

August 2001

Programme Evaluation and Review Techniques (PERT) And Critical Path Method (CPM) By K.K. Bandyopadhyay



Introduction

Programme Evaluation and Review Technique (PERT) and Critical Path Method (CPM) are techniques of project management. These techniques are useful for basic managerial functions of *planning, scheduling and control*.

The *planning* phase of any project involves a listing of tasks or jobs that must be performed to complete the project. The requirements for materials, human resources, costs and duration of various jobs are also determined in this phase.

Scheduling, on the other hand, is the laying out of the actual jobs of the project in the time order in which they have to be performed.

Control, generally regarded as reviewing the difference between the schedule and actual performance once the project has begun. The analysis of and correction of this difference forms the basic aspects of controls.

The historical development of PERT and CPM gives some idea of their purpose and nature and helps to explain both their similarities and differences. PERT was developed in the United States during the 1950s essentially as a means to plan and accelerate development of ballistic missile development project. As for example, what research had to be done to accomplish this, and how should it be planned? How long would the research take? PERT was the planning and scheduling techniques to answer these questions. PERT incorporated these uncertainties into a model, which provides a reasonable framework for answering such queries. It has proved to be a useful tool in planning and scheduling projects, which consists of numerous activities whose completion times are uncertain and which, although they must be completed in some order, are independent of one another. Fundamental to PERT is the concept of an “event” or the reaching of a certain stage of completion of a project. Also basic is the expected time required to complete activities leading up to that event.

Closely akin to PERT, but developed independently, is the technique known as the Critical Path Method (CPM). It is basically concerned with obtaining the trade-off between cost and completion date for projects. CPM emphasises the relationship between applying more human or other resources to shorten the duration of given jobs in a project and the increased costs of these additional resources. With CPM the amount of time needed to complete various facets of the project is assumed to be known with certainty; moreover, the relation between the amount of resources employed and the time needed to complete the project is also assumed known. Thus CPM is not concerned with uncertain job times as in PERT, rather it deals with time-cost trade-offs. Because of these differences, PERT is used more in research and development projects and CPM is used more in projects where there have been some experiences in handling similar endeavors.

Some of the characteristics of project that are essential for analysis by PERT and CPM.

1. The project consists of well-defined collection of jobs, or activities, which when completed mark the end of the project.
2. The jobs may be started and stopped independently of each other, within a given sequence (This requirement eliminates continuous flow processes, where jobs or operations necessarily follow one after another with essentially no time gap).
3. The jobs are ordered: that is, they must be performed in technological sequence.

Programme Evaluation and Review Technique (PERT)

Understanding Terminologies

In this section we will develop the terminology necessary for a thorough understanding of CPM and PERT. The prerequisites for using PERT and CPM techniques include two major steps:

The first step is separating the project to be scheduled into independent jobs or activities and determining an order of precedence for these jobs. That is, we must see which jobs have to be completed before others can be started.

The next step is drawing a picture, or graph, which portrays each of these jobs and the predecessor and successor relations among them.

Let us begin our discussion with a simplified example of the budgeting process of an organisation called DEEP, which is involved in income generation, programme through self-help groups. The SHGs make biscuits and market them through DEEP. The organisation has marketing unit headed by a marketing manager and a production unit headed by a production manager. A number of staff works under each unit. Suppose that the head of the organisation wants here next year's operating budget prepared as soon as possible.

To accomplish this project the sales staff of the organisation must provide unit sales estimates (number of biscuits to be sold) for the period to both the sales manager and the production manager. The sales staff undertakes a survey of competitive pricing simultaneously with the forecasting. Then the sales manager must estimate market prices on the forecast and give these to the financial officer. The production manager must make "dice" schedules of the units (number of biscuits to be made) to be produced and assign "dices" for their production. Next he must give these schedules and dice assignments to the accounting manager who must then provide costs of production to the financial officer. Then the financial officer, using the information provided by the sales and accounting units and making the necessary arrangements for internal financing, prepares the final budget and submit it to the head of the organization. If we assume that there are no revisions necessary, organisation's budget is basically finished when it is handed to the head of organisation.

