PERT are special applications.

Network It is the graphic representation of logically and sequentially connected arrows and nodes, representing activities and events in a project. Networks are also called arrow diagrams.

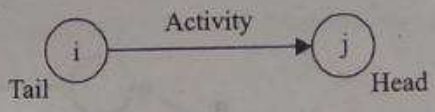
Activity An activity represents some action and is a time consuming effort necessary to complete a particular part of the overall project. Thus, each and every activity has a point of time where it begins and a point where it ends.

It is represented in the network by an arrow,

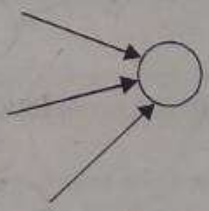


Here *A* is called the activity.

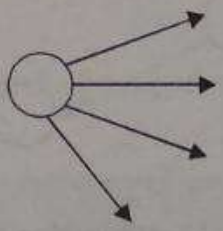
Event The beginning and end points of an activity are called events or nodes. Event is a point in time and does not consume any resources. It is represented by a numbered circle. The head event called the *j*th event always has a number higher than the tail event, which is also called the *i*th event.



Merge and burst events It is not necessary for an event to be the ending event of only one activity as it can be the ending event of two or more activities. Such an event is defined as a merge event.



If the event happens to be the beginning event of two or more activities, it is defined as a burst event



Preceding, succeeding and concurrent activities Activities that must be accomplished before given event can occur, are termed as *preceding activities.*

Activities that cannot be accomplished until an event has occurred, are termed as *succeeding activitie*

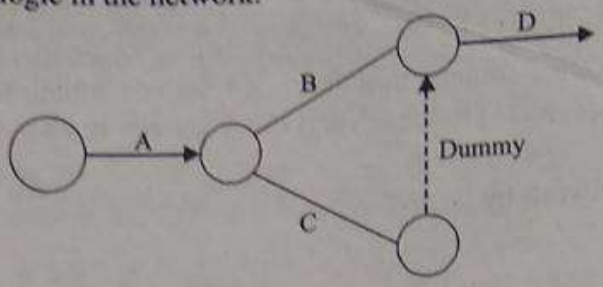
Activities that can be accomplished concurrently, are known as *concurrent activities.*

This classification is relative, which means that one activity can be preceding to a certain event, and th same activity can be succeeding to some other event or it may be a concurrent activity with one or mo activities.

Dummy activity Certain activities, which neither consume time nor resources but are used simply represent a connection or a link between the events are known as dummies. It is shown in the network a dotted line. The purpose of introducing dummy activity is:

(*i*) To maintain uniqueness in the numbering system, as every activity may have a distinct set of eve by which the activity can be identified.

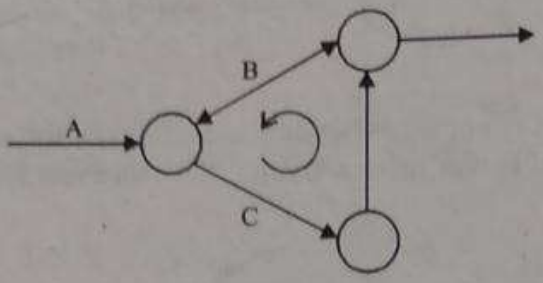
(*ii*) To maintain a proper logic in the network.



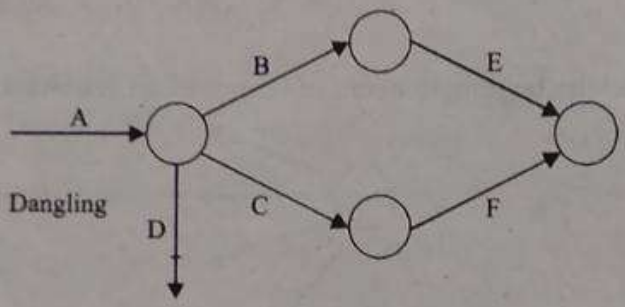
15.3 COMMON ERRORS

Following are the three common errors in a network construction:

