UNIVERSITY OF SOUTHAMPTON

A GENERALIZED ALGORITHM TO EVALUATE PROJECT COMPLETION TIMES AND CRITICALITY INDICES FOR PERT NETWORKS

by

M.A. SOUKHAKIAN

A Thesis submitted for the Degree of

Doctor of Philosophy

Faculty of Social Sciences

Department of Accounting and Management Science

May 1988

To Professor C.B. Chapman

ACKNOWLEDGEMENT

I am very greatly indebted to Professor C.B. Chapman,my supervisor,for his advice, guidance, kind supervision and continuous encouragement.

My thanks are due to Mr S.C. Ward, who read the early drafts and made a number of valuable suggestions, and Dr J.H. Klein chairman of my research advisory group who read some parts of the dissertation and greatly encouraged me with his kind comments.

I also would like to thank Miss S.J. Vear for the many hours she devoted to typing and retyping the final draft.

Finally, I wish to acknowledge Shiraz University and Ministry of Culture and Higher Education the Government of Iran for giving me a scholarship award for this course of study.

LIST OF CONTENTS

UNIVERSITY OF SOUTHAMPTON

ABSTRACT

FACULTY OF SOCIAL SCIENCES

DEPARTMENT OF ACCOUNTING AND MANAGEMENT SCIENCE

Doctor of Philosophy

'A GENERALIZED ALGORITHM TO EVALUATE PROJECT COMPLETION TIMES AND CRITICALITY INDICES FOR PERT NETWORKS'

by Mohammad Ali Soukhakian

For stochastic PERT networks the main difficulty in calculating the probability distribution function (pdf) for project completion time is caused by structural and statistical dependence between activities. This dependency also makes it very hard to identify the most critical paths and activities.

This study presents a method for taking into account dependence between activities and provides a generalized algorithm to evaluate the project completion time and criticality index of each activity and path using a Controlled Interval and Memory (CIM) approach proposed by C.B. Chapman and D.F. Cooper (1983) Risk engineering: basic controlled interval and memory models. Journal of the Operational Research Society 34(1), 51-60.

The procedure allows activity durations to have any continuous or discrete distribution presented in a finite set of ordered pairs. It has been tested using simple activity network models with different statistical and structural dependence between activities.

The proposed procedure provides an exact pdf for project completion time when the duration times of activities are discrete and approximates the pdf of the project completion time when the duration times of activities are continuous. Approximation is due to: (i) discretizing continuous distributions, (ii) convoluting discrete approximations to continuous distribution.

The computational experience shows that the criticality indices obtained using the proposed procedure is very close to the exact criticality indices obtained using complete enumeration and both methods give the same ranking of criticality indices in most PERT networks.

Compared with Monte Carlo simulation, the proposed procedure is comparatively easy to understand and use for simple networks. Moreover, for the same level of precision, Monte Carlo simulaton requires much greater computation effort than the proposed procedure. However, Monte Carlo simulation may maintain a comparative advantage for very complex networks, and the ideal approach to networks in general may be a hybrid.

CHAPTER 1: INTRODUCTION

One of the most important problems in the analysis of PERT networks is the determination of the distribution function for project completion times. When the duration times of the activities of a project are random variables, the completion time of the project is also a random variable, with a distribution function that is a complex function of the distribution functions for each activity.

For networks with a special structure, the distribution function for project completion time can be obtained by reducing the network to a single, equivalent activity starting at an initial node (1) and ending at a terminal node (N). Assuming statistical independence of the durations of the network activities, the reduction is possible through repeated application of two well-known operations: convolution and greatest.

Convolution and greatest operations both involve the combining of probability distributions.

Many well-established approaches for combining probability distributions are available, including: analytical methods like moment-based approaches, simulation or sampling based methods, numerical methods, discrete probability interval distribution methods and interval or histogram representation methods.

If the PERT network satisfies the conditions necessary for the direct use of convolution and greatest operations, then the network is termed reducible; otherwise, it is termed irreducible. If the network is reducible to a single equivalent activity (1,N), then it is termed completely reducible. If the network is completely reducible, the analytical form of the distribution function of the project completion time can be determined. However, irreducibility of the network prevents such analytical determination.

In conventional PERT network models it is assumed that different paths are structurally independent. This is not true for irreducible networks,because in irreducible networks at least two paths share one or more common activities. For instance, in the "Wheatstone bridge" of Figure 1.1, which is the simplest irreducible network, there are three paths. Two (paths 1-2-4 and 1-3-4) can be analysed in a straightforward fashion because they are independent, but the third path (1-2-3-4) can not because of the existence of a common arc between it and each of the other two paths. The third path is structurally dependent upon the other paths.

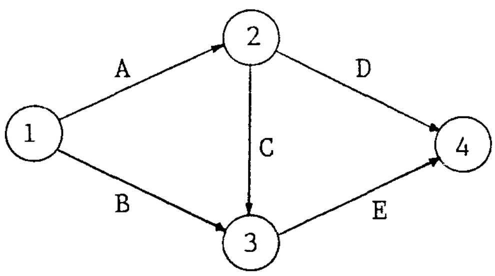


Figure 1.1

In addition, in the conventional PERT network models it is assumed that the completion time distributions of individual activities are statistically independent. In practice there may be dependence between activities. Conditions which affect one activity, causing it to have a rapid completion time or a slow completion time, quite often affect other activities as well. Further, many managers will attempt to improve performance by switching manpower and resources to speed activities which are lagging behind schedule and to take advantage of activities which are ahead of schedule.

Most of the approaches to PERT network analysis proposed to date make the assumption that the duration times of activities have statistically independent distributions, an important possible source of error.

Structural dependence relationships can interact with statistical dependence relationships and produce important effects which cannot be detected and understood using simple expected value calculations. These effects are complex, but they can be identified, modelled and understood in a Controlled Interval and Memory (CIM) framework (Chapman and Cooper, 1983a).

The simplest CIM model is a histogram representation of probability distribution using a common interval or class width, as used for basic descriptive statistics (Driscoll, 1980). The addition of two such distributions using a CI approach produces a result distribution with intervals of the same equal width. Its computation procedure can be viewed as a special case of Discrete Probability Distribution (DPD) approach (ref. Kaplan, 1981).The Controlled Interval and Memory (CIM) approach is the generalized version of the Common Interval approach.

Flexibility of computation operations and flexibility in specifying dependence structures are the two main advantages of the CIM approach over functional integration, numerical integration or moment-based approaches. Monte Carlo approaches have advantages relative to the CI approach if complex non-sequential distribution combination patterns are involved, as in some PERT networks, otherwise a CIM approach provides much greater precision for similar computational effort. The DPD approach has some comparative advantages if precision is not very important, statistical dependence is not present and a basic distribution specification is acceptable, otherwise the CIM framework is preferable.

This dissertation presents a method for taking into account dependence between activities in PERT networks and provides a generalized algorithm to evaluate the project completion time and criticality index of each activity and path using a CIM approach. To accomplish this, we consider different activity network models with different statistical and structural dependence between distributions, and we employ standard CIM calculation procedure which retains a memory of structural links for later use. For example, in the PERT activity network of Figure 1.1 a greatest operation on B and (A plus C), followed by the addition of E, determines the finishing distribution for E. Adding A plus D determines the finishing distribution for D. However, a greatest operation when merging these two paths to determine the project finishing distribution must consider their joint dependence upon A. This requires a memory of A through both paths, allowing the final merge to proceed initially in terms of distributions which are conditional upon the duration of A, then 'forgetting' the A memory by using the probabilities of specific A durations to remove the condition.

The above example could be modelled in a Monte Carlo Simulation framework, but CIM models are comparatively easy to understand and use for simple networks. Moreover, for the same level of precision, Monte Carlo Simulation requires much greater computation effort than the CIM method unless 'memory' dimensions become numerous. A complex PERT network involving several thousand activities, for example, is better handled via simulation.

The retention of a single memory dimension for A of Figure 1.1 involves preserving and working with a matrix of probabilities at each stage. This involves a slight increase in computational effort. In general, n levels of memory involve (n+1) dimensional probability matrices. In order to minimize the memory limitations of a CIM approach this dissertation presents a procedure which solves the network for various conditional values of common activities and then deconditions conditional probability distribution functions.

A generalized form of the proposed procedure determines the probability distribution function of project completion time and a criticality index for all activities and paths. It allows activity durations to have continuous distribution or any distribution function presented in a finite set of ordered pairs. It also allows statistical dependence between activities.

The proposed procedure provides an exact pdf for project completion time when the duration times of activities are discrete and approximates the pdf of the project when the duration times of activities are continuous. Approximation is due to:

(i) discretizing continuous distributions,

(ii) convoluting discrete approximations to continuous distribution.

The structure of this thesis is as follows.

Chapter 2 presents a brief discussion of studies concerned with stochastic PERT network completion times.

Chapter 3 introduces the proposed basic procedure for project completion times.

Chapter 4 introduces the proposed procedure and the algorithm.

Chapter 5 considers other methods relevant to the proposed procedure.

Chapter 6 introduces the proposed procedure for criticality indices.

Chapter 7 deals with PERT networks with structural and statistical dependence relationships.

Chapter 8 presents a brief discussion of discretizing methods.

Chapter 9 considers convoluting discrete approximations within the CIM framework.

Chapter 10 presents a brief discussion of solving stochastic PERT networks using Monte Carlo methods.

Chapter 11 provides a summary and conclusion.

CHAPTER 2: REVIEW OF THE LITERATURE

INTRODUCTION

A number of industrial and management problems have been successfully solved with the aid of quantitative models and techniques based on networks. Such problems include:constructing a dam; determining the shortest or most economical shipping route between two locations; developing an aircraft; planning, scheduling, and controlling the building of a large military weapon system; determining the maximum flow and optimal expansion policies for a gas pipeline system; implementing a new computer system; designing, introducing, and marketing a new product.

The planning, management, and control of projects is a problem area that has been aided by network based techniques, especially CPM (Critical Path Method) and PERT (Program Evaluation and Review Technique). CPM and PERT are the two primary project management network techniques used today.

CPM uses only a single estimate of activity times and does not consider the effects of uncertainty in the activity time. CPM is often used as basis for considering the trade-off between a project's cost and completion date. In this context it focuses on shortening the duration of task or activity time by utilizing more resources, balancing the increased costs against savings associated with project overheads or completion opportunity costs.

PERT was developed in the late 1950s independently of CPM although it is very similar in many respects. It was used extensively in managing military research and development projects. Its first application was the Polaris missile project for the U.S. Navy. Since then, PERT has been successfully used in the construction industry, particularly in the building of large structures. Examples of how it can be applied to the manufacturing function of a firm include: the scheduling of aircraft maintenance, the design, development, and testing of new machines; first production runs; installing fixed assets; and plant lay-out.

In many projects, especially research and development projects, the time durations for various activities are known only with a high degree of uncertainty. It was to cope with this aspect of network planning that the PERT system was created. The approach considers arc lengths (activity durations) as random variables with known distributions. This approach gives rise to two important problems. One is determining the distributions for arcs. The other is solving the model, finding something that corresponds to the project duration and critical path in the deterministic case.

The theoretical problems associated with PERT have been discussed extensively in the literature. MacCrimmon and Ryavec (1964) discuss assumptions that induce bias in the PERT completion time estimate. Among these assumptions are: Beta distributed activity times, the methods of calculation for the mean and variance of activity times, activity independence and the normality assumption for path completion time. Considerable research has been devoted to the estimation of activity duration times. An interesting attempt was reported by Kidd

(1975).

In the next section, the PERT model is characterized and the relevant terms are defined.

In the third section, the solution technique now used is analysed and the important assumptions emphasised.

Proposed methods for solving stochastic PERT problem is the subject matter of the fourth section.

The final section provides a summary and conclusions.

THE PERT MODEL

A PERT network is a connected, directed, acyclic graph, G(N,A), composed of N nodes designated events and A arcs designated activities. Associated with each activity is a non-negative random variable called its duration. All activities leading into a node must be completed before activities leading out of that node can be started. An event is said to have occurred when all activities leading into the node representing that event have been completed.

There are two special events, the initial event (node 1) and the terminal event (node N). No activity leads into the initial node or out of the terminal node. The nodes can be numbered such that an arrow leads from a smaller numbered node to a larger one.

It is usually assumed that activities' duration are independently distributed, each in a finite range. A path is defined as a sequence of activities from the initial node to some other node. A realization of the network is the network with a fixed value for each of its activity durations. For a particular realization of the network, the longest path from the initial to the terminal node is called the "critical path", its length the project duration and the activities on it "critical activities". Any delay in a critical activity will obviously cause a corresponding delay in the entire project.

SOLVING THE PERT MODEL: INITIAL BASIC APPROACH

For the usual theoretical solution of PERT networks, two kinds of standard assumptions are made.

(i) Assumptions that are relevant to the individual activities.

(ii) Assumptions that are relevant to PERT network as a whole.

When the duration times of activities are random variables, denoted by Y_{ij} for activity (ij), the standard approach proposed by the originators of PERT (Malcolm et al, 1959) is simply to replace activity durations with deterministic equivalents and aggregate these to identify the longest path(s) through a network.

The deterministic equivalent is chosen as follows.

Three time estimates, an optimistic, pessimistic and most likely activity duration (a_{ij}, b_{ij}, and m_{ij}), are obtained from the manager/engineer of an activity. From the family of Beta distributions, with these characteristics, the variance for each activity time is assumed to be,

$$\sigma^2_{ij} = (b_{ij} - a_{ij})^2/36;$$

and the mean is approximated by

$$\bar{Y}_{ij} = (a_{ij} + 4m_{ij}+b_{ij})/6 \cdot$$

Let T_c denote the critical path(s). Assume that:

1 - The activities are independent.

2 - The critical path has enough activities so that central limit theorem applies.

3 - The critical path is enough longer than any other path so that the probability of a realization having a different critical path is negligible.

The length of the critical path is itself a random variable, being the sum of random variables, and is identical with the time of realization of node N, denoted by T_N.

$$T_N = \sum_{(ij)\epsilon T_c} Y_{ij} = Z\ (T_c) \qquad (2.1)$$

The above first two assumptions lead to the conclusion that T_N is approximately Normally distributed with mean

$$g_N \cong \sum_{(ij)\epsilon T_c} E(Y_{ij}) = \sum_{(ij)\epsilon T_c} \bar{Y}_{ij} \ ;$$

and variance (2.2)

$$\sigma^2_N = \sum_{(ij)\epsilon T_c} \sigma^2_{ij} \ ;$$

then the probability that event N will occur on or before a specific time t is given by

$$\Pr\left\{ T_N \leqslant t \right\} = \Phi\left\{\frac{t - g_N}{\sigma_N}\right\} \cdot \quad (2.3)$$

Since only the mean and variance of the distribution are used in current calculation method, the assumption of Beta distribution is not fundamental to the subsequent analysis. This assumption may be "logical" and highly convenient, especially under the stipulation of the three time estimates, a, m, and b and the approximations based on them, but it seems that in some instances it would be equally "logical" to assume other forms of df.

The possible errors in the individual activities could, by themselves, cause errors in the calculation of a project mean and variance, although the extent and direction of these errors might be difficult to determine. However, even if the data (i.e. the mean, variance, and distribution) that PERT obtains for each activity are correct, significant error can still be introduced into the calculation of a network mean and variance. As a result, probabilty statements concerning the various completion times of a project can also be incorrect.

Consider the question of probability statements attached to the realization time of an event, say the last node N, in the case where several paths T_1, T_2, ..., T_r lead from the origin, node 1, to node N, as shown in Figure 2.1.

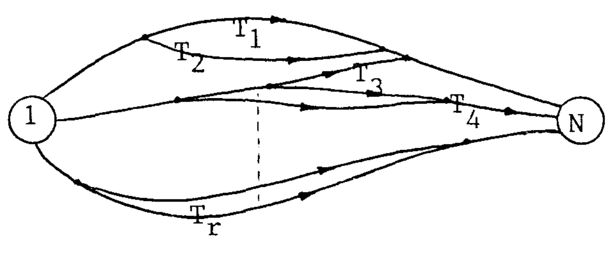


Figure 2.1

Let $Z(T_1)$ denote the duration of path $T_1 \epsilon P$; then the earliest realization time of node N is given by

$$T_N = \max_{T_1 \epsilon P} \left\{ Z(T_1) \right\} \qquad (2.4)$$

or $\quad T_N = \max \left\{ Z(T_1),\ Z(T_2),\ \ldots\ldots ,\ Z(T_r) \right\}$; r=number of paths to N

The paths (T_k) are not independent because they usually share activities. Even if they are considered "approximately" independent, their duration need not be Normally distributed. Even if the duration of each path is Normally distributed, T_N which is the maximum of a finite set of random variables is not Normally distributed. In fact, under the assumption of independence, the pdf of T_N is the product of the individual pdfs;

$$\begin{aligned} \Pr\left[T_N \leqslant t\right] &= \Pr\left[\max \left\{Z(T_1),\ Z(T_2),\ \ldots ,\ Z\ (T_r)\right\} \leqslant t\right] \\ &= \Pr\left[Z(T_1) \leqslant t;\ Z(T_2) \leqslant t\ ; \ldots\ Z(T_r) \leqslant t\ \right] \qquad (2.5) \\ &= \prod_{k=1}^{r} \Pr\left[\ Z(T_k) \leqslant t\right], \text{ by independence .} \end{aligned}$$

Finally, even if the pdf of T_N is approximated by a Normal distribution, it would be a Normal df with a different mean and different variance from that suggested by this initial PERT approach.

Consequently, the probability statements made according to the basic PERT procedure are clearly subject to serious shortcomings. Much of the PERT literature has been concerned with approaches to overcome these shortcomings. These approaches can be classified as analytical, approximation or simulation methods. The next section presents a brief discussion of proposed methods for solving the stochastic PERT problem.

Before presenting proposed methods the basic PERT procedure is illustrated by an Example.

Example 2.1 Consider the PERT network of Figure 2.2 with three time estimate (optimistic:a, most likely:m, and pessimistic:b) as shown beside each activity.

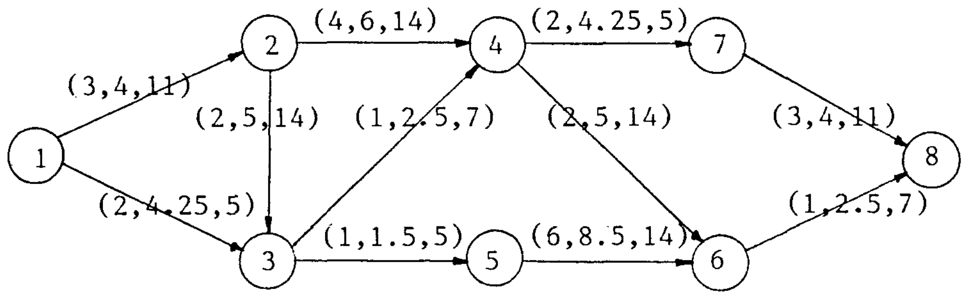


Figure 2.2: Project Network with three time estimate.

Let $\bar{Y}_{ij}$ denote the mean time of activity (ij). Using the following equations, the mean time and variance for each activity can be computed.

$$\bar{Y}_{ij} = (a_{ij}+4m_{ij}+b_{ij})/6 \quad ;$$
$$\sigma^2_{ij} = (b_{ij}-a_{ij})^2/36 \ .$$

Table 2.1 shows these values.

Table 2.1: Mean time and Variance for the activities of PERT network of Figure 2.2.

Activity	Mean Time ($\bar{Y}_{ij}$) (in months)	Variance (σ^2_{ij})
1,2	5	1.78
1,3	4	0.25
2,3	6	4
2,4	7	2.78
3,4	3	1
3,5	2	0.44
4,6	6	4
4,7	4	0.25
5,6	9	1.78
6,8	3	1
7,8	5	1.78

Let,

ES(ij) = Earliest Start time for activity (ij).

EF(ij) = Earliest Finish time for activity (ij).

E_j = Earliest occurrence time for event j.

The following expression can be used to find the earliest finish time for activity (ij)

$$EF(ij) = ES(ij) + \bar{Y}_{ij} \quad (2.6)$$

It is usually assumed that projects start at time zero. Therefore, letting ES(1,2) = ES (1,3) = 0, and given $\bar{Y}_{1,2} = 5$ and $\bar{Y}_{1,3} = 4$, the earliest finish time for activity (1,2) and (1,3) can be computed by using expression (2.6) as follows:

$$EF(1,2) = 0+5 = 5 \quad \text{and} \quad EF(1,3) = 0+4 = 4 .$$

Given $E_1 = 0$, the earliest occurrence time of event j, j= 2,...., N can be computed successively using the following expression, where j is the succeeding event number, and i<j

$$E_j = \max_i \ (E_i + \bar{Y}_{ij}) \cdot \qquad (2.7)$$

Recall that an event does not occur until all activities leading into the node representing that event have been completed. Activities leaving a node cannot be started until all preceding activities have been completed and the event has occurred. Therefore, the earliest start time for activities leaving a particular node is equal to the maximum of the earliest finish times for all activities entering the node which is equal to the earliest occurrence time of that event. Thus, with, i<j<k,

$$ES(jk) = \max_i EF(ij) = E_j \cdot \qquad (2.8)$$

Using expressions (2.6),(2.8) and (2.7), Earliest Finish time and Earliest Start time for each activity and Earliest occurence time for each node can be computed.

For example,

$$ES(2,3) = EF(1,2) = 5 ,$$

$$ES(2,4) = EF(1,2) = 5 ,$$

$$EF(2,3) = ES(2,3) + \bar{Y}_{2,3} = 5 + 6 = 11 ,$$

$$ES(3,4) = \max \{EF(1,3), EF(2,3)\}$$

$$= \max \{4,11\} = 11 .$$

Proceeding in a forward pass through the network, we can find the earliest start and earliest finish time for each activity. These values are shown in Table 2.2

Given $E_1 = 0$, by using expression (2.7).

$$E_2 = E_1 + \bar{Y}_{1,2} = 0 + 5 = 5 ,$$

$$E_3 = \max \{(E_1 + \bar{Y}_{1,3}),(E_2 + \bar{Y}_{2,3})\}$$

$$= \max \{(0 + 4), (5 + 6)\} = \max \{4,11\} = 11 .$$

Proceeding in a forward pass through the network we can find the earliest occurrence time for each node. These values are shown in Table 2.3.

As shown in Table 2.3 the earliest occurrence time for the terminal event (node 8) is 25. Thus the earliest completion time for the entire project is 25 months. The algorithm for finding the critical path then involves a backward pass calculation.

Let,

LS(ij) = Latest Start time for activity (ij).

LF(ij) = Latest Finish time for activity (ij).

L_j = Latest occurrence time for event j.

The following expression can be used to find the start time for activity (ij),

$$LS(ij) = LF(ij) - \bar{Y}_{ij} \cdot \qquad (2.9)$$

Letting $L_N = E_N$ and using the following expression, the latest occurrence time for events N-1, N-2,....,1 can be computed.

$$L_i = \min_j (L_j - \bar{Y}_{ij}) , \qquad (2.10)$$

and also, the latest finish time for an activity entering a particular node can be computed using the following expression.

$$LF(1i) = L_i = \min_j \ LS(ij) \cdot \qquad (2.11)$$

Given $L_8 = E_8 = 25$, by using expressions (2.9), (2.11) and (2.10), Latest Start time and Latest Finish time for each activity can be computed as follows:

$$LF(7,8) = E_8 = 25 ,$$

$$LF(6,8) = E_8 = 25 ,$$

$$LS(7,8) = LF(7,8) - \bar{Y}_{7,8} = 25 - 5 = 20 ,$$

$$LS(6,8) = LF(6,8) - \bar{Y}_{6,7} = 25 - 3 = 22 .$$

Proceeding in a backward pass through the network, we can find the, Latest Finish and Latest Start time for each activity as shown in Table 2.3.

Also, $L_7 = L_8 - \bar{Y}_{7,8} = 25 - 5 = 20$,

$$L_6 = L_8 - \bar{Y}_{6,8} = 25 - 3 = 22 ,$$

$$L_4 = \min \{(L_7 - \bar{Y}_{4,7}),(L_6 - \bar{Y}_{4,6})\}$$
$$= \min \{(20 - 4),(22 - 6)\} = \min \{(16),(16)\} = 16 \cdot$$

Proceeding in a backward pass through the network, we can find the Latest occurrence time for each node. These values are shown in Table 2.3.

By comparing the earliest start time and the latest start time (or earliest finish time and latest finish time) for each activity, we can find the amount of float associated with each of the activities. Float is defined as the length of time an activity can be delayed without affecting the completion date for the porject. Float for activity (ij) is defined by the equation:

$$F_{ij} = LS(ij) - ES(ij) = LF(ij) - EF(ij) . \quad (2.12)$$

Table 2.2: Calculation of Activity times.

Activity	ES(ij)	LS(ij)	EF(ij)	LF(ij)	F_{ij}
1,2	0	0	5	5	0
1,3	0	7	4	11	4
2,3	5	5	11	11	0
2,4	5	9	12	16	4
3,4	11	13	14	16	3
3,5	11	11	13	13	0
4,6	14	16	20	22	2
4,7	14	16	18	20	2
5,6	13	13	22	22	0
6,8	22	22	25	25	0
7,8	18	20	23	25	2

From a practical point of view, float means more time to work, less to worry about, and a chance to transfer men, machinery or supervision to an activity that lies on the critical path.

In general, the critical path activities are the activities with zero float. A critical activity cannot be delayed without affecting the entire project. In other words, if we fall behind on the critical path, the project completion time will fall behind to the same extent.

For all events on the critical path $E_i = L_i$, and on all other non-critical path $E_i < L_i$. The difference , S_i, is called the slack for event i. Slack for each event of Figure 2.2 is shown in Table 2.3. As shown in Table 2.2, float for activities (1,2), (2,3), (3,5), (5,6) and (6,8) is equal to zero. Therefore, these activities are critical,and following path which consists of critical activities is the critical path. Critical path ≡ 1-2-3-5-6-8 .

Table 2.3: Calculation of Event Times.

Event	Earliest occurrence time E_i	Latest occurence time L_i	Event Slack S_i
1	0	0	0
2	5	5	0
3	11	11	0
4	14	16	2
5	13	13	0
6	22	22	0
7	18	20	2
8	25	25	0

Notice that only two events have any slack: events 4 and 7. They have a combined total of 4 months of slack. It means in practice we could fall 4 months behind somewhere on these two events and not interfere with completion of the project on time, at its earliest expected date.

The PERT model uses sum of the mean times of activities on the critical path as the mean of the distribution of project completion time and sum of the variances of activities on the critical path as the variance of the distribution of project completion time. Therfore, by using the following equations the mean and variance of the project completion time can be computed

$$g_N = \sum_{(ij)\epsilon T_c} E(Y_{ij}) = \sum_{(ij)\epsilon T_c} \bar{Y}_{ij} \; ; \qquad (2.2)$$

$$\sigma_N^2 = \sum_{(ij)\epsilon T_c} \sigma_{ij}^2 \; .$$

Mean: $g_8 = 5+6+2+9+3 = 25$

Variance: $\sigma_8^2 = 1.78+4+0.44+1.78+1 = 9$

Under the PERT assumptions, the project completion time is normally distributed with a mean of 25 and a standard deviation of 3.

Now let us look at PERT assumptions.

1 - The activities are independent.

We could switch resources from activities which are ahead of schedule to the activities which are behind schedule or to the critical path. Moreover, in a project when an activity takes a long time we could speed up the following activity, at additional cost. Therefore, this assumption is unrealistic.

2 - The critical path has enough activities so that central limit theorem applies. Use of the central limit theorem for large network is probably not a serious source of error given independence, but for small network of Figure 2.2 the central limit theorem does not hold even with independence and the distribution of project completion time will not be approximately normal.

3 - The critical path is enough longer than any other path so that probability of a realization having a different critical path is negligible.

Thisassumption is not true for the PERT network of Figure 2.2. This network consists of 7 different paths as shown in Table 2.4.

Table 2.4: Different paths of project network of Figure 2.2

Path	Mean of the path length	Variance of the path length
1-2-4-7-8	5+7+4+5 = 21	1.78+2.78+0.25+1.78 = 6.59
1-2-4-6-8	5+7+6+3 = 21	1.78+2.78+4+1 = 9.56
1-2-3-4-6-8	5+6+3+6+3 = 23	1.78+4+1+4+1 = 11.78
1-2-3-5-6-8	5+6+2+9+3 = 25	1.78+4+0.44+1.78+1 = 9
1-3-5-6-8	4+2+9+3 = 18	0.25+0.44 + 1.78+1 = 3.47
1-3-4-7-8	4+3+4+5 = 16	0.25+1+0.25+1.78 = 3.28
1-3-4-6-8	4+3+6+3 = 16	0.25+1+4+1 = 6.25

All of these paths could be critical. As an extreme case, consider path (1-3-4-7-8). Although its mean time is 16 and none of the activities along this path are critical, when all activities on this path accomplished in the pessimistic time (b), then the length of this path is 28, whereas the length of the "critical path" was found to be 25. Since all the other paths also could be critical, even if each has a small probability of being critical, the error can be accumulate significantly.

The deviation of the basic PERT calculation of the mean from the actual mean depends upon the following factors:

1 - Number of paths merging in terminal node that may become critical.

2 - "Closeness" of the expected completion times of the paths.

3 - The variance of the path lengths.

4 - The correlation of paths, i.e, the number of common activities between paths.

Effects of these factors in the project completion time will be described in more detail in the following chapters.

PROPOSED METHODS FOR SOLVING PERT MODELS

Proposed methods for solving the stochastic PERT problem have usually followed one of three main approaches: analytic, approximation, or Monte Carlo simulation. The research literature on this problem can be classified into several groups.

The first group is concerned with approximating the mean completion time. These approximations involve the manipulation of a fixed number of time values and the corresponding probabilities. Frequently quoted are Fulkerson (1962) and Elmaghraby (1967).

The second group is also concerned with approximations. The computation involves the manipulation of distributions parameters. This approach usually assumes a Normal distribution for individual activities. Examples include Clark(1961) and Sculi (1983).

The third group is concerned with the estimation of the distribution function of the project completion time, including:

(i) Determination of the exact probability distribution function (pdf) of networks with special activity configurations.

(ii) Approximating the pdf.

(iii) Bounding the pdf.

Examples include,

(i) :Martin (1965), Hartely and Wortham (1966), Ringer (1966), Burt and Garman (1971b).

(ii) :Dodin (1985b).

(iii) :Robillard and Trahan (1977).

The fourth group uses simulation to estimate the completion time distribution. Examples can be found in Van Slyke (1963) and Cook and Jennings (1979).

In order to illustrate the problem we return to a more detailed discussion of the PERT model.

Consider the PERT network of Figure 2.3. Assume the nodes are numbered such that an arc leads from a smaller numbered node to a higher one.

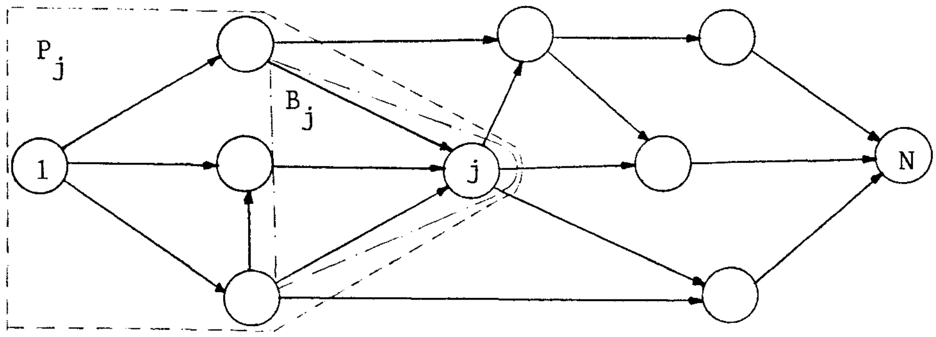


Figure 2.3

Let P_j denote the subnetwork of nodes and arcs up to and including node j as shown in the Figure.

Let Y_{ij} denote the duration time of activity $(ij) \epsilon A$, and $T_N(A)$ denote the critical path given any realization of the network.

(By definition; a realization of a network is a result of a random experiment that assigns a duration y_{ij} to each arc $(ij) \epsilon A$, where y_{ij} is chosen according to the pdf of the random variable Y_{ij} (Elmaghraby, 1977)).

Finally, let $p[y(A)]$ denote the probability of the realization of the arcs in the network where $y(A)$ = vector of realization of all arcs in the network. When Y_{ij} is discretely distributed for all $(ij) \epsilon A$, the project completion time (T_N) is a random variable whose expected value is given by

$$E(T_N) \stackrel{\Delta}{=} e_N = \sum_{y(A)} T_N(A).p[y(A)] \qquad (2.13)$$

The PERT model estimates the expected duration of the project by defining the function g_j recursively as follows:

$$g_1 \equiv 0 \ ,$$

$$g_j = \max_i \{g_i + \bar{Y}_{ij}\} \cdot \qquad j = 2,3 \ldots\ldots, N \qquad (2.14)$$

Note that j is the succeeding event number, and i<j, where

$$\bar{Y}_{ij} = \sum_{y_{ij}} y_{ij} \cdot p(y_{ij})$$

and $p(y_{ij})$ is the marginal probability function of arc (ij).
In the case of a continuous random variable, appropriate changes such as replacing Σ by $\int$ and p(.) by f(.) are made.

Computation of (2.14) is an easy task, even for large PERT networks having thousands of arcs. However, we assert that expected value obtained using PERT model is biased optimistically: the enequalities

$$g_j \leqslant e_j \qquad (j = 2,3 \ldots\ldots, N)$$

always hold, where e_j = expected value of critical path to node j given any realization of the subnetwork P_j j = 2,3 ...., N.

In other words, the maximum value obtained based on substituting the expected values for the duration time of activities along different paths to node j, (j = 2,3 ..... N) as it is used in the basic PERT model is not larger than the expected value of the maximum of the durations of the paths to node j. This assertion follows from a result based on Jensen's inequality (Rao, 1973 p.58).
Proposition: If X_1 ,..., X_s are random variables, then

$$E_{X_1,\ldots, X_s} [\max_j\{f_j(X_1,\ldots,X_s)\}] \geqslant \max_j\{E_{X_1,\ldots,X_s}[f_j \ (X_1,\ldots,X_s)]\}$$

where $E_{X_1,\ldots, X_s}$ is the mathematical expectation involving random variables X_1 ,.., X_s and f_j is a function defined on those random variables.
Before proceeding further, we illustrate (2.13) and (2.14) by means of a small numerical example.

Example 2.2

Consider the network of Figure 2.4 with possible durations shown beside each activity. Assume all time durations are independent and equally probable. Since this network consists of 4 activities and each activity can take 2 values, to evaluate its completion time $2^4 = 16$ deterministic problems each with probability of occurence equal to 1/16 need to be solved.

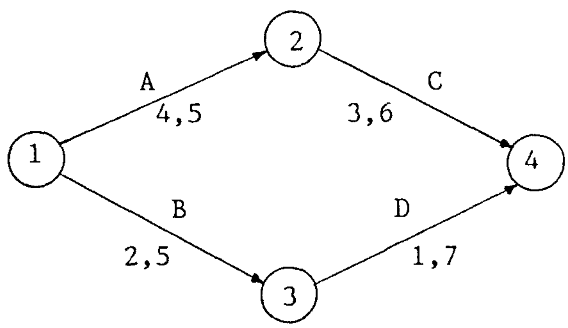


Figure 2.4

Table 2.5 shows all 16 realization times of the project. For each realization time, the critical path is determined.

Table 2.5: Possible realization times of the project of Figure 2.4.

Path (1-2-4)	Path (1-3-4)	Length of Critical Path
I Duration time of (A+C)	II Duration time of (B+D)	III max (I,II)
4+3 = 7	2+1 = 3	max {7,3} = 7
4+3 = 7	2+7 = 9	max {7,9} = 9
4+6 = 10	2+1 = 3	max {10,3} = 10
4+6 = 10	2+7 = 9	max {10,9} = 10
4+3 = 7	5+1 = 6	max {7+6} = 7
4+3 = 7	5+7 = 12	max {7,12} = 12
4+6 = 10	5+1 = 6	max {10,6} = 10
4+6 = 10	5+7 = 12	max {10,12} = 12
5+3 = 8	2+1 = 3	max {8,3} = 8
5+3 = 8	2+7 = 9	max {8,9} = 9
5+6 = 11	2+1 = 3	max {11,3} = 11
5+6 = 11	2+7 = 9	max {11,9} = 11
5+3 = 8	5+1 = 6	max {8,6} = 8
5+3 = 8	5+7 = 12	max {8,12} = 12
5+6 = 11	5+1 = 6	max {11,6} = 11
5+6 = 11	5+7 = 12	max {11,12} = 12

By using equation (2.13),the expected value of project completion time can be found:

$$E(T_N) \triangleq e_N = \sum_{y(A)} T_N\ (A).p[y(A)]$$

$$e_4 = \frac{7+9+10+10+7+12+10+12+\ 8+9+11+11+8+12+11+12}{16} = 9.937$$

e_4 is the actual value.

Now we use (2.14) to obtain,

$g_1 \equiv 0$,

$g_2 = g_1 + \frac{1}{2}(4+5) = 0+4.5 = 4.5$

$g_3 = g_1 + \frac{1}{2}(2+5) = 0+3.5 = 3.5$

$g_4 = \max(g_2+\frac{1}{2}(3+6),\ g_3+\frac{1}{2}\ (1+7))$

$= \max(4.5+4.5, 3.5+4) = \max\ (9,7.5) = 9$

Indeed $g_4 = 9 < e_4 = 9.937$.

In this example the PERT model estimate (g_4) is approximatley 10% optimistically biased. In many situations, this PERT estimate may be very far from the real value. However, this lower bound can be improved using one of the following approaches.

APPROXIMATING THE PROJECT COMPLETION TIME PARAMETERS

(a) Approaches based on approximating activity pdf's with simpler pdf's.
(b) Approaches based on moments.

Approaches based on approximating activity pdf's with simpler pdf's.

One alternative to improve the PERT estimate is to approximate the given activity duration time pdf's with simpler pdf's. For example, the pdf can be approximated by single step functions, which reduces the stocastic problem to an ordinary deterministic one.

The first improvement published is due to Fulkerson (1962). He proposed a lower bound that is a function of the variance associated with each arc for the case where the activity durations are discrete random variable.

Fulkerson's estimate is as follows:

Let B_j denote the set of arcs immediately preceding node j, as shown in Figure 2.3.

Determine a function f_j recursively by

$f_1 \equiv 0$,

$$f_j = \sum_{y(B_j)} \max\ \{f_{i_1}+y_1; f_{i_2} + y_2;\ \ldots; f_{i_r}+y_{r_j}\} \times p[y(B_j)];\quad j=2,3,\ldots,n \quad (2.15)$$

where $y(B_j)$ is the vector of realization of all arcs in the set B_j(assumed to contain r_j arcs) and we wrote y_k as a shorthand for y_{i_k}.

In words, f_j is the expected value (over all realization of B_j) of the maximum length to node j.

Note that g_N as given by Equation (2.14) is the maximum of the expected values. It is intiutively clear that averaging over the maxima is never less than the maximum of the average (Elmaghraby, 1977).

Elmaghraby (1967) proposed two approaches. His first approach is based on the following observation :

If all arrows in a directed acyclic network, such as in PERT, are reversed, the average duration of the project e_N remains unchanged.However intermediate values of f_i, $1 < i \leqslant N$, do not necessarily remain the same. Consequently, if we substitute in the expression of f_i the maximum of these two values obtained from the "as given" and the "reversed" subnetwork P_i, we can only improve f_i (although still approaching e_i from below). Proof of this statement concerning the invariance of e_N under reversal of the arrows is given in Elmaghraby (1977 p.244.)

His second approach involves a generalization of the Fulkerson's approach. It is more accurate than Fulkerson's.

Cliengen (1964) has described another extension of Fulkerson's estimate to include the continuous case and derives a computationally feasible form for his estimate.

Robillard and Trahan (1976) also present a generalization of Fulkerson's estimate and demonstrate that it is always at least as accurate as Fulkerson's and it is potentially better than Elmaghraby's estimate.

In this approach the activity durations can be discrete or continuous.

Lindsey (1972) developed a method of calculating the expected completion time based on a model approximating the actual network.

Approaches based on moments

Another approach to improving basic PERT model estimates is based on moments.

Tippett (1925) contributed one of early important works on the distribution of the largest of independent normal random variables with equal means and variances. His results indicate that the expected value of the completion time increases with the number of parallel elements in the network, and with the variance of the elements. In PERT networks, alternate paths contain common elements and are thus correlated. They also have unequal means and variances. Therefore, although Tippett's work gives us some helpful insight into the problem, his tables are not strictly applicable.

Clark (1961) attacked a problem considerably closer to the one posed by PERT network than did Tippett. He assumed that the elements in the network are normal random variables, that the paths have unequal means and variances, that the paths through the network are correlated, and that the distribution of the largest value is approximately normal. Clark's method is very tedious and also an approximation since it is necessary to assume normality at each iteration.

Sculli (1983) reconsidered the problem of using the normal distribution as a representation of activity durations in PERT networks. He provided a method for deriving estimates of means and variances of the earliest event occurrence times. As opposed to (Clark's 1961) approach, it was assumed, additionally, that the activity completion times entering a common event are treated as independent normal random variables.

Kamburouski (1985) presents a method of determining the lower and upper bounds on the mean event occurrence times. His approach employs only a formula for determining the first moment around zero of $\max(X_1, X_2)$, where X_1, X_2 are independent and normal random variables.

EVALUATION OF THE PDF OF PROJECT COMPLETION TIME

Another approach to solving the PERT probem is based upon determination or approximation of the probability distribution function of the project completion time. One advantage of such determination is that we would then be able to not only determine $E(T_N)$, but also give precise probability statement concerning the completion time of the project.

These approaches can be classified as

a) Analytical

b) Monte Carlo Simulation.

<u>Analytical Approaches</u>

Analytical approaches for determination of the exact pdf or its bounds and approximations involve following four steps (Elmaghraby, 1977).

STEP 1. Identification: Identify various subnetworks as special activity configurations whose pdfs are known (the so-called "generic" subnetworks).

STEP 2. Simplification: Replace the various configuration of Step 1 and their associated completion time distributions by single equivalent activities and completion time distributions.

STEP 3. Decomposition: Decompose the simplified network into several subnetworks by separating subnetworks at each cut vertex. A cut vertex is any node such that every path from the origin to the terminal passes through it. Each subnetwork should be a set of parallel paths.

STEP 4. Synthesis: Reduce each subnetwork to a simple equivalent activity, then combine the set of equivalent activities, which are now in series, in a grand equivalnet activity. The result is the completion time of the entire project. The basic idea for these steps is due to Sielken, Hartley and Arseven (1975).

In order to clarify the analytic approach, we start with a completely reducible network to which we can apply the last step.

Consider network of Figure 2.5.

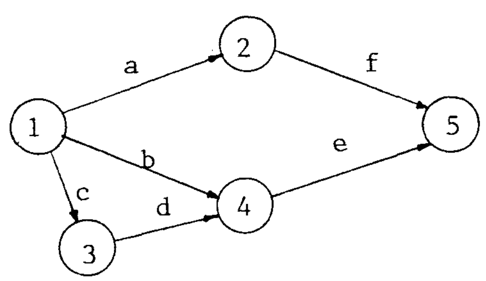


Figure 2.5

This network can be reduced to a single activity through repeated application of the following operations. (i) convolution, and (ii) greatest.

Convolution Operation

Duration times of two activities in series can be added to give rise to a new activity. The distribution function of the duration of the new activity is obtained by convoluting the df's of these activities. This operation is called the convolution operaiton.

Consider two arcs in series, as shown in Figure 2.6. Let Y_a and Y_b denote the duration times of a and b, with pdfs $F_a(t)$, and $F_b(t)$, respectively.

Let c denotes the resultant activity. If activity a starts at time zero, then the realization time of node 3 is a random variable, denoted by Y_c, and given by $Y_c = Y_a+Y_b$. The pdf of Y_c is given by

$$F_c\ (t) \triangleq \Pr[Y_c \leqslant t] = \int_0^t F_a\ (t-y)dF_b(y) = F_a(t-y)*F_b(y) \qquad (2.16)$$

where the asterisk denotes a convolution operation. Note that each convolution operation reduces the number of nodes and the number of arcs each by one: the network gains a new arc, but loses two arcs.

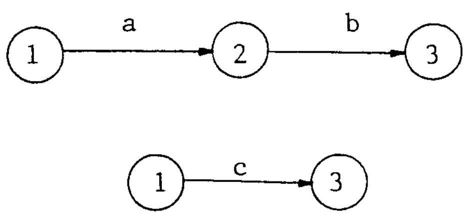


Figure 2.6

Greatest Operation

Taking the maximum duration time of two parallel activities gives rise to a new activity. The distribution function for the duration of the new activity is obtained by multiplying the df's of these activities. This operation is called a greatest or maximum operation.

Consider two arcs in parallel, as shown in Figure 2.7. Let Y_e and Y_f denote the duration times of e and f, with pdfs $F_e(t)$ and $F_f(t)$ respectively. Let g denotes the resultant activity. If node 1 is realized at time zero, then realization time of node 2 is a random variable, denoted by Y_g, and given by $Y_g = \max\{Y_e, Y_f\}$. The pdf of Y_g is given by

$$\begin{aligned} Pr[Y_g \leqslant t] &= Pr[\max(Y_e, Y_f) \leqslant t] \\ &= Pr[Y_e \leqslant t \text{ and } Y_f \leqslant t] \\ &= Pr[Y_e \leqslant t] \times Pr\,[Y_f \leqslant t] \quad \text{assuming independence.} \end{aligned}$$

That is

$$F_g(t) = F_e(t).\ F_f(t) \qquad (2.17)$$

where (.) denotes a greatest operation. Each greatest operation reduces the number of arcs by one: the network gains a new arc but loses two arcs.

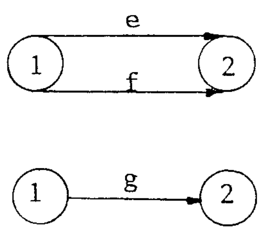


Figure 2.7

Now we return to example network of Figure 2.5

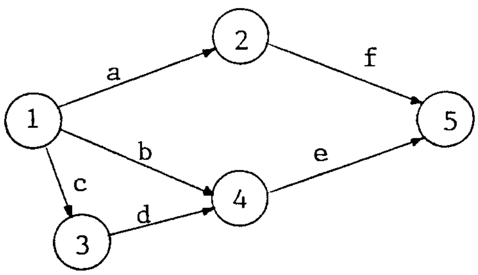


Figure 2.5

Convoluting activities c and d gives rise to a new activity g as shown in Figure 2.8.

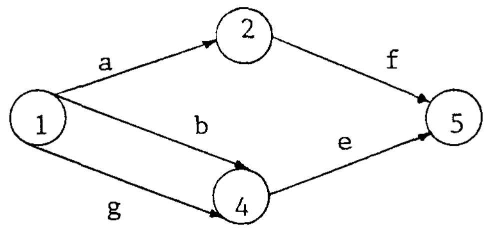


Figure 2.8

A greatest operation for activities b and g gives rise to a new activity h as shown in Figure 2.9.

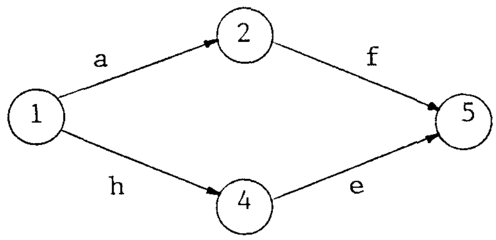


Figure 2.9

Convoluting activities a and f gives rise to a new activity i and convoluting activities h and e gives rise to a new activity j as shown in Figure 2.10.

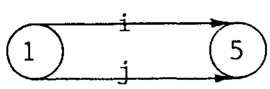


Figure 2.10

Finally, a greatest operation for activities i and j gives rise to a new activity k as shown in Figure 2.11.

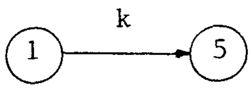


Figure 2.11

For completely reducible networks, the exact value of the pdf is obtained through repeated application of convolution and greatest operations. Now consider the irreducible network of Figure 2.12.

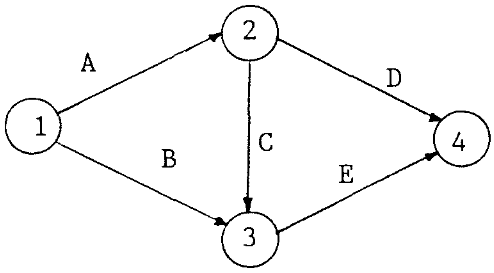


Figure 2.12

A network is called irreducible if we cannot reduce it to a single activity (1,N) through repeated application of convolution and greatest operations. The network configureation of Figure 2.12 is the simplest irreducible network, called a Wheatston bridge or interdictive graph. Dodin (1985c) proved that the network is irreducible if and only if it has the interdictive graph. This proof is provided in Appendix A. The interdictive graph shares with all irreducible networks the following properties.

(1) The number of nodes $N \geqslant 4$, and the number of arcs $A > 4$.

(2) Where I(i) denotes the Indegree of node i (Number of arcs ending at node i) and O(i) denotes the Outdegree of node i (Number of arcs emanating at node i), for each node $i \neq 1$ or N, $I(i)+O(i) \geqslant 3$: there are no arcs in series.

(3) There do not exist two arcs with the same start node and end node, i.e there are no pairs of arcs in parallel.

For irreducible networks, since the paths are correlated because of the common activities, there is no way to combine activities through convolution and greatest operations. In such a case, in order to obtain the pdf of project completon time, the general principle to apply is to evaluate the pdf of the project completion time by conditioning project completion time on the values of common activities, and then remove the conditional nature by integrating over-all values of common activities. The final result is the unconditional pdf of project completion time. For example, in the "Wheatstone bridge" of Figure 2.12 activity, A is common between two paths (1-2-4 and 1-2-3-4) and activity, E is common between two paths (1-2-3-4 and 1-3-4). Let X_i denote duration time of activity i. Since,

$$T_4 = \max \{X_A+X_D, X_A+X_C+X_E, X_B+X_E\}$$

let $Z(T_1) = X_A + X_D$, $Z(T_2) = X_A+X_C+X_E$, $Z(T_3) = X_B+X_E$.

Then, $$F_4(t/X_A,X_E) = F_{Z(T_1)}(t/X_A,X_E)F_{Z(T_2)}(t/X_A,X_E)F_{Z(T_3)}(t/X_A,X_E)$$

and $$F_4(t) = \iint F_{Z(T_1)}(t - X_A).F_{Z(T_2)}(t-X_A-X_E).F_{Z(T_3)}(t - X_E)dF_A(X_A)dF_E(X_E) .$$

Hartley and Wortham (1966) have developed an algorithm for computing the comulative distribution of the completion time for certain PERT networks. These networks (which they term "multiple crossed") are composed of certain simple subnetworks, namely,

(a) two activities in series.

(b) several activities arranged in parallel.

(c) five activities arranged in a "Wheatstone bridge" configuration.

For the precise way in which the above subnetworks are used as building blocks to build up a "multiple crossed" network reference must be made to Hartley and Wortham (1966). However, the use of their algorithm does not necessitate an examination of each network to determine whether it falls into the category of "multiple crossed" the algorithm may be applied to any network and if it happens to be "multiple crossed" the algorithm will automatically compute the df of the project completion time: if, on the other hand, the network is not "multiple crossed" the algorithm will not be able to reduce the network to a single activity. As soon as we find therefore that no reduction of activity and nodes has occurred on two consecutive cycles we would output the reduced network activities and associated cdf's so that it can be solved by a Monte Carlo method or some other algorithm suitable for non-reducible networks.

Ringer (1969) extended the concept of "multiple crossed" networks by adding two additional subnetworks as building blocks to the three mentioned above and developing integral operators for:

(d) The "Double Wheatstone bridge". Figure 2.13

(e) The "Criss-Cross". Figure 2.14

The operations for uncrossed networks and the Wheatstone bridge are given by Hartley and Wortham (1966).

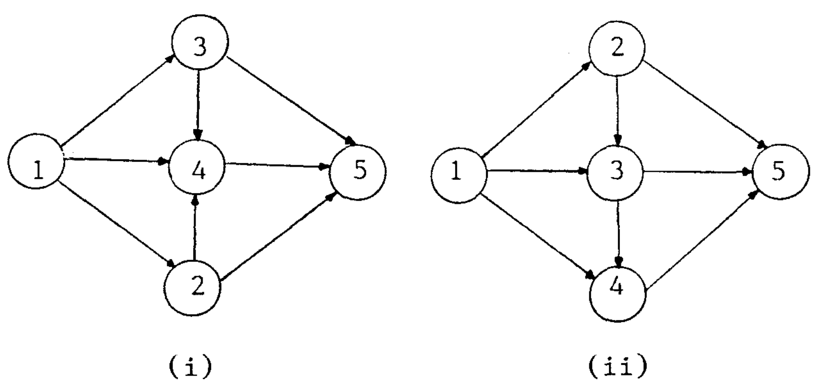


Figure 2.13: Double Wheatstone bridges

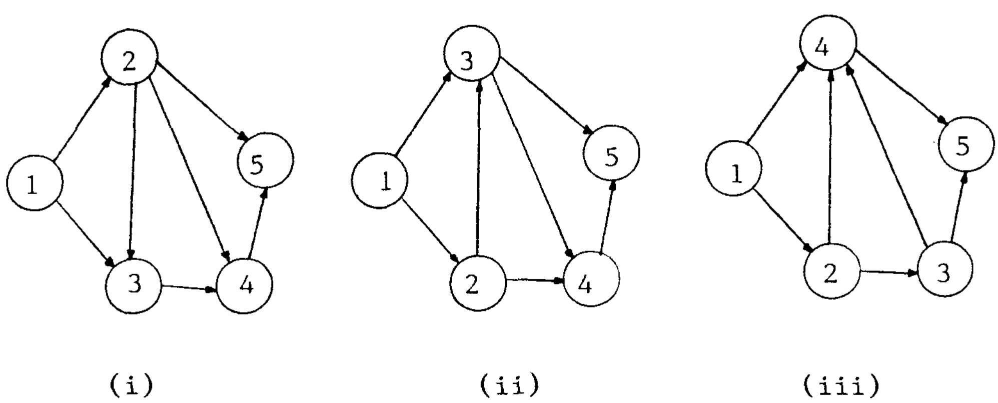


Figure 2.14: Criss-Cross networks

Ringer (1969) also showed that the cdf of project completion time could be expressed in simpler form by conditioning on particular activity times. This concept was also developed independently by Burt and Garman (1971b), who describe and evaluate a technique for performing this multiple integration.

The above discussion should give full meaning to the first two steps of the general procedure. Since the complexity of subnetworks prevents the straightforward step-by-step application of convolution and greatest operations, in the "identification" step various generic subnetwork configurations whose pdfs are known are recognized and the "simplification" step replaces these generic subnetworks by their equivalent activities. In order to illustrate the general procedure we consider the complex network of Figure 2.15.

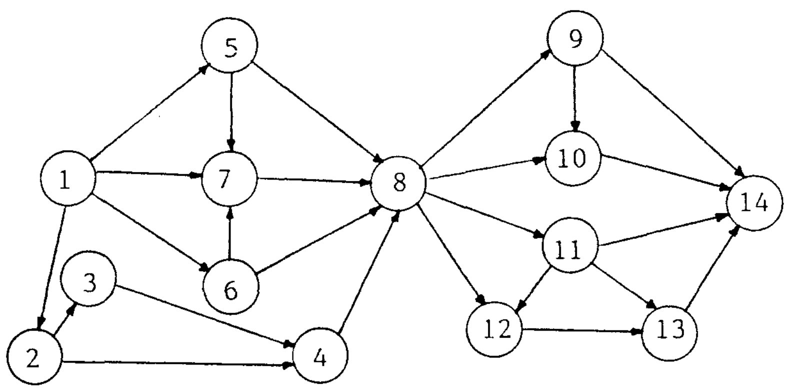


Figure 2.15

Since node 8 is a cut vertex (e.g. the termination event of all activities preceding it) the network is decomposable into two subnetworks as shown in Figure 2.16.

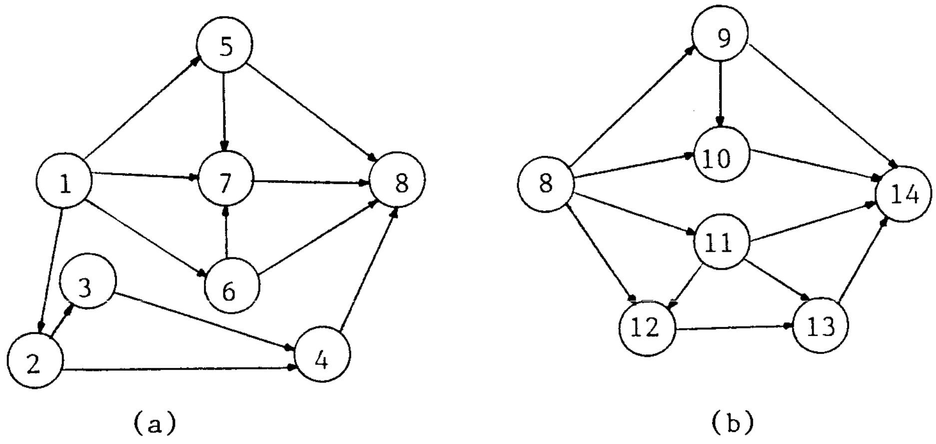


Figure 2.16

The subnetwork of Figure 2.16(a) consists of two subnetworks a Double Wheatstone bridge, {(1,5), (1,6), (1,7), (5,7), (6,7), (5,8), (6,8), (7,8)} and a reducible network, {(1,2), (2,3), (2,4), (3,4), (4,8)}. Let us denote their equivalent activities by A_1 and A_2. The subnetwork of Figure 2.16(b) consists of two subnetworks, a Criss-Cross, {(8,11), (8,12), (11,12), (11,13), (11,14), (12,13), (13,14)} and a Wheatstone bridge, { (8,9), (8,10), (9,10), (9,14), (10,14)}. Let A_3 and A_4 denote their equivalent activities. The original network of Figures 2.15 is now represented by the set of equivalent activities shown in Figure 2.17.

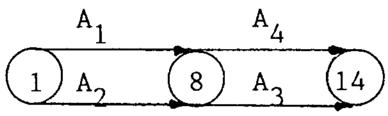


Figure 2.17

This network is completely reducible by using two greatest operations followed by a convolution operation.

A refinement of the above approach was proposed by Martin(1965), who presented a method for the efficient computation of the density function of the project completion time under the assumption that the distribution functions of activity duration are polynomials.

In particular, an algorithm is described that reduces a series - parallel network to a single activity. The specialization of this algorithm to the class of polynomial leads to a detailed examination of the convolution operation for polynomials.

Charnes, Cooper, and Thompson (1964) illustrated their method of "chance constrained and stochastic programming" with exponential cdf's and a network for which cdf of completion time also separates. Burt, Gaver, and Perlas (1970) have investigated the exponential families of cdf's in connection with simple stochastic PERT networks.

Dodin (1985a) presents a practical procedure to approximate the distribution fucntion of the realization time of all events in the network. The approximating procedure consists of three steps: deiscretizing continuous distributions, reducing the network , and sequential approximation. This procedure can be applied to large stochastic networks with any structure, and it allows the activity duration to have a continuous or discrete distributon. This procedure will be discussed in detail in Chapter 5.

The method of bounding distribution is developed in detail by (Kelindorfer, 1971). In practice, this approach concerns only PERT networks where the distribution functions associated with the activities are discrete.

Dodin (1985b) developed a procedure which bounds the exact distribution of the project completion time from below. He believes this bound is tighter than any of the existing lower bounds.

Monte Carlo Methods

Because severe difficulties exist in deriving analytical solutions in PERT networks, many analysts have turned to Monte Carlo methods to derive approximate solution.

Van Slyke (1963) develops the idea of using crude simulation as a tool for finding the cdf of a PERT network's completion time. He also suggests two methods of potentially reducing simulation computation times.

Klingel (1966) used a crude simulation approach to study the direction and magnitude of the errors of the basic PERT solution approach when parallel paths are present in a network.
Antithetic variates, stratification and control variates have been suggested by Burt and Garman (1971b) as ways to reduce the computational effort required in crude simulation. Burt and Garman (1971a) also developed a new simulation procedure called conditional Monte Carlo Simulation in which certain activity times are fixed at their original sampled value thus reducing computation effort and variance.

Min-Max Method

The min-max method was suggested by Van Slyke (1963). This method examines each path in the network twice. The first time the most optimistic time for each activity is assumed and a critical path is identified. The length of this path represents the absolute minimum project completion time. In the second examination, the most pessimistic time is assumed for each network activity and the length of each path is compared to the length of the optimistic critical path. If the length of a path using the pessimistic activity time is less than the length of the optimistic critical path then that path is flagged noncritical. Those paths designated noncritical are disregarded in the ensuing simulation.

Path Deletion Method

The path deletion method, also suggested by Van Slyke (1963), is operationally identical to crude simulation for the first one hundred iterations.

After the first one hundred iterations those paths that had not yet been critical are flagged as non-critical and ignored for the rest of the simulation.

Dynamic Shut-Off Method

The dynamic shut-off method operates exactly like crude simulation except the number of iterations is dynamically controlled. After each one hundred iterations the cumulative density function of project completion time is compared to the cdf from one hundred iterations earlier.

Accuracy and the computation efficiency of the three methods (Min-Max, Path Deletion and Dynamic Shut-off)

Of the three methods (Min-Max, Path Deletion and Dynamic Shut-Off) tested by Cook and Jennings (1979) it was felt that the min-max method would result in the most accurate approximation of the project completion time cdf. The other two methods have a potential of biasing the results because paths that could feasibly be critical paths are deleted from the network. In addition it was felt that the path deletion method would be computationally superior due to the number of activities deleted from the simulation.

Antithetic Variates

Antithetic variates is a widely known technique which is probably the most often and the most easily used variance reduction technique. The basic idea of this approach is as follows. Suppose we have an unknown parameter θ, which we wish to estimate by a statistic t. For instance, in the context of activity networks, θ may be T_N, which we are estimating by the sample duration. We seek another estimation t' having the same expectation as t and a strong negative correlation with t. Then $\frac{1}{2}$ (t+t') will be unbiased in Θ, but its sampling variance is given by

$$Var[\tfrac{1}{2}(t+t')] = \tfrac{1}{4} Var(t) + \tfrac{1}{4} Var(t') + \tfrac{1}{2}Cov(t,t')$$

$$< \frac{1}{2} \text{Var}(t), \text{ because } \text{Cov}(t,t') < 0 \cdot$$

A theorem due to Hammersley and Mauldon (1956) asserts that whenever we have an estimation consisting of a sum of random variables, it is possible to arrange for there to be a strict functional relationship between them such that the estimation remains unbiased, while its variance comes arbitrarily close to the smallest that can be attained with these variables. The basic idea is that we "rearrange" the random variables by permuting finite subintervals, in order to make the sum of the rearranged functions as nearly constant as possible; hence their variance is made as small as possible. If the individual sub-interval sums are exactly a constant, their variance is zero (Elmaghraby, 1977). Detailed discussion of antithetic variates technique is presented in Chapter 10. Sullivan, Hayya and Schaul (1982) investigated the application of antithetic variates technique for the estimation of project completion times in stochastic acyclic networks. They provide a theorem which leads to a guaranteed reduction in variance associated with the application of antithetic sampling in stochastic networks where the arc times are symmetrically distributed about their means. A note by Grant (1983) extends this work by proving that variance reduction is guaranteed for PERT type networks even if the time distributions are not symmetrically distributed.

Sullivan, Hayya, and Schaul (1982) demonstrate that on the average the antithetic variate method can provide the same precision as straight forward Monte Carlo with one fourth the computation time. Furthermore, they note that when activity durations are distributed symmetrically about their means we can expect the antithetic variate method to require less than one tenth the time used by straightforward Monte Carlo.

Control Variates

The basic idea underlying antithetic variates is the generation of a twin random variable that is negatively correlated with the original random variable. Control variates is an alternative approach to generate a random variable that is positively correlated with the original random variable. The control variate technique applies when there is an approximation to the process simulated that can be treated theoretically. By following through the actual and control processes simultaneously, using the same random numbers at corresponding points in the two processes, the difference is observed. This difference is then an estimate of the correction to be applied to the theoretical results of the control process. In order to introduce the concept of control variates in the context of activity networks consider the network of Figure 2.18. Suppose, that all durations of activities are random variables whose distribution functions are amenable to mathematical manipulation except Y_7. Suppose we are interested in estimating project completion time, $E(T_5)$.

It would be very time consuming task to compute the completion time distribution of this network by analytic methods. The control variate procedure requires that we construct a simplifed network "similar" to network of Figure 2.18, such as network of Figure 2.19, whose $E(T_5)$ can be determined analytically, where

$$Y_A = Y_1 + Y_4$$

$$Y_B = Y_1 + Y_3$$

Y_C = a random variable highly correlated with Y_7 but much simpler. The general idea of the control variates technique is to exploit the similarity between Figure 2.18 and 2.19 in order to improve estimates concerning Figure 2.18; in doing so, use is made of the exact knowledge of Figure 2.19. In other words, simulation is used to correct a known result for Figure 2.19 to bring it close to Figure 2.18.

Since $$E(T_5) = E(\hat{T}_5) + E[T_5 - \hat{T}_5]$$

it is possible to estimate $E(T_5)$ by computing $E(\hat{T}_5)$ analytically, and then estimating $E[T_5 - \hat{T}_5]$from a sampling experiment. The latter step involves obtaining K realizations of the eight random variables Y_1, $Y_2, \ldots, Y_8$. Plus the surrogate random variable Y_C. These, in turn, will yield the values $\tau(k)$ and $\hat{\tau}(k)$, $k = 1,2,\ldots,K$, where τ values are the sample realizations of the T values. Averaging, we obtain the estimate of $E[T_5 - \hat{T}_5]$ as given by

$$\sum_{1}^{K} \frac{\tau(k) - \hat{\tau}(k)}{K}$$

Since the arc durations in the two networks share the same random numbers, T_5 and $\hat{T}_5$ are positively correlated. Thus

$$\mathrm{Var}(\bar{T}_5 - \hat{T}_5) = \frac{1}{K} \{\mathrm{Var}\ (T_5) + \mathrm{Var}\ (\hat{T}_5) - 2\mathrm{Cov}(T_5,\hat{T}_5)\}$$

Hence, it follows that if

$$\mathrm{Var}(\hat{T}_5) < 2\ \mathrm{Cov}(T_5,\hat{T}_5)$$

a reduction in variance over crude sampling is achieved. Notice that Figure 2.19 ignores the path containing activity (2,4) of duration Y_5.

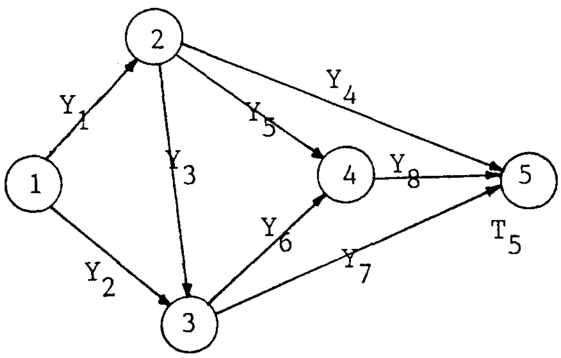


Figure 2.18

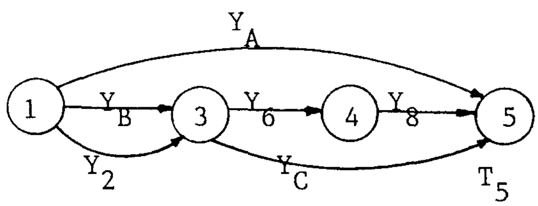


Figure 2.19

Stratified Sampling

In stratified sampling we break the range of the duration of each activity into several intervals, say

$$a_{j-1} \leqslant y_j \leqslant a_j \quad ; \quad j=1,\ldots\ldots,k$$

where $a_0 < a_1 < \ldots\ldots < a_k = M$, and M is the maximum duration of y (or its truncated version). A random variable is chosen from each interval and the analysis proceeds on the basis of the samples drawn.

Hybrid Approaches

Some of the above mentioned approaches may be combined to advantage. For instance, the antithetic approach may be combined with any other approach to yield improved results.

Burt et al (1970) gives some sample results from a combined antithetic variate and control variate simulation of stochastic networks.

Kleijnen (1975), on the other hand, challenges the efficiency of combining variance reduction technique. He notes that while the antithetic variate method generates negative correlations, the control variate method generates positive correlations that tend to offset the negative correlations. The research methodology presented by Sullivan, Hayya and Schaul (1982) can be used to experimentally test the efficiency of combined variance-reduction techniques for simulating stochastic networks.

Conditional Monte Carlo

Conditional Monte Carlo has been shown by Burt and Garman (1971a) to provide improved estimators of the cumulative distribution function (cdf) for stochastic PERT network durations. In their scheme,"unique" activities were identified as those lying upon at most a single path from source to terminus in a given network. The cdf's of the duration of these activities thereupon entered into a network duration cdf estimator in full analytic form; other "nonunique" activities were sampled. One advantage of this approach is that the cdf's of the unique activities appear within the estimator in product form, simplifing computations. Garman (1972) also introduces another conditional sampling method that will, for almost all network, allow the cdf's of many more activities to enter an estimator in analytic form than the conditional Monte Carlo method permits. The approach is based on Martin's (1965) series-parallel reduction of stochastic PERT networks and Hartley and Wortham's (1966) definitions of crossed and multiple-crossed networks. The basic idea is to (a) reduce the network as far as possible by series-parallel reduction; (b) eliminate crossing activities by conditional sampling of their predecessors or successors; (c) repeat (a) and (b) until the network is completely reduced; and (d) use the resulting estimator to estimate the network duration cdf.

SUMMARY AND CONCLUSIONS

It is well known that probability statements made according to the basic PERT procedure are subject to serious shortcomings. Much of the PERT literature has been concerned with approaches to overcoming these shortcomings. These approaches can be classified as analytical, approximation or Monte Carlo Simulation in two area:

1 - Evaluation of the mean project completion time.

These approaches mostly supply lower bounds on the mean value of project completion time and can be classified as:

(a) Approaches involve the manipulation of a fixed number of time values and the corresponding probabilities.

(b) Approaches involve the manipulation of only distribution parameters.

2 - Evaluation of the pdf of project completion time.

The analytic methods evaluate exact pdf of project completion time of special simple networks with simple functional forms for activites' distribution. In general, we cannot expect in a given problem, simple network structure and simple functional forms for activities' distribution. In this case, one alternative is to approximate the pdf of the project completion time by approximating each activity distribution with a simpler one.

In approximation methods, even knowing the accuracy of the individual activity approximations, it is very difficult to predict the accuracy of the pdf of project completion time.

Since analytic methods are unsuitable and approximation methods are not reliable, then there is no choice but to turn to Monte Carlo simulation. Crude Monte Carlo simulation does not require extensive assumptions, but it is very computationally costly for reasonable levels of precision with respect to the resulting distribution.
In order to reduce the computational effort and to enhance precision of crude Monte Carlo method a variety of variance reduction techniques have been proposed. Among these techniques, the Antithetic variates is the easiest to apply and the most widely known.

CHAPTER 3: BASIC CIM MODELS FOR STRUCTURAL DEPENDENCE

INTRODUCTION

This chapter looks briefly at analytical and approximation approaches to combining probability distributions which were not discussed in the previous chapter, discrete probability interval distribution methods and interval or histogram representation methods, to introduce controlled interval and memory (CIM) models. It concentrates on a 'controlled interval' (CI) approach to addition (convolution) and taking the maximum (greatest) of independent probabilistic variables. Finally, CIM treatment of structural dependence in network merge events is considered through an example.

NUMERICAL METHODS

Numerical Integration

Numerical integration approaches are based largely on functional integration results, functional integration being an analytic approach which is used in problem areas like queueing theory. In order to clarify the approach in PERT networks, consider a network with n activities and m complete paths (sequences of activities starting at the beginning and finishing at the end of the project). For the ith complete path the completion time for the path, say u_i, is written as the sum of the appropriate activity completion times. Then, the pdf of the network completion time, F(t), is given by the probability that $u_i \leqslant t$ for all i and is given by

$$F(t) = \int \dots \int dF_1 \dots dF_n$$

where the limits on the integration satisfy $u_i \leqslant t$, $i=1,\dots,m$. In determination of F(t), it is necessary to integrate a product of the distribution functions of activity times. Even if analytic forms for activity time distributions are available, only in rare cases can an analytic solution be obtained for the integral, and dependencies between activity distributions pose a major problem. As mentioned in Chapter 2 the analytic methods tackle above expression by assuming (1) simple functional forms for the dF_j j=1 ,...,n,and (2) special networks for which the mulivariate integral may be seperated into a series of single variate integrals.

Generally speaking, we cannot expect in a given problem to find both simple pdf's and simple network structure. One alternative is to use numerical methods for the evaluation of integration or at the underlying activity times distributions.

If analytic base variable distributions are available, there is a variety of quadrature and finite difference methods for evaluating the convolution integral. However, these do not seem to have been used in PERT problems. A likely explanation for this is due to the multivariate nature of the integrals, their difficult regions of integration, and their computational demands. Therefore, a different approach is needed.

Discrete Probability Distributions

A more flexible numerical approach is to abandon the analytic base variable distribution and to adopt a different representation for it. One such representation is what Kaplan (1981) calls a discrete probability distribution (DPD), or what Winkler and Hays (1970) refer to as a probability mass function.

A DPD is a set of doublets $\{< p_X(x),x >\}$, in which $p_X(x)$ is the proability associated with a particular discrete value x of the base variable X, with $\sum_x p_X(x)=1$. The DPD can be regarded as an approximation to an underlying continous base variable distribution, or it can be considered more directly, and often more usefully, as a specification of our state of knowledge about the base variable.

Combining two base variables in DPD form yields a result which is also a DPD (Kaplan 1981). If X and Y are independent base variables with DPD representations $X = \{ < p_X(x), x > \}$ and $Y = \{ < p_Y(y), y > \}$, then the derived variables $Z = \{ < p_X(x)p_Y(y), f(X,Y) > \}$. This result generalises readily to functions with more than two arguments.

For special case, when Z=X+Y, DPD representation of Z is obtained by convoluting the DPD representation of the base variables using following equation.

$$p_Z(z) = \sum_{x=0}^{z} p_X(x)p_Y(y) \quad (3.1)$$

As an example, assume X and Y have the following DPD representations.

X	$p_X(x)$
2	0.2
3	0.4
4	0.4

Y	$p_Y(y)$
3	0.3
4	0.5
5	0.2

The DPD representation of Z=X+Y can be computed by using equation (3.1) as shown in Table 3.1.

Table 3.1: DPD representation of Z=X+Y .

Z	$p_Z(z)$	
5	0.2 x 0.3	= 0.06
6	0.2 x 0.5 + 0.4 x 0.3	= 0.22
7	0.2 x 0.2 + 0.4 x 0.5 + 0.4 x 0.3	= 0.36
8	0.4 x 0.2 + 0.4 x 0.5	= 0.28
9	0.4 x 0.2	= 0.08

For example, when Z = 5

$$p_Z\ (5) = p_X(2).p_Y(5 - 2)$$
$$= p_X(2).p_Y(3)$$
$$= (0.2)(0.3)$$
$$= 0.06\ ,$$

or, when Z = 6

$$p_Z\ (6) = p_X(2).\ p_Y(6 - 2) + p_X(3).\ p_Y(6 - 3)$$
$$= p_X\ (2).\ p_Y(4) + p_X(3).\ p_Y(3)$$
$$= (0.2)(0.5) + (0.4)(0.3)$$
$$= 0.22\ .$$

If X and Y are in parallel, then the DPD representation of W=max {X,Y} can be computed using following equations.

$$F_W\ (w) = F_X(w).F_Y(w) \qquad (3.2)$$

$p_W\ (w) = F_W(w)-F_W(\bar{w})$ for a $\bar{w}$ slightly less than w.

The calculation process for this greatest operation is as follows.

X	$p_X(x)$	$F_X(x)$=CP(x)
2	0.2	0.2
3	0.4	0.6
4	0.4	1.0

Y	$p_Y(y)$	$F_Y(y)$=CP(y)
3	0.3	0.3
4	0.5	0.8
5	0.2	1.0

where, CP indicates Cumulative Probability, adopting the notations of Chapman and Cooper (1983a) as well as that of Dodin (1980) used earlier. By using equation (3.2), DPD representation of W can be computed as shown in Table 3.2.

Table 3.2: DPD representation of W=max {X,Y}.

W	$p_W(w)$	$F_W(w)$
2	0.0 - 0.0 = 0.0	0.2 x 0.0 = 0.0
3	0.18 - 0.0 = 0.18	0.6 x 0.3 = 0.18
4	0.80 - 0.18 = 0.62	0.8 x 1 = 0.80
5	1 - 0.8 = 0.20	1 x 1 = 1

For example, when W = 2

$$F_W(2) = F_X(2) . F_Y(2)$$
$$= (0.2)(0.0)$$
$$= 0.0 ,$$

or, when W = 3

$$F_W(3) = F_X(3) . F_Y(3)$$
$$= (0.6)(0.3)$$
$$= 0.18 ,$$

and

$$p_W(3) = F_W(3) - F_W(2)$$
$$= 0.18 - 0.0 = 0.18 .$$

While the DPD representation is notationally compact and computationally simple to implement, there are three problem areas associated with it (Cooper and Chapman, 1987). First, if X and Y are DPDs with m and n doublets respectively, then Z = f(X,Y) has mn doublets, so a series of operations on DPDs will quickly lead to storage problems unless some form of aggregation is taken. Aggregation requires that the range of Z be divided into intervals in some way. Doublets with z-values within a specified interval can now be combined into a single new doublet <p,z> with $p = \sum p_j$ and $z = h(z_j)$, where the summation is over the doublets in the interval and h is an aggregation function such as an arithmetic or geometric mean. This is not likely to be major problem in practice.

The second problem area is more serious. Discrete distribution procedures involve inherent bias relative to comparable continuous distribution treatment. Using very large numbers of doublets will limit this bias, but because a DPD approach ignores it, it cannot be controlled. It will accumulate over successive operations, and substantial computational effort is necessary to maintain reasonable confidence in the results.

The third problem area is also serious. The operations on DPDs which have been described above assume independence between the base variables, an important practical limitation.

Interval and Histogram Representations

An alternative representation of base variable distribution is a histogram. With this representation the analytic convolution formulae become sums rather than integrals, and the calculations are both simple in concept and straightforward computationally. Several different histogram representations are possible, based on intervals of equal width (Driscoll, 1980; Chapman and Cooper, 1983a) or equal proability (Colombo and jaarsma, 1980). The Controlled Interval and Memory (CIM) approach, developed by Chapman and Cooper (1983a) is normally based on histograms with intervals of equal width within each distribution and different width for different distributions. Like discrete probability distribution approaches, approaches based on interval and histogram representations involve inherent bias. However, in the CIM framework the bias can be considered and controlled, drawing upon functional and numerical integration techniques, greatly reducing the computation required to maintain confidence in the results. Further, the CIM framework allows flexible treatment to dependence. The next two sections of this chapter provide a simple common interval example of the controlled interval (CI) approach.

Example Problem:

The Kentucky Power and Light Company (KP&L) is currently starting work on a project designed to increase the power capacity of its plants. The project is divided into two major stages: stage 1 (design) and stage 2 (construction). While each stage will be scheduled and controlled as thoroughly as possible, ultimately each stage of the project will be completed either earlier than scheduled, later than scheduled, or on time.

KP&L is primarily concerned with the delay problems that occur whenever a stage of the project is completed late. Such probems are of major concern to KP&L because they cause cost overruns and require additional managerial effort to reschedule related phase of the project. Thus, KP&L's management would like a thorough analysis of the project, including probability assessments for encountering stage-delay problems (Anderson, Sweeney, and Williams, 1978).

In order to clarify the concept of CI approach to addition of probabilistic variables let us consider above example with proposed data in the context of risk analysis.

The objective of the risk analysis in this example is to determine the probability of completing the two sequential stages on time.

Subsequent sections illustrate the progress of the analysis.

COMMON INTERVAL ADDITION

The first stage is 'design'. A simple CI (common interval) definition of the uncertainty associated with this stage uses three durations D1 for this stage in months, with associated probabilities, P(D1), as indicated in Table 3.3.

Table 3.3: Design distribution, D1 months.

D1	P(D1)
2	0.3
3	0.5
4	0.2

Two alternative interpretation of this tabular representation are possible, both useful on occasion. One is the discrete probability tree of Figure 3.1. The other is the rectangular histogram of Figure 3.2, or the equivalent cumulative form of Figure 3.3, where the durations are central classmarks for values over the intervals 1.5 -2.5 months, 2.5 - 3.5 months and 3.5 - 4.5 months. Smooth curves can be used in Figures 3.2 and 3.3, but for most purposes the simplicity employed here is preferable.

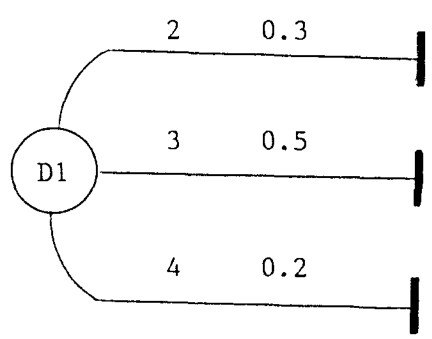


Figure 3.1

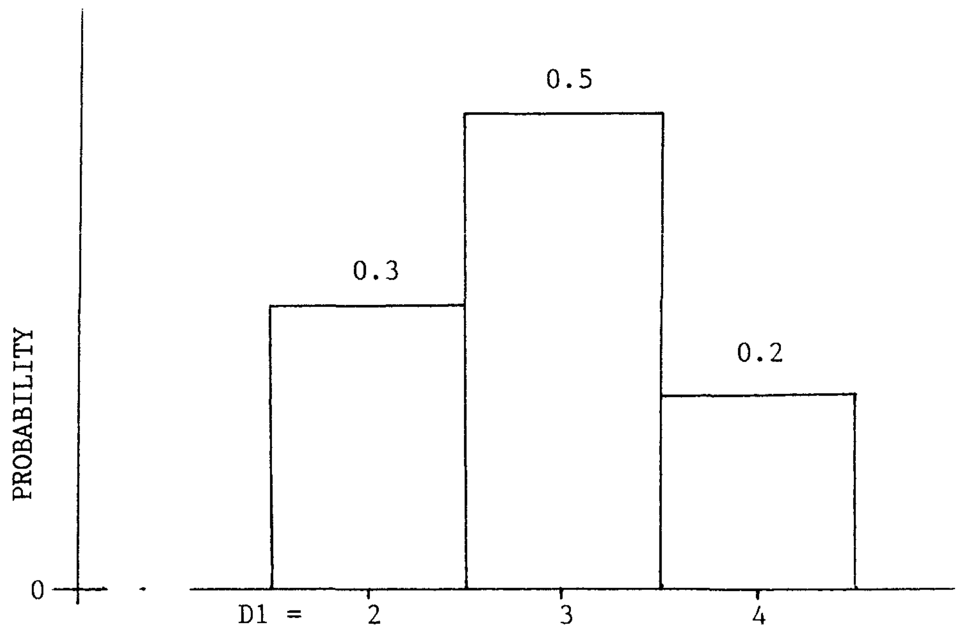


Figure 3.2

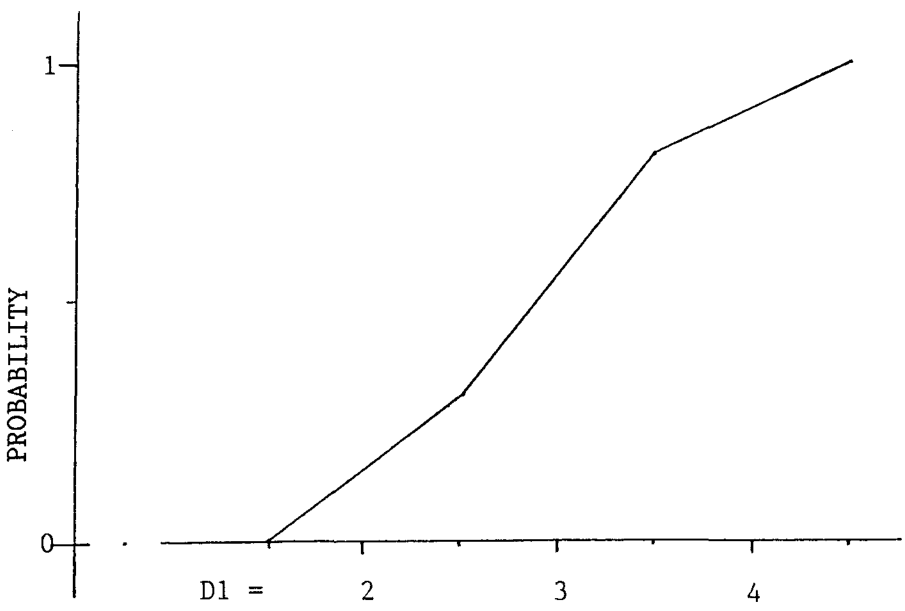


Figure 3.3

Only three cells were used for this example, but computer sofware can allow distributions to be specified in this form using any number of D1 values. A graphical approach using a smooth curve version of Figure 3.3 can be convenient in this context. Computer software can also generate distributions in this form from conventional distributions specified in terms of parameters, like the minimum, maximum and most likely values used to define PERT Beta distributions.

Experience in a project planning context suggests this flexibility and generality is occasionally invaluable and never unwelcome.

The next stage is 'construction'. A simple CI definition of the uncertainty associated with this stage uses three durations D2 for this stage in months, with associated probabilities, P(D2), as indicated in Table 3.4.

Table 3.4: Construction distribution, D2 months.

D2	P(D2)
6	0.3
7	0.6
8	0.1

Figure 3.4 shows discrete probability tree and Figure3.5 shows rectangular histogram of this tabular representation.

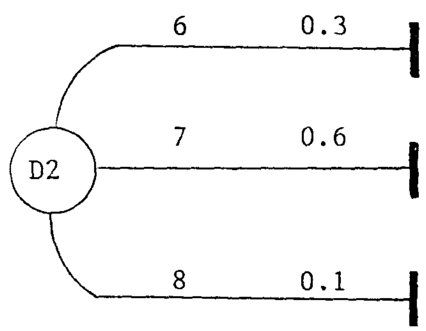


Figure 3.4

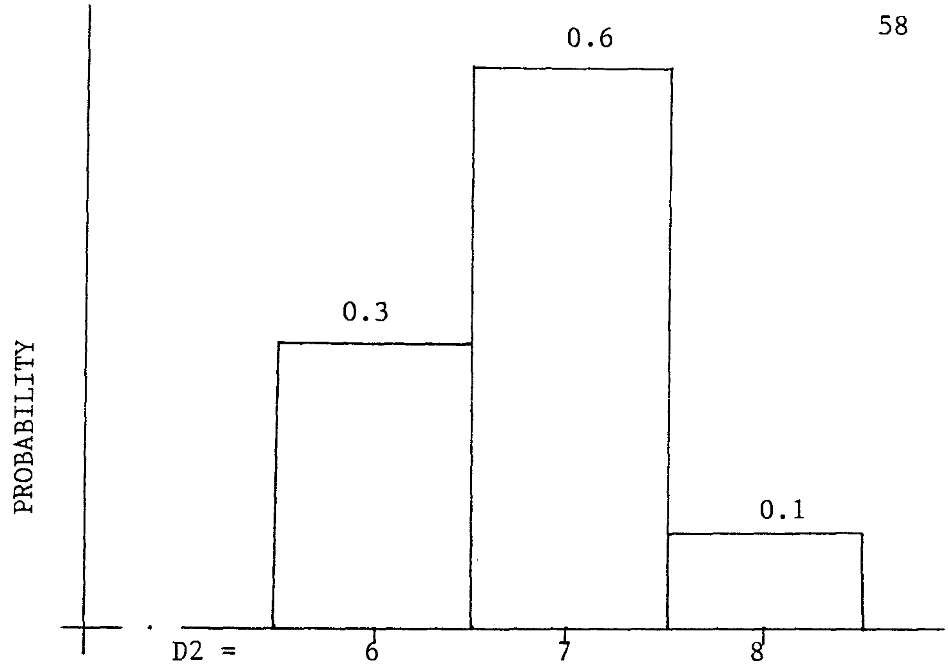
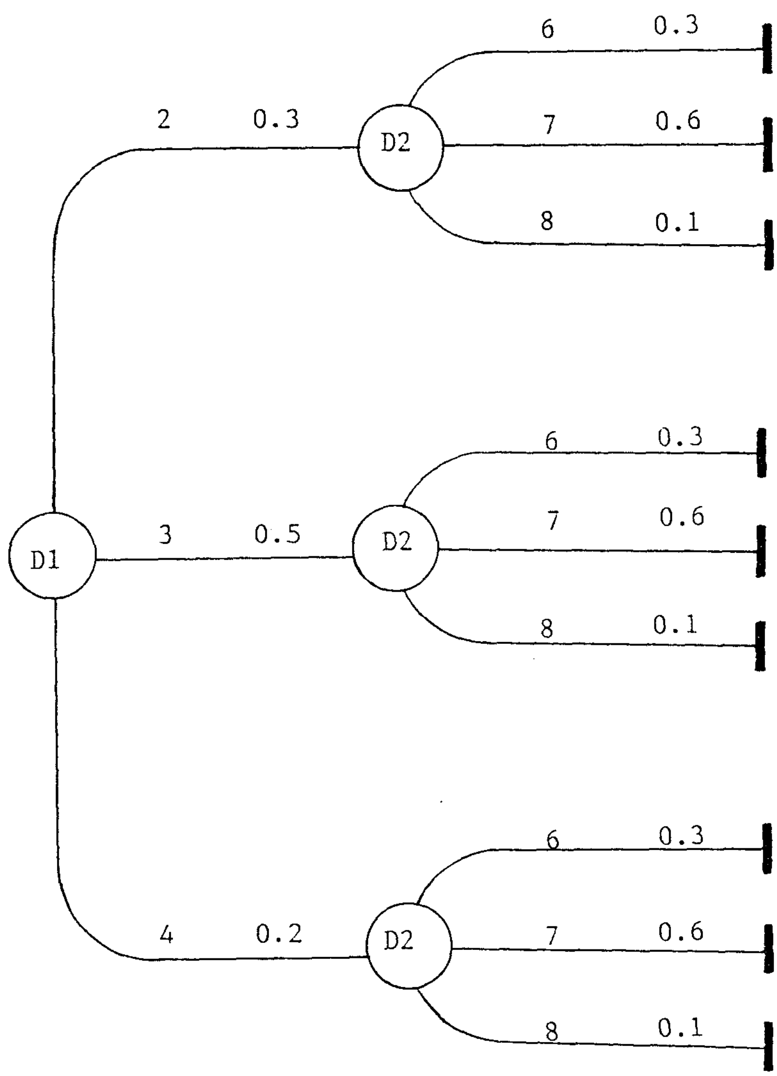


Figure 3.5

Design and construction are assumed to have independent duration distribution. Adding their duration distributions, to determine the duration distribution for the combination of 'design' followed by 'construction', can be performed using a two-level probability tree.

A simple representation of the probability tree is provided in Figure 3.6, D1 defining three branches, each of which is associated with three D2 branches. Table 3.5 provides a tabular form of this tree. Implied values of P(Da), Da = D1 + D2, the joint distribution of design plus construction, may be obtained from the probability tree in the normal way, as shown in Figure 3.5 and Table 3.5. However, the common interval form allows the simplified approach illustrated in Table 3.6. Each possible combination of D1 and D2 is considered, the joint probabilities producing the computation entries, and the entries associated with the same Da value are summed. If we do not need to remember D1 or D2 individually, we can choose to remember only their sum Da.



D1+D2	P
8	0.09
9	0.18
10	0.03
9	0.15
10	0.30
11	0.05
10	0.06
11	0.12
12	0.02

Figure 3.6

Table 3.5: Two level probability tree for design and construction.

DESIGN		CONSTRUCTION		DESIGN PLUS CONSTRUCTION	
D1	P(D1)	D2	P(D2)	Da	P(Da)
2	0.3	6	0.3	2+6 = 8	0.3 x 0.3 = 0.09
		7	0.6	2+7 = 9	0.3 x 0.6 = 0.18
		8	0.1	2+8 = 10	0.3 x 0.1 = 0.03
3	0.5	6	0.3	3+6 = 9	0.5 x 0.3 = 0.15
		7	0.6	3+7 = 10	0.5 x 0.6 = 0.30
		8	0.1	3+8 = 11	0.5 x 0.1 = 0.05
4	0.2	6	0.3	4+6 = 10	0.2 x 0.3 = 0.06
		7	0.6	4+7 = 11	0.2 x 0.6 = 0.12
		8	0.1	4+8 = 12	0.2 x 0.1 = 0.02

Table 3.6: The simplified computation for 'design' plus 'construction'. in the case where the individual 'design' and 'construction' intervals do not need to be remembered.

DESIGN PLUS CONSTRUCTION

Da	Computation	
8	0.3 x 0.3	= 0.09
9	0.3 x 0.6 + 0.5 x 0.3	= 0.33
10	0.3 x 0.1 + 0.5 x 0.6 + 0.2 x 0.3	= 0.39
11	0.5 x 0.1 + 0.2 x 0.6	= 0.17
12	0.2 + 0.1	= 0.02

For illustrative simplicity, the same interval widths have been used for both distributions and only three classes. In practice, more intervals are normally used, with different interval widths for each component distribution and the result, but a constant interval within each distribution. The procedures used here are very basic special cases, but they are important as a basis of understanding for all those involved in actual studies.

COMPUTATION ERROR AND COMPUTATION GENERALIZATION

Had we chosen to interpret the Di values of Tables 3.3 and 3.4 as integer values, we could interpret the Da values of Table 3.6 as integers, and the P(Da) computation of Table 3.6 would be precise and error free.

Alternatively, had we chosen to interpret the Di and P(Di) of Tables 3.3 and 3.4 as doublets which represented the limits of our state of knowledge in the Discrete Probability Distribution (DPD) sense, the Da and P(Da) of Table 3.6 would represent the limits of our state of knowledge, implicitly admitting but explicitly ignoring any computation error associated with "knowledge" not available.

Having interpreted the Di as classmarks with associated class boundaries, representing very large sets of integer values, classified or grouped, with continuous variable Di being a speical limiting case, we must recognise a source of computation error.

Some of this computation error arises because values in the top half of one class together with values in the top half of another class may be associated with a joint classmark value which is one class too low. For example, in Table 3.6, the probability associated with D1 = 2.4 and D2 = 6.4 is associated with a joint duration Da = 2 + 6 = 8. As 2.4 + 6.4 = 8.8, it should be associated with a joint duration Da = 9.

A corresponding computation error arises because values in the bottom half of one class together with values in the bottom half of another class may be associated with a joint classmark value which is one class too high. For example, in Table 3.6, the probability associated with D1 = 3.6 and D2 = 7.6 is associated with a joint duration Da=4+8=12. As 3.6+7.6=11.2, it should be associated with a joint duration Da=11.
If the original classes for D1=3 and D2=6 are interpreted as rectangular histograms, functional integration shows that the joint distribution Da is a triangle (Figure 3.6.)

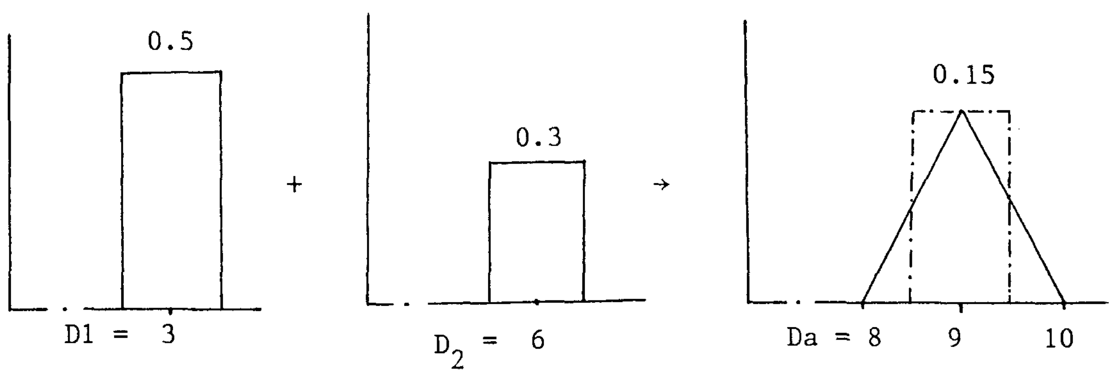


Figure 3.6

Most of the misallocation of probabilities cancels out, but some residual remains. In particular, modal probability values are over-estimated, and extreme values are ignored. In this sense the joint distribution of Table 3.6 under-estimates risk, a distribution in density form free from computation error being slightly wider and flatter.

Complete elimination or partial reduction of this computation error within the CI framework can be achieved via one or more of five approaches:

- derived correction factors;
- interpolated correction factors;
- more classes;
- empirically determined correction factors; or
- more allocations.

Chapter 9 provides complete discussion of above mentioned approaches.

Now return to the example problem. Consider the example PERT network shown in Figure 3.7. Assuming duration times are discretely distributed, an exact project completion time can be obtained from probability tree of Figure 3.6

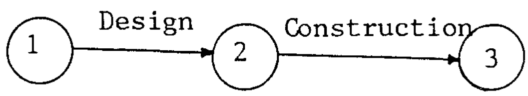


Figure 3.7

using equation (2.13) as follows:

$$E(T_N)=e_N \stackrel{\Delta}{=} \sum_{y(A)} T_N(A) \cdot p[y(A)] \qquad (2.13)$$

$$e_3 = 8x0.09+9x0.18+10x0.03+9x0.15+10x0.30+11x0.05+10x0.06+11x0.12+12x0.02$$

$$= 9.7.$$

Or, we can use Table 3.6 and the following equation.

$$e_3 = \sum_{Da} Da.P(Da) \qquad (3.1)$$

therefore,

$$e_3 = 8x0.09+9x0.33+10\ x0.39+11x0.17+20x0.02$$
$$= 9.7 .$$

Using the basic PERT procedure,

$$g_j = \max_i \sum g_i+\bar{Y}_{ij} \qquad (2.14)$$

where, $\bar{Y}_{12} = 2x0.3+3x0.5+4x0.2 = 2.9$ and

$$\bar{Y}_{23} = 6x0.3+7x0.6+8x0.1 = 6.8$$

g_3, can be computed as follows:

$$g_1 \equiv 0$$

$$g_2 = \max_i \{g_1 + \bar{Y}_{12}\} = \max_i \{0+2.9\} = \max_i \{2.9\} = 2.9$$

$$g_3 = \max_i \{g_2 + Y_{23}\} = \max_i \{2.9+6.8\} = \max_i \{9.7\} = 9.7 .$$

For this example, the project completion time obtained using a basic PERT solution procedure was found to be equal to the exact project completion time, because the project consists of only series activities.

Now consider two similar projects as one integrated project as shown in Figure 3.8.