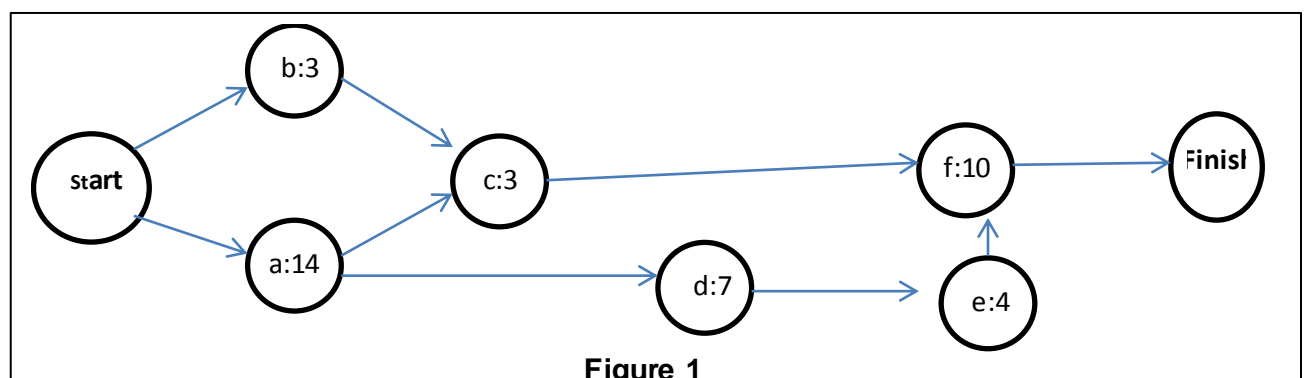
Let us look for closely at how the budget was composed. Before anything else could be done, sales estimates had to be made and simultaneously survey of competitive

pricing had to be done. Only then could sales be priced and trial production schedules prepared. Forecasting the sales, surveying competitive pricing, pricing the own sales, making the production schedules, costing the production and preparing the budget are jobs or activities in the project of making the organisation's budget. Moreover, since the sales forecast had to be done before their own pricing and scheduling, we can say that it was an immediate predecessor of these activities. However, since, survey of competitive pricing was not going to effect the production schedules, we can say it was the immediate predecessor of the pricing sales only. Similarly, pricing was an immediate predecessor of the financial officer's preparing the budget, as was the accounting group's supplying him with cost estimates; and more generally, the forecast was a predecessor of all the subsequent activities. In a like manner preparing the budget is a successor of all other activities, and the immediate successor of pricing.

Table 1
Budgeting Project : DEEP

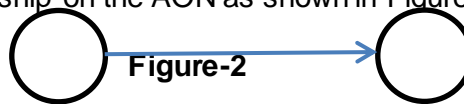
Job identification	Job description	Immediate Predecessors	Time Perform to Job (Days)
a	Forecasting unit sales	-	14
b	Surveying competitive pricing	-	3
c	Pricing sales	a, b	3
d	Preparing production schedules	a	7
e	Costing the production	d	4
f	Preparing the budget	c, e	10

Table 1 shows the project of budgeting at DEEP has been broken into jobs in column 2. Note that we have estimated the time needed for each job (column 4) and assigned identification to each job (column 1), as for example a, b, c and so on. The immediate predecessor relationship of a given activity is shown in column 3



AON Graph: The Budgeting Project of DEEP

The project of budgeting can be portrayed in a project graph or network as shown in Figure 1. This called activity-on-node diagram (AON). The AON graph is constructed so that circles or nodes denote the jobs, and an arrow connecting the two nodes shows the immediate predecessor relationship between two jobs. The arrow's point is at the successor's node. Thus, if job is an immediate predecessor of job b, we portray this relationship on the AON as shown in Figure 2.



The identifications or names for a job and the amount of time required to complete a job is usually inserted in the job node. Also, AON graphs usually contain fictitious nodes labeled “start” and “finish” so that there will be a unique beginning and end to the graph of the project.

