

December 25, 2017

## CHAPTER 4

### SUPPORTING PLANNING AND CONTROL: A CASE EXAMPLE

#### Chapter Outline

#### 4.1 Background

What was the cause of the “desperation” that lead to the development of the Program Evaluation and Review Technique (PERT)?

#### 4.2 Project Planning - The Network Diagram

Explain why “any project involves a number of interrelated activities or jobs.”

What is a “predecessor activity.”

Explain “precedence relationships” in a project.

#### 4.3 Converting Time Estimates into Planning Information

Explain the difference between *time estimates* (data) and *planning information* in a PERT system.

#### 4.4 Trade-Off between Time and Cost in Selecting the "Best" Plan

Explain why time and cost involve a “trade-off in most projects.

What is the Least Cost Method as used in project planning?

#### 4.5 Scheduling the Project

What is *activity scheduling*? How does it relate to *time analysis*?

#### 4.6 Information for Control

What happens in a network when activity time estimates change?

#### 4.7 Summary

Explain how network analysis can be considered a comprehensive DSS.

## 4.1 Background

To illustrate how a DSS can provide actionable information for planning and control, we use Network Analysis, a popular tool for managing projects. We will not discuss the mechanics of this technique since it is covered in all introductory management science books. Our aim is to highlight the "information" aspects of Network Analysis as a DSS to support project management.

For readers who are unfamiliar with Network Analysis terminology, we include a brief glossary of terms at the end of the chapter, and flag terms in the text that are explained in the glossary with the † symbol.

Network Analysis is a generic name for a number of procedures which are all based on the concept of a "network diagram." Some common variants of this basic approach are PERT (Program Evaluation and Review Technique), CPM (Critical Path Method) and RAMPS (Resource Allocation and Multiple Project Scheduling).

### **Program Evaluation and Review Technique - PERT**

PERT was born of sheer desperation. In 1956, during the initial stages of the U.S. Navy's Polaris missile development program, the Special Projects Office in charge of this immense project found that all the conventional management methods were hopelessly inadequate to keep track of the schedule.

Superimposed on the job of coordinating the efforts of 11,000 contractors was a degree of *uncertainty* as to when crucial research and development stages might be completed. PERT was devised then with *time* as the critical factor, and its application is credited with saving two years from the original estimate of five years required to complete the project. An interesting aspect of PERT was the use of three completion time estimates for each activity, and the application of statistical probability theory to forecast the likely chance of completing the project within a given date. The success of this application led the U.S. Department of Defense to specify that all future defense contracts must use PERT.

Since the Polaris project the method has undergone considerable development and is called PERT/Cost. The expanded version of PERT is a

comprehensive system, which encompasses cost and resource aspects in addition to time.

### **Critical Path Method - CPM**

Concurrently with PERT, CPM was developed at DuPont for scheduling chemical plant construction. The entire emphasis in the initial development of PERT was on time since early military applications were intent on completing the project in the shortest possible time, there being no cost constraints. In the case of most "business" projects, however, costs also have to be considered. In general, project time can be reduced but only with an attendant increase in costs. The CPM technique relates *costs* to *time* and shows how to accelerate the project for the lowest possible cost.

### **Resource Allocation and Multiple Project Scheduling - RAMPS**

Neither PERT nor CPM in their early development considered the detailed problems of scheduling and allocating *resources*. Reduction of project duration to a minimum is a desirable aim, but this objective must be tempered with the need to reduce peak resources requirements and to avoid periods when resources are not fully used. Here again, research originated at DuPont and took shape under the name of RAMPS. This technique can be applied to the problem of allocating various resources like manpower, equipment, assembly floor space, etc., over the entire duration of the project in the best possible way. The RAMPS method is also capable, as its name implies, of taking into account several projects running together, with the projects in competition for the same limited resources.

## **4.2 Project Planning - The Network Diagram**

Any project involves a number of interrelated activities or jobs. The crux of Network Analysis is the *network diagram*<sup>†</sup> which reflects the *logical flow* of the work to be done. The activities are represented by arrows which are linked as follows. Activities that logically are to follow each other are drawn in sequence with the direction of the arrow indicating progress. Activities that can be carried out concurrently are represented by parallel arrows. Exhibit 4.1 shows a network diagram, also called an *arrow diagram*<sup>†</sup>, for a small project. The diagram would, of course, be much larger for a complex project and would take a considerable amount of time on the part of the project manager and the project planning team to develop. Nevertheless it is worth the effort since the very act of creating the arrow diagram ensures that all the jobs to be done are represented where they belong without any omission. The final diagram enables the entire project scope to be

immediately, and visually, assimilated. It is superior to a prose description which could run into several pages for a complex project resulting in the interrelationships being lost in the welter of words.

The charting tools that are available in several mainframe and personal computer packages enable the arrow diagram to be easily drawn, stored and revised as necessary. Several customized packages for network analysis are also commercially available.