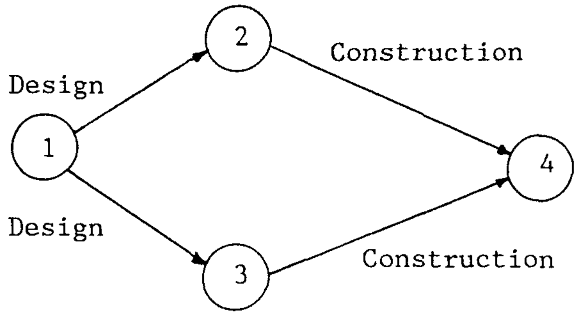


Figure 3.8

Since this network consists of 4 stages and each stage can take 3 values, in evaluation of its completion time, $3^4= 81$ deterministic problems need to be solved. Complete enumeration is a burdensome time consuming task. However, the same result can be obtained by taking the maximum of the duration times of paths 1-2-4 and 1-3-4.

Let, D_a denote resultant of Design and Construction in each sequence of Figure 3.8 as shown in Figure 3.9.

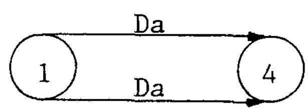


Figure 3.9

The duration distribution of Da is summarised in Table 3.7.

Table 3.7: Da distribution in months.

Da	P(Da)	F(Da) = CP(Da)
8	0.09	0.09
9	0.33	0.42
10	0.39	0.81
11	0.17	0.98
12	0.02	1.0

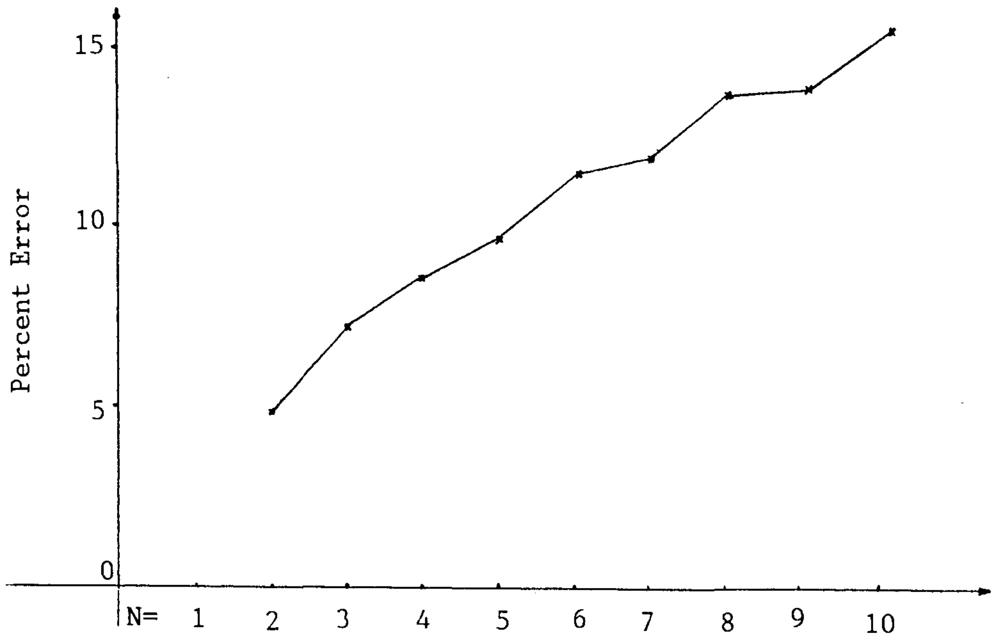
Taking the maximum of the duration times of the similar activities of Figure 3.9 gives the df of the project completion time.Table 3.8 shows this process.

Table 3.8 Project completion time of Figure 3.8.

T_4	$P(T_4)$	
8	0.09x0.09	= 0.0081
9	0.42x0.42-0.09x0.09	= 0.1683
10	0.81x0.81-0.42x0.42	= 0.4797
11	0.98x0.98-0.81x0.81	= 0.3043
12	1 x 1 -0.98x0.98	= 0.0396

$$E = \sum_{T_4} T_4 . P(T_4) = 8x0.0081+9x0.1683+10x0.4797+11x0.3043+12x0.0396 = 10.199 .$$

Using the basic PERT procedure, g_4 is found to be 9.7, which is exactly equal to project completion time of Figure 3.7. As mentioned in the previous chapter, the basic PERT procedure does not consider the effects of parallel activities in the evaluation of project completion time. In this example, the basic procedure calculated mean (g_4 = 9.7) is approximately 4.9% less than actual mean (E = 10.199 = e_4). This bias is an increasing function of the number of parallel paths, as shown in graph 3.1, obtained by complete enumeration.



Number of similar parallel paths
Graph 3.1

Determination of the pdf of project completion time for completely reducible networks is a simple matter through repeated application of convolution and greatest operations using a CIM procedure. However, as mentioned in the previous chapter, for irreducible networks there is no way to combine the activities through convolution and greatest operations because the paths are correlated as a result of the common activities. In such cases the general principle to apply is to evaluate the pdf of the project completion time by conditioning project completion time on the values of common activities, and then removing the conditional nature to obtain unconditional pdf of project completion time. The generalized form of CIM procedure for irreducible networks proposed by Chapman and Cooper (1983a) is based on this general principle. In the following section structural dependence is examined through an example and the CIM treatment of this structural dependence is considered.

Example 3.2:

The example involves 5 activities: A,B,C,D and E precedence relationships are specified in Table 3.9 and shown in precedence (Activity-on-arc) diagram of Figure 3.10.

Table 3.9: Activity list for example 3.2, with precedence relationships.

NUMBER	LABEL	PREDECESSORS
1	A	-----
2	B	-----
3	C	A
4	D	A
5	E	B,C

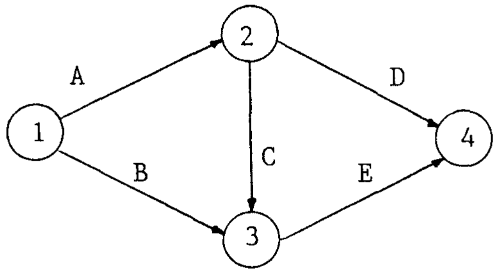


Figure 3.10: Activity-on-arc precedence diagram.

Table 3.10 shows the durations of the activities. For ease of calculations it is assumed that each activity has two duration times. Expected values and variances are also shown.

Table 3.10: Project activity durations,in year units.

1.A	X_A =	3	8
	P =	0.8	0.2
	E = 4 ,	σ^2 = 4	
2.B	X_B =	6	9
	P =	0.6	0.4
	E = 7.2 ,	σ^2 = 2.16	
3.C	X_C =	4	6
	P =	0.3	0.7
	E =5.4 ,	σ^2= 0.84	
4.D	X_D =	4	5
	P =	0.9	0.1
	E = 4.1 ,	σ^2= 0.09	

Table 3.10 (Concluded)

5.E	X_E =	1	2
	P =	0.5	0.5
	E = 1.5 ,	σ^2= 0.25	

The network contains three paths (1-2-4, 1-2-3-4 and 1-3-4), and the time required for project completion is given by T_4 where,

$$T_4 = \max \{Z(T_1), Z(T_2), Z(T_3)\} \qquad (3.2)$$

and

$$Z(T_1) = X_A + X_D$$

$$Z(T_2) = X_A + X_C + X_E$$

$$Z(T_3) = X_B + X_E \ .$$

The pdf of T_4 can not be expressed as a product of the pdf's of $Z(T_1)$, $Z(T_2)$ and $Z(T_3)$. The reason is that these three paths are structurally dependent: activity A is common to paths $Z(T_1)$ and $Z(T_2)$, and activity E is common to paths $Z(T_2)$ and $Z(T_3)$.

The CIM treatment of this structural dependence is as follows:

1 - Retain a memory of common activity A through the network.

2 - Determine the pdf of the project completion time in terms of distributions which are conditional upon the duration of A.

3 - Decondition the conditional pdf's by using the probabilities of specific A durations.

The following section illustrates the computation procedure. Table 3.11 shows the time distributions for this procedure. In Table 3.11A, X_A is the duration time of A which is equal to D's start time. This calculation format keeps a memory of A aspect of distribution functions of start and finish time of activities needed later in computation sequence.

Table 3.11A: Duration time of A.

	A=3	CON	CP	A=8	CON	CP	UNC.P
X_A = 3	0.8	1	1				0.8
8				0.2	1	1	0.2
E =		3			8		4

Table 3.11B shows B's finish time.

Table 3.11B: B's finish time.

	P	CP
F_2 = 6	0.6	0.6
9	0.4	1.0
E =	7.2	

Convoluting A and C provides C's finish time, which is shown in Table 3.11C. The probabilities computed in Table 3.11C in the A = 3 and A = 8 columns are joint probabilities, in the UNC.P column are unconditional probabilities, in the CP columns are cumulative probabilities and in the CON columns are conditional probabilities.

Table 3.11C: C's finish time.

	A=3	CON	CP	A=8	CON	CP	UNC.P
F_3 = 7	0.8x0.3 = 0.24	0.3	0.3				0.24
9	0.8x0.7 = 0.56	0.7	1.0				0.56
12				0.2x0.3 = 0.06	0.3	0.3	0.06
14				0.2x0.7 = 0.14	0.7	1.0	0.14
E =		8.4			13.4		9.4

For example,

$$P(A=3 \text{ and } F_3=7) = P(A=3).P(F_3=7|A=3)$$
$$= P(A=3).P(C=4)$$
$$= 0.8 \text{ x } 0.3$$
$$= 0.24 \text{ ,}$$

and $$E(F_3|A=3) = \sum_{F_3} F_3.P(F_3|A=3)$$
$$= 7 \text{ x } P(F_3=7|A=3) + 9 \text{ x } P(F_3= 9|A=3)$$
$$= 7 \text{ x } 0.3 + 9 \text{ x } 0.7$$
$$= 8.4 \text{ ,}$$

or $$E(F_3|A = 8) = \sum_{F_3} F_3.P(F_3|A = 8)$$
$$= 12 \text{ x } P(F_3 = 12|A=8) + 14 \text{ x } P(F_3=14|A=8)$$
$$= 12 \text{ x } 0.3 + 14 \text{ x } 0.7$$
$$= 13.4 \text{ ,}$$

and $$E(F_3) = E(F_3|A=3).P\ (A=3) + E(F_3|A = 8).P(A=8)$$
$$= 8.4 \text{ x } 0.8 + 13.4 \text{ x } 0.2$$
$$= 9.4 \text{ .}$$

Convoluting A and D provides D's finish time, which is shown in Table 3.11D.

<u>Table 3.11D</u>: D's finish time.

	A=3	CON	CP	A=8	CON	CP	UNC.P
F_4 = 7	.8x.9=.72	.9	.9				.72
8	.8x.1=.08	.1	1.0				.08
12				.2x.9=.18	.9	.9	.18
13				.2x.1=.02	.1	1.0	.02
E =		7.1			12.1		8.1

Taking the maximum of the distributions F_2 and F_3 gives a new distribution which is the start time of E. Table 3.11E shows this distribution.

Table 3.11E: E's start time.

	A=3	CON	CP	A=8	CON	CP	UNC.P
S_5 = 7	.144	.3x.6 = .18	.18				.144
9	.656	1x1-.3x.6 = .82	1.0				.656
12				.06	.3x1 = .3	.3	.06
14				.14	1x1-.3x1 = .7	1.0	.14
E =		8.64			13.4		9.592

For example,

$$P(S_5=7|A=3) = CP(F_2 \leqslant 7|A=3).CP(F_3 \leqslant 7|A=3)$$
$$= 0.6 \times 0.3$$
$$= 0.18\ ,$$

$$P(S_5=7 \text{ and } A=3) = P\ (S_5=7|A=3).P(A=3)$$
$$= 0.18 \times 0.8$$
$$= 0.144\ ,$$

or

$$P(S_5=9|A=3)=CP(F_2 \leqslant 9|A=3).CP(F_3 \leqslant 9|A=3)-CP(F_2 < 9|A=3).CP(F_3 < 9|A=3)$$
$$= 1 \times 1 - 0.6 \times 0.3$$
$$= 0.82\ ,$$

and

$$P(S_5=9 \text{ and } A=3) = P(S_5=9|A=3).P(A=3)$$
$$= 0.82 \times 0.8$$
$$= 0.656\ .$$

Convoluting E and S_5 gives E's finish time. Table 3.11F shows this distribution.

Table 3.11F: E's finish time.

	A=3	CON	CP	A=8	CON	CP	UNC.P
F_5 = 8	.072	.18x.5 = .09	.09				.072
9	.072	.18x.5 = .09	.18				.072
10	.328	.82x.5 = .41	.59				.328
11	.328	.82x.5 = .41	1.0				.328
13				.03	.3x.5 = .15	.15	.03
14				.03	.3x.5 = .15	.30	.03
15				.07	.7x.5 = .35	.65	.07
16				.07	.7x.5 = .35	1.0	.07
E =		10.14			14.9		11.092

Taking the maximum of the distributions F_4 and F_5 gives distribution of the project completion time. Table 3.11G shows this distribution.

Table 3.11G: Project completion time.

	A=3	CON	A=8	CON	UNC.P
FP = 8	.072	.09 x 1 = .09			.072
9	.072	.18x1-.09x1 = .09			.072
10	.328	.59x1-.18x1 = .41			.328
11	.328	1x1-.59x1 = .41			.328
13			.03	.15x1 = .15	.03
14			.03	.3x1-.15x1 = .15	.03
15			.07	.65x1-.3x1 = .35	.07
16			.07	1x1-.65x1 = .35	.07
E =		10.14		14.9	11.092

As it is shown in Table 3.11G, the expected value of the project completion time is obtained by deconditioning the conditional expected values as follows:

$$E(FP) = E(FP|A=3).P(A=3) + E(FP|A=8).P(A=8)$$
$$= 10.14 \times 0.8 + 14.9 \times 0.2$$
$$= 11.092 .$$

The CIM procedure provides an exact pdf of each PERT network with discrete distribution functions if we retain memory of all common activities throughout the procedure. The retention of a single memory dimension for the network configuration of Example 3.2 involved preserving and working with a matrix of probabilities at each stage. This involved a slight increase in computational effort. The network configuration of Figure 3.11 needs five levels of memory (memory of A,B,C,J and F) up to node 6. It thus requires a six dimensional matrix of probabilities in final stage. In general, n levels of memory involve an (n+1)-dimensional probability matrix. Large values of n make this approach computationally demanding.

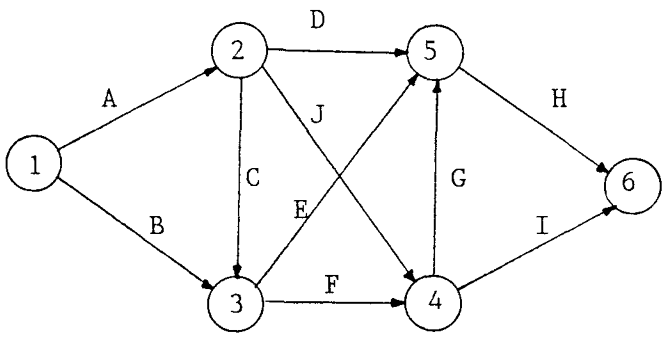


Figure 3.11

In order to minimize the computational implications of the memory aspect of the CIM approach this dissertation presents a procedure which solves the PERT network for various condition values of common activities, and then, by deconditioning conditional pdf's a pdf of project completion time is obtained. Detailed discussion of the proposed procedure for networks given discrete statistically independent distributions and the algorithm are given in Chapter 4. However, a generalized form of the proposed procedure;

(a) Provides an exact pdf for project completion time when the duration times of activities are discrete.

(b) Approximates the pdf of the project completion time when the duration times of activities are continuous. Approximation is due to:

(i) discretizing continuous distributions.

(ii) convoluting discrete approximations to continuous distribution.

(c) Determines a criticality index for all activities and paths.

(d) Allows statistical dependence between activities.

SUMMARY AND CONCLUSIONS

In this chapter the basic common interval form of the Controlled Interval and Memory (CIM) approach is introduced in the context of combining probabilistic variables. A histogram representation of probability distributions is used. The addition of independent distributions is interpreted as a probability tree, and 'collapsed' calculation patterns are derived.

In the last section, structural dependence is examined through an example and the CIM treatment of the structural dependence is considered. This example has shown that the CIM procedure provides an exact pdf for project completion time in PERT networks with discrete distribution functions if we retain memory of all common activities throughout the procedure.

CHAPTER 4: PROPOSED PROCEDURE

INTRODUCTION

This chapter provides detailed discussion of a proposed procedure and presents the algorithm of the proposed procedure for PERT networks when duration times of activities are discretely distributed and only structural dependence is present. Since the proposed procedure is mainly based on Garman's (1972) method of conditional sampling of stochastic networks, the first section looks briefly at conditional sampling methods. Detailed discussion of Monte Carlo methods and comparison between the proposed method based on a CIM approach and Monte Carlo methods is the subject matter of Chapter 10.

Conditional Monte Carlo in The Simulation of Stochastic Networks

Recall that the major reason for the breakdown of the analytic approach is the dependence relationships among paths due to common activities. If such correlation among paths is eliminated, the result would be a set of independent paths, and the df of the completion time would be the product of the dfs of the paths. Conditional Monte Carlo proposed by Burt and Garman (1971a) is an approach using such independence to provide improved estimators of the pdf of the completion time.

Consider once more the network of Figure 2.12. Recall that, $Z(T_1) = X_A+X_D$, $Z(T_2) = X_A+X_C+X_E$, and $Z(T_3)=X_B+X_E$. The straightforward simulation approach would take samples T_A^k, T_B^k, T_C^k, T_D^k, and T_E^k, k=1,2,...,N, and form the estimate of $T_4 = \max\{Z(T_1), Z(T_2), Z(T_3)\}$.

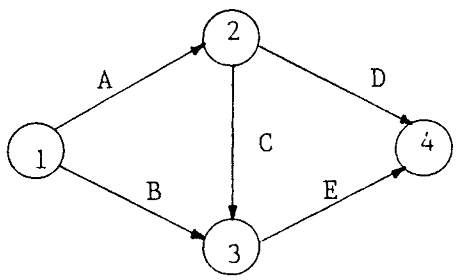


Figure 2.12

If the values of random variables X_A and X_E were fixed then the three paths of the network would be independent and unconditional pdf of project completion time is given by

$$F_4(t) = \int_{X_A}\int_{X_E} F_4/_{X_A,X_E}\, dF_{X_A} dF_{X_E}$$

or $$F_4(t)/_{X_A,X_E} = F_D(t-X_A)F_C(t-X_A-X_E)F_B(t-X_E)\ . \qquad (4.1)$$

In Conditional Monte Carlo the manner in which we set X_A and X_E to constant values is by sampling them: fixing the random variables at some sample values. Expression (4.1) is then said to be "conditioned" on the fact that X_A and X_E have taken on these sample values.

The utility of the conditional sampling approach is crucially dependent on the ratio of unique activities to common activities. A unique activity is one which lies on a single path. The higher the ratio of unique activities to common activities the more useful is the conditional Monte Carlo approach. For example, when using the conditional Monte Carlo estimate for Wheatstone bridge of Figure 2.12 we need take only two samples per realization instead of five as required for the straightforward simulation. This is a saving of 60% of sampling effort. However, more efforts is involved in performing the multiplication of probabilities for each sample. Hence, the net reduction in effort may be only 50%.

The process of conditioning enables us to apply an anlytical approach to the remainder of the network. This, in turn, means that we are able to use all the information available on the dfs of the activity durations, rather than just the sample values as in crude Monte Carlo. Consequently, the variability of the estimate T_4 from its true value is greatly reduced. This is a positive and important gain.

The conditional Monte Carlo sampling approach proposed by Burt and Garman (1971a) should contribute toward a more efficient procedure in two respects:

(a) by reducing the sampling effort itself because one need only sample from the nonunique activities and (b) greatly reducing the variability of the probability estimate because of our ability to utilize all the information on the df of the unique activities. The price to be paid for these two advantages is the added burden of multiplying out the conditional probabilitites.

The following algorithm proposed by Burt and Garman (1971a) determines all unique activities in an arbitrary PERT network.

1 - Check the start node with (√).

2 - Select any unchecked node, all of whose predecessor nodes are checked. This is "Currently Scanned Node" (C.S.N.).

3 - If the C.S.N. has 2 or more immediate predecessor nodes or any immediate predecessor activities marked N, then mark all activities eggressing from the C.S.N. with the letter N. Check (√) the C.S.N.

4 - If all nodes except the finish node are checked (√), continue; otherwise, return to step 2.

5 - Check the finish node with (√̸).

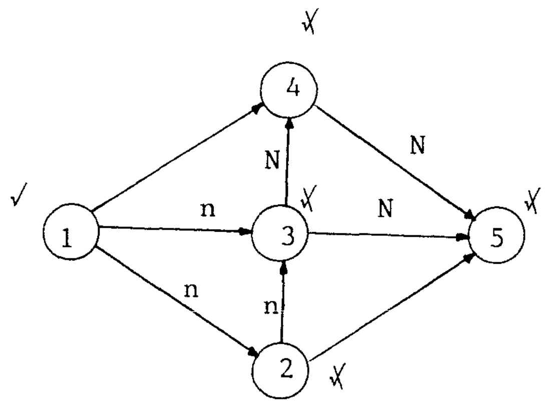
6 - Select any remaining unchecked (√̸)node, all of whose successor nodes are checked (√̸). This is the C.S.N.

7 - If the C.S.N. has 2 or more immediate successor nodes or any successor activity marked n, then mark all activities immediately preceding the C.S.N. with an n. Check (√̸) the C.S.N.

8 - If all nodes except the start node are checked (√̸), then stop; otherwise return to step 6.

Stop. All activities not marked N or n are unique. All activities marked N or n are nonunique.

Example:



The activity connecting nodes 1 and 4, and the activity connecting nodes 2 and 5, are unique.

Garman's Method

A method for generalizing the conditional sampling approach from its current use of product-form estimators to the use of product/convolution-form estimator is suggested by Garman (1972). Garman's approach requires fewer samples per realization at the expense of more computing because of the need to calculate the convolutions of functions. For example, consider again the Wheatstone bridge of Figure 2.12. The conditional Monte Carlo approach identified activities A and E as the common (nonunique) activities. This led to the sampling of X_A and X_E. As an alternative, suppose we consider only X_A constant at T_A^k. Then all paths to node 4 will not be independent, but now there are two paths in parallel from node 1 to node 3, and series-parallel reduction methods to the network of Figure 2.12 yields the estimate

$$F_4(t)/_{X_A} = \{[F_B(t)F_C(t-X_A)]*F_E(t)\}F_D\ (t-\ X_A) \qquad (4.2)$$

where the asterisk in (4.2) denotes the convolution operator. Thus we have reduced the number of activities sampled from two to only one - at the price of computing the convolution in (4.2).

Garman's approach is based on Martin's (1965) series-parallel reduction of stochastic PERT networks and Hartley and Wortham's (1966) definitions of crossed and multiple crossed networks. The basic idea is to (a) reduce the network as far as possible by series-parallel reduction; (b) eliminate crossing activities by conditional sampling of their predecessors or successors; (c) repeat (a) and (b) until the network is completely reduced; and (d) use the resulting estimator to estimate the network duration pdf.

PROPOSED PROCEDURE

The proposed procedure of this dissertation is mainly based on Garman's approach. However, instead of fixing the duration times of chosen activities at sample values, we conditionalize the chosen activities by fixing the random variables at their realization times. Therefore, for activity networks when the duration times of activities are continuously distributed, the first step of proposed procedrue is discretizing the continuous distributions.

Discretizing the continuous distributions will be described in Chapter 8. Now for discretely distributed networks consider again the Wheatstone bridge of Figure 3.10 in order to clarify the proposed procedure. Recall that each activity of this network has two duration times.

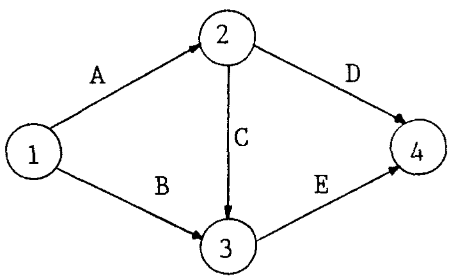


Figure 3.10

By fixing on the first realization time of A, 3 and the first realization time of E, 1, changes the network of Figure 3.10 to that of Figure 4.1, and all path durations are independent.

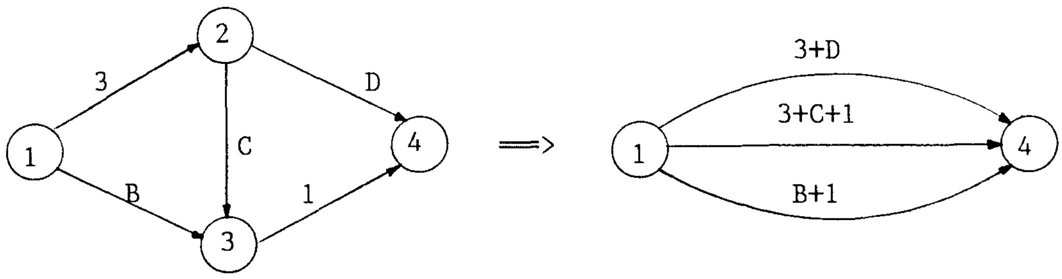


Figure 4.1

The pdf of the project completion time given $X_A = 3$ and $X_E = 1$ can be computed as follows:

Table 4.1 shows duration time of (3+D), Table 4.2 shows duration time of (3+C+1), and Table 4.3 shows duration time of (B+1).

Table 4.1: Duration time of (3+D).

	P	CP
$X_{(3+D)}$= 7	0.9	0.9
8	0.1	1.0

Table 4.2: Duration time of (3+C+1).

	P	CP
$X_{(3+C+1)}$= 8	0.3	0.3
10	0.7	1.0

Table 4.3: Duration time of (B+1)

	P	CP
$X_{(B+1)}$= 7	0.6	0.6
10	0.4	1.0

Taking the maximum of these three parallel paths yields the pdf of the project completion time given $X_A = 3$ and $X_E = 1$, as shown in Table 4.4.

Table 4.4: Project completion time given $X_A = 3$ and $X_E = 1$.

EP = 8	0.3 x 0.6 = 0.18
10	1 - 0.18 = 0.82
E =	9.64

Fixing on the first realization time of A and the second realization time of E changes the network of Figure 3.10 to that of Figure 4.2.

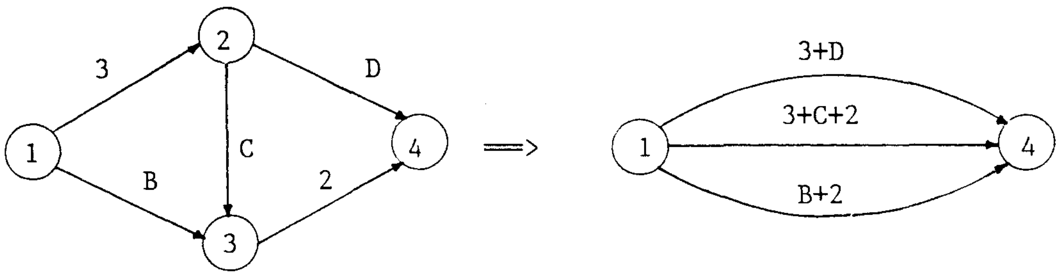


Figure 4.2

Table 4.5 shows the duration time of (3+C+2), and Table 4.6 shows the duration time of (B+2).

Table 4.5: Duration time of (3+C+2).

	P	CP
$X_{(3+C+2)} = 9$	0.3	0.3
11	0.7	1.0

Table 4.6: Duration time of (B+2).

	P	CP
$X_{(B+2)} = 8$	0.6	0.6
11	0.4	1.0

Taking the maximum of the duration times of (3+D), (3+C+2) and (B+2) yields the pdf of the project completion time given $X_A = 3$ and $X_E = 2$.

Table 4.7: Project completion time given $X_A = 3$ and $X_E = 2$.

FP = 9	0.3 x 0.6	= 0.18
11	1 - 0.18	= 0.82
E =		= 10.64

Similarly, by fixing on the second realization time of A and the first realization time of E the network of Figure 3.10 changes to that of Figure 4.3.

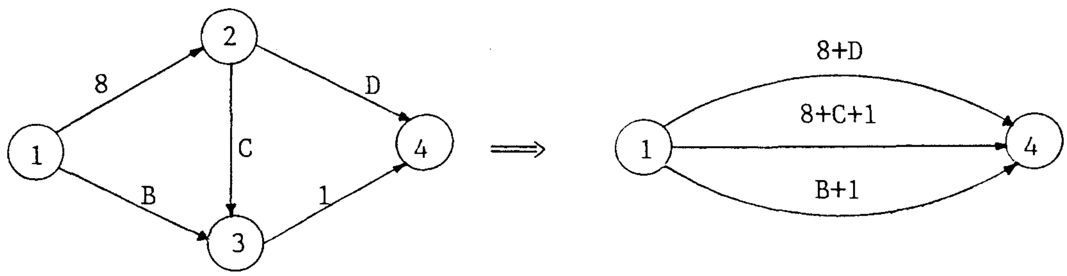


Figure 4.3

Table 4.8 shows duration time of (8+D), Table 4.9 shows duration time of (8+C+1).

Table 4.8: Duration time of (8+D).

	P	CP
$X_{(8+C+1)}$ = 12	0.9	0.9
13	0.1	1.0

Table 4.9: Duration time of (8+C+1).

	P	CP
$X_{(8+C+1)}$ = 13	0.3	0.3
15	0.7	1.0

Taking the maximum of the duration times of (8+D), (8+C+1) and (B+1) yields the pdf of the project completion time given $X_A = 8$ and $X_E = 1$.

Table 4.10: Project completion time given X_A = 8 and X_E = 1.

FP = 13	0.3 x 1 = 0.3
15	1 - 0.3 = 0.7
E =	14.4

Finally, by fixing on the second realization time of A and the second realization time of E the network of Figure 3.10 changes to that of Figure 4.4.

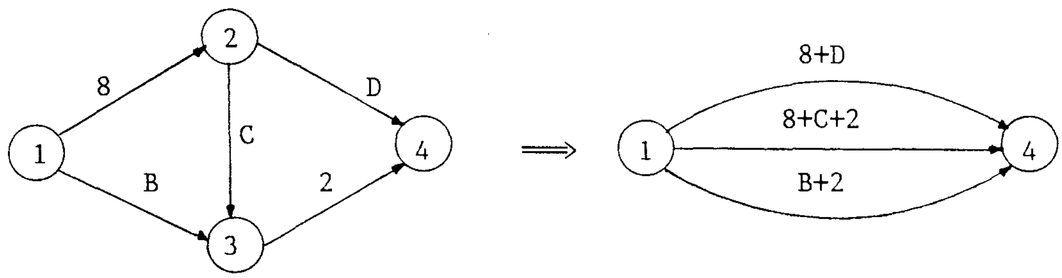


Figure 4.4

Table 4.11 shows the duration time of (8+C+2).

Table 4.11: Duration time of (8+C+2).

	P	CP
$X_{(8+C+2)}$ = 14	0.3	0.3
16	0.7	1.0

Taking the maximum of the duration times of (8+D), (8+C+2) and (B+2) yields the pdf of the project completion time given X_A = 8 and X_E = 2.

Table 4.12: Project completion time given X_A = 8 and X_E = 2.

FP = 14	0.3
16	0.7
E =	15.4

By deconditioning the pdfs of project completion times of Tables 4.4, 4.7, 4.10 and 4.12, the unconditional pdf of project completion time is obtained.

Tables 4.13 through 4.16 show the unconditional pdf of Tables 4.4, 4.7, 4.10 and 4.12 respectively.

Table 4.13: Unconditional pdf of Table 4.4. ($P(X_A=3).P(X_E=1)=0.8x0.5=0.4$).

FP = 8	0.18 x 0.4 = 0.072
10	0.82 x 0.4 = 0.328

Table 4.14: Unconditional pdf of Table 4.7. ($P(X_A = 3).P(X_E = 2)=0.8x0.5=0.4$).

FP = 9	0.18 x 0.4 = 0.072
11	0.82 x 0.4 = 0.328

Table 4.15: Unconditional pdf of Table 4.10. ($P(X_A=8).P(X_E=1) = 0.2x0.5=0.1$).

FP = 13	0.3 x 0.1 = 0.03
15	0.7 x 0.1 = 0.07

Table 4.16: Unconditonal pdf of Table 4.12. ($P(X_A=8).P(X_E=2)=0.2 x 0.5=0.1$).

FP = 14	0.3 x 0.1 = 0.03
16	0.7 x 0.1 = 0.07

Simple addition of probabilities for each realization time of Tables 4.13 through 4.16 gives the unconditional pdf of project completion time as shown in Table 4.17.

Table 4.17: Unconditional project completion time.

EP = 8	0.072
9	0.072
10	0.328
11	0.328
13	0.030
14	0.030
15	0.070
16	0.070
E =	11.092

As an alternative, suppose we conditionalize A only. Then all paths will not be independent, but now the network would be subject to series - parallel reduction.

Figure 4.5A shows the network of Figure 3.10 conditioning A at its first realizatin time, 3. Figure 4.5B shows the network of Figure 3.10 conditioning A at its second realization time, 8.

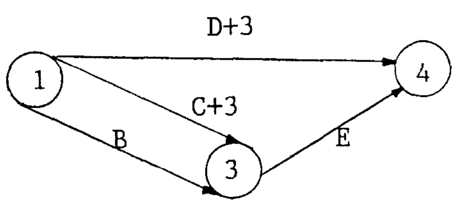


Figure 4.5A

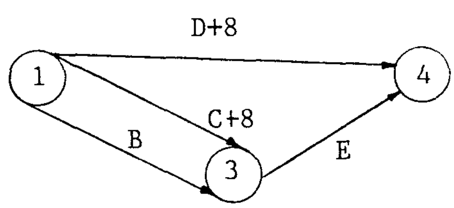


Figure 4.5B

Taking the maximum of (C+3) and B gives F as shown in Figure 4.6A. Taking the maximum of (C+8) and B gives H as shown in Figure 4.6B.

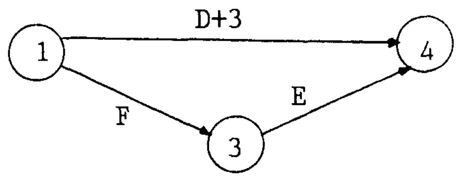


Figure 4.6A

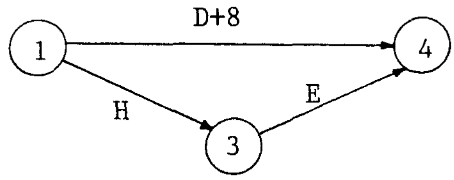


Figure 4.6B

Determining the pdf of the networks of Figure 4.6A and 4.6B is a simple matter. Finally, by deconditioning the pdf's of these networks, the pdf of the main network which is the network of Figure 3.10 is obtained. The calculation procedure is as follows, where * denotes convolution and . denotes greatest operation.

Tables 4.18A through 4.18F show the calculation procedure to determine the project finish time of Figure 4.6A given A=3.

Table 4.18A: Duration time of (D+3).

	P	CP
7	0.9	0.9
8	0.1	1.0
E =	7.1	

Table 4.18B: Duration time of (C+3).

	P	CP
7	0.3	0.3
9	0.7	1.0
E =	8.4	

Table 4.18C: Duration time of B.

	P	CP
6	0.6	0.6
9	0.4	1.0
E =	7.2	

Table 4.18D: Duration time of B.(C+3)=F .

	P	CP
7	0.6 x 0.3 = 0.18	0.18
9	1x1 - 0.18 = 0.82	1.0
E =	8.64	

Convoluting E and F of Figure 4.6A gives G as shown in Figure 4.7A.

Table 4.18E shows duration time of G.

Table 4.18E Duration time of G = F*E .

	P	CP
8	0.18 x 0.5 = 0.09	0.09
9	0.18 x 0.5 = 0.09	0.18
10	0.82 x 0.5 = 0.41	0.59
11	0.82 x 0.5 = 0.41	1.0
E =	10.14	

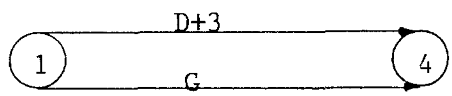


Figure 4.7A

Taking the maximum of the durations of (D+3) and G gives the project completion time given A=3, as shown in Table 4.18F.

Table 4.18F: (Project finish time | A=3)=G.(D+3).

	P	
8	0.09 x 1	= 0.09
9	0.18 x 1 - 0.09 x 1	= 0.09
10	0.59 x 1 - 0.18 x 1	= 0.41
11	1 x 1 - 0.59 x 1	= 0.41
E =		10.14

Tables 4.18G through 4.18K show the calculation procedure to determine the project finish time of Figure 4.6B given A=8.

Table 4.18G: Duration time of (D+8).

	P	CP
12	0.9	0.9
13	0.1	1.0
E =	12.1	

Table 4.18H: Duration time of (C+8).

	P	CP
12	0.3	0.3
14	0.7	1.0
E =	13.4	

Table 4.18I: Duration time of B.

	P	CP
6	0.6	0.6
9	0.4	1.0
E =	7.2	

Table 4.18J: Duration time of H=B.(C+8).

	P	CP
12	0.3	0.3
14	0.7	1.0
E =	13.4	

Convoluting H and E gives I as shown in Figure 4.7B.

Table 4.18K: Duration time of I=H*E.

	P	CP
13	0.3 x 0.5 = 0.15	0.15
14	0.3 x 0.5 = 0.15	0.30
15	0.7 x 0.5 = 0.35	0.65
16	0.7 x 0.5 = 0.35	1.0
E =	14.9	

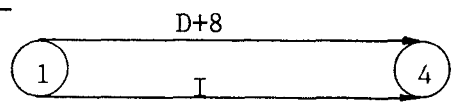


Figure 4.7B

Taking the maximum of the durations of (D+8) and I yields the project completion time given A=8, as shown in Table 4.18L.

Table 4.18L: (Project finish time|A=8)=I.(D+8).

		P
13	0.15 x 1	= 0.15
14	0.30 x 1 - 0.15 x 1	= 0.15
15	0.65 x 1 - 0.30 x 1	= 0.35
16	1 x 1 - 0.65 x 1	= 0.35
E =		14.9

By deconditioning the pdfs of project finish time of Tables 4.18F and 4.18L the unconditional pdf of project finish time is obtained.

Table 4.18M shows the unconditional pdf of Table 4.18F.

Table 4.18M: Unconditional pdf of Table 4.18F.

8	0.09 x 0.8 = 0.072
9	0.09 x 0.8 = 0.072
10	0.41 x 0.8 = 0.328
11	0.41 x 0.8 = 0.328

Table 4.18N shows the unconditional pdf of Table 4.18L

Table 4.18N: Unconditional pdf of Table 4.18L.

13	0.15 x 0.2 = 0.03
14	0.15 x 0.2 = 0.03
15	0.35 x 0.2 = 0.07
16	0.35 x 0.2 = 0.07

Simple addition of probabilities for each realization time of Tables 4.18M and 4.18N gives the unconditional pdf of project finish time as shown in Table 4.18P.

Table 4.18P: Unconditional Project finish time.

EP = 8	0.09 x 0.8 = 0.072
9	0.09 x 0.8 = 0.072
10	0.41 x 0.8 = 0.328
11	0.41 x 0.8 = 0.328
13	0.15 x 0.2 = 0.030
14	0.15 x 0.2 = 0.030
15	0.35 x 0.2 = 0.070
16	0.35 x 0.2 = 0.070
E =	11.092

Tables 4.17 and 4.18P show the same exact pdf of the project completion time. In the first approach, in order to determine the pdf of project completion time by conditioning at all realization times of common activities the network was solved 2 x 2 = 4 times, while in the second approach at the expense of more computation effort in convolution operations the number of the times which the project needed to be solved was reduced from 4 to 2. Following examples show the efficiency of the second approach.

SECOND EXAMPLE

Consider the Double Wheatstone bridge of Figure 4.8. This network is irreducible and may not be further simplified by convolution and greatest operations. However, suppose we conditionalize by first realization time of A, (T_A^1). Since we may now consider A constant (value of its first realization time), we may reduce network of Figure 4.8 to that of Figure 4.9, where $I=G+T_A^1$ and $J=D+T_A^1$. The pdf's of these random variables are $F_I(t)=F_G(t-T_A^1)$ and $F_J(t)=F_D(t-T_A^1)$, which are now independent. Now J and B are parallel activities and so may be replaced by K as in Figure 4.10, where K has the pdf $F_K(t)=F_B(t)F_D(t-T_A^1)$. Next, suppose we conditionalize by first realization time of C, (T_C^1). The network of Figure 4.10 is then equivalent to that of Figure 4.11, where $L=F+T_C^1$ and $M=H+T_C^1$. Now, however the network of Figure 4.11 is subject to complete series-parallel reduction.

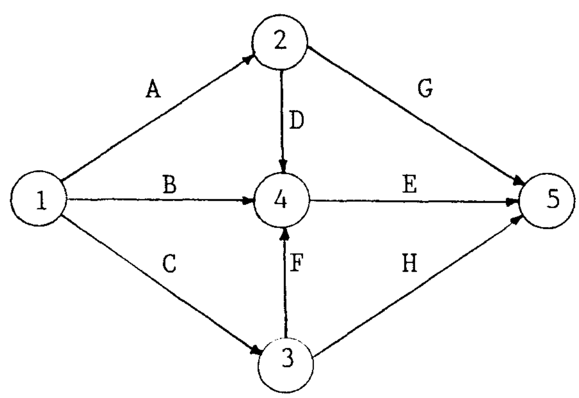


Figure 4.8

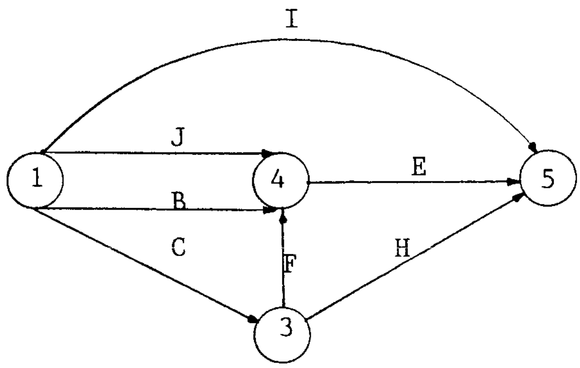


Figure 4.9

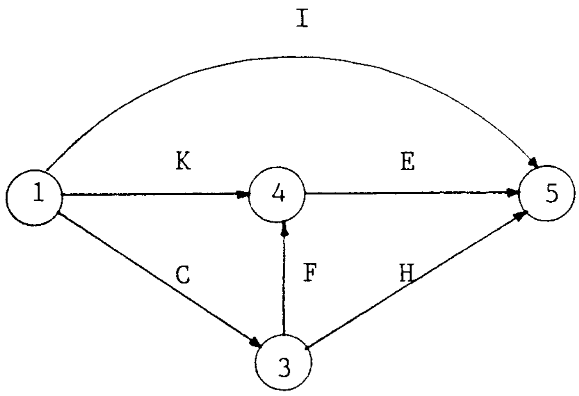


Figure 4.10

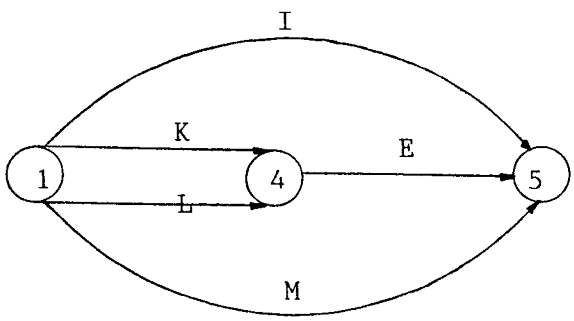


Figure 4.11

In the above example in order to reach a completely reducible network we conditionalized two activities (A and C). Assuming each activity has three realization times, for determination of project completion time the project is needed to be solved $3^2=9$ times, where 3 is the number of the realization times of each conditionalized activity and 2 is the number of the activities which have been conditionalized.

We take advantage of a statement in chapter two to minimize calculation effort in our proposed procedure. Recall that we stated that "if all arrows in a directed acyclic network, such as in PERT, are reversed, the average duration of the project, e_N remains unchanged".
Figure 4.12 shows Figure 4.8 with reversed arrows.

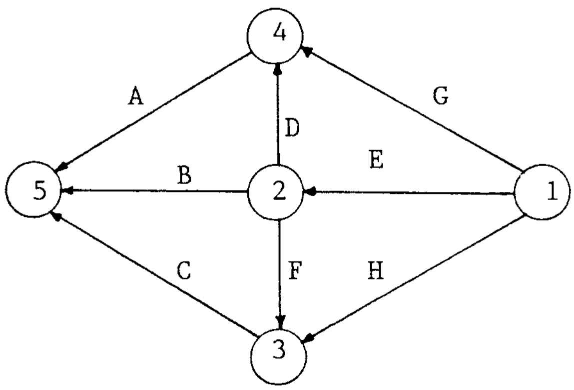


Figure 4.12

Suppose we conditionalized network of Figure 4.12 by first realization time of E, (T_E^1). Since we may now consider E constant (value of its first realization time) we may reduce network of Figure 4.12 to that of Figure 4.13, where $N=D+T_E^1$, $O=B+T_E^1$ and $P=F+T_E^1$. Now, the network of Figure 4.13 is subject to complete series-parallel reduction. Notice that the nodes in Figure 4.12 are numbered such that an arrow leads from a smaller numbered node to a larger one.

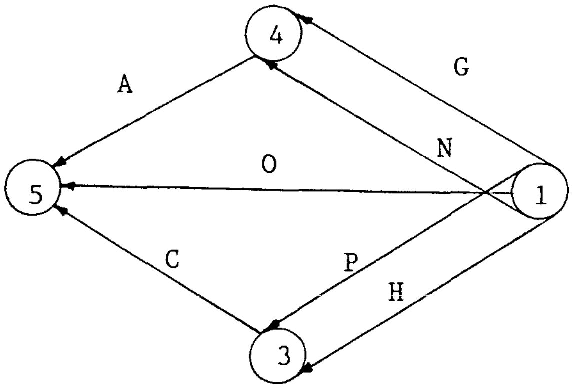


Figure 4.13

Since each activity has three realization times and by conditioning only one activity (E), the project is reached to a completely reducible form, therefore, in order to determine the pdf of project completion time the project is needed to be solved 3 times. Comparing with the 9 times of the previous case (project as given) approximately 67% saving in computation time is very considerable. Notice that since there are 3 common activities (A,C and E) using the first approach (i.e. conditioning at realization times of all common activities) the network is needed to be solved $3^3 = 27$ times. Using CIM approach we need to retain memory of A and C in network as given (Figure 4.8), and memory of E in network with reversed direction of arrows (Figure 4.12).

THIRD EXAMPLE

Consider activity network of Figure 4.14. This network is taken from (Van Slyke, 1963).

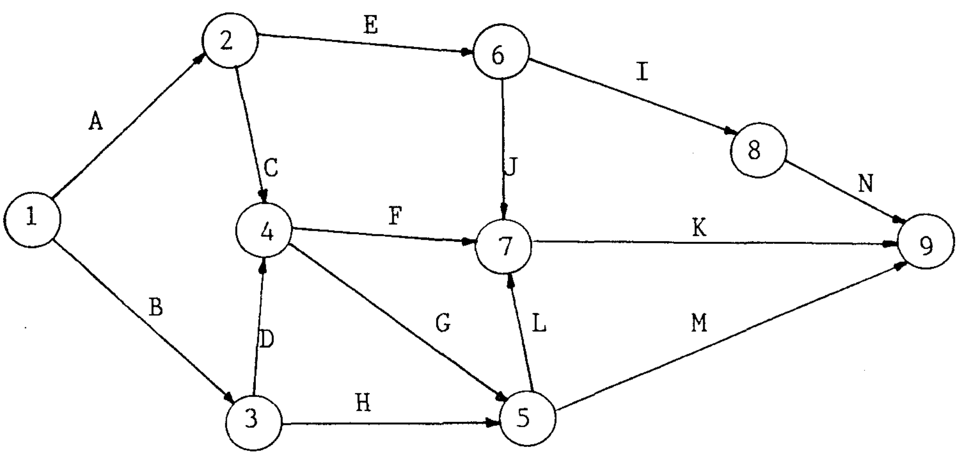
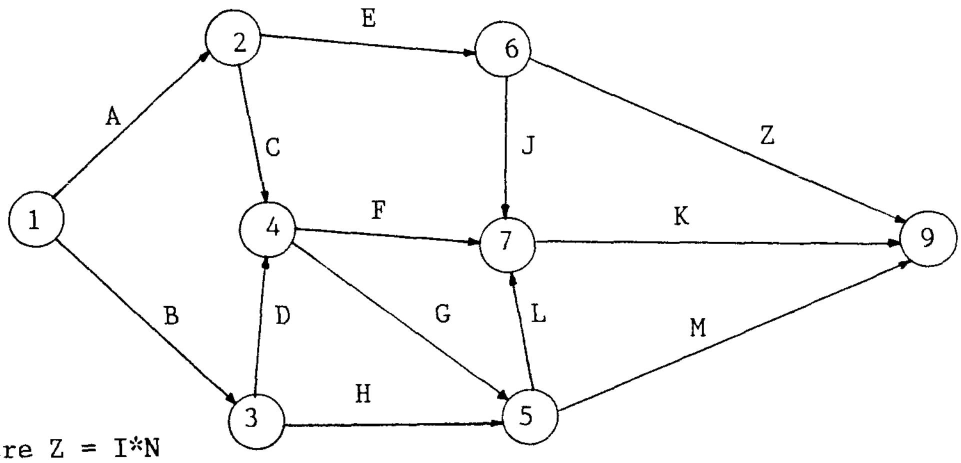


Figure 4.14

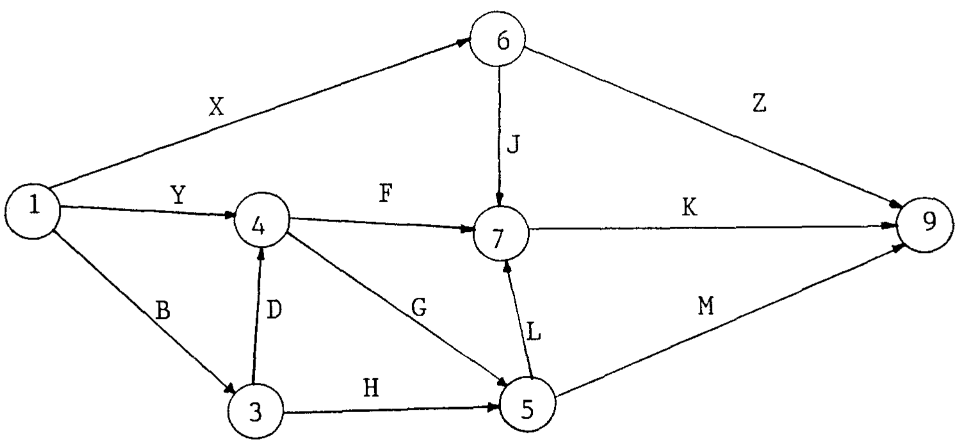
By convoluting I and N, network of Figure 4.14 can be reduced to network of Figure 4.15.



where Z = I*N

Figure 4.15

Following networks show the reduction process of Figure 4.15 by conditioning at first realization time of each appropriate activity in each iteration.



where,

Figure 4.16

$X=T_A^1+E$

$Y=T_A^1+C$

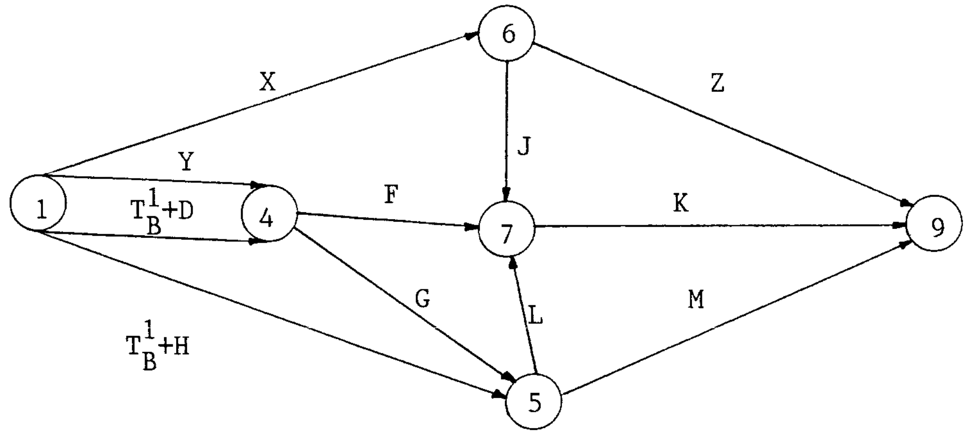


Figure 4.17

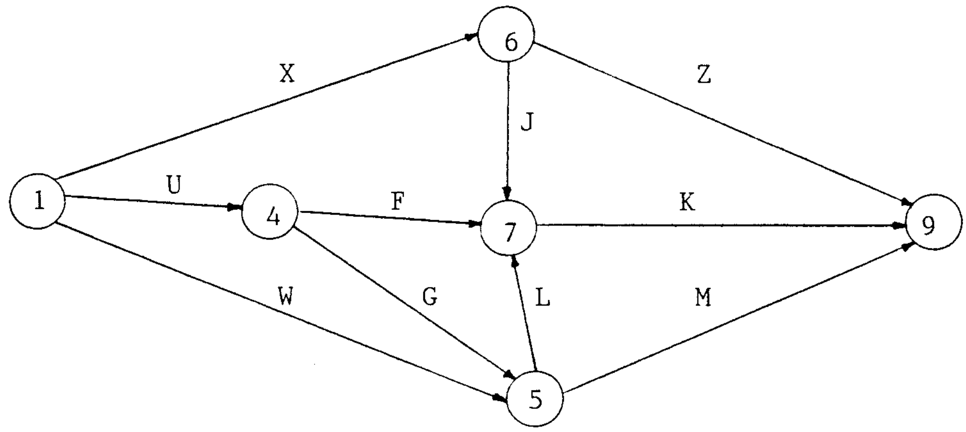


Figure 4.18

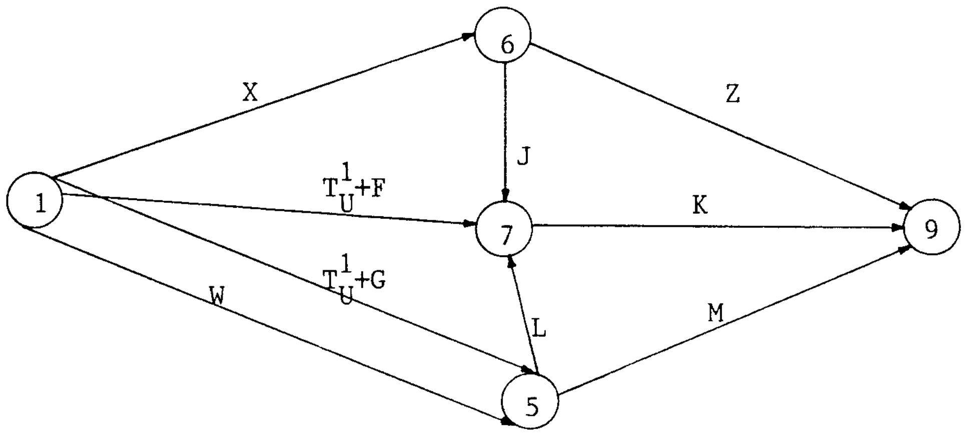


Figure 4.19

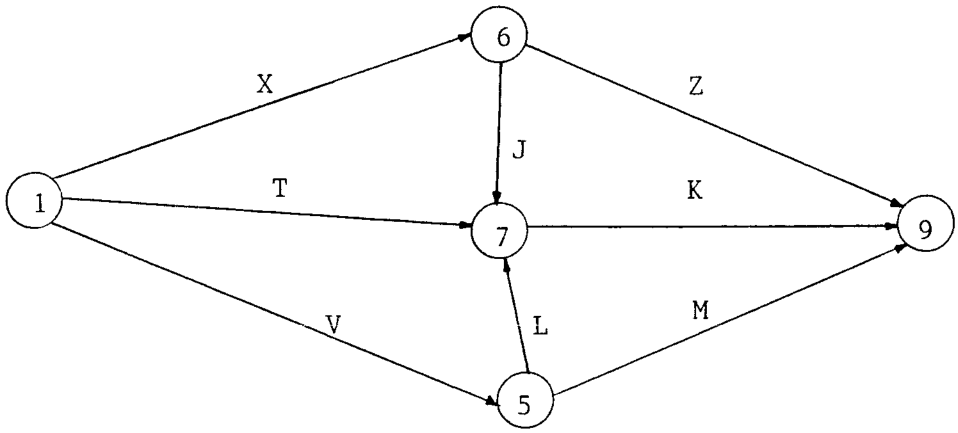


Figure 4.20

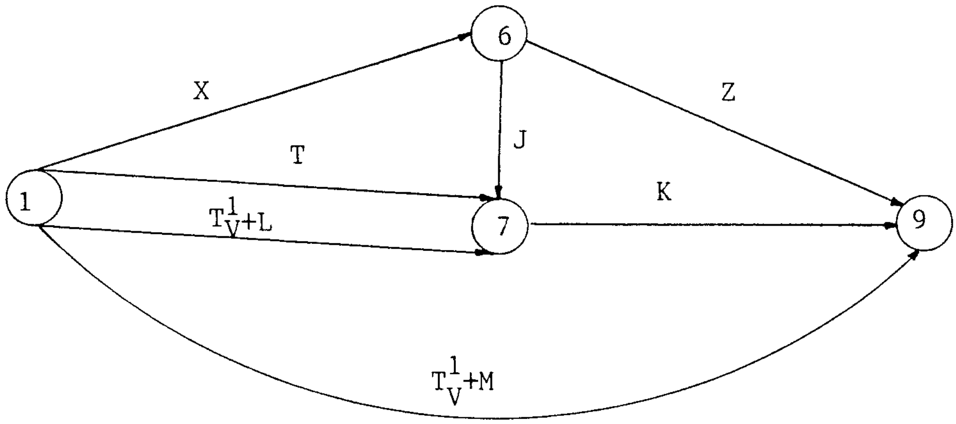


Figure 4.21

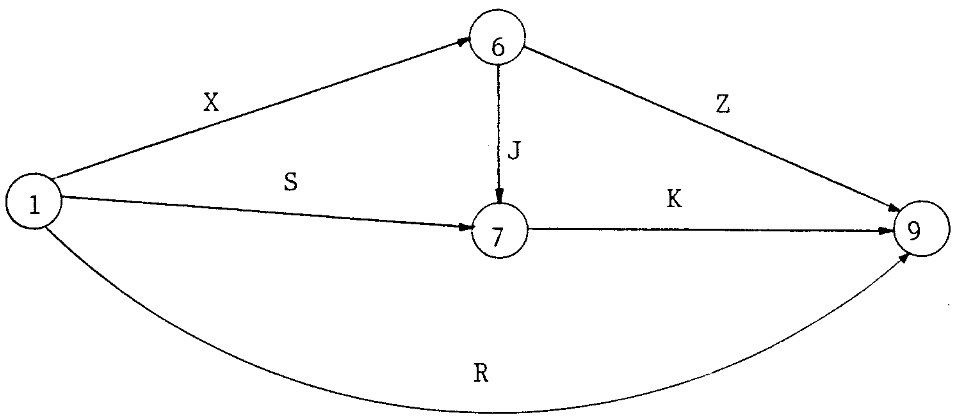


Figure 4.22

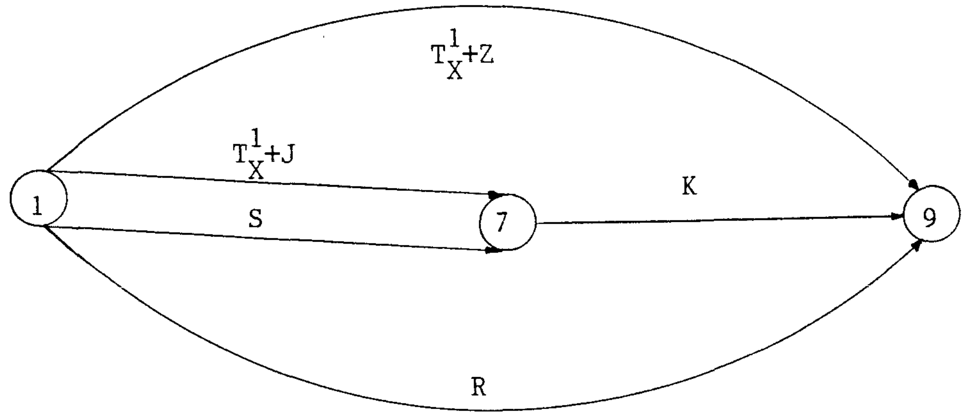


Figure 4.23

Now the network of Figure 4.23 is completely reducible. Using forward pass (network as given) minimum number of activities needed to be conditionalized to reach a completely reducible form is 5 including A,B,U,V and X.

Now let us reverse all arrows of Figure 4.15.

Figrue 2.24 shows Figure 4.15 with reversed arrows.

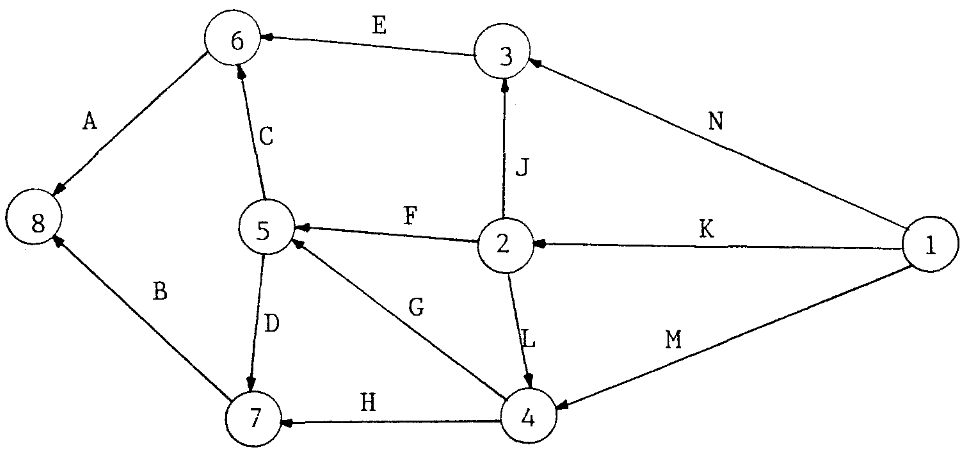


Figure 4.24

The following networks show the reduction process of Figure 4.24 by conditioning at first realization time of each appropriate activity in each iteration. Notice that the nodes in Figure 4.24 are numbered such that an arrow leads from a smaller numbered node to a larger one.

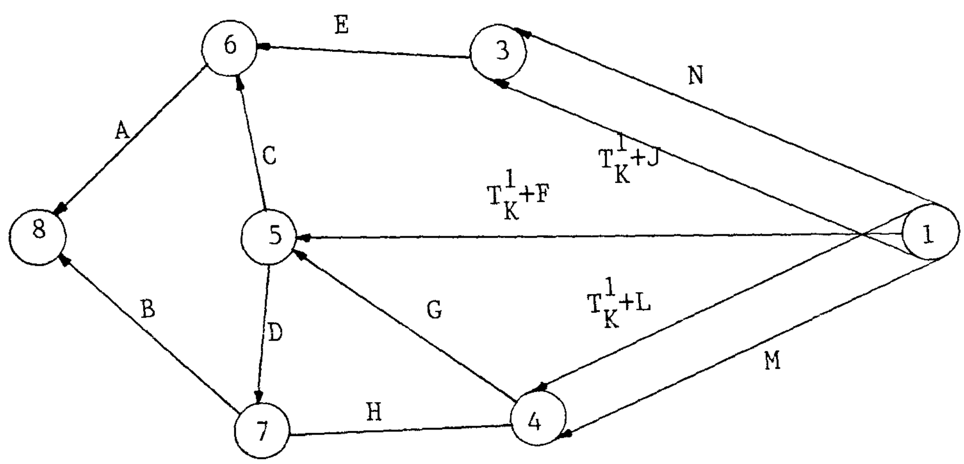


Figure 4.25

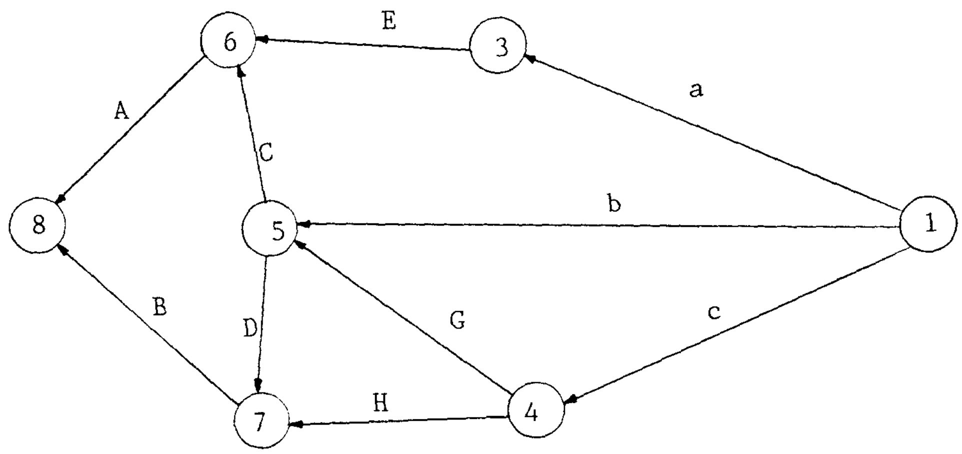


Figure 4.26

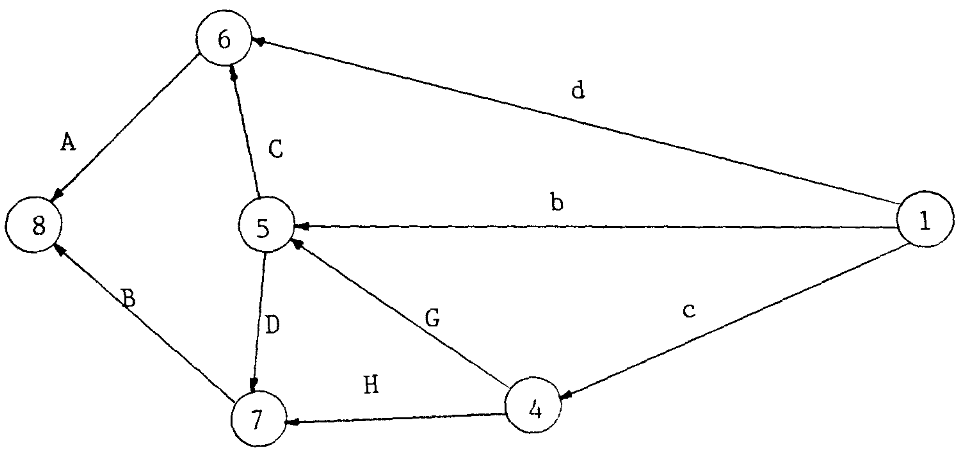


Figure 4.27

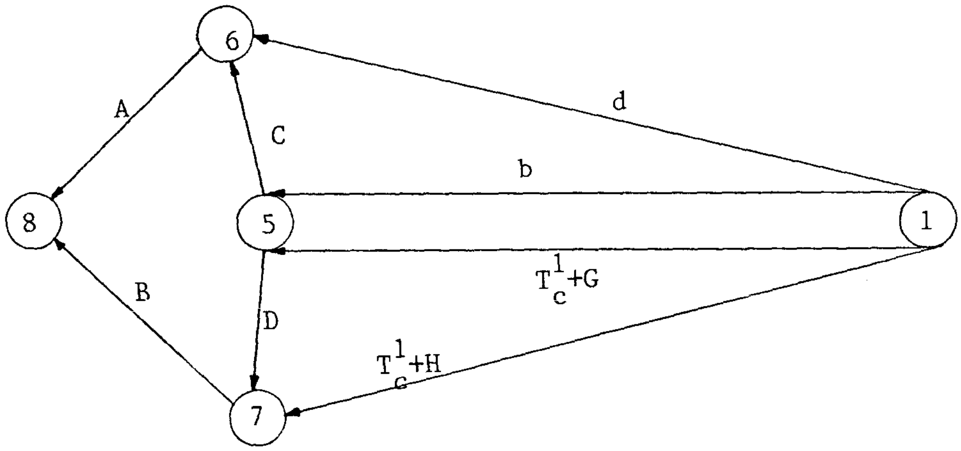


Figure 4.28

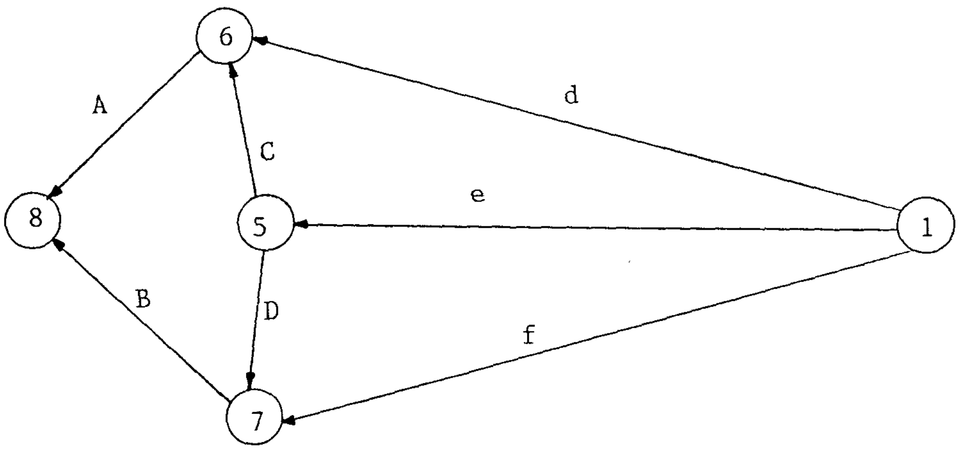


Figure 4.29

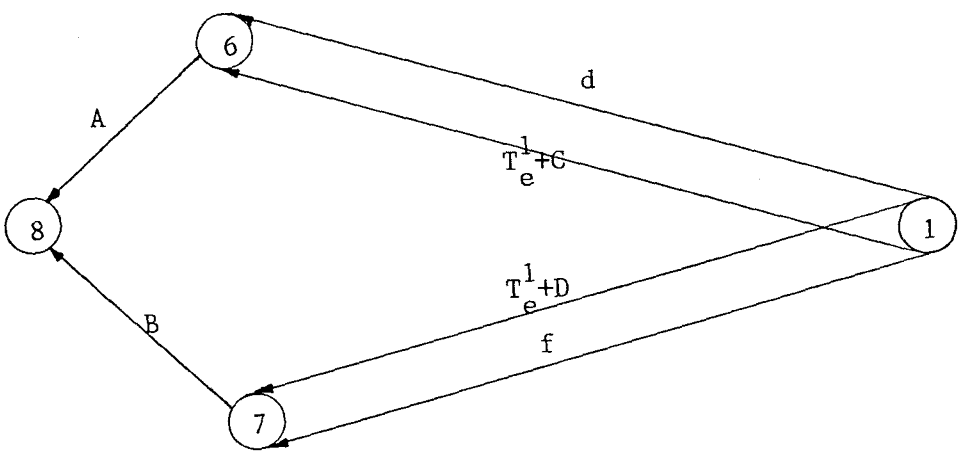


Figure 4.30

The network of Figure 4.30 is completely reducible. Using backward pass minimum number of activities needed to be conditionalized to reach a completely reducible network for this example is 3, including K, c and e. Assuming on average each conditionalized activity has 5 realization times, in forward pass the project needs to be solved 5^5= 3125 times, while using backward pass the project needs to be solved 5^3= 125 times, which needs 4% of the computation effort required in forward pass.

Considering 96% saving in computation effort in this example and approximately 67% in the first example by using backward pass instead of forward pass it is worth testing the project before applying the main algorithm to see the project as given, or the project with reversed arrows needs the minimum number of activities to be conditonalized.

Notice that in Figure 4.15 all activities except activity Z are common activities, therefore, by using the CIM approach we need to retain a memory of activities A,B,C,D,E and H up to node 9. Also in Figure 4.24 by using the CIM approach we need to retain a memory of activities K,M,L,F and G up to node 8.

A GENERAL ALGORITHM

The following section presents the algorithm. The approach is based on Martin's (1965) series parallel reduction of stochastic PERT networks, Ringer's (1969) conditioning on particular activity times and Garman's (1972) conditional sampling of stochastic networks. The general method of implementation is based on the following theorem which has been proved by Garman (1972).

THEOREM 1. Any irreducible network will possess (1) at least one activity 'a' such that 'a' has more than one successor while each of its successors has only 'a' as a predecessor; and (2) at least one activity 'b' such that 'b' has more than one predecessor while each of its predecessors has only 'b' as a successor.

PROOF Consider the set of all activities which emanate from the network's source. If the network is irreducible, there must be at least one of these activities like 'a' which precedes a set of activities which are preceded by no other activity (otherwise, the network would either by cyclic or subject to parallel reduction). If this set consists of only one activity, the network would be subject to series reduction. Therefore we have proved (1); to prove (2), we simply consider activities entering the terminus in analogous fashion.

The reduction process (STEP I) of the following algorithm is developed by Dodin (1985a), this process is also used in Steps III and V with some changes.

THE ALGORITHM

STEP I - Reduce the network to its irreducible form using convolution and greatest operations. The logic used to effect all possible reductions in the network is stated in the following by the name REDUCE.

The reduction process (REDUCE). The following steps are used to identify and perform all the convolution and multiplication operations in the network:

(1) Calculate I(i) and O(i), the indegree and outdegree, for all nodes j=1,2,....,N in the network G(N,A).

(2) If $I(i)+O(i) \geqslant 3$ for all $j \neq 1,N$ then the network is irreducible; stop. If on the other hand $I(i)+O(i)=2$ for at least one $j \neq 1,N$ then the activity network is reducible since a convolution operation is possible; if it is carried out

it might give rise to multiplication and convolution operations. Go to 3.

(3) Use the scanning operator, denoted by SCAN, described below to identify all the convolution and multiplication operations, i.e.

$$\text{SCAN:}\quad G(N,A) \longrightarrow G(N^{(1)},A^{(1)})$$

where $N^{(1)}<N$ and $A^{(1)}<A$. The following are the steps of SCAN:

(a) Rank the arcs in A from 1 to A such that for any two arcs $p=(ij)\epsilon A$ and $q=(lk)\epsilon A$ $p<q$ if $i<l$; if $i=l$ then $j<k$. Set the arc indicator $\delta(p)=1$ for all $p=1,2,\ldots,A$, $M=A$, and $K=1$.

(b) If $\delta(K)=0$, go to e; otherwise let i be the start node of arc K and j be its end node, and set $J=1$.

(c) If $\delta(J)=0$ or $J=K$, go to d; otherwise let α be the start node of arc J and β be its end node, then check for the conditions of either a multiplication or a convolution operation; if $i=\alpha$ and $j=\beta$ then K and J define a multiplication operation; do the necessary bookkeeping, and set $\delta(K)=0$, $\delta(J)=0$, $M \rightarrow M+1$ and $\delta(M)=1$, then go to e. If not, then if either (i) $j=\alpha$ and $I(i)=0(i)=1$, or (ii) $i=\beta$ and $I(i)=0(i)=1$, then K and J define a convolution operation; do the necessary bookkeeping, and set $\delta(K)=0$, $\delta(J)=0$, $M \rightarrow M+1$ and $\delta(M)=1$, then go to e.

(d) Set $J \rightarrow J+1$. If $J>M$, go to e; otherwise go to c.

(e) If $K = M$, stop; otherwise set $K \rightarrow K+1$ and go to b.

If $A^{(1)}=1$ (i.e. the network is reduced to an equivalent activity starting in node 1 and ending in node N) stop. The pdf of the duration of this final activity is equal to F(t).

If $A^{(1)} \neq 1$, then $A^{(1)} \geqslant 5, N^{(1)} \geqslant 4$ and $G(N^{(1)}, A^{(1)})$ is the corresponding irreducible network of G(N,A). In this case the pdf of T_N is obtained through the use of Conditioning Process presented in STEP II. Set (i) → (i+1), KF=0, then go to STEP II.

The reduction process described above starts with a convolution operation, then a sequence of multiplication and convolution operatins may follow in any order. The process may start with any of the initial convolutions available without fear of missing any convolution or multiplication operation; this is clear from step 3 above. The reduction process does not alter the realization of T_N or its pdf.

STEP II - Conditioning Process

(a) All nodes in $N^{(i)}$ are in ascending order i.e. an arc leads from a smaller number node to a larger one. Set the node indicator γ (i')=0 for all i'=immediate succeeding node number of node 1, L'=greatest node number in i', and set node indicator γ(k')=1 for all $k' \epsilon N^{(i)}$, and $k' \neq i'$, I'=2.

(b) If γ(I')=1, go to j;otherwise let I(I') be the indegree of node I' and O(I') be its outdegree, and set J'=2.

(c) If γ(J')=1 or J'=I', go to i; otherwise let I(J') be the indegree of node J' and O(J') be its outdegree, then check for the condition of conditioning operation ; if I(J')>1 or I(J')=I(I')=1 and O(I') $\geqslant$ O(J'), set γ(J')=1 then go to d; otherwise set γ(I')=1, then go to j.

(d) Set J' → J'+1. If J' > L', go to e; otherwise go to c.

(e) Use conditioning operator, denoted by COND, described below to identify conditioning operation, i.e.

$$\text{COND: } G(N^{(i)}, A^{(i)}) \rightarrow G(N^{(i+1)}, A^{(i+1)})$$

where $N^{(i+1)} = N^{(i)} - 1$ and $A^{(i+1)} = A^{(i)} - 1$

The following are the steps of COND: notice that activity $p'=(1I')$ is chosen to be conditionalized.

(f) Determine the set of $A(I')$, which is the set of activities emanating from node I', these activities are in ascending order from I'' to J''. Set the arc indicator $\delta(q')=0$ for all $q'=I'', I''+1,\ldots,J''$, $M''=J''$, $L''=I''$, $K''=I''-1$. Notice that K'' denote number of resultant activity.

(g) If $\delta(L'')=1$, go to h; otherwise for each activity $(I'j'')\epsilon A(I')$, convolute the first realization time of $p'=(1I')$ with $F_{I'j''}(y)$, i.e. $T^1_{p'}$ with the distribution function of $Y_{I'j''}$, denote this convolution by $F^1_{1j''}(y)$.

Therefore, for any value $t > 0$,

$$F^1_{1j''}(y)=Pr(T^1_{p'}+Y_{I'j''}\leqslant t)$$
$$=\sum_y Pr(Y_{I'j''}=y)Pr(T^1_{p'}\leqslant t-y)\ .$$

Do the necessary bookkeeping and go to h.

(h) Set $K'' \rightarrow K''+1$, then if $\delta(L'')=1$ set $\delta(K'')=0$; otherwise set $\delta(K'')=1$. Then if $L''=M''$, set $\delta(p')=0$, $\gamma(I')=0$, $G(N^{(i)},A^{(i)}) \rightarrow G(N^{(i+1)},A^{(i+1)})$, $KF \rightarrow KF+1$, and go to STEP III; otherwise set $L'' \rightarrow L''+1$ and go to g.

(i) Set $J' \rightarrow J'+1$. If $J' > L'$, go to j; otherwise go to c.

(j) If $I'=L'$, go to e; otherwise set $I' \rightarrow I'+1$ and go to b.

STEP III - Reduction Process

Use the scanning operator, denoted by SCAN, described below to identify all the convolution and multiplication operations i.e.

$$SCAN:\ G(N^{(i)},A^{(i)}) \rightarrow G(N^{(i+1)},A^{(i+1)})$$

where $N^{(i+1)} < N^{(i)}$ and $A^{(i+1)}<A^{(i)}$. The following are the steps of SCAN:

(a) For all the arcs in $G(N^{(i)},A^{(i)})$, set the arc indicator $\delta(1')=1$, M=greatest arc number in $A^{(i-1)}$, and K=smallest arc number in $A^{(i)}$.

(b) If $\delta(K)=0$, go to e; otherwise let i be the start node of arc K and j be its end node, and set J=smallest arc number in $A^{(i)}$.

(c) If $\delta(J)=0$ or J=K, go to d; otherwise let α be the start node of arc J and β be its end node, then check for the conditions of either a multiplication or a convolution operation; if $i=\alpha$ and $j=\beta$ then K and J define a multiplication operation; do the necessary bookkeeping, and set $\delta(K)=0$, $\delta(J)=0$, $M \rightarrow M+1$ and $\delta(M)=1$, then go to e. If not, then if either (i) $j=\alpha$ and $I(i)=0(i)=1$ or (ii) $i=\beta$ and $I(i)=0(i)=1$, then K and J define a convolution operation; do the necessary bookkeeping, and set $\delta(K)=0$, $\delta(J)=0$, $M \rightarrow M+1$ and $\delta(M)=1$. $G(N^{(i)},A^{(i)}) \rightarrow G(N^{(i+1)},A^{(i+1)})$, then go to e.

(d) Set $J \rightarrow J+1$. If $J > M$, go to e; otherwise go to c.

(e) If K=M, stop; otherwise set $K \rightarrow K+1$ and go to b .

If $A^{(i+1)}=1$ (i.e., the network is reduced to an equivalent activity starting in node 1 and ending in node N) stop. The pdf of the duration of this final activity is equal to conditional pdf of the project completion time given first realization time of p'.

If $A^{(i+1)}\neq 1$, then set $(i) \rightarrow (i+1)$ and go to STEP II. Every iteration of STEP II - III must reduce the given network by at least one activity. Hence, by induction on the finiteness of the original network, a trivial (i.e. one activity) network must eventually be reached.

STEP IV - Reverse direction of all arrows in the network and renumber nodes such that an arrow leads from a smaller numbered node to a larger one, set KF → KB, then apply STEPS I through III to the network with reversed arrows until the network is reduced to an equivalent activity starting in node 1 and ending in node N.

If KF $\leqslant$ KB apply following steps to the given network; otherwise apply following steps to the network with reversed arrows. KF denotes number of activities which have been conditionalized in the given network and KB denotes number of activities which have been cnditionalized in the network with reversed arrows.

STEP V - Let $T_p^{k'}$ denote kth realization time of activity p' =(1I'), where k=1,....,KI' and MM=KM where KM=min {KF,KB} from STEP IV. Let NF denote number of nodes in the first irreducible network, $G(N^{(i)},A^{(i)})$ denote network configuration indicator, and k denote realization time indicator, set k=1 and apply following iterations to the irreducible network chosen in STEP IV.

(a) Set the node indicator $\gamma(i')=0$ for all i'=immediate succeeding node number of node 1, L'=greatest node number in i', and set node indicator $\gamma(k')=1$ for all $k' \in N^{(i)}$, and $k' \neq i'$, I'=2.

(b) If $\gamma(I')=1$, go to j; otherwise let I(I') be the indegree of node I' and O(I') be its outdegree, and set J'=2.

(c) If $\gamma(J')=1$ or J'=I', go to i; otherwise let I(J') be the indegree of node J' and O(J') be its outdegree, then check for the condition of conditioning operation; if $I(J')>1$ or $I(J')=I(I')=1$ and $O(I') \geqslant O(J')$, set $\gamma(J')=1$ then go to d; otherwise set $\gamma(I')=1$, then go to j.

(d) Set J' → J'+1. If J' > L', go to e; otherwise go to c.

(e) Use conditioning operator, denoted by COND, described below to identify conditioning operation, i.e.

COND: $G(N^{(i)},A^{(i)}) \rightarrow G(N^{(i+1)},A^{(i+1)})$

where $N^{(i+1)}=N^{(i)}-1$ and $A^{(i+1)}=A^{(i)}-1$

The following are the steps of COND:

Notice that activity p'=(1I') is chosen to be conditionalized.

(f) Determine the set of A(I'), which is the set of activities emanating from node I', these activities are in ascending order from I" to J". Set the arc indicator $\delta(q')=0$ for all q'=I", I"+1,.,J"; M"=J"; L"=I"; K"=I"-1. Notice that K" denote number of resultant activitiy.

(g) If $\delta(L")=1$, go to h; otherwise for each activity $(I'j")\epsilon A(I')$, convolute the kth realization time of p'=(1I') with $F_{I'j"}(y)$, i.e. convolute $T^k_{p'}$, with the distribution function of $Y_{I'j"}$, denote this convolution by $F^k_{1j"}(y)$.

Therefore, for any value t > 0,

$$F^k_{1j"}(y)=Pr(T^k_{p'}+Y_{I'j"} \leqslant t)$$

$$= \sum_y Pr(Y_{I'j"}=y). \quad Pr(T^k_{p'} \leqslant t-y) .$$

Do the necessary bookkeeping, and go to h.

(h) Set K" → K"+1, then if $\delta(L")=1$ set $\delta(K")=0$; otherwise set $\delta(K")=1$. Then if L"=M", set $\delta(p')=0$, $\gamma(I')=0$, $G(N^{(i)},A^{(i)}) \rightarrow G(N^{(i+1)},A^{(i+1)})$, MM=MM-1, and go to k; otherwise set L" → L"+1 and go to g.

(i) Set J' → J'+1. If J' > L', go to j; otherwise go to c.

(j) If I'=L', go to e; otherwise set I' →I'+1 and go to b.

(k) Use the scanning operator, denoted by SCAN, described below to identify all the convolution and multiplication operations, i.e.

$$\text{SCAN:} \quad G(N^{(i)},A^{(i)}) \rightarrow G(N^{(i+1)},A^{(i+1)})$$

where $N^{(i+1)} < N^{(i)}$ and $A^{(i+1)} < A^{(i)}$

The following are the steps of SCAN:

(l) For all the arcs in $G(N^{(i)},A^{(i)})$, set the arc indicator $\delta(l')=1$, M=greatest arc number in $A^{(i-1)}$, and K=smallest arc number in $A^{(i)}$.

(m) If $\delta(K)=0$, go to p; otherwise let i be the start node of arc K and j be its end node, and set J=smallest arc number in $A^{(i)}$.

(n) If $\delta(J)=0$ or J=K, go to o; otherwise let α be the start node of arc J and β be its end node, then check for the conditions of either a multiplication or a convolution operation ; if $i=\alpha$ and $j=\beta$ then K and J define a multiplication operation; do the necessary bookkeeping, and set $\delta(K)=0$, $\delta(J)=0$, $M \rightarrow M+1$ and $\delta(M)=1$, then go to p. If not, then if either (i) $j=\alpha$ and $I(i)=0(i)=1$, or (ii) $i=\beta$ and $I(i)=0(i)=1$, then K and J define a convolution operation; do the necessary bookkeeping, and set $\delta(K)=0$, $\delta(J)=0$, $M \rightarrow M+1$ and $\delta(M)=1$, then go to p.

(o) Set $J \rightarrow J+1$. If $J > M$, go to p; otherwise go to n.

(p) If K=M, stop; otherwise set $K \rightarrow K+1$ and go to m. If MM > 0, set $G(N^{(i)},A^{(i)}) \rightarrow G(N^{(i+1)},A^{(i+1)})$, then go to a. If MM=0. The pdf of the duration time of this final activity is equal to pdf of project completion time given kth realization times of activities which have been conditionalized, do the necessary bookkeeping, and set

$G(N^{(i)},A^{(i)}) \rightarrow G(N^{(I)},A^{(I)})$, $k \rightarrow k+1$, and go to q.

(q) If $k > KI'$ go to r; otherwise go to a.

(r) If $N^{(i)}=NF$, go to s; otherwise set $G(N^{(i)},A^{(i)}) \rightarrow$ $G(N^{(I)},A^{(I)})$, i.e. use irreducible network of previous step, and set $k \rightarrow k+1$, then go to a.

(s) Decondition the pdf of the final activities.

(t) Determine the pdf of the project completion time, mean and standard deviation.

FOURTH EXAMPLE

As an example let us apply the proposed algorithm to the PERT network of Figure 4.31.

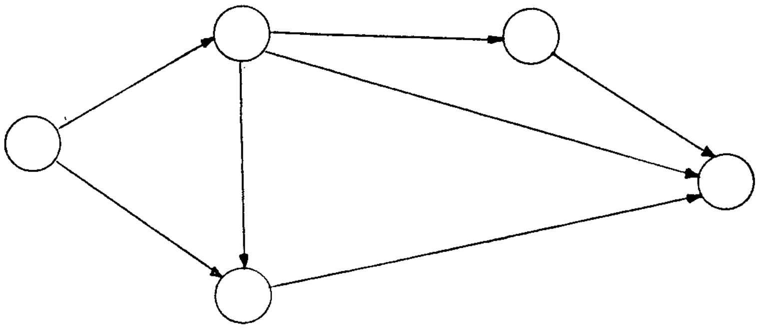


Figure 4.31

This network consists of 5 nodes and 7 activities. In Figure 4.32 the nodes are numbered such that an arc leads from a smaller numbered node to a larger one.

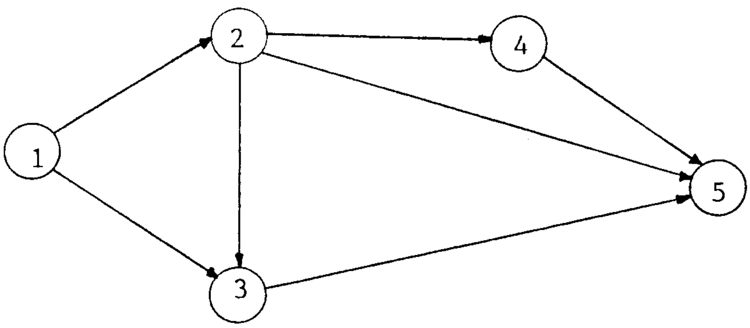


Figure 4.32

The reduction process:

(1) Calculate I(i) and O(i), the indegree and outdegree, for all nodes j=1,2,...N in the network G(N,A).

I(1)=0 ,	I(2)=1 ,	I(3)=2 ,	I(4)=1 ,	I(5)=3
O(1)=2 ,	O(2)=3 ,	O(3)=1 ,	O(4)=1 ,	O(5)=0

(2) Since I(i)+0(i)=2 for node 4, therefore the network is reducible. Scanning operator identifies all the convolution and greatest operations, go to 3.

(3) Use the scanning operator, denoted by SCAN,

$$\text{SCAN:}\quad G(N,A) \rightarrow G(N^{(1)},A^{(1)})$$

(a) Rank the arcs in A from 1 to A such that for any two arcs $p=(ij)\in A$ and $q=(lk)\in A$ $p<q$ if $i<l$; if $i=l$ then $j<k$. Set the arc indicator $\delta(p)=1$ for all $p=1,2,\ldots,A$, $M=A$, $K=1$, and $G(N^{(i)},A^{(i)}) \rightarrow G(N^{(0)},A^{(0)})$.

For two arcs emanating from node 1, $p=(ij)=(1,2)$ and $q=(lk)=(1,3)$ since $i=l$ and $j<k$, rank the arcs such that $(p<q)$ i.e. set $p \rightarrow 1$ and $q \rightarrow 2$. For three arcs (2,3), (2,4) and (2,5) emanating from node 2 since their start nodes is the same, rank the arcs in ascending order of their ending nodes i.e. 3, 4 and 5 respecitvely. By using the same procedure, the algorithm ranks all the arcs as shown in Figure 4.33.

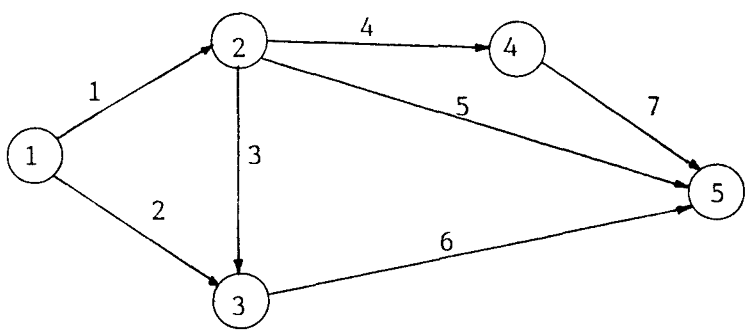


Figure 4.33

Then the algorithm sets the arc indicator $\delta(p)=1$ for all arcs $p=1,2,\ldots,7$, $M=7$, $K=1$ and $G(N^{(i)},A^{(i)}) \rightarrow G(N^{(0)},A^{(0)})$.

(b) If $\delta(K)=0$, go to e; otherwise let i be the start node of arc K and j be its end node, and set $J = 1$.

$\delta(K=1)=1$, $(i=1)$ is the start node of arc 1 and $(j=2)$ is its end node, set $J=1$.

(c) If $\delta(J)=0$ or $J=K$, go to d; otherwise let α be the start node of arc J and β be its end node, then check for the conditions of either a multiplication or a convolution operation; if $i=\alpha$ and $j=\beta$ then K and J define a multiplication operation; do the necessary bookkeeping, and set $\delta(K)=0$, $\delta(J)=0$, $M \rightarrow M+1$ and $\delta(M)=1$, then go to e. If not, then if either (i) $j=\alpha$ and $I(i)=0(i)=1$, or (ii) $i=\beta$ and $I(i)=0(i)=1$, then K and J define a convolution operation; do the necessary bookkeeping, and set $\delta(K)=0$, $\delta(J)=0$, $M \rightarrow M+1$ and $\delta(M)=1$, then go to e.

$\delta(J=1)=1$, but $J=K$, go to d.

(d) Set $J \rightarrow J+1$. If $J > M$, go to e; otherwise go to c.

$\begin{cases} J \rightarrow J+1 \\ 1 \rightarrow 2 \end{cases}$, $(J=2) < (M=7)$, go to c.

(c) $\delta(J=2)=1$, $(\alpha=1)$ is the start node of arc 2 and $(\beta=3)$ is its end node, since $i=\alpha$ but $j \neq \beta$ these two arcs $(K=1)$ and $(J=2)$ don't satisfy conditions of greatest operation, and since $j \neq \alpha$ and $i \neq \beta$ they don't satisfy conditions of convolution operation, go to d.

(d) $\begin{cases} J \rightarrow J+1 \\ 2 \rightarrow 3 \end{cases}$, $(J=3) < (M=7)$, go to c.

(c) $\delta(J=3)=1$, $(\alpha=2)$ is the start node of arc 3 and $(\beta=3)$ is its end node, these two arcs $(K=1)$ and $(J=3)$ don't satisfy conditions of greatest or convolution operations, go to d.

The algorithm repeats this process and in each iteration sets $J \rightarrow J+1$ until all arcs are considered with the first arc (K=1) for the conditions of greatest or convolution operations, go to d.

(d) $\begin{cases} J \rightarrow J+1 \\ 3 \rightarrow 4 \end{cases}$, (J=4) < (M=7), go to c.

(c) δ(J=4)=1, but (K=1) and (J=4) don't satisfy conditions of greatest or convolution operations, go to d. Figure 4.33 shows that arc 1 is not in series or parallel with any other arcs, therefore, this process for K=1 without any reduction in network terminates in e.

(e) (K=1) < (M=7) ===> $\begin{cases} K \rightarrow K+1 \\ 1 \rightarrow 2 \end{cases}$, go to b.

(b) δ(K=2)=1, (i=1) is the start node of arc 2 and (j=3) is its end node, set J=1. Now arc K=2 is considered with the other arcs for the conditions of greatest or convolution operations. This operation for K=2 without any reduction of network terminates in e.

(e) (K=2) < (M=7) ===> $\begin{cases} K \rightarrow K+1 \\ 2 \rightarrow 3 \end{cases}$, go to b.

(b) δ(K=3)=1, (i=2) is the start node of arc 3 and (j=3) is its end node, set J=1. Now arc K=3 is considered with the other arcs for the conditions of greatest or convolution operations, again this operation for K=3 without any reduction of network terminates in e.

(e) (K=3) < (M=7) $\begin{cases} K \rightarrow K+1 \\ 3 \rightarrow 4 \end{cases}$, go to b.

(b) δ(K=4)=1, (i=2) is the start node of arc 4 and (j=4) is its end node, set J=1, go to c.

(c) δ(J=1)=1, (α=1) is the start node of arc 1 and (β=2) is its end node, these two arcs (K=4) and (J=1) don't satisfy conditions of greatest or convolution operations, go to d.

The algorithm repeats this process and in each iteration sets J → J+1. K=4 and any of J ⩽ 6 don't satisfy conditions of greatest or convolution operations, go to d.

(d) $\begin{cases} J \rightarrow J+1 \\ 6 \rightarrow 7 \end{cases}$, (J=7) is not greater than (M=7), go to c.

(c) δ(J=7)=1, (α=4) is the start node of arc 7 and (β=5) is its end node, since j=α=4 and I(4)=0(4)=1, K=4 and J=7 define a convolution operation; do the necessary bookkeeping, and set δ(K=4)=0, δ(J=7)=0, $\begin{cases} M \rightarrow M+1 \\ 7 \rightarrow 8 \end{cases}$, δ(M=8)=1, $G(N^{(i)},A^{(i)}) \rightarrow G(N^{(i+1)},A^{(i+1)})$, then go to e. Notice that arc 8 is the resultant of the two activities in series (K=4 and J=7) as shown in Figure 4.34.

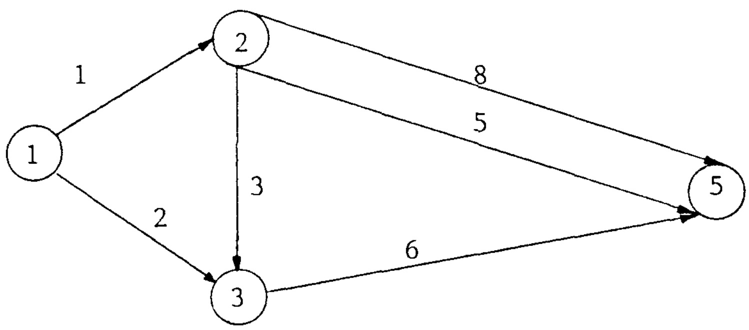


Figure 4.34

(e) (K=4) < (M=8) ===> $\begin{cases} K \rightarrow K+1 \\ 4 \rightarrow 5 \end{cases}$, go to b.

(b) δ(K=5)=1, (i=2) is the start node of arc 5 and j=5 is its end node, set J=1.

(c) δ(J=1)=1, (α=1) is the start node of arc 1 and (β=2) is its end node, these two arcs (K=5) and (J=1) don't satisfy conditions of greatest or convolution operations, go to d. The algorithm repeats this process and in each iteration sets J → J+1, until all arcs are considered with arc (K=5).

(d) $\begin{cases} J \rightarrow J+1 \\ 1 \rightarrow 2 \end{cases}$, (J=2) < (M=8), go to c.

(c) δ(J=2)=1, (α=1) is the start node of arc 2 and (β=3) is its end node, these two arcs (K=5) and (J=2) don't satisfy conditions of greatest or convolution operations go to d.

(d) $\begin{cases} J \rightarrow J+1 \\ 2 \rightarrow 3 \end{cases}$, (J=3) < (M=8), go to c.