Expected Times for Activities

PERT calculates the expected value of an activity duration as a weighted average of the three time estimates. Specifically, it makes the assumption that the optimistic and pessimistic activity times, t_o and t_p , are about equally likely to occur. It also assumes that the most probable activity time, t_m , is four times more likely to occur than either of the other two. So if we apply these weights to the three time estimates we come up with a formula for the average, or expected, time t_e , of an activity:

$$t_e = \frac{t_o + 4t_m + t_p}{6}$$

Critical Path Method (CPM)

Once we reduce the project to a network of activities and we estimate activity duration, we are in a position to determine the minimum time required for completion of the whole project. To do so we must find the *longest path* or sequence of connected activities, through the network. This is called the *critical path* of the network, and its length determines the duration of the project. Let us first clearly define what we mean by a *path* in a network and by the *length* of a path.

Suppose that Mr. X and Mr. Y, both are going to Bangalore from Delhi. Mr. X Lives in Pune and decided to go through that city, spend four hours at home. Mr. Y has a four hour meeting with an organisation in Mumbai and must go there on the way to Bangalore. The time to reach Bangalore from Delhi via Mumbai is 27 hours and via Pune is 29 hours. As both have to work together they want to meet in Bangalore. Their problem is then to decide earliest possible time for their meeting if they both leave at 8.00 a.m. in the morning.

Let us assume that Mr. X and Mr. Y's travel to Bangalore is a project, which will finish at their meeting in Bangalore.

Figure 3
Routes from Delhi to Bangalore

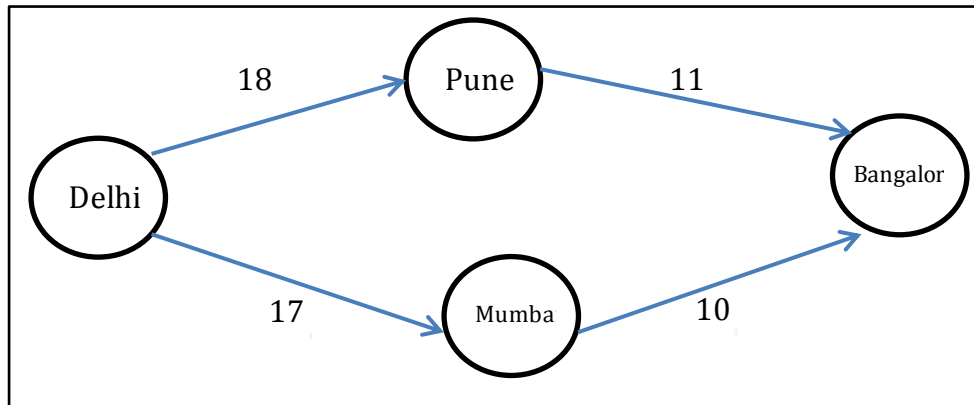


Table 2
Project of Travelling to Bangalore

Job Name	Job Description	Immediate Predecessor	Time (hours)
a	X travels from Delhi to Pune	-	18
b	X spends time at home	A	4
c	X travels from Pune to Bangalore	B	11
d	Y travels from Delhi to Mumbai	-	17
e	Y meet the organisation	D	4
f	Y travels from Mumbai to Bangalore	E	10

To determine the earliest time that Mr. X and Mr. Y can meet, we must find out which one will take longer to reach Bangalore.

The Figure 3 shows that X will take 33 hours (18 from Delhi to Pune, 4 hours at home in Pune and 11 hours from Pune to Bangalore), while Y will take 31 hours to reach Bangalore. Thus if both leave Delhi at 6.00 am in the morning, the earliest time they can meet is 3.00 p.m. next day.