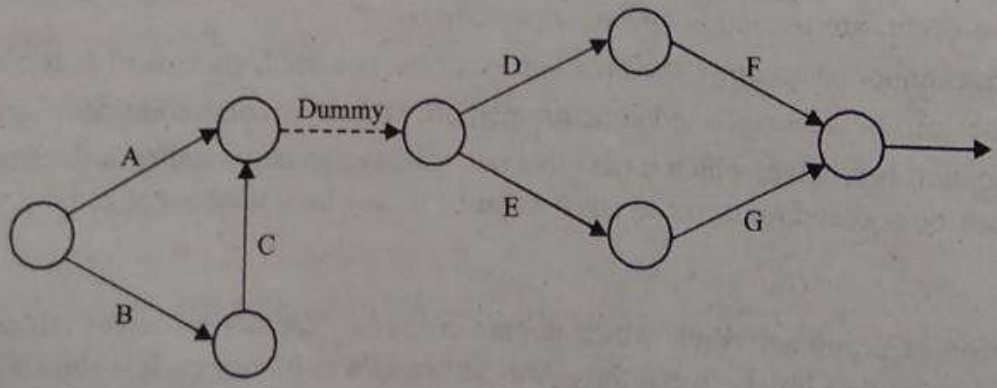
Looping (cycling) In a network diagram, a looping error is also known as cycling error. Drawing an endless loop in a network is known as error of looping. A loop can be formed if an activity is represented as going back in time.



Dangling To disconnect an activity before the completion of all the activities in a network diagram, is known as dangling.



Redundancy If a dummy activity is the only activity emanating from an event and can be eliminated, it is known as redundancy.



15.4 RULES OF NETWORK CONSTRUCTION

There are a number of rules in connection with the handling of events and activities of a project network hat should be followed.

(*i*) Try to avoid arrows that cross each other.
(*ii*) Use straight arrows.
(*iii*) No event can occur until every activity preceding it has been completed.
(*iv*) An event cannot occur twice, i.e., there must be no loops.
(*v*) An activity succeeding an event cannot be started until that event has occurred.
(*vi*) Use arrows from left to right. Avoid mixing two directions, vertical and standing arrows may be used if necessary.
(*vii*) Dummies should be introduced only if it is extremely necessary.
(*viii*) The network has only one entry point called the start event and one point of emergence called the end or terminal event.

15.5 NUMBERING THE EVENTS (FULKERSON'S RULE)

After the network is drawn in a logical sequence, every event is assigned a number. The number sequence must be such so as to reflect the flow of the network. In numbering the events, the following rules should be observed.

(*i*) Event numbers should be unique.
(*ii*) Event numbering should be carried out on a sequential basis, from left to right.
(*iii*) The initial event, which has all outgoing arrows with no incoming arrow is numbered as 1.
(*iv*) Delete all arrows emerging from all the numbered events. This will create at least one new start event, out of the preceding events.
(*v*) Number all new start events 2, 3 and so on. Repeat this process until the terminal event without any successor activity is reached. Number the terminal node suitably.

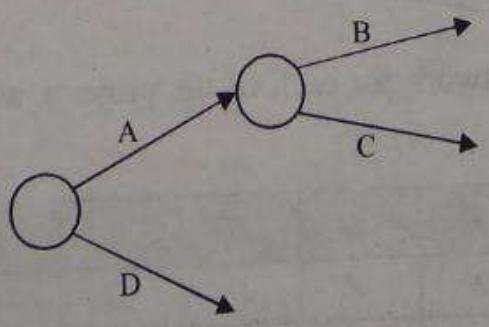
Note: The head of an arrow should always bear a number higher than the one assigned to the tail of the arrow.