Viewed from the perspective of a Decision Support System, the arrow diagram represents the first step to support the *planning process* for a project. It depicts in a graphical form:

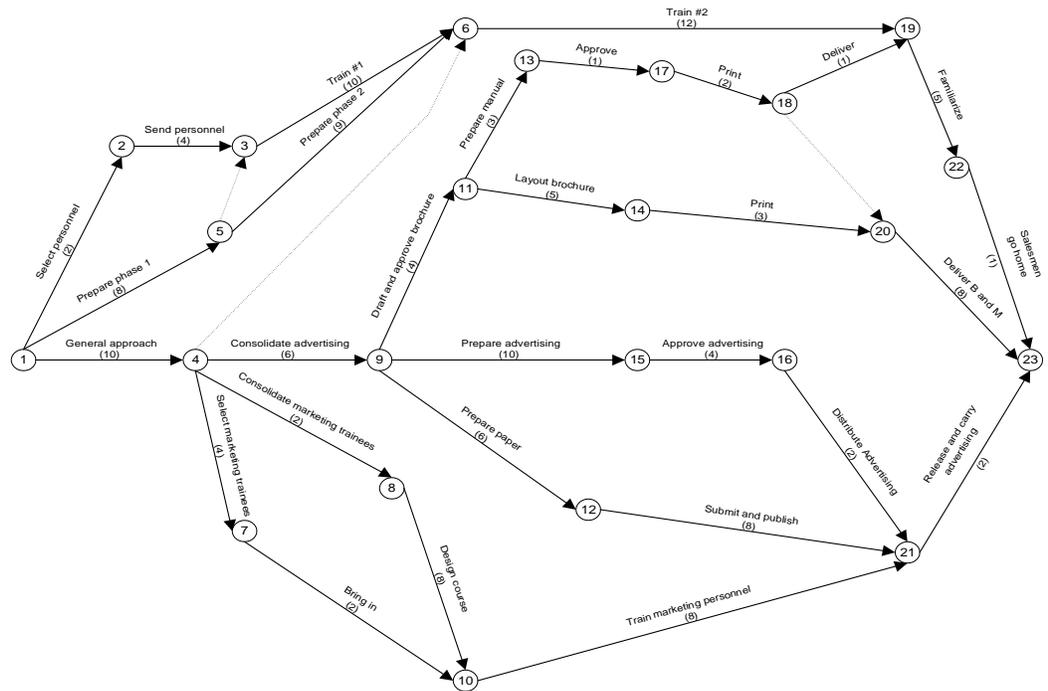
- \* all the activities to be carried out
- \* the prerequisites (or predecessor activities†) for each activity - the activities that must be done before a given activity can be started
- \* activities that are not dependent on each other, and hence can be undertaken concurrently.

### **4.3 Converting Time Estimates into Planning Information**

Having drawn the diagram, the next step in the planning process is to estimate the time required for each activity. In case of uncertainty†, three time estimates can be provided for the activity representing the optimistic, most likely, and pessimistic time estimates to complete that activity. These time estimates can be viewed as the raw data. A time analysis based on a simple mathematical model converts this raw data into useful *planning information* such as:

**Exhibit 4.1**

**Network Diagram  
 for a Project  
 to Introduce  
 a New  
 Computer  
 System**



This diagram is taken from the Instructor's Manual for Casebooks in Production Management: Basic Problems, Concepts and Techniques  
 A. R. Dooley, William K. Holstein, et. al., John W. Wiley & Son, Inc., New York, 1964, p.117.

- \* what is the total *project duration*†?
- \* in the case of uncertainty reflected by the three time estimates for the activities, what is the probability of completing the project in X number of days?
- \* what are the *critical activities*†, i.e., activities that must be completed on time to meet the project deadline?
- \* what is the *slack*† available for the non-critical activities?

The planning information enables the project team to set priorities for executing the various activities. Clearly, the critical activities are top priority. For the noncritical activities†, the slack is a measure of the amount of leeway that is available for each job. A slack time of one day or week or

whatever time unit is used is indicative of a near-critical† job deserving a much higher priority than a a job with a slack time of 10.

There is much more in the variety of planning information that is available from the time analysis. Our aim here is to provide just a flavor for how it converts the raw data on time estimates into useful information for the planning function. An example is provided in section 4.5 where the use of this information for scheduling the activities is discussed.

#### 4.4 Trade-Off between Time and Cost in Selecting the "Best" Plan

The next step, still in the planning process, is to add cost data to the activities to determine the optimal project duration, i.e., the one that incurs the lowest total project cost. For this purpose, "direct" costs of labor, material, equipment, etc. need to be distinguished from "indirect" costs such as overheads and other costs such as cost of lost sales and penalties for contract delays (or bonuses for early completion). Direct costs tend to increase as project duration is reduced since more resources are used, as opposed to indirect costs, which will decrease. The *Least Cost Method* compares the two costs for a range of possible project times to determine the duration with the least *total* cost.

We describe the Least Cost Method here in some detail for several reasons. First, it shows that to improve the quality of the planning information, more data is needed. Next, data by itself, is of no value unless it is converted into actionable information. The Least Cost Method performs this conversion and enables the decision-maker to make a choice from the available alternatives during the project planning phase. Finally, it shows how an *algorithm*, or systematic procedure, can be constructed to determine an optimum solution, in this case, one that minimizes costs.

To apply the Least Cost Method, data on direct costs has to be collected for each activity in the following manner:

1. Determine the *Normal Cost*†, which is defined as the minimum direct cost of carrying out the activity, using normal means and avoiding overtime, use of special staff, resources or materials.
2. The minimum time to do the activity for the Normal Cost is called the *Normal Time*†. It is that activity time for which direct costs are lowest. No amount of extra time will reduce the cost below the Normal Cost. In fact, allowing