(c) δ(J=3)=1, (α=2) is the start node of arc 3 and (β=3) is its end node, K=5 and J=3 don't satisfy conditions of greatest or convolution operations go to d.

(d) $\begin{cases} J \rightarrow J+1 \\ 3 \rightarrow 4 \end{cases}$, (J=4) < (M=8), go to c.

(c) δ(J=4)=0, go to d.

(d) $\begin{cases} J \rightarrow J+1 \\ 4 \rightarrow 5 \end{cases}$, (J=5) < (M=8), go to c.

(c) δ(J=5)$\neq$ 0, but K=J, go to d.

(d) $\begin{cases} J \rightarrow J+1 \\ 5 \rightarrow 6 \end{cases}$, (J=6) < (M=8), go to c.

(c) δ(J=6)=1, (α=3) is the start node of arc 6 and (β=5) is its end node. K=5 and J=6 don't satisfy conditions of greatest or convolution operations go to d.

(d) $\begin{cases} J \rightarrow J+1 \\ 6 \rightarrow 7 \end{cases}$, (J=7) < (M=8), go to c.

(c) δ(J=7)=0, go to d.

(d) $\begin{cases} J \rightarrow J+1 \\ 7 \rightarrow 8 \end{cases}$, (J=8) is not greater than (M=8), go to c.

(c) δ(J=8)=1, (α=2) is the start node of arc 8 and (β=5) is its end node, since i=α=2 and j=β=5, K=5 and J=8 define a greatest operation; do the necessary bookkeeping, and set δ(K=5)=0, δ(J=8)=0, $\begin{cases} M \rightarrow M+1 \\ 8 \rightarrow 9 \end{cases}$, δ(M=9)=1,

$G(N^{(i)},A^{(i)}) \rightarrow G(N^{(i+1)},A^{(i+1)})$, then go to e.

Notice that arc 9 is the resultant of the two parallel activities K=5 and J=8 as shown in Figure 4.35.

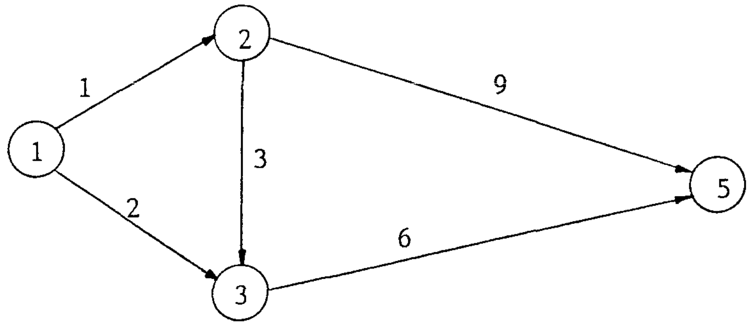


Figure 4.35

(e) (K=5) < (M=9) ===> $\begin{cases} K \rightarrow K+1 \\ 5 \rightarrow 6 \end{cases}$, go to b.

(b) δ(K=6)=1, (i=3) is the start node of arc 6 and j=5 is its end node, set J=1.

(c) δ(J=1)=1, (α=1) is the start node of arc 1 and (β=2) is its end node, K=6 and J=1 don't satisfy conditions of greatest or convolution operations, go to d.

(d) $\begin{cases} J \rightarrow J+1 \\ 1 \rightarrow 2 \end{cases}$, (J=2) < (M=9), go to c.

(c) δ(J=2)=1, (α=1) is the start node of arc 2 and (β=3) is its end node, K=6 and J=2 don't satisfy conditions of greatest or convolution operations, go to d.

(d) $\begin{cases} J \rightarrow J+1 \\ 2 \rightarrow 3 \end{cases}$, (J=3) < (M=9), go to c.

(c) δ(J=3)=1, (α=2) is the start node of arc 3 and (β=3) is its end node, K=6 and J=3 don't satisfy conditions of greatest or convolution operations, go to d.

(d) $\begin{cases} J \rightarrow J+1 \\ 3 \rightarrow 4 \end{cases}$, (J=4) < (M=9), go to c.

(c) δ(J=4)=0, go to d.

(d) $\begin{cases} J \rightarrow & J+1 \\ 4 \rightarrow & 5 \end{cases}$, (J=5) < (M=9), go to c.

(c) δ(J=5)=0, go to d.

(d) $\begin{cases} J \rightarrow & J+1 \\ 5 \rightarrow & 6 \end{cases}$, (J=6) < (M=9), go to c.

(c) δ(J=6)≠0, but J=K, go to d.

(d) $\begin{cases} J \rightarrow & J+1 \\ 6 \rightarrow & 7 \end{cases}$, (J=7)<(M=9) go to c.

(c) δ(J=7)=0, go to d.

(d) $\begin{cases} J \rightarrow & J+1 \\ 7 \rightarrow & 8 \end{cases}$, (J=8) < (J=9), go to c.

(c) δ(J=8)=0, go to d.

(d) $\begin{cases} J \rightarrow & J+1 \\ 8 \rightarrow & 9 \end{cases}$, (J=9) is not greater than (M=9), go to c.

(c) δ(J=9)=1, (α=2) is the start node of arc 9 and (β=5) is its end node, K=6 and J=9 don't satisfy conditions of greatest or convolution operations, go to d.

(d) $\begin{cases} J \rightarrow & J+1 \\ 9 \rightarrow & 10 \end{cases}$, (J=10) > (M=9), go to e.

(e) (K=6) < (M=9) ===> $\begin{cases} K \rightarrow & K+1 \\ 6 \rightarrow & 7 \end{cases}$, go to b.

(b) δ(K=7)=0, go to e.

(e) (K=7) < (M=9) ===> $\begin{cases} K \rightarrow & K+1 \\ 7 \rightarrow & 8 \end{cases}$, go to b.

(b) δ(K=8)=0, go to e.

(e) (K=8) < (M=9) ===> $\begin{cases} K \rightarrow & K+1 \\ 8 \rightarrow & 9 \end{cases}$, go to b.

(b) δ(K=9)=1, (i=2) is the start node of arc 9 and (j=5) is its end node, set J=1, go to c.

(c) δ(J=1)=1, (α=1) is the start node of arc 1 and (β=2) is its end node, K=9 and J=1 don't satisfy conditions of greatest or convolution operations, go to d.

(d) $\begin{cases} J \rightarrow & J+1 \\ 1 \rightarrow & 2 \end{cases}$, (J=2) < (M=9), go to c.

(c) $\delta(J=2)=1$, $(\alpha=1)$ is the start node of arc 2 and $(\beta=3)$ is its end node, K=9 and J=2 don't satisfy conditions of greatest or convolution operations, go to d.

(d) $\begin{cases} J \rightarrow & J+1 \\ 2 \rightarrow & 3 \end{cases}$, $(J=3) < (M=9)$, go to c.

(c) $\delta(J=3)=1$, $(\alpha=2)$ is the start node of arc 3 and $(\beta=3)$ is its end node, K=9 and J=3 don't satisfy conditions of greatest or convolution operations, go to d.

(d) $\begin{cases} J \rightarrow & J+1 \\ 3 \rightarrow & 4 \end{cases}$, $(J=4) < (M=9)$, go to c.

(c) $\delta(J=4)=0$, go to d.

(d) $\begin{cases} J \rightarrow & J+1 \\ 4 \rightarrow & 5 \end{cases}$, $(J=5) < (M=9)$, go to c.

(c) $\delta(J=5)=0$, go to d.

(d) $\begin{cases} J \rightarrow & J+1 \\ 5 \rightarrow & 6 \end{cases}$, $(J=6) < (M=9)$, go to c.

(c) $\delta(J=6)=1$, $(\alpha=3)$ is the start node of arc 6 and $(\beta=5)$ is its end node, K=9 and J=6 don't satisfy conditions of greatest or convolution operations, go to d.

(d) $\begin{cases} J \rightarrow & J+1 \\ 6 \rightarrow & 7 \end{cases}$, $(J=7) < (M=9)$, go to c.

(c) $\delta(J=7)=0$, go to d.

(d) $\begin{cases} J \rightarrow & J+1 \\ 7 \rightarrow & 8 \end{cases}$, $(J=8) < (M=9)$, go to c.

(c) $\delta(J=8)=0$, go to d.

(d) $\begin{cases} J \rightarrow & J+1 \\ 8 \rightarrow & 9 \end{cases}$, $(J=9)$ is not greater than $(M=9)$, go to c.

(c) $\delta(J=9) \neq 0$, but J=K, go to d.

(d) $\begin{cases} J \rightarrow & J+1 \\ 9 \rightarrow & 10 \end{cases}$, $(J=10) > (M=9)$, go to e.

(e) $(K=9)=(M=9)$. Stop.

Since $A^{(i)} \neq 1$, $G(N^{(i)}, A^{(i)})$ is the corresponding irreducible network of G(N,A). In this case the pdf of T_5 is obtained through the use of Conditioning Process presented in STEP II. Notice that all nodes in Figure 4.35 are in ascending order, i.e., an arc leads from a smaller numbered node to a larger one. Set KF=0 (KF denotes number of activities which have been conditionalized up to now).

STEP II - Conditioning Process

(a) Set the node indicator $\gamma(i')=0$ for all i'=immediate succedding node number of node 1, L'=greatest node number in i', and set node indicator $\gamma(k')=1$ for all $k' \in N^{(i)}$, and $k' \neq i'$, I'=2.

$\gamma(2)=0$, $\gamma(3)=0$, L'=3, $\gamma(1)=1$, $\gamma(5)=1$, I'=2.

(b) If $\gamma(I')=1$, go to j; otherwise let I(I') be the indegree of node I' and O(I') be its outdegree, and set J'=2.

$\gamma(I'=2)=0$, $\begin{cases} I(I'=2)=1 \\ O(I'=2)=2 \end{cases}$, J'=2.

(c) If $\gamma(J')=1$ or J'=I', go to i; otherwise let I(J') be the indegree of node J' and O(J') be its outdegree, then check for the condition of conditioning operation; if I(I')=1 and $O(I') \geqslant O(J')$, set $\gamma(J')=1$ then go to d; otherwise set $\gamma(I')=1$, then go to j.

$\gamma(J'=2)=0$, but I'=J', go to i.

(i) $\begin{cases} J' \rightarrow J'+1 \\ 2 \rightarrow 3 \end{cases}$, (J'=3) is not greater than (L'=3), go to c.

(c) $\gamma(J'=3)=0$, I(J'=3)=2, and O(J'=3)=1, since I(I'=2)=1 and (O(I'=2) =2) > (O(J'=3)=1), therefore, set $\gamma(J'=3)=1$, go to d.

(d) Set $J' \to J'+1$. If $J' > L'$ go to e; otherwise go to c.

$\begin{cases} J' \to J'+1 \\ 3 \to 4 \end{cases}$, $(J'=4) > (L'=3)$, go to e.

(e) Use conditioning operator, denoted by COND, described below to identify conditioning operation, i.e.

$$\text{COND:} \quad G(N^{(i)}, A^{(i)}) \quad \to \quad G(N^{(i+1)}, A^{(i+1)})$$

where $\quad N^{(i+1)} = N^{(i)}-1$ and $A^{(i+1)}=A^{(i)}-1$

Notice that activity $p'=1=(1,2)$ is chosen to be conditionalized.

(f) Determine the set of $A(I')$, which is the set of activities emanating from node I', these activities are in ascending order from I'' to J''.

Set the arc indicator $\delta(q')=0$ for all $q'=I'', I''+1,\ldots,J''$

$M''=J''$, $L''=I''$, $K''=I''-1$.

Notice that K'' denotes number of resultant activity.

$A(I'=2)=\{3,9\}$, $\delta(3)=0$, $\delta(9)=0$, $M''=9$, $L''=3$, $K''=3-1=2$.

(g) If $\delta(L'')=1$, go to h; otherwise for each activity $(I'j'')\epsilon A(I')$, convolute the first realization time of $p'=(1I')$ with $F_{I'j''}(y)$ i.e. $T^1_{p'}$, with the distribution function of $Y_{I'j''}$, denote this convolution by $F^1_{1j''}(y)$.

Therefore, for any value $t>0$,

$$F^1_{1j''}(y)=\Pr(T^1_{p'}+Y_{I'j''}\leqslant t)$$
$$= \sum_y \Pr(Y_{I'j''}=y)\ \Pr(T^1_{p'}\leqslant t-y).$$

Do the necessary bookkeeping, and go to h.

$\delta(L''=3)=0$, therefore, convolute $T^1_{1=(1,2)}$ with the distribution function of $Y_{2,3}$ denote this convolution by $F^1_{1,3}(y)$.

$$F^1_{1,3}(y)=\Pr(T^1_1 + Y_{2,3}\leqslant t)$$
$$= \sum_y \Pr(Y_{2,3}=y).\Pr(T^1_1 \leqslant t-y).$$

Do the necessary bookkeeping and go to h.

(h) Set K" → K"+1, then if δ(L")=1 set δ(K")=0; otherwise set δ (K")=1. Then if L"=M", set δ(p')=0, γ(I')=0, $G(N^{(i)},A^{(i)}) \rightarrow G(N^{(i+1)},A^{(i+1)})$, KF → KF+1, and go to STEP III; otherwise set L" → L"+1 and go to g.

$\begin{cases} K'' \rightarrow K''+1 \\ 2 \rightarrow 3 \end{cases}$, δ(L"=3)=0==> δ(K"=3)=1,

(L"=3)≠ (M=9) ===> $\begin{cases} L'' \rightarrow L''+1 \\ 3 \rightarrow 4 \end{cases}$, go to g.

Notice that K" is a new arc as shown in Figure 4.36.

(g) δ(L"=4)=1, go to h.

(h) $\begin{cases} K'' \rightarrow K''+1 \\ 3 \rightarrow 4 \end{cases}$, δ(L"=4)=1 ===> δ(K"=4)=0,

(L"=4)≠(M"=9) ===> $\begin{cases} L'' \rightarrow L''+1 \\ 4 \rightarrow 5 \end{cases}$, go to g.

(g) δ(L"=5)=1 , go to h.

(h) $\begin{cases} K'' \rightarrow K''+1 \\ 4 \rightarrow 5 \end{cases}$, δ(L"=5)=1 ===> δ(K"=5)=0,

(L"=5)≠(M"=9) ===> $\begin{cases} L'' \rightarrow L''+1 \\ 5 \rightarrow 6 \end{cases}$, go to g.

(g) δ(L"=6)=1 , go to h.

(h) $\begin{cases} K'' \rightarrow K''+1 \\ 5 \rightarrow 6 \end{cases}$, δ(L"=6)=1 ===> δ(K"=6)=0,

(L"=6)≠(M"=9) ===> $\begin{cases} L'' \rightarrow L''+1 \\ 6 \rightarrow 7 \end{cases}$, go to g.

(g) δ(L"=7)=1 , go to h.

(h) $\begin{cases} K'' \rightarrow K''+1 \\ 6 \rightarrow 7 \end{cases}$, δ(L"=7)=1 ===> δ(K"=7)=0,

(L"=7)≠(M"=9) ===> $\begin{cases} L'' \rightarrow L''+1 \\ 7 \rightarrow 8 \end{cases}$, go to g.

(g) δ(L"=8)=1 , go to h.

(h) $\begin{cases} K'' \rightarrow K''+1 \\ 7 \rightarrow 8 \end{cases}$, δ(L"=8)=1 ===> δ(K"=8)=0,

$(L''=8) \neq (M''=9) \Longrightarrow \begin{cases} L'' \to & L''+1 \\ 8 \to & 9 \end{cases}$, go to g.

(g) $\delta(L''=9)=0$, convolute T_1^1 , with the distribution function of $Y_{2,5}$, denote this convolution by $F_{1,5}^1(y)$.

$$F_{1,5}^1(y) = \Pr\ (T_1^1 + Y_{2,5} \leqslant t)$$
$$= \sum_y \Pr\ (Y_{2,5}=y) .\ \Pr(T_1^1 \leqslant t-y) .$$

Do the necessary bookkeeping, and go to h.

(h) $\begin{cases} K'' \to & K''+1 \\ 8 \to & 9 \end{cases}$, $\delta(L''=9)=0 \Longrightarrow \delta(K''=9)=1$,

$(L''=9)=(M''=9) \Longrightarrow$ set, $\delta(p'=1)=0$, $\gamma(I'=2)=0$,

$(G(N^{(i)},A^{(i)})=G(4,5)) \to (G(N^{(i+1)},A^{(i+1)}) = G(3,4))$,

$\begin{cases} KF & \to & KF+1 \\ 0 & \to & 1 \end{cases}$, and then go to STEP III.

Notice that KF denotes number of activities which have been conditionalized up to now.

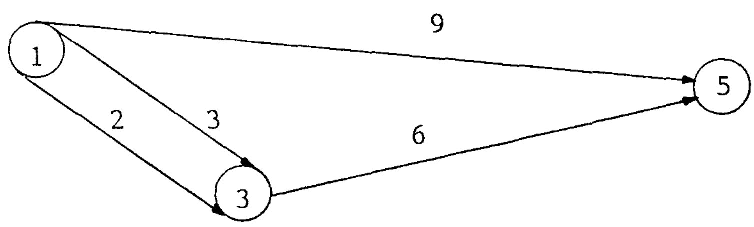


Figure 4.36

STEP III - Reduction Process

Use the scanning operator to identify all the convolution and multiplication operations.

SCAN: $G(N^{(i)},A^{(i)}) \to G(N^{(i+1)},A^{(i+1)})$

where $N^{(i+1)} < N^{(i)}$ and $A^{(i+1)} < A^{(i)}$

(a) $\delta(2)=1$, $\delta(3)=1$, $\delta(6)=1$, $\delta(9)=1$, M=9, K=2.

(b) $\delta(K=2)=1$, (i=1) is the start node of arc 2 and (j=3) is its end node, set J=2, go to c.

(c) $\delta(J=3)=1$, $(\alpha=1)$ is the start node of arc 3 and $(\beta=3)$ is its end node, since $i=\alpha=1$ and $j=\beta=3$, therefore, activities 2 and 3 define a multiplication operation, do the necessary bookkeeping, and set $\delta(K=2)=0$,

$\delta(J=3)=0$, $\begin{cases} M \to M+1 \\ 9 \to 10 \end{cases}$, $\delta(M=10)=1$,

$(G(N^{(i)},A^{(i)}) = G(3,4)) \to (G(N^{(i+1)},A^{(i+1)}) = G\ (3,3))$, then go to e. Notice that arc 10 is the resultant arc of the two parallel arcs 2 and 3 of Figure 4.36 as shown in Figure 4.37.

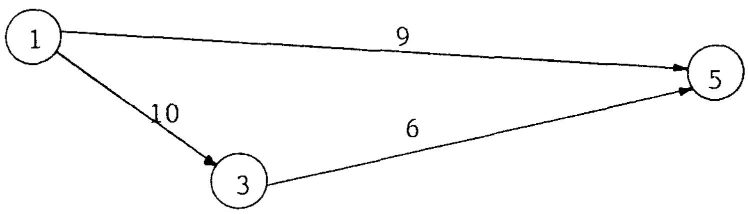


Figure 4.37

(e) $(K=2)\neq(M=10)$ ===> $\begin{cases} K \to K+1 \\ 2 \to 3 \end{cases}$, go to b.

(b) $\delta(K=3)=0$, go to e.

(e) $(K=3)\neq(M=10)$ ===> $\begin{cases} K \to K+1 \\ 3 \to 4 \end{cases}$, go to b.

(b) $\delta(K=4)=0$, go to e.

(e) $(K=4)\neq(M=10)$ ===> $\begin{cases} K \to K+1 \\ 4 \to 5 \end{cases}$, go to b.

(b) $\delta(K=5)=0$, go to e.

(e) $(K=5)\neq(M=10)$ ===> $\begin{cases} K \to K+1 \\ 5 \to 6 \end{cases}$, go to b.

(b) $\delta(K=6)=1$, $(i=3)$ is the start node of arc 6 and $(j=5)$ is its end node, set J=6 and go to c.

(c) $\delta(J=6)=1$, but J=K, go to d.

(d) $\begin{cases} J \rightarrow J+1 \\ 6 \rightarrow 7 \end{cases}$, $(J=7) < (M=10)$, go to c.

(c) $\delta(J=7)=0$, go to d.

(d) $\begin{cases} J \rightarrow J+1 \\ 7 \rightarrow 8 \end{cases}$, $(J=8) < (M=10)$, go to c.

(c) $\delta(J=8)=0$, go to d.

(d) $\begin{cases} J \rightarrow J+1 \\ 8 \rightarrow 9 \end{cases}$, $(J=9) < (M=10)$, go to c.

(c) $\delta(J=9)=1$, $(\alpha=1)$ is the start node of arc 9 and $(\beta=5)$ is its end node, K=6 and J=9 don't satisfy conditions of greatest or convolution operations go to d.

(d) $\begin{cases} J \rightarrow J+1 \\ 9 \rightarrow 10 \end{cases}$, $(J=10)$ is not greater than $(M=10)$, go to c.

(c) $\delta(J=10)=1$, $(\alpha=1)$ is the start node of arc 10 and $(\beta=3)$ is its end node, since $i=\beta=3$, activities K=6 and J=10 define a convolution operation; do the necessary bookkeeping, and set $\delta(K=6)=0$, $\delta(J=10)=0$,

$\begin{cases} M \rightarrow M+1 \\ 10 \rightarrow 11 \end{cases}$, $\delta(M=11)=1$, $(G(N^{(i)},A^{(i)}=G(3,3)) \rightarrow$

$(G(N^{(i+1)},A^{(i+1)}) = G(2,2))$,

then go to e. Notice that arc 11 is the resultant arc of the two arcs in series (K=6) and (J=10) of Figure 4.37 as shown in Figure 4.38.

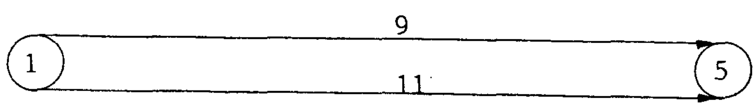


Figure 4.38

(e) $(K=6)\neq(M=11)$ ===> $\begin{cases} K \to K+1 \\ 6 \to 7 \end{cases}$, go to b.

(b) $\delta(K=7)=0$, go to e.

(e) $(K=7)\neq(M=11)$ ===> $\begin{cases} K \to K+1 \\ 7 \to 8 \end{cases}$, go to b.

(b) $\delta(K=8)=0$, go to e.

(e) $(K=8)\neq(M=11)$ ===> $\begin{cases} K \to K+1 \\ 8 \to 9 \end{cases}$, go to b.

(b) $\delta(K=9)=1$, $(i=1)$ is the start node of arc 9 and $(j=5)$ is its end node, set J=9 and go to c.

(c) $\delta(J=9)\neq 0$, but J=K , go to d.

(d) $\begin{cases} J \to J+1 \\ 9 \to 10 \end{cases}$, $(J=10) < (M=11)$, go to c.

(c) $\delta(J=10)=0$, go to d.

(d) $\begin{cases} J \to J+1 \\ 10 \to 11 \end{cases}$, $(J=11)$ is not greater than $(M=11)$, go to c.

(c) $\delta(J=11)=1$, $(\alpha=1)$ is the start node of arc 11 and $(\beta=5)$ is its end node, since $i=\alpha=1$ and $j=\beta=5$, therefore, activities 9 and 11 define a multiplication operation, do the necessary bookkeeping, and set $\delta(K=9)=0$,

$\delta(J=11)=0$, $\begin{cases} M \to M+1 \\ 11 \to 12 \end{cases}$, $\delta(M=12)=1$,

$(G(N^{(i)},A^{(i)})=G(2,2)) \to (G(N^{(i+1)},A^{(i+1)})=G(2,1))$, then go to e.

Notice that arc 12 is the resultant arc of the two parallel arcs K=9 and J=11 of Figure 4.38 as shown in Figure 4.39

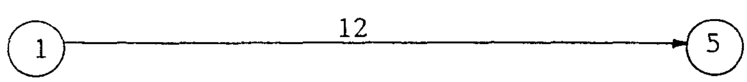


Figure 4.39

(e) $(K=9) \neq (M=12)$ ===> $\begin{cases} K \rightarrow K+1 \\ 9 \rightarrow 10 \end{cases}$, go to b.

(b) $\delta(K=10)=0$, go to e.

(e) $(K=10) \neq (M=12)$ ===> $\begin{cases} K \rightarrow K+1 \\ 10 \rightarrow 11 \end{cases}$, go to b.

(b) $\delta(K=11)=0$, go to e.

(e) $(K=11) \neq (M=12)$ ===> $\begin{cases} K \rightarrow K+1 \\ 11 \rightarrow 12 \end{cases}$, go to b.

(b) $\delta(K=12)=1$, $(i=1)$ is the start node of arc 12 and $(j=5)$ is its end node, set J=12 and go to c.

(c) $\delta(J=12) \neq 0$, but J=K, go to d.

(d) $\begin{cases} J \rightarrow J+1 \\ 12 \rightarrow 13 \end{cases}$, $(J=13)>(M=12)$, go to e.

(e) $(K=12)=(M=12)$. STOP.

STEP IV - Reverse direction of all arrows in the network and renumber nodes such that an arrow leads from a smaller numbered node to a larger one, set KF → KB, then apply STEPS I through III to the network with reversed arrows until the network is reduced to an equivalent activity starting in node 1 and ending in node N.

Figure 4.40 shows Figure 4.31 with reverse direction of arrows.

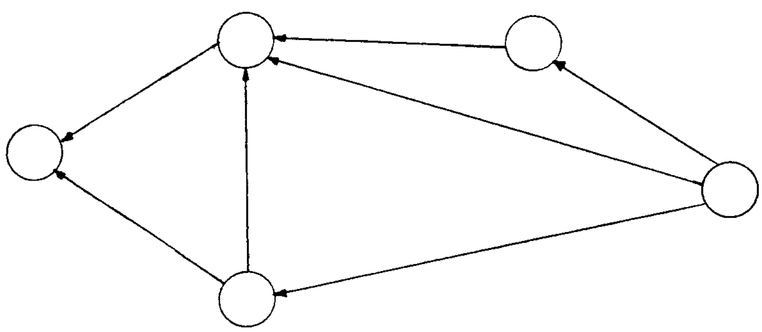


Figure 4.40

In Figure 4.41 nodes are numbered such that an arc leads from a smaller numbered node to a larger one.

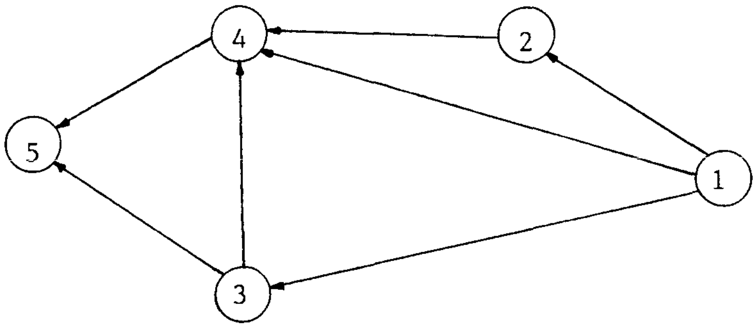


Figure 4.41

First step of scanning operation ranks the arcs as shown in Figure 4.42. Notice that $\overleftarrow{G}(N,A)$ denotes activity network with reverse direction of arrows.

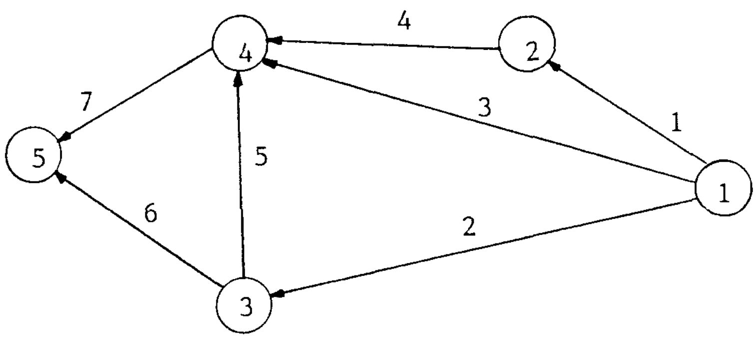


Figure 4.42

Activities 1 and 4 of Figure 4.42 are series, activity 8 is the resultant of these two activities as shown in Figure 4.43.

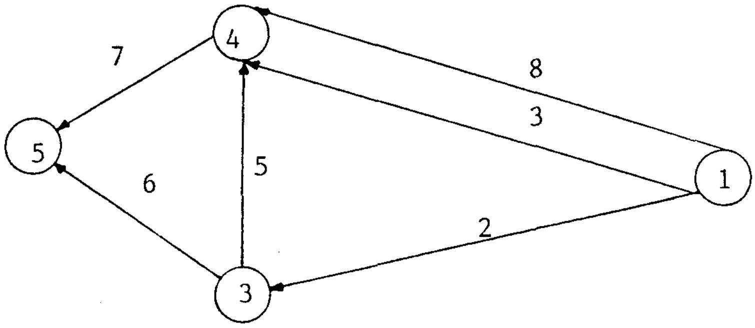


Figure 4.43

Activities 3 and 8 of Figure 4.43 are parallel, activity 9 is the resultant of these two parallel activities as shown in Figure 4.44.

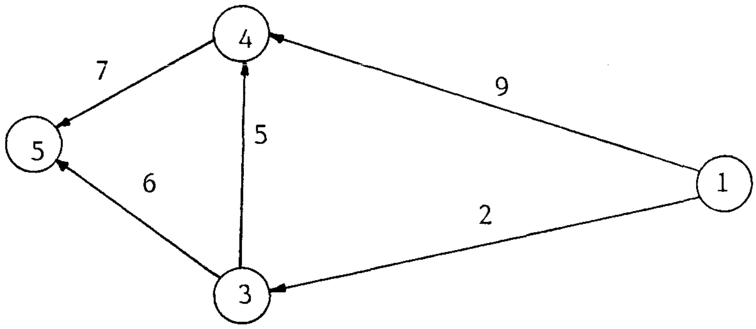


Figure 4.44

Now,activity network of Figure 4.44 is irreducible, therefore, conditioning operator of STEP II identifies activity p'=2=(1,3) to be conditionalized. Figure 4.45 shows activity network of Figure 4.44 by conditioning activity p'=2 at its first realization time.

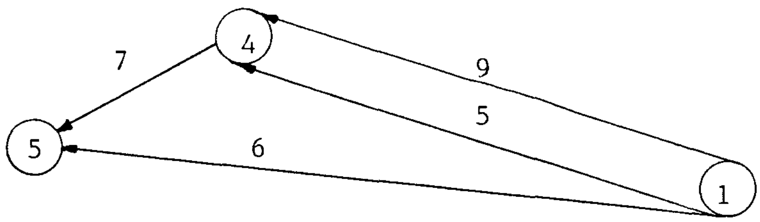


Figure 4.45

Now, reduction process of STEP III identifies all greatest and convolution operations of Figure 4.45 and applies all appropriate iterations until the network is reduced to a single activity as shown in Figure 4.48.

In Figure 4.46 activity 10 is the resultant activity of the two parallel activities 5 and 9 of Figure 4.45.

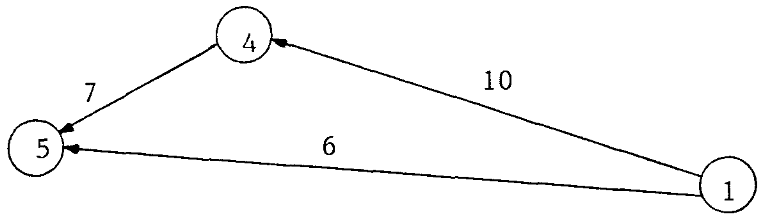


Figure 4.46

In Figure 4.47 activity 11 is the resultant activity of the two activities in series 10 and 7 of Figure 4.46.

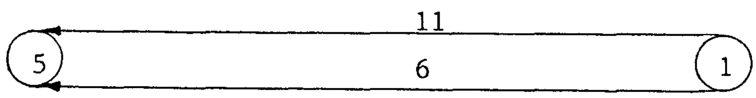


Figure 4.47

In Figure 4.48 activity 12 is the resultant activity of the two parallel activities 6 and 11 of Figure 4.47.

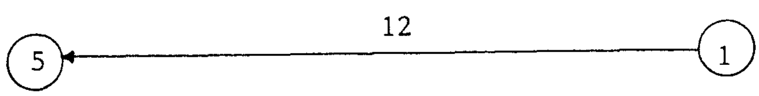


Figure 4.48

Notice that in the network with reverse direction of arrows KB=1 at final stage, therefore, since KF=KB, the algorithm applies following STEPS to the network as given.

STEP V - Let, $T_{p'}^{k}$ denote kth realization time of activity p'=(1I'), where k=1,...., KI' and MM=KM where KM=min {KF, KB} from STEP IV. Let NF denote number of nodes in the first irreducible network of project which is indicated by $G(N^{(I)},A^{(I)})$, and let k denote the realization time indicator, set k=1 and apply following iterations to the irreducible network chosen in STEP IV.

KM=min {KF=1, KB=1} =1, the first irreducible network of the network as given is shown in Figure 4.35.

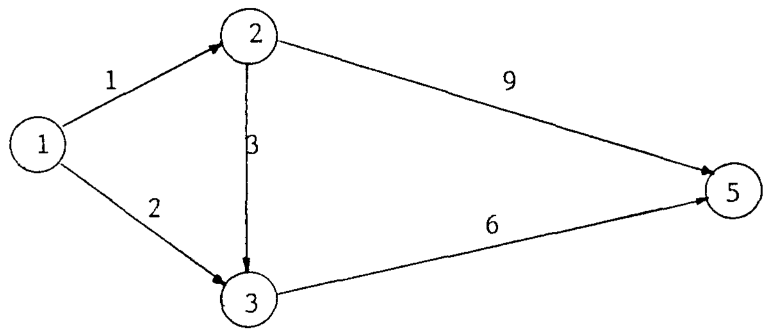


Figure 4.35

NF=4, $G(N^{(I)},A^{(I)})=G(4,5)$. When k=1, the algorithm applies all iterations of STEP V from (a) through (p), in (p) since for this network MM=0, the pdf of the duration time of the final activity is equal to the pdf of project completion time given first realization time of activity 1 (activity which has been conditionalized), do the necessary bookkeeping, and set $G(N^{(i)},A^{(i)}) \rightarrow G(N^{(I)},A^{(I)})$, i.e. use the previous irreducible network, and set $k \rightarrow k+1$ i.e. set $T_1^1 \rightarrow T_1^2$, and go to q.

(q) If k > KI' go to r; otherwise go to a.

(r) If $N^{(i)}$=NF, go to s; otherwise set $G(N^{(i)},A^{(i)} \rightarrow G(N^{(I)},A^{(I)})$, i.e. use irreducible network of previous step, and set $k \rightarrow k+1$ then go to a.

(s) Decondition the pdf of the final activities.

(t) Determine the pdf of the project completion time, mean and standard deviation.

FIFTH EXAMPLE

In the following, the proposed procedure is applied to the network configuration of Figure 4.49 with complete enumeration. Network configuration of Figure 4.49 is an irreducible network, all activities except D and G are common. For ease of calculation assume all activities have discrete distributions with two realization times.

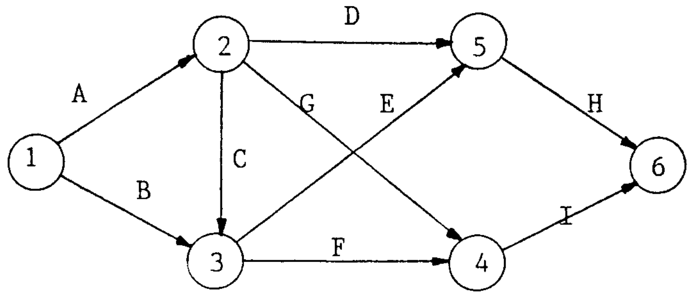


Figure 4.49

Tables 4.19A to 4.19I respectively show duration times of activities A to I.

Table 4.19A: Duration time of A.

	P	CP
2	0.2	0.2
5	0.8	1.0

Table 4.19B: Duration time of B.

	P	CP
3	0.4	0.4
4	0.6	1.0

Table 4.19C: Duration time of C.

1	0.6	0.6
2	0.4	1.0

Table 4.19D: Duration time of D.

	P	CP
3	0.9	0.9
5	0.1	1.0

Table 4.19E: Duration time of E.

	P	CP
4	0.5	0.5
7	0.5	1.0

Table 4.19F: Duration time of F.

	P	CP
3	0.1	0.1
4	0.9	1.0

Table 4.19G: Duration time of G.

	P	CP
3	0.7	0.7
7	0.3	1.0

Table 4.19H: Duration time of H.

	P	CP
3	0.6	0.6
5	0.4	1.0

Table 4.19I: Duration time of I.

	P	CP
4	0.3	0.3
6	0.7	1.0

Figure 4.49 shows that A has three successors (D,G and C) and each of these successors has only A as a predecessor, therefore by fixing on the first realization time of A, 2,changes the network of Figure 4.49 to that of Figure 4.50.

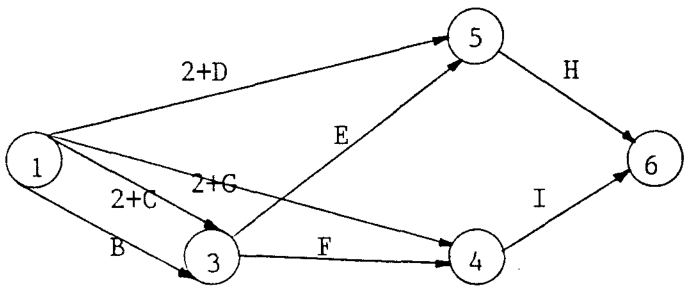


Figure 4.50

Tables 4.20A through 4.20C show duration times of (2+C), (2+D) and (2+G) respectively.

Table 4.20A: Duration time of (2+C).

	P	CP
3	0.6	0.6
4	0.4	1.0

Table 4.20B: Duration time of (2+D).

	P	CP
5	0.9	0.9
7	0.1	1.0

Table 4.20C: Duration time of (2+G).

	P	CP
5	0.7	0.7
9	0.3	1.0

Taking the maximum of B and (2+C) gives K as shown in Figure 4.51.

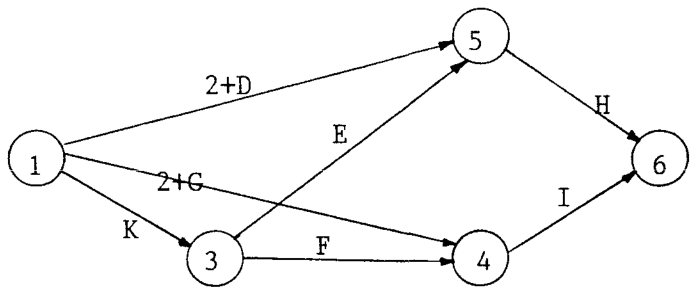


Figure 4.51

Table 4.21 shows duration time of K.

Table 4.21: Duration time of K.

	P	CP
3	0.4 x 0.6 = 0.24	0.24
4	1 x 1 - 0.4 x 0.6 = 0.76	1.0

Figure 4.51 shows that K has two successors (E and F) and each of these successors has only K as a predecessor, by fixing on the first realization time of K,3,changes the network of Figure 4.51 to that of Figure 4.52.

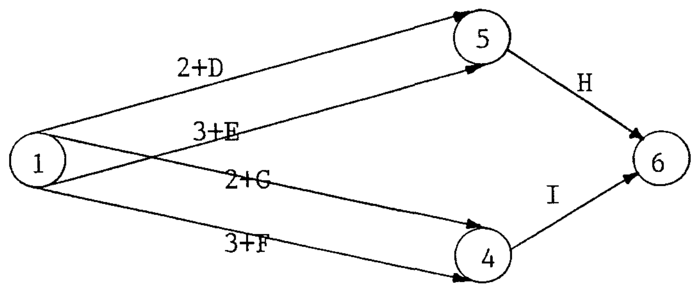


Figure 4.52

Tables 4.22A and 4.22B show duration times of (3+E) and (3+F) respectively.

Table 4.22A: Duration time of (3+E).

	P	CP
7	0.5	0.5
10	0.5	1.0

Table 4.22B: Duratin time of (3+F).

	P	CP
6	0.1	0.1
7	0.9	1.0

Taking the maximum of (2+D) and (3+E) gives L and taking the maximum of (2+G) and (3+F) gives M as shown in Figure 4.53.

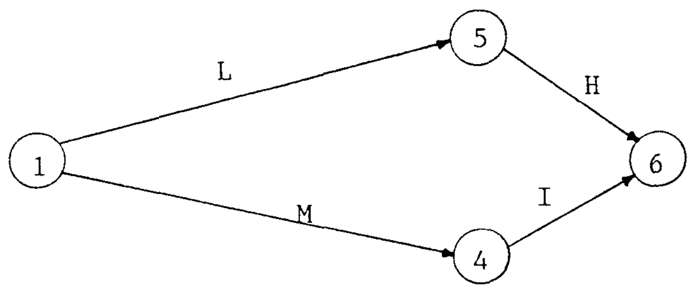


Figure 4.53

Tables 4.23A and 4.23B show duration times of L and M respectively.

Table 4.23A: Duration time of L.

	P		CP
7	1 x 0.5	= 0.5	0.5
10	1 x 1 - 0.5 x 1	= 0.5	1.0

Table 4.23B: Duration time of M.

	P		CP
6	0.1 x 0.7	= 0.07	0.07
7	1 x 0.7 - 0.1 x 0.7	= 0.63	0.70
9	1 x 1 - 1 x 0.7	= 0.30	0.10

Convoluting L and H gives N, and also convoluting M and I gives O as shown in Figure 4.54.

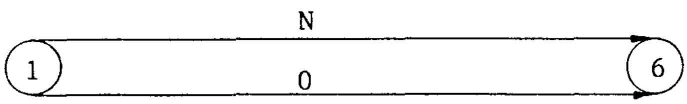


Figure 4.54

Duration time of N is shown in Table 4.24A and duration time of O is shown in Table 4.24B.

Table 4.24A: Duration time of N.

	P	CP
10	0.6 x 0.5 = 0.3	0.3
12	0.4 x 0.5 = 0.2	0.5
13	0.6 x 0.5 = 0.3	0.8
15	0.4 x 0.5 = 0.2	1.0

Table 4.24B: Duration time of O.

	P		CP
10	0.3 x 0.07	= 0.021	0.021
11	0.3 x 0.63	= 0.189	0.210
12	0.7 x 0.07	= 0.049	0.259
13	0.7x 0.63 + 0.3 x 0.3	= 0.531	0.790
15	0.7 x 0.3	= 0.210	1.0

Taking the maximum of N and O gives P as shown in Figure 4.55.

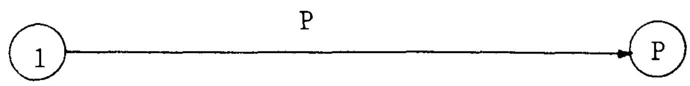


Figure 4.55

Table 4.25 shows the project finish time given A=2 and K=3, where P(A=2)=.2 and P(K=3)=.24 .

Table 4.25: Project finish time | (A=2 and K=3).

10	0.3 x 0.021	= 0.0063
11	0.21 x 0.3 - 0.3 x 0.021	= 0.0567
12	0.259 x 0.5 - 0.210 x 0.3	= 0.0665
13	0.790 x 0.8 - 0.259 x 0.5	= 0.5025
15	1 x 1 - 0.790 x 0.8	= 0.3680

By fixing on the second realization time of K, 4,changes the network of Figure 4.51 to that of Figure 4.56.

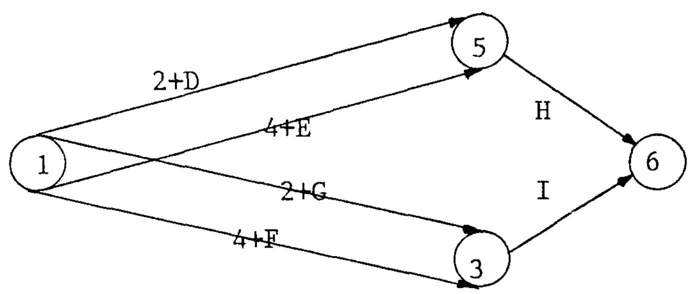


Figure 4.56

Tables 4.26A and 4.26B show duration times of (4+E) and (4+F) respectively.

Table 4.26A: Duration time of (4+E).

	P	CP
8	0.5	0.5
11	0.5	1.0

Table 4.26B: Duration time of (4+F).

	P	CP
7	0.1	0.1
8	0.9	1.0

Taking the maximum of (2+D) and (4+E) gives Q and taking the maximum of (2+G) and (4+F) gives R as shown in Figure 4.57.

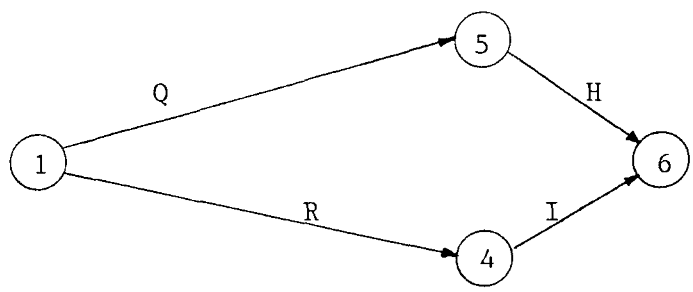


Figure 4.57

Tables 4.27A and 4.27B show duration times of Q and R respectively.

Table 4.27A: Duration time of Q.

	P	CP
8	0.5 x 1 = 0.5	0.5
11	1 x 1 - 0.5 x 1 = 0.5	0.5

Table 4.27B: Duration time of R.

	P	CP
7	0.1 x 7 = 0.07	0.07
8	1 x 0.7 - 0.1 x 0.7 = 0.63	0.70
9	1 x 1 - 1 x 0.7 = 0.30	1.0

Convoluting Q and H gives S, and also convoluting R and I gives T as shown in Figure 4.58.

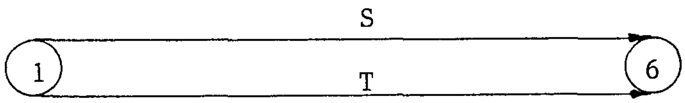


Figure 4.58

Duration time of S is shown in Table 4.28A and duration time of T is shown in Table 4.28B.

Table 4.28A: Duration time of S.

	P	CP
11	0.5 x 0.6 = 0.3	0.3
13	0.5 x 0.4 = 0.2	0.5
14	0.5 x 0.6 = 0.3	0.8
16	0.5 x 0.4 = 0.2	1.0

Table 4.28B: Duration time of T.

	P	CP
11	0.3 x 0.7 = 0.021	0.021
12	0.3 x 0.63 = 0.189	0.210
13	0.7 x 0.07 + 0.3 x 0.3 = 0.139	0.349
14	0.7 x 0.63 = 0.441	0.790
15	0.7 x 0.3 = 0.210	1.0

Taking the maximum of S and T gives U as shown in Figure 4.59.

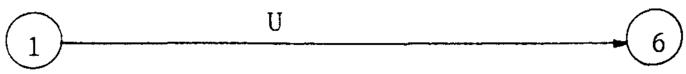


Figure 4.59

Table 4.29 shows the project finish time given A=2 and K=4, where P(A=2)=.2 and P(K=4)=.76 ·

Table 4.29: Project finish time|(A=2, K=4).

11	0.021 x 0.3 = 0.0063
12	0.210 x 0.3 - 0.021 x 0.3 = 0.0567
13	0.349 x 0.5 - 0.210 x 0.3 = 0.1115
14	0.790 x 0.8 - 0.349 x 0.5 = 0.4575
15	1 x 0.8 - 0.790 x 0.8 = 0.1680
16	1 x 1 - 1 x 0.8 = 0.2000

Tables 4.25 and 4.29 show the pdf of the project completion time for K=3 and K=4 respectively given A=2 . 4 is the last realization time of K. By fixing on the second realization time of A, 5,changes the network of Figure 4.49 to that of Figure 4.60.

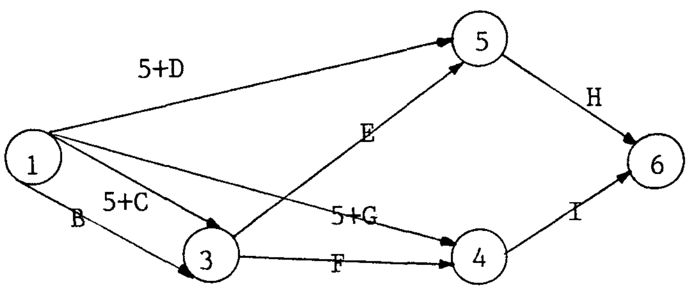


Figure 4.60

Tables 4.30A through 4.30C show duration times of (5+C), (5+D) and (5+G) respectively.

Table 4.30A: Duration time of (5+C).

	P	CP
6	0.6	0.6
7	0.4	1.0

Table 4.30B: Duration time of (5+D).

	P	CP
8	0.9	0.9
10	0.1	1.0

Table 4.30C: Duration time of (5+G).

	P	CP
8	0.7	0.7
12	0.3	1.0

Takin the maximum of B and (5+C) gives V as shown in Figure 4.61.

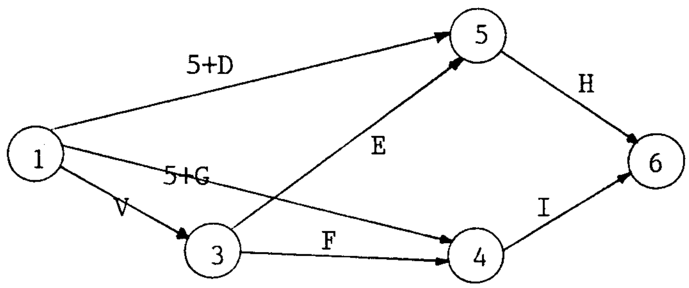


Figure 4.61

Table 4.31 shows duration time of V.

Table 4.31: Duration time of V.

	P		CP
6	0.6 x 1	0.6	0.6
7	1 x 1 - 0.6 x 1 = 0.4		1.0

Figure 4.61 shows that V has two successors (E and F) and each of these successors has only V as a predecessor, by fixing on the first realization time of V, 6,changes the network of Figure 4.61 to that of Figure 4.62.

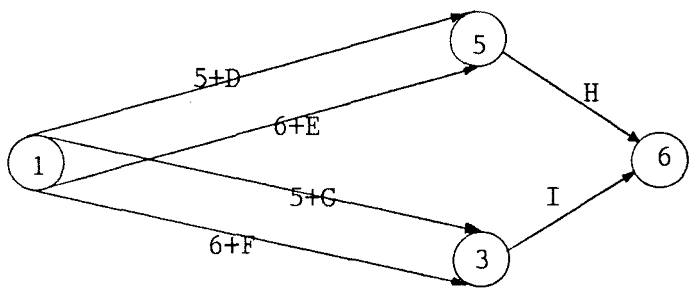


Figure 4.62

Tables 4.32A and 4.32B show duration times of (6+E) and (6+F) respectively.

Table 4.32A: Duration time of (6+E).

	P	CP
10	0.5	0.5
13	0.5	1.0

Table 4.32B: Duration time of (6+F).

	P	CP
9	0.1	0.1
10	0.9	1.0

Taking the maximum of (5+D) and (6+E) gives W and taking the maximum of (5+G) and (6+F) gives X as shown in Figure 4.63.

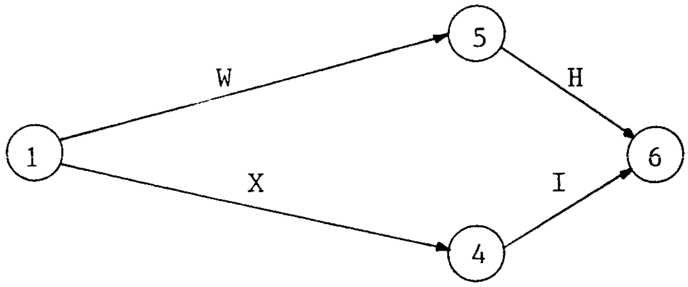


Figure 4.63

Tables 4.33A and 4.33B show duration times of W and X respectively.

Table 4.33A: Duration time of W.

	P	CP
10	1 x 0.5 = 0.5	0.5
13	1 x 1 - 1 x 0.5 = 0.5	1.0

Table 4.33B: Duration time of X.

	P	CP
9	0.1 x 0.7 = 0.07	0.07
10	1 x 0.7 - 0.1 x 0.7 = 0.63	0.70
12	1 x 1 - 1 x 0.7 = 0.30	1.0

Convoluting W and H gives Y and also convoluting X and I gives Z as shown in Figure 4.64.

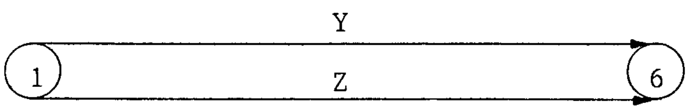


Figure 4.64

Duration time of Y is shown in Table 4.34A and duration time of Z is shown in Table 4.34B.

Table 4.34A: Duration time of Y.

	P	CP
13	0.6 x 0.5 = 0.3	0.3
15	0.4 x 0.5 = 0.2	0.5
16	0.6 x 0.5 = 0.2	0.8
18	0.4 x 0.5 = 0.2	1.0

Table 4.34B: Duration time of Z.

	P		CP
13	0.3 x 0.07	= 0.021	0.021
14	0.3 x 0.63	= 0.189	0.210
15	0.7 x 0.07	= 0.049	0.259
16	0.7 x 0.63 + 0.3 x 0.3	= 0.531	0.790
18	0.7 x 0.3	= 0.210	1.0

Taking the maximum of Y and Z gives α as shown in Figure 4.65.

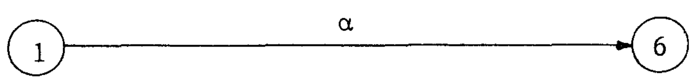


Figure 4.65

Table 4.35 shows the project finish time given A=5 and V=6, where P(A=5)=.8 and P(V=6)=.6 .

Table 4.35: project finish time | (A=5 and V=6).

13	0.3 x 0.021		= 0.0063
14	0.210 x 0.3	- 0.021 x 0.3	= 0.0567
15	0.259 x 0.5	- 0.210 x 0.3	= 0.0665
16	0.790 x 0.8	- 0.259 x 0.5	= 0.5025
18	1 x 1	- 0.790 x 0.8	= 0.3680

By fixing on the second realization time of V, 7, changes network of Figure 4.61 to that of figure 4.66.

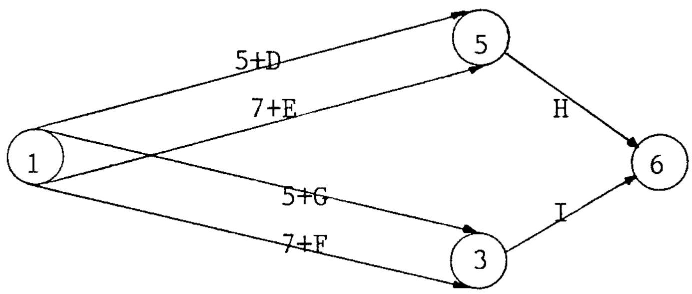


Figure 4.66

Tables 4.36Aand 4.36B show duration times of (7+E) and (7+F) respectively.

Table 4.36A: Duration time of (7+E).

	P	CP
11	0.5	0.5
14	0.5	1.0

Table 4.36B: Duration time of (7+F).

	P	CP
10	0.1	0.1
11	0.9	1.0

Taking the maximum of (5+D) and (7+E) gives β and taking the maximum of (5+G) and (7+F) gives γ as shown in Figure 4.67.

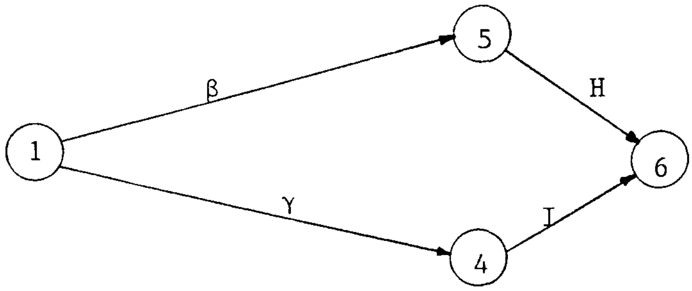


Figure 4.67

Tables 4.37A and 4.37B show duration times of β and γ respectively.

Table 4.37A: Duration time of β.

	P	CP
11	1 x 0.5 = 0.5	0.5
14	1 x 1 - 1 x 0.5 = 0.5	1.0

Table 4.37B: Duration time of γ.

	P	CP
10	0.1 x 0.7 = 0.07	0.07
11	1 x 0.7 - 0.1 x 0.7 = 0.63	0.70
12	1 x 1 - 1 x 0.7 = 0.30	1.0

Convoluting β and H gives δ, and also convoluting γ and I gives θ as shown in Figure 4.68.

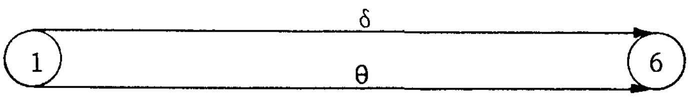


Figure 4.68

Duration time of δ is shown in Table 4.38A and duration time of θ is shown in Table 4.38B.

Table 4.38A: Duration time of δ.

	P	CP
14	0.6 x 0.5 = 0.3	0.3
16	0.4 x 0.5 = 0.2	0.5
17	0.6 x 0.5 = 0.3	0.8
19	0.4 x 0.5 = 0.2	1.0

Table 4.38B: Duration time of θ.

	P		CP
14	0.3 x 0.07	= 0.021	0.021
15	0.3 x 0.63	= 0.189	0.210
16	0.7 x 0.07 + 0.3 x 0.3	= 0.139	0.349
17	0.7 x 0.63	= 0.441	0.790
18	0.7 x 0.3	= 0.210	1.0

Taking the maximum of δ and θ gives λ as shown in Figure 4.69 .

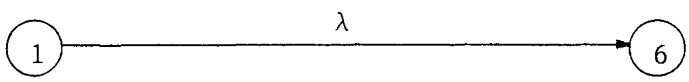


Figure 4.69

Table 4.39 shows the project finish time given A=5 and V=7, where P(A=5)=.8 and P(V=7)=.4.

Table 4.39: Project finish time | (A=5 and V=7).

14	0.021 x 0.3	= 0.0063
15	0.210 x 0.3 - 0.021 x 0.3	= 0.0567
16	0.349 x 0.5 - 0.210 x 0.3	= 0.1115
17	0.790 x 0.8 - 0.349 x 0.5	= 0.4575
18	1 x 0.8 - 0.790 x 0.8	= 0.1680
19	1 x 1 - 1 x 0.8	= 0.2000

By deconditioning the pdfs of project finish times given in Tables 4.25, 4.29, 4.35 and 4.39 the unconditional pdf of project finish time is obtained. Table 4.40 shows the unconditional pdf of Table 4.25.

Table 4.40:

	P(A=2)P(K=3)=(0.2)(0.24)=0.048
10	0.0063 x 0.048 = 0.0003024
11	0.0567 x 0.048 = 0.0027216
12	0.0665 x 0.048 = 0.0031920
13	0.5025 x 0.048 = 0.0241200
15	0.3680 x 0.048 = 0.0176640

Table 4.41 shows the unconditional pdf of Table 4.29.

Table 4.41:

	P(A=2)P(K=4)=(0.2)(0.76)=0.152
11	0.0063 x 0.152 = 0.0009576
12	0.0567 x 0.152 = 0.0086184
13	0.1115 x 0.152 = 0.0169480
14	0.4575 x 0.152 = 0.0695400
15	0.1680 x 0.152 = 0.0255360
16	0.2000 x 0.152 = 0.0304000

Table 4.42 shows the unconditional pdf of Table 4.35.

Table 4.42:

	P(A=5)P(V=6)=(0.8)(0.6)=0.48
13	0.0063 x 0.48 = 0.003024
14	0.0567 x 0.48 = 0.027216
15	0.0665 x 0.48 = 0.031920
16	0.5025 x 0.48 = 0.241200
18	0.3680 x 0.48 = 0.176640

Table 4.43 shows the unconditional pdf of Table 4.39.

Table 4.43

P(A=5)P(V=7)=(0.8)(0.4)=0.32

14	0.0063 x 0.32 = 0.002016
15	0.0567 x 0.32 = 0.018144
16	0.1115 x 0.32 = 0.035680
17	0.4575 x 0.32 = 0.146400
18	0.1680 x 0.32 = 0.053760
19	0.2000 x 0.32 = 0.064000

Simple addition of probabilities for each realization time of Table 4.40 to Table 4.43 gives the unconditional pdf of project finish time as shown in Table 4.44.

Table 4.44: Project finish time.

	P	
10	0.0003024	= 0.0003024
11	0.0027216 + 0.00095760	= 0.0036792
12	0.0031920 + 0.00861840	= 0.0118104
13	0.0241200 + 0.01694800 + 0.0030240	= 0.0440920
14	0.0695400 + 0.02721600 + 0.0020160	= 0.0987720
15	0.0176640 + 0.0255360 + 0.0319200 + 0.0181440	= 0.0932640
16	0.0304000 + 0.2412000 + 0.3568000	= 0.3072800
17	0.1464000	= 0.1464000
18	0.1766400 + 0.0537600	= 0.2304000
19	0.0640000	= 0.0640000
E =		16.3086640

SUMMARY AND CONCLUSIONS

In this chapter a procedure to determine the probability distribution function of project completion time for PERT network with discrete statistically independent distribution is presented. This procedure computationally is based on the CIM approach and possesses the following advantages.

1 - Provides an exact pdf for project completion time in PERT networks with discrete distributions.

2 - Determination of the criticality indices of activities is a simple matter for an activity network in which this procedure is employed in evaluation of its completion time.

3 - It can be applied for PERT networks with statistical and structural dependence relationships between activities.

4 - It can be applied for PERT networks with discrete or continuous distributions .

The algorithm consists of the following steps.

1 - Reduce the network to its irreducible form using convolution and greatest operations.

2 - If the network is reduced to an equivalent activity starting in node 1 and ending in node N, stop. The pdf of the duration time of this final activity is equal to F(t).

3 - If the network is not completely reducible, calculate the indegree and outdegree of every node $i \neq N$, i.e. I(i) and O(i), then choose one activity 'a' such that 'a' has more than one successor while each of its successor has only 'a' as a predecessor.

4 - Conditionalize by setting the chosen activity 'a' at its kth realization time T_a^k; this is done by deleting 'a', adding T_a^k to the successors of 'a', and maintaining the implied precedence of activities in the conditinalized network.

5 - Decondition the df of final activity of step 4.

6 - Determine the df of project completion time, mean and standard deviation.

CHAPTER 5: OTHER METHODS RELEVANT TO PROPOSED PROCEDURE

INTRODUCTION

This chapter looks briefly at other methods relevant to the proposed procedure of Chapter 4. Important factors affecting the magnitude of the bias using other methods are then considered through different examples.

These factors are as follows:

1 - Number of subcritical paths leading to a merge event.

2 - "Closeness" of the expected completion times of the subpaths leading to a merge event.

3 - The variance of the subpaths lengths.

4 - The correlation of the subpaths, i.e., the number of common activities between subpaths.

Fulkerson's Approach

As pointed out in Chapter 2 the first improvement for the PERT estimate is due to Fulkerson (1962). He proposed a lower bound that is a function of the variance associated with each arc for the case where the activity durations are discrete random variables.

Fulkerson's estimate is as follows:

Let B_j denote the set of arcs immediately preceding node j. Determine a function f_j recursively by

$$f_1 \equiv 0$$

$$f_j = \sum_{y(B_j)} \max \{f_{i_1} + y_1; f_{i_2} + y_2; \ldots; f_{i_r} + y_{r_j}\} \cdot p[y(B_j)]; \qquad (2.15)$$

$$j = 2, 3, \ldots, n$$

where $y(B_j)$ is the vector of realization of all arcs in the set B_j.

Recall that the PERT model estimates the expected duration of the project by defining the function g_j recursively as follows:

$g_1 \equiv 0$,

$$g_j = \max_i \{g_i + \bar{Y}_{ij}\} \qquad , j=2,3,\ldots,N . \qquad (2.14)$$

Note that, j is succeeding event number, and i<j.

where $$\bar{Y}_{ij} = \sum_{y_{ij}} y_{ij} \cdot p(y_{ij})$$

and p(ij) is the marginal probability function of arc (ij).

Example 1:

Consider the network of Figure 5.1. The realizations of each activity indicated on the arcs are assumed equally likely. This simple network has 324 different realizations, all equally probable. All 324 realizations were enumerated by Elmaghraby (1977), the CP in each realization evaluated, and then the average evaluated, $e_4=12.3148$. The PERT estimate proceeds as follows:

$\bar{y}_{1,2}=8/3; \bar{y}_{1,3}=5; \bar{y}_{1,4}=11/3; \bar{y}_{2,3}=4; \bar{y}_{2,4}=19/3$, and $\bar{y}_{3,4}=10/3$; and

$$g_1 \equiv 0$$

$$g_2 = \bar{y}_{1,2} = \frac{8}{3}$$

$$g_3 = \max \{0+5; \frac{8}{3} + 4\} = \frac{20}{3}$$

$$g_4 = \max \{0+\frac{11}{3};\ \frac{8}{3} + \frac{19}{3};\ \frac{20}{3} + \frac{10}{3}\} = 10 \cdot$$

To be sure, $g_4 = 10 < 12.3148 = e_4$. We now evaluate the function (f_i).

$$f_1 \equiv 0$$

$$f_2 = \bar{y}_{1,2} = \frac{8}{3}$$

$$f_3 = \frac{1}{4}(5\tfrac{2}{3} + 7\tfrac{2}{3} + 2\text{x}8) = \frac{22}{3}$$

$$f_4 = \frac{1}{27}(2\text{x}8\ \tfrac{1}{3} + 2\text{x}8\ \tfrac{2}{3} + 2\text{x}10 + 6\text{x}11\tfrac{1}{3} + 6\text{x}12\tfrac{1}{3} + 9\text{x}14\tfrac{2}{3})$$

$$= \frac{328}{27} = 12.148 \cdot$$

Indeed, $g_4=10 < f_4=12.148 < e_4=12.3148$. Formally, we have

<u>Theorem 5.1</u>: $g_j \leqslant f_j \leqslant e_j \ \forall j=1,2,3,...,n$. Proof is given in Elmaghraby (1977).

Fulkerson's (1962) approach is not closely related to the proposed approach for determining the pdf of project completion times, but its procedure is similar to the procedure of the proposed method for approximating criticality indices of the activities, the subject matter of Chapter 6.

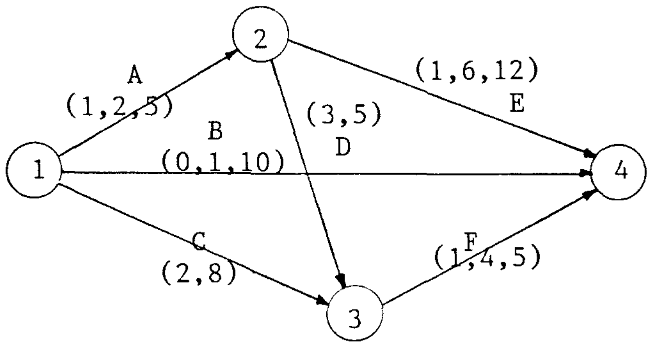


Figure 5.1. Example network with random durations.

Table 5.1 summarizes the data on the CPs, their length, and the frequency of the occurrence, where

Path 1-2-4 is labelled #1

1-2-3-4 is labelled #2

1-4 is labelled #3

1-3-4 is labelled #4

Table 5.1. Analysis of CPs of Network of Figure 5.1

Length of CP	Paths in Network								Frequency of Length	Relative Frequency
	1	2	3	4	1or2	1or4	2or3	2or4		
17	36								36	0.1111
15		12							12	0.0370
14	36	12							48	0.1481
13	30	6		24		6		6	72	0.2222
12		12		24				6	42	0.1296
11	6	18			6				30	0.0926
10		8	24				4		36	0.1111
9		10		16				2	28	0.0864
8	2	6			2				10	0.0309
7	2	2			2				6	0.0185
6		2							2	0.0062
5		2							2	0.0062
Frequency of Path	112	90	24	64	10	6	4	14	324	

THE APPROXIMATING PROCEDURE

The approximating procedure proposed by Dodin (1985a) consists of three steps: discretizing continuous distributions, reducing the network, and sequential approximation. A brief discussion of these three steps; is given in Dodin (1985a) and a detailed discussion and a documentation of their computer program are presented in Dodin (1980). Figure 5.2 summarizes the steps of the procedure.

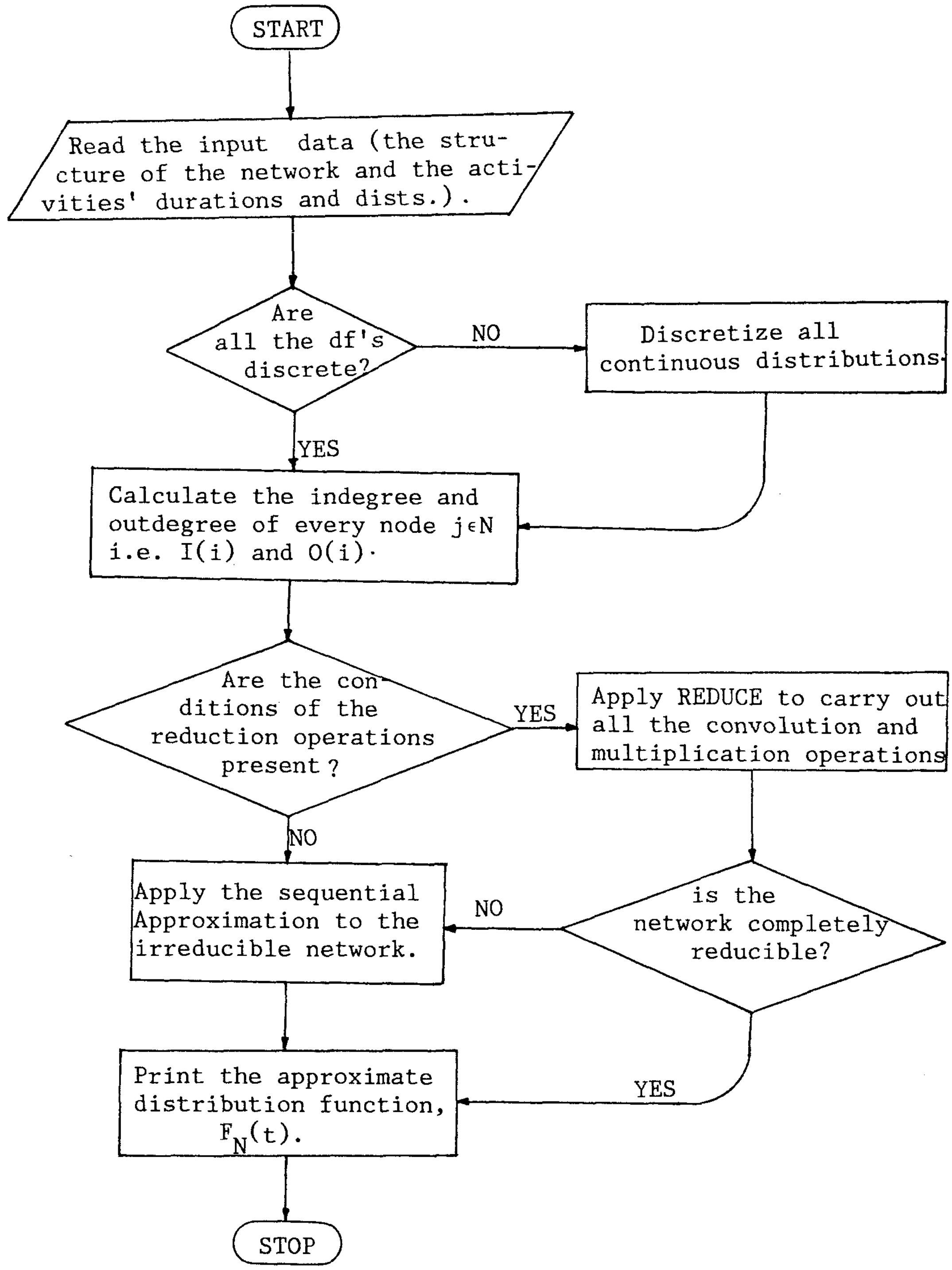


Figure 5.2: A procedure to approximate the distribution function of the longest path in stochastic networks.

1 - Discretizing the continuous distributions

The discretization is done by approximating the continuous df by a dicrete function represented by the set of ordered pairs $F_{ij}(y)= \{y_k, p(y_k))\}$. The accuracy of the discretization can be increased by increasing the cardinality of the set $F_{ij}(y)$.

2 - Reducing the network

If the conditions of either Convolution or Greatest operations exist in the stochastic network, then the network can be reduced by the repeated use of the two operations until one of the following two evaluation occurs:

(i) The network is reduced to an equivalent activity starting in node 1 and ending in node N. The df of the duration of this final activity is equal to F(t). In this case the network is termed completely reducible.

(ii) The network reach a form G(N',A') where $N' \leqslant N$ and $A' \leqslant A$ which does not have the conditions of either of the above two operations. The original network, G(N,A), is termed reducible, and the new network G(N',A') is termed irreducible.

The logic used to effect all possible reductions in the network is stated in the following by the name REDUCE.

The reduction process (REDUCE). The following steps are used to identify and perform all the convolution and multiplication operations in the network:

(1) Calculate I(i) and O(i), the indegree and outdegree, for all nodes j=1,2,..,N in the network G(N,A).

(2) If $I(i)+O(i) \geqslant 3$ for all $j \neq 1$, N then the network is irreducible; stop. If on the other hand $I(i)+O(i)=2$ for at least one $j \neq 1$, N then the Activity Network is reducible since a convolution operation is possible: if it is carried out it might give rise to multiplication and convolution operations. Go to 3.

(3) Use the scanning operator, denoted by SCAN, described below to identify all the convolution and multiplication operations, i.e.

SCAN: $G(N,A) \rightarrow G(N',A')$

where $N' < N$ and $A' < A$. The following are the steps of SCAN:

(a) Rank the arcs in A from 1 to A such that for any two arcs $p=(ij)\epsilon A$ and $q=(lk)\epsilon A$ $p<q$ if $i<l$; if $i=l$ then $j<k$. Set the arc indicator $\delta(p)=1$ for all $p=1,2,\ldots,A, M=A$, and $K=1$.

(b) If $\delta(K)=0$, go to e; otherwise let i be the start node of arc K and j be its end node, and set $J=1$.

(c) If $\delta(J)=0$ or $J=K$, go to d; otherwise let α be the start node of arc J and β be its end node, then check for the conditions of either a multiplication or a convolution operation; if $i=\alpha$ and $j=\beta$ then K and J define a multiplication operation; do the necessary bookkeeping, and set $\delta(K)=0$, $\delta(J)=0$, $M\rightarrow M+1$ and $\delta(M)=1$, then go to e, If not, then if either (i) $j=\alpha$ and $I(i)=0(i)=1$, or (ii) $i=\beta$ and $I(i)=0(i)=1$, then K and J define a convolution operation; do the necessary bookkeeping and set $\delta(K)=0$, $\delta(J)=0$, $M \rightarrow M+1$ and $\delta(M)=1$, then go to e.

(d) Set $J \rightarrow J+1$. If $J > M$, go to e; otherwise go to c.

(e) If $K=M$, stop; otherwise set $K \rightarrow K+1$ and go to b.

If $A'=1$ then the network is completely reducible and the approximating procedure terminates with the df of T_N approximated by the df of the equivalent activity (1,N). If $A'\neq1$, then $A'\geqslant5, N'\geqslant4$ and $G(N',A')$ is the corresponding irreducible network of $G(N,A)$. In this case the approximate df, of T_N is obtained through the use of the Sequential Approximation presented in the next section.

The reduction process described above starts with a convolution operation, then a sequence of multiplication and convolution operations may follow in any order. The process may start with any of the initial convolutions available without fear of missing any convolution or multiplication operation ; this is clear from Step 3 above. The reduction process does not alter the realization of T_N or its df.

Completely reducible networks can be reduced to the equivalent, single activity (1,N) in a fixed number of convolution and multiplication operations. To reduce the network to the activity (1,N) implies that N-2 nodes and A-1 arcs have to be suppressed, but a convolution operation is necessary to reduce the network by one arc and one node. Hence we have N-2 convolution operations and A-N+1 multiplication operations. Therefore the complexity of REDUCE is (at worst) of O(CA), exactly like the complexity of the sequential approximation, where C is the complexity of the convolution and multiplication operations.

3 - Sequential Approximation

The sequential approximation to be described below can be applied to any stochastic network with $N>2$ and $A>1$, reducible or irreducible. The approximation process enters the sequential approximation with the irreducible network G(N',A'). The sequential approximation starts at node 1, which has the df F(1) = {(0,1)}, then proceeds sequentially to approximate the df of the realization times of the next nodes in increasing order, ending with node N. The df of T_j for all $j \in N'$ is approximated using the following procedure:

(1) Without loss of generality, assume the sequential approximation is at node $j \in N'$, then $I(j)>1$ or $O(j)>1$, i.e. $I(j)+O(j) \geq 3$ as is shown in Figure 5.3.

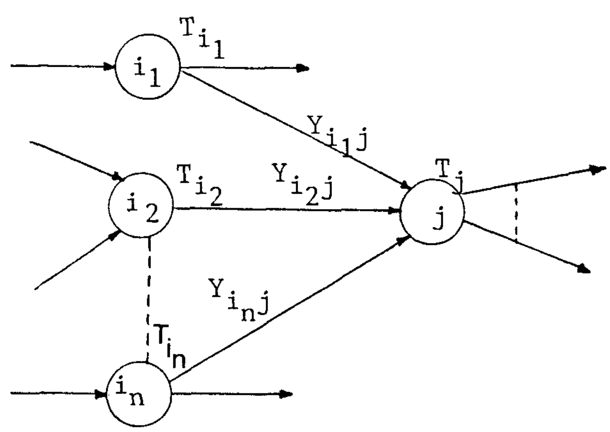


Figure 5.3: A node in an irreducible network.

Determine the set of B(j), which is the set of activities ending at node j, and rank the activities in an ascending order of their starting nodes; see Figure 5.3 for illustration.

(2) For each activity $(ij) \in B(j)$:

(a) Convolute $F_{ij}(y)$ with $F_i(t)$, i.e. the distribution function of Y_{ij} with the distribution function of T_i. Denote this convolution by $\bar{F}(i)$.

Therefore, for any value $t > 0$,

$$\bar{F}(i) = Pr(T_i+Y_{ij} \leqslant t)=\sum_y Pr(Y_{ij}=y)Pr(T_i \leqslant t-y) .$$

(b) Let $K=C(\bar{F}(i))$ which is the number of ordered pairs in the distribution $\bar{F}(i)=\{r_m, p(r_m))\}$. K can be a large number; for example if $C(F_{ij}(y))=20$ and $C(F_i(t))=30$, then $30 \leqslant C(\bar{F}(i)) \leqslant 600$. Thus, if K is greater than a desired number of discrete points (realizations) the analyst would like to have for the df of T_j, then, $\bar{F}(i)$, which has K ordered pairs, is approximated by another df with the desired number of realizations; denote such a number by k. This approximation is done according to the rules:

(i) The full range of the distribution $\bar{F}(i)$ is maintained.

(ii) The intermediate K-2 points are mapped into k-2 points using the following three steps;

(ii.1) Let $\Delta=(r_{K-1}-r_2)/(k-2)$, then we have k-2 intervals; each is of width Δ. The first interval contains all the realizations in the interval $[r_2,r_2+\Delta]$, and the nth interval contains all the realizations $r_m \in [r_2+\Delta(n-1),r_2+n\Delta]$ for $n=2,3,\ldots,k-2$.

(ii.2) For the realizations in the nth interval let

$$x_n = \sum_m r_m\, p(r_m) \text{ and } y_n = \sum_m p(r_m),$$

then

$(r'_n \cdot p(r'_n))=(x_n/y_n,y_n)$,

(ii,3) If the nth interval is empty, i.e. there does not exist any

$r_m \in [r_2+\Delta(n-1), r_2 + n\Delta]$, then

$(r'_n, p(r'_n))=(r_2+(n-0.5),0)$.

(3) $F_j(t)=Pr(T_j \leqslant t)$ max$=\{\bar{F}(i)$ for all $i \in N'$ such that $(ij) \in B(j)\}$;
To avoid any unexpected escalation in the storage requirements, the above maximum operation can be performed sequentially, and the operation of step 2.b above can be used whenever the cardinality of the new distribution is greater than k. For instance, let

$$F(i_1,i_2)=\max\{\bar{F}(i_1), \bar{F}(i_2)\},$$

i.e. for any value $t > 0$

$$F(i_1,i_2)=Pr(\max\{T_{i1}+Y_{i1j}, T_{i2}+Y_{i2j}\} \leqslant t)$$

$$=Pr(T_{i1}+Y_{i1j} \leqslant t)Pr(T_{i2} + Y_{i2j} \leqslant t)$$

then,

$$F(i_2,i_3)=\max\{F(i_1,i_2), \bar{F}(i_3)\},$$

and so on until $F_j(t)$ is finally obtained where

$$F_j(t)=F(i_{n-1}, i_n)=\max\{F(i_{n-2},i_{n-1}),\bar{F}(n)\}.$$

$k > j \in N'$, until finally, node N is reached, and $F_N(t)$ is approximated. The complexity of the sequential approximation is a linear function of the number of the convolution and multiplication operations. In an irreducible network with A arcs and N nodes, Step 2 of the SA implies that we have A-0(1) convolution operations. Step 3 implies that for each $j \neq 1$ or 2 we have $I(j)-1$ multiplication operations, which implies that the total number of the multiplication operations is

$$\sum_{j=2}^{N} (I(j)-1)=A-N+1 \quad .$$

Therefore, the complexity of the SA is of O(CA) where C is the complexity of the convolution and multiplication operations, which depends on the cardinality of the discrete distribution functions. If the cardinality of the discrete distributions is less than or equal to k, then the complexity of the convolution and multiplication is (at worst) of $O(k^2)$.

Example 1.

Consider, again, the network of Figure 5.1. We apply the approximating procedure and evaluate d_4 as an estimate of e_4.

Tables 5.2 to 5.9 show the calculation procedure.

Table 5.2: A's finish time.

	P
1	1/3
2	1/3
3	1/3

Table 5.3: B's finish time.

	P	CP
0	1/3	1/3
1	1/3	2/3
10	1/3	3/3

Table 5.4: C's finish time.

	P	CP
2	1/2	1/2
8	1/2	1/1

Table 5.5: D's finish time.

	P	CP
4	1/6	1/6
5	1/6	2/6
6	1/6	3/6
7	1/6	4/6
8	1/6	5/6
10	1/6	6/6

Table 5.6: E's finish time.

	P	CP
2	1/9	1/9
3	1/9	2/9
6	1/9	3/9
7	1/9	4/9
8	1/9	5/9
11	1/9	6/9
13	1/9	7/9
14	1/9	8/9
17	1/9	9/9

Table 5.7: F's start time.

	P
4	1/12
5	1/12
6	1/12
7	1/12
8	6/12
10	2/12

Table 5.8: F's finish time.

	P	CP
5	1/36	1/36
6	1/36	2/36
7	1/36	3/36
8	2/36	5/36
9	8/36	13/36
10	2/36	15/36
11	4/36	19/36
12	7/36	26/36
13	6/36	32/36
14	2/36	34/36
15	2/36	36/36

Table 5.9: Project finish time using Dodin's approach.

	P	
5	4/972	
6	8/972	
7	12/972	
8	26/972	
9	80/972	
10	95/972	
11	117/972	
12	126/972	
13	204/972	
14	144/972	
15	48/972	
17	108/972	
d_4 =	12.400205	σ=2.5192

Using the proposed procedure of Chapter 4, df of the project completion time can be computed as shown in Table 5.10.

Table 5.10: Project completion time using proposed procedure.

	P	
5	0.0062	
6	0.0062	
7	0.0185	
8	0.0308	
9	0.0864	
10	0.1111	
11	0.0926	
12	0.1296	
13	0.2222	
14	0.1481	
15	0.0370	
17	0.1111	
E_4=	12.3148	σ=2.5735

Notice that the df obtained using the proposed procedure is exactly the same as the df obtained with complete enumeration as shown in Table 5.1.

Example 2

Consider the network of Figure 3.10. In the following, the approximating procedure is applied to approximate the df of the project completion time.

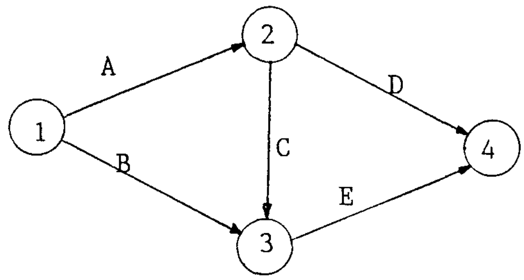


Figure 3.10

Tables 5.11 through 5.18 show the calculation procedure.

Table 5.11: Duration time of A=D's start time.

X_A	P	CP
3	0.8	0.8
8	0.2	1.0

Table 5.12: Duration times of B,C,D and E.

X_B	P	CP	X_C	P	CP	X_D	P	CP	X_E	P	CP
6	0.6	0.6	4	0.3	0.3	4	0.9	0.9	1	0.5	0.5
9	0.4	1.0	6	0.7	1.0	5	0.1	1.0	2	0.5	1.0

Table 5.13: C's finish time.

	P	CP
7	0.8 x 0.3 = 0.24	0.24
9	0.8 x 0.7 = 0.56	0.80
12	0.2 x 0.3 = 0.06	0.86
14	0.2 x 0.7 = 0.14	1.0

Table 5.14: D's finish time.

	P	CP
7	0.8 x 0.9 = 0.72	0.72
8	0.8 x 0.1 = 0.08	0.80
12	0.2 x 0.9 = 0.18	0.98
13	0.2 x 0.1 = 0.02	1.0

Table 5.15: E's start time.

	P			CP
7	0.24 x 0.60		= 0.144	0.144
9	0.80 x 1	- 0.24 x 0.60	= 0.656	0.800
12	0.86 x 1	- 0.80 x 1	= 0.060	0.860
14	1 x 1	- 0.86 x 1	= 0.140	1.0
E =			9.592	

Table 5.16: E's finish time.

	P	CP
8	0.144 x 0.5 = 0.072	0.072
9	0.144 x 0.5 = 0.072	0.144
10	0.656 x 0.5 = 0.328	0.472
11	0.656 x 0.5 = 0.328	0.800
13	0.060 x 0.5 = 0.030	0.830
14	0.060 x 0.5 = 0.030	0.860
15	0.140 x 0.5 = 0.070	0.930
16	0.140 x 0.5 = 0.070	1.0
E =	11.092	

Table 5.17: Project finish time using Dodin's approach.

	P	
8	0.80 x 0.072	= 0.0576
9	0.80 x 0.144 - 0.80 x 0.072	= 0.0576
10	0.80 x 0.472 - 0.80 x 0.144	= 0.2624
11	0.80 x 0.80 - 0.80 x 0.472	= 0.2624
12	0.98 x 0.80 - 0.80 x 0.80	= 0.1440
13	0.83 x 1 - 0.98 x 0.80	= 0.0460
14	0.86 x 1 - 0.83 x 1	= 0.0300
15	0.93 x 1 - 0.86 x 1	= 0.0700
16	1 x 1 - 0.93 x 1	= 0.0700
d_4 =		11.4056 σ=2.077

The project finish time using proposed procedure is given in Table 4.17.

Table 4.17: Unconditional project completion time.

	P	
FP = 8	0.072	
9	0.072	
10	0.328	
11	0.328	
13	0.03	
14	0.03	
15	0.07	
16	0.07	
E =	11.092	σ=2.125

The df of the project completion time can be computed from tree diagram of Figure 6.11. Table 5.18 shows the exact df of the project completion time.

Table 5.18: Exact project completion time.

	P	
8	0.072	
9	0.072	
10	0.328	
11	0.328	
13	0.03	
14	0.03	
15	0.07	
16	0.07	
E=	11.092	σ=2.125

Notice that the df obtained using the proposed procedure is exactly same as the df obtained using complete enumeration.

Example 3.

Consider the network configuration of Figure 4.49. Table 5.19 shows the duration times of activities, and Tables 5.20 through 5.22 show the df of the project completion times using Dodin's approach, complete enumeration, and proposed procedure of Chapter 4 respectively.

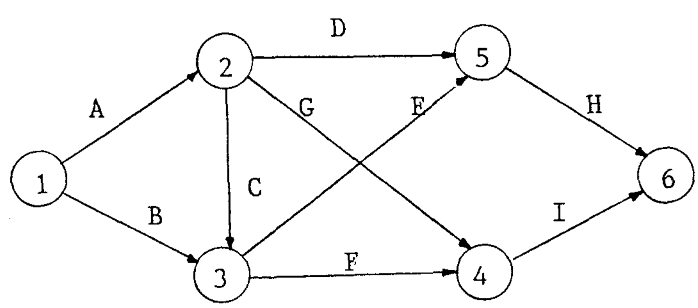


Figure 4.49

Table 5.19: Duration times of activities

X_A	P	CP	X_B	P	CP	X_C	P	CP
2	0.2	0.2	3	0.4	0.4	1	0.6	0.6
5	0.8	1.0	4	0.6	1.0	2	0.4	1.0
X_D	P	CP	X_E	P	CP	X_F	P	CP
3	0.9	0.9	4	0.5	0.5	3	0.1	0.1
5	0.1	1.0	7	0.5	1.0	4	0.9	1.0
X_G	P	CP	X_H	P	CP	X_I	P	CP
3	0.7	0.7	3	0.6	0.6	4	0.3	0.3
7	0.3	1.0	5	0.4	1.0	6	0.7	1.0

Table 5.20: Project completion time using Dodin's approach.

	P	
10	0.0000005	
11	0.000146	
12	0.0022794	
13	0.0135847	
14	0.0872907	
15	0.0786823	
16	0.3230316	
17	0.1938648	
18	0.23712	
19	0.064	
d_6 =	16.556236	σ=1.4139275

Table 5.21: Project completion time using complete enumeration.

	P	
10	0.0003023	
11	0.0036782	
12	0.0118095	
13	0.0441925	
14	0.0987603	
15	0.0931824	
16	0.3069175	
17	0.146392	
18	0.2303077	
19	0.063992	
e_6=	16.300809	σ=1.6511466

Table 5.22: Project completion time using proposed procedure.

	P	
10	0.0003024	
11	0.0036792	
12	0.0118104	
13	0.044092	
14	0.098772	
15	0.093264	
16	0.30728	
17	0.1464	
18	0.2304	
19	0.064	
E_6=	16.308664	σ=1.6509811

Notice that the deviation between corresponding enteries of Tables 5.21 and 5.22 is due to approximation in rounding operation.

Example 4.

Consider the network configuration of Figure 5.4. This network is symmetric, duration times of activities are shown beside each activity with equal probability for occurrence. Tables 5.23 through 5.25 show the df of the project completion times using Dodin's approach, complete enumeration, and proposed procedure respectively.

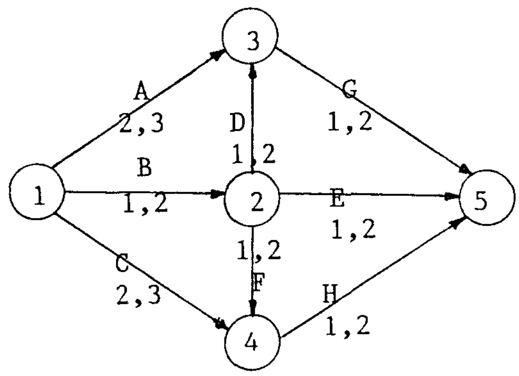


Figure 5.4

Table 5.23: Project finish time using Dodin's approach.

	P	
3	0.0029296	
4	0.1884766	
5	0.5742188	
6	0.234375	
d_5 =	5.0400392	σ=0.6580019

Table 5.24: Project completion time using complete enumeration.

	P	
3	0.0078125	
4	0.21875	
5	0.5546875	
6	0.21875	
e_5=	4.984375	σ=0.6844746

Table 5.25: Project completion time using proposed procedure.

	P	
3	0.0078125	
4	0.21875	
5	0.5546875	
6	0.21875	
E_5=	4.984375	σ=0.6844746

Now let us change realization times of common activity B from (1 and 2) to (1 and 10) as shown in Figure 5.5, and apply three above mentioned approaches to this network.

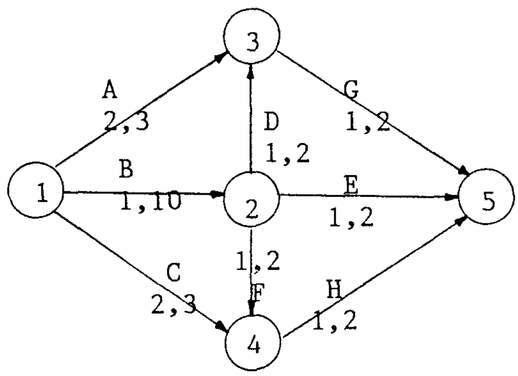


Figure 5.5

Table 5.26: Project completion time using Dodin's approach.

	P	
3	0.0019531	
4	0.046875	
5	0.0761719	
11	0.0625	
12	0.203125	
13	0.375	
14	0.234375	
d_5=	11.855468	σ=2.8713645

Table 5.27: Project completion time using complete enumeration or using proposed procedure.

	P	
3	0.0078125	
4	0.1875	
5	0.3046875	
12	0.03125	
13	0.25	
14	0.21875	
$e_5=E_5=$	8.984375	σ=4.4264551

Notice that in all examples of this chapter the mean value of project completion time using Dodin's approach is pessimistically biased, i.e., Dodin's approach overestimates the mean value of project completion time. On the other hand, Dodin's approach underestimates the standard deviation of project completion time. The bias in Dodin's approach mainly depends on the following two factors.

1 - Number of activities emanating from a merge event.

2 - Standard deviation of the subpaths lenghts.

For instance in Figure 5.4 three arcs emanate from node 2, since in Dodin's approach it is assumed that the paths are independent, by using sequential approximation we take into account common activity B of Figure 5.4 three times as shown in Figure 5.6, therefore, we increase the number of paths in the network, as a result we increase the mean value of project completion time, and decrease the standard deviation of the project completion time.

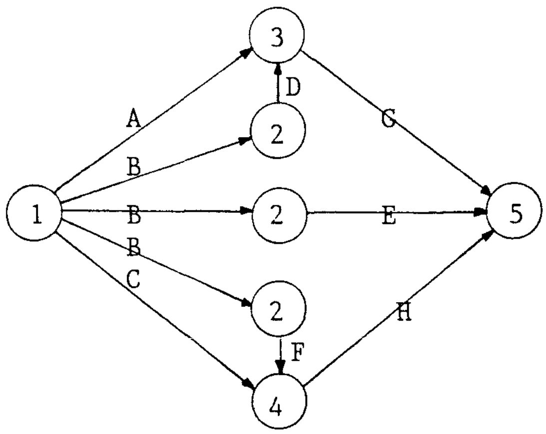


Figure 5.6

Second factor which affects the mean value and standard deviation of project completion time using Dodin's approach is the standard deviation of the activities leading into those merge events for which the outdegree are greater than one. Table 5.27 shows the effect of this factor on the mean value and standard deviation of the project completion times of the networks of Figure 5.5 and Figure 5.6, where in Figure 5.5 the common activity B has two realization times 1 and 2 (σ_B=0.5) and in Figure 5.6 it has two realization tiems 1 and 10 (σ_B=4.5).

Table 5.27:

σ_B=0.5	
Exact-Calculated mean	= 4.984375
Dodin's-Calculated mean	= 5.0400392
per cent error (Dodin's from Exact)	+ 1.12
Exact-Calculated standard deviation	= 0.6844746
Dodin's-Calculated standard deviation	= 0.6580019
per cent error (Dodin's from Exact)	- 3.9

$\sigma_B = 4.5$

Exact-Calculated mean	= 8.98437
Dodin's-Calculated mean	=11.855468
per cent error (Dodin's from Exact)	+ 32
Exact-Calculated standard deviation	= 4.4264551
Dodin's-Calculated standard deviation	= 2.8713645
per cent error (Dodin's from Exact)	- 35

Notice that the per cent error in mean value and standard deviation of the project completion time is an increasing function of the standard deviation of common activity B.

Example 5.

Figure 5.7 shows an irreducible network with four common arcs A,B,G and H. Assume all activities have two realization times 1 and 2 and equal probability.

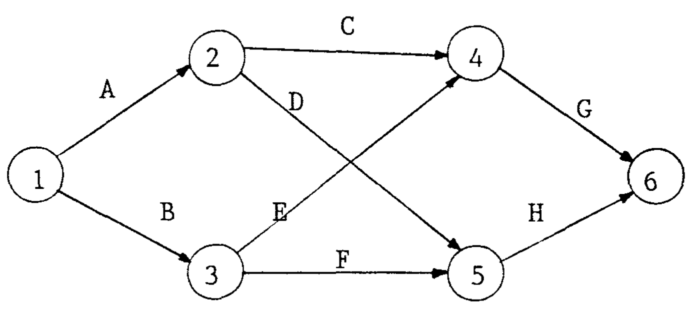


Figure 5.7

In order to evaluate effects of standard deviations of common activities A and B on the mean value and standard deviation of project completion time we gradually increase the duration times of activities A and B as follows:

	A,B	A,B	A,B	A,B	A,B
	X | p	X | p	X | p	X | p	X | p
	1 | 0.5	2 | 0.5	3 | 0.5	4 | 0.5	5 | 0.5
	2 | 0.5	4 | 0.5	6 | 0.5	8 | 0.5	10 | 0.5
E=	1.5	3	4.5	6	7.5
σ=	0.5	1	1.5	2	2.5

	A,B	A,B	A,B	A,B	A,B
	X | p	X | p	X | p	X | p	X | p
	6 | 0.5	7 | 0.5	8 | 0.5	9 | 0.5	10 | 0.5
	12 | 0.5	14 | 0.5	16 | 0.5	18 | 0.5	20 | 0.5
E=	9	10.5	12	13.5	15
σ=	3	3.5	4	4.5	5

The following tables show the project completion time using Dodin's approach. In each case only the final table is shown.

Project completion time.

	P
3	0.0009765
4	0.0966797
5	0.5126953
6	0.3896485
d_6 =	5.291015
σ =	0.6368232

A,B	
1	0.5
2	0.5

Project completion time.

	P
4	0.0009765
5	0.0234375
6	0.140625
7	0.4453125
8	0.3896485

A,B	
2	0.5
4	0.5

$d_6 =$ 7.199219

$\sigma =$ 0.770145

Protection Completion time.

	P
5	0.0009765
6	0.0234375
7	0.038086
8	0.102539
9	0.4453125
10	0.3896485

A,B	
3	0.5
6	0.5

$d_6 =$ 9.136719

$\sigma =$ 0.9232554

Protect completion time.

	P
6	0.0009765
7	0.0234375
8	0.038086
10	0.102539
11	0.4453125
12	0.3896485

A,B	
4	0.5
8	0.5

d_6 = 11.074219

σ = 1.1085419

Project completion time.

	P
7	0.0009765
8	0.0234375
9	0.038086
12	0.102539
13	0.4453125
14	0.3896485

A,B	
5	0.5
10	0.5

$d_6 =$ 13.011719

$\sigma =$ 1.312447

Project completion time.