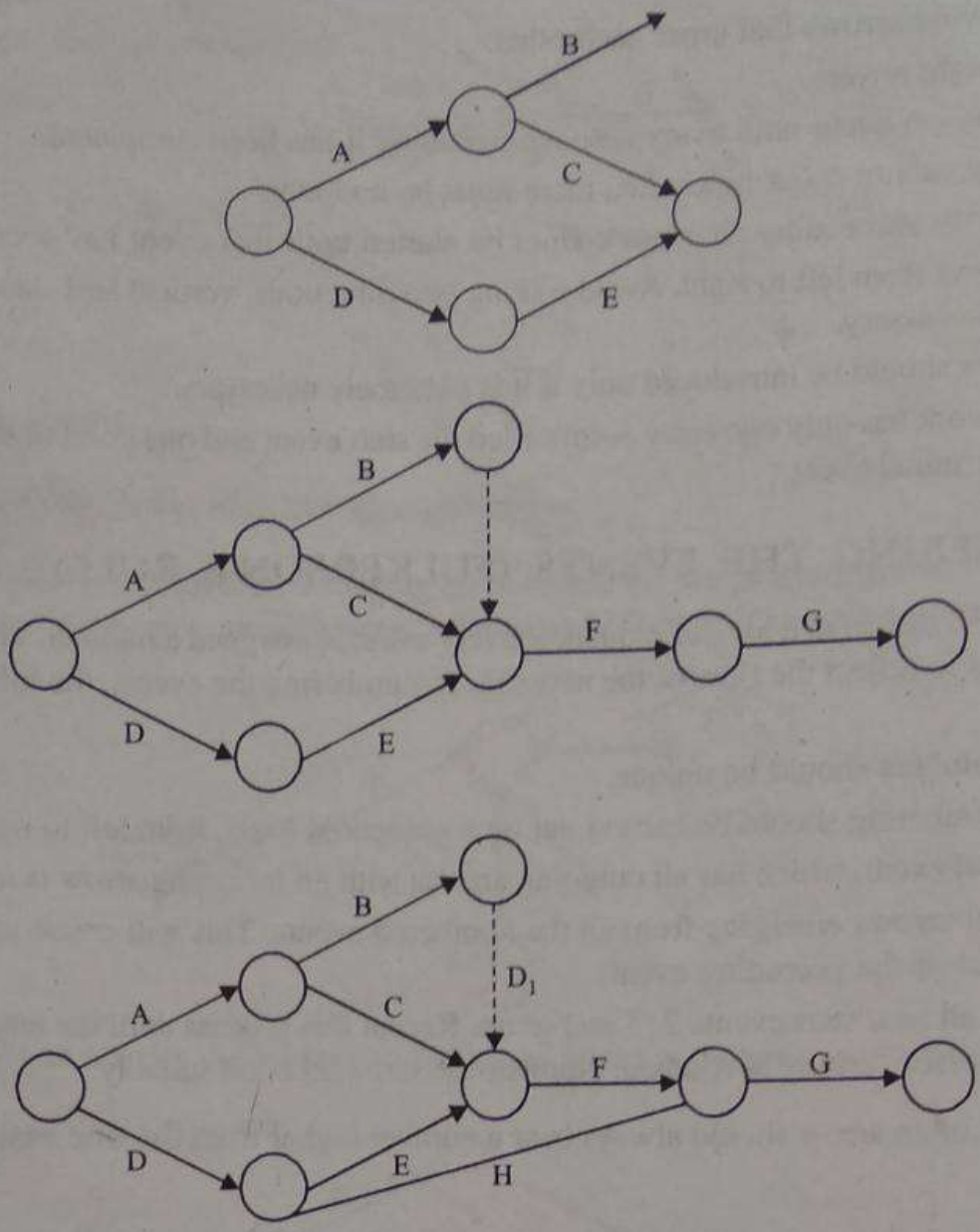
15.5.1 Construction of Network

Example 15.1 Construct a network for the project whose activities and precedence relationships are as given below:

Activities	*A*	*B*	*C*	*D*	*E*	*F*	*G*	*H*	*I*
Immediate Predecessor	–	*A*	*A*	–	*D*	*B, C, E*	*F*	*D*	*G, H*

Solution From the given constraints, it is clear that *A* and *D* are the starting activities and *I* the terminal activity. *B* and *C* are starting with the same event and are both the predecessors of the activity *F*. Also, *E* has to be the predecessor of both *F* and *H*. Hence, we have to introduce a dummy activity.





D_1 is the dummy activity.

Finally, we have the following network.