5.2 Critical path method (CPM) and PERT network modelling and time analysis

- The most widely used scheduling technique is the critical path method (CPM) for scheduling, often referred to as *critical path scheduling*. This method calculates the minimum completion time for a project along with the possible start and finish times for the project activities. Indeed, many texts and managers regard critical path scheduling as the only usable and practical scheduling procedure. Computer programs and algorithms for critical path scheduling are widely available and can efficiently handle projects with thousands of activities.

- The *critical path* itself represents the set or sequence of predecessor/successor activities which will take the longest time to complete. The duration of the critical path is the sum of the activities' durations along the path. Thus, the critical path can be defined as the longest possible path through the "network" of project activities. The duration of the critical path represents the minimum time required to complete a project. Any delays along the critical path would imply that additional time would be required to complete the project.

- There may be more than one critical path among all the project activities, so completion of the entire project could be delayed by delaying activities along any one of the critical paths. For example, a project consisting of two activities performed in parallel that each require three days would have each activity critical for a completion in three days.

- Formally, critical path scheduling assumes that a project has been divided into activities of fixed duration and well defined predecessor relationships. A predecessor relationship implies that one activity must come before another in the schedule. No resource constraints other than those implied by precedence relationships are recognized in the simplest form of critical path scheduling.

- To use critical path scheduling in practice, construction planners often represent a resource constraint by a precedence relation. A constraint is simply a restriction on the options available to a manager, and a resource constraint is a constraint deriving from the limited availability of some resource of equipment, material, space or labor.

- For example, one of two activities requiring the same piece of equipment might be arbitrarily assumed to precede the other activity. This artificial precedence constraint insures that the two activities requiring the same resource will not be scheduled at the same time. Also, most critical path scheduling algorithms impose restrictions on the generality of the activity relationships or network geometries which are used

- The actual computer representation of the project schedule generally consists of a list of activities along with their associated durations, required resources and predecessor activities. Graphical network representations rather than a list are helpful for visualization of the plan and to insure that mathematical requirements are met.

- The actual input of the data to a computer program may be accomplished by filling in blanks on a screen menu, reading an existing data file, or typing data directly to the program with identifiers for the type of information being provided.

In PERT activities are shown as a network of precedence relationships using activity-on arrow network construction – Multiple time estimates – Probabilistic activity times. Used in Project management - for non-repetitive jobs (research and development work), where the time and cost estimates tend to be quite uncertain. This technique uses probabilistic time estimates.

Benefits of PERT/CPM

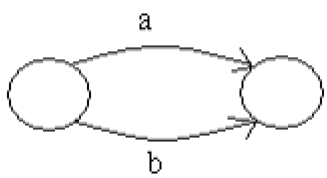
- Useful at many stages of project management
- Mathematically simple
- Give critical path and slack time
- Provide project documentation
- Useful in monitoring costs

Limitations of PERT/CPM

- Clearly defined, independent and stable activities
- Specified precedence relationships
- Over emphasis on critical paths

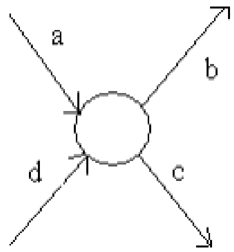
Rules for Drawing Network Diagram

Rule 1 Each activity is represented by one and only one arrow in the network



Rule 2

No two activities can be identified by the same end events



Rule 3

In order to ensure the correct precedence relationship in the arrow diagram, following questions must be checked whenever any activity is added to the network

- What activity must be completed immediately before this activity can start?
- What activities must follow this activity?
- What activities must occur simultaneously with this activity?

In case of large network, it is essential that certain good habits be practiced to draw an easy to follow network

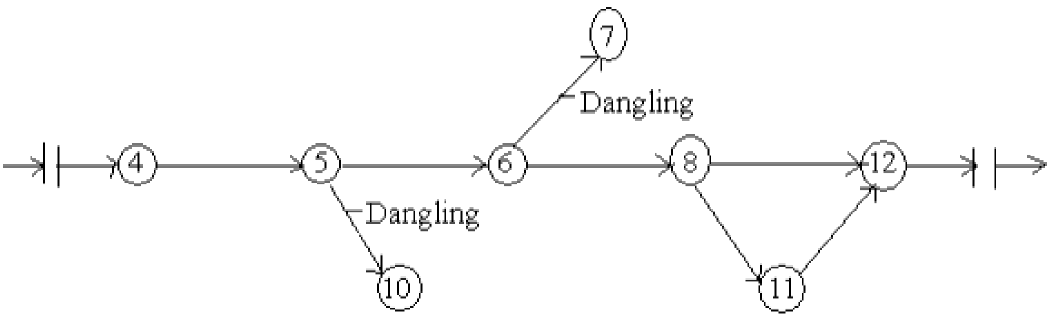
- Try to avoid arrows which cross each other
- Use straight arrows
- Do not attempt to represent duration of activity by its arrow length
- Use arrows from left to right. Avoid mixing two directions, vertical and standing arrows may be used if necessary.
- Use dummies freely in rough draft but final network should not have any redundant dummies.
- The network has only one entry point called start event and one point of emergence called the end event.