	P
8	0.0009765
9	0.0234375
10	0.038086
14	0.102539
15	0.4453125
16	0.3896485

A,B	
6	0.5
12	0.5

$d_6 =$ 14.949219

$\sigma =$ 1.5275329

Project completion time.

	P
9	0.0009765
10	0.0234375
11	0.038086
16	0.102539
17	0.4453125
18	0.3896485

A,B	
7	0.5
14	0.5

$d_6 =$ 16.886719

$\sigma =$ 1.749811

Project completion time.

	P
10	0.0009765
11	0.0234375
12	0.038086
18	0.102539
19	0.4453125
20	0.3896485

A,B	
8	0.5
16	0.5

d_6 = 18.824219

σ = 1.9765117

Project completion time.

	P
12	0.0009765
13	0.0234375
14	0.038086
22	0.102539
23	0.4453125
24	0.3896485

A,B	
10	0.5
20	0.5

d_6 = 22.699218

σ = 2.4389732

Project completion time.

	P
11	0.0009765
12	0.0234375
13	0.038086
20	0.102539
21	0.4453125
22	0.3896485

A,B	
9	0.5
10	0.5

d_6= 20.761719

σ = 2.2065812

The following tables show project completion time using proposed procedure.

In each case only the final table is shown.

Project completion time.

	P
3	0.0039062
4	0.1289062
5	0.4960937
6	0.3710937

A,B	
1	0.5
2	0.5

E_6 = 5.234375
σ = 0.6787435

Project completion time.

	P
4	0.0039062
5	0.09375
6	0.1875
7	0.34375
8	0.3710937

A,B	
2	0.5
4	0.5

E_6 = 6.984375
σ = 0.9841265

Project completion time.

	P
5	0.0039062
6	0.09375
7	0.1523437
8	0.0351562
9	0.34375
10	0.3710937

A,B

3	0.5
6	0.5

E_6 = 8.734375

σ = 1.357417

Project completion time.

	P
6	0.0039062
7	0.09375
8	0.1523437
10	0.0351562
11	0.34375
12	0.3710937

A,B

4	0.5
8	0.5

E_6 = 10.48375

σ = 1.7632723

Project completion time.

	P
7	0.0039062
8	0.09375
9	0.1523437
12	0.0351562
13	0.34375
14	0.3710937

A,B

5	0.5
10	0.5

E_6 = 12.234375

σ = 2.1775998

Project completion time

	P
8	0.0039062
9	0.09375
10	0.1524375
14	0.0351562
15	0.34375
16	0.3710937

A,B

6	0.5
12	0.5

E_6 = 13.984375

σ = 2.5980286

Project completion time.

	P
9	0.0039062
10	0.09375
11	0.15234375
16	0.0351562
17	0.34375
18	0.3710937

A,B

7	0.5
14	0.5

E_6 = 15.7343375

σ = 3.0220133

Project completion time.

	P
10	0.0039062
11	0.09375
12	0.15234375
18	0.0351562
19	0.34375
20	0.3710937

A,B

8	0.5
16	0.5

E_6 = 17.484375

σ = 3.4482423

Project completion time.

	P
11	0.0039062
12	0.09375
13	0.15234375
20	0.0351562
21	0.34375
22	0.3710937

A,B	
9	0.5
18	0.5

$E_6 =$ 19.234375

$\sigma =$ 3.875953

Project completion time.

	P
12	0.0039062
13	0.09375
14	0.15234375
22	0.0351562
23	0.34375
24	0.3710937

A,B	
10	0.5
20	0.5

$E_6 =$ 20.984375

$\sigma =$ 4.3047666

Diagrams 5.1 to 5.4 illustrate effects of standard deviations of common activities A and B on the mean value and standard deviation of project completion time obtained using (a) Dodin's procedure (b) Proposed procedure, and (c) PERT method.

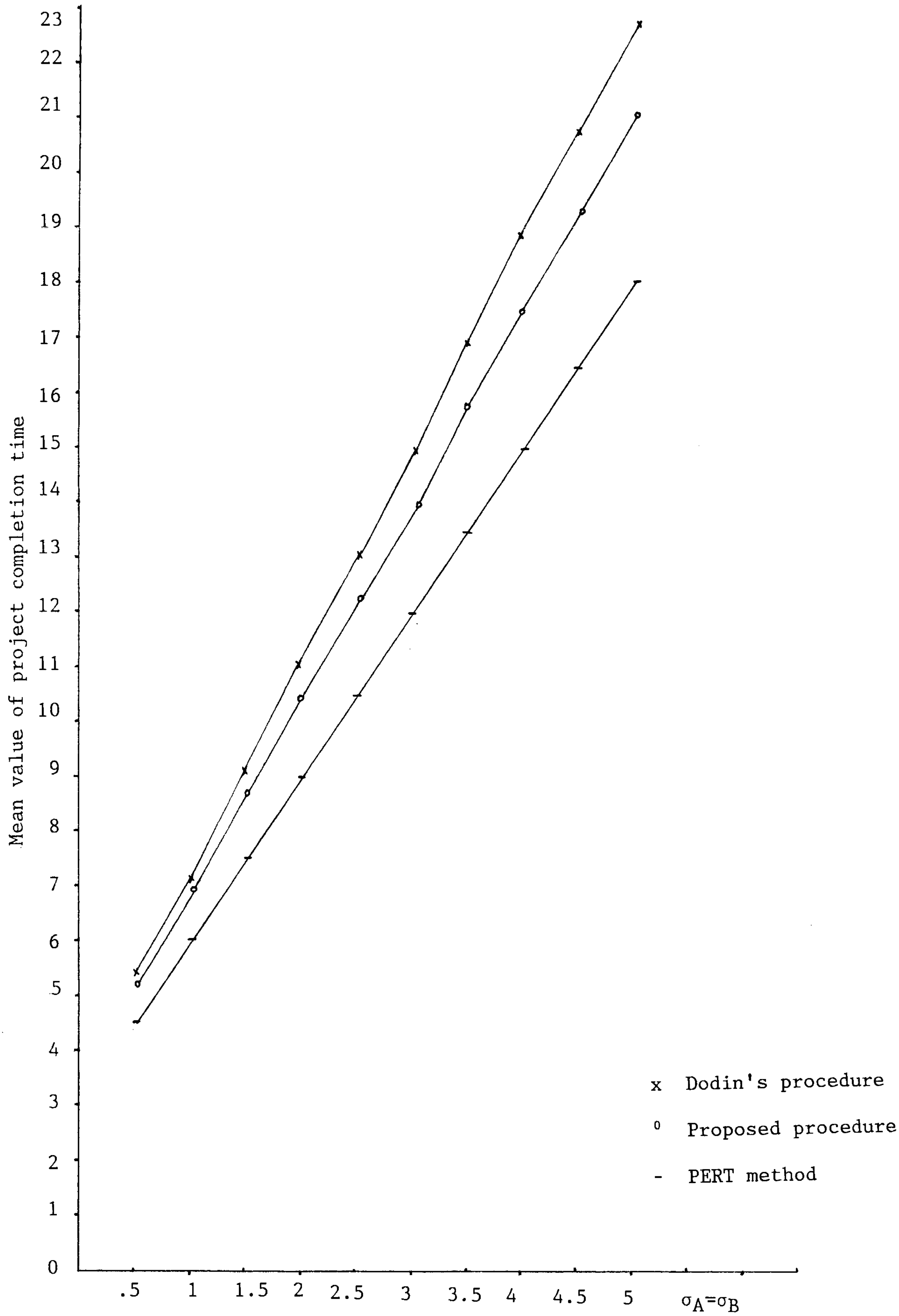


Diagram 5.1: Relationship between mean value of project completion time and standard deviation of common activities A and B.

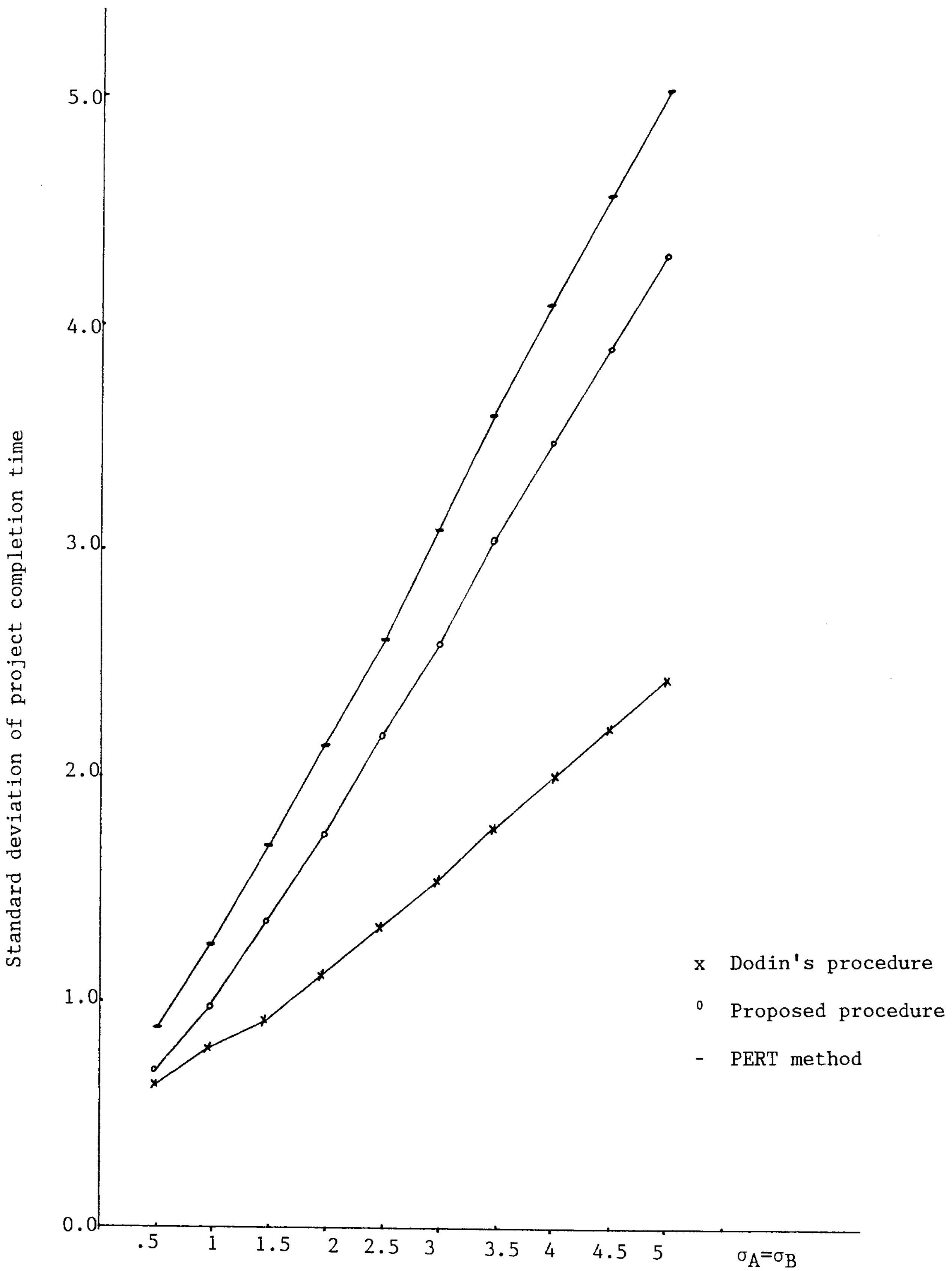


Diagram 5.2: Relationship between standard deviation of project completion time and standard deviation of common activities A and B.

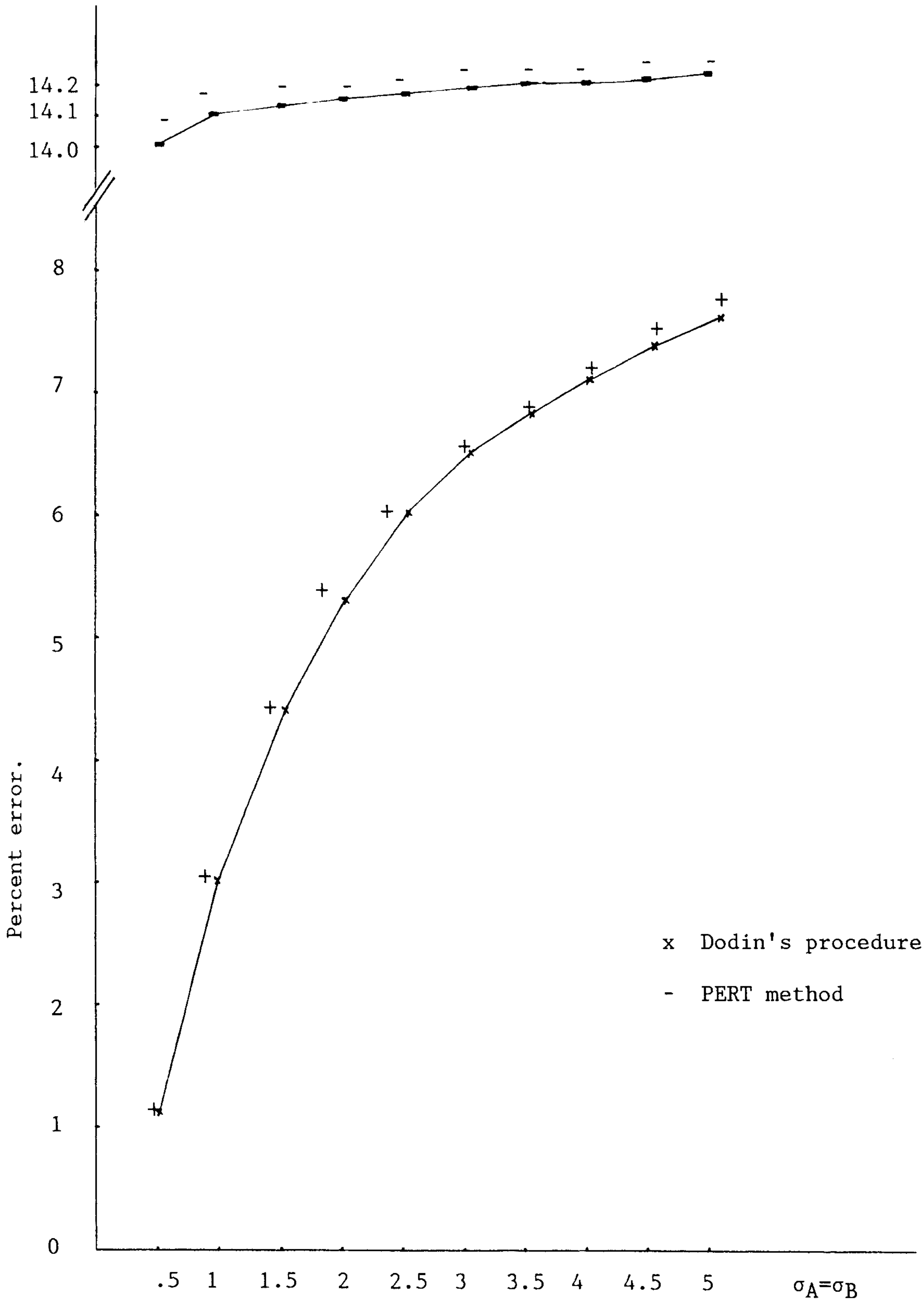


Diagram 5.3: Percent bias on the mean value of project completion time as a function of standard deviation of common activities A and B.

\- denotes optimistically biased and + denotes pessimistically biased.

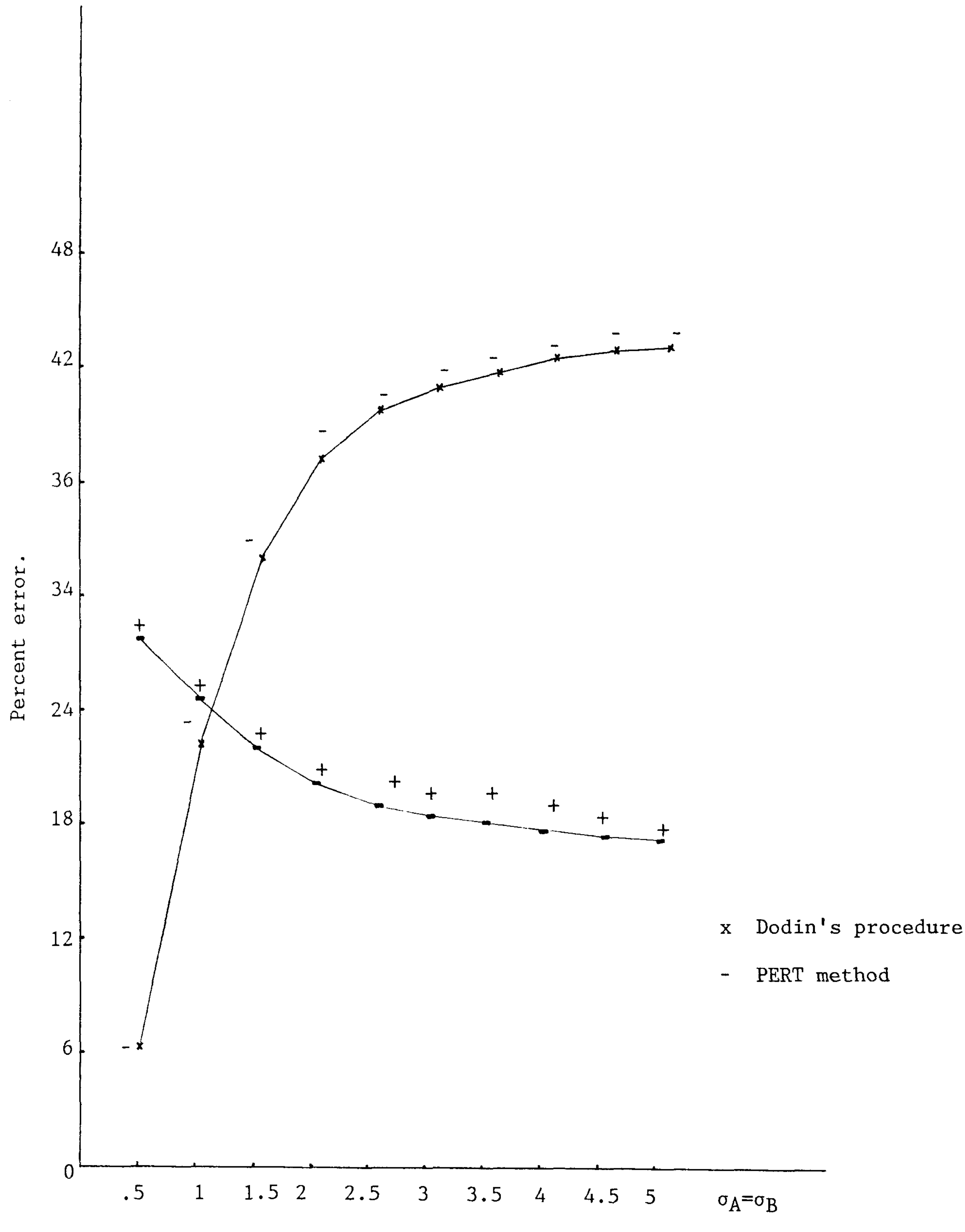


Diagram 5.4: Percent bias on the standard deviation of the project completion time as a function of standard deviation of common activities A and B.

- denotes optimistically biased and + denotes pessimistically biased.

Example 6.

Consider Wheatstone bridge of Figure 5.8. Table 5.28 shows duration times of activities. In order to evaluate effect of standard deviation of common activity A on the mean value and standard deviation of project completion times using Dodin's procedure and conventional PERT method we gradually increase the range of the duration time of activity A as shown in Table 5.29. Notice that for all different duration times of A mean is 10.

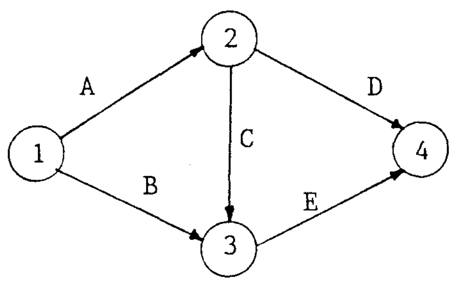


Figure 5.8

Table 5.28: Duration times of activities.

A		B		C		D		E	
X_A	P	X_B	P	X_C	P	X_D	P	X_E	P
9	0.25	19	0.25	9	0.25	9	0.25	1	0.25
10	0.5	20	0.5	10	0.5	10	0.5	2	0.5
11	0.25	21	0.25	11	0.25	11	0.25	3	0.25

Table 5.28: Different duration times of activity A.

A		A		A		A		A	
X_A	P	X_A	P	X_A	P	X_A	P	X_A	P
9	0.25	8	0.25	7	0.25	6	0.25	5	0.25
10	0.5	10	0.5	10	0.5	10	0.5	10	0.5
11	0.25	12	0.25	13	0.25	14	0.25	15	0.25
E = 10		10		10		10		10	
σ =.707		1.414		2.12		2.826		3.53	

A		A		A		A		A	
X_A	P	X_A	P	X_A	P	X_A	P	X_A	P
4	0.25	3	0.25	2	0.25	1	0.25	0	0.25
10	0.5	10	0.5	10	0.5	10	0.5	10	0.5
16	0.25	17	0.25	18	0.25	19	0.25	20	0.25
E= 10		10		10		10		10	
σ = 4.234		4.95		5.65		6.364		7.071	

Diagrams 5.5 to 5.8 illustrate effects of standard deviation of common activity A on the mean value and standard deviation of project completion time obtained using (a) Dodin's procedure (b) Proposed procedure, and (c) PERT method.

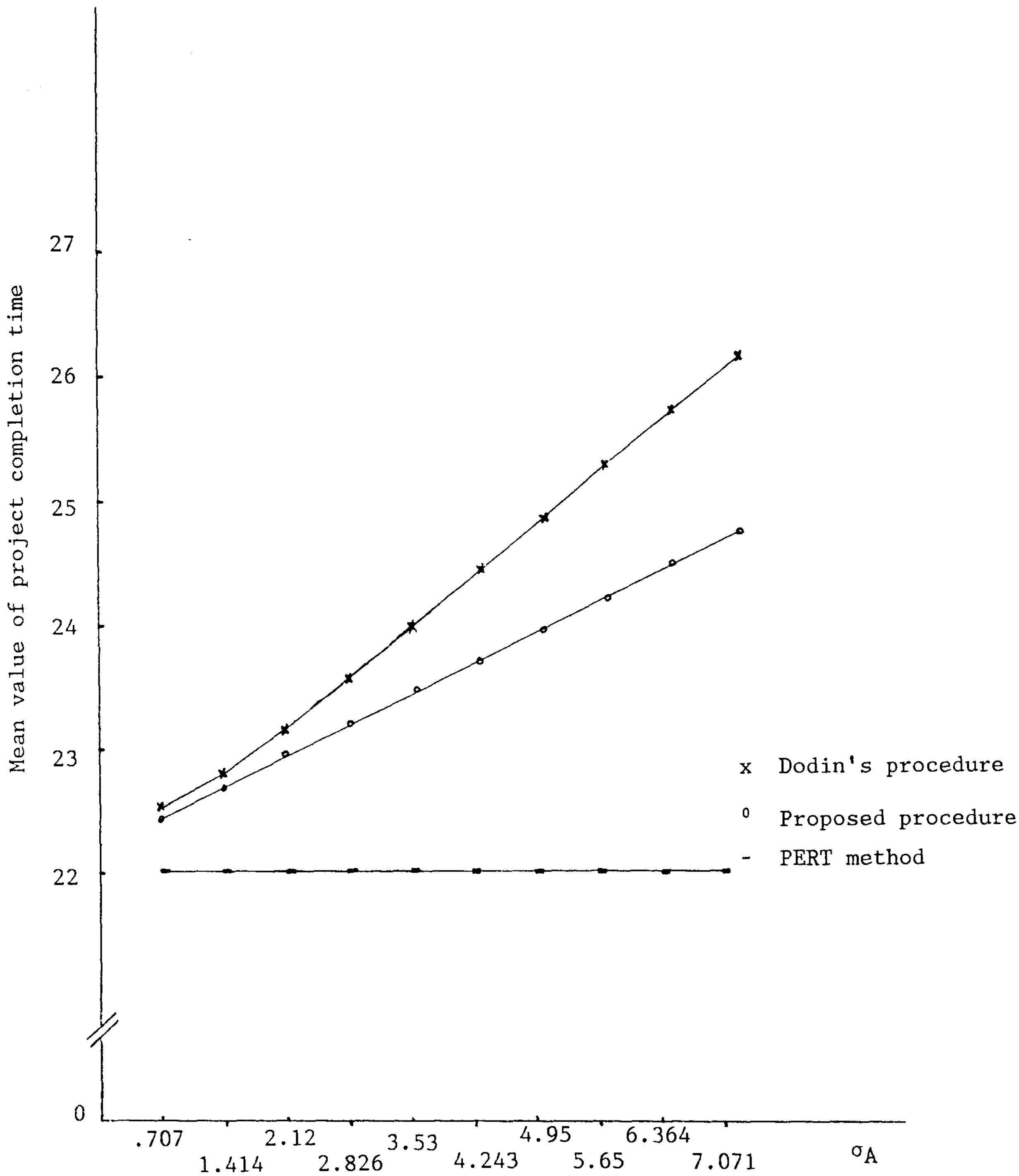


Diagram 5.5: Relationship between mean value of project completion time and standard deviation of common activity A.

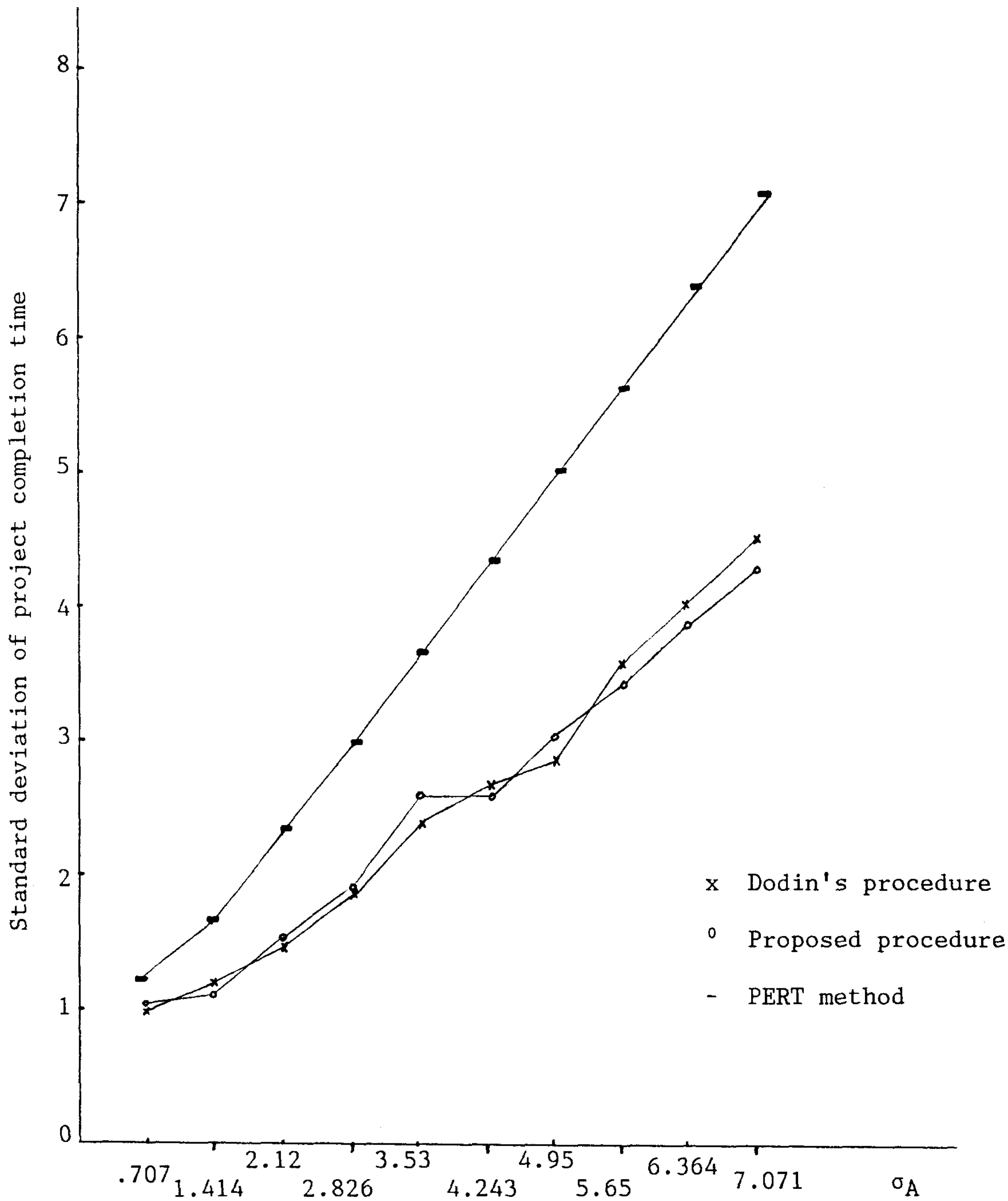


Diagram 5.6: Relationship between standard deviation of project completion time and standard deviation of common activity A.

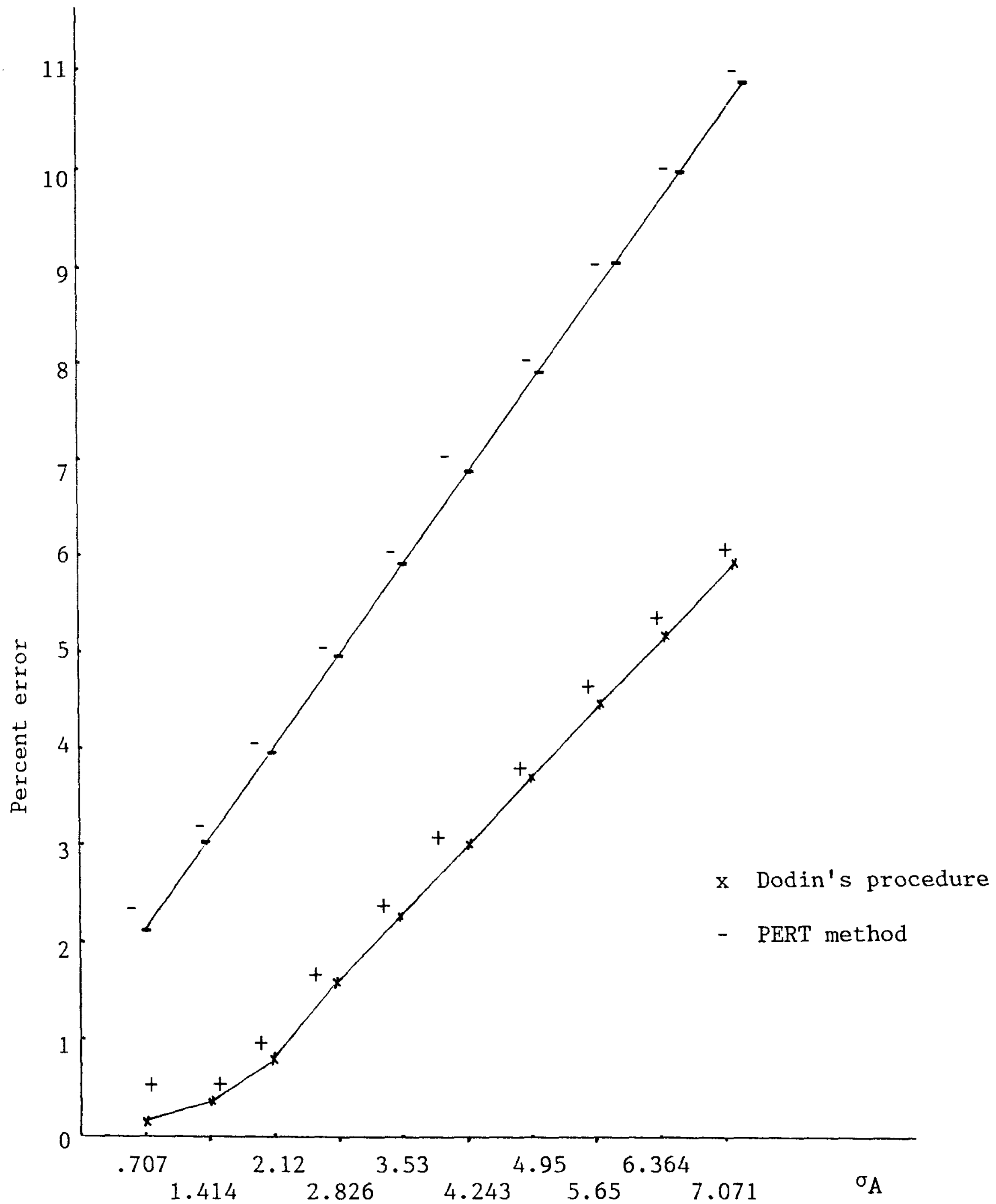


Diagram 5.7: Percent bias on the mean value of project completion time as a function of standard deviation of common activity A.

- denotes optimistically biased and + denotes pessimistically biased.

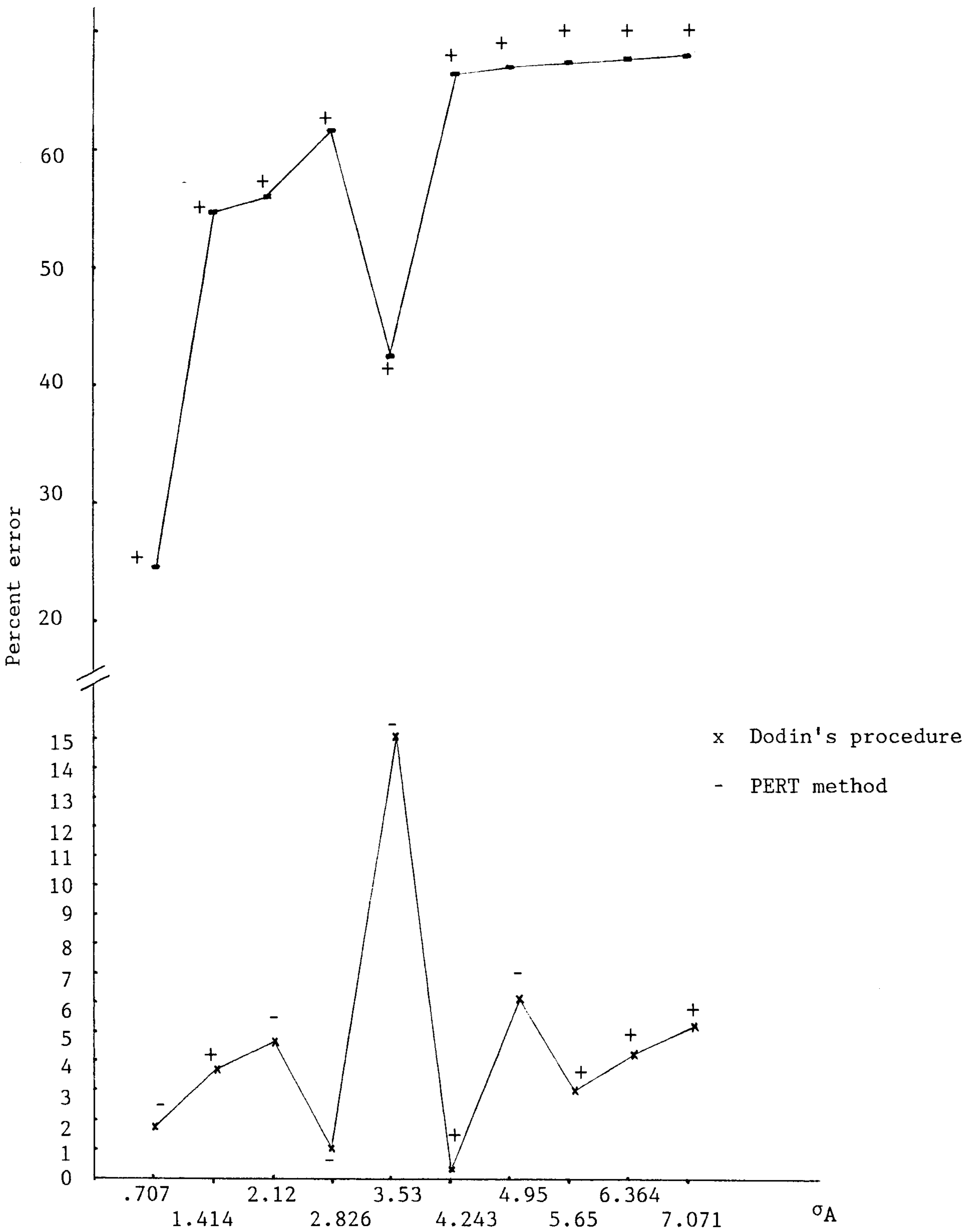


Diagram 5.8: Percent bias on the standard deviation of project completion time as a function of standard deviation of common activity A. - denotes optimistically biased and + denotes pessimistically biased.

Example 7.

Consider PERT network of Figure 5.9. Table 5.30 shows the duration times of activities. This network is originally presented in Mac Crimmon and Ryavec (1962).

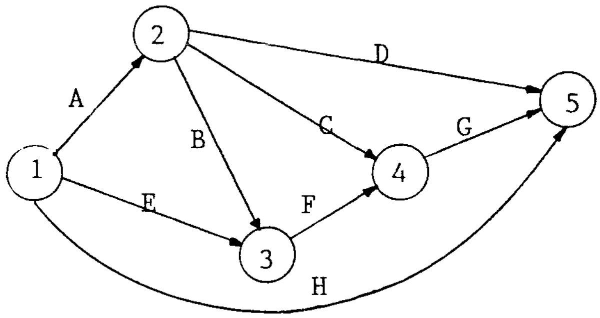


Figure 5.9

Table 5.30: Duration times of activities of Figure 5.9.

A,B,F and G		C and E		D		H	
X	P	X	P	X	P	X	P
1	0.2	1	0.2	2	0.2	3	0.2
2	0.6	3	0.6	5	0.6	7	0.6
3	0.2	5	0.2	8	0.2	11	0.2

Using proposed procedure the pdf of the project completion time which is equal to the exact pdf can be obtained as shown in Table 5.31.

Table 5.31: Exact project completion time.

	P
4	0.0000025
5	0.0002303
6	0.0059007
7	0.1326386
8	0.1882111
9	0.2250546
10	0.1936793
11	0.2526822
12	0.0016000
E =	9.2317133 $\sigma = 1.3931231$

Using Dodin's procedure the pdf of the project completion time can be obtained as shown in Table 5.32.

Table 5.32: Approximate project completion time using Dodin's procedure.

	P
5	0.0001146
6	0.0026084
7	0.1074626
8	0.2089091
9	0.2249392
10	0.1980924
11	0.2562737
12	0.0016000
d_5 =	9.2833219 $\sigma = 1.3495862$

Using PERT model calculation procedure the mean and standard deviation of project completion time can be computed as; mean = 8 and standard deviation = 1.264911. Table 5.33 summarizes the results.

Table 5.33: Summary of Results for Figure 5.9.

Calculation procedure	Project completion time
PERT - calculated mean	8.0
Exact (analytically)-calculated mean	9.2317133
Dodin's-calculated mean	9.2833219
per cent error (PERT from Exact)	-13.34
per cent error (Dodin's from Exact)	+ 0.56

PERT-calculated standard deviation	1.264911
Exact (analytically)-calculated standard deviation	1.3931231
Dodin's-calculated standard deviation	1.3495862
per cent error (PERT from Exact)	-9.20
per cent error (Dodin's from Exact)	-3.13

Diagrams 5.1 through 5.8 show that PERT-calculated mean is less than actual mean and PERT-calculated standard deviation is greater than actual standard deviation, while Table 5.33 shows that both PERT-calculated mean and standard deviation are less than Exact mean and standard deviation respectively. This is because, in PERT method the completion times of paths in the netowrk are assumed to be completely correlated, and the critical path cannot change. Also, Diagrams 5.1 through 5.8 and Table 5.33 show that mean completion time obtained using Dodin's procedure is biased optimistically, while Dodin's procedure calculated standard deviation is biased in either direction. The reason is that, in Dodin's procedure it is assumed that all paths are structurally independent.

In fact independence assumption among paths in Dodin's procedure is one of the sources of error in determination of the mean and standard deviation of the project completion times in irreducible networks and this bias is an increasing function of the standard deviation of common activities and the number of activities which emanate from the nodes in which these common activities terminate.

SUMMARY AND CONCLUSTIONS

In this chapter effects of structural dependence relationships in PERT networks has been shown through examples. It has demonstrated that conventional PERT procedure always leads to an optimistically biased estimate of the occurrence time for the network events, and the Approximating Procedure proposed by Dodin (1985a) which is based on the assumption of complete independence among paths in irreducible networks always leads to a pessimistically biased estimate of the occurrence time of events. On the other hand both PERT and Dodin's calculated standard deviation may be biased in either direction. Precise statement about the magnitude of the error, however, cannot be made since errors in the project mean and variance vary with different network configurations. The two more important factors affecting the magnitude of the merge event bias are as follows:

1 - The number of subcritical paths leading to a merge event.

2 - The variance of the subpaths lengths.

CHAPTER 6: PROPOSED PROCEDURE FOR CRITICALITY INDICES

INTRODUCTION

One of the more misleading aspects of conventional PERT methods is the implication that there is a unique critical path. In general, any of a number of paths could be critical, depending on the particular realization of the random activity durations that actually occur. Thus, it makes sense to talk about a "criticality index", which is the probability that an arc will be on the critical path (Van Slyke, 1963).

If we let P denote the set of all paths in the PERT network, and $Z(T_1)$ denote the duration of path $T_1 \in P$, then: $Z(T_1) = \sum_{(ij)\in T_1} Y_{ij}$ where Y_{ij} is duration of arc $(ij)\in A$.

The criticality of a path $T_1 \in P$ is measured by the probability that its duration $Z(T_1)$ is greater than or equal to the duration of all other paths. This probability is called the criticality index of the path, and is denoted by CP (Dodin and Elmaghraby, 1985). Therefore, for any path $T_1 \in P$:

$$CP(T_1) = Pr\ [Z(T_1) \geqslant Z(T_q) \text{ for all } T_q \in P; T_1 \neq T_q]. \quad (6.1)$$

The criticality index of an activity, denoted by CA, is defined by the sum of the criticality indices of the paths containing it; therefore, for any activity $(ij) \in A$:

$$CA(ij) = \sum_{\substack{T_1 \\ ((ij)\in T_1)}} CP(T_1)\ . \quad (6.2)$$

Evidently, the larger the value of CA(ij), the more crucial is the activity; and conversely. The CA appears to be an exceedingly useful measure of the degree of attention an activity should received by management, since it carries more pointed information than the basic critical path concept now used. Specifically, the CAs indicate which activities are the bottleneck activities and should be expedited if the entire project is to be expedited. It should be added that the probability of an activity being on the critical path is not correlated too well with slack, as computed by conventional PERT procedure which is the factor that usually determines the degree of attention that a particular activity receives. The following example may clarify these points.

Example 1.

Consider the network of Figure 6.1, with the customary time estimates of a, m, and b shown beside the corresponding activity. This example is originally presented in Mac Crimmon and Ryavec (1964) with three time estimates of activity B as 7-8-10.

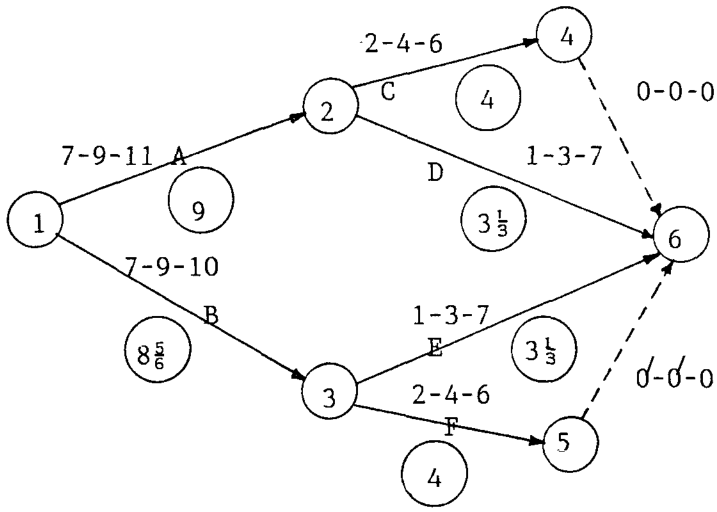


Figure 6.1 PERT network showing activities with associated times.

Using the PERT calculated mean times (given in the circle below the activities), the PERT procedure would choose 1-2-4-6 as the critical path because it has the maximum sum of means (13). Using the proposed procedure of next section, calculations will show that path 1-2-4-6 has the probability .372 of being the longest path, and this is a larger probability than any of the other three paths. The probability of

each activity being on the longest path is A,.515; B,.485; C,.372; D,.143; E,.135 and F,.35
Note that although path 1-2-4-6 is the most probable longest path, it does not contain activity B which is more critical than activity C, which is on this most probable longest path.

This example suggests that a critical activity concept may be more valid in a stochastic model than a critical path concept. In fact, the issue of determining the CPs and the CAs has received little attention in the literature. This is due to the difficulty in evaluating both expressions(6.1) and (6.2). The first attempt to approximate the CAs was introduced by Van Slyke (1963) using "crude" Monte Carlo simulation.

Martin (1965) defined the $CP(T_1)$ and the CA(ij) conceptually without suggesting how to obtain their values.

Sigel et al (1979) suggested the use of conditional Monte Carlo simulation to approximate CPs, then used them to approximate the CAs using (6.2). In the above three references, the evaluation of the CAs requires the enumeration of all the paths in the activity network, the approximation of the corresponding CPs, and the identification of the paths passing through each activity, all of which are burdensome, time consuming tasks.

Recently, theoretical results are developed by Dodin and Elmaghraby (1985) which lead to the first analytical approximating procedure to estimate the CAs without either using Monte Carlo simulation or identifying the paths and the CPs. This procedure (i) measures the "degree of criticality" of an individual activity, and (ii) can be used to generate the criticality list of the "M most critical activities". This procedure needs a small fraction of computing effort of Monte Carlo sampling, which is the only other practical approach to the determination of the criticality of an activity (Dodin and Elmaghraby, 1985).

Computation Procedure

Following section presents computation procedure for criticality indices of simple networks by using the definition of criticality index.

Example 3.

Consider PERT network of Figure 6.2.

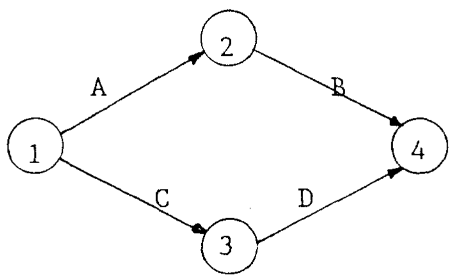


Figure 6.2

Table 6.1 shows the durations of the activities. Expected values and variances are also shown.

Table 6.1 Project activity durations.

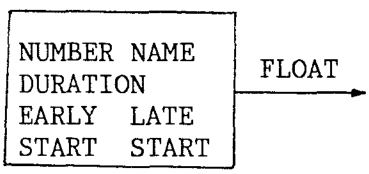
1.A	X_A =	4	5
	P =	0.6	0.4
	E = 4.4 , σ^2=0.24		
2.B	X_B =	6	7
	P =	0.4	0.6
	E = 6.6 , σ^2=0.24		

3.C	X_C	=	1	10
	P	=	0.1	0.9
	E = 9.1		σ^2 = 7.29	

4.D	X_D	=	1	2
	P	=	0.2	0.8
	E = 1.8		σ^2=0.16	

Using the calculated mean times, the PERT procedure would choose path (1-2-4) containing activities A and B the critical path because it has the maximum sum of means (4.4+6.6 = 11), while, using the proposed procedure, calculation will show that path 1-3-4 is more than twice as critical as path 1-2-4. Expected activity durations, earliest and latest activity start time, floats, mean and variance of project completion time are shown in precedence (Activity-on-node) diagram of Figure 6.2A.

KEY:



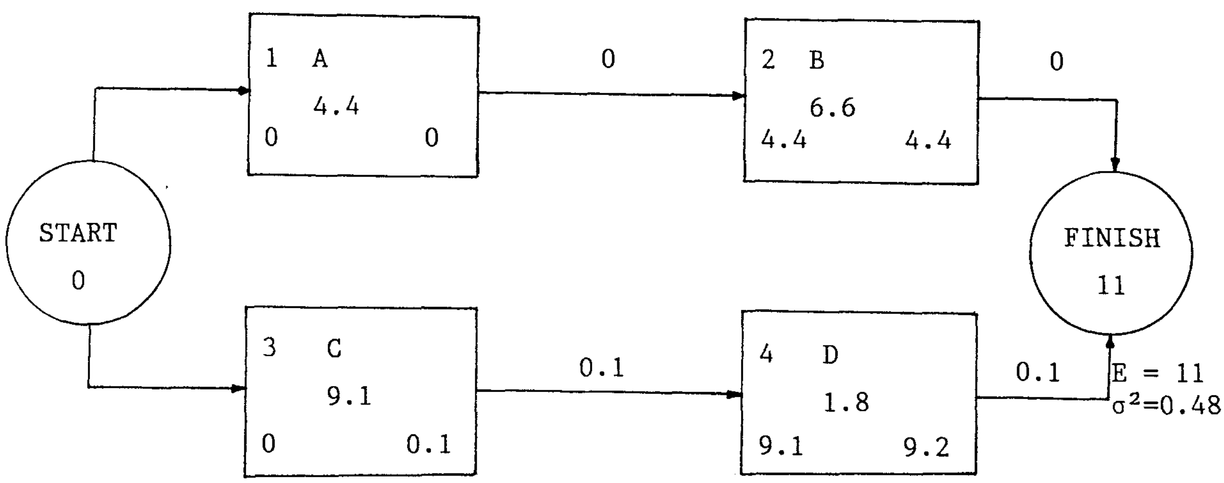


Figure 6.2A: Expected activity durations, activity early and late start time, mean and variance of project completion time and link floats.

The calculation procedure based on the definition of criticality index is as follows:

Using the definition of the convolution operator to A and B gives activity E, and also, convoluting activities C and D gives activity F as shown in Figure 6.3.

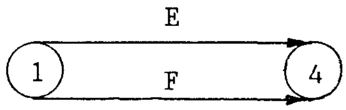


Figure 6.3

Duration time of E is given in Table 6.2 and Table 6.3 shows duration time of F (path 1-3-4).

Table 6.2: Duration time of E.

	P	CP
10	0.6 x 0.4 = 0.24	0.24
11	0.6 x 0.6 + 0.4 x 0.4 = 0.52	0.76
12	0.4 x 0.6 = 0.24	1.0

E = 11, $\sigma^2 = 0.48$

Table 6.3: Duration time of F.

	P	CP
2	0.1 x 0.20 = 0.02	0.02
3	0.1 x 0.8 = 0.08	0.10
11	0.9 x 0.20 = 0.18	0.28
12	0.9 x 0.80 = 0.72	1.0

E = 10.9 , $\sigma^2 = 7.45$

The criticality of path 1-2-4 is measured by the probability that its duration is greater than or equal to the duration time of all other paths, i.e., path 1-3-4 and the criticality of path 1-3-4 is measured by the probability that its duration is greater than or equal to the duration time of all other paths i.e. path 1-2-4. Hence,

Criticality Index of path 1-2-4 equals ,

$P(E=10).P(F \leqslant 10)+P(E=11).P(F \leqslant 11)+P(E=12).P(F \leqslant 12)$

$= 0.24 \times 0.1 + 0.52 \times 0.28 + 0.24 \times 1.0$

$= 0.4096$,

and

Criticality Index of path 1-3-4 equals ,

$P(F=2).P(E \leqslant 2)+P(F=3).P(E \leqslant 3)+P(F=11).P(E \leqslant 11)+P(F=12).P(E \leqslant 12)$

$= 0.02 \times 0.0 + 0.08 \times 0.0 + 0.18 \times 0.76 + 0.72 \times 1$

$= 0.8568$,

These operations are summarized in Table 6.4.

Table 6.4: Criticality Indices of paths 1-2-4 and 1-3-4.

	path 1-2-4	path 1-3-4
10	0.24 x 0.1 = 0.0240	
11	0.52 x 0.28 = 0.1456	0.18 x 0.76 = 0.1368
12	0.24 x 1 = 0.2400	0.72 x 1 = 0.7200
CP =	0.4096	0.8568
Normalized CP =	$\frac{0.4096}{0.4096+0.8568} = 0.323$,	$\frac{0.8568}{0.4096+0.8568} = 0.677$

As it is shown in Table 6.4 the sum of CP (0.4096 + 0.8568=1.2664) is greater than one, because when the duration time of each path is 11 or 12, two paths are critical simultaneously.

By normalizing criticality indices we change the sum of CPs to unity. Table 6.4 shows that path 1-3-4 is more than twice as critical as path 1-2-4, which has been chosen critical by PERT procedure. The reason is that, in PERT, only the means of the activity durations are used in determining of the critical path and the stochastic element - the variance of activity duration-is not incorporated (mean value of path 1-2-4 is 11 and its variance is 0.48, while mean value of path 1-3-4 is 10.9 and its variance is 7.45).

In this example, criticality index of activity A is equal to criticality index of activity B and equals 0.323, because these two are series activities and also criticality index of activity C is equal to criticality index of activity D and equals 0.677.

This example also suggests that a critical activity concept is more valid in a stochastic model than a critical path concept, especially since the PERT-calculated critical path is not even necessarily the most probable longest path.

Since all activities of Figure 6.2 are unique activities, therefore, criticality indices obtained in the above are exact, i.e. they are exactly the same as criticality indices obtained by complete enumeration as shown in the following. Tree diagram of Figure 6.4 shows critical path and probability of each realization time of the network of Figure 6.2.

Realization times of activities and probability of each realization.								Project realization time and critical path.		Probability of each realization.
A	P	B	P	C	P	D	P	AB	CD	
						1	0.2	10		0.0048
				1	0.1					
						2	0.8	10		0.0192
		6	0.4							
						1	0.2		11	0.0432
				10	0.9					
						2	0.8		12	0.1728
4	0.6									
						1	0.2	11		0.0072
				1	0.1					
						2	0.8	11		0.0288
		7	0.6							
						1	0.2	11	11	0.0648
				10	0.9					
						2	0.8		12	0.2592
						1	0.2	11		0.0032
				1	0.1					
						2	0.8	11		0.0128
		6	0.4							
						1	0.2	11	11	0.0288
				10	0.9					
						2	0.8		12	0.1152
6	0.4									
						1	0.2	12		0.0048
				1	0.1					
						2	0.8	12		0.0192
		7	0.6							
						1	0.2	12		0.0432
				10	0.9					
						2	0.8	12	12	0.1728
										1.0

Figure 6.4

Criticality index of path AB equals sum of the probabilities of realizations when path AB is critical, i.e.

CP(AB)=0.0048+0.0192+0.0072+0.0288+0.0648+0.0032+0.0128+0.0288+0.0048+0.0192+0.0432+0.1728=0.4096 ,

and similarly,

CP(CD)=0.0432+0.1728+0.0648+0.2592+0.0288+0.1152+0.1728 = 0.8568 .

Example 3.

Now, let us consider the PERT network of Figure 3.10 of Chapter 3.

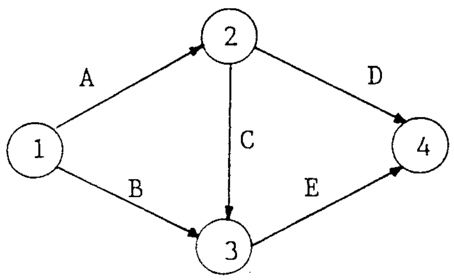


Figure 3.10

In order to determine the exact criticality index of each activity it is necessary that all path durations be independent, therefore, by fixing on the realization times of A and E, we change network of Figure 3.10 to networks of Figure 6.5 through 6.8 and all path durations would be independent.

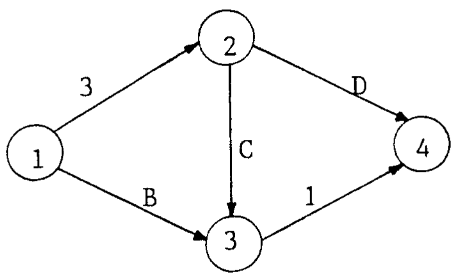


Figure 6.5

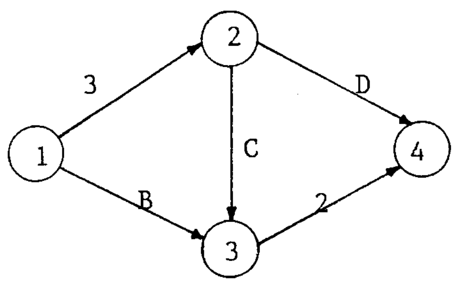


Figure 6.6

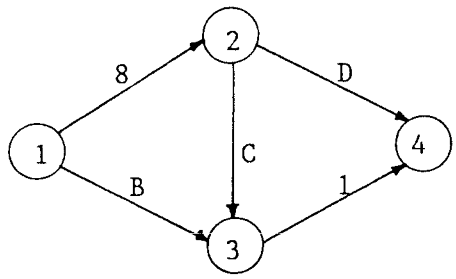


Figure 6.7

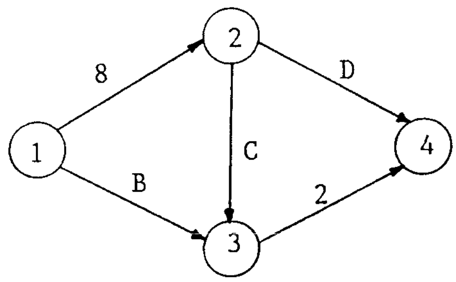


Figure 6.8

Now we determine the criticality indices of activities of Figure 6.5 through Figure 6.8.

Tables 6.5 through 6.7 show duration times of paths of Figure 6.5.

Table 6.5: Duration time of (3+D).

	P	CP
7	0.9	0.9
8	0.1	1.0

Table 6.6: Duration time of (3+C+1).

	P	CP
8	0.3	0.3
10	0.7	1.0

Table 6.7: Duration time of (B+1).

	P	CP
7	0.6	0.6
10	0.4	1.0

Table 6.8 shows project finish time given A=3 and E=1.

Table 6.8: Project finish time given A=3 and E=1.

	P
8	0.18
10	0.82
E =	9.64

Table 6.9 shows criticality indices of the paths given A=3 and E=1.

Table 6.9: Criticality indices of paths (3+D), (3+C+1) and (B+1).

	(3+D)	(3+C+1)	(B+1)
8	0.1x0.3x0.6 = 0.018	0.3 x1x0.6 = 0.18	-
10	-	0.7 x1x1 = 0.7	0.4x1x1 = 0.4
CP =	0.018	0.88	0.4

Table 6.10 through 6.12 show duration times of paths of Figure 6.6.

Table 6.10: Duration time of (3+D).

	P	CP
7	0.9	0.9
8	0.1	1.0

Table 6.11: Duration time of (3+C+2).

	P	CP
9	0.3	0.3
11	0.7	1.0

Table 6.12: Duration time of (B+2).

	P	CP
8	0.6	0.6
11	0.4	1.0

Table 6.13 shows project finish time given A=3 and E=2.

Table 6.13: Project finish time given A=3 and E=2.

	P
9	0.18
11	0.82
E =	10.64

Table 6.14 shows criticality indices of the paths given A=3 and E=2.

Table 6.14: Criticality indices of paths (3+D), (3+C+2) and (B+2).

	(3+D)	(3+C+2)	(B+2)
9	-	0.3x0.6x1=0.18	-
11	-	0.7x1x1 =0.7	0.4x1x1=0.4
CP=	0.0	0.88	0.4

Tables 6.15 through 6.17 show the duration times of paths of Figure 6.7.

Table 6.15: Duration time of (8+D).

	P	CP
12	0.9	0.9
13	0.1	1.0

Table 6.16: Duration time of (8+C+1).

	P	CP
13	0.3	0.3
15	0.7	1.0

Table 6.17: Duration time of (B+1).

	P	CP
7	0.6	0.6
10	0.4	1.0

Table 6.18 shows project finish time given A=8 and E=1.

Table 6.18: Project finish time given A=8 and E=1.

	P
13	0.3
15	0.7
E=	14.4

Table 6.19 shows criticality indices of the paths given A=8 and E=1.

Table 6.19: Criticality indices of paths (8+D), (8+C+1) and (B+1).

	(8+D)	(8+C+1)	(B+1)
13	0.1x0.3x1 = 0.03	0.3 x 1 x 1 = 0.3	-
15	-	0.7 x 1 x 1 = 0.7	-
CP =	0.03	1.0	0.0

Tables 6.20 through 6.22 shows the duration times of paths of Figure 6.8.

Table 6.20: Duration time of (8+D).

	P	CP
12	0.9	0.9
13	0.1	1.0

Table 6.21: Duration time of (8+C+2).

	P	CP
14	0.3	0.3
16	0.7	1.0

Table 6.22: Duration time of (B+2).

	P	CP
8	0.6	0.6
11	0.4	1.0

Table 6.23 shows the project finish time given A=8 and E=2.

Table 6.23: Project finish time given A=8 and E=2.

	P
14	0.3
16	0.7
E =	15.4

Table 6.24 shows the criticality indices of the paths given A=8 and E=2.

Table 6.24: Criticality indices of paths (8+D), (8+C+2) and (B+2).

	(8+D)	(8+C+2)	(B+2)
14	-	0.3x1x1 = 0.3	-
16	-	0.7x1x1 = 0.7	-
CP =	0.0	1.0	0.0

By deconditioning the expected values of project finish times given in Tables 6.8, 6.13, 6.18 and 6.23 the exact value of project finish time can be computed as follows:

E = 9.64x0.4+10.64x0.4+14.4x0.1+15.4x0.1

= 11.092 .

Also, by deconditioning the criticality indices of paths given in Tables 6.9, 6.14, 6.19 and 6.24 the exact value of criticality indices of paths and activities can be computed as follows:

CA(B) = 0.4x0.4+0.4x0.4+0.0x0.1+0.0x0.1

= 0.32 ,

CA(C) = 0.88x0.4+0.88x0.4+1x0.1+1x0.1

= 0.904 ,

CA(D) = 0.018x0.4+0.0x0.4+0.03x0.1+0.0x0.1

= 0.0102 ,

CA(A) = CA(C)+CA(D)

= 0.904+0.0102

= 0.9142 ,

$CA(E) = CA(B)+CA(C)$

$= 0.32+0.904$

$= 1.224$.

Criticality index of each path equals the minimum of the criticality index of the activities on that path. Therefore,

CP(path 1-2-4) = min {CA(A), CA(D)}

= min {0.9142, 0.0102}

= 0.0102 ,

CP(path 1-2-3-4) = min {CA(A), CA(C), CA(E)}

= min {0.9142, 0.904, 1.224}

= 0.904 ,

CP(path 1-3-4) = min {CA(B), CA(E)}

= min {0.32, 1.224}

= 0.32 .

Criticality indices of activities are shown in Figure 6.9.

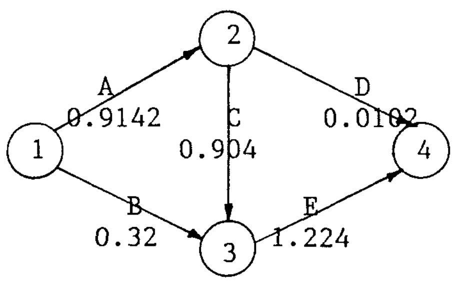


Figure 6.9

Normalized criticality indices of activities are shown in Figure 6.10.

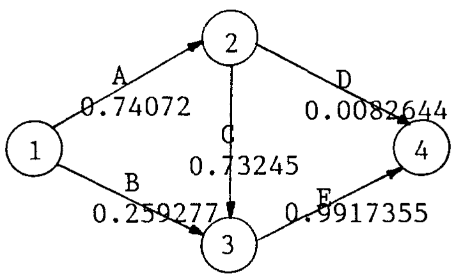


Figure 6.10

These exact values also can be obtained by complete enumeration as shown in tree diagram of Figure 6.11.

Realization times of activities and probability of each realization.										Project realization time and critical path.			Probability of each realization
A	P	B	P	C	P	D	P	E	P	1-2-4	1-2-3-4	1-3-4	
3	0.8	6	0.6	4	0.3	4	0.9	1	0.5		8		0.0648
								2	0.5		9		0.0648
						5	0.1	1	0.5	8	8		0.0072
								2	0.5		9		0.0072
				6	0.7	4	0.9	1	0.5		10		0.1512
								2	0.5		11		0.1512
						5	0.1	1	0.5		10		0.0168
								2	0.5		11		0.0168
		9	0.4	4	0.3	4	0.9	1	0.5			10	0.0432
								2	0.5			11	0.0432
						5	0.1	1	0.5			10	0.0048
								2	0.5			11	0.0048
				6	0.7	4	0.9	1	0.5		10	10	0.1008
								2	0.5		11	11	0.1008
						5	0.1	1	0.5		10	10	0.0112
								2	0.5		11	11	0.0112
8	0.2	6	0.6	4	0.3	4	0.9	1	0.5		13		0.0162
								2	0.5		14		0.0162
						5	0.1	1	0.5	13	13		0.0018
								2	0.5		14		0.0018
				6	0.7	4	0.9	1	0.5		15		0.0378
								2	0.5		16		0.0378
						5	0.1	1	0.5		15		0.0042
								2	0.5		16		0.0042
		9	0.4	4	0.3	4	0.9	1	0.5		13		0.0108
								2	0.5		14		0.0108
						5	0.1	1	0.5	13	13		0.0012
								2	0.5		14		0.0012
				6	0.7	4	0.9	1	0.5		15		0.0252
								2	0.5		16		0.0252
						5	0.1	1	0.5		15		0.0028
								2	0.5		16		0.0028

Figure 6.11

Suppose we apply the proposed procedure of Chapter 4 and conditionalize A only. Then all the paths will not be independent but the network would be subject to series-parallel reduction and the approximate value of criticality index of each activity and path can be obtained as follows:

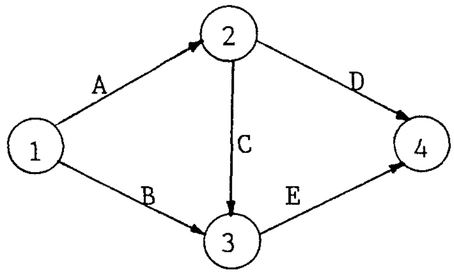


Figure 3.10

When A=3, the approximate criticality indices of activities (D+3) and G, where G is equivalent of series activities F and E are determined from Tables 4.18A and 4.18E.

Table 4.18A: Duration time of (D+3).

	P	CP
7	0.9	0.9
8	0.1	1.0

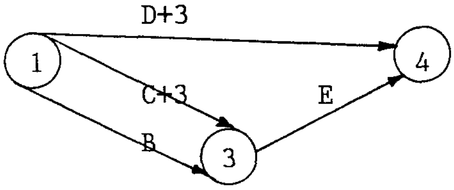


Figure 4.5A

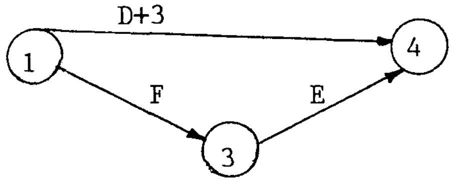


Figure 4.6A

Table 4.18E: Duration time of G=F*E.

	P	CP
8	0.18 x 0.5 = 0.09	0.09
9	0.18 x 0.5 = 0.09	0.18
10	0.82 x 0.5 = 0.41	0.59
11	0.82 x 0.5 = 0.41	1.0

Table 6.25 shows the approximate criticality indices of (D+3) and G.

Table 6.25: Approximate criticality indices of (D+3) and G.

	(D+3)	G
8	0.1x0.09 = 0.009	0.09x1 = 0.09
9		0.09x1 = 0.09
10		0.41x1 = 0.41
11		0.41x1 = 0.41
CAP =	0.009	1.0
Normalized CAP = (NCAP) =	0.0089	0.9911

Criticality index of E equals 0.9911 and also criticality index of F equals 0.9911, F is resultant of two parallel activities (C+3) and B, therefore, criticality indices of (C+3) plus B is equal to criticality index of F. Tables 4.18B and 4.18C show duration times of (C+3) and B respectively.

Table 4.18B: Duration time of (C+3).

	P	CP
7	0.3	0.3
9	0.7	1.0

Table 4.18C: Duration time of B.

	P	CP
6	0.6	0.6
9	0.4	1.0

In order to determine criticality indices of (C+3) and B, first, normalized criticality indices of each one is determined, then by multiplying each normalized CAP by criticality index of F, criticality index of each one is obtained as shown in Table 6.26.

Table 6.26: Approximate criticality indices of (C+3) and B.

	(C+3)	B
7	0.3x0.6 = 0.18	
9	0.7x1 = 0.7	0.4x1 = 0.4
CAP =	0.88	0.4
NCAP =	0.6875	0.3125
NCAP • CAP(F) =	0.6875 x 0.9911 = 0.6814	0.3125x0.9911 = 0.3097

As a proposition in the next section we demonstrate that for any node i≠1, N in a PERT network, the sum of the criticality indices of the arcs ending in node i equals the sum of the criticality indices of the arcs emanating from node i. Therefore, in this example, criticality index of A is equal to sum of the criticality indices of C and D. Note that criticality index of (C+3) is in fact criticality index of C.

Table 6.27 shows the approximate criticality indices of all activities given A=3.

Table 6.27: Approximate criticaltiy indices of activities given A=3.

Activity	CAP
E	0.9911
D	0.0089
C	0.6814
B	0.3097
A 0.6814+0.0089 =	0.6903

Given A=8, the approximate criticality indices of activities (D+8), and I which is equivalent of series activities E and H are determined from Table 4.18G and Table 4.18K.

Table 4.18G: Duration time of (D+8).

	P	CP
12	0.9	0.9
13	0.1	1.0

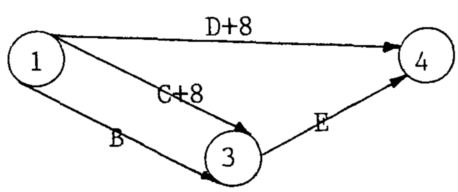


Figure 4.5B

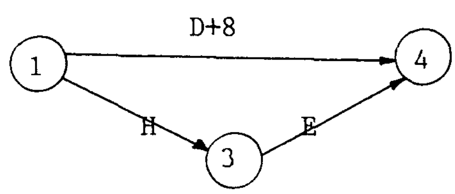


Figure 4.6B

Table 4.18K Duration time of I=H*E.

	P	CP
13	0.15	0.15
14	0.15	0.30
15	0.35	0.65
16	0.35	1.0

Table 6.28 shows the approximate criticality indices of (D+8) and I.

Table 6.28: Approximate criticality indices of (D+8) and I.

	(D+8)	I
13	0.1x0.15 = 0.015	0.15x1=0.15
14		0.15x1=0.15
15		0.35x1=0.35
16		0.35x1=0.35
CAP =	0.015	1.0
NCAP =	0.0148	0.9852

The approximate criticality index of E is equal to 0.9852 and also the approximate criticality index of H is equal to 0.9852. H is the resultant of two parallel activities (C+8) and B, therefore, the appproximate criticality indices of (C+8) plus B is equal to the approximate criticality index of H. Table 4.18H and Table 4.18I show the duration times of (C+8) and B respectively.

Table 4.18H: Duration time of (C+8)

	P	CP
12	0.3	0.3
14	0.7	1.0

Table 4.18I: Duration time of B.

	P	CP
6	0.6	0.6
9	0.4	1.0

Table 6.29 shows the approximate criticaltiy indices of (C+8) and B.

Table 6.29: Approximate criticality indices of (C+8) and B.

	(C+8)	B
12	0.3x1 = 0.3	-
14	0.7x1 = 0.7	-
CAP =	1.0	0
CAP.CAP(F) =	1x0.9852 = 0.9852	0

Table 6.30 shows the approximate criticality indices of all activities given A=8.

Table 6.30: Approximate criticality indices of activities given A=8.

Activity		CAP
E		0.9852
D		0.0148
C		0.9852
B		0.0
A	0.0148 +0.9852 =	1.0

By deconditioning the approximate criticality indices of Tables 6.27 and 6.30 the approximate criticality indices of network of Figure 3.10 is obtained.

Multiplying entries of Table 6.27 by (P(A=3)=.8) and enteries of Table 6.30 by (P(A=8)=.2) and simple addition of the approximate criticality indices of corresponding values gives the normalized value of the approximate criticality index of each activity as shown in Table 6.31 and Figure 6.11.

Table 6.31 Normalized values of the approximate criticality indices of activities.

Activity	NCAP
E	0.9911x0.8+0.9852x0.2 = 0.9899
D	0.0089x0.8+0.0148x0.2 = 0.0101
C	0.6814x0.8+0.9852x0.2 = 0.7421
B	0.3097x0.8+0.0x0.2 = 0.2478
A	0.6903x0.8+1x0.2 = 0.7522

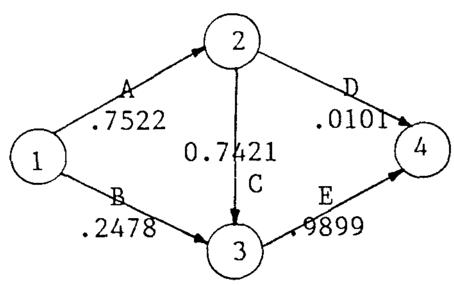


Figure 6.11

The approximate criticality index of each path (CPP) equals the minimum of the approximate criticality index of the activities which are on this path. Hence the approximate criticality index of path 1-2-4 of Figure 6.4 equals min {CAP(A), CAP(D)}

= min {0.7522, 0.0101}

= 0.0101 ,

similarly, CPP (path 1-2-3-4) = min {CAP(A), CAP(C), CAP(E)}

= min {0.7522, 0.7421, 0.9899}

= 0.7421 ,

and CPP(path 1-3-4) = min {CAP(B), CAP(E)}

= min {0.2478, 0.9899}

= 0.2478 .

Notice that values shown in Table 6.31 and Figure 6.11 are the normalized values of the approximate criticality indices obtained using proposed procedure. Table 6.32 shows the exact values of criticality indices (CA) obtained from the tree diagram of Figure 6.11 and also shows the normalized values of exact criticality indices (NCA)

Table 6.32:

Activity	CA	NCA
E	1.224	0.9917
D	0.0102	0.0083
C	0.904	0.7324
B	0.32	0.2593
A	0.9142	0.7407

In order to determine the goodness of proposed procedure, in the following we calculate the correlation coefficient between normalized values of exact criticality indices and normalized values of criticality indices obtained using proposed procedure.

Activity	NCA X	NCAP Y	$x=X-\bar{X}$	$y=Y-\bar{Y}$	x^2	y^2	xy
E	0.9917	0.9899	0.44522	0.44148	0.1982208	0.1949045	0.1965557
D	0.0083	0.0101	-.53818	-.53832	0.2896377	0.2897884	0.2897130
C	0.7324	0.7421	0.18592	0.19368	0.0345662	0.0375119	0.0360089
B	0.2593	0.2478	-.28718	-.30062	0.0824723	0.0903723	0.0863320
A	0.7407	0.7522	0.19422	0.20378	0.0377214	0.0415262	0.0395781
	$\Sigma X=$ 2.7324 $\bar{X}=$ 0.54648	$\Sigma Y=$ 2.7421 $\bar{Y}=$ 0.54842			$\Sigma x^2=$ 0.6426184	$\Sigma y^2=$ 0.6541033	$\Sigma xy=$ 0.6481877

Correlation Coefficient: $r_{X,Y}=\frac{\Sigma(xy)}{\sqrt{(\Sigma x^2)(\Sigma y^2)}}=\frac{0.6481877}{\sqrt{(0.6426184)(0.6541033)}}$

$= 0.9997721.$

Before introducing the proposed procedure, next section presents a summary of first analytical approximating procedure to estimate the criticality indices proposed by Dodin and Elmaghraby (1985).

APPROXIMATING THE CRITICALITY INDICES OF THE ACTIVITIES IN PERT NETWORKS

Notation

The following is a list of the notations used throughout this paper:

A: the set of activities (arcs) in the network G(N,A); also the cardinality of the set.

ACAP(ij): the approximate criticality index of activity (ij).

C_j: the cutset at node j.

CA(ij): the exact criticality index of activity (ij).

CAP(ij): the probability that the maximum of the paths in L(ij) is longer than the maximum of the paths in the complement set $\bar{L}(ij)$. It is a lower bound on the value of CA(ij).

CI: criticality index.

CN(i): the criticality index of node i=the sum of the CIs of the paths containing node i.

CP: critical path (defined only for DANs).

$CP(T_h)$: the criticality path index of path T_h= the probability that path T_h is not shorter than any other path.

DAN: deterministic activity network.

$E[Z(T_1^r)]$: the mathematical expectation of the r.v. $Z(T_1^r)$.

L(ij): the set of paths containing arc (ij). Its complement is $\bar{L}(ij)=P-L(ij)$.

MCS: Monte Carlo Sampling.

N: the set of nodes in the network G(N,A); also the cardinality of the set.

n_j: the in-degree of node j.

P: the set of paths from node 1 to node N: also the cardinality of the set.

T_h: the hth path; $T_h \epsilon P$.

PAN: probabilistic activity network.

pdf: probability distribution function.

r.v. random variable

T_i^f: the duration of the longest path "forward" from node 1 to node i.

T_j^b: the duration of the longest path "backwards" from node N to node j.

V_{ij}: the duration of the longest path not containing arc (ij).

W_{ij}: the duration of the longest path containing arc (ij).

Y_{ij}: the duration of activity (ij), a r.v.

$Z(T_h)$: the duration of path $T_h \epsilon P$.

1. Introduction

One of the main purposes of using network analysis for project planning and control stems from the need to identify the paths and activities that are critical to the achievement of the project objectives. In deterministic activity network (DANs), in which the activity duration is a constant, it is relatively easy to answer questions such as: which is the critical path(s)? Which are the M most critical activities? Which is the critical list (CL) of activities? (The Critical List is a list of the activities ranked in decreasing order of their criticalities).

In PANs, such as the PERT model, one must phrase these questions in probabilistic terms, such as: Which path (or paths) is the most probable to be critical? Which activities are the most critical activities? Can the activities be ranked in decreasing order of their relative criticalities?

These and other questions are the subject matter of this paper. To guide the reader through the development, Dodin and Elmaghraby (1985) explain their strategy as follows. They first give precise definitions to the (probabilistic) notions of path and activity criticalities. It will then become evident that the definitional expressions cannot be used for computing purposes because they demand the complete enumeration of the paths in the network, and the determination of the criticality of each path-an onerous task at best for any realistic network. Thus they are driven to define a surrogate measure to the (exact) criticality of an activity. They demonstrate that the surrogate measure always underestimates the exact measure. Thus the price paid for not having to enumerate the paths and determine their individual criticalities is not to be able to determine the criticality index exactly, but only to bound it from below. Finally, and in order to avoid the evaluation of multiple integrals that are required by the surrogate measure, they apply the approximating procedure developed in Dodin (1980) to calculate the surrogate measure. They demonstrate, through extensive computing experience, that the approximation of the surrogate measure is quite good. They therefore achieve the answers to the questions posed above at a miniscule fraction of the computing effort of Monte Carlo Sampling (MSC). This brings these concepts to within the grasp of operating managers. In the following section theoretical results are developed which lead to the first analytical approximating procedure to estimate the CAs without either using MCS or identifying the paths and their CPs. Such a procedure is a direct application of the approximation of the distribution function (df) of the project completion time developed by Dodin (1980), which was briefly outlined in Chapter 5.

The procedure starts at node N, approximates the CAs of the arcs ending in node N, then proceeds recursively to nodes N-1, N-2 and so on until it finally reaches node 2 where the CA of arc (1,2) is estimated. The procedure and an illustrative example are presented in section 3. A suggestion of its accuracy, based on computational experience, is the subject of section 4.

2. Theoretical Results

For any arc $(ij)\epsilon A$, let W_{ij} denote the duration of the longest path containing arc(ij). We are interested in the distribution function of W_{ij}. For computing purposes we split W_{ij} into three segments as follows:

$$W_{ij} = T_i^f + Y_{ij} + T_j^b \qquad (6.3)$$

where $T_i^f = \max_{T_{1i}} \{Z(T_{1i})\}$, and $T_j^b = \max_{T_{Nj}} \{Z(T_{Nj})\} = \max_{T_{jN}} \{Z(T_{jN})\}$. Basically, T_i^f measures the duration of the longest path "forward" from node 1 to node i, while T_j^b measures the duration of the longest path "backwards" from N to node j. The rationale for this split will become evident momentarily. Clearly, any path from node 1 to node N containing arc (ij), say T_1, consists of three segments: the subpath T_{1i}, arc (ij) and the subpath T_{jN}. Hence the duration of T_1 is

$$Z(T_1)=Z(T_{1i}(1))+Y_{ij}+Z(T_{jN}(1)). \qquad (6.4)$$

Consequently, if the number of subpaths of the form T_{1i} is equal to m and the number of subpaths of the form T_{jN} is equal to n, then the number of paths containing arc (ij) is equal to mn. This discussion results in the following proposition:

PROPOSITION 1. $W_{ij} = \text{Max}_{((ij)\epsilon T_1)}\{Z(T_1)\}$, where $Z(T_1)$ is as defined in (6.4).

The proof is by direct substitution in the definition of W_{ij} in (6.3), and is relegated, with other proofs, to the section 6.

Let L(ij) denote the set of paths containing arc (ij), i.e.:

$$L(ij) = \{T_1 \epsilon P:(ij) \epsilon T_1\}. \qquad (6.5)$$

Now, the exact value of the CA of arc (ij), defined in (6.2), is given by:

$$CA(ij) = \Sigma CP(T_1)= \Sigma Pr(Z(T_1) \geqslant Z(T_q) \text{ for all } T_q \epsilon P; \quad T_1 \neq T_q) \qquad (6.6)$$

where the summation are taken over $T_1 \epsilon L(ij)$. In words, CA(ij) measures the "weight" attached to the event that any $T_1 \epsilon L(ij)$ is longer than any other path in the network. This weight is a measure of the criticality of activity (ij). Unfortunately, it is extremely difficult to calculate CA(ij) directly from (6.6). To eliminate this difficulty, we appeal to the r.v. W_{ij} defined above and to the concept of "directed cutset", which is defined as follows (see Figure 6.12): the directed cutset at node j for j>1, denoted by C_j, is the set of arcs connecting the nodes with numbers less than j to the nodes with numbers greater than or equal to j, i.e.

$$C_j= \{(ik) \epsilon A: i<j \text{ and } k \geqslant j\}. \qquad (6.7)$$

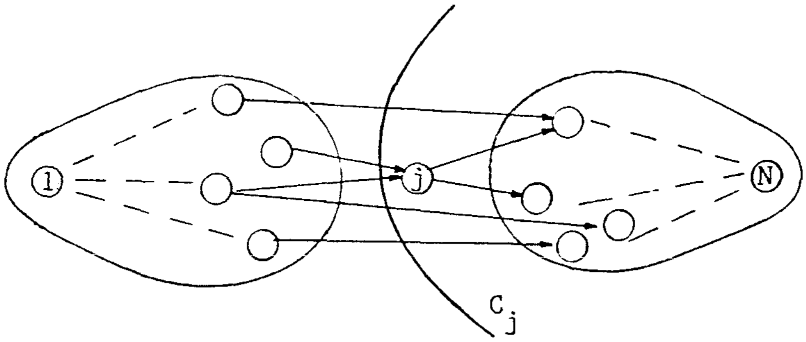


Figure 6.12. The Directed Cutset C_j.

Clearly, any path from node 1 to node N, say T_1, must pass through only one arc of the set C_j. Therefore, if $|C_j|$ is the cardinality of the set C_j, then the set of all paths in the network, P, can be partitioned into $|C_j|$ subsets. They constitute the basis for the definition of the set L(ij). Let

$$CAP(ij) = Pr\ [W_{ij} \geqslant W_{1k} \text{ for all } (1k) \in C_j;\ (ij) \neq (1k)] \quad (6.8)$$

where C_j is as defined in (6.7). Close scrutiny of (6.8) reveals that CAP(ij) measures the probability that the maximum of the paths in the subset L(ij) is longer than the maximum of the paths not containing (ij), i.e. in the complementary set $\bar{L}(ij)=P-L(ij)$. The latter paths are precisely the paths that contain all arcs in C_j except arc (ij). We assert below that CAP(ij) always underestimates CA(ij).

PROPOSITION 2.

$$CAP(ij) \leqslant CA(ij) \text{ for all } (ij) \in A. \quad (6.9)$$

To facilitate the use of equation (6.8) to approximate the criticality indices of some or all the activities in PERT networks we apply the following proposition:

PROPOSITION 3.

For any node $i \neq 1$, N in a PERT network, the sum of the CAs of the arcs ending in node i equals the sum of the CAs of the arcs emanating from node i, i.e.,

$$\sum_{j \in B(i)} CA(ji) = \sum_{j \in A(i)} CA(ij) \qquad (6.10)$$

where A(i) is the set of arcs emanating from node i, and B(i) is the set of arcs incident into node i.

The following two results are immediate consequence of Proposition 3:

COROLLARY 1. The criticality indices of nodes 1 and N are identical and equal to the criticality index of any directed cutset, which is given by the constant value $\Sigma_{T_1 \epsilon P} CP(T_1)$, i.e if CN denotes the criticality index of the node, then

$$CN(1) = CN(N) = \sum_{T_1 \epsilon P} CP(T_1) \geqslant 1.0. \qquad (6.11)$$

The two equalities in (6.11) are rather obvious; the inequality follows from the definition of $CP(T_1)$ as probability, and the fact that in an activity network two or more paths may be critical simultaneously.

COROLLARY 2. The CP of any path is equal to the CA of any unique arc on the path (an arc belonging to no other path). Moreover, all unique arcs on the same path have the same criticality index.

Proposition 3 can be used to approximate the CAs of some arcs without using (6.8). If arc(ij) ϵ A is the only arc terminating in node j then

$$CA(ij) = CN\ (j) = \sum_{k \epsilon A(j)} CA(jk).$$

However, if there are n arcs terminating in node j and the CAs of n-1 of these are known, then the remaining CA(ij) is obtained from the equation:

$$CA(ij) = CN(j) - \sum_{\substack{k \in B(j) \\ (k \neq i)}} CA(kj) .$$

We recapitulate the development thus far. We have concentrated on determining the criticality index of an arc CA(ij) because it is the key to answering the questions raised in the Introduction. To circumvent the need to enumerate the paths in the network we defined the "directed cutset" at a node j and the new measure CAP(ij) as a surrogate for CA(ij), and asserted that CAP(ij) underestimates CA(ij).

We now concern ourselves with the computation of CAP(ij). One of the difficulties in implementing expression (6.7) is the identification of the elements of C_j.
This difficulty is resolved using the following proposition:

PROPOSITION 4.

For any node $j \neq 1$ the set C_j is given by the difference equation

$$C_j = C_{j+1} + \{(ij) \in A: \ i < j\} - \{(jk) \in A: k > j\} \qquad (6.12)$$

with the initial condition $C_N = \{(iN) \in A\}$. In other words, $C_j = C_{j+1}$ + the arcs incident into node j - arcs emanating from node j.

Another difficulty in evaluating the right-hand side of (6.8) is the determination of the df of the random variable W_{ij} for all $(ij) \in C_j$. But from (6.3), it is clear that the problem of determining the value of CAP(ij) reduces to the calculation of the df of the r.v.'s T_i^f and T_j^b and performing the necessary convolutions in equation (6.3).
The calculation of the df's of T_i^f and T_j^b is precisely the problem discussed in Dodin (1980). Therefore the techniques used in Dodin (1980) to approximate the df of the project completion time are now used

to approximate the df of W_{ij}. As it is discussed in Dodin (1980), the error in this approximation occurs due to three causes: (i) Discretizing the df of the activity duration if it is continuous; (ii) Convoluting two random variables, one with m realizations and the other with n realizations, that give rise to a third random variable which might have mn realizations. To avoid the uncontrolled increase in the number of realizations as computation progresses, we require the reduction of the realizations of the new random variable to a predetermined number of discrete points; (iii) Assuming the independence of the paths. It was concluded in Dodin (1980) that the approximation is accurate if all the activities have discrete df's or if the error in the discretization of the continuous distributions is very small. Most importantly, using the techniques of Dodin (1980) to evaluate (6.3) gives an approximation to the CAP(ij). Denote such approximation by ACAP(ij). Unfortunately, it cannot be asserted that ACAP(ij) underestimates CA(ij). However, empirical evidence given in section 4 indicates that the values of ACAP's form a very close estimate to the corresponding CAs obtained by extensive Monte Carlo sampling.

3. The Algorithm

The theory presented in the previous section is used to develop the algorithm described below, and it is illustrated by an example which directly follows the statement of the algorithm.

Algorithm

This algorithm is used to approximate the criticality indices of all the activities and events in PERT network. It starts at node N, then moves sequentially to nodes N-1, N-2,....,2. In each step it approximates the CAs of the arcs ending in the node under consideration, and the criticality index of the immediately preceding node. The algorithm proceeds as follows:

1 - For each node i=2,3,....,N-1 determine the df of the two random

variables T_i^f and T_i^b using the Approximating Procedure of Dodin (1980).

2 - Let j=N; determine the criticality indices of all the arcs ending in node j, i.e., the elements of C_N, as follows:

(a) Let n_j be the indegree of node j and suppoese the arcs ending in node j are arranged in nondecreasing order of their starting nodes as shown in Figure 6.13.

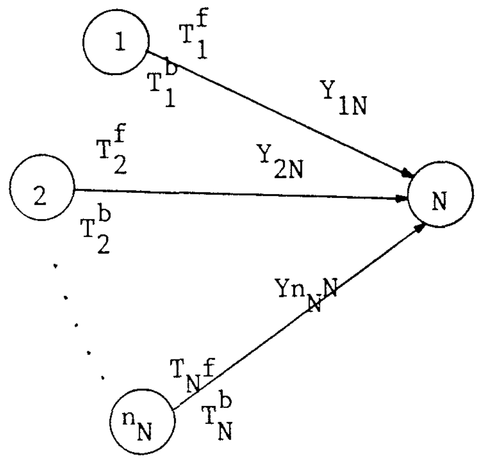


Figure 6.13. Order of the Arcs Ending in Node N.

Notice that if n_N=1, then we can let j=N-1 and we use Proposition 3 to obtain ACAP(N-1,N): so we can always assume that we start with a node j such that $n_j > 1$. If n_j=1 for all j=2,3,...,N, then all activities are critical and each criticality index is equal to 1; the network has only one path.

(b) Determine W_{ij} (of (6.3)) and V_{ij} (defined below) for all $(ij)\epsilon C_j$ using the formulas:

$$W_{ij}=T_i^f + Y_{ij} + T_j^b \quad \text{and} \quad V_{ij} = \max_{\substack{(lk)\epsilon C_j \\ (lk)\neq(ij)}} \{W_{lk}\} \cdot$$

V_{ij} is easily seen to denote the maximum duration of the paths not containing arc (ij). Consequently, CAP(ij) of (6.8) may be written as CAP(ij) = Pr $(W_{ij} \geqslant V_{ij})$. This rewritting of (6.8) is the form utilized in the following step (c).

(c) ACAP(ij)=Pr$(W_{ij} \geqslant V_{ij})$ for all $(ij)\epsilon C_j$. Assuming the independence of the two random variables W_{ij} and V_{ij}, then calculate

$$ACAP(ij) = \sum Pr(W_{ij}=t)\cdot Pr(V_{ij} \leqslant t),$$

where the summation is over all values t in the domain of W_{ij}.

(d) $ACN(N) = \sum_{(ij) \in C_N} ACAP(ij)$, and $ACN(N-1) = ACAP(N-1,N)$, where $ACN(i)$ is the approximate value of $CN(i)$.

3 - Iteratively, the process moves on to the next smaller numbered node, i.e. j is set to j-1. If $n_j=1$, then the CA of the arc ending in node j is approximated by ACN(j); go to 4. Otherwise, if $n_j>1$, then the CAs of the arcs ending in node j are approximated as follows:

(a) Determine the set C_j using (6.12).

(b) Determine W_{ij} for each arc ending in node j, i.e. for all (ij) in the subset $\{(ij)\in A:i<j\}$. The r.v.'s W_{lk}'s of the remaining arcs in C_j are known from the previous steps.

(c) Determine V_{ij} for the arcs ending in node j, then for each (ij) of these arcs $ACAP(ij)=Pr(W_{ij} \geqslant V_{ij})$.

(d) $ACN(j-1) = \sum_{\{(j-1,k) \in A :k > j-1\}} ACAP(j-1,k)$.

4 - If j=2, stop; otherwise go to 3.

<u>Example 4</u>.

The above algorithm is applied to the project represented by the PERT network of Figure 6.14. The realization of each activity indicated on the arcs of Figure 6.14 are assumed equally likely. Using the approximating procedure of Dodin (1980) we obtain the df's of the r.v.'s T_2^f, T_2^b, T_3^f and T_3^b; these df's are given in Table 6.33.

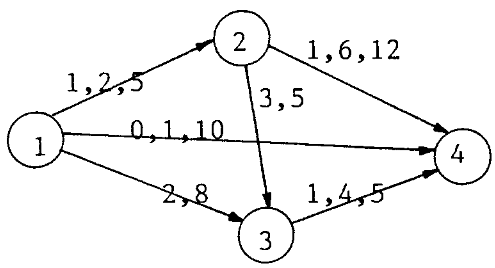


Figure 6.14. A PERT Network.

Table 6.33: The df's of T_2^f, T_2^b, T_3^f and T_3^b

T_2^f	t	1	2	5				
	p(t)x3	1	1	1				
T_2^b	t	4	6	7	8	9	10	12
	p(t)x18	1	3	2	2	2	2	6
T_3^f	t	4	5	6	7	8	10	
	p(t)x12	1	1	1	1	6	2	
T_3^b	t	1	4	5				
	p(t)x3	1	1	1				

Starting at node 4 we recognize that its indegree $n_4 = 3$; we calculate the df's of the r.v.'s. W_{14}, W_{24}, W_{34} and df's of the corresponding r.v.'s V_{14}, V_{24}, and V_{34}, which are given in Table 6.34. From rule 2.c of the Algorithm,

ACAP (1,4) = 0.0772, ACAP(2,4) = 0.3940, ACAP(3,4) = 0.5844,

and from rule 2.d we obtain

ACN(4) = 1.0556, ACN(3) = 0.5844.

The process moves to node j=3 where the CAs of arcs (1,3) and (2,3) are to be approximated. Step 3.a gives

$$C_3 = \{(1,4), (2,4), (3,4)\} + \{(1,3), (2,3)\} - \{3,4\}$$
$$= \{(1,4), (2,4), (1,3), (2,3)\},$$

Table 6.34: The df's of the r.v.'s W_{i4} and V_{i4} for i=1,2, and 3

W_{14}	w	0	1	10									
	p(w)x3	1	1	1									
V_{14}	v	5	6	7	8	9	10	11	12	13	14	15	17
	p(v)x324	2	4	6	13	40	10	39	42	68	48	16	36
W_{24}	w	2	3	6	7	8	11	13	14	17			
	p(w)x9	1	1	1	1	1	1	1	1	1			
V_{24}	v	5	6	7	8	9	10	11	12	13	14	15	
	p(v)x108	2	2	2	4	16	19	12	21	18	6	6	
W_{34}	w	5	6	7	8	9	10	11	12	13	14	15	
	p(w)x36	1	1	1	2	8	2	4	7	6	2	2	
V_{34}	v	2	3	6	7	8	10	11	13	14	17		
	p(v)x27	2	2	2	2	2	5	3	3	3	3		

and from rules 3.b and 3.c we calculate the df of the r.v.'s W_{13}, W_{23} and V_{13} which are shown in Table 6.35.

Table 6.35: The df's of the r.v.'s W_{13}, W_{23}, and V_{13}

W_{13}	w	3	6	7	9	12	13						
	p(w)x6	1	1	1	1	1	1						
W_{23}	w	5	6	7	8	9	10	11	12	13	14	15	
	p(w)x18	1	1	1	2	3	2	3	2	1	1	1	
V_{13}	v	5	6	7	8	9	10	11	12	13	14	15	17
	p(v)x486	4	8	12	26	30	70	84	36	66	72	24	54

Therefore

$$ACAP(1,3)=Pr(W_{13} \geqslant V_{13})=0.2476, \text{ and}$$

ACAP(2,3)=ACN(3) - ACAP(1,3) = 0.3368.

Then from rule 3.d ACN(2) = 0.3368+0.3940 = 0.7308. Finally, j is set equal to 2; thus, ACAP (1,2) = ACN (2) = 0.7308.

Figure 6.15 summarizes the estimates of the criticality indices of all the activities obtained by the algorithm. If they are compared with the corresponding "exact" CAs obtained by complete enumeration, which are given in Figure 6.16, it is seen that the estimates bound the exact CAs from below, and the maximum absolute value of the difference between the exact CA and the corresponding approximate CA is less than 0.04. Furthermore, if the activities in the PERT network are ranked in decreasing order of their CAs, both methods (estimation and enumeration) give the same ranking (as shown in Figures 6.15 and 6.16).

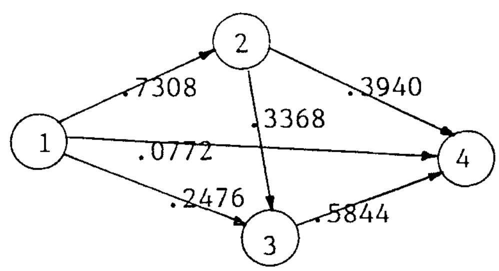


Figure 6.15. The Estimates of the CAs.

Activity	Rank of Criticality
(1,2)	1
(3,4)	2
(2,4)	3
(2,3)	4
(1,3)	5
(1,4)	6

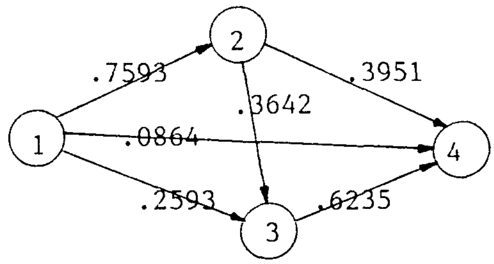


Figure 6.16. The Exact Values of the CAs.

Activity	Rank of Criticality
(1,2)	1
(3,4)	2
(2,4)	3
(2,3)	4
(1,3)	5
(1,4)	6

From the definition of the CN of node N we have:

$$CN(N)=\sum_{i=1}^{N-1} CA(iN)= \sum_{T_1 \epsilon P} CP(T_1) = 1.1050.$$

The parameter CP has a probability interpretation while CN does not have such an interpretation. In fact, the above Example CN(4) = 1.105. Also, the CAs of the arcs do not have a probability interpretation. We think of CA(ij) for an (ij) ϵA as a weight which relates the activity (ij) to the other activities. To help the interpretation of CA and CN as probability measures (between 0 and 1) we resort to "normalization". For example, the CAs of the activities in Figures 6.15 and 6.16, when "normalized," give the values shown in Figures 6.17 and 6.18 respectively. From Figure 6.18 we say that activity (2,4) is critical 35.76% of the time. The algorithm can be modified to give the normalized estimates of CAs.