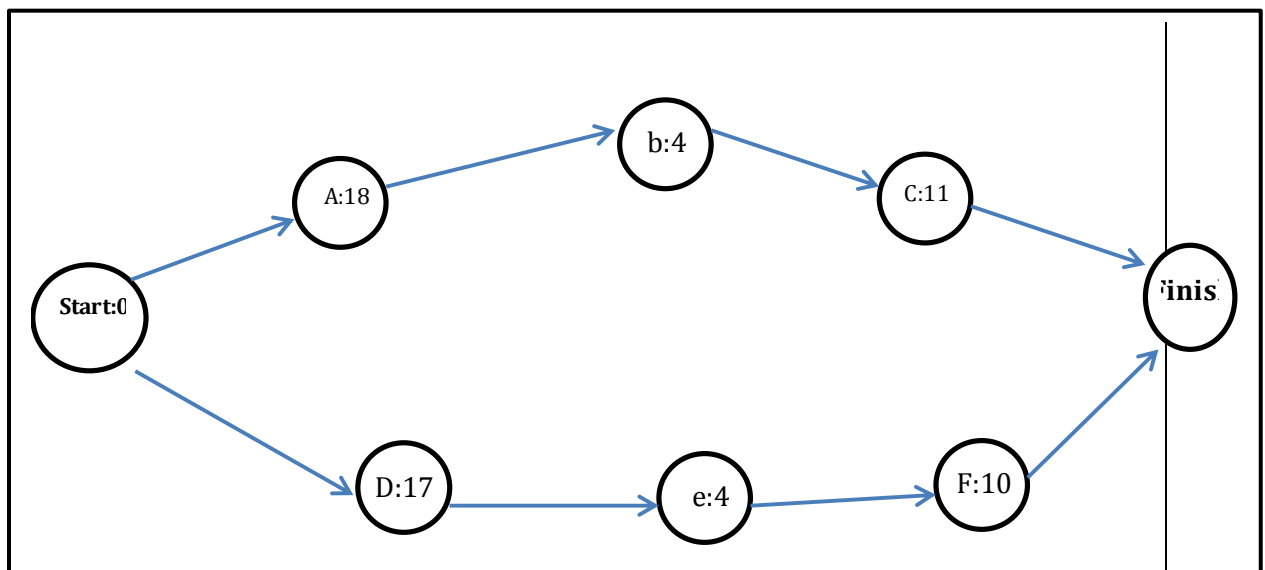


Figure 4
AON Diagram: Project of Travelling to Bangalore

The AON diagram for the Delhi to Bangalore travel project is shown on Figure 4. On this network there are two ways to begin at start node and traverse the network to finish node. These two ways are called **paths**. A **path** is a set of nodes connected by arrows, which begin at the start node of a network and end at the finish node. In the figure 4 there are two paths, start-a-b-c-finish and start-d-e-f-finish. The length of a path in a network is the total time it takes to travel the path. The time is calculated by adding the individual times of connected nodes on the path.

A path is called critical path if it is the longest path in a project network. Thus Mr. X's path is the only critical path in the project of travelling to Bangalore. Jobs on critical paths are called critical jobs.

These jobs are critical in determining the project duration: To shorten the time to complete a project, we must shorten the jobs on the longest path in its network – the critical path.

To see this, note that if Mr. Y decides to eliminate his stopover in Mumbai, activity e, thus shortening his travel time to 27 hours, it still takes 33 hours before he and Mr. X can meet in Bangalore. On the other hand, if Mr. X cuts his stay at home from 4 hours to 1 hour, they can meet in 30 hours rather than 33 hours. Therefore, Mr. X's path and jobs along it are critical in determining the length of the project that is when he and Mr. Y can meet in Bangalore.

Job Slack

Going back to our original example where X's path is critical, we note that Y can leave 2 hours later than X and still not delay their meeting in Bangalore. Alternatively Y could travel more slowly or take longer meeting time in Mumbai as long as the total delay did not exceed 2 hours. We can say then that Y has some slack in his path.

Generally speaking, jobs not on the critical path have slack (relative to critical jobs). Such information is extremely useful to project managers because it tells them how much flexibility they have in scheduling various jobs.