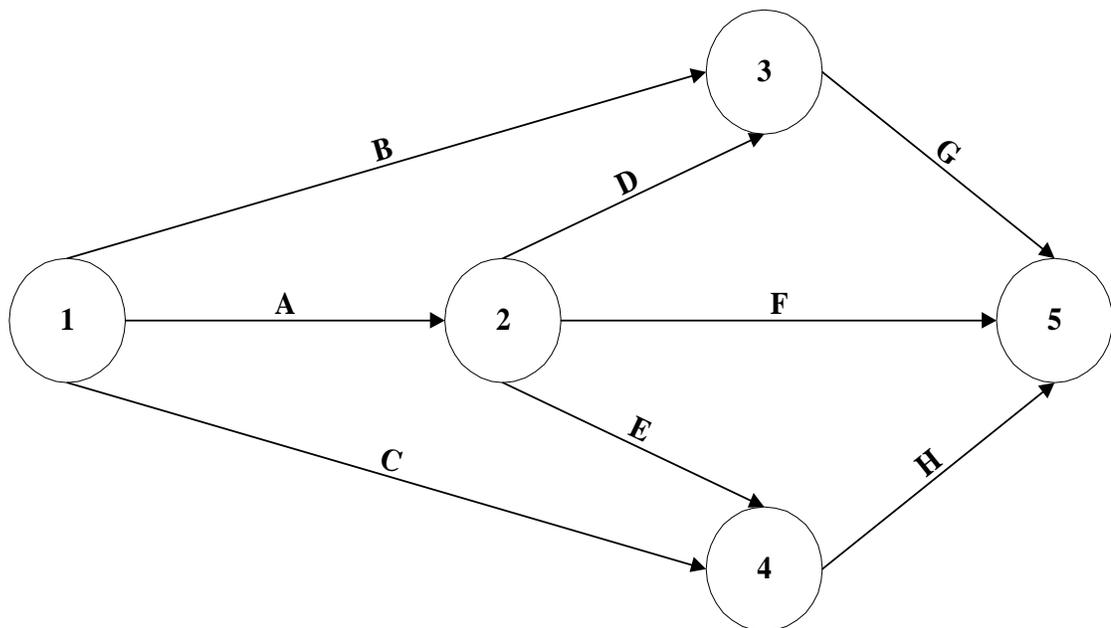
too much time can produce an inefficient situation and may increase the cost.

3. As the time for performing an activity is compressed, its direct costs will in general increase. The shortest time for completing an activity using whatever means is available is called the *Crash Time*†.
4. Costs are higher in the crash situation than in the normal situation because of extra resources being used. The minimum direct cost for performing the activity in the Crash Time is defined to be the *Crash Cost*†. No amount of extra expense beyond the crash cost will reduce the duration below the Crash Time.

To illustrate the Least Cost Method, consider the sample network drawn in Exhibit 4.2 with the Normal and Crash times and costs for the activities as given in Exhibit 4.3.

**Exhibit 4.2**

**Sample  
Project  
Network  
Diagram**



**Exhibit 4.3**

**Normal  
 and Crash  
 Time and  
 Cost Data  
 for  
 Diagram in  
 Exhibit 4.2**

Activity	Normal Days	Cost of Normal Cost	Crash Days	Crash Cost	\$ per Day
A	3	\$140	2	\$210	70
B	6	\$215	5	\$275	60
C	2	\$160	1	\$240	80
D	4	\$130	3	\$180	50
E	2	\$170	1	\$250	80
F	7	\$165	4	\$285	40
G	4	\$210	3	\$290	80
H	3	\$110	2	\$160	50

The Least Cost Method begins with an analysis of this data. The lowest direct cost of the project, \$1,300 in the example, is achieved when all the activities are performed at their Normal Times. The *Normal Project Duration*, which can be determined by the time analysis, is by definition the longest - 11 days in the example. The next question is; What is the shortest time for completing the project? An immediate answer can be provided by using the Crash Times instead of the Normal Times for all activities in the time analysis. In our example, the *Crash Project Duration* is 8 days. The project cost when all activities are speeded up (crashed), is \$1,890. But, is it necessary to crash all the activities to achieve the Crash Project Duration?

The algorithm operates as follows to answer this question:

*Step 1:* Start with the network with the Normal Times and crash *only one day* from the project duration. Why? Because only the critical activities have to be considered since the noncritical activities will have slack of at least one day.

Which critical activities to crash? If there is only one critical path, only one activity on that path has to be crashed. In the case of multiple paths, each path has to be shortened by one day. In our example, there is only one critical path: A-D-G. Select the activity that costs the least to crash by one day - Activity D in the example has a crashing cost of \$50 per day, compared to \$70 per day for A and \$80 per day for G.

Note that a linear *approximation* is used between the costs for the Normal Time and the Crash Time to determine the Crashing Cost per day†. Also, there could be other factors that might

favor a more expensive critical activity for crashing such as the reliability of the person doing the job to get it done on time. But these factors are not easy to quantify and, hence, such considerations have to be applied by the user when reviewing the solution generated by the Least Cost Method.

*Step 2:* Redo the time analysis of the network with the selected critical activity crashed by one day. In our example, the project duration with D crashed by one day will be 10 days, with an additional cost of \$50, or a total project cost of \$1,350.

This is the best (i.e. the least cost) schedule for completing the project in 10 days. The old critical path, A-D-G, will still be critical. But there could be new critical paths. In this example, two new critical paths have emerged; B-G and A-F. Note that the three critical paths have some common activities.