This modification is limited to adding one more calculation to steps 2c and 3c to obtain the normalized ACAP(ij) for all (ij) ∊ A. The normalized values of ACNs can be obtained from steps 2d and 3d after replacing ACAP(ij) by the corresponding normalized values.

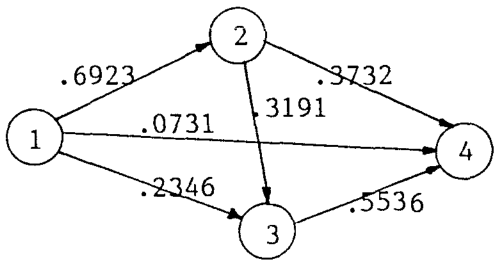


Figure 6.17. The Normalized Values of Figure 6.15.

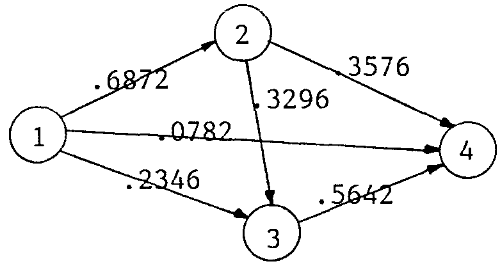


Figure 6.18. The Normalized Values of Figure 6.16.

4. Computational Experience

Before discussing the computational experience with this algorithm Dodin & Elmaghraby (1985) introduce the concept of criticality list which will be used as a measure of performance. In DANs an activity is either critical, which is the case if its float time is equal to zero, or noncritical with float time greater than zero. However, a noncritical activity with float time equal to s units is more critical than another noncritical activity with float time equal to ks units, where k>1. The highest ranking noncritical activity should receive more attention from management than the others, since any slippage in the critical path might cause such an activity to become critical. The list obtained by ranking the activities in increasing order of their float time, breaking ties in favor of the activity with the smaller starting node, is called the criticality list (CL). In case of PANs the CL is obtained by ranking the activities in decreasing order of the CAs and breaking ties in favor of the activity with the smaller starting node.

The concept of CL has not been used in project management and control even though, for a given activity, its position on the CL gives a measure of its importance. The list also shows the top M critical activites, i.e., it shows the top five, ten or 10% of the critical activities. This identification enhances the task of meting out the attention given to individual activities.
Dodin & Elmaghraby (1985) use the CL to measure the accuracy of the approximate ACAPs obtained by the algorithm. The CL generated by the algorithm is compared with the exact CL obtained by extensive Monte Carlo Sampling (MCS), since it is not possible to obtain exact CAs analytically.

They propose two measures of "goodness" of their procedure because the literature is void of any measure of performance or any computational experience with which to compare (mainly because the concept of CL as introduced here is new). The measures are: Compare the top M activities in both lists to see how many of them match, and calculate the correlation coefficient between the two lists.

The variations in the measures of performance depend on the sample size used in MCS, the df's of the activites, the accuracy of discretizing continuous distributions, and on the size and structure of the PERT network; see Dodin (1980).

The following paragraphs presents the impact of these factors on the proposed measures of performance. The section terminates with conclusions concerning the use of the algorithm.

Monte Carlo Sampling of the PERT network is performed using the method of Dodin (1980). It assigns a random value to each arc in the network generated from the original df of the activity, continuous or discrete.

Then the Critical Path method of DANs is used to identify the critical path(s) and activities. These two steps are repeated for a satisfactory number of times. In each experiment, frequencies of the critical activities are updated. The sample size should be "large enough" to guarantee that the CL obtained by MCS is very close to the exact CL. For example, Table 6.36 shows that for a G(10,15), generated at random, using the random activity network generated presented in Dodin (1980), with each activity having the same discrete distribution, the sampled CL stabilizes as the sample size increases, the table shows that the sampled CL converges to the CL obtained by the algorithm. The same conclusions can be derived from the correlation coefficient, denoted by r, which appears in the last row of Table 6.36, between the approximate CL and each of the samples CLs. In fact, r indicates for this PERT network that for a large sample size the two CLs are almost identical.

Table 6.37 shows for the same PERT network, the approximate CAs generated by both procedure, the algorithm and MCS. These approximate values appear in the table under the headings ACAP and CAS respecitvly. The table also shows the corresponding normalized values and CLs. The column head by D is the difference between the normalized CAs.

Tables 6.38 through 6.41 show the impact of size and density of the PERT network on the measures of performance. In Tables 6.38 and 6.39 we list the number of matching top five and top ten activities in the two criticality lists for all the problems tested. In Table 6.38 the same discrete distribution is used for each activity in the randomly generated network, while in Table 6.39 the same uniform distribution is used for each activity. Tables 6.40 and 6.41 show the correlation coefficients between the two criticality lists for all the problems considered in Tables 6.38 and 6.39, respectively. The entries in Tables 6.38 and 6.39 and the correlation coefficient in Tables 6.40 and 6.41

indicate the closeness of two criticality lists. The tables also indicate that the accuracy of the algorithm may not be affected by the size or density of the network. The four tables do not indicate any significant difference between using either discrete or continuous df, even though it was expected to have the numbers in the Tables 6.38 and 6.40 to be higher since the networks in Tables 6.38 and 6.40 are free of discretizing errors. The lack of significant difference between the corresponding entries in the Tables 6.38 and 6.39 on the one hand and Tables 6.40 and 6.41 on the other hand is due to the accurate discretization of the uniform distribution.

Table 6.42 shows further the impact of the density of PERT networks on the measures of performance. In all the six problems considered in Table 6.42 the same discrete distribution was assigned to each activity. The table lists the number of matching top 5 and 10 activities in the corresponding criticality lists. The last column of the table shows the correlation coefficient between the corresponding CLs. Again Table 6.42 shows that the algorithm performs equally well for all densities.

The CPU time of the approximating procedure (excluding the MCS) is minimal. For any of the 38 problems tested, it is always less than 30 seconds on AMDAHL V-70. The CPU time requirement for MCS depend on the size of the network, sample size and the df of the arc lengths. For large networks the sample size has to be large enough to have confidence in the sampled CL; in this case the CPU requirements are high. For example, in a network G(50,150) generated at random, where each activity is uniformly distributed between 0 and 10, the CPU time was twelve minutes on AMDAHL V-7 for a sample of size 10,000. The CPU time requirements for sampling the same PERT network to guarantee a 0.95 confidence in the sampled CL is over 30 minutes, since a sample of minimum size equal to 25,000 experiments is required for such accuracy.

Table 6.36: The CL's Generated by the Algorithm and MCS.

Arc#	CL of Algorithm 1 [CL(A)]	CL obtained by MCS CL(S)			
		s.s*=700	s.s.= 1500	s.s.= 3000	s.s.= 5000
1	6	6	6	6	6
2	12	10	12	12	12
3	10	10	10	10	10
4	3	3	3	3	3
5	15	10	12	12	12
6	6	6	6	6	6
7	6	6	6	6	6
8	12	10	12	12	12
9	11	10	10	10	10
10	12	10	12	12	12
11	4	4	4	4	4
12	9	9	9	9	9
13	4	4	4	4	4
14	2	1	2	2	2
15	1	1	1	1	1
r between CL[A] and CL[S]		0.960	0.985	0.985	0.985

*s.s. is the sample size in MCS

Table 6.37: Sample Output

The criticality index of the 15 activities in a network with 10 nodes.

Arc j	ACAP(j)*	CAS(j)*	Normalized ACAP(j)*	Normalized CAS(j)*	D*	CL(A)	CL(S)
1	0.543	0.576	0.543	0.493	-0.050	6	6
2	0.000	0.000	0.000	0.000	-0.000	12	12
3	0.002	0.001	0.002	0.001	-0.001	10	10
4	0.604	0.591	0.604	0.506	-0.098	3	3
5	0.000	0.000	0.000	0.000	-0.000	15	12
6	0.543	0.576	0.543	0.493	-0.050	6	6
7	0.543	0.576	0.543	0.493	-0.050	6	6
8	0.000	0.000	0.000	0.000	-0.000	12	12
9	0.002	0.001	0.002	0.001	-0.001	11	10
10	0.000	0.000	0.000	0.000	-0.000	12	12
11	0.543	0.581	0.543	0.498	-0.045	4	4
12	0.061	0.010	0.061	0.009	-0.052	9	9
13	0.543	0.580	0.543	0.497	-0.045	4	4
14	0.999	1.166	0.999	0.999	0.000	2	2
15	1.000	1.167	1.000	1.000	0.000	1	1

* The values are truncated for the third decimal point.

Table 6.38: The Matching of the Top 5 and 10 Activities in the CLs When All Activities Have Discrete Distribution.

Nodes	10		20		30		40	
Measure / Density	Top 5	Top 10	Top 5	Top 10	Top 5	Top 10	Top 5	Top 10
1.5	5	10	5	10	5	10	4	10
2.0	4	10	5	10	3	10	4	10
2.5	4	9	4	10	5	8	5	10
3.0	5	10	4	9	5	9	4	10

Table 6.39 : The Matching of Top 5 and 10 Activities in the CLs When All Activities Have Uniform Distribution.

Nodes	10		20		30		40	
Measure / Density	Top 5	Top 10	Top 5	Top 10	Top 5	Top 10	Top 5	Top 10
1.5	3	10	5	9	5	9	5	10
2.0	5	10	4	9	3	9	4	8
2.5	5	9	5	10	5	10	4	10
3.0	5	10	4	10	4	8	4	10

Table 6.40: The Correlation Between the Two Criticality Lists When All Activities Have Discrete Distributions.

Density \ Nodes	10	20	30	40
1.5	0.985	0.960	0.861	0.994
2.0	0.970	0.851	0.851	0.960
2.5	0.950	0.904	0.886	0.959
3.0	0.978	0.923	0.955	0.914

Table 6.41: The Correlation Between the Two Criticality Lists When All Activities Have Uniform Distributions.

Density \ Nodes	10	20	30	40
1.5	0.939	0.736	0.958	0.986
2.0	0.986	0.880	0.792	0.836
2.5	0.989	0.929	0.946	0.931
3.0	0.988	0.874	0.831	0.923

Table 6.42: The Impact of the PERT Network Density on the Measure of Performance.

Density	No. of nodes = 15		
	Top 5	Top 10	Correlation Coefficient
2	5	10	0.941
3	4	10	0.934
4	5	10	0.946
5	4	9	0.918
6	4	10	0.925
7	5	9	0.892

5. Conclusions

The objective of the authors of this paper was to offer the manager the probabilisitc equivalents of the commonly known concepts in DANs. They developed the approximate criticality index of an activity, ACAP, which; (i) measures the "degree of criticality" of an individual activity, and (ii) can be used to generate the criticality list (CL) of the "M most critical activities". ACAP is determined at a small fraction of the computing effort of Monte Carlo Sampling (MCS), which is the only other practical approach to the determination of the criticality of an activity.

Extensive computing experience reveals several interesting and significant results, which is summarized as follows:

The most critical activities generally appear at the beginning and end of a project. This is a natural consequence of the definition of the CA given in (6.2), its surrogate CAP of (6.8), and its approximation

ACAP. It is also a reasonable conclusion, considering the logical structure of the network. If used in a dynamic fashion it gives the most critical activities conditional upon the completion of certain segments of the project.

Most interestingly, the CL obtained by MCS converges to the approximate CL generated by Dodin and Elmaghraby's procedure as the sample size increases. This is an unexpected result. It certainly assures them of the "stability" of their CL relative to the MCS - generated list.

Finally, the error in approximation is drastically reduced when the discretization of the activity df is made finer. This is in consonance with the results obtained in the estimation of the df of the project completion time presented in Dodin (1980). Unfortunately, we could not obtain a bound on the error committed because it is dependent on several factors, some of which are not even measurable (such as the structure of the network).

6. Proofs of Propositions

The following complete proofs of the four theoretical results given in the text are taken from (Dodin and Elmaghraby, 1985).

Proof of Proposition 1

By definition

$$\begin{aligned} W_{ij} &= T_i^f + Y_{ij} + T_j^b \\ &= \max_{T_{1i}} \{Z(T_{1i})\} + Y_{ij} + \max_{T_{jN}} \{Z(T_{jN})\} \\ &= \max_{T_{1i}} \{ \max_{T_{jN}} \{Z(T_{1i}) + Y_{ij} + Z(T_{jN})\}\} \\ &= \max_{\substack{T_1 \\ ((ij)T_1)}} \{Z(T_1)\}. \end{aligned}$$

Proof of Proposition 2

We first argue heuristically that CAP(ij) $\leqslant$ CA(ij), then present a more formal proof of the proposition. If any path $T_1 \in L(ij)$ is longer than any path $T_q \in \bar{L}(ij)$, then, max $\{Z(T_k);\ T_k \in L(ij)\}$ is longer than max $\{Z(T_q):\ T_q \in \bar{L}(ij)\}$. Therefore, the set on which CA(ij) is defined contains the set on which CAP(ij) is defined. Moreover, the sum of probabilities defining CA(ij) is no less than the probability of the union of the events in the set, which is no smaller than that defining the probability of CAP(ij). Consequently, CAP(ij) $\leqslant$ CA(ij).

Now, as it is stated in (6.6) and (6.8) respectively,

$$CA(ij) = \sum_{T_1 \in L(ij)} Pr\ [Z(T_1) \geqslant Z(T_q) \text{ for all } T_q \in P,\ T_q \neq T_1] \text{ and}$$

$$CAP(ij) = Pr\ [W_{ij} \geqslant W_{lk} \text{ for all } (lk) \in C_j]\ ,$$

which, using Proposition 1, may be written as

$$CAP(ij) = Pr\ [\max_{T_1 \in L(ij)} \{Z(T_1)\} \geqslant \max_{T_q \in \bar{L}(ij)} \{Z(T_q)\}].$$

We prove that (6.9) holds for the case where L(ij) has three paths and $\bar{L}$(ij) may have any finite number of paths. All other cases can be proved similarly. Suppose L(ij) = $\{T_1, T_m, T_n\}$, then

$$CA(ij) = Pr[Z(T_1) \geqslant Z(T_q) \text{ for all } T_q \in P,\ T_q \neq T_1] + Pr[Z(T_m) \geqslant Z(T_q) \text{ for all } T_q \in P,\ T_q \neq T_m] + Pr[Z(T_n) \geqslant Z(T_q) \text{ for all } T_q \in P,\ T_q \neq T_n], \text{ and} \quad (A1)$$

$$CAP(ij) = Pr\ [\max\{Z(T_1),\ Z(T_m), Z(T_n)\} \geqslant Z(T_q) \text{ for all } T_q \in P].$$

Partitioning the event of the maximum leads to:

$$\begin{aligned} CAP(ij) = &Pr[Z(T_1) \geqslant Z(T_q) \text{ for all } T_q \in \bar{L}(ij),\ Z(T_1) \geqslant Z(T_m),\ Z(T_1) \geqslant Z(T_n)] \\ &+ Pr[Z(T_m) \geqslant Z(T_q) \text{ for all } T_q \in \bar{L}(ij),\ Z(T_m) > Z(T_1),\ Z(T_m) \geqslant Z(T_n)] \\ &+ Pr[Z(T_n) \geqslant Z(T_q) \text{ for all } T_q \in \bar{L}(ij),\ Z(T_n) > Z(T_1),\ Z(T_n) > Z(T_m)]. \quad (A2) \end{aligned}$$

We consider (A1) and (A2) term by term in evaluating the difference CA(ij) - CAP(ij). It is obvious that the first term of (A1) is identical to the first term of (A2); therefore the two will cancel. Now,

$$\Pr[Z(T_m)\geqslant Z(T_q) \text{ for all } T_q\in P]-\Pr[Z(T_m)\geqslant Z(T_q) \text{ for all } T_q\in\bar{L}(ij), Z(T_m)>Z(T_1), Z(T_m)\geqslant Z(T_n)]$$

$$= \Pr[Z(T_m)\geqslant Z(T_q) \text{ for all } T_q\in\bar{L}(ij),\ Z(T_m)=Z(T_1)].$$

Finally

$$\Pr[Z(T_n)\geqslant Z(T_q) \text{ for all } T_q\in P]$$

$$- \Pr[Z(T_n)\geqslant Z(T_q) \text{ for all } T_q\in\bar{L}(ij),\ Z(T_n)>Z(T_1), Z(T_n)>Z(T_m)]$$

$$= \Pr[Z(T_n)\geqslant Z(T_q) \text{ for all } T_q\in\bar{L}(ij),\ Z(T_n)>Z(T_1), Z(T_n)=Z(T_m)]$$

$$+\Pr[Z(T_n)\geqslant Z(T_q) \text{ for all } T_q\in\bar{L}(ij),\ Z(T_n)=Z(T_1), Z(T_n)>Z(T_m)]$$

$$+\Pr[Z(T_n)\geqslant Z(T_q) \text{ for all } T_q\in\bar{L}(ij),\ Z(T_n)=Z(T_1), Z(T_n)=Z(T_m)].$$

Collecting terms, we have

$$\begin{aligned} CA(ij) - CAP(ij) = {} & \Pr\ [Z(T_m)\geqslant Z(T_q) \text{ for all } T_q\in\bar{L}(ij),\ Z(T_m)=Z(T_1)] \\ & +\Pr\ [Z(T_n)\geqslant Z(T_q) \text{ for all } T_q\in\bar{L}(ij),\ Z(T_n)=Z(T_1), Z(T_n)>Z(T_m)] \\ & +\Pr\ [Z(T_n)\geqslant Z(T_q) \text{ for all } T_q\in\bar{L}(ij),\ Z(T_n)>Z(T_1), Z(T_n)=Z(T_m)] \\ & +\Pr\ [Z(T_n)\geqslant Z(T_q) \text{ for all } T_q\in\bar{L}(ij),\ Z(T_n)=Z(T_1), Z(T_n)=Z(T_m)] \\ & \geqslant 0 \end{aligned}$$

since each term on the r.h.s is $\geqslant 0$.

Proof of Proposition 3

If we let $C_i^1 = \sum_{j\in B(i)} CA(ji)$ and $C_i^2 = \sum_{j\in A(i)} CA(ij)$, and denote the criticality index of node i by CN(i) which is defined by:

$$CN(i) = \sum_{T_1(i\in T_1)} CP(T_1),$$

then we want to show that $C_i^1 = C_i^2 = CN(i)$. Let

n_i = the indegree of node i

n'_i = the outdegree of node i, and

$L_i = \{T_1 \epsilon P : i \epsilon T_1\}$.

Partition the set L_i into n_i disjoint subsets such that the first subset consists of all the paths containing the first arc terminating in node i, the second subset consists of all paths containing the second arc terminating in node i and so forth, and the subset n_i consists of all paths containing the last arc terminating in node i, i.e., $L_i = \sum_{j\epsilon B(i)} L(ji)$. Therefore

$$C_i^1 = \sum_{j\epsilon B(i)} \left[\sum_{T_1 \epsilon L(ji)} CP(T_1) \right] = \sum_{T_1 \epsilon L_i} CP(T_1) = CN(i).$$

To prove that $C_i^2 = CN(i)$ we partition L_i into n'_i disjoint subset where the jth subset consists of all paths containing the jth arc emanating from node i,i.e. $L_i = \sum_{j\epsilon A(i)} L(ij)$. Clearly ,

$$C_i^2 = \sum_{j\epsilon A(i)} \left[\sum_{T_1 \epsilon L(ij)} CP(T_1) \right] = \sum_{T_1 \epsilon L_i} CP(T_1) = CN(i).$$

Thus, $C_i^1 = C_i^2 = CN(i)$.

Proof of Proposition 4.

We use induction from above, i.e., we show that the expression given in (6.10) is true for j=N-1 and assume that it holds for k+1, where $2 < k+1 < N-1$, then show that it holds for j=k.

Indeed, if j=N-1 then the jth directed cutset is given by

$$C_{N-1} = \{(ij)\epsilon A : i \neq N-1 \text{ or } N \text{ and } j=N-1 \text{ or } N\}$$
$$= \{(ij)\epsilon A : i<N-1 \text{ and } j=N\} \cup \{(ij)\epsilon A : i<N-1 \text{ and } j=N-1\}.$$

But $\{(ij)\epsilon A : i<N-1 \text{ and } j=N\} = C_N - \{N-1,N\}$ which does not overlap with $\{(ij)\epsilon A : i<N-1 \text{ and } j=N-1\}$. Therefore

$$C_{N-1} = C_N + \{(i,N-1)\epsilon A : i < N-1\} - \{(N-1, N)\}.$$

Assume that (6.12) holds for k+1 such that $2 < k+1 < N-1$; we want to show that it holds for j=k.

The kth directed cutset is given by

$$C_k = \{(ij) \in A:\ i<k \text{ and } j \geq k\}$$
$$= \sum_{j=1}^{N} \{(ij) \in A : i<k\}$$
$$= \{(ik) \in A : i<k\} + \sum_{j=k+1}^{N} \{(ij) \in A: i<k\}.$$

But $$\sum_{j=k+1}^{N} \{(ij) \in A : i<k\} = \sum_{j=k+1}^{N} [\{(ij) \in A : i < k+1\} - \{(kj)\}]$$
$$= C_{k+1} - \{(kj) \in A : j > k\}.$$

Thus, $$C_k = C_{k+1} + \{(ik) \in A : i<k\} - \{(kj) \in A : j>k\}.$$

****

PROPOSED PROCEDURE

The proposed procedure to be described is operationally identical to the proposed procedure of Chapter 4 for determining the pdf of the project completion time, hence resultant network of that procedure is used to approximate the criticality indices of activities.

Since in proposed procedure of Chapter 4 resultant network consists of only series-parallel activities as shown in Figure 6.19, it is very easy to estimate CA(ij) directly from (6.2).

$$CA(ij) = \sum_{T_1} CP(T_1) = \sum Pr\ [(Z(T_1) \geq Z(T_q) \text{ for all } T_q \in P,\ T_1 \neq T_q]. \quad (6.2)$$

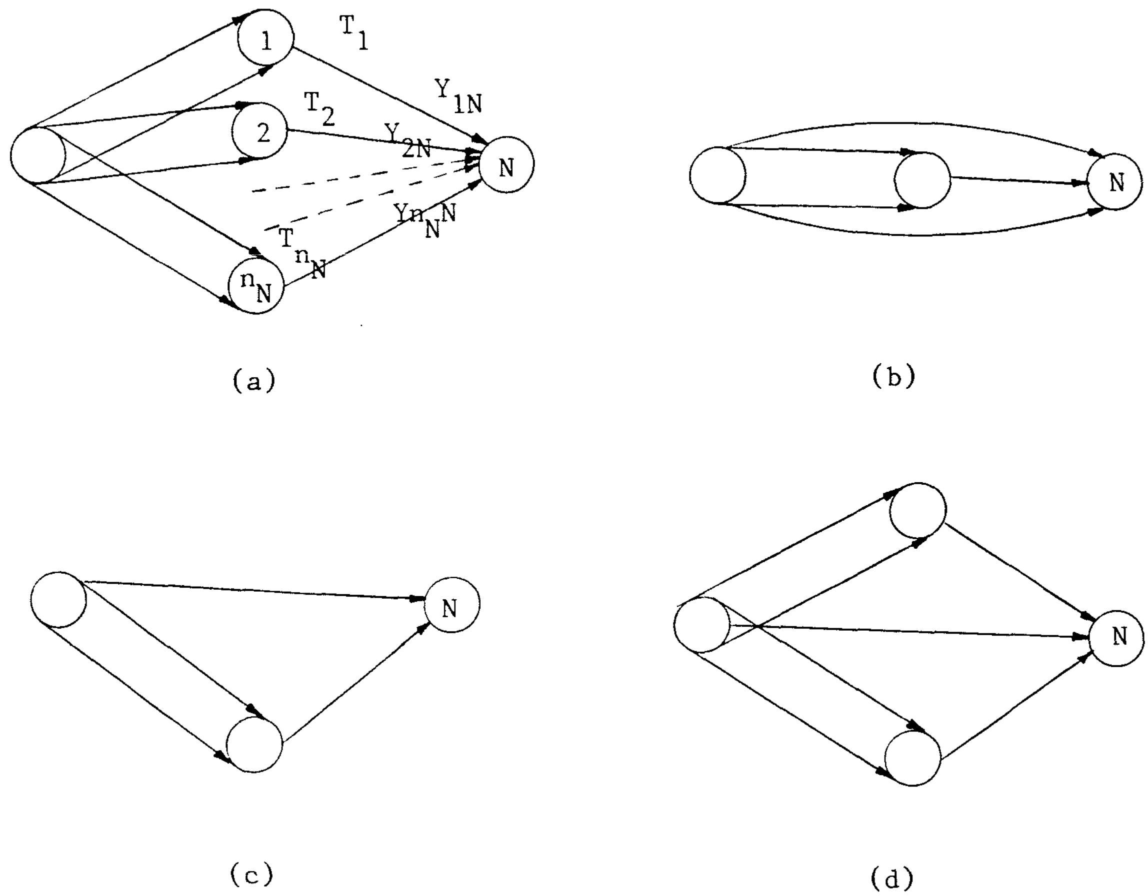


Figure 6.19

Same as Dodin and Elmaghraby's procedure the proposed procedure for determining the criticality indices of all the activities starts at node N of final networks, determines the criticality indices of the arcs ending in node N, then proceeds recursively to nodes N-1, N-2 and so on until it finally reachs node 2 where the criticality index of arc (1,2) is determined.

Next section presents the Algorithm.

The following is a list of the notations used in this algorithm:

B_j: set of arcs ending in node j.

CAP(ij): approximate criticality index of activity (ij).

CNP(j): approximate criticality index of node j.

CPP(T_h) : approximate criticality index of path T_h.

$I(j)$: indegree of node j.

For arc $(ij)\epsilon A$, let W_{ij} denote the duration of the path from the initial node to node j containing arc (ij).

Let W_{1j} denote durations of the paths ending in node j not containing arc (ij), such that $(1j)\epsilon B_j$ and $(1j)\neq(ij)$.

THE ALGORITHM

This algorithm is used to approximate the criticality indices of all the activities and events in PERT network.

It starts at node N, then moves sequentially to nodes N-1, N-2,....,2. In each stage it approximates the CAs of the arcs ending in the node under consideration, and the criticality index of the immediately preceding node. The algorithm proceeds as follows:

1 - For each node i=2,3,...,N-1 determine the df of the random variable T_i (realization time of node i) using the proposed procedure of Chapter 4.

2 - Let j=N. Determine the criticality inidices of all arcs ending in node j, i.e., the elements of B_N as follows:

   a - Let I(j) be the indegree of node j and suppose the arcs ending in node j are arranged in nondecreasing order of their starting nodes as shown in Figure 6.19 (Notice that if I(N)=1, then we can let j=N-1 and we use proposition 3 to obtain CAP(N-1,N); so we can always assume that $I(j) > 1$. If I(j)=1 for all j=2,3,..N then all activities are critical and each criticality index is equal to 1; the network has only one path).

   b - Determine W_{ij} for all $(ij)\epsilon B_j$ using the formula $W_{ij}=T_i+Y_{ij}$. (6.13)

   c - $CAP(ij)=Pr(W_{ij} \geqslant W_{1j})$ for all $(ij)\epsilon B_j$ (6.14)
   since random variables W_{ij} and W_{1j} are independent, then calculate

$$CAP(ij) = \sum_t Pr\,(W_{ij}=t) \cdot \prod_{\substack{l=1 \\ l\neq i}}^{n_j} Pr(W_{lj} \leqslant t), \qquad (6.15)$$

where the summation is over all values t in the domain of W_{ij}

d - $CNP(N) = \sum_{(ij)\in B_N} CAP(ij)$, and $CNP(N-1) = CAP(N-1, N)$ · (6.16)

3 - Iteratively, the process moves on the next smaller numbered node, i.e., j is set to j-1. If I (j)=1, then the CAP of the arc ending in node j is equal to CNP(j); go to 4. Otherwise, if I(j) >1, then the CAPs of the arcs ending in node j are determined as follows:

a - Determine the set B_j.

b - Determine W_{ij} for each arc ending in node j.

c - For each arc ending in node j,

$CAP(ij) = Pr\,(W_{ij} \geqslant W_{lj})$ for all $(ij)\in B_j$

d - $CNP(j-1) = \sum_{\{(j-1,k)\in A, k>j-1\}} CAP\,(j-1,k)$. (6.17)

4 - If j=2, stop; otherwise go to 3.

5 - Decondition the approximate criticality indices of activities.

6 - Determine the approximate criticality indices of activities and paths.

Example 5.

As an example let us apply the proposed algorithm to the PERT network of Figure 4.31 of Chapter 4.

Figures 4.33 through 4.39 show the reduction process of network configuration of Figure 4.31 using the proposed procedure of Chapter 4 by fixing on the first realization time of common activity 1.

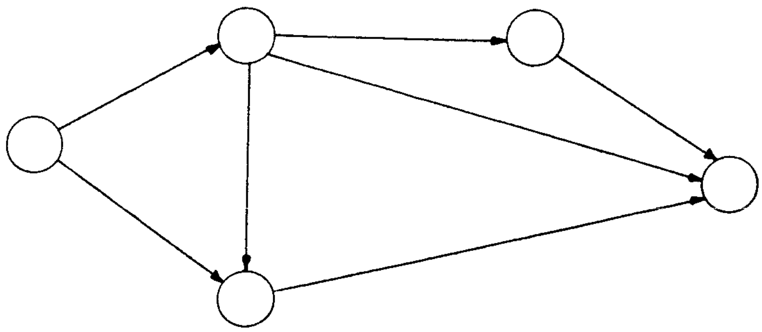


Figure 4.31

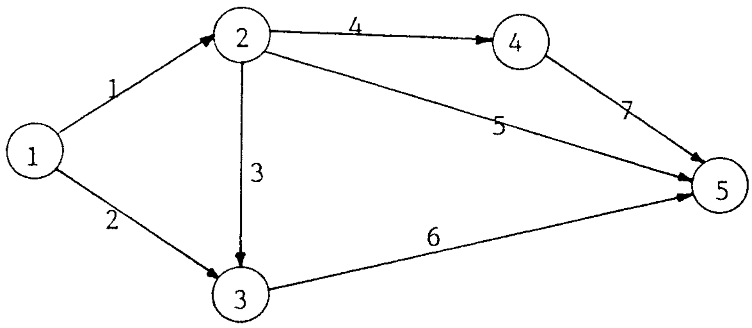


Figure 4.33

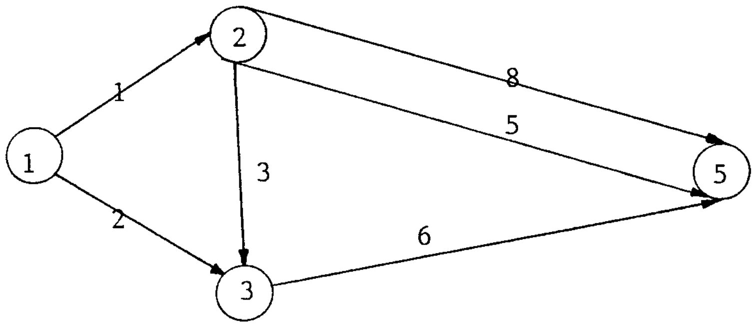


Figure 4.34

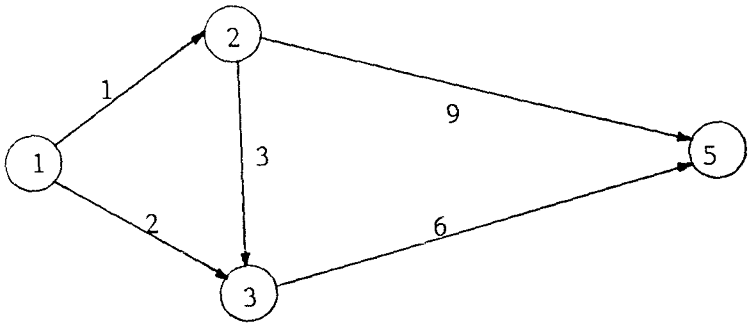


Figure 4.35

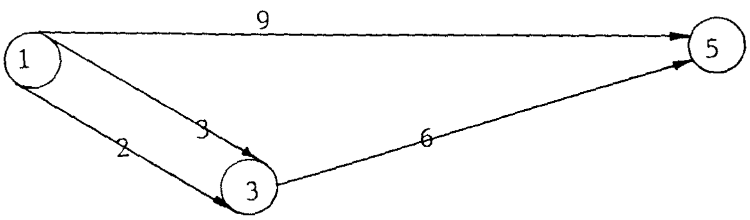


Figure 4.36

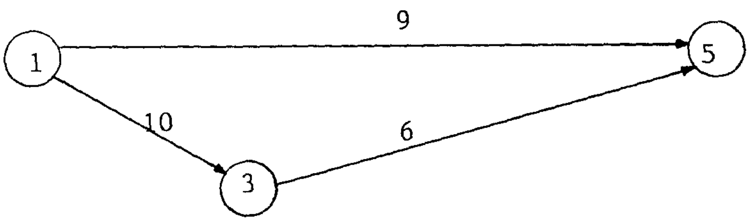


Figure 4.37

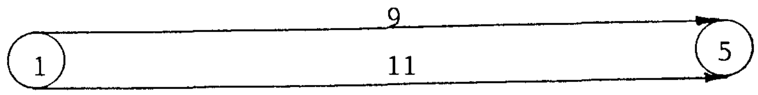


Figure 4.38

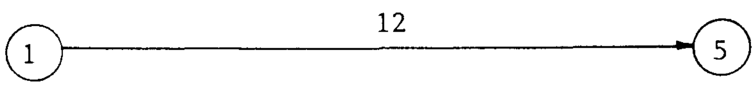


Figure 4.39

1 - The random variables, T_i (realization time of node i , i=2,3,4,5) are known from Chapter 4.

2 - Let j=5. Determine the approximate criticality indices of arcs 9 and 11 of Figure 4.38 as follows:

a - I(j=5)=2.

b - Determine W_{ij} for all (ij) ϵB_j using formula $W_{ij} = T_i + Y_{ij}$

$W_9 = T_1 + Y_9 = 0 + Y_9 = Y_9$ and $W_{11} = T_1 + Y_{11} = 0 + Y_{11} = Y_{11}$.

c - $CAP(9) = Pr\ (W_9 \geqslant W_{11})$ and $CAP\ (11) = Pr\ (W_{11} \geqslant W_9)$

then calculate

$CAP(9) = \sum_t Pr\ (W_9=t).Pr(W_{11} \leqslant t)$, and $CAP(11)= \sum_t P(W_{11} = t).Pr(W_9 \leqslant t)$.

In Figure 4.37 the approximate criticality index of arc 6 is equal to the approximate criticality index of arc 10 because these two arcs are series and also equals to the approximate criticality index of arc 11 since arc 11 is the resultant arc of these two arcs.

Therefore, CAP(6) = CAP(10) = CAP(11) ,

d - CNP(5) = CAP(9) + CAP(11) = CNP(1) , and CNP(3) = CAP(6) ·

In Figure 4.36 CAP(2) + CAP(3) = CNP(3) ,

in Figure 4.35 CAP(1) = CAP(3) + CAP(9) = CNP(2) ,

in Figure 4.34 CAP(5) + CAP(8) = CAP(9) ,

finally in Figure 4.33 CAP(4) = CAP(7) = CAP(8) = CNP(4) ·

The criticality indices obtained in above are the approximate conditional criticality indices of the activities and nodes given first realization time of common activity 1. Similarly the approximate conditional criticality indices of the arcs and nodes can be computed for all realization times of common activity and then by doconditioning these conditional criticality indices the approximate criticality indices can be determined. Following examples illustrate that the criticality indices obtained using proposed procedure is better than the criticality indices obtained using Dodin and Elmaghraby's (1985) procedure. Most importantly the criticality indices obtained using proposed procedure of this dissertaton are consistent while those obtained using proposed procedure of Dodin and Elmaghraby are not consistent. The inconsistency in approximate criticality indices obtained using Dodin and Elmaghraby's (1985) approach stems from the manner of choosing activities for determining the approximate criticality indices in any Directed Cutset. For example, in Page 233 the Directed Cutset of node 3 is determined to be C_3 = {(1,4), (2,4), (1,3), (2,3)}, since the approximate criticality indices of activities (1,4) and (2,4) previously determined as ACAP (1,4) = 0.0772 and ACAP (2,4) = 0.3940, in order to calculate the approximate values of criticality indices of activities (1,3) and (2,3) i.e., ACAP (1,3) and ACAP (2,3) we can choose one of the following alternatives.

1 - We can calculate W_{13}, W_{23}, and $V_{13} = \max \{W_{23}, W_{14}, W_{24}\}$, which are shown in Table 6.35. Therefore,

$$ACAP(1,3) = Pr\ (W_{13} \geqslant V_{13}) = 0.2476$$

and $$ACAP(2,3) = ACN(3)-ACAP(1,3) = 0.3368$$

as we have done in Pages 234 and 235

or,

2 - We can calculate W_{13}, W_{23} and $V_{23} = \max \{W_{13}, W_{14}, W_{24}\}$, which are shown in Table 6.43 and calculate ACAP (2,3) first and then ACAP (1,3) as follows:

Table 6.43: The df's of the r.v.'s W_{13}, W_{23}, and V_{23}

W_{13}	w	3	6	7	9	12	13					
	p(w)x6	1	1	1	1	1	1					
W_{23}	w	5	6	7	8	9	10	11	12	13	14	15
	p(w)x18	1	1	1	2	3	2	3	2	1	1	1
V_{23}	v	3	6	7	8	9	10	11	12	13	14	17
	p(v)x162	4	8	12	6	10	20	12	18	36	18	18

Therefore,

$$ACAP(2,3) = Pr\ (W_{23} \geqslant V_{23}) = 0.3944$$

and $$ACAP(1,3) = ACN\ (3) - ACAP(2,3) = 0.5844 - 0.3944 = 0.1900$$

Then from rule 3.d ACN(2) = 0.3944 + 0.3940 = 0.7884 . Finally j is set equal to 2; thus, ACAP(1,2) = ACN(2) = 0.7884.

Figure 6.20 summarizes the estimates of the criticality indices of all the activities obtained by the algorithm using second alternative. If they are compared with the corresponding "exact" CAs obtained by complete enumeration, which are given in Figure 6.16, it is seen that the estimates don't bound the exact CAs from below for two activities (1,2) and (2,3), and the maximum absolute value of the difference between the exact CA and the corresponding approximate CA is less than 0.07. Notice that in this case both methods (estimation and enumeration) don't give the same ranking of their CAs (as shown in Figures 6.16 and 6.20).

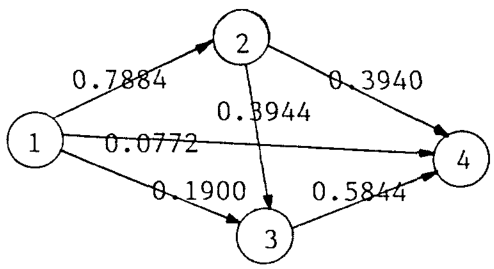


Figure 6.20. The Estimates of the CAs.
using second alternative.

Activity	Rank of Criticality
(1,2)	1
(3,4)	2
(2,3)	3
(2,4)	4
(1,3)	5
(1,4)	6

Figure 6.21 shows the normalized values of Figure 6.20.

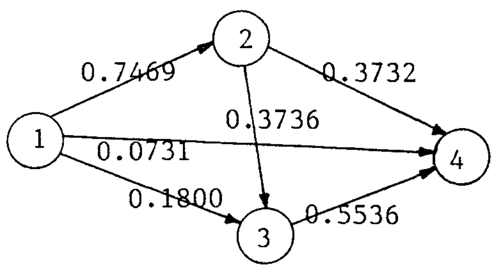


Figure 6.21. The Normalized Values of Figure 6.20.

Let X denote the exact normalized criticality indices (NCA) of Figure 6.18, Z_1 denote the normalized criticality indices (NACAP) of Figrure 6.17, and Z_2 denote the normalized criticality indices (NACAP) of Figure 6.21. In the following we calculate the correlation coefficient between X and Z_1 and also between X and Z_2.

Activity	(NCA) X	(NACAP) Z_1	$x=X-\bar{X}$	$z_1=Z_1=-\bar{Z}_1$	x^2	x_1^2	xz_1
(1,2)=A	0.6872	0.6923	0.3119667	0.3179834	0.0973232	0.1011134	0.0992002
(1,4)=B	0.0782	0.0731	-.2970333	-.3012166	0.0882287	0.0907314	0.0894713
(1,3)=C	0.2346	0.2346	-.1406333	-.1397166	0.0197777	0.0195207	0.0196488
(2,3)=D	0.3296	0.3191	-.0456333	-.0552166	0.0020823	0.0030488	0.0025197
(2,4)=E	0.3576	0.3732	-.0176333	-.0011166	0.0003109	0.0000012	0.0000196
(3,4)=F	0.5642	0.5536	0.1889667	0.1792834	0.0357084	0.0321425	0.0338785
	$\Sigma X=$ 2.2514 $\bar{X}=$ .3752333	$\Sigma Z_1=$ 2.2459 $\bar{Z}_1=$ .3743166			$\Sigma x^2=$ 0.2434312	$\Sigma z_1^2=$ 0.246558	$\Sigma xz_1=$ 0.2447381

Correlation Coefficient: $r_{X,Z_1} = \dfrac{\Sigma(xz_1)}{\sqrt{(\Sigma x^2)(\Sigma z_1^2)}} = \dfrac{(0.2447381)}{\sqrt{(0.2434312)(0.246558)}}$

$= 0.9990137.$

Activity	(NCA) X	(NACAP) Z_2	$x=X-\bar{X}$	$z_2=Z_2-\bar{Z}_2$	x^2	z_2^2	xz_2
(1,2)=A	0.6872	0.7469	0.3119667	0.3635	0.0973232	0.1321322	0.1133998
(1,4)=B	0.0782	0.0731	-.2970333	-.3103	0.0882287	0.096286	0.0921694
(1,3)=C	0.2346	0.1800	-.1406333	-.2034	0.0197777	0.0413715	0.0286048
(2,3)=D	0.3296	0.3736	-.0456333	-.0098	0.0020823	0.000096	0.0004472
(2,4)=E	0.3576	0.3732	-.0176333	-.0102	0.0003109	0.000104	0.0001798
(3,4)=F	0.5642	0.5536	0.1889667	0.1702	0.0357084	0.028968	0.0321621
	$\Sigma X=$ 2.2514 $\bar{X}=$ .3752333	$\Sigma Z_2=$ 2.3004 $\bar{Z}=$ 0.3834			$\Sigma x^2 =$ 0.2434312	$\Sigma z_2^2=$ 0.2989577	$\Sigma xz_2=$ 0.2669631

Correlation coefficient: $r_{X,Z_2} = \dfrac{\Sigma(xz_2)}{\sqrt{(\Sigma x^2)(\Sigma z_2^2)}} = \dfrac{(0.2669631)}{\sqrt{(0.2434312)(0.2989577)}}$

$= 0.989597.$

By applying the proposed procedure of this dissertation the approximate criticality indices of activities are calculated as shown in Figure 6.22. Figure 6.23 shows the normalized values of Figure 6.22.

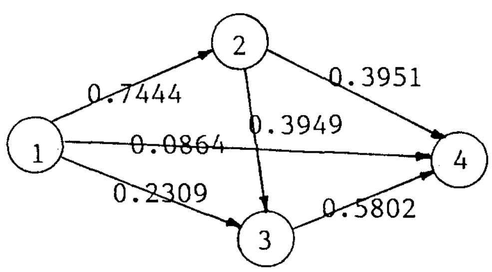


Figure 6.22. The Estimates of CAs using the proposed procedure.

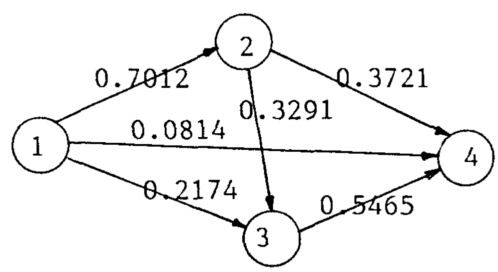


Figure 6.23. The Normalized Values of Figure 6.22.

Let Y denote the normalized criticality indices (NCAP) of Figure 6.23, in the following we calculate the correlation coefficient between X and Y.

If the criticality indices obtained using the proposed algorithm are compared with the corresponding "exact" CAs obtained by complete enumeration as shown in Figure 6.16 it is seen that the maximum absolute value of the difference between the exact CA and the corresponding approximate CA is less than 0.045, and both methods give the same ranking.

Activity	(NCA) X	(NCAP) Y	$x=X-\bar{X}$	$y=Y-\bar{Y}$	x^2	y^2	xy
(1,2)=A	0.6872	0.7012	0.3119667	0.3265834	0.0973232	0.1066567	0.1018831
(1,4)=B	0.0782	0.0814	-.2970333	-.2932166	0.0882287	0.0859759	0.087095
(1,3)=C	0.2346	0.2174	-.1406333	-.1572166	0.0197777	0.024717	0.0221098
(2,3)=D	0.3296	0.3291	-.0456333	-.0455166	0.0020823	0.0020717	0.002077
(2,4)=E	0.3576	0.3721	-.0176333	-.0025166	0.0003109	0.0000063	0.0000443
(3, 4)=F	0.5642	0.5465	0.1889667	0.1718834	0.0357084	0.0295439	0.0324802
	X= 2.2514 $\bar{X}$= .375233	Y= 2.2477 $\bar{Y}$= .3746166			Σx^2 = 0.2434312	Σy^2 = 0.2489715	Σxy= 0.2456894

$$\text{Correlation Coefficient:} r_{X,Y} = \frac{\Sigma(xy)}{\sqrt{(\Sigma x^2)(\Sigma y^2)}} = \frac{(0.2456894)}{\sqrt{(0.2434312)(0.2489715)}}$$

$$= 0.997984$$

Example 3.

Consider again the Wheatstone bridge of Figure 3.10. By appling the Approximating Procedure proposed by Dodin and Elmaghraby (1985), the criticality indices of activities are calculated as shown in Figures 6.24

through 6.27. Since, in order to approximate the criticality indices of activities leading into node 3 where $C_3 = \{(1,3), (2,3), (2,4)\}$ we have two choices (1 - we can choose to approximate the criticality index of activity (2,3) first and then (1,3) or 2 - we can choose to approximate the criticality index of activity (1,3) first then (2,3)) therefore, we have two series of answers. First group of the answers are shown beside the activites of Figure 6.24, and second group of the answers are shown beside the activities of Figure 6.25. Figures 6.26 and 6.27 show the normalized values of the criticality indices of Figures 6.24 and 6.25 respectively.

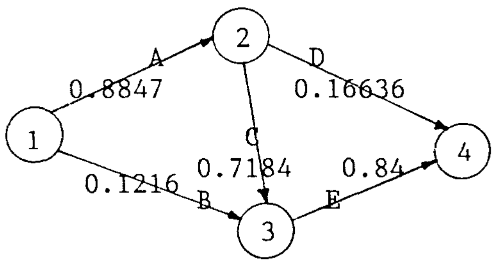


Figure 6.24

Activity	Rank of Criticality
(1,2)=A	1
(3,4)=E	2
(2,3)=C	3
(2,4)=D	4
(1,3)=B	5

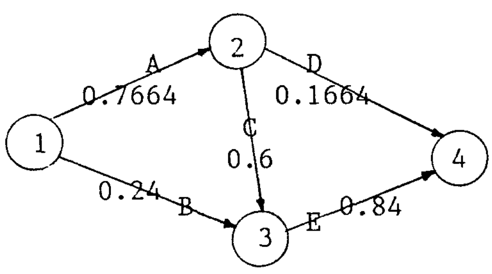


Figure 6.25

Activity	Rank of Criticality
(3,4)=E	1
(1,2)=A	2
(2,3)=C	3
(1,3)=B	4
(2,4)=D	5

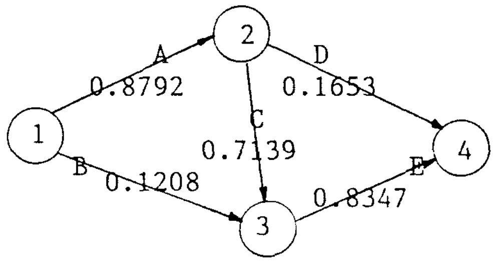


Figure 6.26

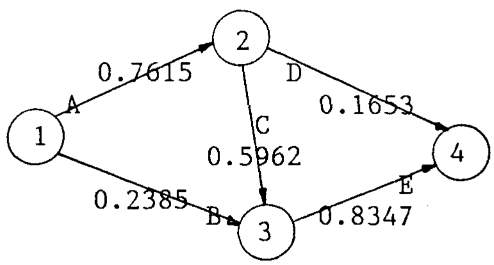


Figure 6.27

Figure 6.11 shows the normalized values of the criticality indices obtained using proposed procedure of this dissertation (NCAP), and Figure 6.10 shows the normalized values of the criticality indices obtained by complete enumeration (NCA) (i.e. the normalized values of the exact CAs). Comparing the criticality indices shown beside the activities of Figure 6.24 with the exact CAs of Figure 6.10, we see that criticality indices shown beside the activities of Figure 6.26 which are obtained by using

Approximating Procedure proposed by Dodin and Elmaghraby don't give the same ranking as exact CAs.

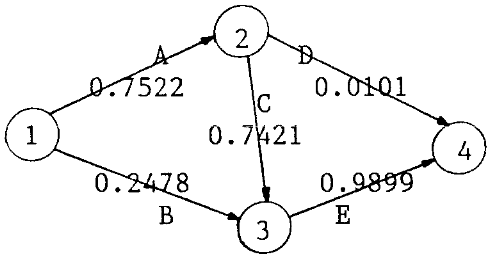


Figure 6.11

Activity	Rank of Criticality
(3,4)=E	1
(1,2)=A	2
(2,3)=C	3
(1,3)=B	4
(2,4)=D	5

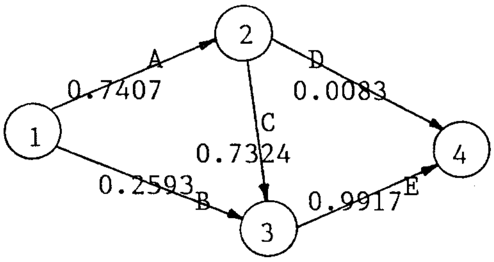


Figure 6.10

Activity	Rank of Criticality
(3,4)=E	1
(1,2)=A	2
(2,3)=C	3
(1,3)=B	4
(2,4)=D	5

Let X denote the criticality indeices of Figure 6.10, Z_1 denote the criticality indices of Figure 6.26 and Z_2 denote the criticality indices of Figure 6.27. In the following we calculate the correlation coefficient between X and Z_1 and also between X and Z_2.

Activity	(NCA) X	(NACAP) Z_1	$x=X-\bar{X}$	$z_1=Z_1-\bar{Z}_1$	x^2	z_1^2	xz_1
(3,4)=E	.9917	.8347	0.44522	0.29192	0.1982208	.0852172	.1299686
(2,4)=D	.0083	.1653	-.53818	-.37748	0.2896377	.1424911	.2031521
(2,3)=C	.7324	.7139	0.18592	0.17112	0.0345662	.029282	.0318146
(1,3)=B	.2593	.1208	-.28718	-.42198	0.0824723	.1780671	.1211842
(1,2)=A	.7407	.8792	0.19422	0.33642	0.0377214	.1131784	.0653394
	$\Sigma X=$ 2.7324 $\bar{X}=$ .54648	$\Sigma Z=$ 2.7139 $\bar{Z}_1=$ .54278			$\Sigma x^2=$ 0.6426184	$\Sigma z_1^2=$ .5482358	$\Sigma xz_1=$ .5514589

Correlation Coefficient: $r_{X,Z_1} = \dfrac{\Sigma(xz)}{\sqrt{(\Sigma x^2)(\Sigma x_1^2)}} = \dfrac{0.5514589}{\sqrt{(.6426184)(.5482358)}}$

$= 0.9290795$.

Activity	(NCA) X	(NACAP) Z_2	$x=X-\bar{X}$	$z_2=Z_2-\bar{Z}_2$	x^2	z_2^2	xz_2
(3,4)=E	.9917	.8347	0.44522	0.31546	0.1982208	0.099515	.1404491
(2,4)=D	.0083	.1653	-.53818	-.35394	0.2896377	0.1252735	.1904834
(2,3)=C	.7324	.5962	0.18592	0.07696	0.0345662	0.0059228	.0143084
(1,3)=B	.2593	.2385	-.28718	-.28074	0.0824723	0.0788149	.0806229
(1,2)=A	.7407	.7615	0.19422	0.24226	0.0377214	0.0586899	.0470517
	$\Sigma X=$ 2.7324 $\bar{X}=$ .54648	$\Sigma Z_2=$ 2.5962 $\bar{Z}_2=$ .51924			$\Sigma x^2=$ 0.6426184	$\Sigma z_2^2=$ 0.3682161	$\Sigma xz_2=$ 0.4729155

Correlation Coefficient: $r_{X,Z_2} = \frac{\Sigma(xz_2)}{\sqrt{(\Sigma x^2)(\Sigma z_2^2)}} = \frac{0.4729155}{\sqrt{(.6426184)(.368261)}}$

$$= 0.972141.$$

Comparing, r_{X,Z_1} and r_{X,Z_2} with $r_{X,Y}$ = .9997721 which is obtained previously by using the proposed procedure (see page 222), we see that the proposed procedure is more accurate than Dodin and Elmaghraby's procedure especially because the criticality indices shown in Figure 6.26 don't give the same ranking as exact CAs.

Example 6.

Consider the network configuration of Figure 5.4 Notice that this network is symmetric, the duration times of activities are shown beside each activity with equal probability for occurrence. This example may clarify the two following weak points of the Approximating Procedure proposed by Dodin and Elmagraby (1985).

1 - Since the network is symmetric the criticality indices (CIs) of similar activities are actually the same, while Dodin and Elmagraby's procedure doesn't give the same CAs for similar activities.

2 - Dodin and Elmaghraby's procedure is not very accurate if common activities have relatively large variances relative to the variances of the other activities.

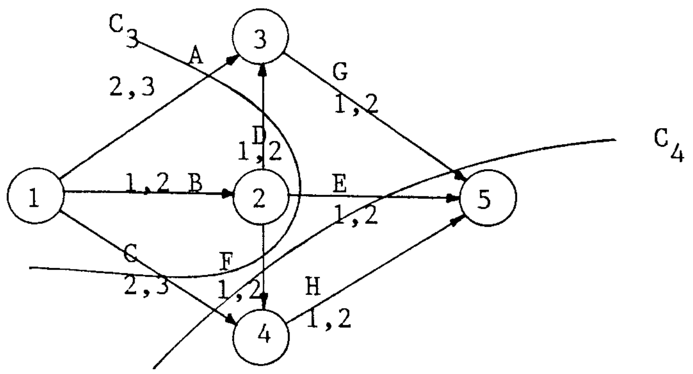


Figure 5.4

As mentioned previusly, Dodin and Elmaghraby's procedure gives more than one series of answers for the CIs of activities in irreducible networks.

The number of series of answers depends on the number of activities in the Directed Cutset of the merge events for which we need to approximate the criticality indices of activites from criticality indices of the other activities. For instance, in order to estimate the CIs of the activities of Figure 5.4 using Dodin and Elmaghraby's procedure first step is approximating the CIs of G,E, and H.Having done that second step is determining the Directed Cutset of node 4 which is C_4 = {(3,5), (2,5), (2,4), (1,4)}. Since we know the CIs of activities (3,5) and (2,5) from previous step, therefore, we have two unknowns, including the CI of activity (1,4) and the CI of activity (2,4). Thus, we can choose to approximate the CI of activity (1,4) first and then the CI of activity (2,4) or vice versa we can choose to approximate the CI of activity (2,4) first then the CI of activity (1,4). Next step is approximating the criticality indices of activities leading into node 3, where C_3={(1,3), (2,3), (2,5), (2,4), (1,4)}, notice that although nodes 3 and 4 are structurally symmetric but these two sets (C_3 and C_4) are not equal i.e., they don't have similar elements and this is the reason that Dodin and Elmagraby's procedure gives different CIs for similar activities. Since we know the CIs of activities (2,5), (2,4) and (1,4) in the set C_3 from previous steps we have two unknowns, including the CIs of activities (1,3) and (2,3). Thus, we can choose to approximate the CI of activity (1,3) first and then the CI of activity (2,3), or we can choose to approximate the CI of activity (2,3) first and then the CI of activity (1,3). We had two choices for estimating the CIs of activities (1,4) and (2,4) and now we have two choices for estimating the CIs of activities (1,3) and (2,3) therefore we have 2x2=4 different series of answers for approximating the CIs of activities.

Table 6.44 shows the exact normalized criticality indices of activities obtained using complete enumeration, four different series of criticality indices obtained using Dodin and Elmaghraby's procedure and also shows the Correlation Coefficient between exact and estimate criticality indices for each series of answers, and the rank (R) of Criticality.

Table 6.44:

Activity	(NCA) X	R	(NACAP) Z_1	R	(NACAP) Z_2	R	(NACAP) Z_3	R	(NACAP) Z_4	R
(1,2)=B	0.6588486	1	0.7262668	1	0.5797278	1	0.8480393	1	0.7015003	1
(3,5)=G	0.4946695	2	0.4822051	2	0.4822051	2	0.4822051	2	0.4822051	2
(4,5)=H	0.4946695	3	0.4822051	3	0.4822051	3	0.4822051	3	0.4822051	3
(2,3)=D	0.3240938	4	0.2779135	5	0.2779135	4	0.3996859	5	0.3996859	4
(2,4)=F	0.3240938	5	0.4127636	4	0.2662247	5	0.4127636	4	0.2662247	5
(1,3)=A	0.1705756	6	0.2042916	6	0.2042916	7	0.0825191	6	0.0825191	7
(1,4)=C	0.1705756	7	0.0694415	7	0.2159804	6	0.0694415	7	0.2159804	6
(2,5)=E	0.0106609	8	0.0355896	8	0.0355896	8	0.0355896	8	0.0355896	8
Correlaton Coefficient between X and Z_i, i=1,4			r_{X,Z_1} = .96949		r_{X,Z_2} = .98554		r_{X,Z_3}=.96123		r_{X,Z_4}=.97214	

Notice that in none of the cases the Rank of Criticality is same as the Rank of Exact NCAs. Now in Table 6.45 we compare the exact normalized CAs with the normalized criticality indices obtained using the proposed procedure of this dissertation.

Table 6.45:

Activity	(NCA) X	R	(NCAP) Y	R
(1,2)=B	0.6588486	1	0.6536187	1
(3,5)=G	0.4946695	2	0.4930477	2
(4,5)=H	0.4946695	3	0.4930477	3
(2,3)=D	0.3240938	4	0.3198572	4
(2,4)=F	0.3240938	5	0.3198572	5
(1,3)=A	0.1705756	6	0.1731905	6
(1,4)=C	0.1705756	7	0.1731905	7
(2,5)=E	0.0106609	8	0.0139043	8
Correlation Coefficient between X and Y	$r_{X,Y}$ = .99995			

Notice that the correlation coefficient obtained usiny proposed procedure is greater than all other correlation coefficients obtained using Dodin and Emaghraby's procedure, and also both methods (proposed and enumeration) give the same ranking of their criticality indices, most importantly criticality indices obtained for similar activities using the proposed procedure are the same, i.e., the proposed procedure always gives the same criticality indices for similar activities in any symmetric network.

Example 6 (continued):

Let us examine the effect of variance of common activitiy B on the accuracy of Dodin and Elmaghraby's procedure. Assume we change the realization times of activity B from (1,2) to (1,10) as shown in Figure 5.5.

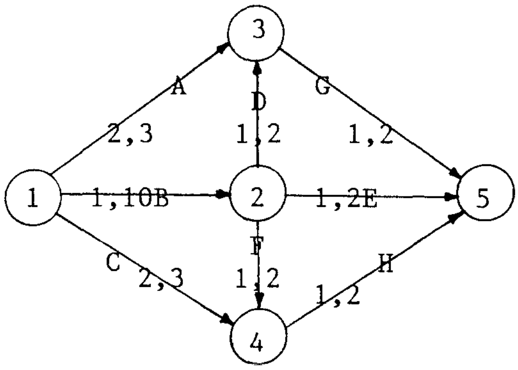


Figure 5.5

Table 6.46 shows the exact normalized criticality indices of activities of Figure 5.5, four different series of normalized criticality indices obtained using Dodin and Elmaghraby's procedure and also shows the Correlation Coefficient between exact and estimate CAs for each series of answers, and the rank (R) of Criticality.

Table 6.46

Activity	(NCA) X	R	(NACAP) Z_1	R	(NACAP) Z_2	R	(NACAP) Z_3	R	(NACAP) Z_4	R
(1,2)=B	0.7070938	1	.9105231	1	0.8783784	1	0.8915265	1	0.9236712	1
(3,5)=G	0.4942791	2	0.439737	2	0.439737	2	0.439737	2	0.439737	2
(4,5)=H	0.4942791	3	0.439737	3	0.439737	3	0.439737	3	0.439737	3
(2,3)=D	0.347826	4	0.3962748	4	0.3962748	4	0.4094229	4	0.4094229	4
(2,4)=F	0.347826	5	0.3937181	5	0.3615777	5	0.3615777	5	0.3937181	5
(1,3)=A	0.146453	6	0.0434622	8	0.0434622	8	0.0303141	8	0.0303141	8
(1,4)=C	0.146453	7	0.0460188	7	0.0781593	7	0.0781593	7	0.0460188	7
(2,5)=E	0.0114416	8	0.1205258	6	0.1205258	6	0.1205258	6	0.1205258	6
Correlation Coefficient between X and Z_i, i=1,4			r_{X,Z_1}= .93454		r_{X,Z_2}= .94110		r_{X,Z_3}= .93708		r_{X,Z_4}= .93123	

Notice that again in none of the cases the Rank of Criticality is the same as the Rank of Exact CA. In Table 6.47 we compare the exact normalized CAs with the normalized criticality indices obtained using proposed procedure.

Table 6.47:

Activity	(NCA) X	R	(NCAP) Y	R
(1,2)=B	0.7070938	1	0.7513965	1
(3,5)=G	0.4942791	2	0.4947571	2
(4,5)=H	0.4942791	3	0.4947571	3
(2,3)=D	0.347826	4	0.3704554	4
(2,4)=F	0.347826	5	0.3704554	5
(1,3)=A	0.146453	6	0.1243016	6
(1,4)=C	0.146453	7	0.1243016	7
(2,5)=E	0.0114416	8	0.0104855	8
Correlation Coefficient between X and Y			$r_{X,Y} = .99775$	

Notice that again the correlation coefficient obtained using proposed procedure is greater than all other correlation coefficients obtained using Dodin and Elmaghraby's procedure, and also both methods (proposed and enumeration) give the same ranking of their criticality indices, and the criticality indices obtianed for similar activities using the proposed procedure are the same.

Example 7:

Consider the network configuration of Figure 6.28, the duration times of activities are shown beside each activity with equal probability of occurrence.

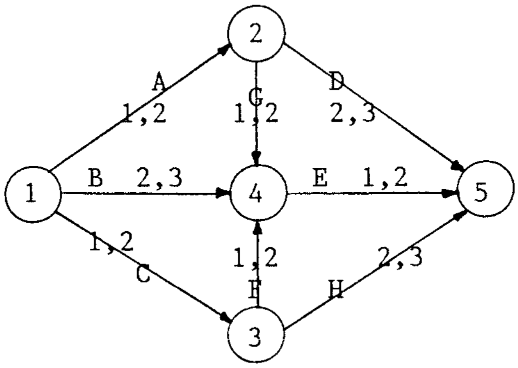


Figure 6.28

For this network again the proposed procedure provides better estimate than Dodin and Elmaghraby's procedure.

Networks of Figures 6.29 through 6.31 have been solved using three approaches (complete enumeration, Dodin and Elmaghraby's procedure and proposed procedure of this Chapter) for determining criticality indices of activities. In all projects the answers obtained using proposed procedure were found to be better than Dodin and Elmaghraby's procedure and for each project, the propcsed and enumeration method gave the same ranking of their criticaltiy indices. The realizations of each activity indicated in the arcs are assumed equally likely.

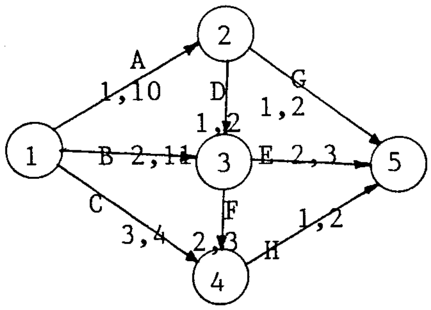


Figure 6.29

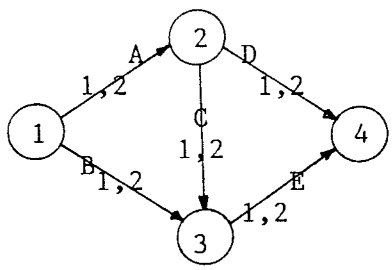


Figure 6.30

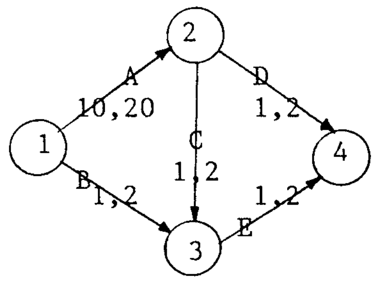


Figure 6.31

In the following we apply the proposed procedure to the network of Figure 4.49 as final example with complete enumeration.

Example 8:

Consider the network of Figuer 4.49 and the consecutive networks of that by conditioning at common activites (A and K) as shown in Figures 4.50 through 4.52.

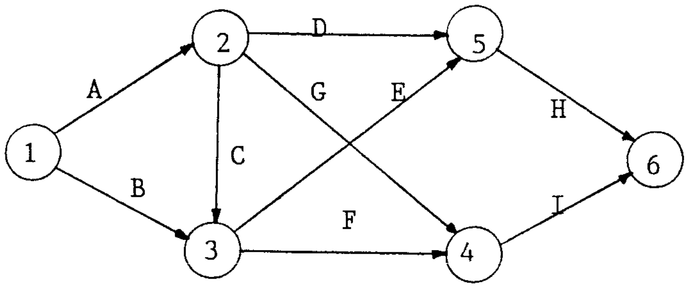


Figure 4.49

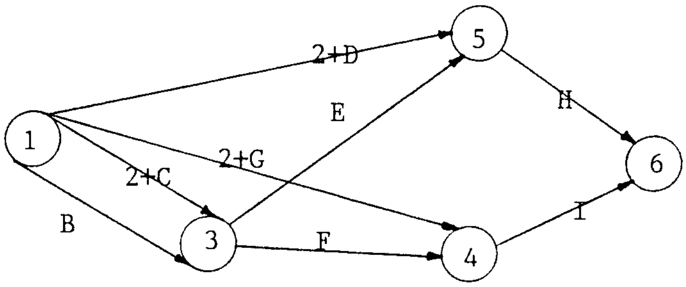


Figure 4.50

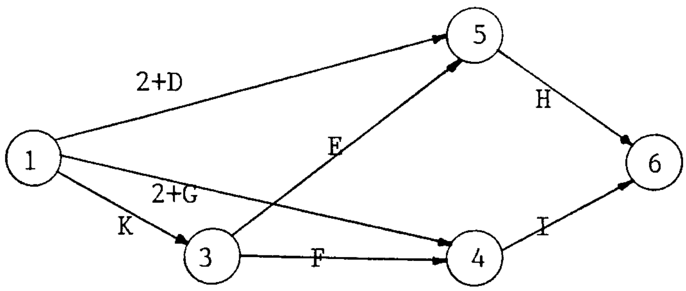


Figure 4.51

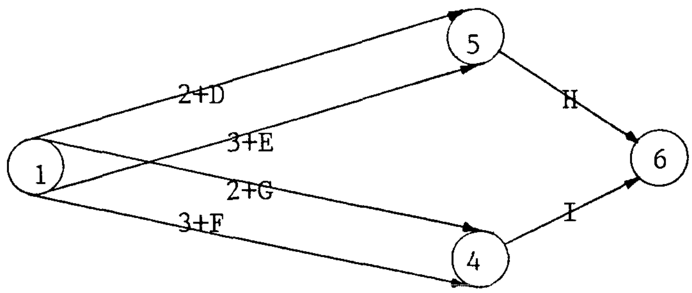


Figure 4.52

Figure 4.52 is the conditional network of Figure 4.49 given A=2 and K=3. The approximate criticality indices of activities H and I of Figure 4.52 is determined by using Tables 4.24A and 4.24B as shown in Table 6.47.

Table 6.47: Approximate criticality indices of activites H and I given A = 2 and K = 3.

	H	I
10	0.3x0.021 = 0.0063	0.021x0.3 = 0.0063
11		0.189x0.3 = 0.0567
12	0.2x0.259 = 0.0518	0.049x0.5 = 0.0245
13	0.3x0.790 = 0.2370	0.531x0.8 = 0.4248
15	0.2 x 1 = 0.2000	0.210 x 1 = 0.2100
CAP =	0.4951	0.7223
NCAP =	0.4067	0.5933

The approximate criticality indices of activities (2+G) and (3+F) is obtained from Tables 4.20C and 4.22B as shown in Table 6.48.

Table 6.48: Approximate criticality indices of activities (2+G) and (3+F).

	(2+G)	(3+F)
6		0.1x0.7 = 0.07
7		0.9x0.7 = 0.63
9	0.3x1 = 0.3	
CAP =	0.3	0.7
NCAP =	0.3	0.7
NCAP.CAP(I) =	0.3x0.5933 = 0.178	0.7x0.5933 = 0.4133

Also the approximate criticality indices of activities (2+D) and (3+E) is obtained from Tables 4.20B and 4.22A as shown in Table 6.49.

Table 6.49: Approximate criticality inidces of activites (2+D) and (3+E) given A=2 and K=3.

	(2+D)	(3+E)
7	0.1x0.5 = 0.05	0.5x 1 = 0.5
10		0.5x 1 = 0.5
CAP =	0.05	1.0
NCAP =	0.048	0.952
NCAP.CAP(H)=	0.048x0.4067 = 0.0195	0.952x0.4067 = 0.3872

Now, proposition 3 can be used to determine the approximate criticality indices of activities (2+C) and B of Figure 4.50.

For node 3 of Figure 4.50, the sum of the approximate criticality indices of E and F equals the sum of the approximate criticality indices of B and (2+C). i.e.,

CAP(B) + CAP(2+C) = CAP(E)+CAP(F)

= 0.3872 + 0.4153

= 0.8025 .

In order to find the contribution of each one from 0.8025, we need to determine the normalized value of the approximate criticality index of (2+C) and B from Tables 4.20A and 4.19B as shown in Table 6.50. Notice that K=3.

Table 6.50: Approximate criticality indices of activities (2+C) and B given A=2 and K=3.

	(2+C)	B
3	0.6	0.4
CAP =	0.6	0.4
NCAP =	0.6	0.4
(NCAP)(0.8025) =	0.4815	0.321

Finally, the approximate criticality index of A equals the sum of the approximate criticality indices of D,C and G which is equal to .0195+.4815 + .178 = .679 .

Table 6.51 shows the approximate criticality indices of all activities given A=2 and K=3.

Table 6.51: Approximate criticality indices given A=2 and K=3.

Activity	NCAP
H	0.4067
I	0.5933
G	0.1780
E	0.3872
F	0.4153
D	0.0195
C	0.4815
B	0.3210
A	0.6790

By using the same procedure for each set of conditions, the approximate criticality indices of network given specific set of conditions is obtained. In the following, final Table of each set of conditions is given.

Table 6.52: Approximate criticality indices of activites given A=2 and K=4.

Activity	NCAP
H	0.4400
I	0.5600
G	0.1680
E	0.4400
F	0.3920
D	0.0000
C	0.3328
B	0.4992
A	0.5008

Table 6.53: Approximate criticality indices of activities given A=5 and V=6.

Activity	NCAP
H	0.4067
I	0.5933
G	0.1780
E	0.3872
F	0.4153
D	0.0195
C	0.8025
B	0.0000
A	1.0000

Table 6.54: Approximate criticality indices of activites given A=5 and V=7.

Activity	NCAP
H	0.4400
I	0.5600
G	0.1680
E	0.4400
F	0.3920
D	0.0000
C	0.8320
B	0.0000
A	1.0000

By deconditioning the approximate criticality indices of Tables 6.51 through 6.54 and simple addition of corresponding criticality indices, the approximate criticality index of each activity is obtained. Tables 6.55 to 6.58 show unconditional criticality indeces of Tables 6.51 to 6.54 respectively.

Table 6.55: Unconditional criticality indices of Table 6.51.
P(A=2).P(K=3)=(0.2).(0.24) = 0.048

Activity	UNC.NCAP
H	0.4067x0.048 = 0.0195216
I	0.5933x0.048 = 0.0284784
G	0.1780x0.048 = 0.0085440
E	0.3872x0.048 = 0.0185856
F	0.4153x0.048 = 0.0199344
D	0.0195x0.048 = 0.0009360
C	0.4815x0.048 = 0.0231200
B	0.3210x0.048 = 0.0154080
A	0.6790x0.048 = 0.0325920

Table 6.56: Unconditional criticality indices of Table 6.52.
P(A=2).P(K=4)=(0.2).(0.76) = 0.152

Activity	UNC.NCAP
H	0.4400x0.152 = 0.0668800
I	0.5600x0.152 = 0.0851200
G	0.1680x0.152 = 0.0255360
E	0.4400x0.152 = 0.0668800
F	0.3920x0.152 = 0.0595800
D	0.0000x0.152 = 0.0000000
C	0.3328x0.152 = 0.0505856
B	0.4992x0.152 = 0.0758784
A	0.5008x0.152 = 0.0761216

Table 6.57: Unconditional criticality indices of Table 6.53.

P(A=5).P(V=6)=(0.8).(0.6)=0.48

Activity	UNC.NCAP
H	0.4067x0.48 = 0.195216
I	0.5933x0.48 = 0.284784
G	0.1780x0.48 = 0.085440
E	0.3872x0.48 = 0.185856
F	0.4153x0.48 = 0.199344
D	0.0195x0.48 = 0.009360
C	0.8025x0.48 = 0.385200
B	0.0000x0.48 = 0.000000
A	1.0000x0.48 = 0.480000

Table 6.58 Unconditional criticality indices of Table 6.54.
P(A=5).P(V=7)=(0.8).(0.4)= 0.32

Activity	UNC.NCAP
H	0.440x0.32 = 0.14080
I	0.560x0.32 = 0.17920
G	0.168x0.32 = 0.05376
E	0.440x0.32 = 0.14080
F	0.392x0.32 = 0.12544
D	0.000x0.32 = 0.00000
C	0.832x0.32 = 0.26624
B	0.000x0.32 = 0.00000
A	1.000x0.32 = 0.32000

Table 6.59 shows approximate criticality indices of activities.

Table 6.59: Approximate criticality indices of activities.

Activity	NCAP
H	0.0195216+0.0668800+0.195216+0.14080 = 0.42240
I	0.0284784+0.0851200+0.284784+0.17920 = 0.57760
G	0.0085440+0.0255360+0.085440+0.05376 = 0.17330
E	0.0185856+0.0668800+0.185856+0.14080 = 0.41210
F	0.0199344+0.0595840+0.199344+0.12544 = 0.40430
D	0.0009360+0.0000000+0.009360+0.00000 = 0.01030
C	0.0231200+0.0505856+0.385200+0.26624 = 0.72514
B	0.0154080+0.0758784+0.000000+0.00000 = 0.09128
A	0.0325920+0.0761216+0.480000+0.32000 = 0.90870

The approximate criticality index of each activity is shown in Figure 6.32.

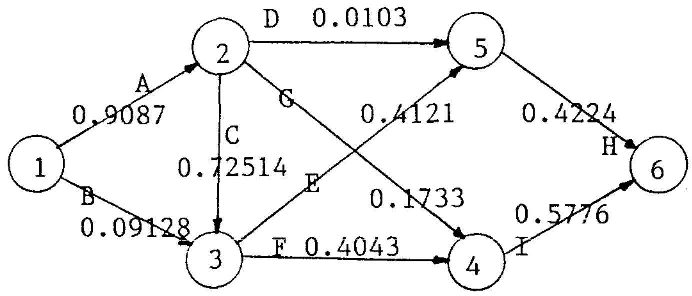


Figure 6.32

The approximate criticality index of each path is shown in Table 6.60.

Table 6.60: Approximate criticality indices of paths.

Path	NCPP	
1-2-5-6	min{CA(A),CA(D), CA(H)}	= 0.01030
1-2-3-5-6	min{CA(A),CA(C), CA(E), CA(H)}	= 0.41210
1-2-3-4-6	min{CA(A),CA(C), CA(F), CA(I)}	= 0.40430
1-2-4-6	min{CA(A),CA(G), CA(I)}	= 0.17330
1-3-5-6	min{CA(B),CA(E), CA(H)}	= 0.09128
1-3-4-6	min{CA(B),CA(F), CA(I)}	= 0.09128

SUMMARY AND CONCLUSIONS

A fundamental problem in PERT networks is to identify the activities which are critical to the achievement of the project objectives.

In an activity network if the duration of the activity is not a random variable, then the criticality of each activity represented by its float time. However, when the duration of any activity is a random variable, it is not easy to identify the criticality of each activity. In this case the criticality of an activity is known as the "criticality index," which is defined as the sum of the criticality indices of the paths containing it. The criticality index of a path is the probability that the duration of the path is greater than or equal to the duration of every other path in the network.

In this chapter a procedure to estimate criticality indices of activities in PERT networks with discrete statistically independent distributions is presented. This procedure operationally is identical to proposed procedure of Chapter 4 for determining pdf of project completion time and resultant network of that procedure is used to approximate the criticality indices of activities and paths.

Criticality indices obtained using proposed procedure are more accurate than criticality indices obtained using Approximating Procedure proposed by Dodin and Elmaghraby (1985). If common activity times have relatively large variance or the number of activities emanating from merge events are more than two, criticality indices obtained using Dodin and Elmagraby's procedure are less accurate, while these factors don't affect the accuracy of criticality indices obtained using proposed procedure.Moreover proposed procedure and enumeration method both give the same ranking of criticality indices in most PERT networks. Finally, proposed procedure could be applied in PERT networks with statistical and structural dependence relationships between activities which will be discussed in the next chapter.

CHAPTER 7 PERT NETWORKS WITH STATISTICAL AND STRUCTURAL DEPENDENCE RELATIONSHIPS

INTRODUCTION

In the conventional PERT network models it is assumed that the completion time distributions of individual activities are statistically independent. In practice there may be dependence between activities. Conditions which affect one activity, causing it to have a rapid completion time or a slow completion time, quite often affect other activities as well. Further, many managers will attempt to improve performance by switching manpower and resources to speed activities which are ahead of schedule. Most of the statistical theories of PERT network analysis proposed to date make the assumption that the duration times of activities have statistically independent distributions. This assumption is one of the possible sources of error in project completion times and criticaltiy indices.

Structural dependence relationships can interact with statistical dependence relationships and produce important effects which cannot be detected and understood using simple expected value calculations. These effects are complex, but they can be identified and understood in a Controlled Interval and Memory framework.

In this chapter effects of statistical and structural dependence relationships are examined through examples. The chapter starts with an example used by Cooper and Chapman (1987) involving an activity network model, where structural dependence arises in a natural and obvious way in network merge events, and considers the CIM treatment of it. Then the chapter presents proposed procedure and the algorithm for the PERT networks with statistical and structural dependence relationships between activities.

Example 1

The example concerns part of the construction of an offshore oil platform. This specific applicaton area is incidental : the example was designed to illustrate the CIM approach to structural dependence, to indicate the importance of structural dependence in risk calculations. The numbers provide an outline framework for discussion; insights are the major objective.

The example involves six activities: the design, materials acquisition and fabrication of the topsides modules for an offshore platform; and the design, materials acquisition and fabrication of the steel jacket itself . Finish-to-start precedence relationships are specified in Table 7.1, and shown in the precedence (activity-on-node) diagram of Figure 7.1.

Table 7.1: Activity list for an offshore platform, with finish-to-start precedence relationships.

NUMBER	LABEL	PREDECESSORS
MODULES SEQUENCE		
1	Design	
2	Materials	1
3	Fabrication	2
JACKET SEQUENCE		
4	Design	
5	Materials	4
6	Fabrication	1,5

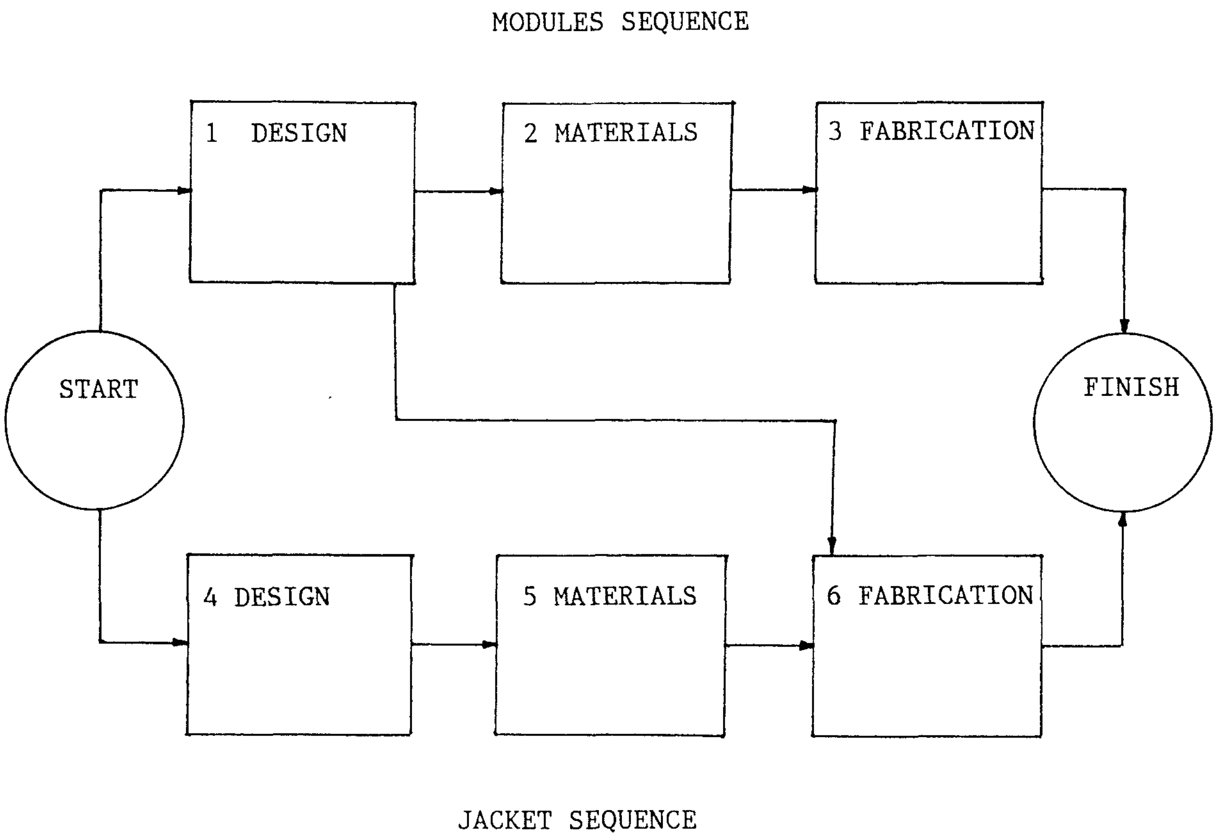


Figure 7.1: Activity-on-node precedence diagram, with finish-to-start links.

Table 7.2 shows the durations of the activities in the Modules sequence, with time expressed in half-year units. Under 'Modules Fabrication', the second and third lines show conditional probabilities, conditional on the Modules Design duration D1; for example,

$$P(D3=3|D1=4)=0.3.$$

The fourth line shows unconditional probabilities; for example,

$$\begin{aligned} P(D3=3) &= P(D3=3|D1=4)P(D1=4) \\ &\quad +P(D3=3|D1=5)P(D1=5) \\ &= 0.3x0.6+0.8x0.4 \\ &= 0.5. \end{aligned}$$

There is negative statistical dependence between Modules Fabrication and Modules Design: if Design takes a long time, then Fabrication can be speeded up, at additional cost. The table shows expected values:

it is useful to keep track of expected values, and this can be done authomatically with appropriate software.

Table 7.2: Modules sequence activity durations, in half-year units.

1. MODULES DESIGN	D1 =	4	5	
	P =	0.6	0.4	E=4.4
2. MODULES MATERIALS	D2 =	2	3	
	P =	0.3	0.7	E=2.7
3. MODULES FABRICATION	D3 =	3	4	
P GIVEN MODULES DESIGN	D1 = 4	0.3	0.7	E=3.7
	5	0.8	0.2	3.2
	P =	0.5	0.5	E = 3.5

Table 7.3 shows the durations of the activities in the Jacket sequence. There is positive statistical dependence between Jacket Design and Modules Design, and between Jacket Fabrication and Modules Design. Problems with the design of the modules, particularly if they involve changes in the number or size of modules affecting their total weight, will lead to related problems with the design and fabrication of the jacket. Again, the dependence is specified in terms of conditional probabilities, and unconditional probabilities and expected values are also shown.

Table 7.3: Jacket sequence activity durations, in half-year units.

4. JACKET DESIGN	D4 =		2	3	
P GIVEN MODULES DESIGN	D1 =	4	0.7	0.3	E=2.3
		5	0.2	0.8	2.8
	P =		0.5	0.5	2.5
5. JACKET MATERIALS	D5 =		2	3	
	P =		0.3	0.7	E=2.7
6. JACKET FABRICATION	D6 =		4	5	
P GIVEN MODULES DESIGN	D1 =	4	0.7	0.3	E=4.3
		5	0.2	0.8	4.8
	P =		0.5	0.5	E=4.5

The simple six-activity structure of this example may be appropriate in practice. A coarse activity structure is often used in practice in order to facilitate a detailed treatment of the risks associated with each activity. Because the activities used here are composites of many lower level (more detailed) activities, there may be precedence relationship overlaps, as shown in Table 7.4. To examine all the activities at a level of detail which avoids the overlaps would often be too complicated and unnecessary for risk analysis purposes. The same distribution of overlap has been assumed in each here, for simplicity. Expected values are shown, for ease of interpretation.

Table 7.4: Activity Precedence Relationship Overlaps.

1 - 2	Modules Design and Modules Materials			
	L =	0	1	
	P =	0.5	0.5	E=0.5
2 - 3	Modules Materials and Modules Fabrication			
	L =	0	1	
	P =	0.5	0.5	E=0.5
5 - 6	Jacket Materials and Jacket Fabrication			
	L =	0	1	
	P =	0.5	0.5	E=0.5

EXPECTED VALUE CALCULATIONS

Given the expected activity durations and the expected overlaps, standard network calculations can be performed to provide an initial assessment of the project schedule. A forward pass calculates activity earliest start times and project earliest finish time, and a backward pass calculates activity latest start times. Link float can be computed using

FLOAT (link i - j) = Latest Start (node j)
- Earliest Finish (node i)
+ Overlap (link i - j)
= LS(j) - ES(i) - D(i) + OL(i,j).

Expected activity durations and overlaps, earliest and latest activity start times, project finish time and floats are shown in Figure 7.2. The floats indicates that the Modules sequence is critical, and that the Jacket sequence is not critical. On the basis of this expected value assessment, the link 1 - 6 has the largest link float and looks irrelevant.

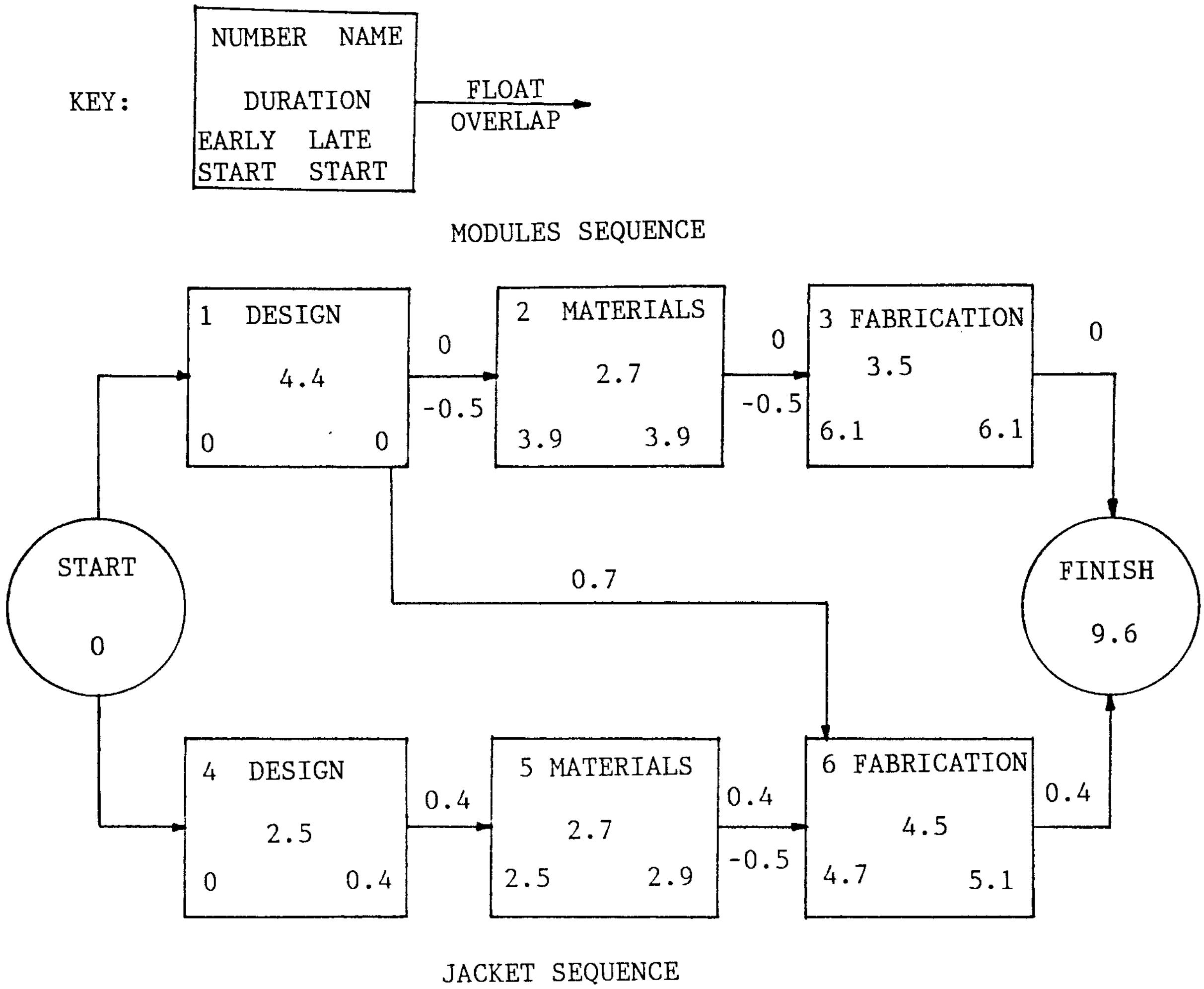


FIGURE 7.2: Expected activity durations and overlaps, activity early and late start times, project finish time, and link floats.

DISTRIBUTION CALCULATIONS

The expected value calculations of Figure 7.2 are easy to perform. However, they do not take into account the probabilistic aspects of the dependence relationships that exist between the activities in the project network structure. It is necessary to examine the distributions of start and finish time for the activities and for the project network as a whole if the effects of these relationships are to be understood fully.

Modules Sequences

Table 7.5 shows the time distributions for the Modules sequence. Table 7.5A, Modules Materials start time, S2, is the Modules Design distribution minus the overlap between Design and Materials, performed as a standard CIM subtraction calculation. This calculation format keeps a memory of the Modules Design aspect of S2 uncertainty, needed later in the computation sequence. The probabilities computed in Table 7.5 in the D1=4 and 5 columns are joint probabilities, rather than conditional probabilities like those in Table 7.2, 7.3 and 7.4; for example,

Table 7.5A: Modules Materials start time.

	P|MODULES DESIGN		UNCONDITIONAL
	D1 = 4	D1 = 5	P
S2 = 3	0.5x0.6 = 0.3		0.3
4	0.5x0.6 = 0.3	0.5x0.4 = 0.2	0.5
5		0.5x0.4 = 0.2	0.2
E =	3.5	4.5	3.9

Table 7.5B: Modules Materials finish time.

	P|MODULES DESIGN		UNCONDITIONAL
	D1 = 4	D1 = 5	P
F2 = 5	0.3x0.3 = 0.09		0.09
6	0.3x0.3+0.7x0.3 = 0.30	0.3x0.2 = 0.06	0.36
7	0.7x0.3 = 0.21	0.3x0.2+0.7x0.2 = 0.20	0.41
8		0.7x0.2 = 0.14	0.14
E =	6.2	7.2	6.6

Table 7.5C: Modules Fabrication start time.

	P|MODULES DESIGN D1 = 4	D1 = 5	UNCONDITIONAL P
S3 = 4	0.5x0.09 = 0.045		0.045
5	0.5x0.09 + 0.5x0.30 = 0.195	0.5x0.06 = 0.03	0.225
6	0.5x0.30 + 0.5x0.21 = 0.255	0.5x0.06 + 0.5x0.20 = 0.130	0.385
7	0.5x0.21 = 0.105	0.5x0.20 + 0.5x0.14 = 0.170	0.275
8		0.5x0.14 = 0.070	0.070
E =	5.7	6.7	6.1

Table 7.5D: Modules Fabrication finish time.

	P|MODULES DESIGN D1 = 4	D1 = 5	UNCONDITIONAL P
F3 = 7	0.3x0.045 = 0.0135		0.0135
8	0.3x0.195+0.7x0.045 = 0.0900	0.8x0.03 = 0.0240	0.1140
9	0.3x0.255+0.7x0.195 = 0.2130	0.8x0.13+0.2x0.03 = 0.1100	0.3230
10	0.3x0.105+0.7x0.255 = 0.2100	0.8x0.17+0.2x0.13 = 0.1620	0.3720
11	0.7x0.105 = 0.0735	0.8x0.07+0.2x0.17 = 0.0900	0.1635
12		0.2x0.07 = 0.0140	0.0140
E =	9.4	9.9	9.6

P(S2=4 and D1=4) = P(no overlap) P(D1=4)

= 0.5 x 0.6

= 0.3.

Because these are joint probabilities, the expected value calculations are a little different; for example,

E(S2|D1=4) = (3x0.3+4x0.3)/(0.3+0.3)

= 3.5.

Tables 7.5B, 7.5C and 7.5D show the distributions of Modules Materials finish time (5A plus the Modules Materials duration distribution), Modules Fabrication start time (5B minus the overlap between materials and Fabrication), and Modules Fabrication finish time (5C plus the Modules Fabrication duration distribution). The memory of the Modules Design duration is retained throughout, for later use.

Figure 7.3 illustrates the start and finish time distributions for the Modules sequence in Table 7.5. In Figure 7.3, the overlap allows a 'jump back' in the time sequence of activities. The 'curves' plotted here are based on histogram intervals, centred on whole numbers of half year units.

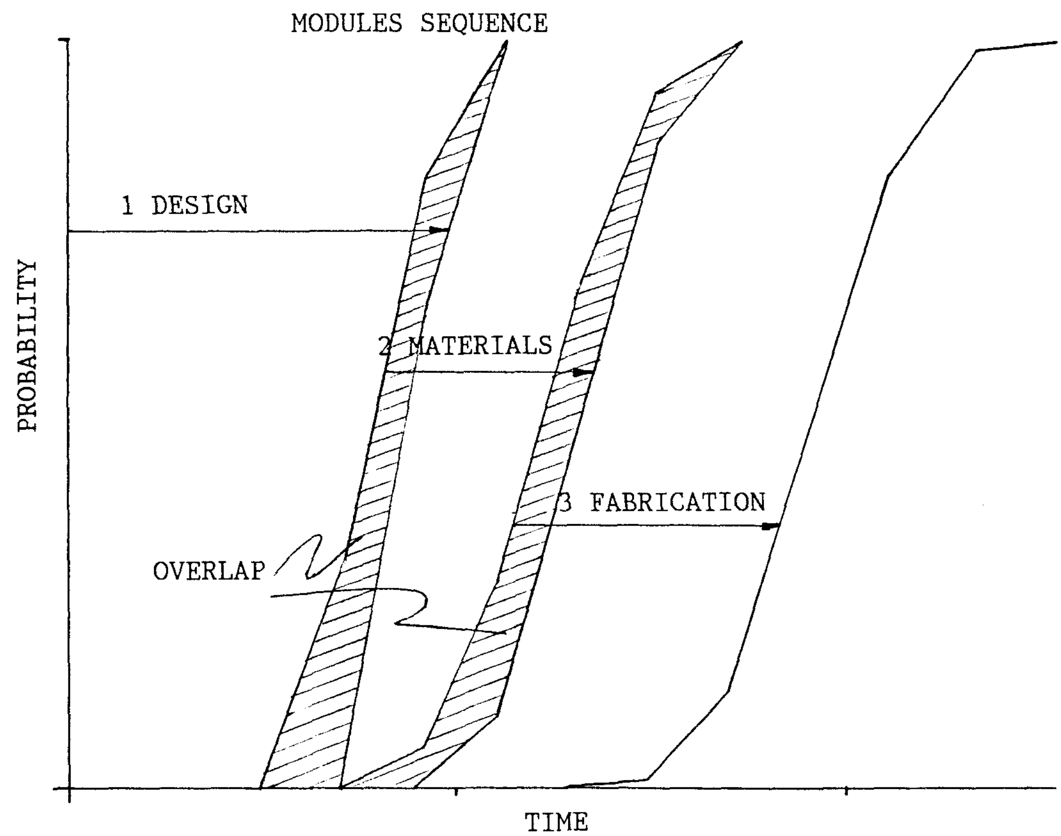


FIGURE 7.3: Time distributions for the Modules sequence.