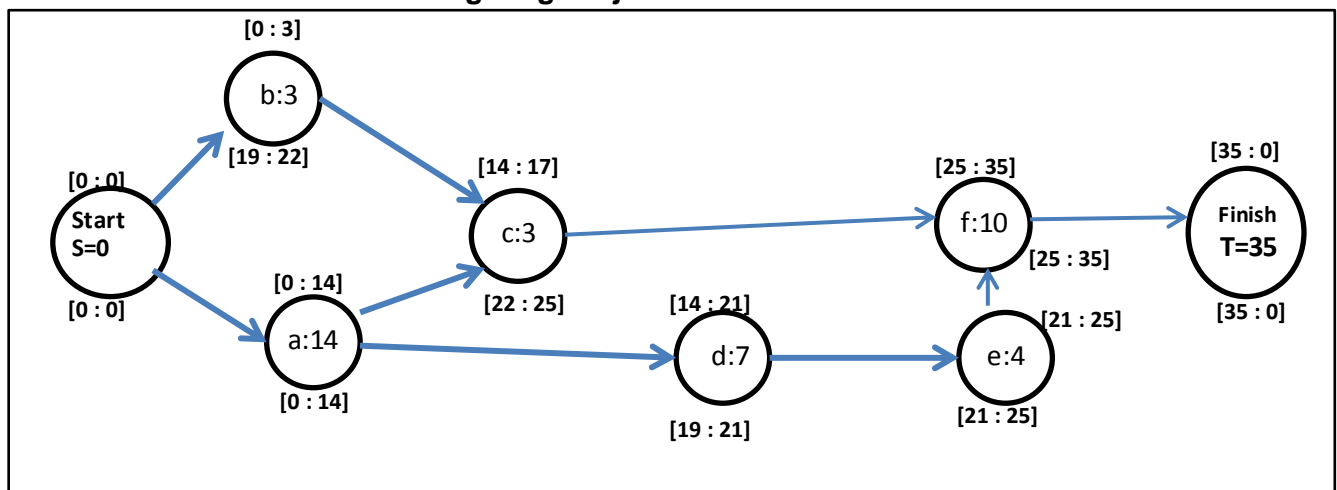
An algorithm for Finding the Critical path

In the previous section we showed how jobs not on the critical path could be delayed or lengthened without changing the time needed to complete the project. Let us now devise a set of procedure that will identify the critical path and show how the start and finish times of certain activities may be changed without affecting the duration of the project. In general, a set of procedures or collection of rules specifying calculations, which lead to a desired result, is referred to as an algorithm. We will present an algorithm for finding the critical path of a project and the start and finish times of all the activities within the project. We will use the budget example of DEEP.

Early Start and Early Finish Times

We begin with some definitions. The **early start** of a job in a project is the earliest possible time that the job can begin, and we label it $ES()$, where the job names appear in the parenthesis. The **early finish** of a job denoted by $EF()$, is its early start time plus the time needed to complete the job. For job a this would be $EF(a) = ES(a) + t(a)$.

Figure 5
Budgeting Project of DEEP



Now before a job can start all its immediate predecessors must have been completed. Suppose that the project of budgeting at DEEP begins at time 0. In general, we indicate the **start time** of a project by the symbol S . For jobs with no predecessors, then $ES = S$. Since job a and b have no predecessors, then $ES(a)$ is 0 and $ES(b)$ is 0 and their corresponding early finish times are $EF(a) = 0 + 14 = 14$ and $EF(b) = 0 + 3 = 3$.

The job c cannot start until its immediate predecessors a and b both are completed. The early finish time of a is 14 and that of b is 3. Since both have to be completed before c begins; c cannot begin until day 14. So we write $ES(c) = 14$. In general the early start time of a job is the largest, or maximum, of the early finish times of all its

immediate predecessors. Job c's early finish time is its early start time plus the time needed to complete it, or $EF(c) = ES(c) + t(c) = 14 + 3 = 17$.