*Step 3:* Go back to Step 1. Crash one day from the new network. In our example, we have three critical paths:

A (3 days) - D (3 days) - G (4 days)	Total = 10
B (6 days) - G (4 days)	Total = 10
A (3 days) - F (7 days)	Total = 10

D cannot be crashed any more. The alternatives for crashing one day from each path can now be listed:

1. A and B
2. G and F
3. A and G

Note that crashing activities A, B and F is also an alternative, but it is ruled out since it will cost more than crashing just A and B. The crashing costs for the three alternatives are:

1. Crash A and B by one day each: Cost = \$130
2. Crash G and F by one day each: Cost = \$120
3. Crash A and G by one day each: Cost = \$150

On the face of it, Alternative 2 costs the least, but is it really so? Look at Alternative 3. When A and G are crashed by one day each, the A-D-G path is now at 8 days, whereas the other two paths, B-G and A-F will take 9 days. This brings up the

possibility of "uncrashing" D from 3 days to its Normal Time of 4 days, leading to a saving of \$50. The final cost of Alternative 3, when D is uncrashed, is \$100, which is then the cheapest alternative. Thus the algorithm for crashing one day at a time, although straightforward in principle, is tedious to work out because of the different combinations that have to be considered in multiple critical paths as in the case of even a simple example.

This is where computer technology comes in - the algorithm can be programmed to operate using the principles that we applied in our simple example. Further, commercial software packages are available because of the popularity of Network Analysis.

*Step 4:* Cycle through Steps 1 and 2 until the Crash Project Duration is achieved. In our example, the schedules for the range of durations from 11 days, the Normal Project Duration, to 8 days, the Crash Project Duration, is shown in Exhibit 4.4. Note that the least cost schedule for achieving the Crash Project Duration does not entail crashing all the activities in the network - only those that matter (C, E and H were not crashed at all in this example) and, further, only to the extent necessary (F was crashed only by one day).

The Least Cost Method has thus generated the best schedules for the range of project durations, best in terms of the lowest direct cost for each duration. When plotted, the Direct Project Time-Cost Curve will look like the one shown in Exhibit 4.5.

The indirect costs are then added to the graph for each project duration. These costs are not delineated here, but generally indirect costs will increase as the project duration is increased. A straight-line indirect cost function is assumed here.

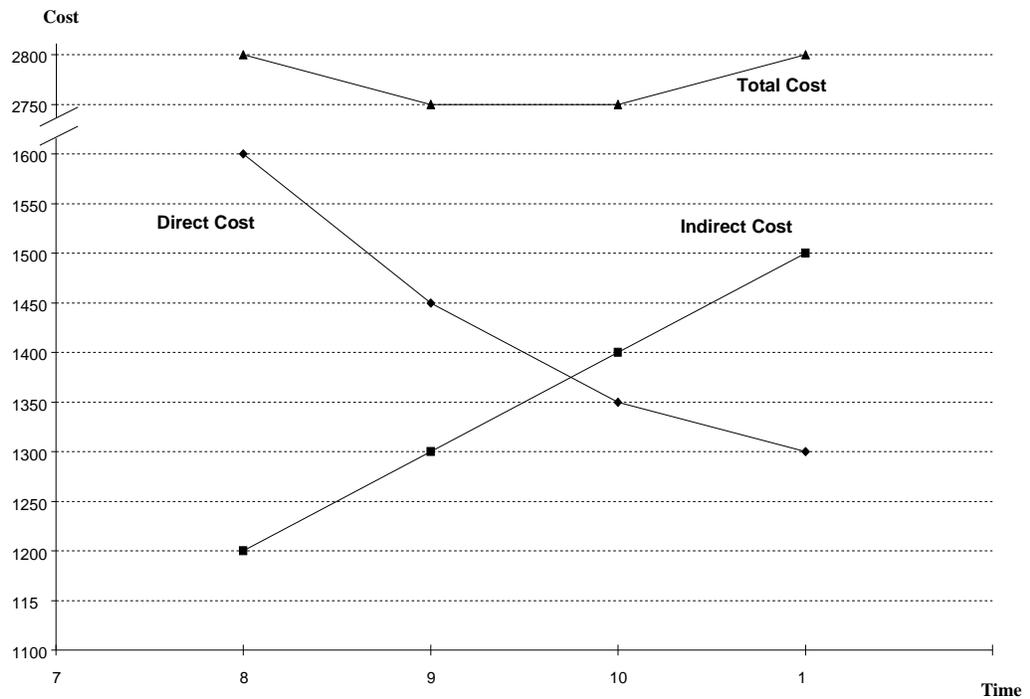
The Total Project Cost, which represents the sum of the direct and indirect costs, will be an U-shaped curve as shown in Exhibit 4.5 and yields the optimum project duration, i.e., the one that costs the least. The shape of the U-curve provides valuable information to the user on the impact of deviations from the optimum duration. A relatively flat U-curve shows that the optimum solution is less sensitive to a deviation than a sharp U-curve.

**Exhibit 4.4**

	Duration	Schedule	Total Direct Costs
<b>Project Duration versus Direct Costs</b>	11 days	All Activities at Normal Times	\$1,300
	10 days	Crash only D by one day	\$1,350
	9 days	Crash A and G by one day each and "Uncrash" D	\$1,450
	8 days	Crash D, B and F by one day each	\$1,600