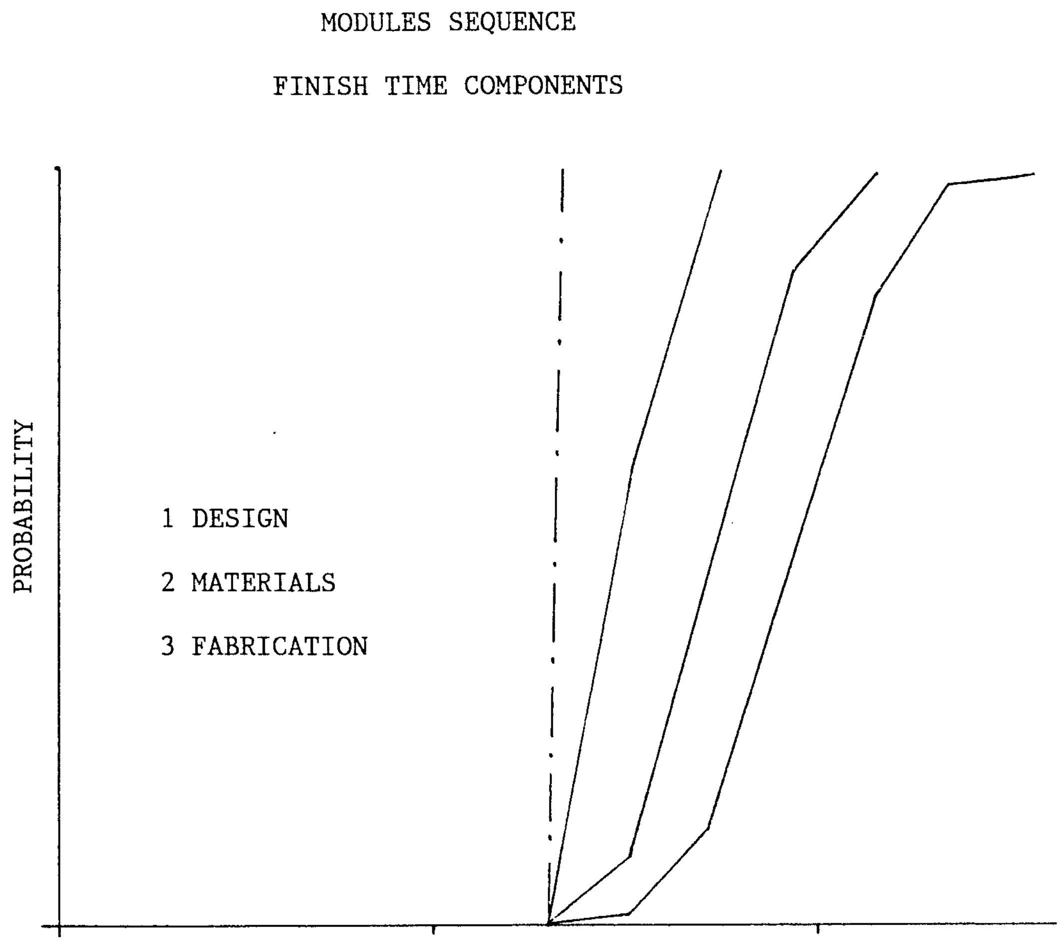


FIGURE 7.4: Contributions of each activity variation to the Modules sequence finish time variation. (Note that the Materials and Fabrication components include a variation due to overlap.)

The contributions of each component activity variations to the modules sequence finish time are shown in Figure 7.4. The lines shown are the Design start time S1, the Design finish time F1, the Materials finish time F2, and the Fabrication finish time F3 (which is the Modules sequence finish time), all plotted from a common zero probability point. The areas between the lines indicate the variation contributions of each activity plus associated contributions from the overlap with prior activities.

In the Modules sequence, the expected start times derived from the distributional results in Table 7.5 are the same as the start times calculated from the expected activity durations and expected overlaps of Figure 7.2.

Jacket Sequence

Table 7.6 shows a sequence of calculations similar to Table 7.5, for the Jacket sequence. Again, the calculations keep a memory of Modules Design. Table 7.6A, Jacket Design finish time, converts the Table 7.3 values from conditional probabilities to joint probabilities; for example

$$P(F4=2 \text{ and } D1=4) = P(D1=4)\ P(F4=2|D1=4)$$
$$= 0.6 \times 0.7$$
$$= 0.42.$$

Table 7.6A: Jacket Design finish time.

	P|MODULES DESIGN D1 = 4	D1 = 5	UNCONDITIONAL P
F4 = 2	0.6x0.7 = 0.42	0.4x0.2 = 0.08	0.5
3	0.6x0.3 = 0.18	0.4x0.8 = 0.32	0.5
E =	2.3	2.8	2.5

Table 7.6B: Jacket Materials finish time.

	P|MODULES DESIGN D1 = 4	D1 = 5	UNCONDITIONAL P
F5 = 4	0.3x0.42 = 0.126	0.3x0.08 = 0.024	0.15
5	0.3x0.18+0.7x0.42 = 0.348	0.3x0.32+0.7x0.08 = 0.152	0.50
6	0.7x0.18 = 0.126	0.7x0.32 = 0.224	0.35
E =	5.0	5.5	5.2

Table 7.6C: Jacket Fabrication start time, considering materials only.

	P|MODULES DESIGN D1 = 4	P|MODULES DESIGN D1 = 5	UNCONDITIONAL P
S6' = 3	0.5x0.126 = 0.063	0.5x0.024 = 0.012	0.075
4	0.5x0.126 + 0.5x0.348 = 0.237	0.5x0.024+0.5x0.152 = 0.088	0.325
5	0.5x0.348 + 0.5x0.126 = 0.237	0.5x0.152+0.5x0.224 = 0.188	0.425
6	0.5x0.126 = 0.063	0.5x0.224 = 0.112	0.175
E =	4.5	5.0	4.7

Table 7.6D: Jacket Fabrication start time.

	P|MODULES DESIGN D1 = 4	P|MODULES DESIGN D1 = 5	UNCONDITIONAL P
S6 = 4	0.063+0.237 = 0.300		0.300
5	0.237	0.012+0.088+0.188 = 0.288	0.525
6	0.063	0.112	0.175
E =	4.605	5.280	4.875

Table 7.6E: Jacket Fabrication finish time.

	P|MODULES DESIGN		UNCONDITIONAL
	D1 = 4	D1 = 5	P
F6 = 8	0.7x0.300 = 0.2100		0.2100
9	0.7x0.237+0.3x0.300 = 0.2559	0.2x0.288 = 0.0576	0.3135
10	0.7x0.063+0.3x0.237 = 0.1152	0.2x0.112+0.8x0.288 = 0.2528	0.3680
11	0.3x0.063 = 0.0189	0.8x0.112 = 0.0896	0.1085
E =	8.905	10.080	9.375

Table 7.6B, Jacket Materials finish time, is Table 7.6A plus the Jacket Materials duration distribution. Table 7.6C, the Jacet Fabrication start time considering Materials only, is 6B minus the overlap. Table 7.6D, jacket Fabrication start time, takes into account the Modules Design finish time. Because this is a merge event, a 'greatest' operation is used rather than addition, but the CIM principle are the same; for example,

$$
\begin{aligned}
P(S6=4 \text{ and } D1=4) &= P(S6=3 \text{ and } D1=4) \\
&+ P(S6=4 \text{ and } D1=4) \\
&= 0.063+0.237 \\
&= 0.300.
\end{aligned}
$$

Table 7.6E, Jacket Fabrication finish time, is 6D plus the Jacket Fabrication duration distribution.

Figure 7.5 shows the start and finish times for the activities in the Jacket sequence. Between Materials and Fabrication, there is a 'jump back' due to the overlap between these activities, and a 'jump forward' due to the precedence link from Modules Design. The Modules Design component, shown shaded in Figure 7.5, is a variation attributable to structural dependence of a more complex form than the additions and subtractions considered prior to this in the precedence network framework.

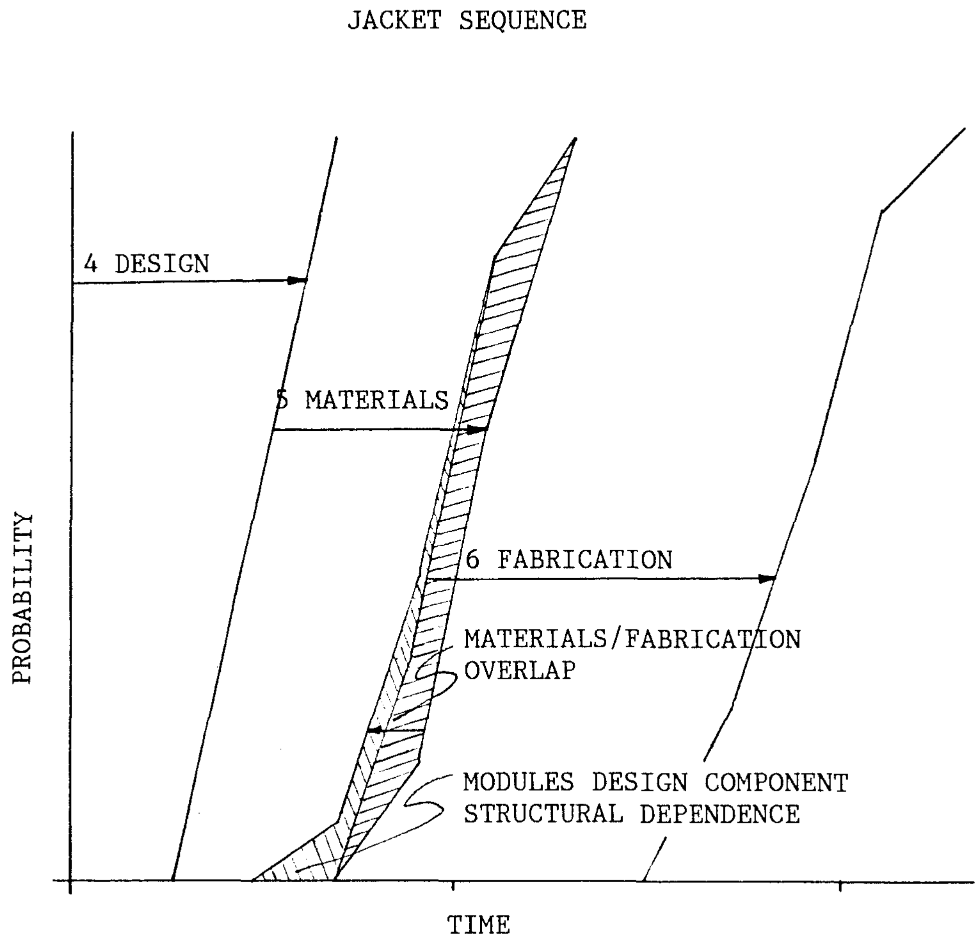


FIGURE 7.5 Time distributions for the Jacket sequence.

The expected activity start times for the Jacket sequence calculated from the CIM distributional results in Table 7.6 are shown in Figure 7.6, with the values calculated from the expected activity durations shown in parentheses. The structural dependence embodied in the link 1 - 6 has increased the expected duration of the Jacket sequence from 9.2 to 9.4 half-year time units. At the merge event at the start of Jacket Fabrication, there is an expected delay of 0.2 due to Modules Design delays. The link 1 - 6 will be critical at least some of the time, although link 1 - 6 seemed irrelevant when judged solely on the basis of expected values.

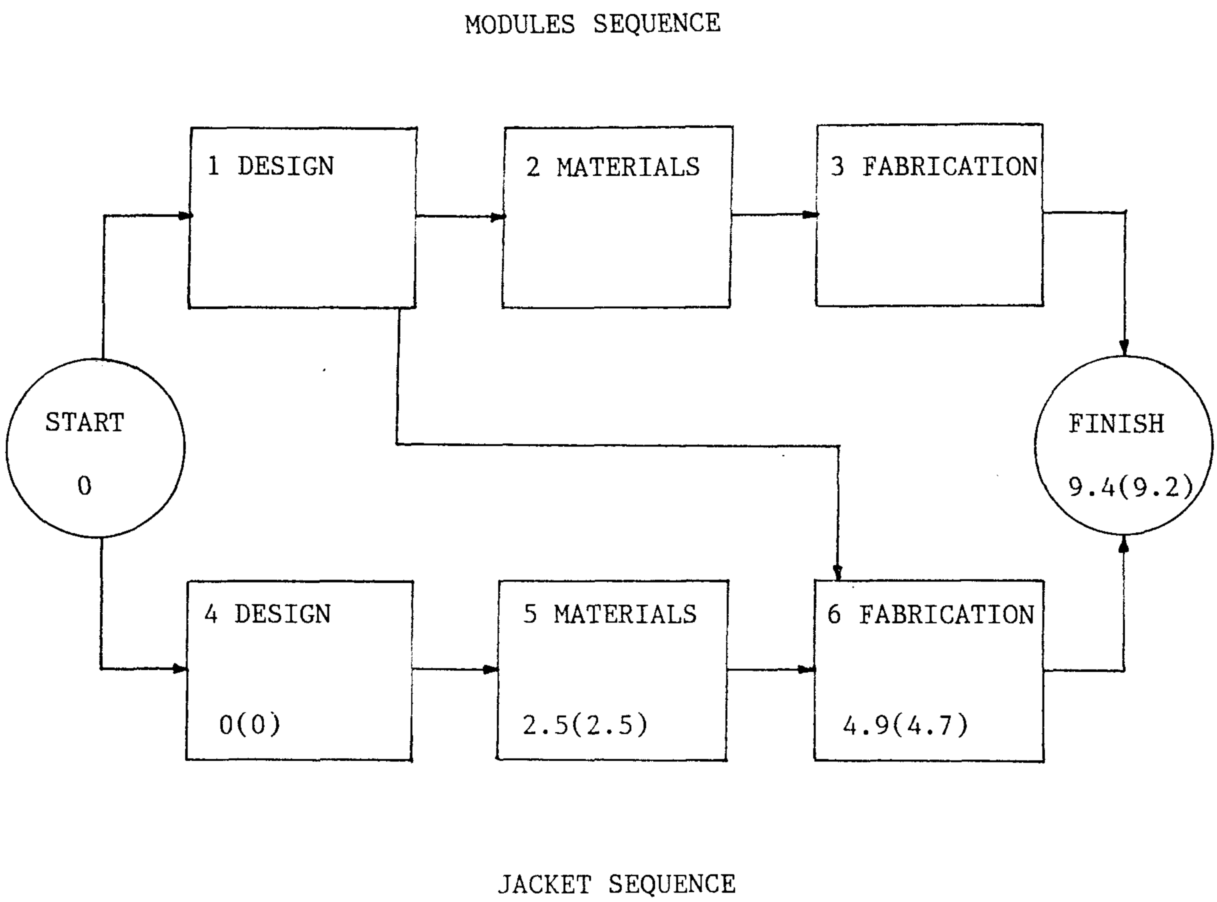


FIGURE 7.6: Expected activity start times and sequence finish time for the jacket sequence, calculated using CIM models, with the previous values shown in brackets.

Criticality Indices

To further examine the links leading into the merge events at the start of Jacket Fabrication, the criticality indices of the links can be calculated. The criticality index of a link indicated the conditional probability that the link will be critical. Calculations based on joint probabilities are shown in Table 7.7, and criticality is illustrated in Figure 7.7. For example, when the Modules Design duration D1 = 4 and the Jacket Fabrication start time S6=4, the probability that the Jacket Fabrication start time is constrained by link 1 - 6 (link 1 - 6 is critical) is

$$P(\text{constrained by } D1=4) = P(S6=3 \text{ and } D1=4) + P(S6=4 \text{ and } D1=4) = 0.063+0.237 = 0.300;$$

the probability that it is constrained by link 5 - 6 is P(S6=4 and D1=4) = 0.237: and 0.237 is also the probability that it is constrained by both. In Table 7.7, the 'Total' row indicates the joint probability that a link is critical and the particular Modules Design duration D1 is obtained. The criticality index is the corresponding conditional probability, calculated by dividing the joint probability by the probability associated with the particular D1 value; for example,

$$C(\text{link } 1 - 6 \mid D1=4) = P(1 - 6 \text{is critical and } D1=4)/P(D_1=4) = 0.3/0.6 = 0.5.$$

Table 7.7: Jacket Fabrication start time criticality indices, joint probabilities.

MODULES DESIGN	D1 = 4			D1 = 5			D1 = 4 or 5		
LINK	1-6	5-6	BOTH	1-6	5-6	BOTH	1-6	5-6	BOTH
S6 = 4	P = 0.300	0.237	0.237				0.300	0.237	0.237
5		0.237		0.288	0.188	0.188	0.288	0.425	0.188
6		0.063			0.112			0.175	
TOTAL	P = 0.300	0.537	0.237	0.288	0.300	0.188	0.588	0.837	0.425
CRITICALITY INDEX	C = 0.500	0.895	0.395	0.720	0.750	0.470	0.588	0.837	0.425

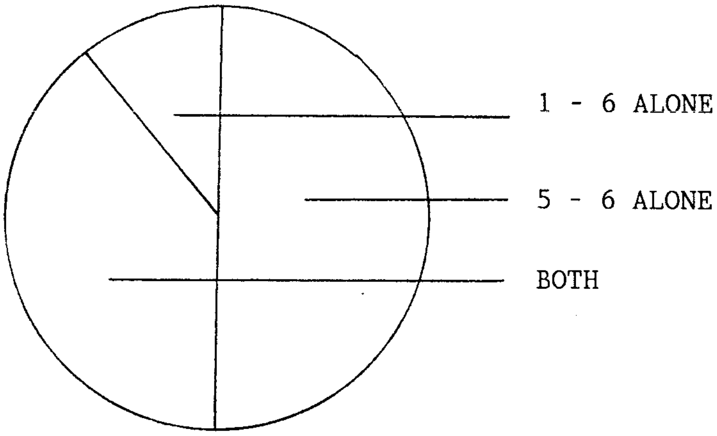
FIGURE 7.7:

Jacket Fabrication start time criticality pie diagrams.

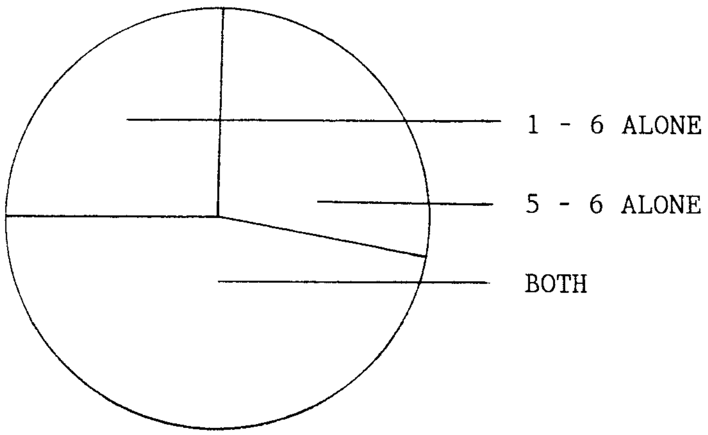
MODULES

DESIGN

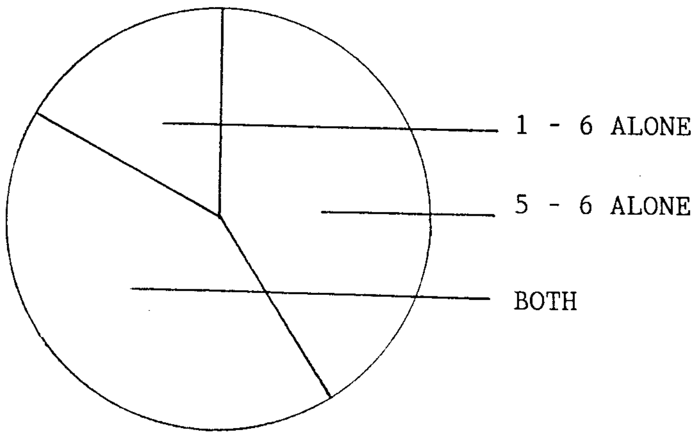
D1 = 4



D1 = 5



D1 = 4 OR 5



When the Modules Design duration D1=4, the link 1 - 6 alone is critical for only a small proportion (0.500 - 0.395 = 0.105) in comparison with link 5 - 6 alone (0.895 - 0.395 = 0.500). Link 1 - 6 looks more important when the Modules Design duration is longer (D1=5), which is not surprising. Overall, link 1 - 6 has a criticality index of nearly 0.6, in marked contrast to the initial expected-value assessment.

THE PROJECT AS A WHOLE

It is now possible to look at the project as a whole, using Table 7.8. Tables 7.8A and 7.8B convert the joint distributions for Modules Fabrication finish time (from Table 7.5D) and Jacket Fabrication finish time (from Table 7.6E) to rounded conditional distributions; for example, in Table 7.8A,

$$
\begin{aligned}
P(F3=11 \mid D1=4) &= P(F3=11 \text{ and } D1=4)/P(D1=4) \\
&= 0.0735/0.6 \\
&= 0.1225 \\
&= 0.12.
\end{aligned}
$$

Table 7.8C gives the project finish time conditional distribution, calculated using a CIM 'greatest' operation at the merge event; for example,

$$
\begin{aligned}
P(F3 \leqslant 9 \mid D1=4) &= P(F3=7 \mid D1=4) + P(F3=8 \mid D1=4) \\
&\quad + P(F3=9 \mid D1=4) \\
&= 0.02 + 0.15 + 0.36 \\
&= 0.53, \\
P(F6 \leqslant 8 \mid D1=4) &= P(F6=8 \mid D1=4) \\
&= 0.35, \\
P(FP=9 \mid D1=4) &= P(F6=8 \mid D1=4)\ P(F3=9 \mid D1=4) \\
&\quad + P(F6=9 \mid D1=4)\ P(F3 \leqslant 9 \mid D1=4) \\
&= 0.35 \times 0.36 + 0.43 \times 0.53 \\
&= 0.3539.
\end{aligned}
$$

Table 7.8D shows the corresponding joint distribution for Project finish time; for example,

$$P(FP=9 \text{ and } D1=4) = P(FP=9 \mid D1=4)\ P(D1=4)$$
$$= 0.3539 \times 0.6$$
$$= 0.21234.$$

The Modules sequence, Jacket sequence and Project finish time distributions are illustrated in Figure 7.8.

Table 7.8A: Modules Fabrication finish time rounded conditional distributions

	P|MODULES DESIGN	
	D1 = 4	D1 = 5
F3 = 7	0.02	
8	0.15	0.06
9	0.36	0.28
10	0.35	0.40
11	0.12	0.23
12		0.03
E =	9.4	9.9

Table 7.8B: Jacket Fabrication finish time rounded conditional distributions.

	P|MODULES DESIGN	
	D1 = 4	D1 = 5
F6 = 8	0.35	
9	0.43	0.14
10	0.19	0.64
11	0.03	0.22
E =	8.9	10.1

Table 7.8C: Project finish time conditional distributions.

	P|MODULES DESIGN			
	D1 = 4		D1 = 5	
FP = 8	.35x17	=.0595		
9	.35x.36+.43x.53	=.3539	.14x.34	=.0476
10	.35x.35+.43x.35+.19x.88	=.4402	.14x.40+.64x.74	=.5296
11	.35x.12+.43x.12+.19x.12+.03x1	=.1464	.14x.23+.64x.23+.22x.97	=.3928
12			.14x.03+.64x.03+.22x.03	=.0300
E =		9.7		10.4

Table 7.8D: Project finish time joint distributions.

	P|MODULES DESIGN		UNCONDITIONAL
	D1 = 4	D1 = 5	P
FP = 8	.03570		.04
9	.21234	.01904	.23
10	.26412	.21184	.47
11	.08784	.15712	.25
12		.01200	.01
E =	9.7	10.4	10.0

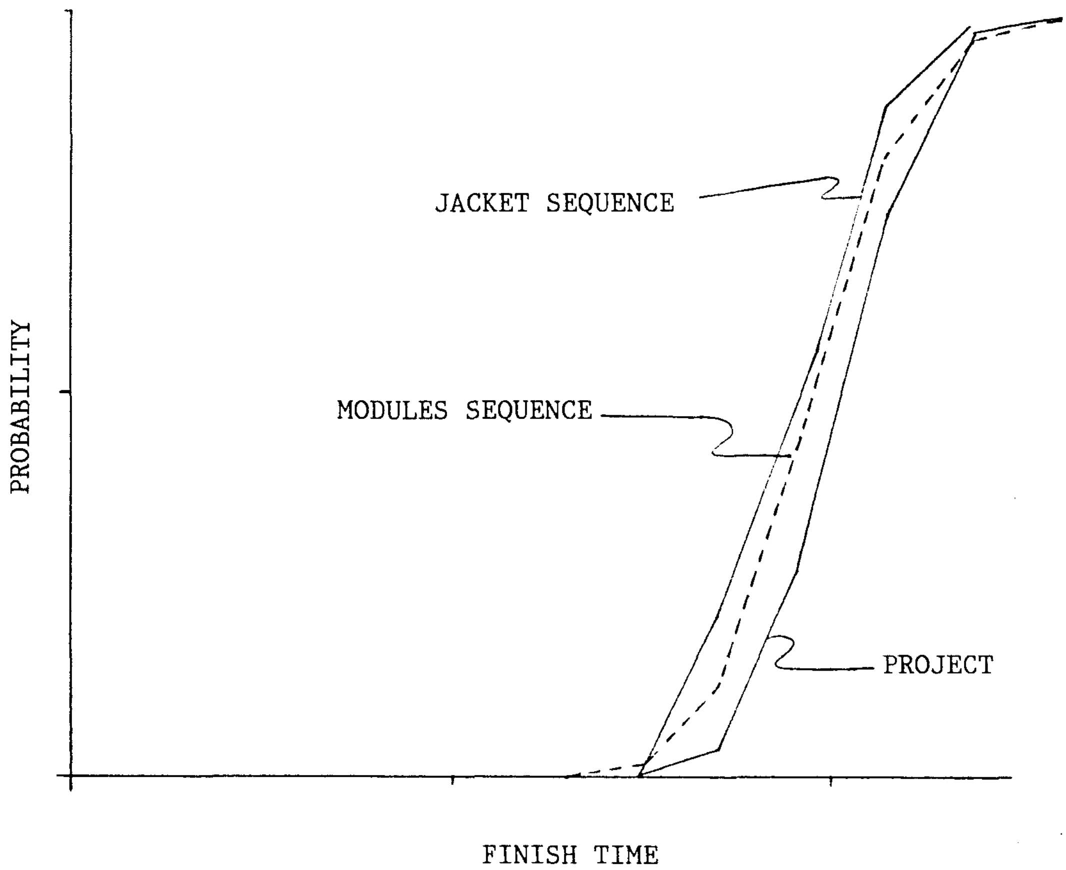


FIGURE 7.8: Sequence finish time distributions and project finish time distribution.

Table 7.9 shows the criticality calculations for the project as a whole. The computation process is similar to that used for the criticality indices of the links leading into the Jacket Fabrication activity (Table 7.7), but joint probabilities are used here instead of conditional probabilities. The approach illustrated is slightly different, but the principles are the same. For example, in Table 7.9A with the Modules Design duration D1=4 and a project duration FP=9,

$$P\,(FP=9 \text{ and Jacket sequence critical} \mid D1=4)$$
$$= P(F6=9 \mid D1=4)\ P(F3 \leqslant 9 \mid D1=4)$$
$$= 0.43 \times 0.53$$
$$= 0.2279,$$

P (FP=9 and Modules sequence critical|D1=4)

= P(F6=8|D1=4) P(F3=9|D1=4) + P (F6=9|D1=4) P(F3=9|D1=4)

= 0.35x0.036 + 0.43x0.036

= 0.2808,

P (FP=9 and both sequences critical|D1=4)

= P(F6=9|D1=4) P(F3=9|D1=4)

= 0.43x0.36

= 0.1548.

Table 7.9A: Project finish time criticality indices, Part 1 of 3.

P|MODULES DESIGN D1 = 4

	JACKET SEQUENCE	MODULES SEQUENCE		BOTH
FP=8	.35x.17 = .0595	.35x.15	= .0525	.35x.15=.0525
9	.43x.53 = .2279	.35x.36+.43x.36	= .2808	.43x.36=.1548
10	.19x.88 = .1672	.35x.35+.43x.35+.19x.35	= .3395	.19x.35=.0665
11	.03x1.00= .0300	.35x.12+.43x.12+.19x.12+0.03x.12	= .1200	.03x.12=.0036
C =	.4846		.7928	.2774

Table 7.9B: Project finish time criticality indices, Part 2 of 3.

P | MODULES DESIGN D1 = 5

	JACKET SEQUENCE	MODULES SEQUENCE		BOTH
FP=8				
9	.14x.34=.0476	.14x.28	= .0392	.14x.28=.0392
10	.64x.74=.4736	.14x.40+.64x.40	= .3120	.64x.40=.2560
11	.22x.97=.2134	.14x.23+.64x.23+.22x.23	= .2300	.22x.23=.0506
12		.14x.03+.64x.03+.22x.03	= .0300	
C =	.7346		.6112	.3458

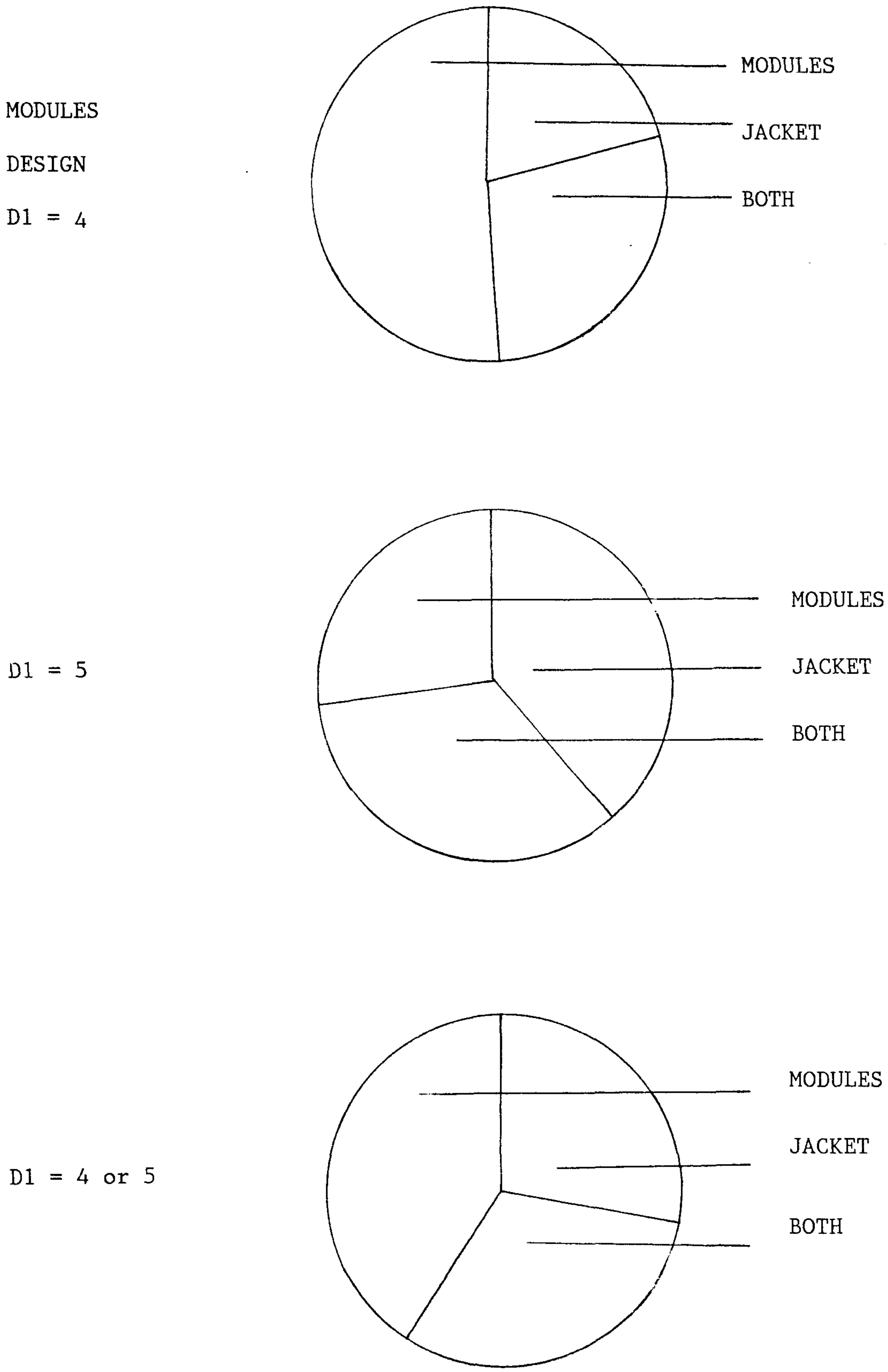
Table 7.9C: Project finish time criticality indices, Part 3 of 3.

P | MODULES DESIGN D1 = 4 or 5

	JACKET SEQUENCE	MODULES SEQUENCE	BOTH
FP=8	.03570	.03150	.03150
9	.15578	.18416	.10856
10	.28976	.32850	.14230
11	.10336	.16400	.02240
12		.01200	
C =	.58460	.72016	.30476

FIGURE 7.9:

Project finish time criticality pie diagrams.



The criticality indices are illustrated in Figure 7.9. When the Modules Design duration is short (D1=4), the Modules sequence is more important than the Jacket sequence, but when the Modules Duration is longer (D1=5), then the Jacket sequence is more important! This counter-intuitive result arises for a number of reasons: the negative statistical dependence between Modules Design and Modules Fabrication offsets the effect of increased Modules Design duration on the Modules sequence; the positive statistical dependence between Modules Deisgn and both Jacket Design and Jacket Fabrication means Modules Design delays have a large positive impact on Jacket sequence delays; the structural precedence link between Modules Design and Jacket Fabrication introduces further postitive dependence. Thus Modules Design delays impacts the Jacket sequence more than the Modules sequence, due to both statistical and structural dependence.

Figure 7.10 shows the expected activity start and Project finish times calculated usng the CIM approach as just outlined, with the previous estimates in brackets. The expected values are higher than the original estimates for both merge events. The structural dependence relationships have had a big impact on expected values, with an increase of 0.2 at each merge. Criticality indices are also shown. These criticality indices indicate that the Jacket sequence is almost as important as the Modules sequence for project completion, a very different picture from the misleading representation of Figure 7.2.

Figure 7.11 shows Project finish time distributions conditional on Modules Design duration .The gap between these curves and the associated difference in expected values, from E = 9.7 when D1 =4 to E = 10.4 when D1 =5, indicates that the Modules Design activity is very important. This might lead to a change in plans, to avoid problems at the outset. For example, a revised precedence arrangement between sequences and a delayed start to the Jacket sequence might be considered, as illustrated by Figure 7.12. This would change the structural dependence relationships, and ought to affect the statistical dependence of the Jacket sequence on Modules Design as well.

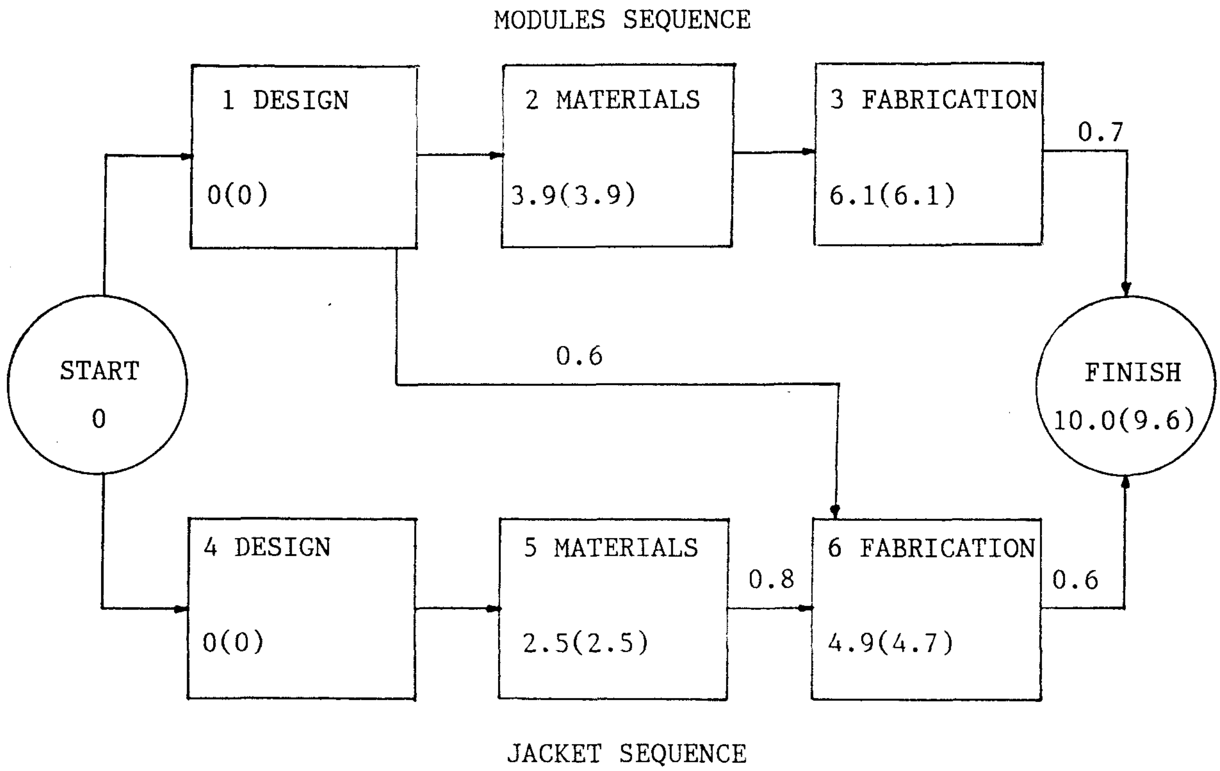


FIGURE 7.10: Expected activity start and project finish times, calculated using CIM models, with the previous values shown in brackets. Link criticality indices are also shown.

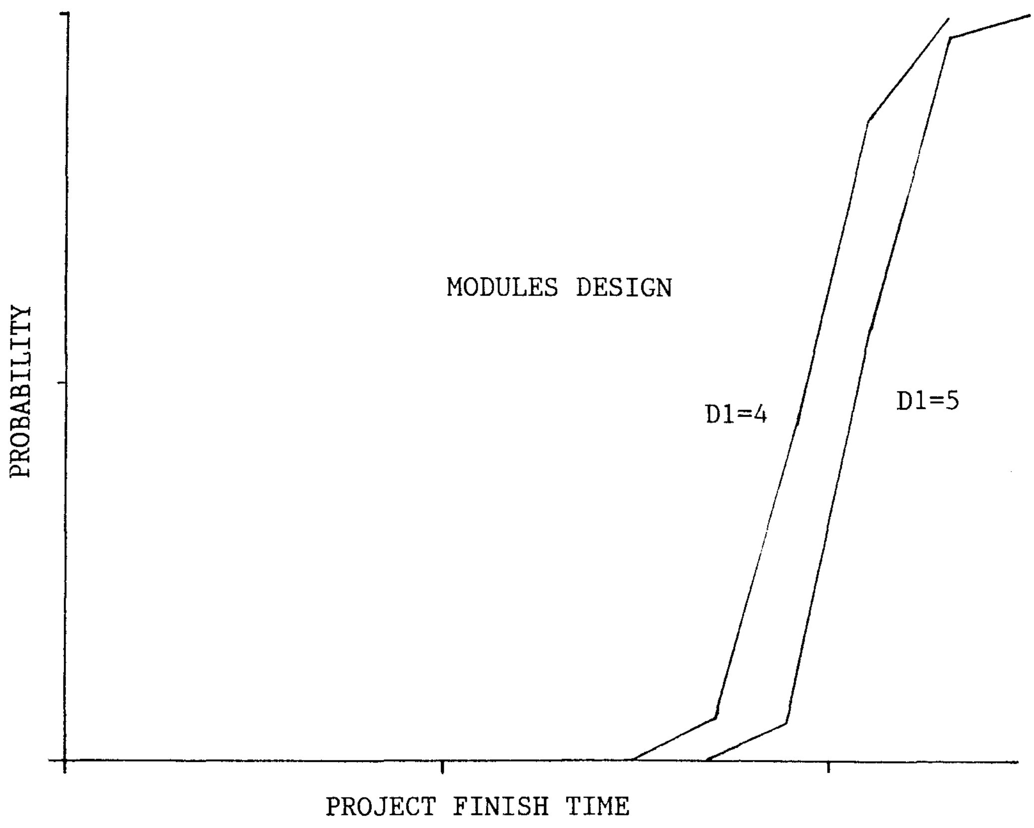


FIGURE 7.11: Project finish time conditional distributions.

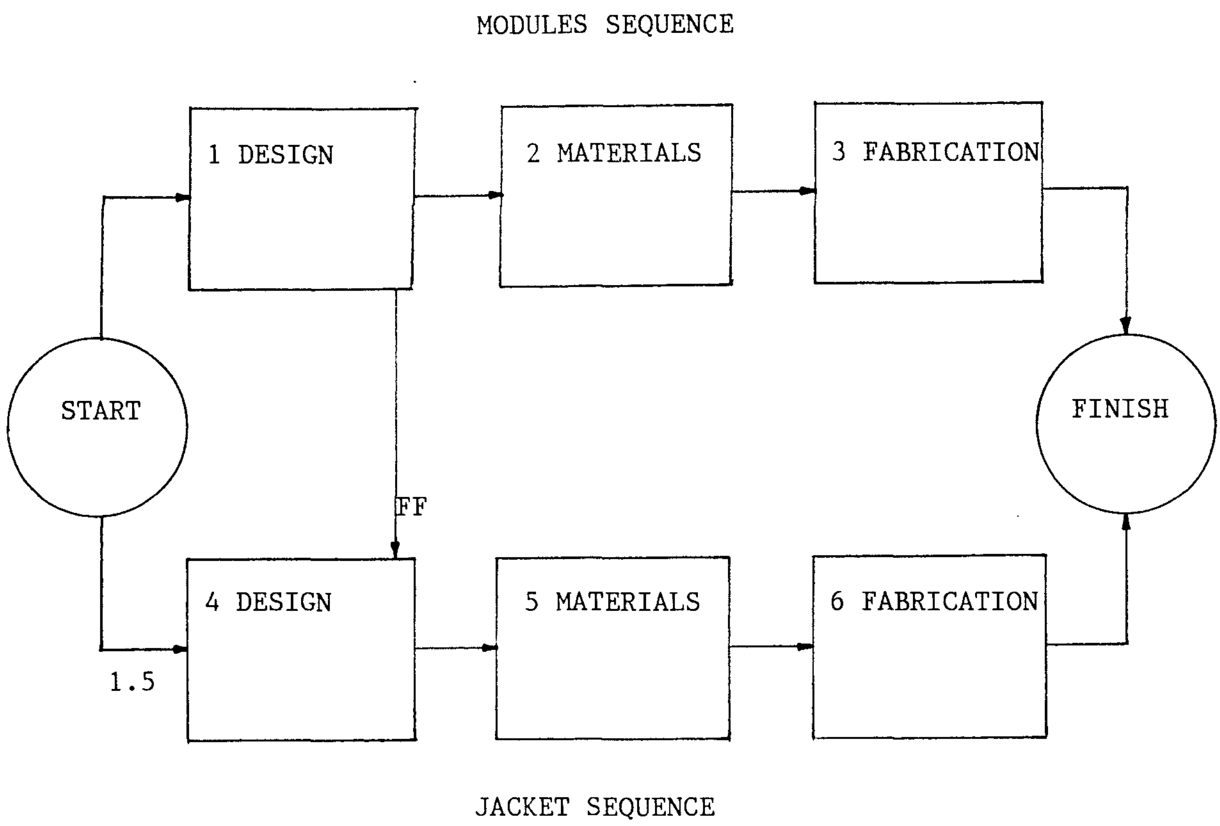


FIGURE 7.12: Revised project plan, with a delayed start to the Jacket sequence and a finish-to-finish precedence link.

In the following, the proposed procedure is applied to the network configuration of the above example. Figure 7.13 shows activity-on-arc diagram of Example 1. Notice that overlaps between activities are shown as activities but with negative duration times. 01 denotes overlap between Modules Design and Moduels Materials, 02 denotes overlap between Modules Materials and Modules Fabrication, and 03 denotes overlap between Jacket Materials and Jacket Fabrication. Table 7.10 shows durations of the activites. Expected values are also shown.

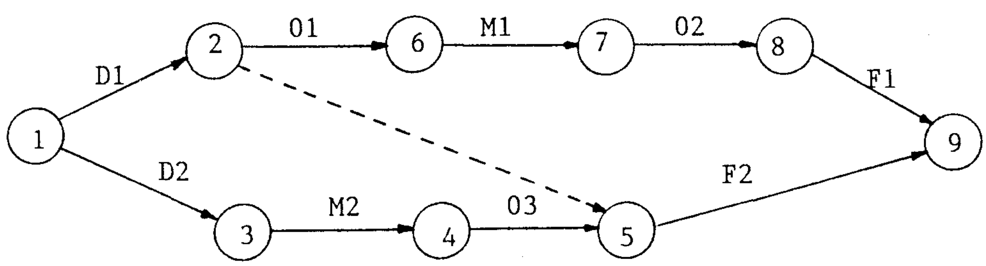


Figure 7.13

Table 7.10 Project activity durations.

D1 =	4	5		M1 =	2	3	
P =	0.6	0.4	E = 4.4	P =	0.3	0.7	E = 2.7
F1 =	3	4		D2 =	2	3	
P GIVEN D1 = 4	0.3	0.7	E = 3.7	P GIVEN D1 = 4	0.7	0.3	E = 2.3
5	0.8	0.2	3.2	5	0.2	0.8	2.8
P =	0.5	0.5	E = 3.5	P =	0.5	0.5	E = 2.5
M2 =	2	3		F2 =	4	5	
P	0.3	0.7	E = 2.7	P GIVEN D1 = 4	0.7	0.3	E = 4.3
				5	0.2	0.8	4.8
01 =	-1	0					
P =	0.5	0.5	E =-.5	P =	0.5	0.5	E = 4.5
02 =	-1	0		03 =	-1	0	
P =	0.5	0.5	E=-.5	P =	0.5	0.5	E=-.5

By fixing on the first realization time of D1, 4, changes the network of Figure 7.13 to that of Figure 7.14, and all path durations are independent.

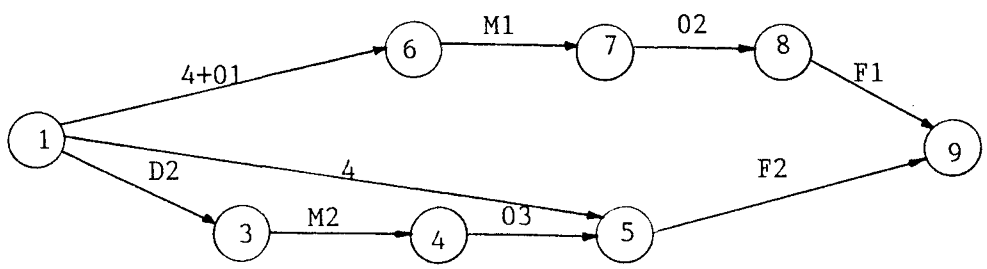


Figure 7.14

Following section shows calculation process.

M1's start time.

	P
3	0.5
4	0.5

M1's finish time.

	P
5	0.5x0.3 = 0.15
6	0.5x0.7+0.5x0.3 = 0.5
7	0.5x0.7 = 0.35

02's finish time.

	P
4	0.15x0.5 = 0.075
5	0.15x0.5+0.5x0.5 = 0.325
6	0.5x0.5+0.35x0.5 = 0.425
7	0.35x0.5 = 0.175

F1's finish time.

	P	CP
7	0.075x0.3 = 0.0225	0.0225
8	0.325x0.3+0.075x0.7 = 0.15	0.1725
9	0.425x0.3+0.325x0.7 = 0.355	0.5275
10	0.175x0.3+0.425x0.7 = 0.35	0.8775
11	0.175x0.7 = 0.1225	1.0

M2's finish time.

	P
4	0.7x0.3 = 0.21
5	0.7x0.7+0.3x0.3 = 0.58
6	0.3x0.7 = 0.21

03's finish time.

	P	CP
3	0.21x0.5 = 0.105	0.105
4	0.58x0.5+0.21x0.5 = 0.395	0.500
5	0.21x0.5+0.58x0.5 = 0.395	0.895
6	0.21x0.5 = 0.105	1.000

Duration time of D1=4.

	P	CP
4	1	1

Taking the maximum of D1=4 and 03's finish time yields F2's start time as follows:

F2's start time.

	P
4	1x0.5 = 0.5
5	1x0.895-1x0.5 = 0.395
6	1x1 - 1x0.895 = 0.105

F2's finish time.

	P		CP
8	0.5x0.7	=0.35	0.35
9	0.5x0.3+0.395x0.7	=0.4265	0.7765
10	0.395x0.3+0.105x0.7	=0.192	0.9685
11	0.105x0.3	=0.0315	1.0000

Taking the maximum of F1's finish time and F2's finish time yields the pdf of the project completion time given D_1=4.

Project completion time|D1=4.

	P
8	0.060375
9	0.3492287
10	0.440255
11	0.1501413
E =	9.6801626

By fixing on the second realization time of D1, 5, changes the network of Figure 7.13 to that of Figure 7.15, and all path durations are independent.

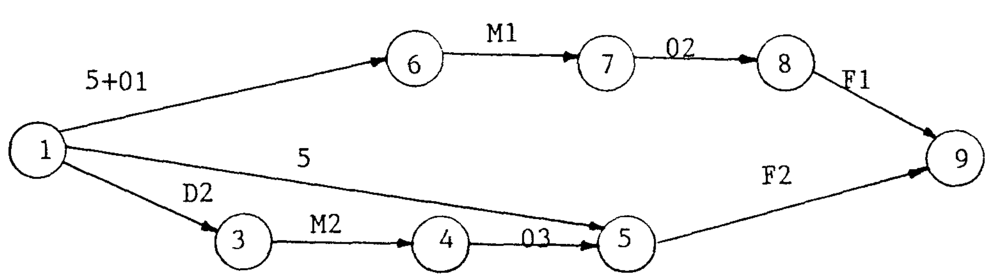


Figure 7.15

Following section shows calculation process.

M1's start time.

	P
4	0.5
5	0.5

M1's finish time.

	P
6	0.5x0.3 = 0.15
7	0.5x0.7+0.5x0.3 = 0.5
8	0.5x0.7 = 0.35

02's finish time.

	P
5	0.15x0.5 = 0.075
6	0.15x0.5+0.5x0.5 = 0.325
7	0.5x0.5+0.35x0.5 = 0.425
8	0.35x0.5 = 0.175

F1's finish time.

	P	CP
8	0.075x0.8 = 0.06	0.06
9	0.075x0.2+0.325+0.8 = 0.275	0.335
10	0.325x0.2+0.425x0.8 = 0.405	0.74
11	0.425x0.2+0.175x0.8 = 0.225	0.965
12	0.175x0.2 = 0.035	1.0

M2's finish time.

	P
4	0.2x0.3 = 0.06
5	0.2x0.7+0.8x0.3 = 0.38
6	0.8x0.7 = 0.56

03's finish time.

	P	CP
3	0.06x0.5 = 0.03	0.03
4	0.38x0.5+0.06x0.5 = 0.22	0.25
5	0.56x0.5+0.38x0.5 = 0.47	0.72
6	0.56x0.5 = 0.28	1.00

Taking the maximum of D1=5 and 03's finish time yields F2's start time as follows:

F2's start time.

	P
5	1x0.72 = 0.72
6	1x1 - 1x0.72 = 0.28

F2's finish time.

	P	CP
9	0.72x0.2 = 0.144	0.144
10	0.72x0.8+0.28x0.2 = 0.632	0.776
11	0.28x0.8 = 0.224	1.000

Taking the maximum of F1's finish time and F2's finish time yields the pdf of the project completion time given $D_1 = 5$.

Project completion time | D1=5.

	P
9	0.04824
10	0.526
11	0.39076
12	0.035
E =	10.4152

By deconditioning the pdfs of the project completion times given D1=4 and D1=5 unconditional pdf of the project completion time can be obtained as follows:

Unconditional pdf of project completion time.

	P	
8	0.060375x0.6	= 0.036225
9	0.3492287x0.6+0.04824x0.4	= 0.2288332
10	0.440255x0.6+0.526x0.4	= 0.474553
11	0.1501413x0.6+0.39076x0.4	= 0.2463887
12	0.035x0.4	= 0.014

Following table shows the rounded pdf of the project completion time, which is exactly the same as the pdf of the project completion time obtained using CIM procedure.

Rounded pdf of project completion time.

	P
8	.04
9	.23
10	.47
11	.25
12	.01
E =	10.0

In the following we apply the proposed procedure for criticality indices of Chapter 6 in order to approximate the criticality indices of activities of Figure 7.13. Given D1=4 the approximate criticality indices of activities of Figure 7.14 can be computed as follows:

Approximate criticality indices of activities F1 and F2 given D1=4.

	F1		F2	
8	0.15x0.35	= 0.0525	0.35x0.1725	= 0.060375
9	0.355x0.7765	= 0.2756575	0.4265x0.5275	= 0.2249787
10	0.35x0.9685	= 0.338975	0.192x0.8775	= 0.16848
11	0.1225x1	= 0.1225	0.0315x1	= 0.0315
CAP =		0.7896325		0.4853337

Approximate criticality indices of activities M2 and D1=4, given D1=4.

	M2	D1=4
4	0.395	0.5
5	0.395	
6	0.105	
CAP =	0.895	0.5
Normalized CAP =	0.641577	0.3584229
Nor.CAP x CAP(F2) =	0.3113789	0.1739547

Approximate Criticality Indices of Activities Given D1=4.

Activity	CAP
D1	0.9635872
M1	0.7896325
F1	0.7896325
D2	0.3113789
M2	0.3113789
F2	0.4853337

Given D1=5 the approximate criticality indices of activities of Figure 7.15 can be computed as follows:

Approximate criticality indices of activities F1 and F2 given D1=5.

	F1		F2	
9	0.275x0.144	= 0.0396	0.144x0.335	= 0.04824
10	0.405x0.776	= 0.31428	0.632x0.74	= 0.46768
11	0.225x1	= 0.225	0.224x0.965	= 0.21616
12	0.035x1	= 0.035		
CAP =		0.61388		0.73208

Approximate criticality indices of activities M2 and D1=5, given D1=5.

	M2	D1=5
5	0.47	0.72
6	0.28	
CAP =	0.75	0.72
Normalized CAP =	0.510204	0.4897959
Nor.CAP x CAP (F2) =	0.3735101	0.3585697

Approximate Criticality Indices of Activities Given D1=5.

Activity	CAP
D1	0.9724497
M1	0.61388
F1	0.61388
D2	0.3735101
M2	0.3735101
F2	0.73208

By deconditioning the conditional criticality indices and simple addition of corresponding criticality indices, the approximate criticaltiy index of each activity is obtained as follows:

Approximate Criticality Indices of Activities.

Activity	CAP
D1	0.9635872x0.6+0.9724497x0.4 = 0.9671321
M1	0.7896325x0.6+0.61388 x0.4 = 0.7189315
F1	0.7896325x0.6+0.61388 x0.4 = 0.7189315
D2	0.3113789x0.6+0.3735101x0.4 = 0.3362313
M2	0.3113789x0.6+0.3735101x0.4 = 0.3362313
F2	0.4853337x0.6+0.73208 x0.4 = 0.5840322

Following section presents the Algorithm.

THE ALGORITHM

STEP I - Check for the condition of conditioning operation. If duration time of activity, say i, is conditional upon duration time of one of the previous activities, say j, conditionalize by setting the duration time of activity j at its kth realization time.

STEP II - Reduce the network to its irreducible form using convolution and greatest operations.
If the network is reduced to an equivalent activity starting in node 1 and ending in node N, stop. The pdf of the duration time of this final activity is equal to F(t) given kth realization time of the activity which has been conditionalized in STEP I. Go to STEP I.
If the network is not completely reducible, calculate the indegree and outdegree of every node $i \neq N$, i.e. I(i) and O(i), then choose one activity 'a' such that 'a' has more than one successor while each of its successor has only 'a' as a predecessor.

STEP III - Conditionalize by setting the chosen activity 'a' at its kth realization time T_a^k; this is done by deleting 'a', adding T_a^k to the implied precedence of activities in the conditionalized network.

STEP IV - Decondition the df of the final activity.

STEP V - Determine the df of the project completion time, mean and standard deviation.

Example 2:

Consider again the example problem of Chapter 3, recall that the example problem consists of two stages, Design and Construction as shown in Figure 3.7 with three time estimates for each stage as shown in Tables 3.3 and 3.4.

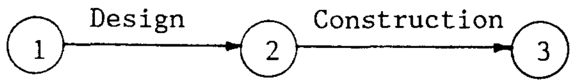


Figure 3.7

Table 3.3: Design distribution, D1 months.

D1	P(D1)
2	0.3
3	0.5
4	0.2

Table 3.4: Construction distributon, D2 months.

D1	P(D2)
6	0.3
7	0.6
8	0.1

The pdf of the project completion time was found as shown in Table 3.6, with mean equal to 9.7 months and standard deviation equal to 0.922 months.

Table 3.6: Project completion time = Design plus Construction.

	P
8	0.09
9	0.33
10	0.39
11	0.17
12	0.02

Let us assume that the duration time of Construction is statistically dependent upon the duration time of Design as shown in Table 7.11.

Table 7.11: Construction distribution, D2 months.

	D2 =	6	7	8
P Given D1 =	2	0.1	0.7	0.2
	3	0.5	0.5	0
	4	0.1	0.7	0.2
Unconditional P =		0.3	0.6	0.1

Using the proposed procedure, project completion time can be computed as follows:

Given D1=2, project completion time is shown in Table 7.12.

Table 7.12: Project completion time given D1=2.

	P	CP
8	0.1	0.1
9	0.7	0.8
10	0.2	1.0

Given D1=3, project completion time is shown in Table 7.13.

Table 7.13: Project completion time given D1=3.

	P	CP
9	0.5	0.5
10	0.5	1.0
11	0.0	1.0

Finally given D1=4, project completion time is shown in Table 7.14.

Table 7.14: Project completion time given D1=4.

	P	CP
10	0.1	0.1
11	0.7	0.8
12	0.2	1.0

By deconditioning the pdfs of Tables 7.12 through 7.14, and simple addition of corresponding probabilities unconditional pdf of the project completion time can be computed as shown in Table 7.15.

Table 7.15: Unconditional pdf of the project completion time.

	P	
8	0.1x0.3	= 0.03
9	0.7x0.3+0.5x0.5	= 0.46
10	0.2x0.3+0.5x0.5+0.1x0.2	= 0.33
11	0.0x0.5+0.7x0.2	= 0.14
12	0.2x0.2	= 0.04
E =		9.7 and σ= 0.888

Notice that although the unconditional probability of Table 7.11 is exactly the same as probability of Construction distribution of Table 3.4, but pdfs of the project completion times of Tables 3.6 and 7.15 are not the same, it is because of the statistical dependence relationship between duration times of Design and Construction. Hence, the mean values of the project completion times are the same but the standard deviations are not.

Now let us consider the two similar projects of the example problem as one integrated project as shown in Figure 3.8. Assume that the duration time of Construction in each sequence is conditional upon the duration time of Design as shown in Table 7.11.
Using the proposed procedure, project completion time can be computed as shown in Table 7.16.

Table 7.16: Project completion time.

	P	
8	0.003	
9	0.314	
10	0.485	
11	0.126	
12	0.072	
E =	9.95	σ = 0.8588

Comparing with the project completion time obtained in Chapter 3 for similar project without statistical dependence between duration times of Design and Construction for each sequence as shown in Table 3.8, we can see that not only the standard deviations of these pdfs are not the same but also the mean values are not the same. It is because of the effects of the statistical dependence relationships between duration times of Design and Construction and also because of the effect of greatest operation over the duration times of two parallel paths.

Table 3.8: Project completion time.

	P	
8	0.0081	
9	0.1683	
10	0.4797	
11	0.3043	
12	0.0396	
E =	10.199	σ = 0.7898

Proposed procedure for Criticality Indices of activities in PERT networks with Statistical and Structural dependence relationships between activities is exactly the same as proposed procedure of Chapter 6.

Example 3:

Consider again, the Example 2 of Chapter 6 as shown in Figure 6.2. Table 6.1 shows the durations of the activities. Expected values and variances are also shown.

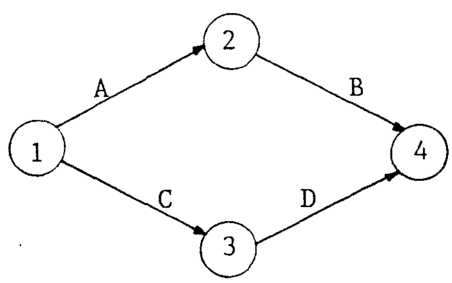


Figure 6.2

Table 6.1: Project activity durations.

1.A	X_A =	4	5
	P =	0.6	0.4
	E = 4.4 ,	σ^2 = 0.24	
2.B	X_B =	6	7
	P =	0.4	0.6
	E = 6.6 ,	σ^2 = 0.24	
3.C	X_C =	1	10
	P =	0.1	0.9
	E = 9.1 ,	σ^2 = 7.29	
4.D	X_D =	1	2
	P =	0.2	0.8
	E = 1.8 ,	σ^2 = 0.16	

Using the proposed procedure of Chapter 6, the approximate criticality indices of activites were found to be, CAP(A)=CAP(B)=0.4096 and CAP(C)=CAP(D)=0.8568, and also the Normalized Criticality Indices were found to be, NCAP(A)=NCAP(B)=0.323 and NCAP(C)=NCAP(D)=0.677.

Let us assume that the duration time of activity B is statistically dependent upon the duration time of activity A as shown in Table 7.17.

Table 7.17: Duration time of activity B.

	X_B	6	7
P Given X_A	= 4	0.2	0.8
	5	0.7	0.3
Unconditional P		0.4	0.6

Notice that although network configuration of Figure 6.2 is subject to series-parallel reduction but since the duration time of activity B is conditional upon the duration time of activity A the first step in order to find the pdf and approximate criticality indices of activities is to conditionalize activity A at its different realization times. Therefore, by conditioning at first realization time of activity A, 4, changes the network of Figure 6.2 to the network of Figure 7.17 and two paths would be statistically and structurally independent. Convoluting 4 and B gives duration time of E and convoluting C and D gives duration time of F as shown in Tables 7.18 and 7.19 respectively.

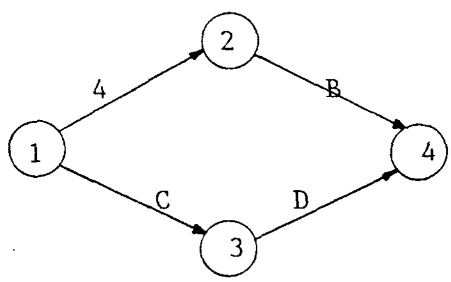


Figure 7.17

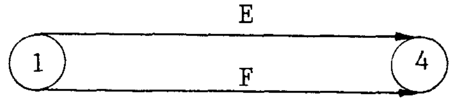


Figure 7.18

Table 7.18: Duration time of E.

	P	CP
10	0.2	0.2
11	0.8	1.0

Table 7.19: Duration time of F.

	P	CP
2	0.02	0.02
3	0.08	0.1
11	0.18	0.28
12	0.72	1.0

Table 7.20 shows Project finish time given A=4.

Table 7.20: Project finish time given A=4.

	P
10	0.02
11	0.26
12	0.72
E =	11.7

Table 7.21 shows criticality indices of activities given A=4.

Table 7.21: Criticality indices of activites given A=4.

	C=D	A=B
10		0.02
11	0.18	0.224
12	0.72	
CAP =	0.90	0.244

By conditioning at second realization time of A,5, changes the network of Figure 6.2 to that of Figure 7.19. Convoluting 5 and B gives duration time of G as shown in Table 7.22.

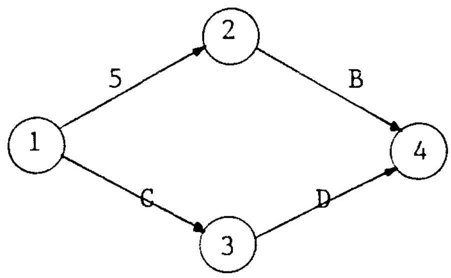


Figure 7.19

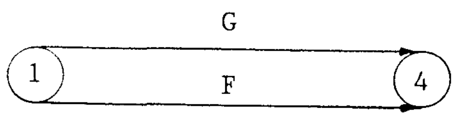


Figure 7.20

Table 7.22: Duration time of G.

	P	CP
11	0.7	0.7
12	0.3	1.0

Taking the maximum of the duration times of F and G gives project finish time given A=5 as shown in Table 7.23.

Table 7.23: Project finish time given A=5.

	P
11	0.196
12	0.804
E =	11.804

Table 7.24 shows criticality indices of activities given A=5.

Table 7.24: Criticality indices of activites given A=5.

	C=D	A=B
11	0.126	0.196
12	0.72	0.3
CAP =	0.846	0.496

By deconditioning the pdfs of the project finish times of Tables 7.20 and 7.23, unconditional pdf of project finish time can be obtained as shown in Table 7.25, and also by deconditioning criticality indices of Tables 7.21 and 7.24, criticality indices of activities can be obtained as shown in Table 7.26.

Table 7.25: Unconditional project finish time.

	P
10	0.02x0.6 = 0.012
11	0.26x0.6+0.196x0.4 = 0.2344
12	0.72x0.6+0.804x0.4 = 0.7536
E =	11.7416

Table 7.26: Criticality indices of activities.

Activity	CAP	Normalized CAP
A	0.3448	0.2819
B	0.3448	0.2819
C	0.8784	0.7181
D	0.8784	0.7181

Unconditional project finish time and criticality indices shown in Tables 7.25 and 7.26 are exact. These values also can be obtained with complete enumeration as shown in tree diagram of Figure 7.21.

Notice that statistical dependence between duration times of activites A and B has changed the pdf of project completion time from Table 7.27 which is obtained from Figure 6.4, to that of Table 7.25, and also has changed the criticality indices of activities from Table 6.4 to that of Table 7.26.

Table 7.27: Project finish time.

10	0.024
11	0.1888
12	0.7872
E =	11.7632

Realization times of activities and probability of each realization.								Project realization time and critical path.		Probability of each realization.
A	P	B	P	C	P	D	P	AB	CD	
4	0.6	6	0.2	1	0.1	1	0.2	10		0.0024
						2	0.8	10		0.0096
				10	0.9	1	0.2		11	0.0216
						2	0.8		12	0.0864
		7	0.8	1	0.1	1	0.2	11		0.0096
						2	0.8	11		0.0384
				10	0.9	1	0.2	11	11	0.0864
						2	0.8		12	0.3456
5	0.4	6	0.7	1	0.1	1	0.2	11		0.0056
						2	0.8	11		0.0224
				10	0.9	1	0.2	11	11	0.0504
						2	0.8		12	0.2016
		7	0.3	1	0.1	1	0.2	12		0.0024
						2	0.8	12		0.0096
				10	0.9	1	0.2	12		0.0216
						2	0.8	12	12	0.0864

Figure 7.21

Example 4:

Consider again example 3 of Chapter 6 assume duration time of activity C is conditional upon duration time of activity A as shown in Table 7.28. Notice that unconditional probability of C in Table 7.28 is the same as probability distribution of C of example 3 of Chapter 6.

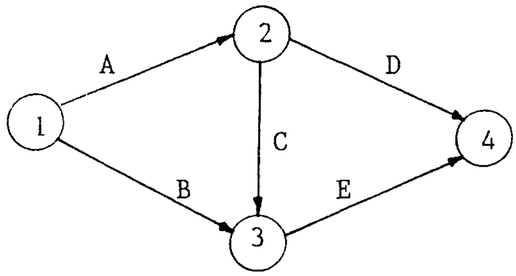


Figure 3.10

Table 7.28: Duration time of activity C.

	X_C =	4	6
P GIVEN X_A = 3		0.2	0.8
8		0.7	0.3
Unconditional P		0.3	0.7

Using the proposed procedure, the pdf of project completion time and the approximate criticality indices of the activities can be computed. Project completion time and approximate criticality indices are shown in Tables 7.29 and 7.30 respectively.

Table 7.29: Project completion time.

	P
8	0.048
9	0.048
10	0.352
11	0.352
13	0.07
14	0.07
15	0.03
16	0.03
E =	11.028

Table 7.30: Normalized values of the approximate criticality indices.

Activity	Normalized CAP
A	0.759
B	0.241
C	0.7475
D	0.0115
E	0.9885

Using tree diagram of Figure 7.22, the project completion time and the exact criticality indices of activities can be determined as shown in Tables 7.31 and 7.32 respectively.

Table 7.31: project completion time obtained from tree diagram .

	P
8	0.048
9	0.048
10	0.352
11	0.352
13	0.07
14	0.07
15	0.03
16	0.03
E =	11.028

Table 7.32: Exact criticality indices of activities.

Activity	CA	Normalized CA
A	0.9478	0.7476
B	0.32	0.2524
C	0.936	0.7383
D	0.0118	0.0093
E	1.256	0.9907

Realization times of activities and probability of each realization.										Project realization time and critical path.			Probability of each realization.
A	P	B	P	C	P	D	P	E	P	1-2-4	1-2-3-4	1-3-4	
3	0.8	6	0.6	4	0.2	4	0.9	1	0.5		8		0.0432
								2	0.5		9		0.0432
						5	0.1	1	0.5	8	8		0.0048
								2	0.5		9		0.0048
				6	0.8	4	0.9	1	0.5		10		0.1728
								2	0.5		11		0.1728
						5	0.1	1	0.5		10		0.0192
								2	0.5		11		0.0192
		9	0.4	4	0.2	4	0.9	1	0.5			10	0.0288
								2	0.5			11	0.0288
						5	0.1	1	0.5			10	0.0032
								2	0.5			11	0.0032
				6	0.8	4	0.9	1	0.5		10	10	0.1152
								2	0.5		11	11	0.1152
						5	0.1	1	0.5		10		0.0128
								2	0.5		11		0.0128
8	0.2	6	0.6	4	0.7	4	0.9	1	0.5		13		0.0375
								2	0.5		14		0.0375
						5	0.1	1	0.5	13	13		0.0042
								2	0.5		14		0.0042
				6	0.3	4	0.9	1	0.5		15		0.0162
								2	0.5		16		0.0162
						5	0.1	1	0.5		15		0.0018
								2	0.5		16		0.0018
		9	0.4	4	0.7	4	0.9	1	0.5		13		0.0252
								2	0.5		14		0.0252
						5	0.1	1	0.5	13	13		0.0028
								2	0.5		14		0.0028
				6	0.3	4	0.9	1	0.5		15		0.0108
								2	0.5		16		0.0108
						5	0.1	1	0.5		15		0.0012
								2	0.5		16		0.0012

Figure 7.22

In order to determine the goodness of proposed procedure, in the following we calculate the correlation coefficient between normalized values of exact criticality indices of Table 7.32 and normalized values of criticality indices obtained using proposed procedure of Table 7.30.

Let X denote the normalized values of criticality indices and Y denote the normalized values of criticality indices obtained using proposed procedure.

Activity	(NCA) X	(NCAP) Y	$x=X-\bar{X}$	$y=Y-\bar{Y}$	x^2	y^2	xy
A	0.7476	0.759	0.19994	0.2095	0.039976	.0438902	0.0418874
B	0.2524	0.241	-.29526	-.3085	0.0871784	.0951722	0.0910877
C	0.7383	0.7475	0.19064	0.198	0.0363436	.039204	0.0377467
D	0.0093	0.0115	-.53836	-.538	0.2898314	.289444	0.2896376
E	0.9907	0.9885	0.44304	0.439	0.1962844	.192721	0.1944945
	$\Sigma X=$ 2.7383 $\bar{X}=.54766$	$\Sigma Y=$ 2.7475 $\bar{Y}=.5495$			$\Sigma x^2=$ 0.6496138	$\Sigma y^2=$ .6604314	$\Sigma xy=$ 0.6548539

Correlation Coefficient: $r_{X,Y} = \frac{\Sigma(xy)}{\sqrt{(\Sigma x^2)(\Sigma y^2)}} = \frac{0.6548539}{\sqrt{(.6496138)(.6604314)}}$

$= 0.9997766.$

Example 5:

Consider the Wheatstone bridge of Figure 7.23 assume duration times of activities C and D are conditional upon duration time of activity B. Table 7.33 shows duration times of activities.

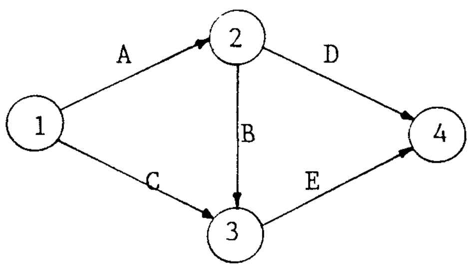


Figure 7.23

Table 7.33: Project activity durations.

X_A	=	4	5	X_B	=	1	2
P	=	0.6	0.4	P	=	0.3	0.7
X_C	=	5	6	X_D	=	2	3
P GIVEN X_B	= 1	0.3	0.7	P GIVEN X_B	= 1	0.9	0.1
	2	0.8	0.3		2	0.4	0.6
X_E	=	1	2				
P	=	0.3	0.7				

Using the proposed procedure, the pdf of project completion time and the approximate criticality indices of activities can be computed. Project completion time and approximate criticality indices are shown in Tables 7.34 and 7.35 respectively.

Table 7.34: Project completion time.

	P
6	0.01458
7	0.23562
8	0.5538
9	0.196
E =	7.93122

Table 7.35: Normalized value of approximate criticality indices.

Activity	Normalized CAP
A	0.781
B	0.62
C	0.219
D	0.161
E	0.839

Using tree diagram of Figure 7.24, project completion time and exact criticality indices of activities can be determined as shown in Tables 7.36 and 7.37 respectively.

Realization times of activities and probability of each realization.										Project realization time and critical path.			Probability of each realization.
A	P	B	P	C	P	D	P	E	P	1-2-4	1-2-3-4	1-3-4	
4	0.6	1	0.3	5	0.3	2	0.9	1	0.3	6	6	6	0.01458
								2	0.7		7	7	0.03402
						3	0.1	1	0.3	7			0.00162
								2	0.7	7	7	7	0.00378
				6	0.7	2	0.9	1	0.3			7	0.03402
								2	0.7			8	0.07938
						3	0.1	1	0.3	7		7	0.00378
								2	0.7			8	0.00882
		2	0.7	5	0.8	2	0.4	1	0.3		7		0.04032
								2	0.7		8		0.09408
						3	0.6	1	0.3	7	7		0.06048
								2	0.7		8		0.14112
				6	0.2	2	0.4	1	0.3		7	7	0.01008
								2	0.7		8	8	0.02352
						3	0.6	1	0.3	7	7	7	0.01512
								2	0.7		8	8	0.03528
5	0.4	1	0.3	5	0.3	2	0.9	1	0.3	7	7		0.00972
								2	0.7		8		0.02268
						3	0.1	1	0.3	8			0.00108
								2	0.7	8	8		0.00252
				6	0.7	2	0.9	1	0.3	7	7	7	0.02268
								2	0.7		8	8	0.05292
						3	0.1	1	0.3	8			0.00252
								2	0.7	8	8	8	0.00588
		2	0.7	5	0.8	2	0.4	1	0.3		8		0.02688
								2	0.7		9		0.06272
						3	0.6	1	0.3	8	8		0.04032
								2	0.7		9		0.09408
				6	0.2	2	0.4	1	0.3		8		0.00672
								2	0.7		9		0.01568
						3	0.6	1	0.3	8	8		0.01008
								2	0.7		9		0.02352

Figure 7.24

Table 7.36: Project completion time obtained from tree diagram .

	P
6	0.01458
7	0.23562
8	0.5538
9	0.196
E =	7.93122

Table 7.37: Normalized criticality indices.

Activity	Normalized CA
A	0.7556
B	0.6176
C	0.2444
D	0.1380
E	0.8620

In the following the correlation coefficient between normalized values of exact criticality indices and criticality indices obtained using proposed procedure is determined.

Activity	Exact CA X	Estimate CA Y	$x=X-\bar{X}$	$y=Y-\bar{Y}$	x^2	y^2	xy
A	0.7556	0.781	0.23208	0.257	0.0538611	0.066049	0.0596445
B	0.6176	0.62	0.09408	0.096	0.008851	0.009216	0.0090316
C	0.2444	0.219	-.27912	-.305	0.0779079	0.093025	0.0851316
D	0.1380	0.161	-.38552	-.363	0.1486256	0.131769	0.1399437
E	0.8620	0.839	0.33848	0.315	0.1145687	0.099225	0.1066212
	$\Sigma X=$ 2.6176 $\bar{X}=0.52352$	$\Sigma Y=$ 2.62 $\bar{Y}=0.524$			$\Sigma x^2=$ 0.4038143	$\Sigma y^2=$ 0.399284	$\Sigma xy=$ 0.4003726

Correlation Coefficient: $r_{X,Y} = \frac{\Sigma(xy)}{\sqrt{(\Sigma x^2)(\Sigma y^2)}} = \frac{(.4003726)}{\sqrt{(.4038143)(.399284)}}$

$= 0.9970862.$

Example 6:

Consider network configuration of Figure 7.25, assume duration time of activity D is conditional upon duration time of activity A. Table 7.38 shows duration times of activities.

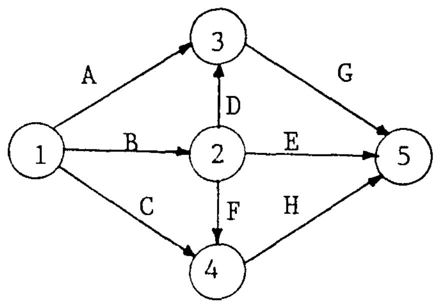


Figure 7.25

Table 7.38: Project activity durations.

X_A	=	2	3		X_B		=	1	2
P	=	0.5	0.5		P		=	0.5	0.5
X_C	=	2	3		X_D		=	1	2
				P GIVEN	X_A	=	2	0.7	0.3
P	=	0.5	0.5				3	0.2	0.8
X_E	=	1	2		X_F		=	1	2
P	=	0.5	0.5		P		=	0.5	0.5
X_G	=	1	2		X_H		=	1	2
P	=	0.5	0.5		P		=	0.5	0.5

Project completion time and approximate criticality indices obtained using proposed procedure are shown in Tables 7.39 and 7.40 respectively. Project completion time and exact criticality indices using complete enumeration are shown in Tables 7.41 and 7.42 respectively.

Table 7.39: Project completion time.

	P
3	0.0109375
4	0.228125
5	0.5328125
6	0.228125
E =	4.978125

Table 7.40: Normalized values of criticality indices using proposed procedure.

Activity	CAP
A	0.1722
B	0.6494
C	0.1784
D	0.3142
E	0.0139
F	0.3213
G	0.4864
H	0.4997

Table 7.41: project completion time.

	P
3	0.010937
4	0.2281234
5	0.5328092
6	0.2281236
E =	4.9780922

Table 7.42: Normalized values of exact criticality indices using complete enumeration.

Activity	CA
A	0.1567
B	0.6715
C	0.1718
D	0.3427
E	0.0104
F	0.3184
G	0.4994
H	0.4902

In the following the correlation coefficient between normalized values of exact criticality indices and criticality indices obtained using proposed procedure is determined.

Activity	Exact CA X	Estimate CA Y	$x=X-\bar{X}$	$y=Y-\bar{Y}$	x^2	y^2	xy
A	0.1567	0.1722	-.1759375	-.1572375	0.30954	.030954	.0276639
B	0.6715	0.6494	.3388625	.3199625	0.1148277	.102376	.1084232
C	0.1718	0.1784	-.1608375	-.1510375	0.0258687	.0228123	.0242924
D	0.3427	0.3142	.0100625	-.0152375	0.0001012	.0002321	-.0001533
E	0.0104	0.0139	-.3222375	-.3155375	0.103837	.0995639	.101678
F	0.3184	0.3213	-.0142375	-.0081375	0.0002027	.0000662	.0001158
G	0.4994	0.4864	.1667625	.1569625	0.0278097	.0246372	.0261754
H	0.4902	0.4997	.1575625	.1702625	0.0248259	.0289893	.0268269
	$\Sigma X=$ 2.6611 $\bar{X}=$ .3326375	$\Sigma Y=$ 2.6355 $\bar{Y}=$ .3294375			$\Sigma X^2=$ 0.3284269	$\Sigma y^2=$ .3034006	$\Sigma xy=$ .3150223

Correlation Coefficient: $r_{X,Y} = \frac{\Sigma(xy)}{\sqrt{(\Sigma x^2)(\Sigma y^2)}} = \frac{(0.3150223)}{\sqrt{(.3284269)(.3034006)}}$

$= 0.9979614.$

Example 7:

Consider network configuration of Figure 7.26, assume duration times of activities C and G are conditional upon duration time of activity A, and duration times of activities D and H are conditional upon duration time of activity B as shown in Table 7.43.

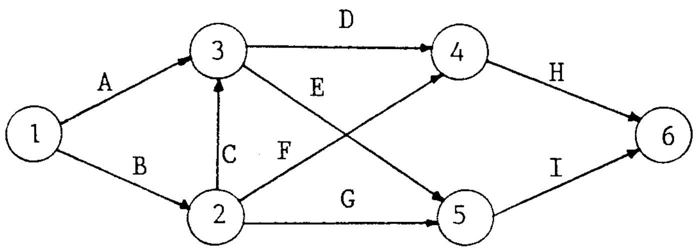


Figure 7.26

Table 7.43: Project activity durations.

X_A =		2	8	X_B =		3	4
P =		0.2	0.8	P =		0.4	0.6
X_C =		3	4	X_D =		3	5
P GIVEN X_A =	2	0.7	0.3	P GIVEN X_B =	3	0.2	0.8
	8	0.2	0.8		5	0.6	0.4
X_E =		4	6	X_F =		6	8
P =		0.7	0.3	P =		0.4	0.6
X_G =		7	8	X_H =		4	6
P GIVEN X_A =	2	0.2	0.8	P GIVEN X_B =	3	0.4	0.6
	8	0.7	0.3		4	0.9	0.1
X_I =		4	6				
P =		0.8	0.2				

Project completion time and approximate criticality indices obtained using proposed procedure are shown in Tables 7.44 and 7.45 respectively.

Table 7.44: Project completion time using proposed procedure.

	P
14	0.0002007
15	0.0169558
16	0.2234766
17	0.2481867
18	0.290754
19	0.1702656
20	0.05016
E =	17.523764

Table 7.45: Normalized values of criticality indices using proposed procedure.

Activity	CAP
A	0.469022
B	0.5309779
C	0.2794486
D	0.3590374
E	0.3894332
F	0.1344112
G	0.1171173
H	0.4934492
I	0.5065511

Project completion time and criticality indices using complete enumeration are shown in Tables 7.46 and 7.47 respectively.

Table 7.46: Project completion time.

	P
14	0.0002007
15	0.0169537
16	0.223465
17	0.2481489
18	0.2906766
19	0.1702004
20	0.0500819
E =	17.51871

Table 7.47: Normalized values of exact criticality indices using complete enumeration.

Activity	CA
A	0.4797383
B	0.5202616
C	0.3198927
D	0.315716
E	0.4837151
F	0.1126227
G	0.0877461
H	0.4283387
I	0.5716612

In the following the correlation coefficient between normalized values of exact criticality indices and criticality indices obtained using proposed procedure is determined.

Activity	Exact CA X	Estimate CA Y	$x=X-\bar{X}$	$y=Y-\bar{Y}$	x^2	y^2	xy
A	0.4797383	0.469022	0.1108614	0.1046389	0.0122902	0.0109492	0.0116004
B	0.5202616	0.5309779	0.1513847	0.1665948	0.0229173	0.0277538	0.0252199
C	0.3198927	0.2794486	-.0489842	-.0849345	0.0023994	0.0072138	0.0041604
D	0.315716	0.3590374	-.0531609	-.0053457	0.002826	0.0000285	0.0002841
E	0.4839151	0.3894332	0.1150382	0.0250501	0.0132337	0.0006275	0.0028817
F	0.1126227	0.1344112	-.2562542	-.2299719	0.0656662	0.052887	0.0589312
G	0.0877461	0.1171173	-.2811308	-.2472658	0.0790345	0.0611403	0.069514
H	0.4283387	0.4934492	0.0594618	0.1290661	0.0035357	0.016658	0.0076745
I	0.5716612	0.5065511	0.2027843	0.142168	0.0411214	0.0202117	0.0288294
	$\Sigma X=$ 3.3198924 $\bar{X}=$ .3688769	$\Sigma Y=$ 3.2794479 $\bar{Y}=$.3643831			$\Sigma x^2=$ 0.2430244	$\Sigma y^2=$ 0.1974698	$\Sigma xy=$ 0.2090956

Correlation Coefficient: $r_{X,Y} = \dfrac{\Sigma(xy)}{\sqrt{(\Sigma x^2)(\Sigma y^2)}} = \dfrac{(.2090956)}{\sqrt{(.2430244)(.1974698)}}$

$= 0.9544872.$

SUMMARY AND CONCLUSIONS

This chapter in the first part has provided an illustration of the importance of structural dependence relationships in the context of an offshore construction schedule. It has demonstrated how such relationships can interact with statistical dependence relationships to produce important effects which cannot be detected and understood usign simple expected value calculations. These effects are considered in a CIM framework.

In the second part proposed procedure to determine probability distribution function of the project completion time in PERT networks with statistical and structural dependence relationships between activities when duration times of activities are discretly distributed is presented. This procedure provides the exact pdf of project completion times.

Finally, proposed procedure of Chapter 6 to estimate criticality indices of activities and paths is applied to the PERT networks with statistical and structural dependence relationships between activities. The correlation coefficients between criticality indices obtained using proposed procedure and exact criticality indices obtained using complete enumeration showed that the estimate criticality indices are very close to the exact criticality indices.

CHAPTER 8: DISCRETIZING CONTINUOUS DISTRIBUTIONS

INTRODUCTION

As mentioned in Chapter 4 for PERT networks, when the duration times of activities are continuously distributed the first step in proposed procedure is discretizing the continuous distributions. This chapter looks briefly at different discretizing methods and the accuracy of the most efficient method is examined through simple examples.

Discretizing methods

Discretizing is done by determining a set of ordered pairs denoting F(a). The cardinality of F(a) dependes on the desired accuracy of the discretization. Using the closeness to the exact values of the first five moments as a criterion to determine the count of points in the pdf F(a), denoted by CF(a); three methods have been proposed and tried by Dodin(1980). The most efficient in terms of accuracy and computer time is a hybrid of methods 2 and 3 described below. These three methods are:

1- The 2m Method: If we decide that CF(a)=m for any activity a with continuous distribution, then from the definition of F(a) we have 2m unknowns; m realizations and the corresponding m probabilities. The first 2m moments of the continuous distribution can be used to construct the following system of 2m nonlinear equations:

$$\sum_{k=1}^{m} x_k^n p(x_k) = e_n, \quad \text{for} \quad n=0,1,2,\ldots\ldots,2m-1 \qquad (8.1)$$

where

$$e_n = E(x^n) = \int_{-\infty}^{+\infty} x^n \, dF\,(x)dx, \text{ the } n^{th} \text{ moment.}$$

In a matrix form we have:

$$VP=E$$

where V is the Vandermonde matrix of dimension 2mxm and P is the probability vector with m components, and E is the vector of the 2m moments. Two methods have been tried to solve this system of nonlinear equations, but neither succeeded for m>8. These two methods are:

(a) Brown Method: Which is documented in IMSL(42) library under the name ZSYSTM. Starting with an initial solution ZSYSTM is supposed to converge to a solution within e from a feasible solution. However, many runs to different values of m and different initial solutions proved that ZSYSTM was not converging, and often terminated because of a singularity that occurred in the iteration, due mainly to the nature of Vandermonde matrix V. Two other packages SBROWN and SNGINT developed by the Argonne National Labratory have been tried; neither succeeded in solving the above system. The failure led to the search for other methods. The following method was successful in solving the above system, but only for small value of m(⩽8).

(b) Gaussian Quadrature (39): To solve the above system of nonlinear equations the procedure is as follows:

(i) Determine the sample polynomial

$$\prod(x) = \sum_{k=0}^{m} C_k \ x^k,$$

The coefficient C_k are determined uniquely using the following system of linear equations after setting $C_m=1$.

$$C_0e_0+C_1e_1+C_2e_2+\ldots+C_{m-1}e_{m-1}+e_m=0$$

$$C_0e_1+C_1e_2+C_2e_3+\ldots+C_{m-1}e_m+e_{m+1}=0$$

$$C_0e_2+C_1e_3+C_2e_4+\ldots+C_{m-1}e_{m+1}+e_{m+2}=0$$

$$\vdots$$

$$C_0e_{m-1}+C_1e_m+C_2e_{m+1}+\ldots+C_{m-1}e_{2m-2}+e_{2m-1}=0$$

(ii) The set of realizations (discrete points) are determined by solving the polynomial

$$\sum_{k=0}^{m} C_k x^k = 0$$

where all m solutions are simple and real (since $0 \leqslant u \leqslant x \leqslant v$).

(iii) The corresponding probabilities are determined by substituting for x_k in the first m equations of (8.1) then solve uniquely for $p(x_k)$.

This algorithm was programmed and tested; it works for $m \leqslant 8$. It is not recommended for discretization since it is very sensitive to the values of $E(x^n)$, and requires the solution of two systems each of m linear equations, and the solution of a polynomials of the m^{th} degree, each time it is used to approximate a distribution. Furthermore, the user would never know when the procedure will "blow-up".

2- Using Equal Distances: Based on the distribution of the activity, the minimum and maximum realization values u and v can be determined; then by the use of an appropriate spacing Δ, depending on the desired accuracy, the range (v-u) can be subdivided into equal intervals. In this case

$$x_k = u + \Delta(k-1) \forall k = 1,2,3\ldots,m \text{ where } m = [\tfrac{v-u}{\Delta}]$$

As a rule in Dodin's (1980) study the minimum and maximum realizations are determined such that

$$p(x<u) = p(x>v) = 0.0005$$

The corresponding probabiliies are determined according to

$$p(x_k) = \int_{x_k - \Delta/2}^{x_k + \Delta/2} dF(x)dx \quad \text{for each } k = 2,3,4\ldots,m-1,$$

and $p(x_1) = p(u) = \int_{-\infty}^{u+\Delta/2} dF(x)dx$ and $p(x_m) = p(v) = \int_{v-\Delta/2}^{\infty} dF(x)dx.$

For a small Δ (large m) the determination of the probability can be approximated by $p(y_k)=\Delta f(y_k)$ for each $k=1,2,\ldots,m$ where y_k is the center of the k^{th} interval, i.e., $y_k=u+\Delta(k-1)+\Delta/2$ and $f(y)=dF(y)$. This approach, as it appears in Figure 8.1, treates all points in the range of the r.v. in a uniform fashion, i.e., we partition the range into equal distances. This approach is very convenient for some distributions such as the uniform and the triangular distributions, and some other distributions when their skewness or peaks are not very acute. If sharp peaks are present, such as the case in the exponential distribution with large paramenter α, or the normal distribution with small σ, then very small value of Δ are used to minimize the errors of approximation. This drawback led to the use of the following alternative.

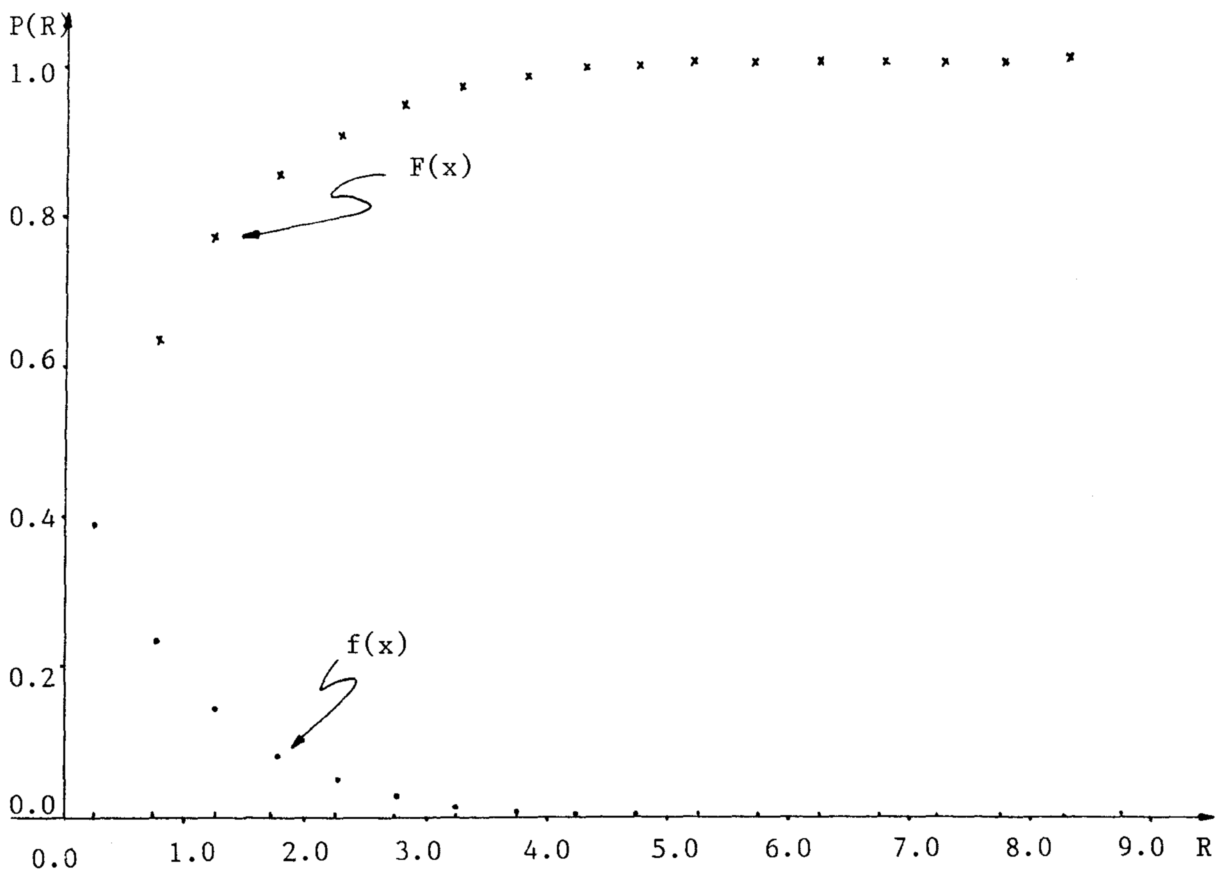


Figure 8.1

Equal Distance Discretization for the Exponential Distribution.

3- Using Equal Probabilities: Here, again, u and v are determined in the same way as in Method 2. Then F(a) is determined according to:

$$\Delta = p(x_k) - 1/m \text{ for a given } m,$$

and

$$x_k = F^{-1}\left(\sum_{j=1}^{k} p(x_j) - \Delta/2\right)$$

using the continuous distribution function. This scheme is suitable for all distributions under consideration (i.e. Uniform, Exponential, Triangular, Normal, Gamma, and Beta). However, it may not be easy (or it can be time consuming) to invert some of the distribution functions. Hence its use is limited to the exponential distributions, where it is needed the most, while the method of equal distance is used for the remaining five distributions. Figures 8.1 and 8.2 illustrate methods 2 and 3 respectively for the exponential distribution with parameter $\alpha=1$.

Figure 8.2 shows that method 3 responds to the peak of the pdf by taking more realizations, where Figure 8.1 shows that method 2 does not respond to peaks.

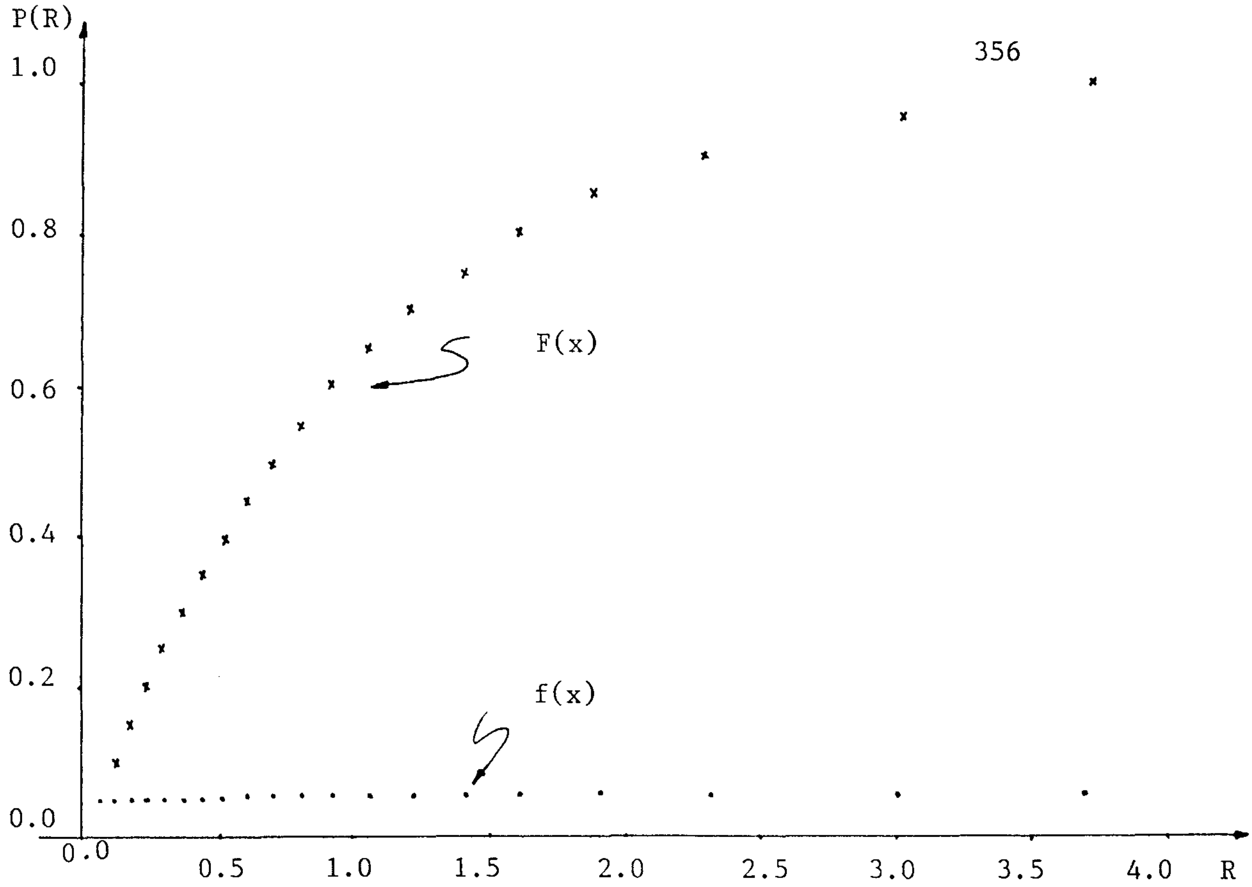


Figure 8.2

Equal Proabability Discretization for the Exponential Distribution.

TESTING THE ACCURACY OF THE APPROXIMATE PDF DUE TO DISCRETIZATION

The accuracy of the approximate pdf of the project completion time due to discretizing denoted by F(N) can be measured by its closeness to the "true" pdf, denoted by F'(N). Such "closeness" is measured either by the maximum value of the absolute deviation of F(N) from F'(N), denoted by MDV, or the average value of the absolute deviations, denoted by ADV.

Since the CIM approach is based on histograms of equal width within each distribution, in the following we apply the second method (Equal distances) to small projects, and then in Chapter 9 we compare the accuracy of the CIM approach and the Equal Distances method by measuring their closeness to the "true" pdf.

Example 1:

consider PERT network of Figure 8.3 assume activities are similar and duration times of activities are normally distributed with mean 100 and variance 100.649.