Job 'd' early start time is 14, the early finish time of a, its only predecessor. It follows that $EF(d) = 14 + 7 = 21$. Similarly the ES (e) is 21 and EF (e) is 25. Now we can calculate ES (f) as the maximum of the EF times of c and e, its immediate predecessors. Since 25 is the maximum, then $ES(f) = 25$ and $EF(f) = 35$. Thus the budget can be completed in 35 days. We usually refer to the completion, or finish date, as T, so $T = 35$ in our example.

Obviously, if the earliest completion date of the budget is 35 days after its beginning, the longest path through its network must be 35 days length. A fast perusal of Figure 5 will show this to be correct; the path consisting of jobs a, d, e and f is 35 days long. This sequence, then, is the project's **critical path** – the longest path in the network.

To briefly summarise our procedure, we start at the beginning of the project network and calculate the early start and then the early finish times for each of the beginning jobs (those with no predecessors). Then we do the same for their successors, their successors' successors, and so on until all jobs in the project have been considered. This procedure is called the **forward pass** through the network. Notice that no job can be considered until all its immediate predecessors have been. Thus the jobs are examined in technological orders. The **forward pass** then, yields an ES and EF for each activity, and the earliest finish date, T, for the project.

Late Start and Late Finish Times

As we mentioned before, activities that are not in critical path can be delayed without delaying the completion date (T) of the project. Reasonable questions are: How much can they be delayed? How late a particular activity be started and still not lengthen the project duration? It will be helpful in answering these questions if first we define the **late start** (LS) of an activity as the latest time it can begin without pushing the finish date of the project further into the future. Similarly, the **late finish** (LF) of an activity is its late start time plus its duration. In symbols for job a $LF(a) = LS(a) + t(a)$, or in a form that will be more useful, $LS(a) = LF(a) - t(a)$.

To calculate late start and late finish times, we begin at the end of the network and work backward. This time we go through a backward pass. In our budget example, it means that we start at finish node. The only job leading into finish node is job f. It must be completed by day 35 so as not to delay the project; therefore, day 35 is its late finish. In general jobs with no successors, we set $LF = T$. Since it takes 10 days to do job f, it must begin at day $35 - 10$ or day 25; which is its late start.

Now there are two jobs, e and c, which are predecessors of f. The late finish time of both have to be 25, or 'f' would be delayed beyond its late start and the project not completed by day 35. The c's late start is 22, since $LS(c) = LF(c) - t(c) = 25 - 3 = 22$. Similarly, e's late start is 21; since the job had duration of 4 days.

Job d is the only activity leading into node e, and its LF becomes 21 since that is the LS of d's lone successor, job e, which emanates from the node. Subtracting, d's duration of 7 days, we obtain $LS(d) = 21 - 7 = 14$. Similarly, LS (b) is 22 and LF (b)

is 19. Two jobs c and d are the successors of job a. The LS times of c and d are 22 and 14. So the late finish time of job a must be the smallest or minimum of the late starts of the jobs c and d. Hence LF (a) is 14, the minimum of 14 and 22. LS (a) then equals LF (a) + (a) = 14 - 14 + 0. So a must begin at day 0. We have completed our backward pass, which yielded late start and finish times for all jobs.

Total Slack

Figure 5 shows that for some of the jobs, late starts and early starts (or similarly, late finishes and early finishes) are identical. Thus ES (a) = LS (a) = 0 and EF (e) = LF (e) = 25. On the other hand, for example, ES (c) = 14 and LS (c) is 22. This means that job c may start any time between day 14 and day 22 and still not delay the completion of the project. We say then that job c has slack; and we define the **Total Slack** (TS) of a job, or activity as the difference between its late start and early start times (or equivalently), as the difference between its late finish and early finish times.