**Exhibit 4.5**

**Project Time/Cost Graph**

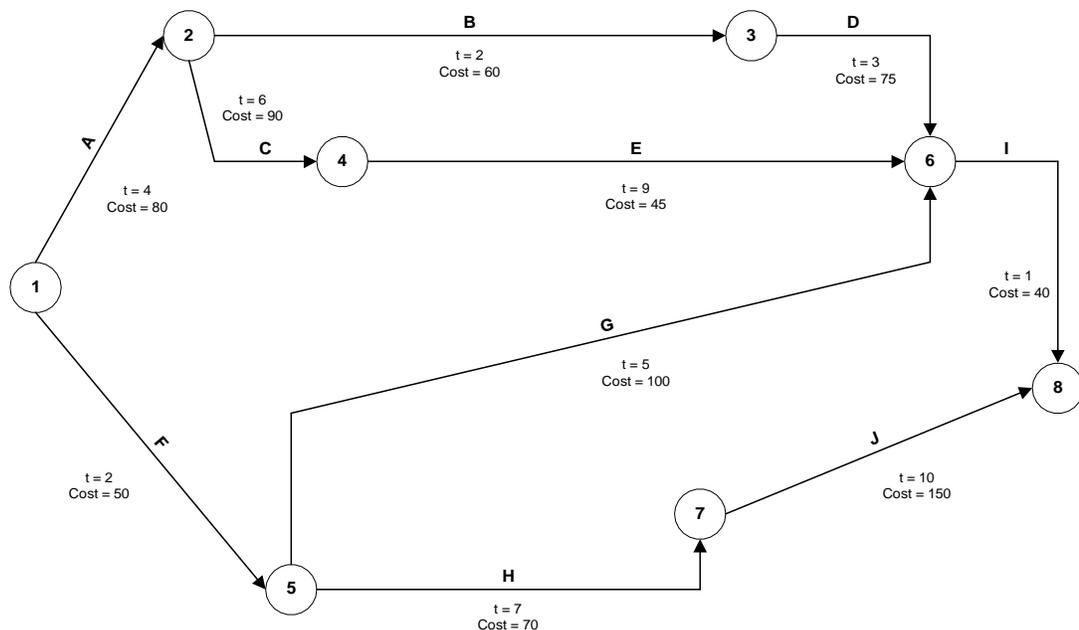


## 4.5 Scheduling the Project

Having determined the most practicable project duration from the U-shaped Project Time-Cost Curve, the next step in the planning function is to *schedule*† the various activities and organize the necessary resources for implementing these activities. To illustrate how the information from the time analysis is used for the purpose, let us consider the project network shown in Exhibit 4.6. Normal Time estimates and the cost for completing each activity in the Normal Time are displayed. The desired completion time for the project is assumed to be "as early as possible," or in the minimum time. This implies that the length of the critical path will determine the completion time of the project.

Exhibit 4.6

### Sample Network Diagram



The time analysis for the network in Exhibit 4.6 is shown in Exhibit 4.7. The critical activities have zero slack with no leeway in their start times. The noncritical activities have leeway in their start times depending on the amount of slack, in this case from 1 month (activities F, H and J) to 12 months (activity G).

In scheduling the project, the critical activities leave no choice. They have to be scheduled at their earliest start times. But there are choices to be made

for the start times of the noncritical activities - they can be scheduled to start at any time between their Early Start and Late Start times. The cost implications of the way in which the activities are scheduled can be observed in Exhibit 4.8, which depicts the two extreme cases - the Early Start Schedule†, in which every activity is started at its Early Start Time, and the Late Start Schedule†, in which every activity's start is delayed as long as possible. Observe that even though both the schedules end up with the same total cost, \$760, the way in which the expenditures are incurred over the life of the project is significantly different - the money is obviously spent sooner in the Early Start Schedule. However, in the Late Start Schedule, there is no slack available for noncritical activities since everything has been scheduled as late as possible (in essence, "using up" any available slack before an activity is started).

**Exhibit 4.7**  
  
**Time**  
**Analysis for**  
**Network in**  
**Exhibit 4.6**

Activity	Duration (months)	Early Start Time	Late Start Time	Slack	Total Cost	Cost per Month
A	4	0	0	0	80	20
B	2	4	14	10	60	30
C	6	4	4	0	90	15
D	3	6	16	10	75	25
E	9	10	10	0	45	5
F	2	0	1	1	50	25
G	5	2	14	12	100	20
H	7	2	3	1	70	10
I	1	19	19	0	40	40
J	10	9	10	1	150	15

Between the two extreme schedules is a range of feasible schedules that can be generated by shifting the start time of noncritical activities between the earliest and latest start time values. With the latest technology in personal computers that allow users to use handwriting on a computer screen or panel, managers can try different schedules and see the cash flow implications on a graph similar to those shown in Exhibit 4.9 for the Early Start and Late Start Schedules.