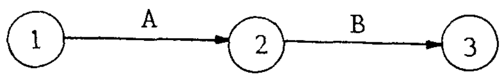


Figure 8.3

To approximate the normal by a discrete distribution, we first divide the range of the random variable x into appropriate intervals. Naturally, the smaller the intervals, the better the approximation. In Dodin's (1980) study in which the discretization is based on equal distances in most of the cases 20 cells are used in discretizing procedure, while in Yong's (1985) study in which addition of distributions are based on CIM approach it is concluded that for the addition operation, an initial input of 45 cells and a squeeze level of 0.12% will result in 30 cells with very little error for both the Normal and the skewed distribution after squeezing, therefore, in order to compare the accuracy of these two methods, we use 40 cells for input distribution and a squeeze level of 0.166%.

From the standard normal tables, we obtain the information in Table 8.1.

Table 8.1

z	F(z)	z	F(z)
$-\infty$	0.0000	0.1	0.5398
-3.7	0.0001	0.3	0.6179
-3.5	0.0002	0.5	0.6915
-3.3	0.0005	0.7	0.7580
-3.1	0.0010	0.9	0.8159
-2.9	0.0019	1.1	0.8643
-2.7	0.0035	1.3	0.9032
-2.5	0.0062	1.5	0.9332
-2.3	0.0107	1.7	0.9554
-2.1	0.0179	1.9	0.9713
-1.9	0.0287	2.1	0.9821
-1.7	0.0446	2.3	0.9893
-1.5	0.0668	2.5	0.9938
-1.3	0.0968	2.7	0.9965
-1.1	0.1357	2.9	0.9981
-0.9	0.1841	3.1	0.9990
-0.7	0.2420	3.3	0.9995
-0.5	0.3085	3.5	0.9998
-0.3	0.3821	3.7	0.9999
-0.1	0.4602	∞	1.0000

To approximate the standard normal by a discrete distribution using intervals of 0.2 would be equivalent to replacing the continuous scale $-\infty<z<\infty$ with the discrete values -3.8,-3.6,-3.4,....,3.4,3.6, and 3.8. In other words, we have replaced each interval by its midpoint. Table 8.2 shows the discrete approximation. By replacing z in Table 8.2 by $\mu+\sigma z$ where $\mu=100$ and $\sigma = \sqrt{100.649} = 10.032$, we can discretize the continuous distributions of activities of Figure 8.3 as shown in Table 8.3.

Table 8.2

z	p(z)	z	p(z)
-3.8	0.0001	0.2	0.0781
-3.6	0.0001	0.4	0.0736
-3.4	0.0003	0.6	0.0665
-3.2	0.0005	0.8	0.0579
-3.0	0.0009	1.0	0.0484
-2.8	0.0016	1.2	0.0389
-2.6	0.0027	1.4	0.0300
-2.4	0.0045	1.6	0.0222
-2.2	0.0072	1.8	0.0159
-2.0	0.0108	2.0	0.0108
-1.8	0.0159	2.2	0.0072
-1.6	0.0222	2.4	0.0045
-1.4	0.0300	2.6	0.0027
-1.2	0.0389	2.8	0.0016
-1.0	0.0484	3.0	0.0009
-0.8	0.0579	3.2	0.0005
-0.6	0.0665	3.4	0.0003
-0.4	0.0736	3.6	0.0001
-0.2	0.0781	3.8	0.0001
0.0	0.0796		

Table 8.3

x	p(x)	F(x)	x	p(x)	F(x)
61.7796	0.0001	0.0001	102.0116	0.0781	0.6179
63.7912	0.0001	0.0002	104.0232	0.0736	0.6915
65.8028	0.0003	0.0005	106.0348	0.0665	0.7580
67.8144	0.0005	0.0010	108.0464	0.0579	0.8159
69.8260	0.0009	0.0019	110.0580	0.0484	0.8643
71.8376	0.0016	0.0035	112.0696	0.0389	0.9032
73.8492	0.0027	0.0062	114.0812	0.0300	0.9332
75.8608	0.0045	0.0107	116.0928	0.0222	0.9554
77.8724	0.0072	0.0179	118.1044	0.0159	0.9713
79.8840	0.0108	0.0287	120.1160	0.0108	0.9821
81.8956	0.0159	0.0446	122.1276	0.0072	0.9893
83.9072	0.0222	0.0668	124.1392	0.0045	0.9938
85.9188	0.0300	0.0968	126.1508	0.0027	0.9965
87.9304	0.0389	0.1357	128.1624	0.0016	0.9981
89.9420	0.0484	0.1841	130.1740	0.0009	0.9990
91.9536	0.0579	0.2420	132.1856	0.0005	0.9995
93.9652	0.0665	0.3085	134.1872	0.0003	0.9998
95.9768	0.0736	0.3821	136.2088	0.0001	0.9999
97.9884	0.0781	0.4602	138.2204	0.0001	1.0000
100.0000	0.0796	0.5398			

Mean and variance of the data of Table 8.3 can be computed as follows:

$$\mu = \sum_x xp(x)=100 \ ,$$

$$\sigma^2 = \sum_x x^2 \ p(x) - (\sum_x xp(x))^2$$

$$= 10101.532 - 10000=101.532 \ .$$

Notice that the deviation between variance of the discrete approximate and the variance of the continuous distribution i.e. (101.532-100.649 = 0.883) is due to discretizing procedure.

As mentioned previously, the approximation can be improved by choosing smaller intervals.

By convoluting the distribution function of Table 8.3 with itself, approximate pdf of the project completion time of Figure 8.3 can be obtained as shown in Table 8.4.

Table 8.4

y=x*x	p(y)	F(y)	y=x*x	p(y)	F(y)
133.6172	0.0000004	0.0000004	202.0116	0.0557610	0.5838901
135.6288	0.0000012	0.0000016	204.0232	0.0541188	0.6380089
137.6404	0.0000028	0.0000044	206.0348	0.0514878	0.6894967
139.6520	0.0000062	0.0000106	208.0464	0.0480184	0.7375151
141.6636	0.0000114	0.0000220	210.0580	0.0438994	0.7914145
143.6752	0.0000208	0.0000428	212.0696	0.0393414	0.8207559
145.6868	0.0000376	0.0000804	214.0812	0.0345608	0.8553167
147.6984	0.0000650	0.0001454	216.0928	0.0297620	0.8850787
149.7100	0.0001092	0.0002546	218.1044	0.0251236	0.9102023
151.7216	0.0001794	0.0004340	220.1160	0.0207898	0.9309921
153.7332	0.0002878	0.0007218	222.1276	0.0168646	0.9478567
155.7448	0.0004520	0.0011738	224.1392	0.0134100	0.9612667
157.7564	0.0006950	0.0018688	226.1508	0.0104532	0.9717199
159.7680	0.0010464	0.0029152	228.1624	0.0079876	0.9797075
161.7796	0.0015438	0.0044590	230.1740	0.0059834	0.9856909
163.7912	0.0022320	0.0066910	232.1856	0.0043938	0.9900847
165.8028	0.0031630	0.0098540	234.1972	0.0031630	0.9932477
167.8144	0.0043938	0.0142478	236.2088	0.0022320	0.9954797
169.8240	0.0059834	0.0202312	238.2204	0.0015438	0.9970235
171.8376	0.0079876	0.0282188	240.2320	0.0010464	0.9980699
173.8492	0.0104532	0.0386720	242.2436	0.0006950	0.9987649
175.8608	0.0134100	0.0520820	244.2552	0.0004520	0.9992169
177.8724	0.0168646	0.0689466	246.2668	0.0002878	0.9995047
179.8840	0.0207898	0.0897364	248.2784	0.0001794	0.9996841
181.8956	0.0251236	0.1148600	250.2900	0.0001092	0.9997933
183.9072	0.0297620	0.1446220	252.3016	0.0000650	0.9998583
185.9188	0.0345608	0.1791828	254.3132	0.0000376	0.9998959

Table 8.4 (Continued)

y=x*x	p(y)	F(y)	y=y*y	p(y)	F(y)
187.9304	0.0393414	0.2185242	256.3248	0.0000208	0.9999167
189.9420	0.0438994	0.2624236	258.3364	0.0000114	0.9999281
191.9536	0.0480184	0.3104420	260.3480	0.0000062	0.9999343
193.9652	0.0514878	0.3619298	262.3596	0.0000028	0.9999371
195.9768	0.0541188	0.4160486	264.3712	0.0000012	0.9999383
197.9884	0.0557610	0.4718096	266.3828	0.0000004	0.9999387
200.0000	0.0563195	0.5281291			

Table 8.4 shows the pdf of project completion time. Mean and variance of the project completion time can be computed as follows:

$$\mu_T = \sum_y yp(y) = 199.98755 ,$$

$$\sigma_T^2 = \sum_y y^2 p(y) - (\sum_y yp(y))^2$$

$$= 40200.457 - 39995.02 = 205.437 .$$

Notice that the number of ordered pairs (realizations) in Table 8.4 is 67, and since we are not interested in extreme values associated with distribution tails with respect to the final results, therefore, we use the same truncation procedure (squeezing) as it is used in (Yong, 1985) which is the squeezing procedure for truncation used in the British Petroleum International limited second generation risk analysis software.

The squeezing procedure for truncation used in BP second generation risk analysis software is based on a delibrate misallocation of probabilities in order to lose unwanted distribution tails at a certain specific cut off level and then allocating the latter uniformly over the whole of the remaining probability distribution.

In the software, squeezing is performed if any cell has an upper boundary cumulative probability level less than a certain level, d, for example, or a lower boundary cumulative probability level greater than a level (1-d) where d is set in the software.

Squeezing will involve cutting off a total probability of E_L at the lower end and E_U at the upper end (Figure 8.4) where:

$$E_L + E_U = E \text{ (with E being set in the software)}$$

$$E_L = \frac{(m - a)}{(b - a)} \times E$$

m = mean of distribution

a = absolute minimum value of the distribution

b = absolute maximum value of the distribution .

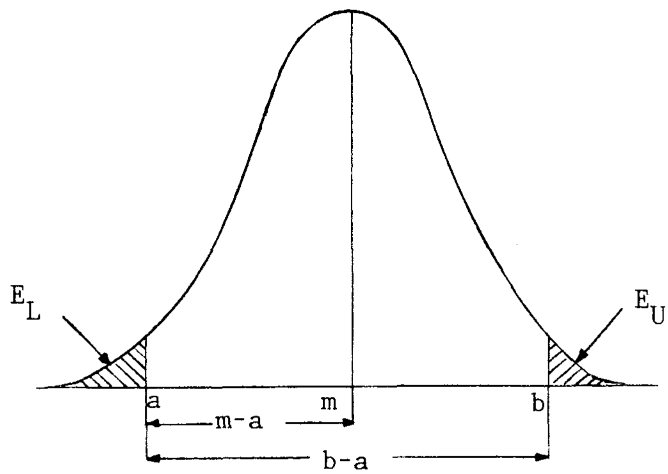


Figure 8.4: Distribution Showing the Mean, Extreme Values and Extreme Tail Probabilities.

If the E_L cumulative probability levels fall within any one cell then the probability attached to the cell will be reduced by the relevant amount. As already mentioned, the E probability cut off will be allocated evenly to the remaining cells. Finally, cells may remain with relevant cumulative probability levels less than d or greater than 1 - d, as squeezing is performed only once on each result distribution. The way in which squeezing is actually performed is well documented in "Appendix XII of the BP's Second Generation Risk Analysis Software User Specification".

Now we return to example problem, let us truncate the first and last

11 realization times of Table 8.4 and allocate the (2x0.0007218 = 0.0014436) probability cut off evenly to the remaining ordered pairs as shown in Table 8.5.

Since in Table 8.5 number of ordered pairs is 45 and is greater than 24 which we would like to have to make a comparison between the accuracy of Equal Distances method and the CIM approach in the next chapter, therefore, we approximate the pdf of Table 8.5 by another pdf with 24 ordered pairs as shown in Table 8.6 according to the following rules as used by Dodin (1980).

(i) The full range of the distribution F(i) is maintained.

(ii) The intermediate K-2 points are mapped into k-2 points using the following three steps;

(ii.1) Let $\Delta=(r_{K-1}-r_2)/(k-2)$, then we have k-2 intervals; each is of width Δ. The first interval contains all the realizations in the interval $[r_2,r_2+\Delta]$, and the nth interval contains all the realizations $r_m \epsilon [r_2+\Delta(n-1),r_2+n\Delta]$ for n=2,3,....,k-2.

(ii.2) For the realizations in the nth interval let

$$x_n=\sum_m r_m p(r_m) \quad \text{and} \quad y_n = \sum_m p(r_m),$$

then

$$(r'_n \, , \, p(r'_n))=(x_n/y_n, \, y_n),$$

(ii.3) If the nth interval is empty, i.e. there does not exist any

$$r_m \epsilon [r_2+\Delta(n-1),r_2+n], \quad \text{then}$$

$$(r'_n, \, p(r'_n))=(r_2+(n-0.5),0).$$

Table 8.5

y=x*x	p(y)	y=x*x	p(y)
155.7448	0.0004800	202.0116	0.0557930
157.7564	0.0007270	204.0232	0.0541508
159.7680	0.0010784	206.0348	0.0515198
161.7796	0.0015758	208.0464	0.0480504
163.7912	0.0022640	210.0580	0.0439314
165.8028	0.0031950	212.0696	0.0393734
167.8144	0.0044258	214.0812	0.0345928
169.8260	0.0060154	216.0928	0.0297940
171.8376	0.0080196	218.1044	0.0251556
173.8492	0.0104852	220.1160	0.0208218
175.8608	0.0134420	222.1276	0.0168966
177.8724	0.0168966	224.1392	0.0134420
179.8840	0.0208218	226.1508	0.0104852
181.8956	0.0251556	228.1624	0.0080196
183.9072	0.0297940	230.1740	0.0060154
185.9188	0.0345928	232.1856	0.0044258
187.9304	0.0393734	234.1972	0.0031950
189.9420	0.0439314	236.2088	0.0022640
191.9536	0.0480504	238.2204	0.0015758
193.9652	0.0515198	240.2320	0.0010784
195.9768	0.0541508	242.2436	0.0007270
197.9884	0.0557930	244.2552	0.0004840
200.0000	0.0563515		

Table 8.6

y=x*x	p(y)	F(y)=F(N)	y=x*x	p(y)	F(y)=F(N)
155.74480	0.0004840	0.0004840	202.011600	0.0557930	0.5839363
158.95795	0.0018054	0.0022894	205.003940	0.1056706	0.6896069
162.96562	0.0038398	0.0061292	209.224854	0.0919818	0.7815887
166.97102	0.0076208	0.0137500	213.010380	0.0739662	0.8555549
170.97542	0.0140350	0.0277850	217.013630	0.0549496	0.9105045
174.97928	0.0239272	0.0517122	221.017120	0.0377184	0.9482229
178.98286	0.0377184	0.0893060	225.020700	0.0239272	0.9721501
182.98630	0.0549496	0.1443802	229.024560	0.0140350	0.9861851
186.98959	0.0739662	0.2183464	233.028940	0.0076208	0.9938059
190.99283	0.0919818	0.3103282	237.034320	0.0038398	0.9976457
194.99606	0.1056706	0.4159988	241.041920	0.0018054	0.9994511
198.99920	0.1121445	0.5281433	244.255200	0.0004840	0.9999351

Mean and variance of the data of Table 8.6 can be computed as follows:

$$\mu = \sum_y yp(y)=200.00695 \quad ,$$

$$\sigma^2 = \sum_y y^2p(y)-(\sum_y yp(y))^2$$

$= 40205.239 - 40002.78 = 202.459$.

Table 8.6 shows the approximate pdf of the project completion time. Following section presents a method for determining the accuracy of the approximate pdf.

This method is presented in Dodin (1980) for comparing the accuracy of the pdf of the project completion time obtained using his approach with the true pdf obtained using Monte Carlo Simulation.

Comparing the Approximate pdf with the "true" pdf

Here, the "true" pdf of the project completion time is represented by the pdf obtained from the standard normal tables denoted by F'(N) as shown in Table 8.7; the cardinality of F'(N) is 24. This distribution is compared with the approximate pdf denoted by F(N) which has 24 realizations. The objective of the comparison is to determine the maximum absolute deviation (MDV), the average value of the absolute deviations (ADV) between two distributions, and the mean and the standard deviation of each distribution.

The deviations are computed using linear interpolation. Figure 8.5 illustrates the concept of linear interpolation where the points generating the solid line represent the approximate pdf, F(N), and the scattered points in the plane (R,P(R)) represent the "true" pdf, F'(N).

Table 8.7

Realization	Probability	F'(N)	Realization	Probability	F'(N)
155.74778	0.00140	0.00140	201.92390	0.10687	0.60687
159.59579	0.00209	0.00349	205.77191	0.09935	0.70622
163.44380	0.00379	0.00728	209.61992	0.08584	0.79206
167.29181	0.00772	0.01500	213.46793	0.06891	0.86097
171.13982	0.01370	0.02870	217.31594	0.05150	0.91247
174.98783	0.02293	0.05163	221.16395	0.03590	0.94837
178.83584	0.03590	0.08753	225.01196	0.02293	0.97130
182.68385	0.05150	0.13903	228.85997	0.01370	0.98500
186.53186	0.06891	0.20794	232.70798	0.00772	0.99272
190.37987	0.08584	0.29378	236.55599	0.00379	0.99651
194.22788	0.09935	0.39313	240.40400	0.00209	0.99860
198.07589	0.10687	0.50000	244.25201	0.00140	1.00000

$\mu=200$ and $\sigma^2=201.75$

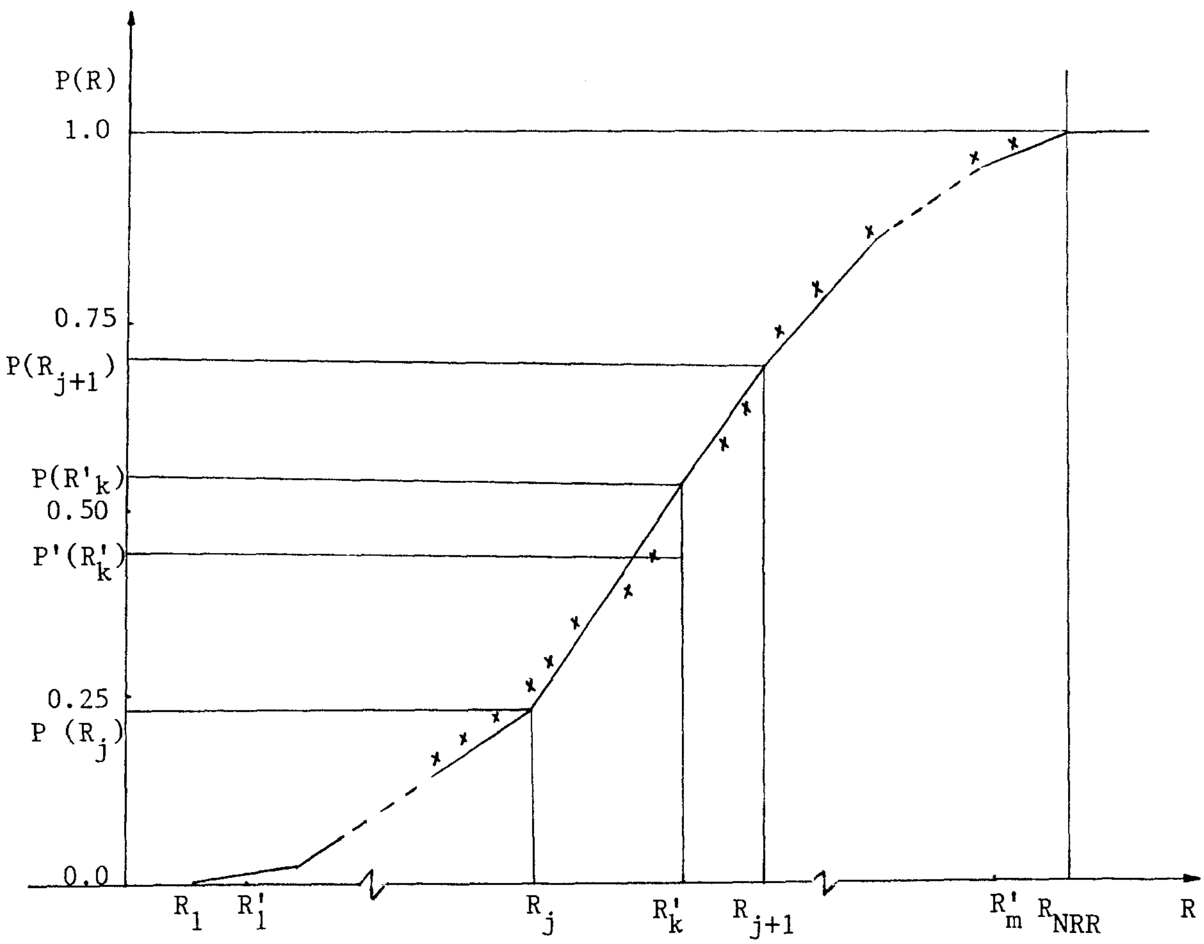


Figure 8.5

Linear Interpolation

If the number of realizations in F'(N) is n=NIN, then we have a deviations.

Let

D_k: be the k^{th} deviation where $k=1,2,\ldots,n$.

Hence,

$$MDV=\max_k \{|D_k|\}$$

$$ADV = \sum_{k=1}^{n} |D_k|/n.$$

In the approximate pdf the minimum and maximum realization times of the project are always preserved; they are represented by R_1 and R_{NNR} respectively.

Hence,

$$R'_1 \geqslant R_1$$

and
$$R'_n \leqslant R_{NRR}$$

Therefore, for any R'_k of the true realizations, $k=1,2,\ldots,n$, there exists an approximate realization R_j where $j<NRR$ such that

$$R_j \leqslant R'_k \leqslant R_{j+1}$$

Using this relation we can obtain the equation of the line segment connecting the two points $(R_j,P(R_j))$ and $(R_{j+1},P(R_{j+1}))$. If such a line is denoted by $y = rx + s$

where x is the realization axis and y is the probability axis, then the slope of the line is

$$r = (P(R_{j+1}) - P(R_j))/(R_{j+1} - R_j)$$

and the intercept is

$$s = P(R_j) - rR_j.$$

Hence, if R'_k is given, then using the line equation we calculate the corresponding approximate probability $P(R'_k)$ where

$$P(R'_k) = y = rR'_k + s,$$

then
$$D_k = P(R'_k) - P'(R'_k).$$

See Figure 8.5 for illustration.

The linear interpolation is carried out for all realizations except for those which lie in the 0.01 left and right tails of the "true" distribution. The exclusion of the two tails does not alter the values of MDV and may alter ADV only slightly, and speeds up the interpolation since many realizations with negligible probabilities might lie in the tails. For example, the application of the linear interpolation to Tables 8.6 and 8.7 led to the exclusion of the first and last three realizations of Table 8.7. Table 8.8 has the complete output of the linear interpolation; the third column in the table headed by "APRXMTD PROB." represents $P(R_k')$ and the last column represents D_k. Figure 8.6 is a digital plot of the first three column of Table 8.8.

The symbol (·) represents the "true" distribution where (-) represents the approximate distribution and (×) is used whenever (·) and (-) are to be printed in the same location in the xy-plane.

I	REALIZATION	EXACT PROB.	APRXMTD PROB.	ACTUAL DIFFERENCE
1	167.29181	0.01500	0.0148743	-.0001257
2	171.13982	0.02870	0.0287674	.0000674
3	174.98783	0.05163	0.0517927	.0001627
4	178.83584	0.08753	0.0880454	.0005154
5	182.68385	0.13903	0.1402288	.0011988
6	186.53186	0.20794	0.2098892	.0019492
7	190.37987	0.29378	0.2962443	.0024643
8	194.22788	0.39313	0.3957216	.0025916
9	198.07589	0.50000	0.5022775	.0022775
10	201.92390	0.60687	0.5823119	-.0245581
11	205.77191	0.70622	0.7063424	.0001224
12	209.61992	0.79206	0.7893080	-.0027520
13	213.46793	0.86097	0.8618353	.0008653
14	217.31594	0.91247	0.9133526	.0008826
15	221.16395	0.94837	0.9491004	.0007304
16	225.01196	0.97130	0.9720978	.0007978
17	228.85997	0.98500	0.9856081	.0006081
18	232.70798	0.99272	0.9931950	.0004750

Table 8.8

Comparison of the Exact and the Approximate Probability Distribution Functions.

The Average of the Absolute Values of the Deviations = .0023969

The Maximum of the Absolute Values of the Deviations = .0245581. It is No. 10.

Number of Positive Deviates = 15

Number of Negative Deviates = 3

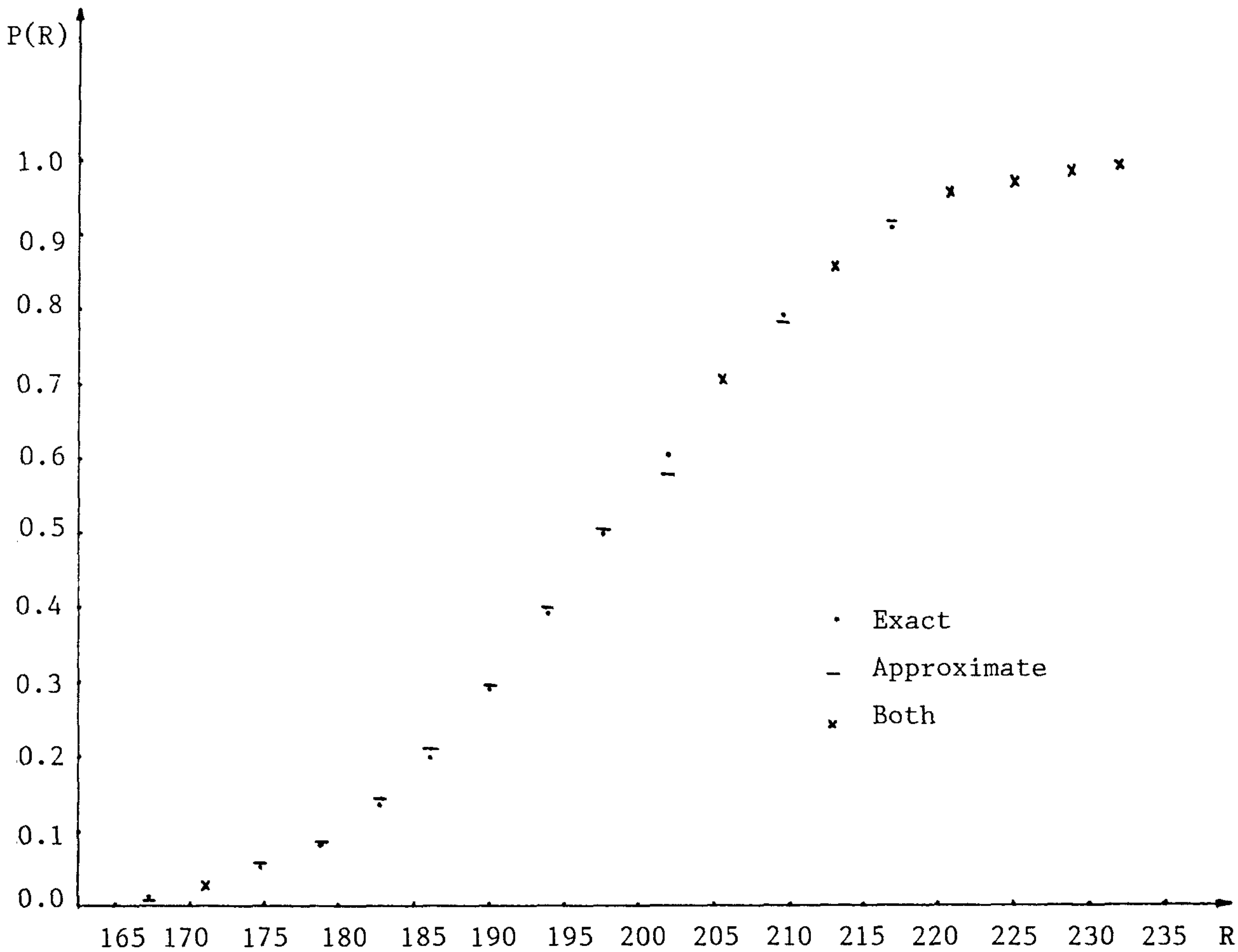


Figure 8.6

Comparison of the Exact and the Approximate Probability Distribution Functions of Example 1.

Example 2:

Let us assume activities A and B of Figure 8.3 are parallel as shown in Figure 8.7, then the probability distribution function of T=max{A,B} can be computed as shown in Table 8.9.

Table 8.9 shows the approximate pdf of the project completion time using 40 cells in discretizing the input distribution.

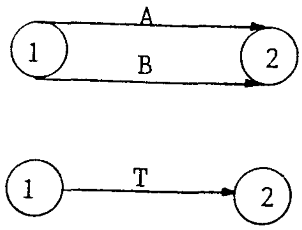


Figure 8.7

The approximate pdf of the project completion time obtained from the standard normal tables by taking the maximum of the duration times of a normal distribution with $\mu=100$ and $\sigma=10.032$ using 140 cells as shown in Table 8.10 assumed to be error free and considered as "true" pdf.

Table 8.9

x	p(x)	F(x)	p(z)	F(z)
61.7796	.0001	.0001	$(.0001)^2 - (.0000)^2 = .0000000$	.0000000
63.7912	.0001	.0002	$(.0002)^2 - (.0000)^2 = .0000000$	.0000000
65.8028	.0003	.0005	$(.0005)^2 - (.0002)^2 = .0000002$	.0000002
67.8144	.0005	.0010	$(.0010)^2 - (.0005)^2 = .0000008$	.0000010
69.8260	.0009	.0019	$(.0019)^2 - (.0010)^2 = .0000026$	.0000036
71.8376	.0016	.0035	$(.0035)^2 - (.0019)^2 = .0000086$	.0000122
73.8492	.0027	.0062	$(.0062)^2 - (.0035)^2 = .0000262$	.0000384
75.8608	.0045	.0107	$(.0107)^2 - (.0062)^2 = .0000760$	.0001144
77.8724	.0072	.0179	$(.0179)^2 - (.0107)^2 = .0002060$	.0003204
79.8840	.0108	.0287	$(.0287)^2 - (.0179)^2 = .0005032$	.0008236
81.8956	.0159	.0446	$(.0446)^2 - (.0287)^2 = .0011655$	.0019891
83.9072	.0222	.0668	$(.0668)^2 - (.0446)^2 = .0024731$	.0044622
85.9188	.0300	.0968	$(.0968)^2 - (.0668)^2 = .0049080$	.0093702
87.9304	.0389	.1357	$(.1357)^2 - (.0968)^2 = .0090442$	.0184144
89.9420	.0484	.1841	$(.1841)^2 - (.1357)^2 = .0154784$	.0338928
91.9536	.0579	.2420	$(.2420)^2 - (.1841)^2 = .0246712$	.0585640
93.9652	.0665	.3085	$(.3085)^2 - (.2420)^2 = .0366082$	.0951722
95.9768	.0736	.3821	$(.3821)^2 - (.3085)^2 = .0508282$	.1460004
97.9884	.0781	.4602	$(.4602)^2 - (.3821)^2 = .0657836$	.2117840
100.0000	.0796	.5398	$(.5398)^2 - (.4602)^2 = .0796000$	.2913840
102.0116	.0781	.6179	$(.6179)^2 - (.5398)^2 = .0904164$	.3818004
104.0232	.0736	.6915	$(.6915)^2 - (.6179)^2 = .0963718$	.4781722
106.0348	.0665	.7580	$(.7580)^2 - (.6915)^2 = .0963918$	.5745640
108.0464	.0579	.8159	$(.8159)^2 - (.7580)^2 = .0911288$	.6656928
110.0580	.0484	.8643	$(.8643)^2 - (.8159)^2 = .0813216$	.7470144
112.0696	.0389	.9032	$(.9032)^2 - (.8643)^2 = .0687558$	.8157702
114.0812	.0300	.9332	$(.9332)^2 - (.9032)^2 = .0550920$	.8708622
116.0928	.0222	.9554	$(.9554)^2 - (.9332)^2 = .0419269$	.9127891
118.1044	.0159	.9713	$(.9713)^2 - (.9554)^2 = .0306345$	.9434236
120.1160	.0108	.9821	$(.9821)^2 - (.9713)^2 = .0210968$	.9645204
122.1276	.0072	.9893	$(.9893)^2 - (.9821)^2 = .0141940$	.9787144
124.1392	.0045	.9938	$(.9938)^2 - (.9893)^2 = .0089240$	.9876384
126.1508	.0027	.9965	$(.9965)^2 - (.9938)^2 = .0053738$	.9930122
128.1624	.0016	.9981	$(.9981)^2 - (.9965)^2 = .0031914$	.9962036
130.1740	.0009	.9990	$(.9990)^2 - (.9981)^2 = .0017974$	.9980010
132.1856	.0005	.9995	$(.9995)^2 - (.9990)^2 = .0009992$	.9990002
134.1972	.0003	.9998	$(.9998)^2 - (.9995)^2 = .0005998$	.9996000
136.2088	.0001	.9999	$(.9999)^2 - (.9998)^2 = .0002000$	.9998000
138.2204	.0001	1.0000	$(1.000)^2 - (.9999)^2 = .0002000$	1.0000000

where z=x.x

$\mu_z = 105.67481$,

$\sigma_z = 8.3246621$.

Table 8.10

x	p(x)	F(x)	p(t)	F(t)
61.2802	.0001	.0001	0.0000000	0.0000000
61.8028	.0001	.0002	0.0000000	0.0000000
62.3254	.0001	.0003	0.0000000	0.0000000
62.9020	.0001	.0004	0.0000001	0.0000001
63.4516	.0001	.0005	0.0000001	0.0000002
64.0012	.0001	.0006	0.0000001	0.0000003
64.5508	.0001	.0007	0.0000001	0.0000004
65.1004	.0001	.0008	0.0000002	0.0000006
65.6500	.0001	.0009	0.0000002	0.0000008
66.1996	.0002	.0011	0.0000004	0.0000012
66.7492	.0002	.0013	0.0000004	0.0000016
67.2988	.0003	.0016	0.0000009	0.0000025
67.8484	.0003	.0019	0.0000011	0.0000036
68.3980	.0003	.0022	0.0000012	0.0000048
68.9476	.0004	.0026	0.0000019	0.0000067
69.4972	.0004	.0030	0.0000023	0.0000090
70.0468	.0005	.0035	0.0000032	0.0000122
70.5964	.0005	.0040	0.0000038	0.0000160
71.1460	.0007	.0047	0.0000060	0.0000220
71.6956	.0007	.0054	0.0000071	0.0000291
72.2452	.0008	.0062	0.0000093	0.0000384
72.7948	.0009	.0071	0.0000120	0.0000504
73.3444	.0011	.0082	0.0000168	0.0000672
73.8940	.0012	.0094	0.0000211	0.0000883
74.4436	.0013	.0107	0.0000261	0.0001144
74.9932	.0015	.0122	0.0000344	0.0001488
75.5428	.0017	.0139	0.0000444	0.0001932
76.0924	.0019	.0158	0.0000564	0.0002496
76.6420	.0021	.0179	0.0000708	0.0003204
77.1916	.0023	.0202	0.0000876	0.0004080
77.7412	.0026	.0228	0.0001118	0.0005198
78.2908	.0028	.0256	0.0001355	0.0006553
78.8404	.0031	.0287	0.0001683	0.0008236
79.3900	.0035	.0322	0.0002132	0.0010367
79.9396	.0037	.0359	0.0002520	0.0012897
80.4892	.0042	.0401	0.0003192	0.0016089
81.0388	.0045	.0446	0.0003831	0.0019920
81.5884	.0049	.0495	0.0004611	0.0024531
82.1380	.0053	.0548	0.0005528	0.0030059
82.6876	.0058	.0606	0.0006693	0.0036752
83.2372	.0062	.0668	0.0007899	0.0044651
83.7868	.0067	.0735	0.0009400	0.0054051
84.3364	.0073	.0808	0.0011264	0.0065315
84.8860	.0077	.0885	0.0013036	0.0078351
85.4356	.0083	.0968	0.0015380	0.0093731
85.9852	.0088	.1056	0.0017811	0.0111542
86.5348	.0095	.1151	0.0020967	0.0132509
87.0844	.0100	.1251	0.0024020	0.0156529
87.6340	.0106	.1357	0.0027644	0.0184173
88.1836	.0112	.1468	0.0031358	0.0215531
88.7332	.0118	.1587	0.0036354	0.0251885
89.2828	.0124	.1711	0.0040896	0.0292781
89.8324	.0130	.1841	0.0046176	0.0338957

Table 8.10 (Continued)

x	p(x)	F(x)	p(t)	F(t)
90.3820	.0136	.1977	.0051924	.0390881
90.9316	.0142	.2119	.0058164	.0449045
91.4812	.0147	.2266	.0064459	.0513504
92.0308	.0154	.2420	.0072165	.0585669
92.5804	.0158	.2578	.0078968	.0664637
93.1300	.0164	.2742	.0087248	.0751885
93.6796	.0170	.2912	.0096118	.0848003
94.2292	.0173	.3085	.0103748	.0951751
94.7788	.0179	.3264	.0113647	.1065398
95.3284	.0182	.3446	.0122122	.1187520
95.8780	.0186	.3632	.0131651	.1319171
96.4276	.0189	.3821	.0140862	.1460033
96.9772	.0192	.4013	.0150412	.1610445
97.5268	.0194	.4207	.0159468	.1769913
98.0764	.0197	.4404	.0169637	.1939550
98.6260	.0198	.4602	.0178319	.2117869
99.1756	.0199	.4801	.0187120	.2304989
99.7252	.0199	.5000	.0195040	.2500029
100.2748	.0199	.5199	.0202960	.2702989
100.8244	.0199	.5398	.0210880	.2913869
101.3740	.0198	.5596	.0217681	.3131550
101.9236	.0197	.5793	.0224363	.3355913
102.4732	.0194	.5987	.0228532	.3584445
103.0228	.0192	.6179	.0233588	.3818033
103.5724	.0189	.6368	.0237138	.4055171
104.1220	.0186	.6554	.0240349	.4295520
104.6716	.0182	.6736	.0241878	.4537398
105.2212	.0179	.6915	.0244353	.4781751
105.7708	.0173	.7088	.0242252	.5024003
106.3204	.0170	.7258	.0243882	.5267885
106.8700	.0164	.7422	.0240752	.5508637
107.4196	.0158	.7580	.0237032	.5745669
107.9692	.0154	.7734	.0235835	.5981504
108.5188	.0147	.7881	.0229541	.6211045
109.0684	.0142	.8023	.0225836	.6436881
109.6180	.0136	.8159	.0220076	.6656957
110.1676	.0130	.8289	.0213824	.6870781
110.7172	.0124	.8413	.0207104	.7077885
111.2668	.0118	.8531	.0199940	.7277825
111.8164	.0112	.8643	.0192348	.7470173
112.3660	.0106	.8749	.0184356	.7654529
112.9156	.0100	.8849	.0175980	.7830509
113.4652	.0095	.8944	.0169033	.7999542
114.0148	.0088	.9032	.0158189	.8157731
114.5644	.0083	.9115	.0150620	.8308351
115.1140	.0077	.9192	.0140964	.8449315
115.6636	.0073	.9265	.0134736	.8584051
116.2132	.0067	.9332	.0124600	.8708651
116.7628	.0062	.9394	.0116101	.8824752
117.3124	.0058	.9452	.0109307	.8934059
117.8620	.0053	.9505	.0100472	.9034531
118.4116	.0049	.9554	.0093389	.9127920
118.9612	.0045	.9599	.0086189	.9214109
119.5108	.0042	.9641	.0080808	.9294917

Table 8.10 (Concluded)

x	p(x)	F(x)	p(t)	F(t)
120.0604	.0037	.9678	.0071480	.9366397
120.6100	.0035	.9713	.0067680	.9434265
121.1596	.0031	.9744	.0060317	.9494582
121.7092	.0028	.9772	.0054645	.9549227
122.2588	.0026	.9798	.0050882	.9600109
122.8084	.0023	.9821	.0045124	.9645233
123.3580	.0021	.9842	.0041292	.9686525
123.9076	.0019	.9861	.0037436	.9723961
124.4572	.0017	.9878	.0033556	.9757517
125.0068	.0015	.9893	.0029656	.9787173
125.5564	.0013	.9906	.0025739	.9812912
126.1060	.0012	.9918	.0023789	.9836701
126.6556	.0011	.9929	.0021832	.9858533
127.2052	.0009	.9938	.0017880	.9876413
127.7548	.0008	.9946	.0015907	.9892320
128.3044	.0007	.9953	.0013940	.9920189
129.4036	.0005	.9965	.0009962	.9930151
129.9532	.0005	.9970	.0009968	.9940119
130.5028	.0004	.9974	.0007977	.9948096
131.0524	.0004	.9978	.0007981	.9956077
131.6020	.0003	.9981	.0005988	.9962065
132.1516	.0003	.9984	.0005989	.9968054
132.7012	.0003	.9987	.0005991	.9974045
133.2508	.0002	.9989	.0003996	.9978041
133.8004	.0002	.9991	.0003996	.9982037
134.3500	.0001	.9992	.0001998	.9984035
134.8996	.0001	.9993	.0001998	.9986033
135.4492	.0001	.9994	.0001999	.9988032
135.9988	.0001	.9995	.0001999	.9990031
136.5484	.0001	.9996	.0001999	.9992030
137.0980	.0001	.9997	.0001999	.9994029
137.6476	.0001	.9998	.0002000	.9996029
138.1972	.0001	.9999	.0002000	.9998029
138.7468	.0001	1.0000	.0002000	1.0000000

where t=x.x

$$\mu_t = 106.39181 \ ,$$

$$\sigma_t = 6.4543783 \ .$$

The application of the linear interpolation to Tables 8.10 and 8.9 led to the exclusion of the first 45 and last 21 realizations of Table 8.10. Table 8.11 has the complete output of the linear interpolation.

I	REALIZATION	EXACT PROB.	APRXMTD PROB.	ACTUAL DIFFERENCE
1	85.9852	0.0111542	0.0096687	-.0014855
2	88.1936	0.0215531	0.0203626	-.0011905
3	90.3820	0.0390881	0.0392891	.0002010
4	92.0308	0.0585669	0.0599689	.0014020
5	94.2292	0.0951751	0.1018428	.0066677
6	96.4276	0.1460033	0.1607424	.0147391
7	98.0764	0.1939550	0.2152662	.0213112
8	100.2748	0.2702989	0.3037355	.0334366
9	102.4732	0.3584445	0.4039147	.0454702
10	104.1220	0.4295520	0.4829640	.0533544
11	106.3204	0.5267885	0.5875021	.0607136
12	108.5188	0.6211045	0.6847901	.0636856
13	110.1676	0.6870781	0.7507604	.0636823
14	112.3660	0.7654529	0.8238877	.0584348
15	114.5644	0.8308351	0.8308351	.0500981
16	116.2132	0.8708651	0.9146226	.0437575
17	118.4116	0.9127920	0.9466453	.0338533
18	120.6100	0.9434265	0.9680060	.0245795
19	122.2588	0.9600109	0.9792964	.0192855
20	124.4572	0.9757517	0.9884878	.0127361
21	126.6556	0.9858533	0.9938130	.0079597

Table 8.11

Comparison of the Exact and the Approximate Probability Distribution Functions of Example 2.

The Average of the Absolute Values of the Deviations = 0.0294306

The Maximum of the Absolute Values of the Deviations = 0.0636856. It is No.12.

Number of Positive Deviates = 19

Number of Negative Deviates = 2

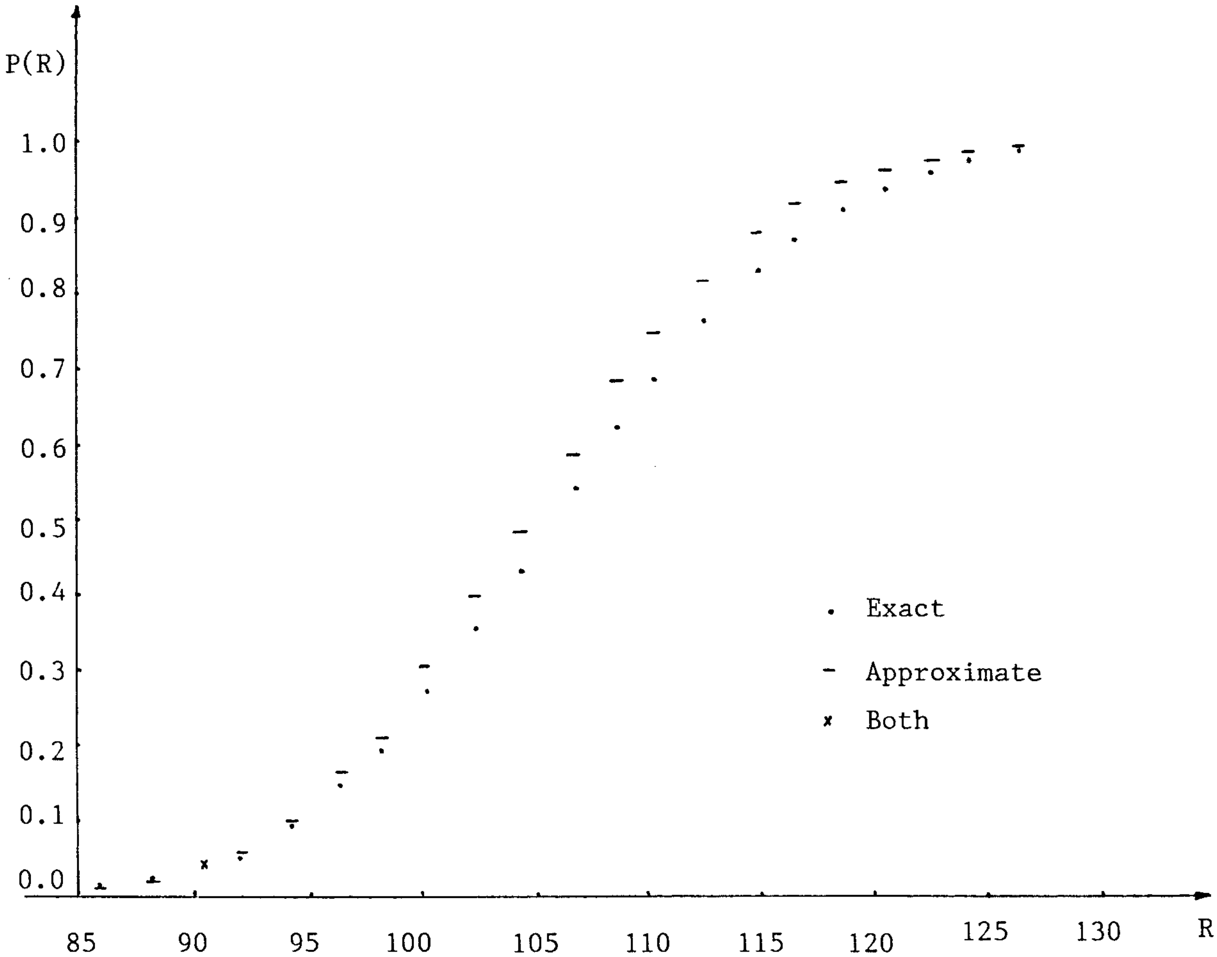


Figure 8.8

Comparison of the Exact and the Approximate Probability Distribution Functions of Example 2.

Example 3:

Let us assume duration times of activities A and B of Figure 8.9 are similar and have Beta distribution with parameters r=3 and s=5. This gave rise to a distribution with a range of 0 and 1 and a mean of .375 for each activity as shown in Figure 8.10.

To approximate the Beta by a discrete distribution, we first divide the range of the distribution into 30 cells, then from the tables of the Incomplete Beta-Function (Pearson, 1956), with these parameters we obtain the information in Table 8.12.

Since it is difficult to work out the actual or theoretical values corresponding to these 30 discrete points, the tables of the Incomplete Beta-Function with 100 cells for r=3 and s=5 as shown in Table 8.13 taken as being correct (error free) for calculation purposes.

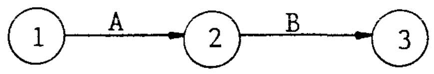


Figure 8.9

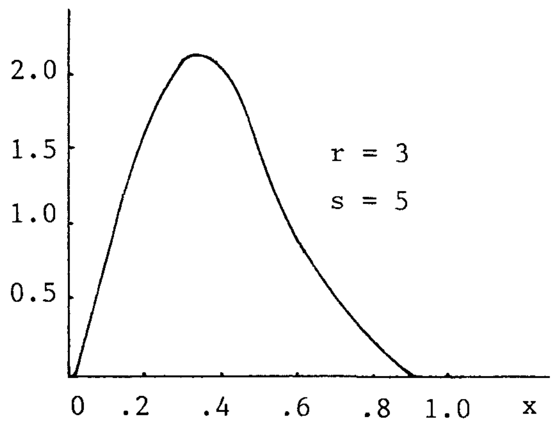


Figure 8.10

Beta Density

Table 8.12

x	p(x)	F(x)	x	p(x)	F(x)
.0333	.0015579	.0015579	.5333	.0451853	.8183889
.0666	.0090902	.0106481	.5667	.0372095	.9007837
.1000	.0209268	.0315749	.6000	.0296919	.9304756
.1333	.0349290	.0665039	.6333	.0226974	.9531730
.1667	.0483066	.1148105	.6667	.0168131	.9699861
.2000	.0601524	.1749629	.7000	.0119001	.9818862
.2333	.0698516	.2448145	.7333	.0078847	.9897709
.2667	.0763758	.3211903	.7667	.0049538	.9947247
.3000	.0799547	.4011450	.8000	.0028643	.9975890
.3333	.0806365	.4817815	.8333	.0014572	.9990462
.3667	.0786691	.5604506	.8667	.0006532	.9996994
.4000	.0745391	.6349897	.9000	.0002349	.9999343
.4333	.0684861	.7034758	.9333	.0000568	.9999911
.4667	.0613729	.7648487	.9667	.0000073	.9999984
.5000	.0535402	.8183889	1.0000	.0000002	.9999986

$$\mu = m_1' = \sum_x xp(x) = .374699, \quad m_2' = \sum_x x^2p(x) = .1651698 \quad ,$$

$$m_3' = \sum_x x^3\ p(x) = .0824376 \ ,$$

$$\sigma^2 = m_2 = m'_2 - (m_1')^2 = .1651698 - (.374699)^2 = .0247705 \implies \sigma = .1573864 \ ,$$

$$m_3 = m_3' - 3m_1'm_2' + 2(m_1')^3 = .0824376 - 3(.374699)(.1651698) + 2(.374699)^3$$

$$= .0019856 \ ,$$

$$g_1 = m_3/m_2^{\frac{3}{2}} = .0019856/(.0247705)^{\frac{3}{2}} = .509324 \ ,$$

where g_1 = coefficient of skewness.

Table 8.13

x	p(x)	F(x)	x	p(x)	F(x)
.01	.0000340	.0000340	.51	.0160761	.7895136
.02	.0002296	.0002636	.52	.0154085	.8049221
.03	.0005994	.0008630	.53	.0147325	.8196546
.04	.0011208	.0019838	.54	.0140509	.8337055
.05	.0017732	.0037570	.55	.0133668	.8470723
.06	.0025370	.0062940	.56	.0126829	.8597552
.07	.0033936	.0096876	.57	.0120020	.8717571
.08	.0043264	.0140140	.58	.0113264	.8830835
.09	.0053195	.0193335	.59	.0106589	.8937424
.10	.0063580	.0256915	.60	.0100015	.9037439
.11	.0074286	.0331201	.61	.0093566	.9131005
.12	.0085187	.0416388	.62	.0087260	.9218265
.13	.0096170	.0512558	.63	.0081117	.9299382
.14	.0107127	.0619685	.64	.0075155	.9374537
.15	.0117967	.0737652	.65	.0069387	.9443924
.16	.0128608	.0866260	.66	.0063828	.9507752
.17	.0138941	.1005201	.67	.0058490	.9566242
.18	.0148946	.1154147	.68	.0053384	.9619626
.19	.0158530	.1312677	.69	.0048518	.9668144
.20	.0167643	.1480320	.70	.0043900	.9712044
.21	.0176242	.1656562	.71	.0039534	.9751578
.22	.0184283	.1840845	.72	.0035425	.9787003
.23	.0191736	.2032581	.73	.0031576	.9818579
.24	.0198569	.2231150	.74	.0027984	.9846563
.25	.0204763	.2435913	.75	.0024652	.9871215
.26	.0210299	.2646212	.76	.0021575	.9892790
.27	.0215166	.2861378	.77	.0018751	.9911541
.28	.0219357	.3080735	.78	.0016173	.9927714
.29	.0222868	.3303603	.79	.0013835	.9941549
.30	.0225702	.3529305	.80	.0011730	.9953279
.31	.0227864	.3757169	.81	.0009847	.9963126
.32	.0229362	.3986531	.82	.0008178	.9971304
.33	.0230208	.4216739	.83	.0006711	.9978015
.34	.0230418	.4447157	.84	.0005433	.9983448
.35	.0230010	.4677167	.85	.0004335	.9987783
.36	.0229002	.4906169	.86	.0003399	.9991182
.37	.0227418	.5133587	.87	.0002615	.9993797
.38	.0225284	.5358871	.88	.0001968	.9995765
.39	.0222623	.5581494	.89	.0001443	.9997208
.40	.0219466	.5800960	.90	.0001026	.9998234
.41	.0215839	.6016799	.91	.0000704	.9998938
.42	.0211775	.6228574	.92	.0000461	.9999399
.43	.0207303	.6435877	.93	.0000287	.9999686
.44	.0202456	.6638333	.94	.0000166	.9999852
.45	.0197266	.6835599	.95	.0000087	.9999939
.46	.0191767	.7027366	.96	.0000040	.9999979
.47	.0185988	.7213354	.97	.0000015	.9999994
.48	.0179967	.7393321	.98	.0000004	.9999998
.49	.0173733	.7567054	.99	.0000001	.9999999
.50	.0167321	.7734375	1.00	.0000001	1.0000000

$\mu=.375$, $\sigma=.15504624$, $g_1=.3098386$.

Notice that the exact mean, variance, and coefficient of skewness of each activity can be computed as follows:

Mean = r/(r+s)

= 3/(3+5)

= .375 ,

Variance = $rs/(r+s+1)(r+s)^2$

= $3x5/(3+5+1)(3+5)^2$

= .0260416 ==> σ=.161374 ,

Coefficient of skewness = $2(s-r)(r+s+1)^{\frac{1}{2}}/(r+s+2)(rs)^{\frac{1}{2}}$

= $2(5-3)(3+5+1)^{\frac{1}{2}}/(3+5+2)(3x5)^{\frac{1}{2}}$

= .3098386 ·

Project completion time can be approximated using one of the following approaches.

(i) Convoluting the pdf of the activities using 30 cells for input distribution.

(ii) By interpolation of the incomplete Beta function.

(i) Approximating the pdf of the project completion time using 30 cells for input distribution.

The approximate pdf of project completion time using 30 cells can be obtained from Table 8.12 as shown in Table 8.14.

Table 8.14

y=x*x	p(y)	F(y)	y=x*x	p(y)	F(y)
.0667	.0000024	.0000024	.9667	.0332526	.8604786
.1000	.0000282	.0000306	1.0000	.0285586	.8890372
.1333	.0001478	.0001784	1.0333	.0240944	.9131316
.1667	.0004892	.0006676	1.0667	.0199670	.9330986
.2000	.0012232	.0018908	1.1000	.0162496	.9493820
.2333	.0025274	.0044182	1.1333	.0129808	.9623290
.2667	.0045528	.0089710	1.1667	.0101740	.9725030
.3000	.0073996	.0163706	1.2000	.0078188	.9803218
.3333	.0110962	.0274668	1.2333	.0058864	.9862082
.3667	.0155924	.0430592	1.2667	.0038590	.9900672
.4000	.0207592	.0638184	1.3000	.0031236	.9931908
.4333	.0264048	.0902232	1.3333	.0021958	.9953866
.4667	.0322856	.1225088	1.3667	.0015036	.9968902
.5000	.0381304	.1606392	1.4000	.0010014	.9978916
.5333	.0436598	.2042990	1.4333	.0006470	.9985386

Table 8.14 (Concluded)

y=x*x	p(y)	F(y)	y=x*x	p(y)	F(y)
.5667	.0486104	.2529094	1.4667	.0004036	.9989422
.6000	.0527492	.3056586	1.5000	.0002432	.9991854
.6333	.0558930	.3615516	1.5333	.0001400	.9993254
.6667	.0579154	.4194970	1.5667	.0000770	.9994024
.7000	.0587546	.4782216	1.6000	.0000402	.9994426
.7333	.0584130	.5366346	1.6333	.0000196	.9994622
.7667	.0569518	.5935864	1.6667	.0000086	.9994708
.8000	.0544864	.6480728	1.7000	.0000034	.9994742
.8333	.0511722	.6992450	1.7333	.0000012	.9994754
.8667	.0471930	.7464380	1.7667	.0000002	.9994756
.9000	.0427490	.7891870	1.8000	.0000001	.9994757
.9333	.0380390	.8272260	1.8333	.0000000	.9994757

$\mu = m_1' = .7385594$, $m_2' = .5939056$, $m_3' = .5126808$,

$\sigma^2 = m_2' - (m_1')^2 = .5939056 - (.7385594)^2 = .0484357 \implies \sigma = .2200811,$

$m_3 = m_3' - 3m_1'm_2' + 2(m_1')^3$

$= .5126808 - 3(.7385594)(.5939056) + 2(.7385594)^3 = .002501,$

$g_1 = m_3/(m_2)^{\frac{3}{2}} = .002501/(.0484357)^{\frac{3}{2}} = .2346219$.

(ii) Approximating the pdf of the project completion time using interpolation.

The approximate pdf of project completion time using the closeness to the exact values of the first three moments as a criterion can be obtained by interpolating the incomplete Beta function. The best fitted curve was found to be a Beta function with parameters r=6.5 and s=10.5988. The pdf of this function is shown in Table 8.15.

Table 8.15

z=x*x	p(z)	F(z)	z=x*x	p(z)	F(z)
.02	.0000002	.0000002	.92	.0246374	.7816812
.04	.0000008	.0000010	.94	.0231098	.8047910
.06	.0000051	.0000061	.96	.0215495	.8263405
.08	.0000177	.0000238	.98	.0199754	.8463159
.10	.0000475	.0000713	1.00	.0184058	.8647217
.12	.0001069	.0001782	1.02	.0168571	.8815788
.14	.0002106	.0003888	1.04	.0153442	.8969230

Table 8.15 (Concluded)

z=x*x	p(z)	F(x)	z=x*x	p(z)	F(z)
.16	.0003763	.0007651	1.06	.0138806	.9108036
.18	.0006226	.0013877	1.08	.0124771	.9232807
.20	.0009688	.0023565	1.10	.0111434	.9344241
.22	.0014325	.0037890	1.12	.0098865	.9443106
.24	.0020295	.0058185	1.14	.0087124	.9530320
.26	.0027729	.0085914	1.16	.0076246	.9606476
.28	.0036713	.0122627	1.18	.0066249	.9672725
.30	.0047295	.0169922	1.20	.0057141	.9729866
.32	.0059471	.0229393	1.22	.0048912	.9778778
.34	.0073187	.0302580	1.24	.0041537	.9820315
.36	.0088339	.0390919	1.26	.0034989	.9855304
.38	.0104780	.0495699	1.28	.0029220	.9884524
.40	.0122318	.0618017	1.30	.0024188	.9908712
.42	.0140721	.0758738	1.32	.0019837	.9928549
.44	.0159731	.0918469	1.34	.0016114	.9944663
.46	.0179068	.1097537	1.36	.0012949	.9957612
.48	.0198431	.1295968	1.38	.0010300	.9967912
.50	.0217518	.1513486	1.40	.0008099	.9976011
.52	.0235366	.1749511	1.42	.0006292	.9982303
.54	.0253661	.2003172	1.44	.0004826	.9987129
.56	.0270149	.2273321	1.46	.0003650	.9990779
.58	.0285231	.2558552	1.48	.0002723	.9993502
.60	.0298683	.2857235	1.50	.0001998	.9995500
.62	.0310314	.3167549	1.52	.0001442	.9996942
.64	.0319959	.3487508	1.54	.0001023	.9997965
.66	.0327505	.3815013	1.56	.0000710	.9998675
.68	.0332870	.4147883	1.58	.0000484	.9999159
.70	.0336013	.4483896	1.60	.0000321	.9999480
.72	.0336933	.4820829	1.62	.0000208	.9999688
.74	.0335668	.5156497	1.64	.0000131	.9999819
.76	.0332286	.5488783	1.66	.0000079	.9999898
.78	.0326896	.5815679	1.68	.0000047	.9999945
.80	.0319624	.6135303	1.70	.0000027	.9999972
.82	.0310628	.6445931	1.72	.0000014	.9999986
.84	.0300083	.6746014	1.74	.0000007	.9999993
.86	.0288183	.7034197	1.76	.0000004	.9999997
.88	.0275123	.7309320	1.78	.0000002	.9999999
.90	.0261118	.7570438	1.80	.0000001	1.0000000

$\mu = m_1' = .7502554$, $m_2' = .6150564$, $m_3' = .5423512$,

$\sigma^2 = m_2' - (m_1')^2 = .6150564 - (.7502554)^2 = .0521733 \Rightarrow \sigma = .2284147$,

$$m_3 = m_3' - 3m_1'm_2' + 2(m_1')^3$$

$$= .5423512 - 3(.7502554)(.6150564) + 2(.7502554)^3 = .0026151 ,$$

$$g_1 = m_3/(m_2)^{\frac{3}{2}} = .0026151/(.0521733)^{\frac{3}{2}}$$

$$= .2194409 .$$

Notice that the exact mean, variance, and coefficient of skewness of the project completion time can be computed using the following equations: Since duration times of activities A and B are statistically independent;

(i) mean value of the project completion time (T=A+B) equals the mean value of A plus the mean value of B.

Hence,

$$m_1'(T)=m_1'(A) + m_1'(B)$$

or

$$\mu_T = \mu_A + \mu_B \ ,$$

(ii) variance of the project completion time equals the variance of A plus the variance of B.

Hence, $m_2(T)=m_2(A) + m_2(B)$ or $\sigma_T^2 = \sigma_A^2 + \sigma_B^2$,

(iii) third moment about the mean of the project completion time equals the third moment about the mean of A plus the third moment about the mean of B.

Hence,

$$m_3(T) = m_3(A) + m_3(B) \ ,$$

(iv) and finally coefficient of skewness of the project completion time

$$g_1(T) = m_3(T)/[m_2(T)]^{\frac{3}{2}} \ ,$$

where:

m_1' = first moment about the origin = (mean).

m_2 = second moment about the mean = (variance).

m_3 = third moment about the mean.

g_1 = coefficient of skewness = (moment coefficient of skewness).

By using the above equations the exact mean, variance, and coefficient of skewness of the project completion time of example 3 can be computed as follows:

From page 380 we have exact mean for activities A and B; $\mu_A=\mu_B= .375$, exact variance of activities A and B; $\sigma_A^2 = \sigma_B^2 = .0260416$, and exact coefficient of skewness of activities A and B; $g_1(A) = g_1(B) = .3098386$. Therefore, Exact mean of the project completion time;

$$\mu_T = \mu_A + \mu_B$$
$$= .375 + .375 = .75 .$$

Exact variance of the project completion time;

$$\sigma_T^2 = \sigma_A^2 + \sigma_B^2$$
$$= .0260416 + .0260416 = .0520832 .$$

Exact coefficient of skewness of the project completion time;

$$g_1 (T) = m_3(T)/[m_2(T)]^{\frac{3}{2}}$$
$$= (m_3(A) + m_3(B))/[m_2(A) + m_2 (B)]^{\frac{3}{2}} ,$$

since $m_3(A)=m_3(B)$ and $m_2(A)=m_2(B)$, therefore,

$$g_1(T) = (2m_3(A))/[2m_2(A)]^{\frac{3}{2}}$$
$$= m_3(A)/2^{\frac{3}{2}}[m_2(A)]^{\frac{3}{2}} + m_3(A)/2^{\frac{3}{2}}[m_2(A)]^{\frac{3}{2}}$$
$$= 2g_1(A)/2^{\frac{3}{2}} \text{ or } g_1(A)/2^{\frac{1}{2}}$$
$$= .3098386/1.4142135$$
$$= .2190889 .$$

Notice that the difference between exact mean, variance and coefficient of skewness and the mean, variance and coefficient of skewness obtained in second approach (Approximating pdf of the project completion time using interpolation), as shown in Table 8.16 is not significant, therefore, we assume the pdf of the project completion time obtained using second approach to be error free and considered as "true" pdf. In the following the "true" pdf is compared with the approximate pdf of Table 8.14 which is obtained by using 30 cells for each activity.

Table 8.16

	Approximate value	Exact value	% Error
Mean	.7502554	.750	.03
Variance	.0521733	.0520832	.0017
Coefficient of skewness	.2194409	.2190889	.0016

The application of the linear interpolation to Tables 8.14 and 8.15 led to the exclusion of the first 13 and the last 26 realizations of Table 8.15. Table 8.17 has the complete output of the linear interpolation. Figure 8.11 is a digital plot of the first three columns of Table 8.17.

Table 8.17

I	REALIZATION	TRUE PROB.	APROXMTD PROB.	ACTUAL DIFFERENCE
1	.28	.0122627	.0119259	-.0003368
2	.30	.0169922	.0163706	-.0006216
3	.32	.0229393	.0230342	.0000949
4	.34	.0302580	.0305925	.0003345
5	.36	.0390919	.0399308	.0008389
6	.38	.0495699	.0513474	.0017775
7	.40	.0618017	.0638184	.0020167
8	.42	.0758738	.0796599	.0037861
9	.44	.0918469	.0966992	.0048523
10	.46	.1097537	.1160315	.0062778
11	.48	.1295968	.1377370	.0081402
12	.50	.1513286	.1606392	.0092906
13	.52	.1749511	.1868584	.0119073
14	.54	.2003172	.2140475	.0137303
15	.56	.2273321	.2431552	.0158231
16	.58	.2558552	.2739123	.0180571
17	.60	.2857235	.3056586	.0199351
18	.62	.3165490	.3392261	.0226771
19	.64	.3487508	.3731743	.0244235
20	.66	.3815013	.4078719	.0263706
21	.68	.4147883	.4429504	.0281621
22	.70	.4483896	.4782216	.0298320
23	.72	.4820829	.5133026	.0312197
24	.74	.5156497	.5480567	.0324070
25	.76	.5488783	.5821615	.0332832
26	.78	.5815679	.6153461	.0337782
27	.80	.6135303	.6480728	.0345425
28	.82	.6445931	.6788055	.0342124
29	.84	.6746014	.7087090	.0341076
30	.86	.7034197	.7369695	.0335498
31	.88	.7309320	.7635100	.0325780
32	.90	.7570438	.7891870	.0321432
33	.92	.7816812	.8120308	.0303496
34	.94	.8047910	.8338936	.0291026
35	.96	.8263405	.8538068	.0274663
36	.98	.8463159	.8718840	.0255681
37	1.00	.8647217	.8890372	.0243155
38	1.02	.8815788	.9035056	.0219268
39	1.04	.8969230	.9171345	.0202115
40	1.06	.9108036	.9290926	.0182890
41	1.08	.9232807	.9395880	.0163073
42	1.10	.9344241	.9493482	.0149241
43	1.12	.9443106	.9571439	.0128333
44	1.14	.9530230	.9643679	.0113449

I	REALIZATION	TRUE PROB.	APRXMTD PROB.	ACTUAL DIFFERENCE
45	1.16	.9606476	.9704607	.0098131
46	1.18	.9672725	.9756231	.0083506
47	1.20	.9729866	.9803218	.0073352
48	1.22	.9778778	.9838563	.0059785
49	1,24	.9820315	.9869806	.0049491
50	1.26	.9855304	.9892920	.0037616
51	1.28	.9884524	.9914134	.0028610

Table 8.17 (Concluded)

Comparison of the Exact and the Approximate Probability Distribution Functions of Example 3.

The Average of the Absolute Values of the Deviations = .017182

The Maximum of the Absolute Values of the Deviations = .0345425. It is No. 27.

Number of Positive Deviates = 49

Number of Negative Deviates = 2

Example 4:

Let us assume activities A and B of example 3 are parallel, then the pdf of T=max{A,B} can be computed as shown in Tables 8.18 and 8.19. Table 8.18 shows the approximate pdf of the project completion time using 30 cells for activities A and B.

Table 8.19 shows the approximate pdf of the project completion time using 100 cells for activities A and B, this pdf is assumed to be correct (error free) for calculation purposes.

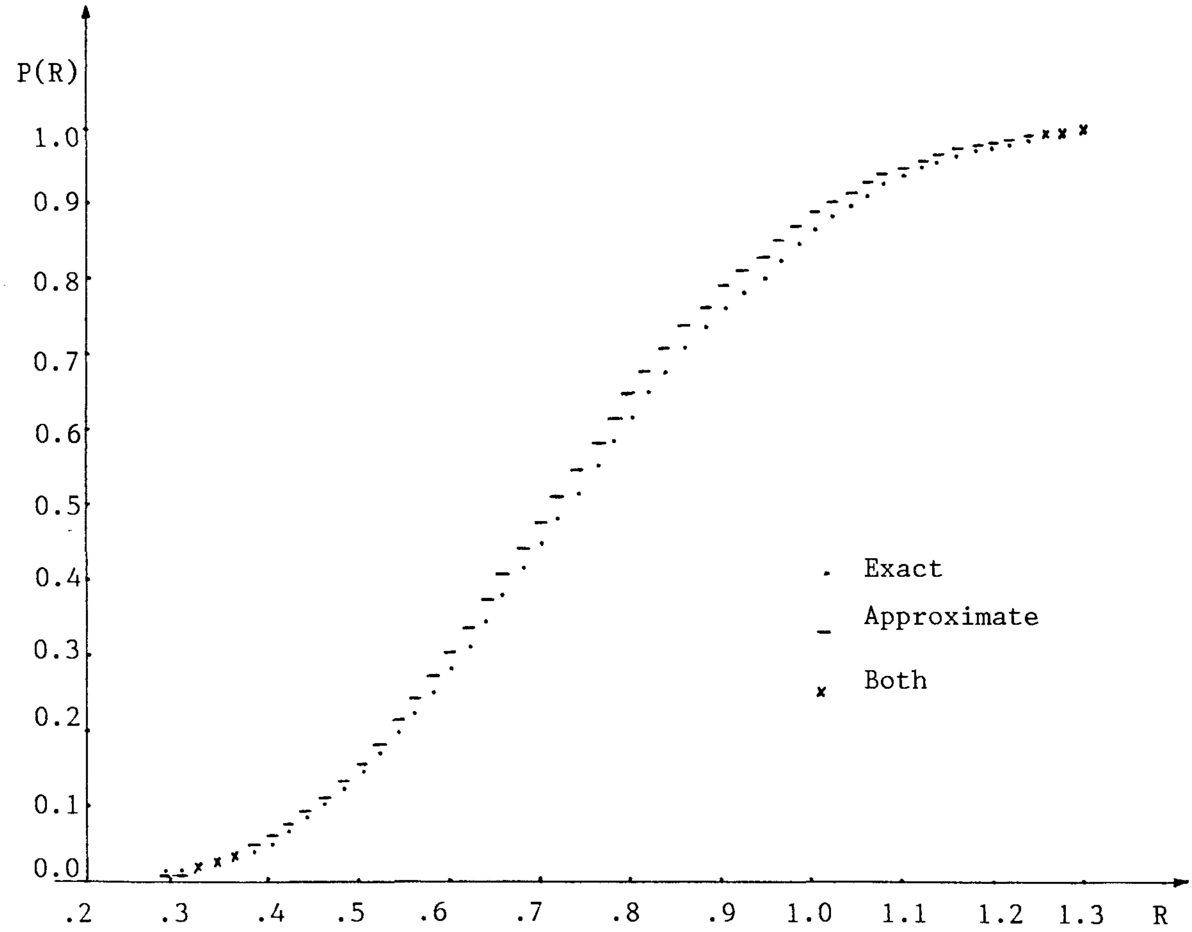


Figure 8.11

Comparison of the Exact and the Approximate Probability Distribution Functions of Example 3.

Table 8.18

w=x*x	p(w)	F(w)	w=x.x	p(w)	F(w)
.0333	.0000024	.0000024	.5333	.0588286	.7285889
.0666	.0001109	.0001133	.5667	.0828223	.8114112
.1000	.0008836	.0009969	.6000	.0543736	.8657848
.1333	.0034258	.0044227	.6333	.0427539	.9085387
.1667	.0087587	.0131814	.6667	.0323343	.9408730
.2000	.0174306	.0306120	.7000	.0232275	.9641005
.2333	.0293221	.0599341	.7333	.0155459	.9796464
.2667	.0432291	.1031632	.7667	.0098308	.9894772
.3000	.0577541	.1609173	.8000	.0057066	.9951838
.3333	.0711961	.2321134	.8333	.0029095	.9980933
.3667	.0819914	.3141048	.8667	.0013055	.9993988
.4000	.0891071	.4032119	.9000	.0004698	.9998686
.4333	.0916663	.4948782	.9333	.0001136	.9999822
.4667	.0901153	.5849935	.9667	.0000146	.9999968
.5000	.0847668	.6697603	1.0000	.0000004	.9999972

$\mu = m_1' = .4582186$, $m_2' = .229189$, $m_3' = .1231471$,

$\sigma^2 = m_2' - (m_1')^2 = .229189 - (.4582186)^2 = .0192248 \implies \sigma = .1386535$,

$m_3 = m_3' - 3m_1'm_2' + 2(m_1')^3 = .1231471 - 3(.4582186)(.229189) + (.4582186)^3 = .0005102$,

$g_1 = m_3/(m_2)^{\frac{3}{2}} = .0005102/(.0192248)^{\frac{3}{2}} = .0005102/.0026655 = 0.1914087$.

Table 8.19

v=x.x	p(v)	F(v)	v=x.x	p (v)	F(v)
.01	.0000000	.0000000	.51	.0251262	.6233317
.02	.0000000	.0000000	.52	.0245678	.6478995
.03	.0000007	.0000007	.53	.0239341	.6718336
.04	.0000032	.0000039	.54	.0232312	.6950648
.05	.0000102	.0000141	.55	.0224666	.7175314
.06	.0000255	.0000396	.56	.0216476	.7391790
.07	.0000542	.0000938	.57	.0207814	.7599604
.08	.0001025	.0001963	.58	.0198760	.7798364
.09	.0001774	.0003737	.59	.0189390	.7987754
.10	.0002863	.0006600	.60	.0179776	.8167530
.11	.0004369	.0010969	.61	.0169995	.8337525
.12	.0006368	.0017337	.62	.0160115	.8497640
.13	.0008934	.0026271	.63	.0150210	.8647850
.14	.0012129	.0038400	.64	.0140344	.8788194
.15	.0016013	.0054413	.65	.0130576	.8918770
.16	.0020627	.0075040	.66	.0120964	.9039734
.17	.0026002	.0101042	.67	.0111564	.9151298
.18	.0032163	.0133205	.68	.0102422	.9253720
.19	.0039107	.0172312	.69	.0093580	.9347300
.20	.0046822	.0219134	.70	.0085079	.9432379
.21	.0055285	.0274419	.71	.0076948	.9509327
.22	.0064452	.0338871	.72	.0069215	.9578542
.23	.0074267	.0413138	.73	.0061907	.9640449
.24	.0084665	.0497803	.74	.0055031	.9695480
.25	.0095564	.0593367	.75	.0048608	.9744088
.26	.0106876	.0700243	.76	.0042641	.9786729
.27	.0118505	.0818748	.77	.0037135	.9823864
.28	.0130344	.0949092	.78	.0032086	.9855950
.29	.0142287	.1091379	.79	.0027489	.9883439
.30	.0154220	.1245599	.80	.0023337	.9906776
.31	.0166032	.1411631	.81	.0019611	.9926387
.32	.0177611	.1589242	.82	.0016303	.9942690
.33	.0188846	.1778088	.83	.0013388	.9956078
.34	.0199320	.1977720	.84	.0010845	.9966923
.35	.0209869	.2187589	.85	.0008657	.9975580
.36	.0219460	.2407049	.86	.0006791	.9982371
.37	.0228322	.2635371	.87	.0005226	.9987597
.38	.0236378	.2871749	.88	.0003934	.9991531
.39	.0243558	.3115307	.89	.0002885	.9994416
.40	.0249806	.3365113	.90	.0002052	.9996468
.41	.0255074	.3620187	.91	.0001408	.9997876
.42	.0259326	.3879513	.92	.0000922	.9998798
.43	.0262538	.4142051	.93	.0000574	.9999372
.44	.0264695	.4406746	.94	.0000332	.9999704

Table 8.19 (Concluded)

v=x.x	p(v)	F(v)	v=x.x	p(v)	F(v)
.45	.0265795	.4672541	.95	.0000174	.9999878
.46	.0265846	.4938387	.96	.0000080	.9999958
.47	.0264860	.5203247	.97	.0000030	.9999988
.48	.0262872	.5466119	.98	.0000008	.9999996
.49	.0259911	.5726030	.99	.0000002	.9999998
.50	.0256025	.5982055	1.00	.0000002	1.0000000

$\mu = m_1' = .4717789$, $m_2' = .2427476$, $m_3' = .1339443$,

$\sigma^2 = m_2' - (m'_1)^2 = .2427476 - (.4717789)^2 = .0201723 \implies \sigma .1420292$,

$m_3 = m_3' - 3m_1'm_2' + 2(m_1')^3 = .1339443 - .3435695 + .2100126 = .0003874$,

$$g_1 = m_3/(m_2)^{\frac{3}{2}} = .0003874/(.0201723)^{\frac{3}{2}}$$

$$= .1352181.$$

The application of the linear interpolation to Tables 8.18 and 8.19 led to the exclusion of the first 16 and last 21 realizations of Table 8.19. Table 8.20 has the complete output of the linear interpolation. Figure 8.12 is a digital plot of the first three columns of Table 8.20.

I	REALIZATION	TRUE PROB.	APRXMTD PROB.	ACTUAL DIFFERENCE
1	.17	.01014042	.0149081	.0048039
2	.20	.02191340	.0306120	.0086986
3	.24	.04978030	.0686047	.0188244
4	.27	.08187480	.1088839	.0270091
5	.30	.12455990	.1609173	.0363574
6	.34	.19777200	.2485595	.0507875
7	.37	.26353710	.3229336	.0593965
8	.40	.33651130	.4032119	.0667006
9	.44	.44067460	.5129530	.0722784
10	.47	.52032470	.5933928	.0730681
11	.50	.59820550	.6697603	.0715548
12	.54	.69506480	.7452026	.0501378
13	.57	.75996040	.8167983	.0568379
14	.60	.81675300	.8657848	.0490318
15	.64	.87881940	.9150237	.0362043
16	.67	.91512980	.9431733	.0280435
17	.70	.94323790	.9641005	.0208626
18	.74	.96954800	.9816164	.0120684
19	.77	.98238640	.9900417	.0076553
20	.80	.99067760	.9951838	.0045062

Table 8.20

Comparison of the Exact and the Approximate Probability Distribution Functions of Example 4.

The Average of the Absolute Values of the Deviations = .0372413

The Maximum of the Absolute Values of the Deviations = .0730681. It is No. 10.

Number of Positive Deviates = 20.

Number of Negative Deviates = 0.

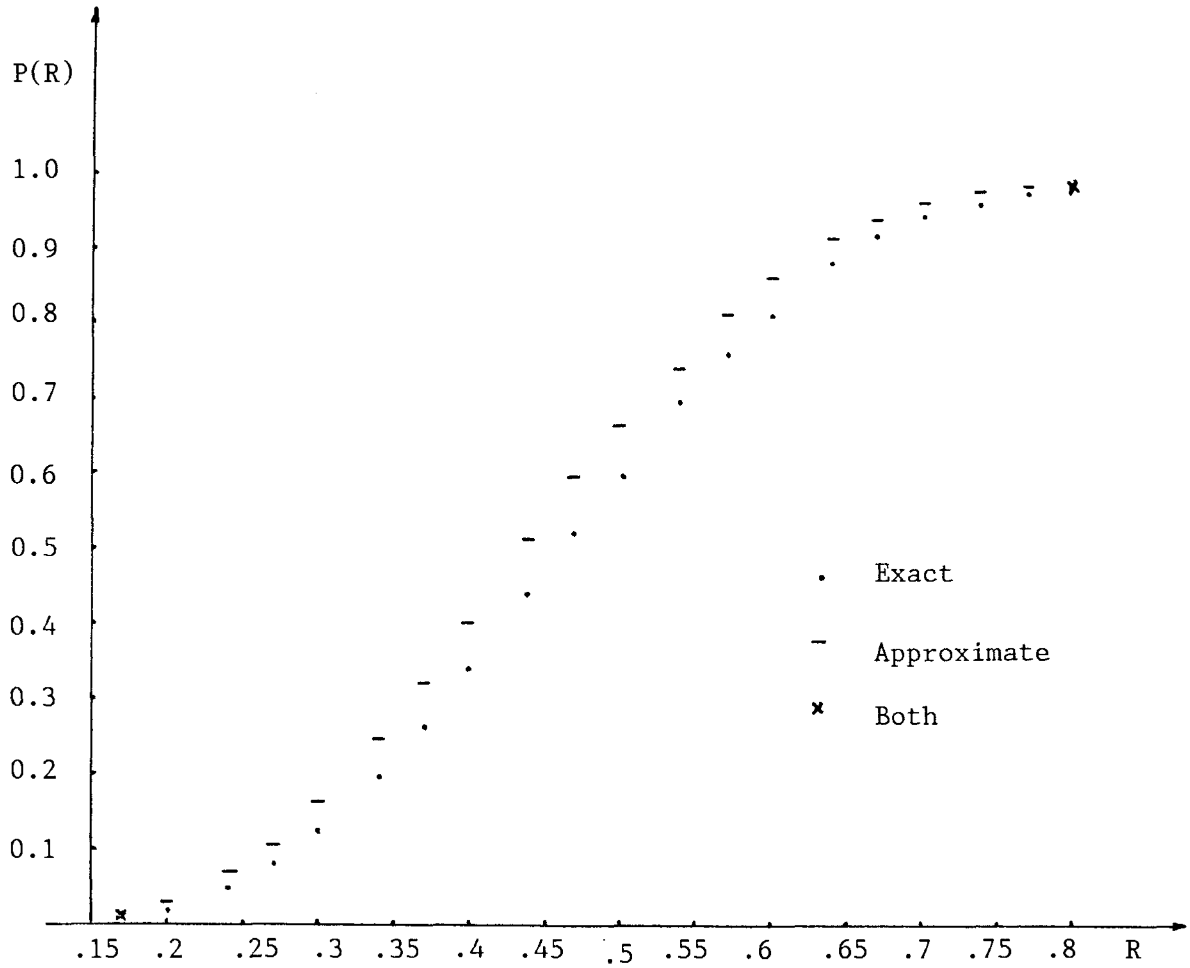


Figure 8.12

Comparison of the Exact and the Approximate Probability Distribution Functions of Example 4.

Example 5:

Consider again network configuration of Figure 8.9, assume activities A and B have Beta distribution with parameters r=2 and s=8, notice that in this case the activities have extreme skewed distribution, recall that in example 3 the activities had moderate skewed distribution.

The extreme skewed distribution with parameters r=2 and s=8 gave rise to a distribution with a range of 0 and .1 and a mean of .2 for activities A and B as shown in Figure 8.13.

Same as example 3 to approximate the Beta by a discrete distribution, we first divide the range of the distribution into 30 cells, then from the tables of the Incomplete Beta - Function (Pearson, 1956), with these parameters we obtain the information in Table 8.21.

The tables of the Incomplete Beta-Function with 100 cells for r=2 and s=8 as shown in Table 8.22 taken as being correct for calculation purposes.

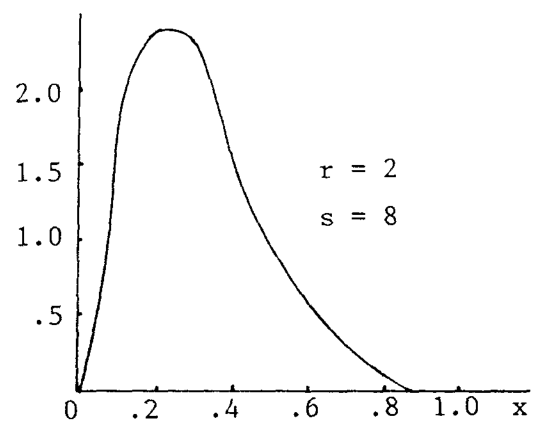


Figure 8.13

Table 8.21

x	p(x)	F(x)	x	p(x)	F(x)
.0333	.0416989	.0416989	.5333	.0048717	.9932666
.0667	.0966020	.1383009	.5667	.0029732	.9962398
.1000	.1221857	.2604866	.6000	.0017309	.9979707
.1333	.1281315	.3886181	.6333	.0009394	.9989101
.1667	.1220010	.5106191	.6667	.0004890	.9993991
.2000	.1082053	.6188244	.7000	.0001363	.9996354
.2333	.0921844	.7110088	.7333	.0001021	.9997375

Table 8.21 (Concluded)

x	p(x)	F(x)	x	p(x)	F(x)
.2667	.0756150	.7866238	.7667	.0000406	.9997781
.3000	.0600106	.8466344	.8000	.0000141	.9997922
.3333	.0458183	.8924527	.8333	.0000039	.9997961
.3667	.0342376	.9266903	.8667	.0000006	.9997967
.4000	.0248352	.9515255	.9000	.0000001	.9997968
.4333	.0172955	.9688210	.9333	.0000000	.9997968
.4667	.0117987	.9806197	.9667	.0000000	.9997968
.5000	.0077752	.9883949	1.0000	.0000000	.9997968

$\mu=m_1' = .1999617$, $m_2' = .0526811$, $m_3' = .0168492$,

$\sigma^2=m_2' - (m_1')^2 = .0526811 - (.1999617)^2 = .0126965 \implies \sigma=.1126787$,

$m_3=m'_3 - 3m_1'm_2' + 2(m_1')^3 = .0168492 - .0316026 + .0159906 =.0012372$,

$$g_1=m_3/(m_2)^{\frac{3}{2}} = .0012372/(.0126965)^{\frac{3}{2}} = .8648119 \quad .$$

Table 8.22

x	p(x)	F(x)	x	p(x)	F(x)
.01	.0034357	.0034357	.47	.0042017	.9703644
.02	.0096792	.0131149	.48	.0037611	.9741255
.03	.0150433	.0281582	.49	.0033567	.9774822
.04	.0196076	.0477658	.50	.0029866	.9804688
.05	.0234456	.0712114	.51	.0026489	.9831177
.06	.0266266	.0978380	.52	.0023418	.9854595
.07	.0292144	.1270524	.53	.0020634	.9875229
.08	.0312686	.1583210	.54	.0018118	.9893347
.09	.0328447	.1911657	.55	.0015851	.9909198
.10	.0339933	.2251590	.56	.0013818	.9923016
.11	.0347623	.2599213	.57	.0011998	.9935014
.12	.0351950	.2951163	.58	.0010376	.9945390
.13	.0353319	.3304482	.59	.0008936	.9954326
.14	.0352098	.3656580	.60	.0007663	.9961989
.15	.0348628	.4005208	.61	.0006541	.9968530
.16	.0343222	.4348430	.62	.0005557	.9974087
.17	.0336160	.4684590	.63	.0004698	.9978785
.18	.0327706	.5012296	.64	.0003950	.9982735
.19	.0318093	.5330389	.65	.0003303	.9986038
.20	.0307535	.5637924	.66	.0002747	.9988785
.21	.0296224	.5934148	.67	.0002270	.9991055
.22	.0284336	.6218484	.68	.0001864	.9992919
.23	.0272025	.6490509	.69	.0001520	.9994439
.24	.0259429	.6749938	.70	.0001231	.9995670
.25	.0246675	.6996613	.71	.0000988	.9996658
.26	.0233867	.7230480	.72	.0000788	.9997446
.27	.0221106	.7451586	.73	.0000622	.9998068

Table 8.22 (Concluded)

x	p(x)	F(x)	x	p(x)	F(x)
.28	.0208473	.7660059	.74	.0000487	.9998555
.29	.0196039	.7856098	.75	.0003877	.9998932
.30	.0183870	.8039968	.76	.0000289	.9999221
.31	.0172014	.8211982	.77	.0000218	.9999439
.32	.0160517	.8372499	.78	.0000164	.9999603
.33	.0149415	.8521914	.79	.0000120	.9999723
.34	.0138735	.8660649	.80	.0000088	.9999811
.35	.0128501	.8789150	.81	.0000062	.9999873
.36	.0118727	.8907877	.82	.0000044	.9999917
.37	.0109426	.9017303	.83	.0000030	.9999947
.38	.0100603	.9117906	.84	.0000020	.9999967
.39	.0092260	.9210166	.85	.0000013	.9999980
.40	.0084395	.9294561	.86	.0000008	.9999988
.41	.0077005	.9371566	.87	.0000006	.9999994
.42	.0070079	.9441645	.88	.0000003	.9999997
.43	.0063610	.9505255	.89	.0000001	.9999998
.44	.0057583	.9562838	.90	.0000001	.9999999
.45	.0051986	.9614824	.91	.0000001	1.0000000
.46	.0046803	.9661627	.92	.0000000	1.0000000

$\mu=.2$, $\sigma = .12060454$, $g_1=.8291561$.

Exact mean, variance, and coefficient of skewness of each activity can be computed as follows:

$$\text{Mean} = r/(r+s)$$
$$= 2/(2+8)$$
$$= .2 \; ,$$

$$\text{Variance} = rs/(r+s+1)(r+s)^2$$
$$= 2x8/(2+8+1)(2+8)^2$$
$$= .0145454 \quad ==> \quad \sigma = .1206043 \; ,$$

$$\text{Coefficient of skewness} = 2(s-r)(r+s+1)^{\frac{1}{2}}/(r+s+2)(rs)^{\frac{1}{2}}$$
$$= 2(8-2)(2+8+1)^{\frac{1}{2}}/(2+8+2)(2x8)^{\frac{1}{2}}$$
$$= .8291561 \, .$$

In this example the pdf of project completion time using 30 cells for input distribution is compared with the pdf of project completion time obtained by complete convolution operation of the duration times of activites A and B using 100 cells for each activity. The latter is considered to be correct (error free) for calculation purposes.

(i) Approximating the pdf of the project completion time using 30 cells for input distribution.

The approximate pdf of the project completion time using 30 cells can be obtained from Table 8.21 as shown in Table 8.23.

Table 8.23

y=x*x	p(y)	F(y)	y=x*x	p(y)	F(y)
.0667	.0017387	.0017387	.7333	.0113078	.9730333
.1000	.0080562	.0097949	.7667	.0082546	.9812879
.1333	.0195218	.0293167	.8000	.0059086	.9871965
.1667	.0342924	.0636091	.8333	.0041488	.9913453
.2000	.0498592	.1134683	.8667	.0028572	.9942025
.2333	.0639086	.1773769	.9000	.0019296	.9961321
.2667	.0748244	.2522013	.9333	.0012778	.9974099
.3000	.0818226	.3340239	.9667	.0008296	.9982395
.3333	.0847540	.4187779	1.0000	.0005276	.9987671
.3667	.0839188	.5026967	1.0333	.0003284	.9990955
.4000	.0799506	.5826473	1.0667	.0001998	.9992953
.4333	.0736608	.6563081	1.1000	.0001184	.9994137
.4667	.0658528	.7221609	1.1333	.0000688	.9994825
.5000	.0572754	.7794363	1.1667	.0000386	.9995211
.5333	.0485694	.8280057	1.2000	.0000210	.9995421
.5667	.0402152	.8682209	1.2333	.0000108	.9995529
.6000	.0325494	.9007703	1.2667	.0000054	.9995583
.6333	.0257784	.9265487	1.3000	.0000026	.9995609
.6667	.0199904	.9465391	1.3333	.0000010	.9995619
.7000	.0151864	.9617255	1.3667	.0000002	.9995621

$\mu=m_1'= .3998142$, $m_2' = .1852844$, $m_3' = .0968716$,

$\sigma^2=m_2-(m_1')^2 = .1852844 - (.3998142)^2 = .0254331$ ==> $\sigma = .1594775$,

$m_3=m_3' - 3m_1'm_2' +2(m_1')^3 = .0968716 - .222238 + .1278216 = .0024552$,

$g_1=m_3/(m_2)^{\frac{3}{2}} = .6053254$.

(ii) Approximating the pdf of the project completion time using 100 cells for input distribution.

The approximate pdf of the project completion time using 100 cells can be obtained from Table 8.22 as shown in Table 8.24.

Table 8.24

z=x*x	p(z)	F(z)	z=x*x	p(z)	F(z)
.02	.0000118	.0000118	.66	.0072936	.9182698
.03	.0000664	.0000782	.67	.0068040	.9250738
.04	.0001968	.0002750	.68	.0063356	.9314094
.05	.0004258	.0007008	.69	.0058908	.9373002
.06	.0007666	.0014674	.70	.0054666	.9427668
.07	.0012264	.0026938	.71	.0050626	.9478294
.08	.0018056	.0044994	.72	.0046846	.9525140
.09	.0025006	.0070000	.73	.0043252	.9568392
.10	.0033032	.0103032	.74	.0038976	.9608268
.11	.0042038	.0145070	.75	.0036698	.9644966
.12	.0051896	.0196966	.76	.0033718	.9678684
.13	.0062470	.0259436	.77	.0030988	.9709672
.14	.0073610	.0333046	.78	.0028308	.9737980
.15	.0085172	.0418218	.79	.0025880	.9763860
.16	.0097008	.0515226	.80	.0023624	.9787484
.17	.0108966	.0624192	.81	.0021516	.9809000
.18	.0120908	.0745100	.82	.0019572	.9828572
.19	.0132706	.0877806	.83	.0017766	.9846338
.20	.0144232	.1022038	.84	.0016108	.9862446
.21	.0155382	.1177420	.85	.0014576	.9877022
.22	.0166048	.1343468	.86	.0013178	.9890200
.23	.0176150	.1519618	.87	.0011876	.9902076
.24	.0185598	.1705216	.88	.0010696	.9912772
.25	.0194338	.1899554	.89	.0009614	.9922386
.26	.0202312	.2101866	.90	.0008632	.9931018
.27	.0209476	.2311342	.91	.0007736	.9938754
.28	.0215800	.2527142	.92	.0006922	.9945676
.29	.0221278	.2748420	.93	.0006178	.9951854
.30	.0225856	.2974276	.94	.0005504	.9957358
.31	.0229610	.3203886	.95	.0004902	.9962260
.32	.0232472	.3436358	.96	.0004348	.9966608
.33	.0234488	.3670846	.97	.0003858	.9970466
.34	.0235674	.3906520	.98	.0003414	.9973880
.35	.0236052	.4142572	.99	.0003012	.9976892
.36	.0235660	.4378232	1.00	.0002644	.9979536
.37	.0234532	.4612764	1.01	.0002340	.9981876
.38	.0232706	.4845470	1.02	.0002044	.9983920
.39	.0230222	.5075692	1.03	.0001794	.9985714
.40	.0227124	.5302816	1.04	.0001560	.9987274
.41	.0223466	.5526282	1.05	.0001360	.9988634
.42	.0219292	.5745574	1.06	.0001236	.9989870
.43	.0214646	.5960220	1.07	.0001040	.9990910
.44	.0209590	.6169810	1.08	.0000894	.9991804
.45	.0204154	.6373964	1.09	.0000770	.9992574
.46	.0198396	.6572360	1.10	.0000664	.9993238
.47	.0192362	.6764722	1.11	.0000570	.9993808
.48	.0186098	.6950820	1.12	.0000474	.9994282
.49	.0179642	.7130462	1.13	.0000418	.9994700
.50	.0173088	.7303550	1.14	.0000398	.9995098
.51	.0166332	.7469882	1.15	.0000300	.9995398
.52	.0159560	.7629442	1.16	.0000254	.9995652
.53	.0152752	.7782194	1.17	.0000216	.9995868
.54	.0145940	.7928134	1.18	.0000178	.9996046

Table 8.24 (concluded)

z=x*x	p(z)	F(z)	z=x*x	p(z)	F(z)
.55	.0139160	.8067294	1.19	.0000146	.9996192
.56	.0132446	.8199740	1.20	.0000118	.9996310
.57	.0125804	.8325544	1.21	.0000100	.9996410
.58	.0119272	.8444816	1.22	.0000082	.9996492
.59	.0112878	.8557694	1.23	.0000060	.9995520
.60	.0106620	.8664314	1.24	.0000052	.9996604
.61	.0100532	.8764846	1.25	.0000040	.9996644
.62	.0094620	.8859466	1.26	.0000030	.9996674
.63	.0088896	.8948362	1.27	.0000022	.9996696
.64	.0083362	.9031724	1.28	.0000016	.9996712
.65	.0078038	.9109762	1.29	.0000012	.9996724
			1.30	.0000006	.9996730

$\mu = m_1' = .4097325$, $m'_2 = .1969569$, $m_3' = .1073794$,

$\sigma^2 = m_2 = .0290762 \Longrightarrow \sigma = .1705174$, $m_3 = .0028528$,

$g_1 = m_3 / (m_2)^{\frac{3}{2}} = .0028528 / (.0290762)^{\frac{3}{2}} = .5754049$.

The exact mean, variance, and coefficient of skewness can be computed as follows:

From page 393 we have exact mean for activities A and B, $\mu_A = \mu_B = .2$; exact variance of activites A and B, $\sigma_A^2 = \sigma_B^2 = .0145454$, and exact coefficient of skewness of activities A and B, $g_1(A) = g_1(B) = .8291561$.

Therefore,

Exact mean of the project completion time;

$$\mu_T = \mu_A + \mu_B$$

$$= .2 + .2 = .4 .$$

Exact variance of the project completion time,

$$\sigma_T^2 = \sigma_A^2 + \sigma_B^2$$

$$= .0145454 + .0145454 = .0290908 .$$

Exact coefficient of skewness of the project completion time;

Since $m_3(A)=m_3(B)$ and $m_2(A)=m_2(B)$, therefore ,

$$g_1(T)=g_1(A)/2^{\frac{1}{2}}$$

$$= .5863019 .$$

The application of the linear interpolation to Tables 8.23 and 8.24 led to the exclusion of the first 8 and last 44 realizations of Table 8.24. Table 8.25 has the complete output of the linear interpolation. Figure 8.14 is a digital plot of the first three columns of Table 8.25.

I	REALIZATION	TRUE PROB.	APRXMTD PROB.	ACTUAL DIFFERENCE
1	.1	.0103032	.0097949	-.0005083
2	.14	.0333046	.0361939	.0028893
3	.17	.0624192	.0685490	.0061298
4	.20	.1022038	.1134683	.0112645
5	.24	.1705216	.1923858	.0218642
6	.27	.2311342	.2603094	.0291752
7	.30	.2974276	.3340239	.0365963
8	.34	.3906520	.4356102	.0449582
9	.37	.4612764	.5106186	.0493422
10	.40	.5302816	.5826473	.0523657
11	.44	.6169810	.6695176	.0525366
12	.47	.6764722	.7278365	.0513643
13	.50	.7303550	.7794363	.0490813
14	.54	.7928134	.8360715	.0432581
15	.57	.8325544	.8714461	.0388917
16	.60	.8664314	.9007703	.0343389
17	.64	.9031724	.9305576	.0273852
18	.67	.9250738	.9480436	.0229698
19	.70	.9427668	.9617255	.0189587
20	.74	.9608268	.9746889	.0138621
21	.77	.9709672	.9818704	.0109032
22	.80	.9787484	.9871965	.0084481
23	.84	.9862446	.9919171	.0056725

Table 8.25

Comparison of the Exact and the Approximate Probability Distribution Functions of Example 5.