For job c, we have

$$TS(c) = LS(c) - ES(c) = 22 - 14 = 8 \text{ days}$$

Or

$$TS(c) = LF(c) - EF(c) = 25 - 17 = 8 \text{ days}$$

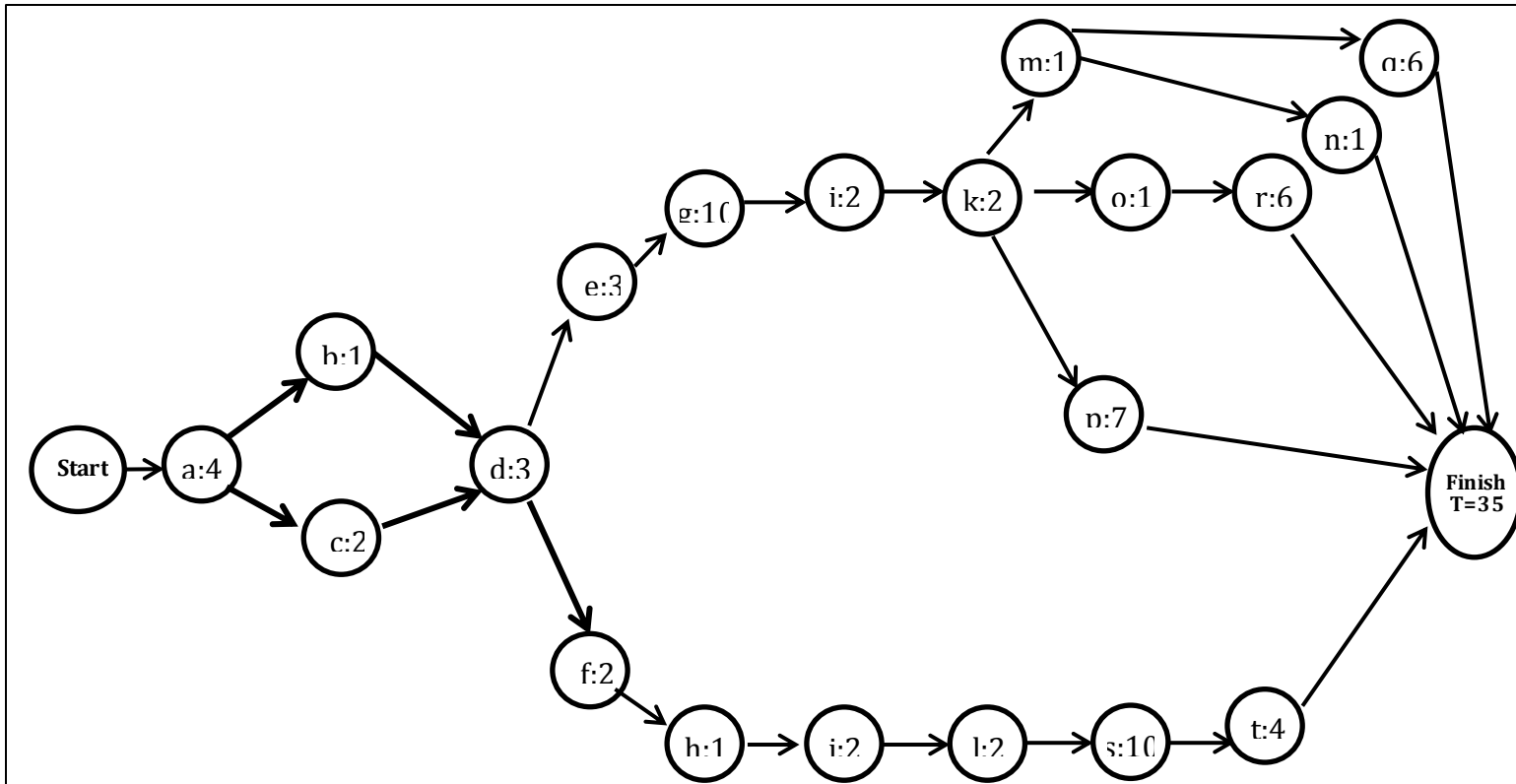
Obviously, if a job is on the critical or longest path, delaying its start would delay the project finish date. Hence its late and early start times must be identical. It follows that jobs on the critical path have zero total slack, for their LS = ES, then the difference between the two must be zero. Another definition of a critical path, therefore, is one whose jobs have zero total slack. If we delay the start of non-critical jobs (one that has slack), we will frequently delay the start of the jobs succeeding it. For example, suppose that we delay job c until day 18. Then its EF time is 18 + 3 = 21. If job a is completed by day 14 as before, c still cannot be started until day 21, the completion date of d. Thus c's ES increases from 14 to 21, and its total slack is reduced from 8 days to 1 day (22 - 21).