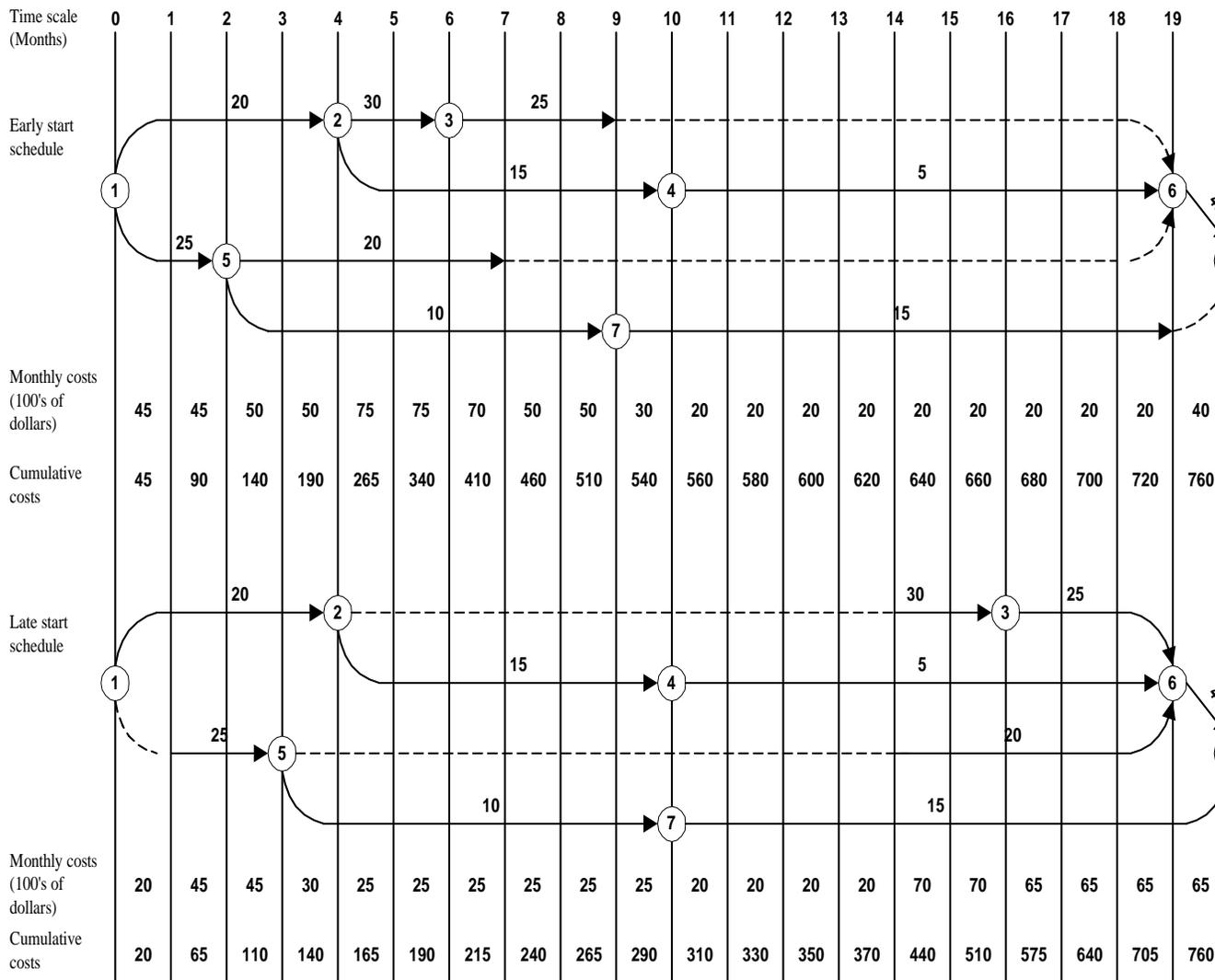
A similar analysis will have to be carried out for other critical resources such as people and machinery. Here again, there is no choice available for scheduling the critical activities. If the necessary resources for performing these activities are not made available, the project completion date will be

affected. But the noncritical activities do present choices of management. These choices need to be evaluated in terms of their resource implications to enable a satisfying schedule to be selected.

Our point here is to demonstrate that, to make these choices, the manager needs a sizable amount of data - the activity times and costs,

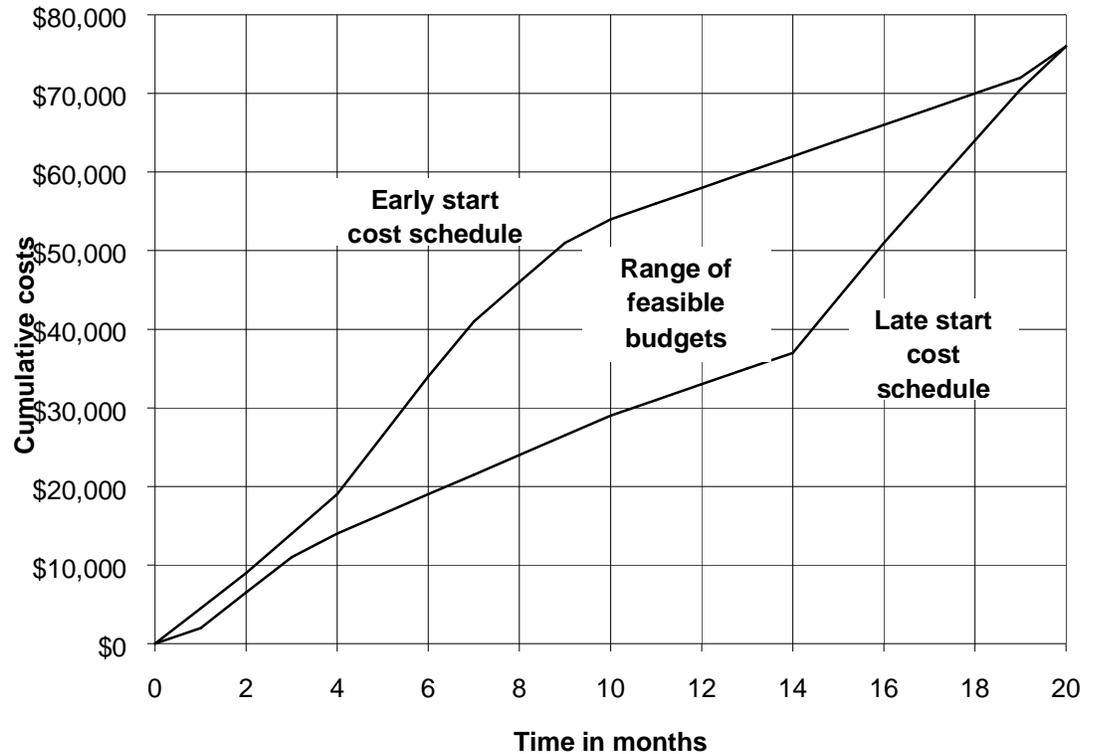
**Exhibit 4.8**

**Early and Late Start Schedule Graphs**



This material was adapted from A Management Guide to PERT/CPM  
 Jerome D. Wiest and Ferdinand K. Levy

**Exhibit 4.9**  
**Cumulative Costs for Early and Late Start Schedules**



Prentice-Hall, Englewood Cliffs, NJ, 1969, p. 87.

the relationships between and among the activities (the precedence relationships) - *and* the means to convert this data into useable *information* for decision-making. The time analysis model, with its emphasis on slack and earliest and latest start times, is of great value in organizing and analyzing the network data.