Common Errors in Drawing Networks

The three types of errors are most commonly observed in drawing network diagrams

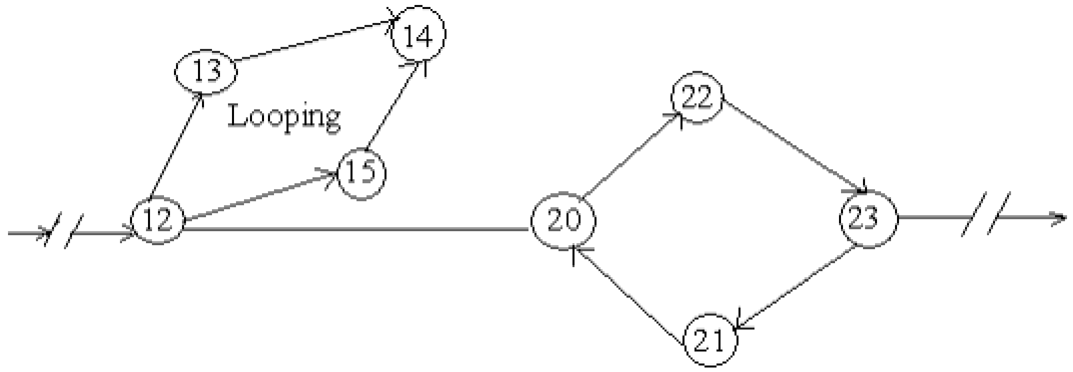
1. **Dangling**

To disconnect an activity before the completion of all activities in a network diagram is known as dangling. As shown in the figure activities (5 – 10) and (6 – 7) are not the last activities in the network. So the diagram is wrong and indicates the error of dangling



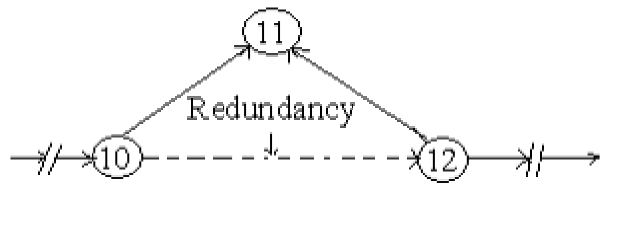
2. Looping or Cycling

Looping error is also known as cycling error in a network diagram. Drawing an endless loop in a network is known as error of looping as shown in the following figure.



3. Redundancy

Unnecessarily inserting the dummy activity in network logic is known as the error of redundancy as shown in the following diagram



Advantages and Disadvantages

PERT/CPM has the following advantages

- A PERT/CPM chart explicitly defines and makes visible dependencies (precedence relationships) between the elements,
- PERT/CPM facilitates identification of the critical path and makes this visible,
- PERT/CPM facilitates identification of early start, late start, and slack for each activity,
- PERT/CPM provides for potentially reduced project duration due to better understanding of dependencies leading to improved overlapping of activities and tasks where feasible.

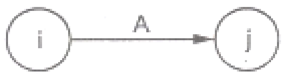
PERT/CPM has the following disadvantages:

- There can be potentially hundreds or thousands of activities and individual dependency relationships,
- The network charts tend to be large and unwieldy requiring several pages to print and requiring special size paper,
- The lack of a timeframe on most PERT/CPM charts makes it harder to show status although colours can help (e.g., specific colour for completed nodes),
- When the PERT/CPM charts become unwieldy, they are no longer used to manage the project.

CRITICAL PATH METHOD IN NETWORK ANALYSIS

Critical Path Method (CPM) was developed in the late 1950s as a method to resolve the issue of increased costs due to inefficient scheduling. Since then, CPM has become popular for planning projects and prioritizing tasks. It helps to break down complex projects into individual tasks and gain a better understanding of the project's flexibility. The Key Concept used by CPM is that a small set of activities, which make up the longest path through the activity network control the entire project. Such activity is called as critical activity. A critical path in project management is the longest sequence of activities that must be finished on time in order for the entire project to be complete. Any delays in critical tasks will delay the rest of the project. Non-critical activities can be re-planned, rescheduled and resources for them can be reallocated flexibly, without affecting the whole project. CPM revolves around discovering the most important tasks in the project timeline, identifying task dependencies, and calculating task durations. CPM has single time estimate which is assumed to be deterministic.