Free Slack

In our budget example notice that the early start time of c is 14. To put it another way, c cannot start until job a is completed on day 14. Job b is also a predecessor of c, but b takes only 3 days to complete. Thus it can start as late as day 11 and still not delay the early start of c. We say then that b has **free slack** (FS), which we define as the amount of time a job can be delayed without affecting the early start time of any other job. For calculation purposes, the free slack of job is the difference between its early finish time and the earliest of the early start times of all its immediate predecessors. Thus FS (b) = ES (c) - EF (b) = 14 - 3 = 11. Free slack, of course, can never exceed total slack; moreover, all jobs that have total slack do not necessarily have free slack. In general, a job has free slack if it has more total slack than one of its immediate predecessors.

Watershed Project

Job Name	Job Description	Immediate predecessors	Time (days)
a	Identification and delineation of watershed area	-	4
b	Meeting with villagers of watershed area	a	1
c	Meeting with Panchayat	a	2
d	Formation of watershed committee	b, c	3
e	Training of selected village youth for surveys on land and water uses and forest resources	d	3
f	Training of selected village youth for socio-economic survey	d	2
g	Land, water and forest resource survey	e	10
h	Socio-economic survey	f	10
i	Identification of problems and options related to land, water and forest resources	g	2
j	Identification of problems and options related to socio-economic development	h	2
k	Preparation of natural resource development plan	i	2
l	Preparation of socio-economic development plan	j	2
m	Nursery preparation	k	15
n	Plantation in degraded forest area	m	10
o	Contour bunding	k	10
p	Vegetative fencing	k	7
q	Pasture land development	m	6
r	Check dam preparation	o	6
s	SHG formation	l	10
t	Income generation training for SHGs	s	4

AON Diagram: Watershed Project**Different Paths and Critical Path of Watershed Project**

A: start-a-c-d-f-h-j-l-s-t-finish = $0+4+2+3+2+10+2+2+10+4+0 = 39$

B: start-a-b-d-f-h-j-l-s-t-finish = $0+4+1+3+2+10+2+2+10+4+0 = 38$

C: start-a-b-d-e-g-i-k-m-q-finish = $0+4+1+3+3+10+2+2+15+6+0 = 46$

D: start-a-c-d-e-g-i-k-m-q-finish = $0+4+2+3+3+10+2+2+15+6+0 = 47$

E: start-a-b-d-e-g-i-k-m-n-finish = $0+4+1+3+3+10+2+2+15+10+0 = 50$

F: start-a-c-d-e-g-i-k-m-n-finish = $0+4+2+3+3+10+2+2+15+10+0 = 51$

G: start-a-c-d-e-g-i-k-o-r-finish = $0+4+2+2+3+3+10+2+2+10+6+0 = 42$

H: start-a-b-d-e-g-i-k-o-r-finish = $0+4+1+3+3+10+2+2+10+6+0 = 41$

I: start-a-b-d-e-g-i-k-p-finish = $0+4+1+3+3+10+2+2+7+0 = 32$

J: start-a-c-d-e-g-i-k-p-finish = $0+4+2+3+3+10+2+2+7+0 = 33$

© 2001 PRIA. The text may be reproduced for non-commercial purposes, provided credit is given to PRIA. To obtain permission for uses beyond those outlined in the Creative Commons license, please contact PRIA Library at library@pria.org. Please use the following citation:

Bandyopadhyay, KK. (2001). Programme Evaluation and Review Techniques (PERT) And Critical Path Method (CPM):PRIA



Participatory Research in Asia

42, Tughlakabad Institutional Area, New Delhi-110062

Ph:+91-011-29960931/32/33

Web: www.pria.org