#### 4.6 Information for Control

Let us turn next to a consideration of how Network Analysis supports the control function. To begin, the critical path concept provides a means for applying *management by exception*. Clearly, the critical activities have to receive top priority since any slippage in these activities will affect the planned completion date.

Non-critical activities have to be controlled as well depending on the amount of available slack. Further, the time estimates for the activities could change as the project moves forward, making originally non-critical activities

critical, and possibly making some of the originally critical activities noncritical.

The time analysis procedure has a simple mechanism for incorporating the project status at any point:

- \* Activities that are completed are assigned a zero time
- \* Activities in progress are assigned the time needed to complete them
- \* Activities that have not yet started are assigned their most recent (updated) time estimates

The analysis is then repeated with the new time estimates resulting in updated information on project duration, critical path(s), slack for non-critical activities, etc.

A review of the updated information will result in the planning process starting anew for the activities that are in progress or yet to be started.

We go on to illustrate the richness of the control information by looking at project costs in addition to time. The basic concept of control says that comparing Actual to Budget means nothing since it all depends on the amount of work done for that expenditure. What should be compared with the Actual is not the Budget, but the *value of work completed*. For example, suppose the plan calls for the manufacture of 50 units of an item in 10 days at a budgeted cost of \$3 per unit, or a total "project" cost of \$150. At the end of 6 days, the actual expenditure was \$120. The budgeted expenditure for these 6 days is \$90 (assuming, as most budgeting procedures do, that the expenditures throughout the 10 days were uniform at \$15 per day). Does this mean that there is a cost overrun?

We cannot say for sure without knowing how much work was completed in the 6 days. If only 20 units were produced, then the value of the work completed is \$60 and there is a cost overrun of \$60 or 100 percent. Also, there is a slippage in time since, according to the original plan, 20 units should have been completed in 4 days, not 6. The "project" is therefore 2 days behind schedule. The analysis would obviously be quite different if 40 units had been completed in 6 days rather than 20.

We now illustrate how Network Analysis supports the control of project costs through a comparison of work scheduled with work actually

accomplished. Graphics provide an excellent medium for providing the control information as can be seen from the following example.

Exhibit 4.10 shows the Cumulative Budgeted Cost (Line A) corresponding to the planned schedule for starting the various jobs in the project. The project is scheduled to begin in January, 1990, and to end by the first of April, 1991, at a total cost of \$6 million. Suppose the project has been underway for some time - the current date is marked by the vertical line labeled "Today." The dashed line B shows the actual costs expended on the project and indicate that expenditures have exceeded the budget as of today.

Of and by itself, this fact does not mean much. Perhaps progress on the project is ahead of schedule and the higher cost reflects this. On the other hand, if the project is just on schedule or, worse, behind schedule based on the work actually accomplished, then actuals have overrun the budget.

What is the work that has been completed relative to the dollars budgeted for this work? Dotted line C displays the value of work completed, measured on the basis of the original cost estimates. Since line C is below line B, an *overrun* in costs has occurred. An *underrun* would be indicated when line C is above line B.

The amount of overrun at any point in time can be seen in the *vertical* distance between the two lines. As of today, actual costs (\$4.7 million) have exceeded the value of work completed (\$2.4 million) by \$2.3 million, an overrun of 95 percent.

The lower part of Exhibit 4.10 displays line D which is a running calculation of the percent overrun (or underrun, if it occurs), from the following formula:

$$\% \text{ Overrun (Underrun)} = \frac{\text{Actual Cost} - \text{Value of Work Completed}}{\text{Value of Work Completed}}$$

In this example, Line C also falls beneath line A, indicating that the project is behind schedule. The *horizontal* distance between line C at any point in time and line A shows the amount of delay at that time point. As of today (November 1), work valued at \$2.4 million has been completed, which was originally scheduled to have been completed by the third week in August. The project is hence about 2.25 months behind schedule.

Line E at the bottom of Exhibit 4.10 is a running tally of the number of months behind. It indicates that the project has progressively fallen further and further behind schedule and that initial delays are not being erased by accelerating subsequent work. Rather, delays are being further compounded.