Basic Scheduling Computations in CPM



The basic notations used in CPM can be explained as follows:

For the given example,

(i,j) = Activity "A" with tail event "i" and head event "j"

E_i = Earliest occurrence time of event i

E_j = Latest al owable occurrence time of event j

D_{ij} = Estimated completion time of activity (i, j)

$(E_S)_U$ = Earliest starting time of activity (i, j)

$(E_F)_{ij}$ = Earliest finishing time of activity (i, j)

$(L_S)_{ij}$ = Latest starting time of activity (i, j)

$(L_f)_{ij}$ = Latest finishing time of activity (i, j)

(i) Determination of Earliest time (E_j): Forward Pass computation

Step 1

The computation begins from the start node and move towards the end node. For easiness, the forward pass computation starts by assuming the earliest occurrence time of zero for the initial project event.

Step 2

Earliest starting time of activity (i, j) is the earliest event time of the tail end event

i.e. $\mathbf{(E_s)_{ij} = E_i}$

Earliest finish time of activity (i, j) is the earliest starting time + the activity time

i.e. $\mathbf{(E_f)_{ij} = (E_s)_{ij} + D_{ij}}$ **or** $\mathbf{(E_f)_{ij} = E_i + D_{ij}}$

Earliest event time for event j is the maximum of the earliest finish times of all activities ending in to that event

i.e. $\mathbf{E_j = max\ [(E_f)_{ij}}$ **for all immediate predecessor of (i, j)] or**

$\mathbf{E_j = max [E_i + D_{ij}]}$

(ii) Backward Pass computation (for latest allowable time)

Step 1

For ending event assume E = L.

Also all E's have been computed by forward pass computations.

Step 2

Latest finish time for activity (i, j) is equal to the latest event time of event j.

i.e. $\mathbf{(L_f)_{ij} = L_j}$

Step 3

Latest starting time of activity (i, j) = the latest completion time of (i, j) - the activity time bollo ons omisioak

i.e. $\mathbf{(L_s)_{ij} = (L_f)_{ij} - D_{ij}}$ **or** $\mathbf{(L_s)_{ij} = L_j - D_{ij}}$

Step 4

Latest event time for event " is the minimum of the latest start time of all activities originating from that event

i.e**.** $\mathbf{L_i = min [(L_s)_{ij}}$ **for all immediate successor of (i, j)] min** $\mathbf{[L_f)_{ij} - D_{ij}] = min [L_j – D_{ij}]}$

(iii) Determination of floats and slack times

There are three kinds of floats as follows:

❖ **Total float -** The amount of time by which the completion of an activity could be delayed beyond the earliest expected completion time without affecting the overall project duration time.

$(T_f)_{ij}$ Latest start - Earliest start) for activity (i –j)

i.e. = $\mathbf{(T_f)_{ij} = (L_s)_{ij} - (E_s)_{ij}}$ **or** $\mathbf{(T_f)_{ij} = (T_L - T_E) - t_{ij}}$

❖ **Free float** The time by which the completion of an activity can be delayed beyond the earliest finish time without affecting the earliest start of a subsequent activity.

$(F_f)_{ij}$ = Total float - Head event slack

i.e. $\mathbf{(F_f) = (E_j – E_i) - t_{ij}}$

❖ **Independent float -** The amount of time by which the start of an activity can be delayed without effecting the earliest start time of any immediately following activities, assuming that the preceding activity has finished at its latest finish time. The negative independent float is always taken as zero.

$(I_f)_{ij}$ = Free float - Tail event slack

i.e. $(I_f)_{ij} = (E_j - L_i) - t_{ij}$

❖ **Event slack -** It is defined as the difference between the latest event and earliest event times.

Head event slack = $L_j - E_j$

Tail event slack = $L_i - E_i$

(iv) Determination of critical path

❖ **Critical event -** The events with zero slack times are called critical events. In other words the event i is said to be critical if $E_i = L_i$

❖ **Critical activity -** The activities with zero total float are known as critical activities. In other words an activity is said to be critical if a delay in its start will cause a further delay in the completion date of the entire project.

❖ **Critical path -** The sequence of critical activities in a network is called critical path. The critical path is the longest path in the network from the starting event to ending event and defines the minimum time required to complete the project.