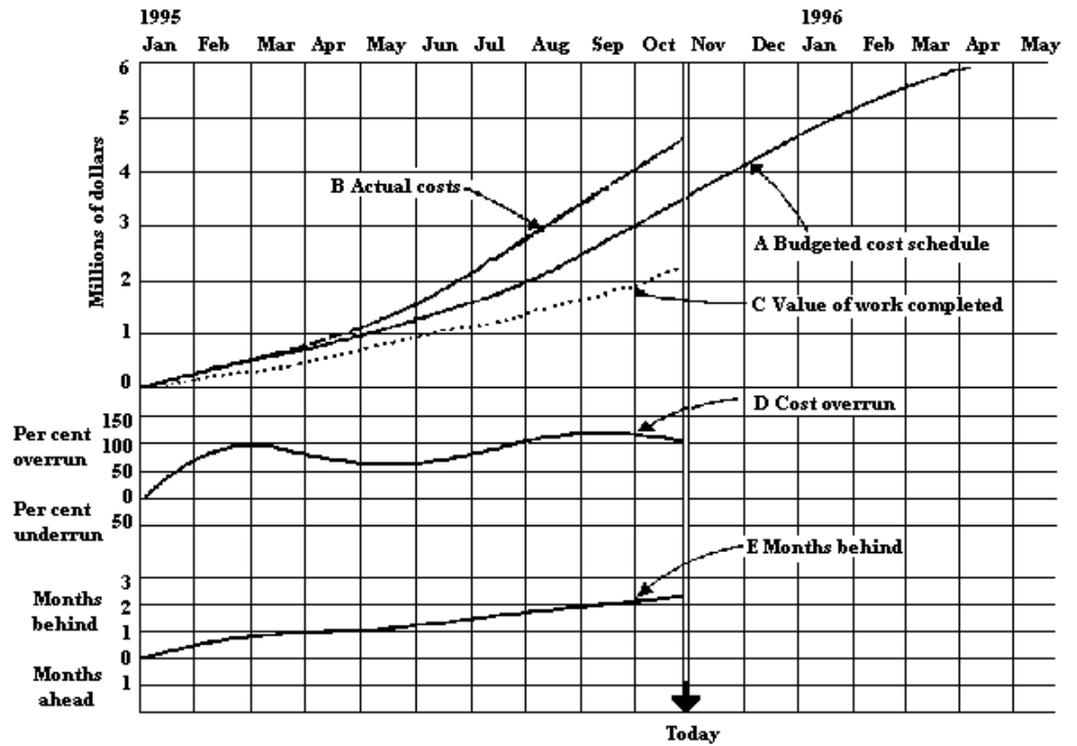
The project in this example obviously has serious problems to worry about. After only 10 months, the project is already more than 2 months behind schedule. Actual costs are almost double what they should be for the work accomplished. With the project's scheduled completion date rapidly approaching, the manager's immediate concern will probably be how soon can the project be completed? At what cost?

#### **4.7 Summary**

Network Analysis is an excellent example of a comprehensive DSS that supports project management in the full management cycle of planning and control. It provides a logical and disciplined framework for classifying objectives and assumptions, for highlighting what decisions must be made when those decisions must be made. It provides a framework for evaluating alternative strategies for executing the project. In particular, the critical path concept provides a means of *managing by exception*, rather than managing every detail. This principle that is not practiced often enough, in part because exceptions and "must-do" activities are not easily identified and brought to the manager's attention by the information system.

**Exhibit 4.10**

**Cumulative  
 Actual Costs  
 vs.  
 Budgeted  
 Costs and  
 Work  
 Completed**



This exhibit and the associated discussion of Value of Work Completed was adapted from A Management Guide to PERT/CPM Jerome D. Wiest and Ferdinand K. Levy Prentice-Hall, Englewood Cliffs, NJ, 1969, p. 90-92.

## **Glossary of Terms**

---

<b>Critical Path:</b>	The longest path in the network. The path along which any delay will cause a delay in the project completion time.
<b>Network Diagram:</b>	A logical flow diagram showing the work to be performed in a project. Activities are usually represented by arrows that are connected in a manner which shows which activities have to be performed before others can begin.
<b>Predecessor Activity:</b>	Any activity that must be completed before another activity can be performed.
<b>Uncertainty:</b>	Refers to the degree of error inherent in the time estimate for an activity's completion.
<b>Project Duration:</b>	The total time required to complete all of the activities that comprise the project. Since some activities may be performed simultaneously, the project duration is equal to the time required for the longest path (measured in time) within the network.
<b>Near-Critical Activity:</b>	An activity with a slack time that is close to zero, e.g. one day or one time unit.
<b>Critical Activity:</b>	An activity with no (zero) slack. An activity that falls along the critical path.
<b>Non-Critical Activity:</b>	An activity with a substantial amount of slack time.
<b>Slack:</b>	The time that an activity's start can be delayed without delaying the completion time of the project. The additional time available to complete an activity without increasing the project duration.
<b>Normal Cost:</b>	The regular direct cost of performing the activity in the normal planned time. Usually the minimum cost for performing the activity.
<b>Normal Time:</b>	The regular planned time needed to perform an activity at the Normal Cost. Usually the minimum time for performing the activity.
<b>Crash Cost:</b>	The direct cost associated with performing an activity in the crash time. Since the crash time is shorter than the normal time, the crash cost is higher than the normal cost.
<b>Crash Time:</b>	The shortest time for completing an activity using whatever means that are available. Since reducing (crashing) activity time expends resources, the costs of achieving the crash time are higher than the normal cost.

**Crashing Cost Per Day:** The direct cost per day for reducing the time needed to complete an activity. Usually determined by calculating the per day cost between the crash time and cost and the normal time and cost.

**Uncrashing:** Increasing the time to perform an activity in the interest of saving direct cost. Typically done when an activity was crashed, but the effect was unnecessary because of delays in other parts of the network.

**Schedule (activities):** The process of specifying which activities will be performed at what time. Usually done by setting the start date or time of each activity in the network.

**Early Start Schedule:** The schedule in which all activities are started at their earliest possible start time.

**Latest Start Schedule:** The schedule in which all activities are started at their latest possible start time.