The Average of the Absolute Values of the Deviations = .0275114

The Maximum of the Absolute Values of the Deviations = .0525366.

It is No.11.

Number of Positive Deviates = 22.

Number of Negative Deviates = 1.

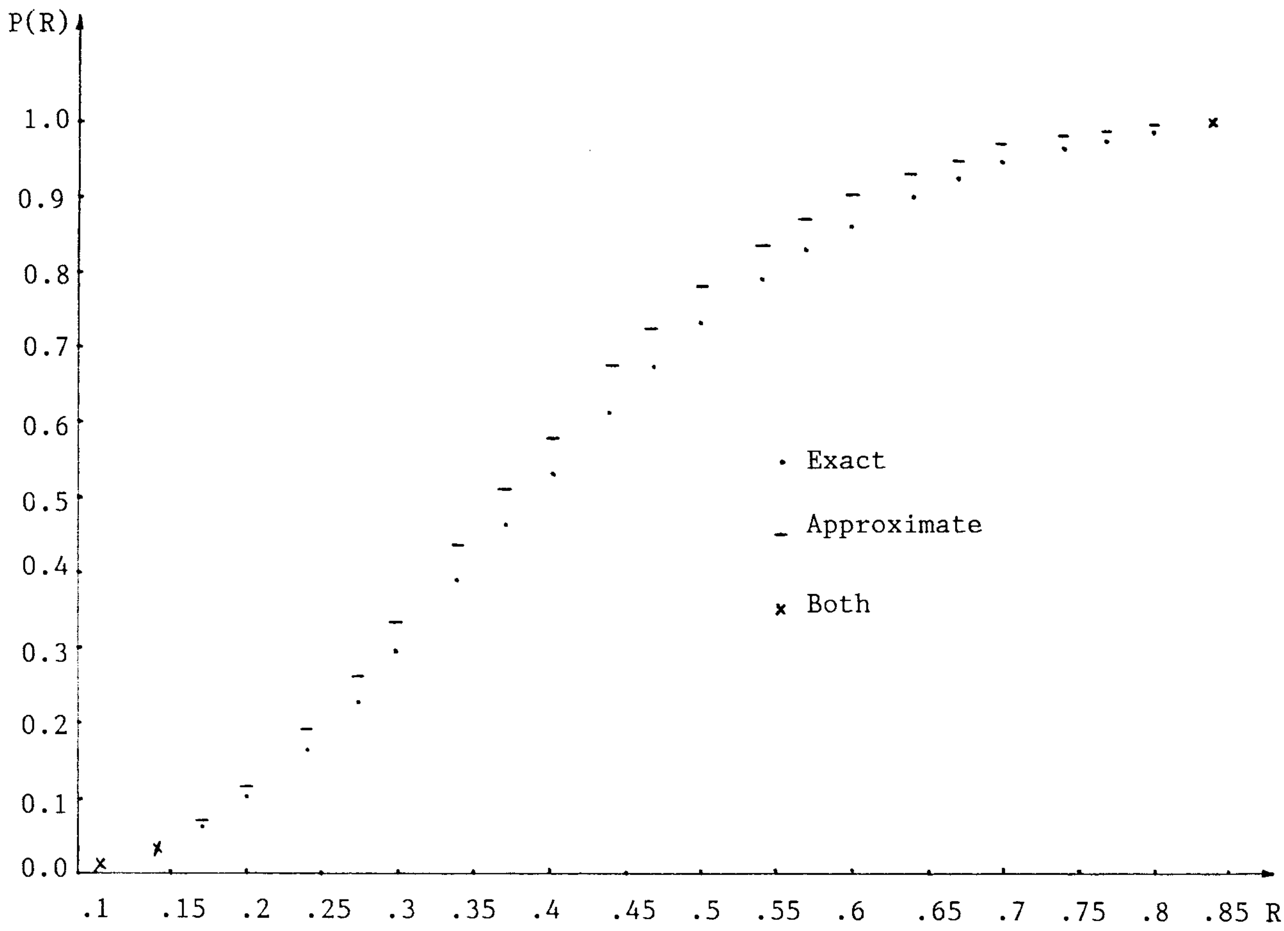


Figure 8.14

Comparison of the Exact and the Approximate Probability Distribution Functions of Example 5.

Example 6:

Let us assume activities A and B of example 5 are parallel, then the pdf of the project completion time T=max{A,B} can be computed as shown in Tables 8.26 and 8.27.

Table 8.26 shows the approximate pdf of the project completion time using 30 cells for activities A and B.

Table 8.27 shows the approximate pdf of the project completion time using 100 cells for activities A and B, this pdf is assumed to be correct (error free) for calculation purposes.

Table 8.26

w=x.x	p(w)	F(w)	w=x.x	p(w)	F(aw)
.0333	.0017387	.0017387	.5000	.0153095	.9769244
.0667	.0173884	.0191271	.5333	.0096541	.9865785
.1000	.0487261	.0678532	.5667	.0059152	.9924937
.1333	.0831708	.1510240	.6000	.0034518	.9959455
.1667	.1097078	.2607318	.6333	.0018758	.9978213
.2000	.1222118	.3829436	.6667	.0009772	.9987985
.2333	.1225899	.5055335	.7000	.0004724	.9992709
.2667	.1132435	.6187770	.7333	.0002041	.9994750
.3000	.0980128	.7167898	.7667	.0000812	.9995562
.3333	.0796820	.7964718	.8000	.0000282	.9995844
.3667	.0622831	.8587549	.8333	.0000078	.9995922
.4000	.0466458	.9054007	.8667	.0000012	.9995934
.4333	.0332134	.9386141	.9000	.0000002	.9995936
.4667	.0230008	.9616149	.9333	.0000000	.9995936

$\mu=m_1' = .2665017$, $m_2' = .0821502$, $m_3' = .028809$,

$\sigma^2=m_2 = .0111271$ ==> $\sigma = .105485$,

$m_3 = .0009849$,

$$g_1 = m_3/(m_2)^{\frac{3}{2}} = .8391411 .$$

Table 8.27

v=x.x	p(v)	F(v)	v=x.x	p(v)	F(v)
.01	.0000118	.0000118	.47	.0081367	.9416070
.02	.0001602	.0001720	.48	.0073134	.9489204
.03	.0006208	.0007928	.49	.0065510	.9554714
.04	.0014887	.0022815	.50	.0058476	.9613190
.05	.0027895	.0050710	.51	.0052014	.9665204
.06	.0045012	.0095722	.52	.0046100	.9711304
.07	.0065701	.0161423	.53	.0040710	.9752014
.08	.0089232	.0250655	.54	.0035817	.9787831
.09	.0114788	.0365443	.55	.0031389	.9819220
.10	.0141522	.0506965	.56	.0027404	.9846624
.11	.0168625	.0675590	.57	.0023826	.9870450
.12	.0195346	.0870936	.58	.0020628	.9891078
.13	.0221024	.1091960	.59	.0017782	.9908860
.14	.0245097	.1337057	.60	.0015262	.9924122
.15	.0267112	.1604169	.61	.0013037	.9937159
.16	.0286715	.1890884	.62	.0011082	.9948241
.17	.0303654	.2194538	.63	.0009374	.9957615
.18	.0317773	.2512311	.64	.0007884	.9965499
.19	.0328993	.2841304	.65	.0006596	.9972095
.20	.0337314	.3178618	.66	.0005487	.9977582
.21	.0342793	.3521411	.67	.0004536	.9982118
.22	.0345543	.3866954	.68	.0003725	.9985843

Table 8.27 (Concluded)

v=x.x	p(v)	F(v)	v=x.x	p(v)	F(v)
.23	.0345716	.4212670	.69	.0003038	.9988881
.24	.0343496	.4556166	.70	.0002460	.9991341
.25	.0339093	.4895259	.71	.0001976	.9993317
.26	.0332725	.5227984	.72	.0001575	.9994892
.27	.0324629	.5552613	.73	.0001244	.9996136
.28	.0315037	.5867650	.74	.0000974	.9997110
.29	.0304177	.6171827	.75	.0000754	.9997864
.30	.0292281	.6464108	.76	.0000578	.9998442
.31	.0279556	.6743664	.77	.0000436	.9998878
.32	.0266209	.7009873	.78	.0000328	.9999206
.33	.0252428	.7262301	.79	.0000240	.9999446
.34	.0238383	.7500684	.80	.0000176	.9999622
.35	.0224231	.7724915	.81	.0000124	.9999746
.36	.0210112	.7935027	.82	.0000088	.9999834
.37	.0196148	.8131175	.83	.0000060	.9999894
.38	.0182445	.8313620	.84	.0000040	.9999934
.39	.0169095	.8482715	.85	.0000026	.9999960
.40	.0156171	.8638886	.86	.0000016	.9999976
.41	.0143738	.8782624	.87	.0000012	.9999988
.42	.0131842	.8914466	.88	.0000006	.9999994
.43	.0120521	.9034987	.89	.0000002	.9999996
.44	.0109800	.9144787	.90	.0000002	.9999998
.45	.0099697	.9244484	.91	.0000002	1.0000000
.46	.0090219	.9334703	.92	.0000000	1.0000000

$\mu = m_1' = .271869$, $m_2' = .0881726$, $m_3' = .0322659$,

$\sigma^2 = m_2 = .0142599 \implies \sigma = .1194148$,

$m_3 = .0005408$,

$$g_1 = m_3 / (m_2)^{\frac{3}{2}} = .3175945 .$$

The application of the linear interpolation to Tables 8.26 and 8.27 led to the exclusion of the first 6 and last 35 realizations of Table 8.27. Table 8.28 has the complete output of the linear interpolation. Figure 8.15 is a digital plot of the first three columns of Table 8.28.

I	REALIZATION	TRUE PROB.	APRXMTD PROB.	ACTUAL DIFFERENCE
1	.07	.0161423	.0239529	.0078106
2	.10	.0506965	.0678532	.0171567
3	.14	.1337057	.1730299	.0393242
4	.17	.2194538	.2728399	.0533861
5	.20	.3178618	.3829436	.0650818
6	.24	.4556166	.5282490	.0726324
7	.27	.5552613	.6284887	.0732274
8	.30	.6464108	.7167898	.0703790
9	.34	.7500684	.8089628	.0588944
10	.37	.8131175	.8633765	.0502590
11	.40	.8638886	.9054007	.0415121
12	.44	.9144787	.9432278	.0287491
13	.47	.9416070	.9631314	.0215244
14	.50	.9613190	.9769244	.0156054
15	.54	.9787830	.9877641	.0089811
16	.57	.9870450	.9928330	.0057880

Table 8.28
Comparison of the Exact and the Approximate Probability Distribution Functions of Example 6.

The Average of the Absolute Values of the Deviations = .0393944

The Maximum of the Absolute Values of the Deviations = .0732274. It is No. 7.

Number of Positive Deviates = 16.

Number of Negative Deviates = 0.

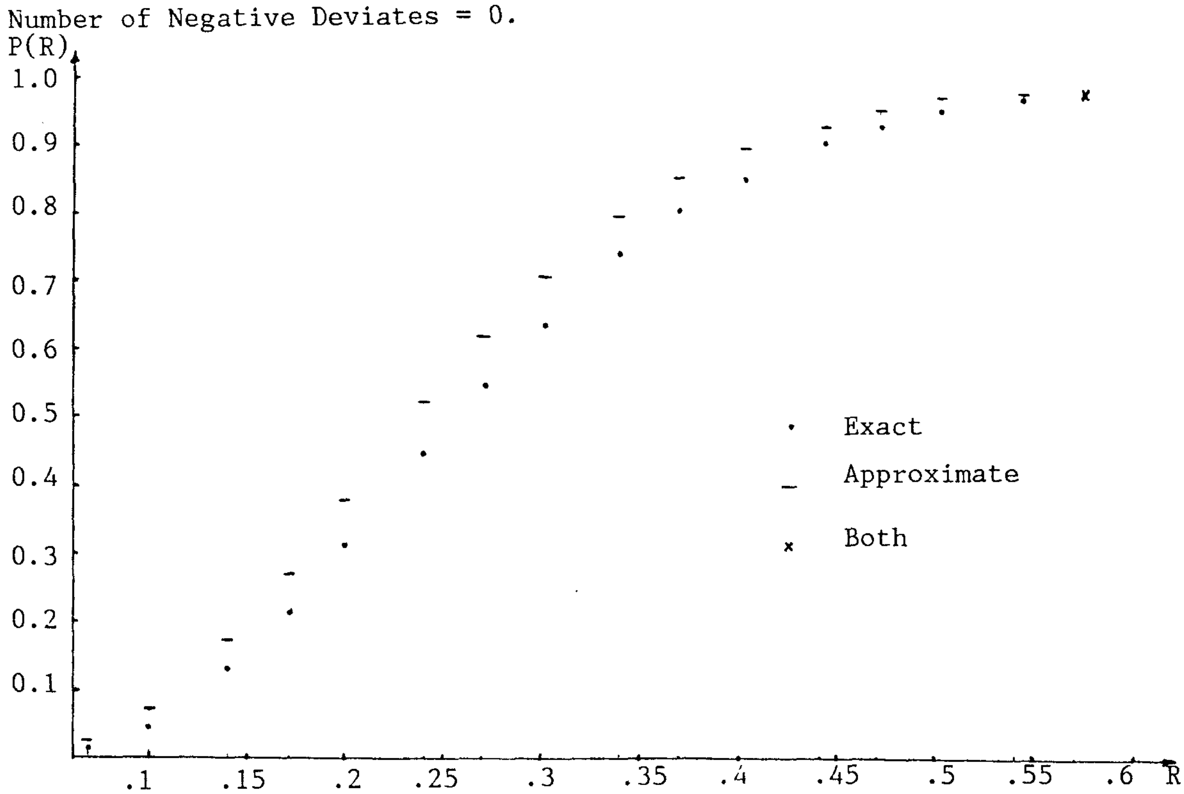


Figure 8.15

SUMMARY AND CONCLUSIONS

In this chapter different discretizing methods have been shown and the most efficient method has been used in discretizing distribution functions of simple networks. As it was clear from the previous section we had to compare the approximate pdf of the project completion time with those obtained from Normal Tables (using 140 cells), and Incomplete Beta-Function Tables (using 100 cells) for input distribution using the following five measures of performance:

1 - Mean value of the project completion time.

2 - Standard Deviation.

3 - Coefficient of Skewness.

4 - The maximum of the absolute values of the deviations (MDV).

5 - The average of the absolute values of the deviations (ADV).

The variations in the measures of performance generally depend on the structure and the size of the Activity Network, the distributions of the activity times, the accuracy of the discretization, and the values of NRR and NIN.

In the following we discuss the impact of the distribution type on the measures of performance for the six examples of the chapter.

Table 8.29 shows how the distribution type affects the measures of performance. In each of the six problems considered in Table 8.29, the approximate mean value of the project completion time is within 3% of the exact mean, and the approximate standard deviation of the project completion time is within 29% of the exact standard deviation. The approximate mean and standard deviation both are less than exact mean and standard deviation except for example 1, while the approximate coefficient of skewness is higher than exact coefficient of skewness. The approximate coefficient of skewness is within 42% of the exact coefficient of skewness except for example 6 which is very high; i.e. approximately 164% higher than the exact value.

The graph of the density functions of each problem has the form shown in Figure 8.15. The approximate graph converges to the true graph as the number of cells increases.

The values of MDV and ADV vary with the type of operation. In the six problems of Table 8.29, MDV and ADV for problems 2,4, and 6 are greater than MDV and ADV for problems 1,3, and 5 respectively. MDV is always less than .074 and ADV is less than .04. The smallest values of MDV and ADV were obtained in example 1 where the activities are series and have normal distribution. The accuracy of the approximation can be enhanced by using more cells. The graphs of the distributions of the problems in Table 8.29 have the general form of Figure 8.15 and 8.16.

Table 8.29

Sensitivity of the approximating method to the PDF's

Example	Distribution Type	Operation Type	Comparison of the Approximate PDF with the True PDF							
			Mean		Standard Deviation		Coefficient of Skewness		MDV	ADV
			APRX.	EXACT	APRX.	EXACT	APRX.	EXACT		
1	Normal	Convolution	200.00695	200.00000	14.228808	14.187952	0	0	.0245581	.0023969
2	Normal	Greatest	105.67480	106.39181	8.324662	6.4543783	0	0	.0636856	.0294306
3	Beta(Moderate Skewed)	Convolution	.73855940	.75000000	.2200811	.22821740	.2346219	.2190889	.0345425	.0171820
4	Beta(Moderate Skewed)	Greatest	.45821860	.47177890	.1386535	.14202920	.1914087	.1352181	.0730681	.0372413
5	Beta(Extreme Skewed)	Convolution	.39981420	.40000000	.1594775	.17056020	.6053254	.5863019	.0525366	.0275114
6	Beta(Extereme Skewed)	Greatest	.26650170	.27186900	.1054850	.11941480	.8391411	.3175945	.0732274	.0393944

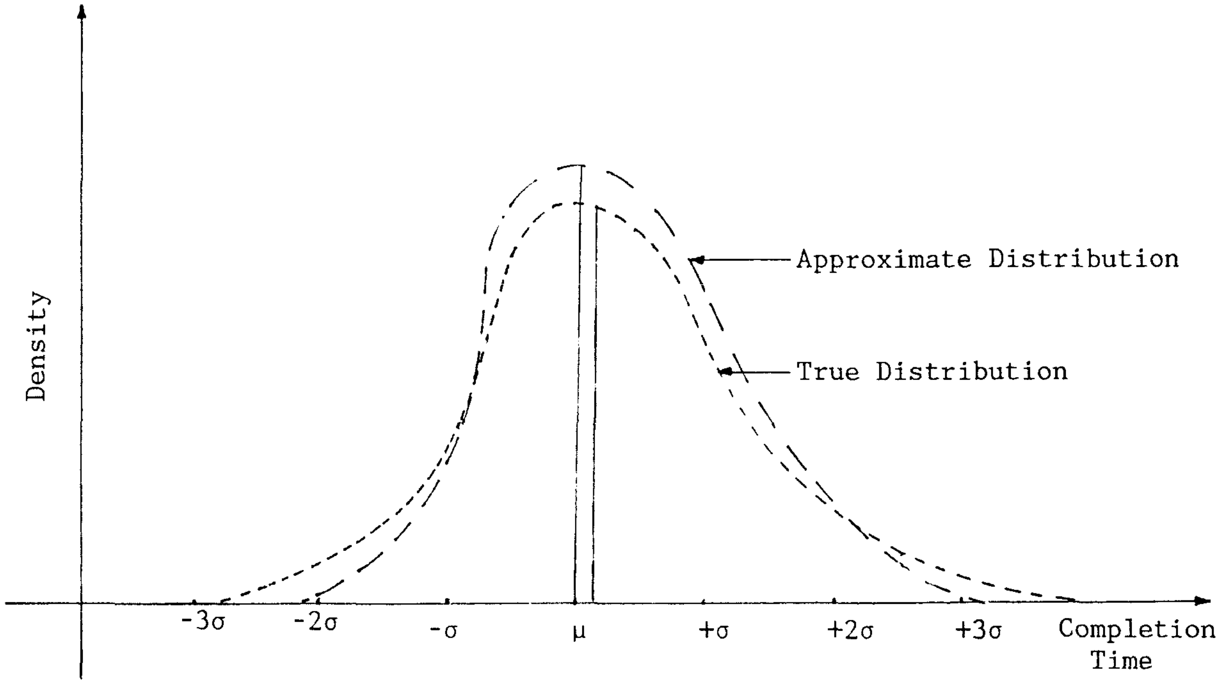


Figure 8.16

General Forms of the Probability Density Functions of the Project Completion time.

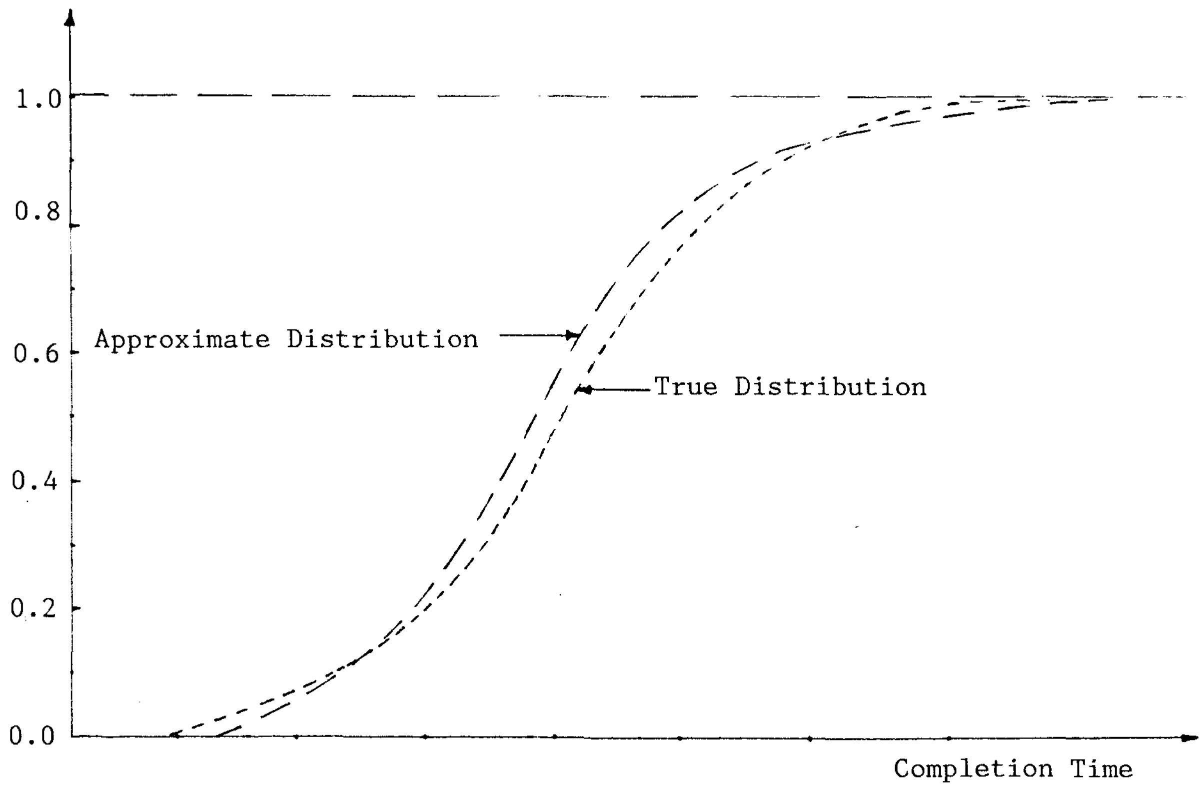


Figure 8.17

General Form of the True and the Approximate Distributions.

CHAPTER 9: CONVOLUTING DISCRETE APPROXIMATION TO CONTINUOUS DISTRIBUTION WITHIN THE CIM FRAMEWORK

INTRODUCTION

As mentioned in Chapter 3 an alternative representation of base variable distribution is a histogram. With this representation the analytic convolution formulae become sums rather than integrals, and the calculations are both simple in concept and straightforward computationally. The Controlled Interval and Memory (CIM) approach is based on histograms with intervals of equal width. Like discrete probability distribution approach, approaches based on interval and histogram representation involve inherent bias. However, in the CIM framework the bias can be considered and controlled, drawing upon functional and numerical techniques, greatly reducing the computation required to maintain confidence in the results.

This chapter in the beginning looks at example problem of Chapter 3. This example is used throughout to illustrate and explain the major points of CIM approach as presented in (Cooper and Chapman 1987). The ways of specifying distributions so as to control distribution specification error are then considered. The next section briefly compares the generalised CI approach with alternative approaches, in the context of independent addition, in terms of computation efficiency and precision, errors in the specification of distributions and loss of information.

The final section compares the accuracy of the DPD and the CIM approaches through simple examples.

Example problem

Reconsider example problem of Chapter 3. Recall that the project consists of two stages in series. 1(design) and 2(construction).

A simple CI definition of the uncertainty associated with design stage uses three durations D1 in months,with associated probabilities, P(D1), as indicated in Table 3.3, or alternatively as rectangular histogram of Figure 3.2.

Table 3.3 Design distribution, D1 months.

D1	P(D1)
2	.3
3	.5
4	.2

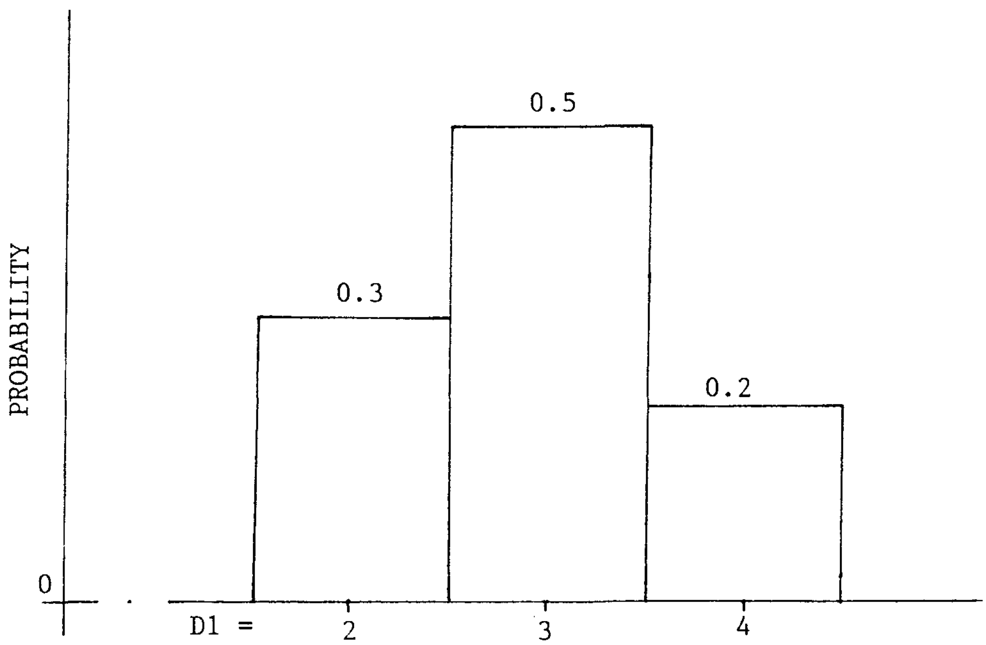


Figure 3.2

The next stage is 'construction'. A simple CI definition of the uncertainty associated with this stage uses three durations D2 for this stage in months, with associated probabilities, P(D2), as indicated in Table 3.4, or as rectangular histogram of Figure 3.5.

Table 3.4 Construction distribution, D2 months.

D2	P(D2)
6	.3
7	.6
8	.1

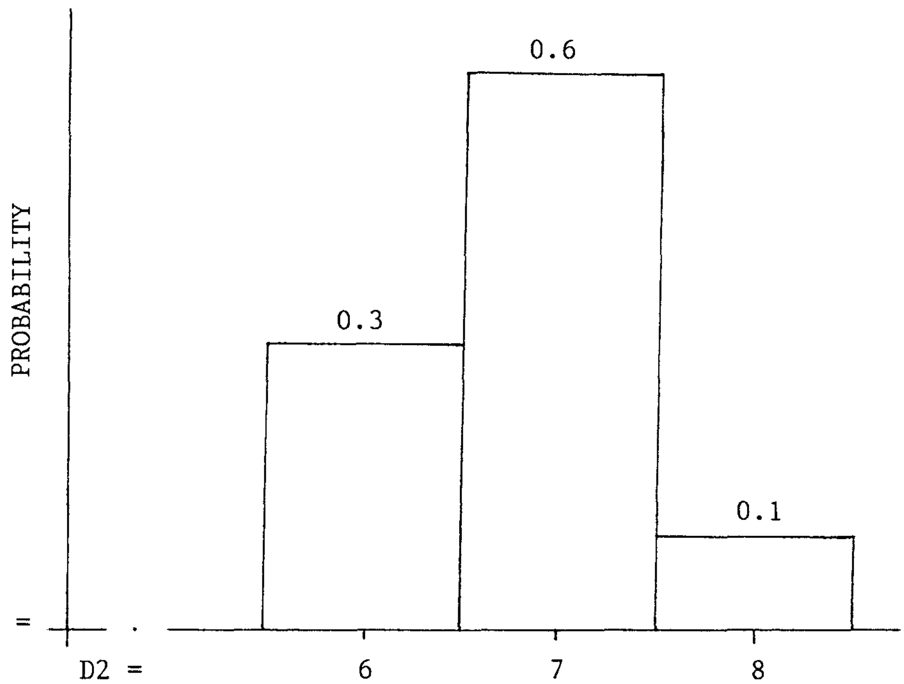


Figure 3.5

Design and construction are assumed to have independent duration.

The approach for combining distributions adopted by BP, in the old software, uses the common interval method as used in this example. The addition of two distributions in histogram form with equal or common cell division using old software produces a result distribution which has equal cell interval as shown in Table 3.6 and Figure 9.1.

Table 3.6 The simplified computation for 'design' plus 'construction', in the case where the individual 'design' and 'construction' intervals do not need to be remembered.

DESIGN PLUS CONSTRUCTION				
Da	Computation			
8	0.3x0.3			0.09
9	0.3x0.6	+ 0.5x3		0.33
10	0.3x0.1	+ 0.5x0.6	+ 0.2x0.3	0.39
11		0.5x0.1	+ 0.2x0.6	0.17
12			0.2x0.1	0.02

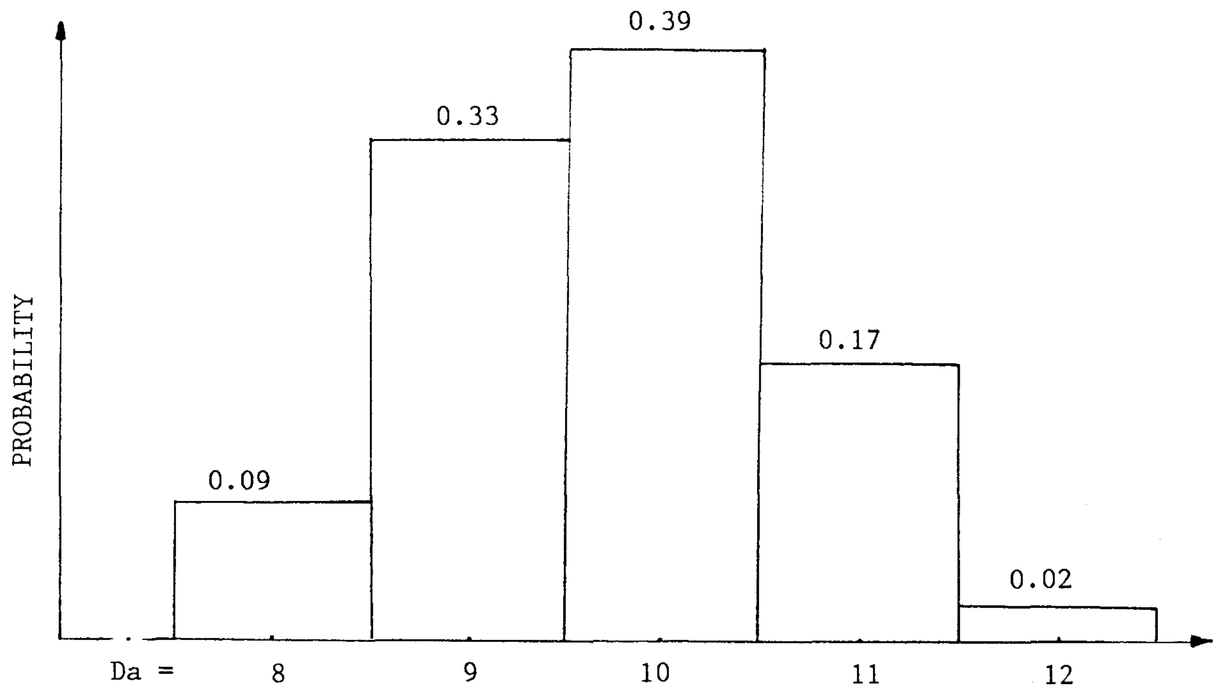


Figure 9.1

In other words using the old software, when rectangular distribution or cell of histogram is combined repeatedly, the result distribution is allocated as a rectangular. However, functional integration or finite difference techniques have proved that when a rectangular distribution or cell of histogram is combined with another rectangular distribution or cell of histogram, then the result distribution should be of a triangular form.* Hence to assume that the resulting distribution is rectangular when it should be triangular in shape is inappropriate. Although a lot of the errors of the above type cancel out, the net result still bias the distribution variance on the low side because the actual result distribution should be wider than it actually is.

Complete elimination or partial reduction of this computation error within the CI framework can be achieved via one or more of five approaches:

- derived correction factors;
- interpolated correction factors;
- more classes;
- empirically determined correction factors; or
- more allocation.

Derived Correction Factors

This approach involves assuming specific within-class probability distributions for any two component items or item groups, and deriving correction factors based upon the associated probability distribution function for the within-class distributions associated with the result. For example, if the distributions of Tables 3.3 and 3.4 are associated with rectangular density form as shown in Figures 3.2 and 3.5, the joint distributions has the trapezoidal* density form illustrated in Figure 9.2.

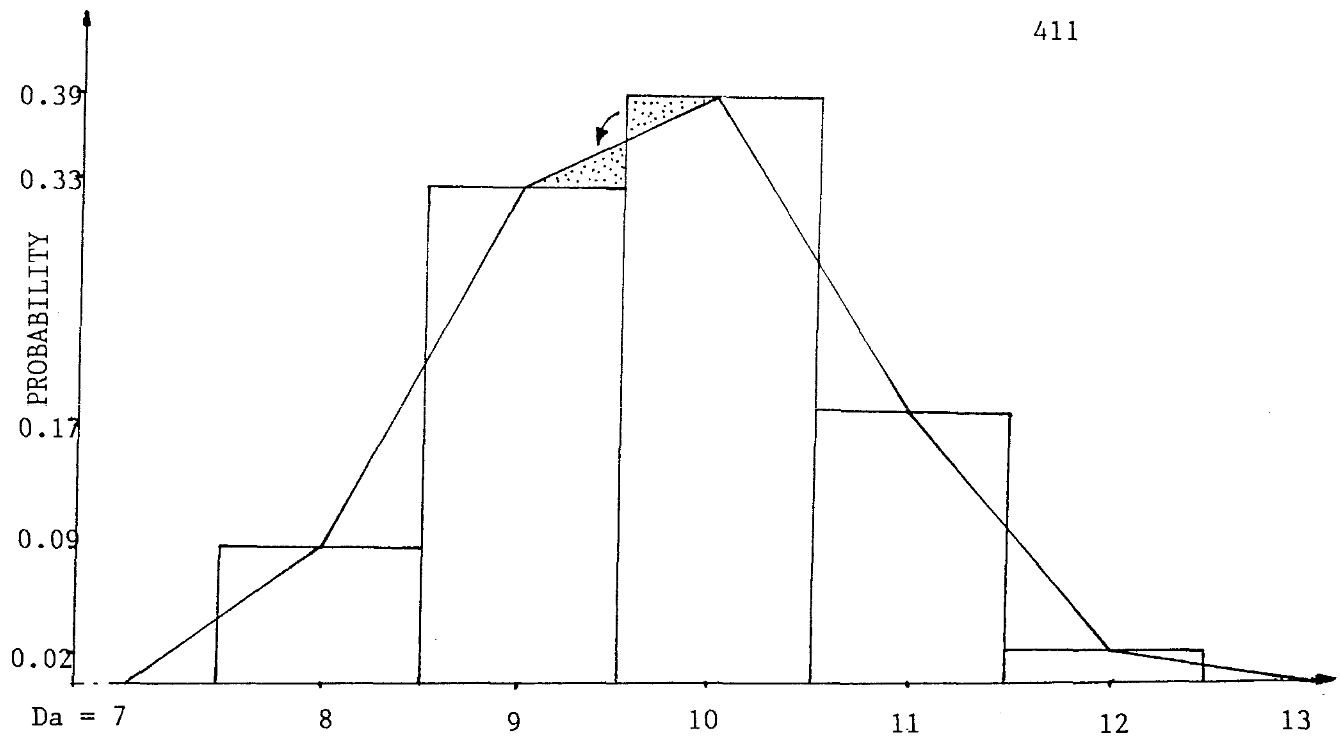


Figure 9.2

---

* If T_k represents the sum of k independently and uniformly distributed random variables, then

$$f_{T_1}(x) = \begin{cases} 1 & 0 \leqslant x \leqslant 1 \\ 0 & \text{other } x, \end{cases} \tag{9.1}$$

and

$$f_{T_2}(x) = \begin{cases} x & 0 \leqslant x \leqslant 1 \\ 2-x & 1 \leqslant x \leqslant 2 \\ 0 & \text{other } x. \end{cases}$$

The two densities are plotted in Figure (i). The rectitude of the latter density may be established via moment-generating functions, noting that

$$m_{T_1}(t) = 0 + \int_0^1 e^{tx}.1.dx = (e^t - 1)/t$$

is the moment-generating function for the uniformaly distributed random variable T_1. Since T_2 is the sum of two such independently distributed random variables, its moment-generating function must be

$$m_{T_2}(t) = E\,[e^{tT_2}] = E[e^{t(U_1+U_2)}] = E[e^{tU1}]E[e^{tU_2}],$$

or

$$m_{T_2}(t) = [m_{T_1}(t)]^2 = (e^{2t} - 2e^t + 1)/t^2.$$

A comparison of tabulations of Laplace transforms will reveal that $m_{T_2}(t)$ is the moment-generating function corresponding to the triangular function

$f_{T_2}(x)$ of Equation (9.1). Owing to the unicity property of Laplace transforms, $f_{T_2}(x)$ must be then the probability density function for T_2, the sum of two independently and uniformly distributed random variables.

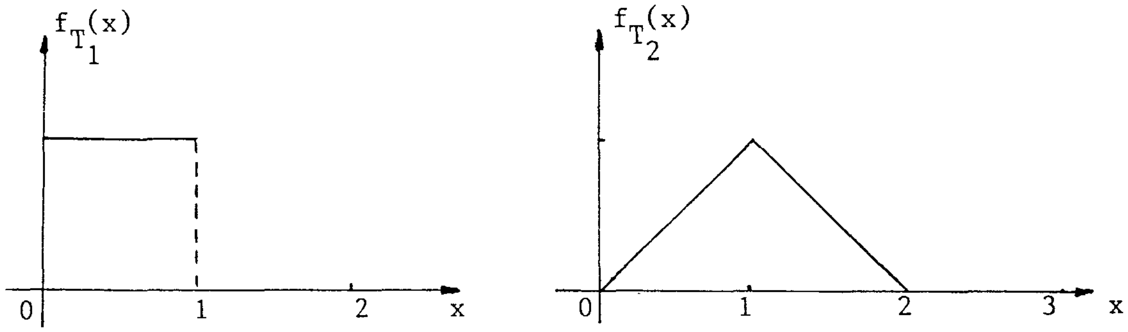


Figure (i)

Associating the joint distribution of Table 3.6 with a rectangular density form introduces error, the necessary net correction involving a transfer from higher probability classes to adjacent lower probability classes equal to one eighth of the differences between adjacent probability values, also illustrated in Figure 9.2. The necessary correction is equivalent to cutting off the corners of the rectangular density form and relocating them as indicated.

In this case the key errors are a probability of 0.01125 which should be allocated to the classmark 7, and a probability of 0.0025 which should be allocated to the classmark 13. More generally, 0.01 is the order of magnitude of the errors associated with extreme classmarks, other errors being comparatively trivial, if only three classes are used for component distributions as for this example. Adding succesive distributions will lead to smoother and smoother joint distribution curves, involving smaller and smaller corrections.

One difficulty with this approach is the more and more complex form of the corrections required as more and more items are considered, or if smoother curves are used to begin with. Continual application of corrections based on rectangular histograms is inconsistent. It overstates risk, because it over-correct relative to the smoother distributions which must result from successive distribution combinations. However, as most other sources of bias under-state risk, a compensating bias can be attractive, if it can be justified.

Interpolated Correction Factors

This approach involves using the correction procedure derived above as an upper bound, in relation to a lower bound provided by the uncorrected procedure of Table 3.6. This yields a modified correction procedure which might be interpreted as an unbiased interpolation between these bounds. The difficulty with this approach is defining and justifying the interpolation point. However, it provides more flexibility than the first on its own.

More Classes

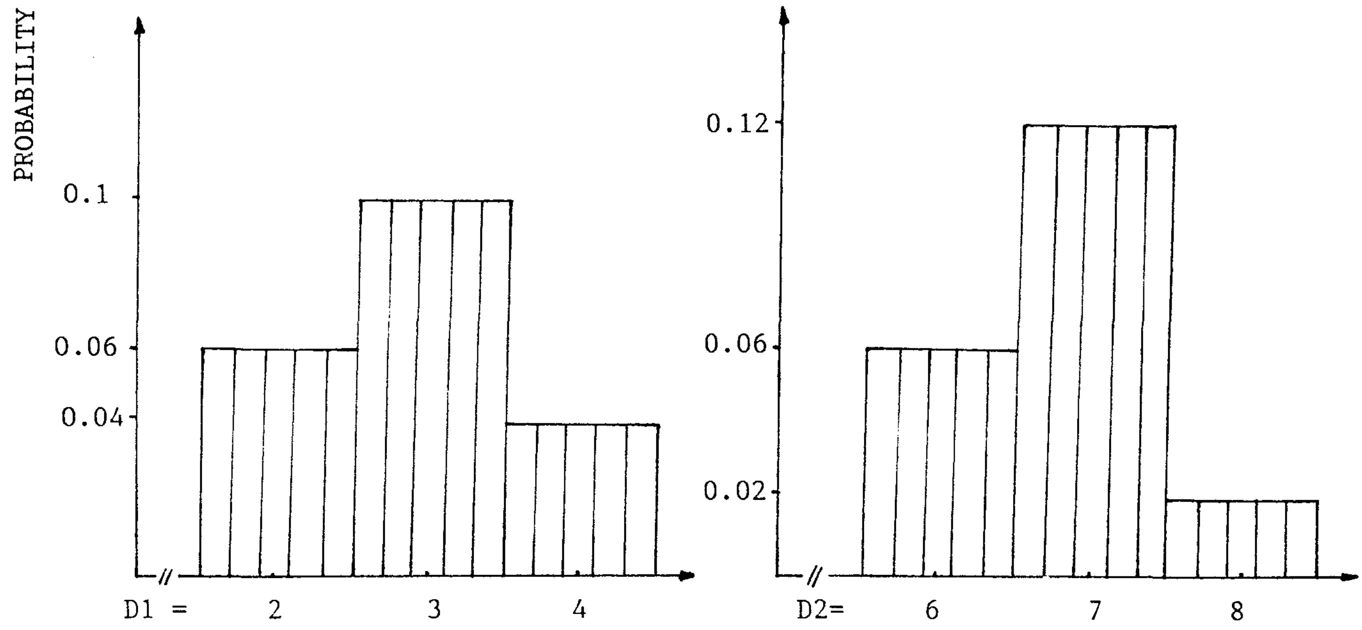
This approach involves recognizing that computation error decreases in proportion to the difference in adjacent probability values, as illustrated by Figure 9.2. This means computation error decreases as the number of classes increases. In principle it does not disappear entirely until each class contains a single integer, a special limiting form of within-class distribution which is error free. In practice it decreases to zero very rapidly, for whatever degree of precision is required. Direct specification of a finer class width structure is possible, but not necessary. Any convenient class for specification purposes can be used to define a narrower width version of Tables 3.3 and 3.4 for computation purposes, using a suitable interpretation of the distribution shape. For example, the classes associated with D1 and D2 in Tables 3.3 and 3.4 might be separated into 5 sub-classes, defined assuming uniform within-class distribution, as indicated by Table 9.1 and Figure 9.3.

Table 9.1 A finer interval structure for D1 and D2, assuming uniform within-class distributions in the origional specifications in Tables 3.3 and 3.4

DESIGN		CONSTRUCTION	
D1	P(D1)	D2	P(D2)
1.6	0.06	5.6	0.06
1.8	0.06	5.8	0.06
2.0	0.06	6.0	0.06
2.2	0.06	6.2	0.06
2.4	0.06	6.4	0.06
2.6	0.1	6.6	0.12
.			
.			
.			
etc.		etc.	

Table 9.2 The simplified probability-tree computation for Da = $D_1 + D_2$ using the finer interval structure of Table 9.1.

DESIGN PLUS CONSTRUCTION		
Da	Computation	P(Da)
7.2	0.06x0.06	0.0036
7.4	0.06x0.06 + 0.06x0.06	0.0072
7.6	0.06x0.06 + 0.06x0.06 + 0.06x0.06	0.0108
7.8	0.06x0.06 + 0.06x0.06 + 0.06x0.06 etc.	0.0144
8.0	0.06x0.06 + 0.06x0.06 + 0.06x0.06 ...	0.0180
8.2	0.06x0.12 + 0.06x0.06 + 0.06x0.06 ...	0.0276
8.4	0.06x0.12 + 0.06x0.12 + 0.06x0.06 ...	0.0372
8.6	0.06x0.12 + 0.06x0.12 + 0.06x0.12 ...	0.0468
.		.
.		.
etc.		


PROBABILITY
0.1
0.06
0.04
D1 =
2
3
4
0.12
0.06
0.02
D2=
6
7
8

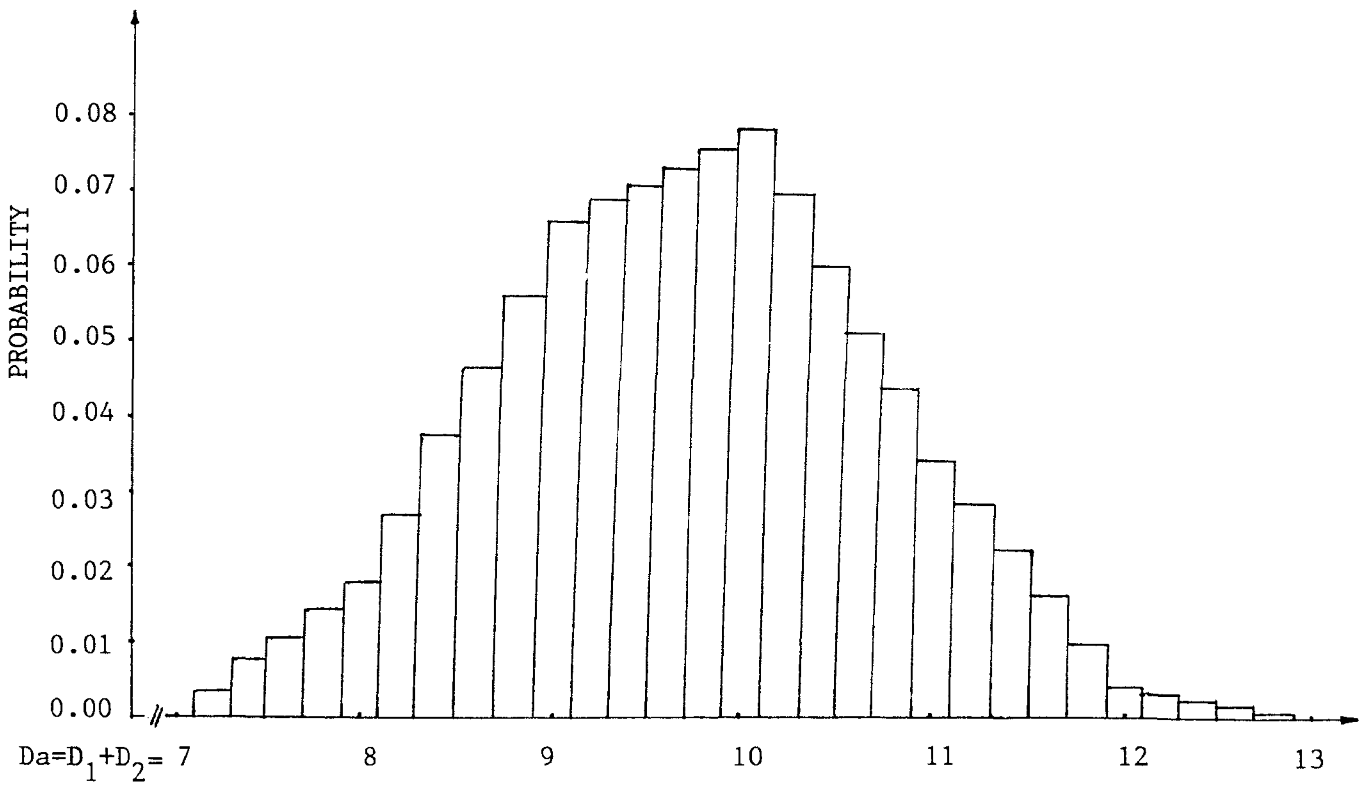

PROBABILITY
0.08
0.07
0.06
0.05
0.04
0.03
0.02
0.01
0.00
$Da=D_1+D_2=$ 7
8
9
10
11
12
13

Figure 9.3

The P(D1) = 0.06 associated with D1=1.6 in Table 9.1 was obtained via 0.3/5, the P(D1)=0.3 for D1=2 of Table 3.3 providing the 0.3, and so on for the 3x5=15 classes of each distribution. If the distributions associated with Table 9.1 were combined usign the procedure of Table 3.5, the result would be as indicated by Table 9.2 and Figure 9.3.

This clearly provides a result much closer to the trapezoidal cumulative form illustrated by Figure 9.2 than Table 3.6. Indeed, the 0.0036 + 0.0072 = 0.0108 probability allocated to 7.2 and 7.4 is very close to the 0.01125 which should have been allocated to Da = 7 in Table 3.6, as noted earlier. More generally, 0.001 is the order of magnitude of the errors associated with extreme classmarks, other errors being comparatively trivial. All such errors decrease as a function of n^2, where n is the number of classes. The availability of this approach makes the procedure of Table 3.6 and extensions like that illustrated by Table 9.2 inherently error free. Computation error is simply an option, which can be accepted if it is not worth the effort to reduce or eliminate it, at any appropriate level.

Empirically Determined Correction Factors

This approach involves empirical experiments to assess correction factors akin to those associated with the first and second approaches. Any pair of distributions can be combined using a very fine class structure, so fine it can be treated as error free to the level or precision required. The result can then be summarized in a variety of simple frameworks, which can be treated as error-free results. Simplified versions of the component distributions can then be combined, and compared to the error-free equivalent. Error measured in this way can be related to the class width, the number of classes, the difference in adjacent probability values, or other parameters suggested by the first or second approach. This approach demands extensive experimentation, but it lends further flexibility to the first three.

More Allocations

This approach to computation error involves allocating each joint probability component in a manner which reduces error to an acceptable level. For example, instead of redefining Tables 3.3 and 3.4 in the form of Table 9.1 with 5 times as many classes, and then combining the results as illustrated by Table 9.2, the same result could be obtained more directly. The factor set

0.04 0.08 0.12 0.16 0.20 0.16 0.12 0.08 0.04

could be applied to the 0.3x0.3=0.09 , 0.3x0.6=0.18, 0.5x0.3=0.15 and subsequent entries of Table 3.6 to generate the entries of Table 9.3.

Table 9.3 The computation of Da = D1 + D2 using the factor set 0.04, 0.08, 0.12, 0.16, 0.20, 0.16, 0.12, 0.08, 0.04

DESIGN PLUS CONSTRUCTION		
Da	Computation	P(Da)
7.2	0.04x0.09	0.0036
7.4	0.08x0.09	0.0072
7.6	0.12x0.09	0.0108
7.8	0.16x0.09	0.0144
8.0	0.20x0.09	0.0108
8.2	0.16x0.09 + 0.04x0.18 + 0.04x0.15	0.0276
8.4	0.12x0.09 + 0.08x0.18 + 0.08x0.15	0.0372
8.6	0.08x0.09 + 0.12x0.18 + 0.12x0.15	0.0468
8.8	0.04x0.09 + 0.16x0.18 + 0.16x0.15	0.0564
.		.
.		.
etc.		etc.

The factor set thus serves to integrate each pair of component distribution class intervals, allocating the associated probability to a joint probability distribution with a different class structure. The chosen factor set sums to one and spans twice the class width of Table 3.6, using a triangular pattern conforming to the result class width as for 9.2. For independent component distributions based on equal intervals, a triangular factor set provides a simple and attractive allocation pattern, but any of the first three approaches to reducing computation error can be applied to the definition of alternative factor set shapes. The triangular shape is based upon integrating two within class distributions which are both uniform (constant probability) and both the same width. Its validity given these assumptions is illustrated by Table 9.2.

An obvious advantage of using more allocatons in comparison to using more classes is the considerable gain in computational efficiency. Apart from avoiding the need to refine the component distribution scales as the number of intervals is increased for the result, computation effort increases in a linear manner, whereas the approach of Tables 9.1 and 9.2 involves an increase in computation effort which is proportional to the square of the increase in the number of intervals.

A further advantage of the more allocations approach is that it generalises easily to component item and result classes which are not the same size, an obvious advantage when the items are of different magnitude, allowing a controlled number of classes for all component and result distributions. As with Table 9.3, each class interval combination involves a form of integration of the associated within-class distributions. The minimum value of the resulting distribution is the sum of the component class distribution minima. The maximum value of the resulting distribution is the sum of the component class distribution maxima. The shape of the allocation defined by the functional equivalent of the factors can reflect specific within-class distribution assumption, adjustments based on bounds, or adjustments based on empirical experiments, including adjustments designed to balance the effects of truncating unwanted distribution tails. Each component distribution and the resulting sum distribution can use different interval widths.

Another advantage of the more allocations approach is the ability to drop common intervals within each distribution, if precision within one particular region is more important than precision in other regions, as may occur when very long distribution tails are of interest.

A final advantage of the more allocations approach is that it generalises easily to other distribution combination operations.

Generalised CI Approaches

Detailed treatment of generalised CI approaches, embedding the first four approaches as necessary in the fifth, is not considered in (Cooper and Chapman 1987). However, the examples chosen should clarify the principles involved, and illustrate the order of magnitude of the computation effort and error. The initial simple example of Table 3.6 provides errors of the order of 0.01, which the example of Table 9.2 reduces to the order of 0.001. Computation error of the order of 0.01 is acceptable for simple examples. Computation error of the order of 0.001 may not be acceptable for analysis purposes, but it is easily reduced by further orders of magnitude, at negligible cost in modern computing terms. Computation cost and error balancing can be an explicit part of CI computer software, either automatically or through manual intervention.

Generalising the addition of two indpendent probabilistic variables to three or more poses no new problems.

SPECIFICATION ERROR AND SPECIFICATION GENERALIZATION

Computation error of the order of 0.01 for the joint distribution of Table 3.6 should be considered in the context of errors in specifying the probability distribution which may be of the order 0.1, ten times as large. Controlling specification error is much more important than controlling computation error. Cooper and Chapman (1987) distinguished two kinds of specification error: approximation error and residual error. There are a number of ways of reducing or eliminating specification approximation error:

- direct use of more classes;
- maximum order polynomials for input distributions;
- less than maximum order polynomials for output distributions;
- standard probability level specifications;
- other input distribution functions.

Direct Use of More Classes

The use of one month class intervals for the distributions of Tables 3.3 and 3.4 involves a specification approximation error which the smaller class interval of Table 9.1 could reduce by using a smoother curve specification like that of Table 9.4 and Figure 9.4. The probability allocations of Table 9.4 are quite different from those of Table 9.1, and differences of this kind could be argued to be a specification approximation error associated with Tables 3.3 and 3.4, corrected by Table 9.4.

Sometimes it is convenient to reduce specification approximation error to an acceptable level by the direct use of more classes, as illustrated by Table 9.4. Flexibility and simplicity are maximised. Any distribution shape can be used, to any degree of specification precision. The existence of this approach makes the CI approach inherently free from specification approximation error. Approximation is simply an option, to be accepted if the saving in specification effort is worthwhile.

Maximum Order Polynomials for Input Distributions

This approach involves fitting a smooth curve to a specification which is cruder than that used for computation purposes, and interpereting the curve choice as part of the specification. For example, the 3 classes of P(D1) in Table 3.3 could be associated with a cumulative distribution shape defined by the 3rd order polynomial

$$b_0 + b_1X + b_2X^2 + b_3X^3 = p,$$

where for X = 1.5, 2.5, 3.5 and 4.5, p = 0, 0.3, 0.8 and 1.0 (Figure 9.5). Solving the associated 4 linear equations for the b_i and using them to allocate probabilities to classmarks 1.6, 1.8, 2.0 ,..., 4.4 will yield a shape more like Table 9.4 than Table 9.1. Using still finer classes would involve no additional specification effort, and computation effort could be balanced against both specification approximation error and computation error. More generally, nth order polynomials with n classes, given an n greater than 3, provide a useful complement to direct specification of more intervals.

Table 9.4 A finer interval structure for D1 and D2 assuming a smooth-curve specification.

DESIGN		CONSTRUCTION	
D1	P(D1)	D2	P(D2)
1.2	0.01	5.2	0.01
1.4	0.01	5.4	0.01
1.6	0.01	5.6	0.01
1.8	0.04	5.8	0.02
2.0	0.06	6.0	0.05
2.2	0.08	6.2	0.08
2.4	0.09	6.4	0.11
2.6	.10	6.6	0.11
2.8	.10	6.8	0.12
etc.		etc.	

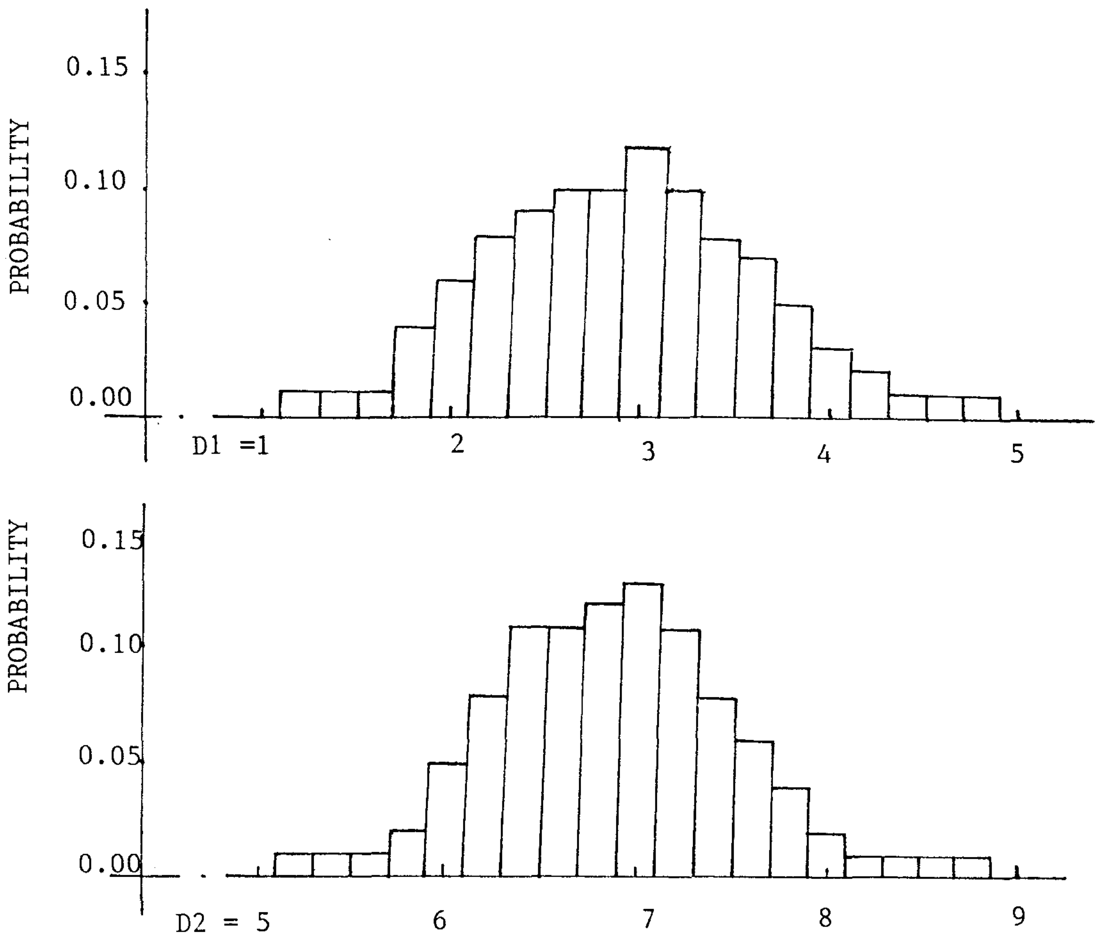


Figure 9.4

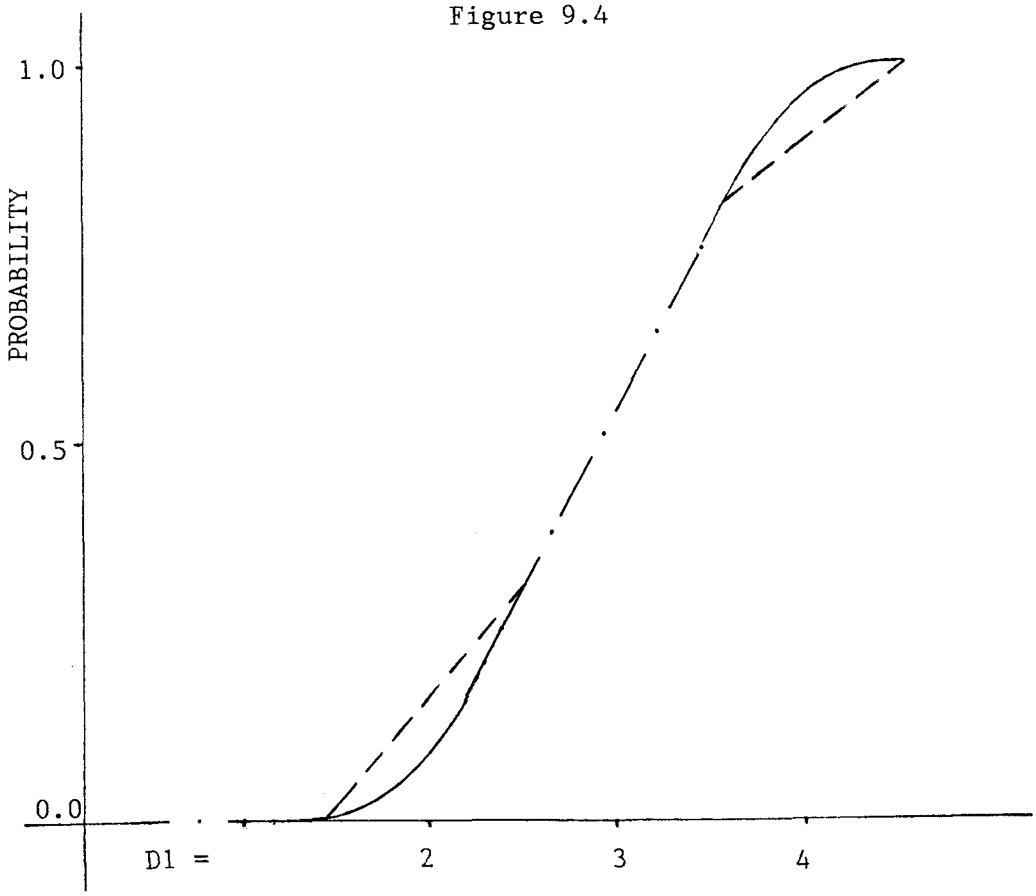


Figure 9.5

Less than Maximum Order Polinomials for output Distributions

In addition to, or instead of, smoothing input or component distributions, result or output distributions could be smoothed using regression analysis and polynomials of lower order than the number of classes would permit.

Standard Probability Level Specifications

An alternative involves using standard probability levels, like quartiles or deciles, and fitting a polynomial in a manner akin to that just discussed.

Other Input Distribution Functions

A final alternative involves using distribution functions other than polynomials, specified in terms of suitable parameters. For example, Normal (Gaussian) distributions can be specified in terms of mean and variance, and Beta distributions can be specified in terms of optimistic (minimum), pessimistic (maximum) and most likely (modal) values. Hybrids are also useful. For example, Normal distribution tails can be used in conjunction with a Beta distribution between confidence limits associated with optimistic and pessimistic values.

Specifications like this may be used for reasons of convenience, or they may be based on theoretical results.

Computer software which facilitates user choices with all these options makes a CI approach very flexible, and associated specification approximation error is easy to control. Adequate treatment of this source of error will often provide more than sufficient freedom from computation error as an indirect spinoff.

Specification Residual Error

Specification residual error, all other forms of specification error, is the most important kind of error, because it is the most difficult to control. No approaches can fully resolve these difficulties, including the CI approach. However, three characteristics not shared by most other approaches make the CI approach comparatively attractive.

First, computer software which allows a flexible approach to specification approximation error as just discussed also allows the user to select a means of specifying any particular distribution which is the most comfortable for the circumstances. The specificaton method can be adjusted to the user and the problem, rather than forcing the user to conform to a rigid model.

Second, given such software, it is an easy matter to extend it to allow comparisons of different approaches to specifying a distribution as a means of checking consistency.

Third, if n distributions are summed, the computation procedure automatically provides n-2 intermediate sums, a useful basis for further consistency tests, including representations showing the relative contribution to uncertainty of each source distribution.

COMPARISON WITH ALTERNATIVE APPROACHES

The CI approach is closely related to the DPD approach. If component and joint distributions are associated with rectangular histograms, a convenient convention, class mark values are also conditional expectations, expected values within each interval. This means Tables 3.3 to 3.6 can be given a DPD interpretation. However, the DPD approach ignores computation error, making it impossible to control explicitly, and limiting implicit control to the use of more Di and P(Di) doublets, or different doublet patterns. Further, it lacks the specification flexibility of the CI approach, with implications for specification approximation error and residual error. Finally, it involves a loss of information if large numbers of doublets are summarised. Its use may be preferable in some cases, but it would seem such circumstances must be very limited.

The CI approach is also related to a functional intergration approach. If the errors indicated by Figure 9.2 are corrected, using the allocations approach to computation errors and the factor set

0.125 0.750 0.125,

assuming rectangular histogram components and a trapezoidal histogram joint distribution, the resulting distribution is the same as a functional integration approach would provide. However, the CI approach is clearly much simpler in its basic forms, as illustrated by Tables 3.3 to 3.6, and much more flexible in general. Functional integration is only preferable when simple special cases are involved, as in basic random arrival and departure queueing systems.

The CI approach is also related to a numerical integration approach. If the rectangular histogram assumption for component distributions is interpreted as an approximation to an underlying smooth curve, the results are the same. However, the CI approach is simpler in its basic forms, and much more flexible in general. Numerical integration has no apparent advantage, although in some respects it is the alternative to CI which is closest conceptually.

Given a large number of distributions to sum, with none dominating because of size or variability, summing distribution means and variances to define a joint distribution mean and variance is an attractive approach. However, when these conditions do not hold, moment based approaches become complex by comparison to Table 3.6 and generalisations of it, without any guarantee of a decrease in computation cost or error. Further, a serious loss of information associated with component and resulting distributions is involved. Using such approaches to replace Table 3.6 should make this clear.

Monte Carlo approaches offer computational simplicity and flexibility when complex non-sequential distribution combination structures are involved. However, specification flexibility is not greater in the present context, and computation (sampling) error is much greater for comparable levels of computation effort. To provide a single sample, many random number generation processes would involve a computational effort comparable to that employed by Table 3.6, and computational precision of the same order would involve an increase in computational effort of several orders of magnitude.

OPERATION FLEXIBILITY AND OPERATION GENERALISATION

The basic common interval independent addition operation of Table 3.6 will generalise easily to accommodate dependent distributions using conditional specifications, and simplified percentage dependence specifications (akin to coefficient correlation specifications) are also possible. Subtraction and 'greatest' operations involve equally simple generalisations. However, the basic common interval approach will not generalise directly to multiplication and division. These operations in independent and dependent forms require the generalised controlled interval (CI) approach with allocation factor sets as illustrated by Table 9.3, or functional equivalents, which reflect the operation in question. For example, multiplication of two within-class distributions must be related to maximum and minimum values defined by products of component class maxima and minima.

However, it should be clear that within the context of a generalised CI approach incorporating the multiple allocation concept illustrated by Table 9.3, many other operations, involving dependence if required, pose no new special difficulties in terms of distribution specification effort, computational effort or computational error. This is not true of the functional integration, numerical integration or moment-based approaches to combining distributions. Monte Carlo approaches have no new advantages relative to the CI approach, unless complex non-sequential distribution combination patterns are involved, as in some PERT networks.

The DPD approach has some new comparative advantages, but only if precision is not very important and a basic distribution specification approach is acceptable.

In the following section we compare the accuracy of the CIM approach and the DPD approach through simple examples.

Example 1.

Consider the network configuration of Figure 9.6. Duration times of activities A, B and C are similar as shown in Table 9.5, or alternatively as rectangular histogram of Figure 9.7.

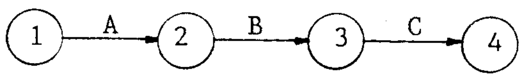


Figure 9.6

Table 9.5

X	p(x)
1	0.25
2	0.50
3	0.25

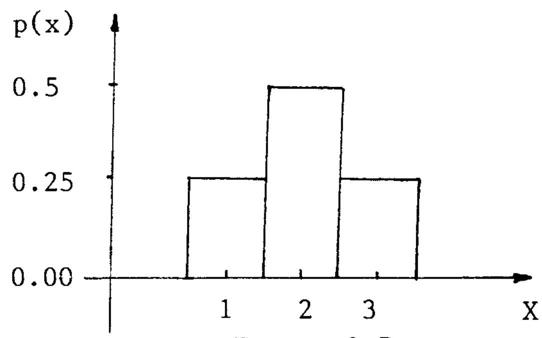


Figure 9.7

Considering duration times of activities A, B and C as discrete values, the pdf, mean, and variance of the realization times of nodes 2,3 and 4 can be computed as follows:

Tables 9.6 to 9.8 show the pdf, mean and variance of the realization times of nodes 2,3 and 4 respectively.

Table 9.6: Realization time of node 2.

X	p(x)
1	0.25
2	0.50
3	0.25
$E = 2$,	$\sigma^2 = 0.5$

Table 9.7: Realization time of node 3.

Y	p(y)
2	0.0625
3	0.25
4	0.375
5	0.25
6	0.0625
$E = 4$,	$\sigma^2 = 1.0$

Table 9.8: Realization time of node 4.

Z	p(z)
3	0.015625
4	0.09375
5	0.234375
6	0.3125
7	0.234375
8	0.09375
9	0.015625
E = 6 ,	σ^2 = 1.5

Considering duration times of activities as rectangular histogram, the exact mean and variance of the realization times of nodes 2,3 and 4 can be computed as follows:

Figures 9.8 and 9.9 show the density functions of the realization times of nodes 3 and 4 respectively.

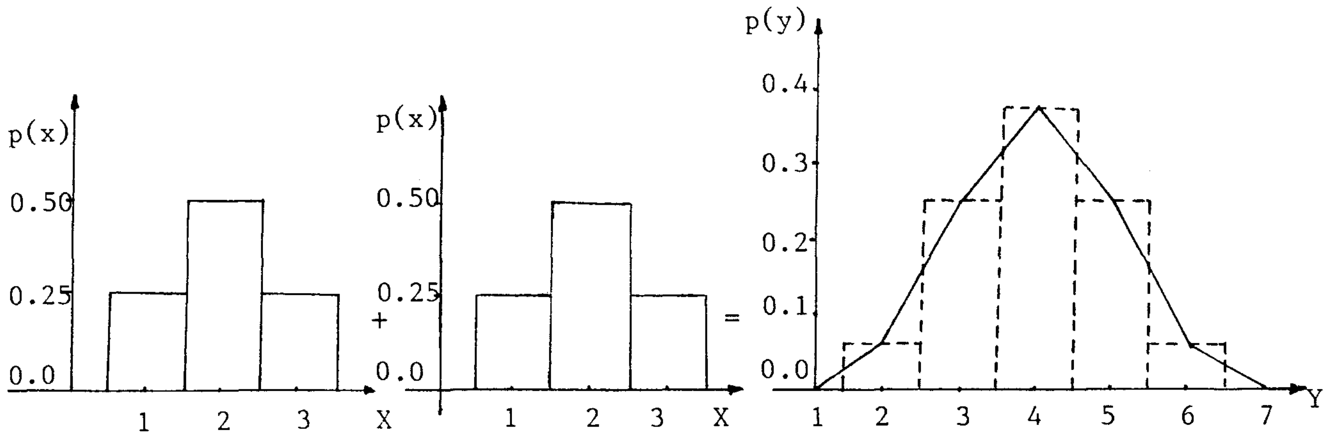


Figure 9.8: Density function of the realization time of node 3.

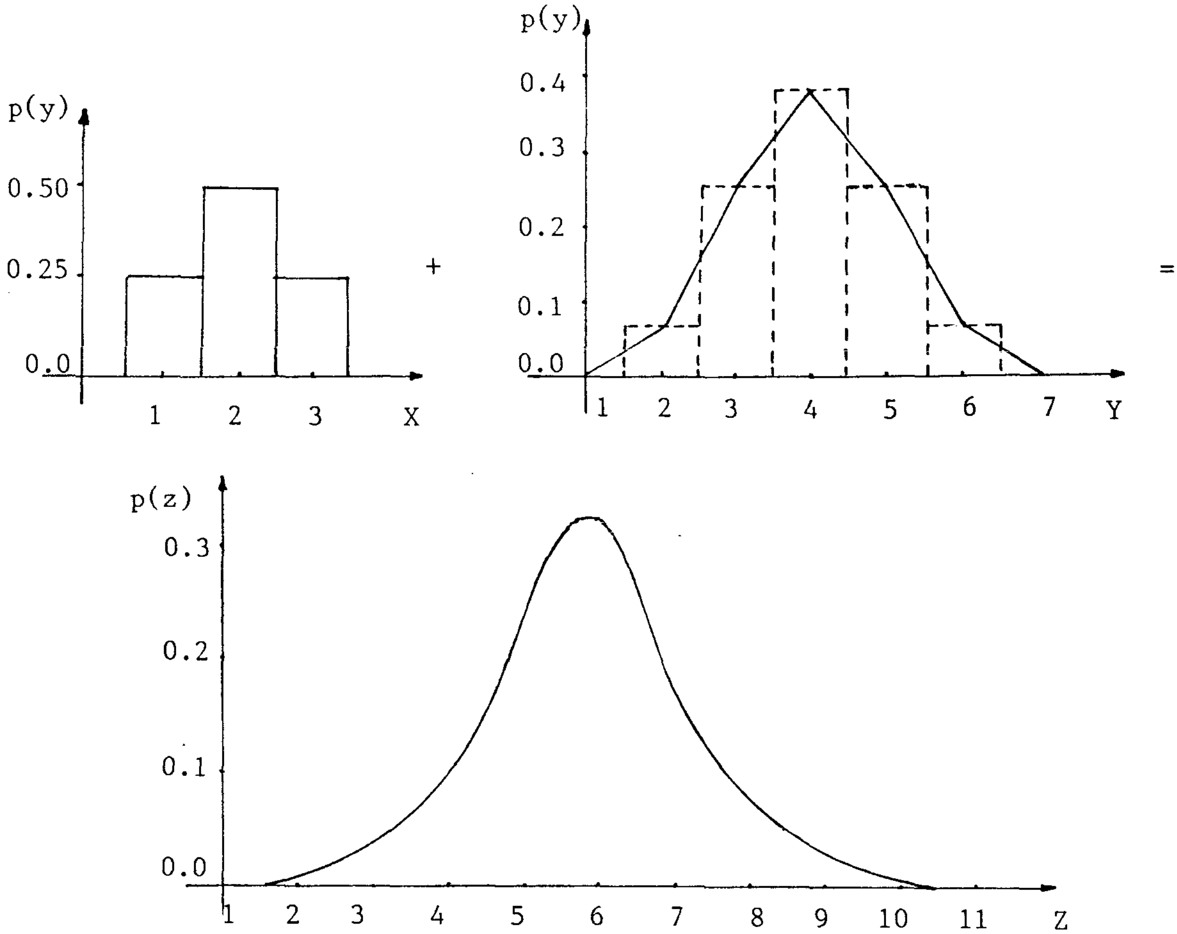


Figure 9.9: Density function of the realization time of node 4.

For node 2;

$$E(x) = \int_{.5}^{1.5} .25x\,dx + \int_{1.5}^{2.5} .5x\,dx + \int_{2.5}^{3.5} .25x\,dx = 2 \, ,$$

$$E(x^2) = \int_{.5}^{1.5} .25x^2\,dx + \int_{1.5}^{2.5} .5x^2\,dx + \int_{2.5}^{3.5} .25x^2\,dx = 4.583333 \, ,$$

$$\sigma^2_X = E(x^2) - (E(x))^2 = 4.583333 - 4 = .583333 \, .$$

For node 3;

$E(y) = 2xE(x) = 2x2 = 4$,

and

$\sigma^2_Y = 2x\sigma^2_X = 2x.583333 = 1.166666$,

where $Y = X+X$.

For node 4;

$E(z) = E(x) + E(y) = 2+4=6$,

and

$\sigma_Z^2 = \sigma_X^2 + \sigma_Y^2 = .583333 + 1.166666 = 1.75$,

where $Z = X + Y$.

Now, the mean and variance of the realization times of nodes 3 and 4 can be computed using CIM approach as follows:

Notice that in the new generation of risk analysis software used by BP, each time the input distribution is used, it is interpreted as a histogram as indicated by heavy line in Figures 9.10 and 9.11 rather than a trapezoid that it really is. Figures 9.10 and 9.11 show the density functions of the realization times of nodes 3 and 4 using CIM approach.

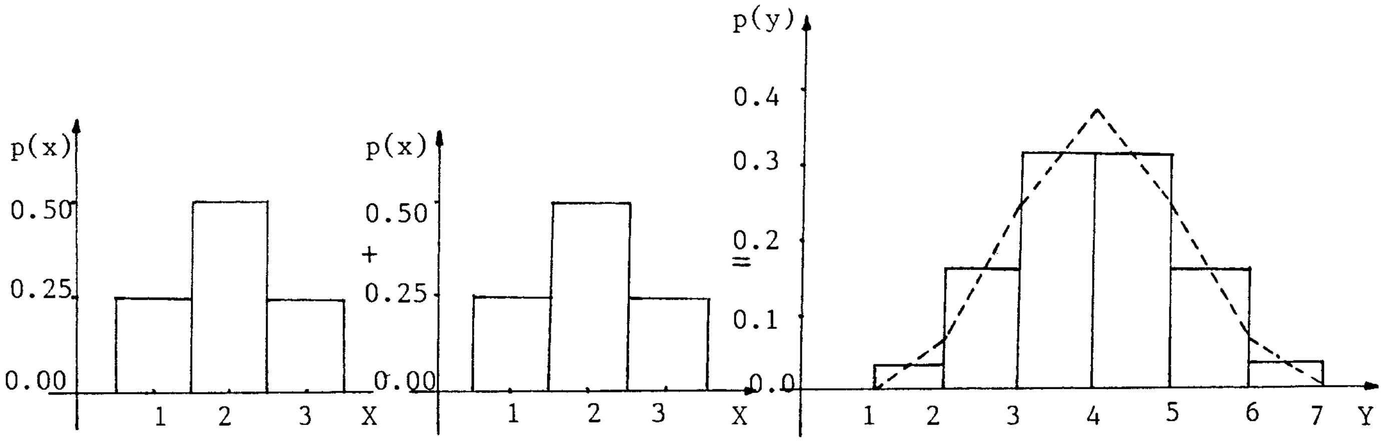


Figure 9.10: Density function of the realization time of node 3 using CIM approach.

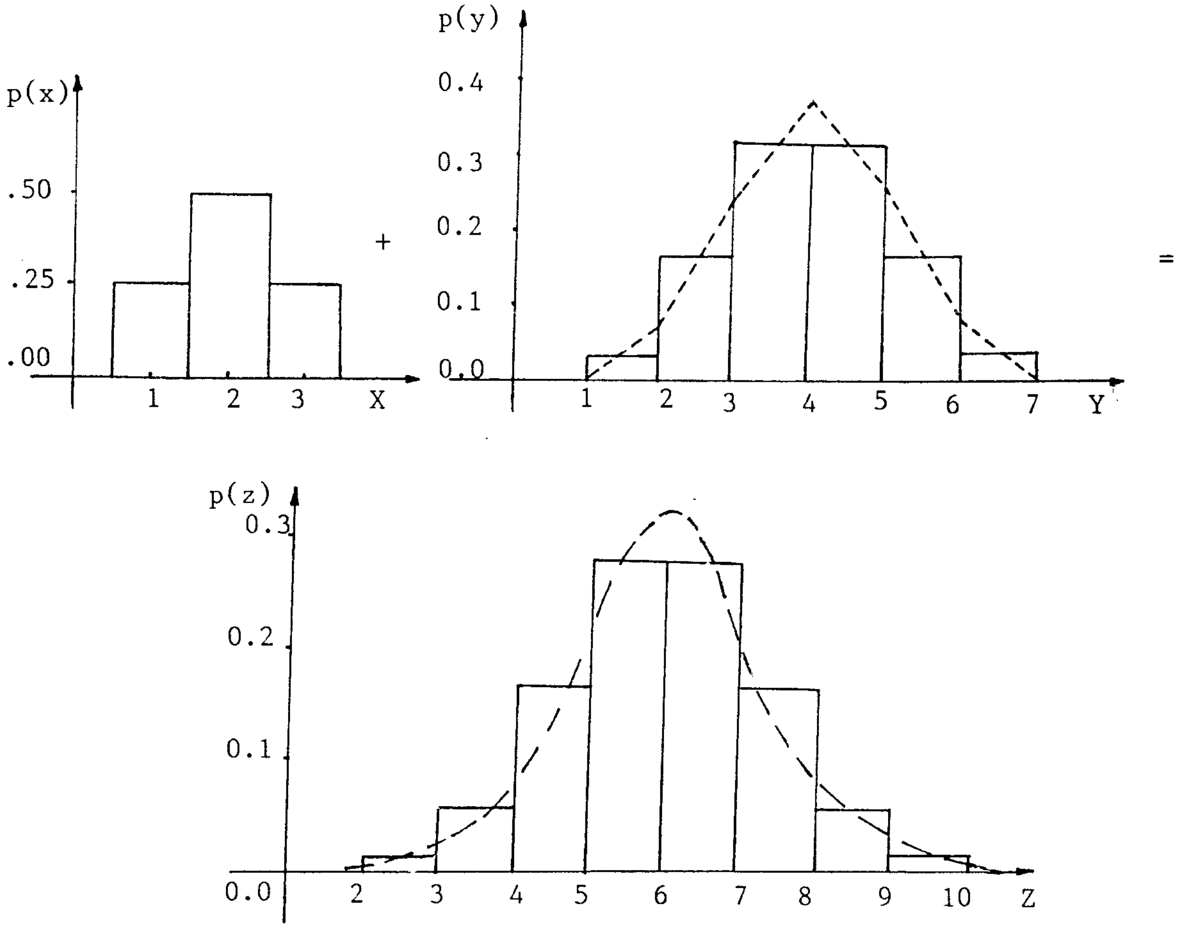


Figure 9.11: Density function of the realization time of node 4 using CIM approach.

For node 3;

$$E(y) = \int_1^2 .03125y\,dy + \int_2^3 .15625y\,dy + \int_3^4 .3125y\,dy + \int_4^5 .3125y\,dy$$

$$+ \int_5^6 .5625y\,dy + \int_6^7 .03125y\,dy = 4 ,$$

and

$$E(y^2) = \int_1^2 .0312y^2dy + \int_2^3 .15625y^2dy + \int_3^4 .3125y^2dy + \int_4^5 .3125y^2dy$$

$$+ \int_5^6 .15625y^2dy + \int_6^7 .03125y^2dy = 17.33333 ,$$

$$\sigma_Y^2 = E(y^2) - (E(y))^2 = 17.33333 - 16 = 1.33333 .$$

For node 4;

$$E(z) = \int_2^3 .0078215zdz + \int_3^4 .0546875zdz + \int_4^5 .1640625zdz + \int_5^6 .2734375zdz$$

$$+ \int_6^7 .27347zdz + \int_7^8 .1640625zdz + \int_8^9 .0546875zdz + \int_9^{10} .0078125zdz$$

$$= 6 \ ,$$

and

$$E(z^2) = \int_2^3 .0078125z^2dz + \int_3^4 .0546875z^2dz + \int_4^5 .1640625z^2dz$$

$$+ \int_5^6 .2734375z^2dz + \int_6^7 .273475z^2dz + \int_7^8 .1640625z^2dz$$

$$+ \int_8^9 .0546875z^2dz + \int_9^{10} .0078125z^2dz = 37.83333 \ ,$$

$$\sigma_Z^2 = E(z^2) - (E(z))^2 = 37.83333 - 36 = 1.83333 \ .$$

Table 9.9 shows the per cent error of the variance of the realization times of nodes 2,3 and 4 using DPD and CIM approaches.

Table 9.9

Node No.	Exact Variance	Discrete Approx.	Percent error Discrete from Exact.	CIM Approx.	Percent error CIM from Exact.
2	.583333	.5	-14.3	.583333	0
3	1.1666667	1.0	-14.3	1.333333	+14.3
4	1.75	1.5	-14.3	1.833333	+ 4.8

Example 2:

Reconsider example 1 of Chapter 8 which is shown in Figure 8.3.

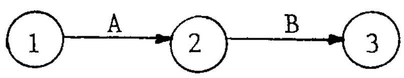


Figure 8.3

The activities are similar and duration times of activities are normally distributed with mean 100 and variance 100.649.

Distribution percentile of the addition of two normally distributed variables with these values for mean and variance using 40 cells for input distribution with optimal squeezing at 0.166% obtained in Yong's (1985) study using CIM approach is given in Table 9.10.

Table 9.10

CUMULATIVE			PERCENTAGE			PROBABILITY		VALUES
0	1	5	25	50	75	95	99	100
152.000	167.083	176.592	190.343	200.00	209.657	223.408	232.917	248.00

Recall that in Chapter 8 the exact pdf of project completion time obtained as shown in Table 8.7.

In order to compare the accuracy of the approximate pdf obtained using CIM approach with that obtained using DPD we use the procedure of Chapter 8, i.e. the true pdf is compared with the approximate pdf obtained using CIM approach to determine maximum absolute deviation (MDV), and the average value of the absolute deviations (ADV) between two distributions, then these values are compared with the corresponding values obtained in Chapter 8 for example 1.

The application of the linear interpolation to Tables 9.10 and 8.7 led to the exclusion of the first and last three realizations of Table 8.7. Table 9.11 has the complete output of the linear interpolation.

Table 9.11: Comparison of the Exact and the CIM approximate Probability Distribution Functions of Example 2.

I	REALIZATION	EXACT PROB.	APRXMTD PROB.	ACTUAL DIFFERENCE
1	167.083	.0106888	.01	-.0006888
2	176.592	.0497070	.05	.0002930
3	190.343	.2500391	.25	-.0000391
4	200.000	.5000000	.50	0.0
5	209.657	.7499698	.75	.0000302
6	223.408	.9502780	.95	-.0002780
7	232.917	.9892797	.99	.0007203

The Average of the absolute Values of the Deviations = 0.0002927

The Maximum of the Absolute Values of the Deviations = 0.0007203. It is No. 7.

Number of Positive Deviates = 3.

Number of Negative Deviates = 3.

Figure 9.12 is a digital plot of the first three columns of Table 9.11.

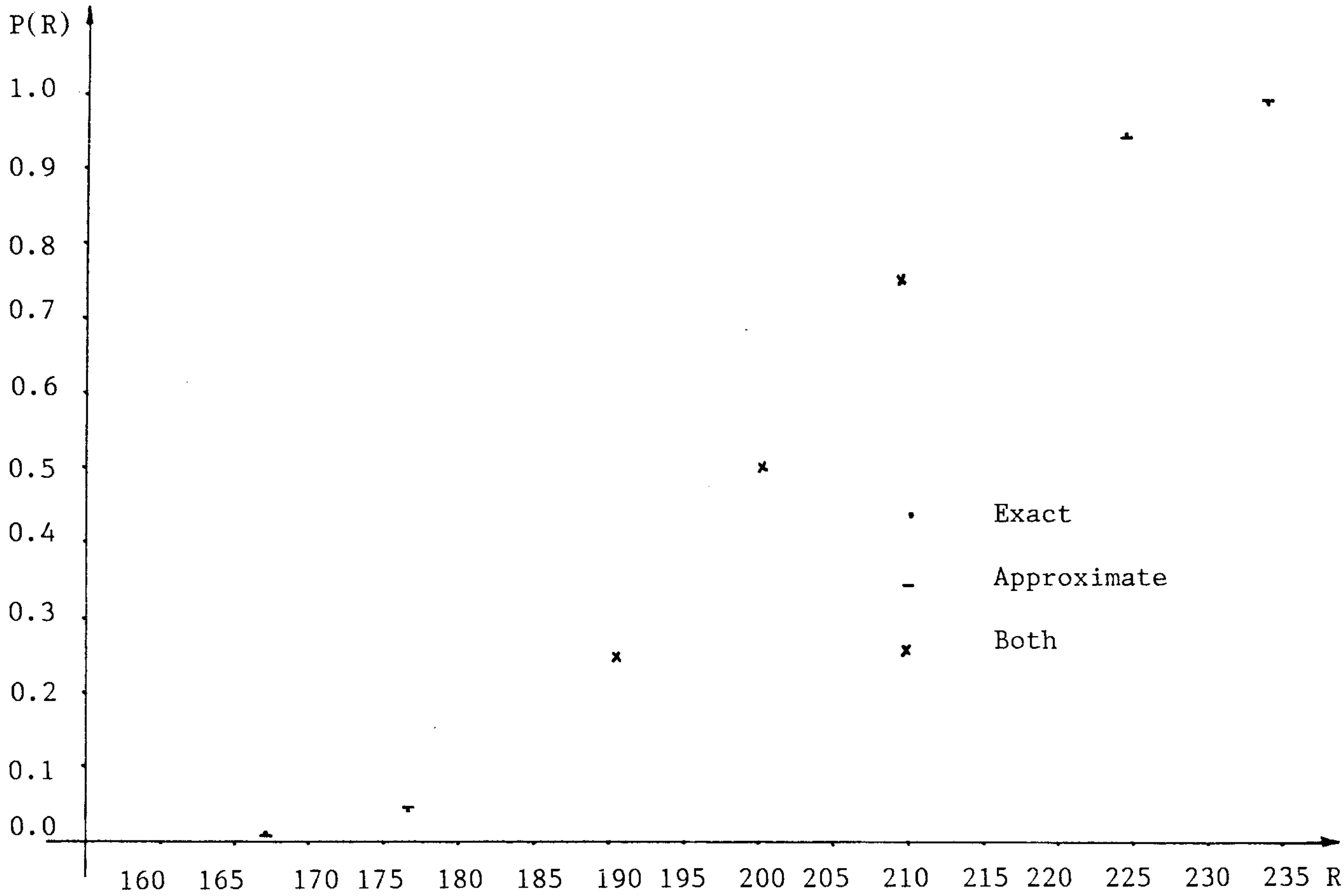


Figure 9.12

Comparison of the Exact and the Approximate Probability Distribution Functions of Example 2 using CIM approach.

Example 3:

Reconsider example three of Chapter 8. Recall that in that example it is assumed that duration times of activities A and B of Figure 8.9 are similar and have Beta distribution with parameters r=3 and s=5.

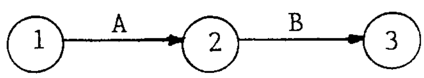


Figure 8.9

Distribution percentile of the addition of the two Beta distribution with parameters r=3 and s=5 using 30 cells for input distribution without squeezing obtained in Yong's (1985) study using CIM approach is given in Table 9.12.

Table 9.12

CUMULATIVE		PERCENTAGE			PROBABILITY		VALUES	
0	1	5	25	50	75	95	99	100
0.000	0.2717	0.3857	0.5849	0.7411	0.9057	1.1448	1.3102	2.000

Recall that in Chapter 8, the pdf of project completion time obtained using interpolation as shown in Table 8.16 assumed to be error free and considered as "true" pdf.

In the following the "true" pdf of Table 8.16 is compared with the approximate pdf of Table 9.12.

The application of the linear interpolation to Tables 9.12 and 8.16 led to the exclusion of the first 13 and the last 26 realizations of Table 8.16. Table 9.13 has the complete output of the linear interpolation. Figure 9.13 is a digital plot of the first three columns of Table 9.13.

Table 9.13: Comparison of the Exact and CIM approximate Probability Distribution Functions of Example 3.

I	REALIZATION	EXACT PROB.	APRXMTD PROB.	ACTUAL DIFFERENCE
1	0.2717	0.0089014	.01	.0010986
2	0.3857	0.0473119	.05	.0026881
3	0.5849	0.2485721	.25	.0014279
4	0.7411	0.5007029	.50	-.0007029
5	0.9057	0.7514220	.75	-.0014220
6	1.1448	0.9507506	.95	-.0007506
7	1.3102	0.9908812	.99	-.0008812

The Average of the Absolute Values of the Deviations = .0012816

The Maximum of the Absolute Values of the Deviations = .0026881. It is No. 2.

Number of Positive Deviates = 3.

Number of Negative Deviates = 4.

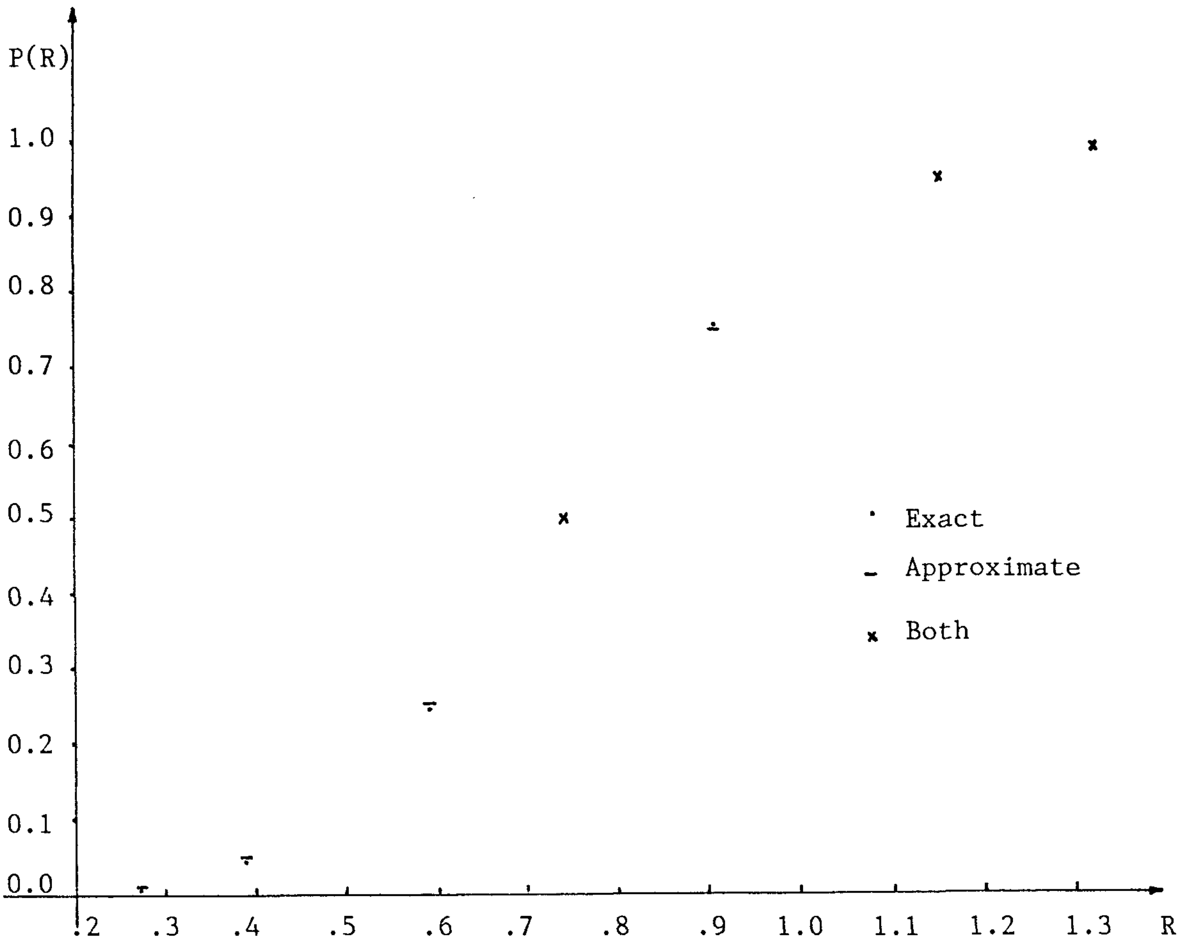


Figure 9.13

Comparison of the Exact and the Approximate Probability Distribution Functions of Example 3 using CIM approach.

Example 4:

Reconsider example five of Chapter 8. Recall that in example 5 of Chapter 8 it is assumed that duration times of activities of Figure 8.9 are similar and have Beta distribution with parameters r=2 and s=8.

Distribution percentile of the addition of the two Beta distribution with parameters r=2 and s=8 using 30 cells for input distribution without squeezing obtained in Yong's (1985) study using CIM approach is given in Table 9.14.

Table 9.14

CUMULATIVE		PERCENTAGE		PROBABILITY		VALUES		
0	1	5	25	50	75	95	99	100
0.000	0.080	0.147	0.2729	0.3823	0.5097	0.7144	0.8644	2.0000

Recall that in Chapter 8, the pdf of project completion time obtained using 100 cells for input distribution as shown in Table 8.25 assumed to be error free and considered as "true" pdf.

In the following the "true" pdf of Table 8.25 is compared with the approximate pdf of Table 9.14.

The application of the linear interpolation to Table 9.14 and 8.25 led to the exclusion of the first 8 and last 44 realizations of Table 8.25. Table 9.15 has the complete output of the linear interpolation. Figure 9.14 is a digital plot of the first three columns of Table 9.15.

Table 9.15: Comparison of the Exact and the CIM approximate Probability Distribution Functions of Example 4.

I	REALIZATION	EXACT PROB.	APRXMTD PROB.	ACTUAL DIFFERENCE
1	0.08	0.0035966	.01	.0064034
2	0.147	0.0350046	.05	.0149954
3	0.2729	0.2267266	.25	.0232734
4	0.3823	0.4782564	.50	.0217436
5	0.5097	0.738165	.75	.011835
6	0.7144	0.9475168	.95	.0024832
7	0.8644	0.9889322	.99	.0010678

The Average of the Absolute Values of the Deviations = .0116859

The Maximum of the Absolute Values of the Deviations = .0232734. It is No. 3.

Number of Positive Deviates =7.

Number of Negative Deviates =0.

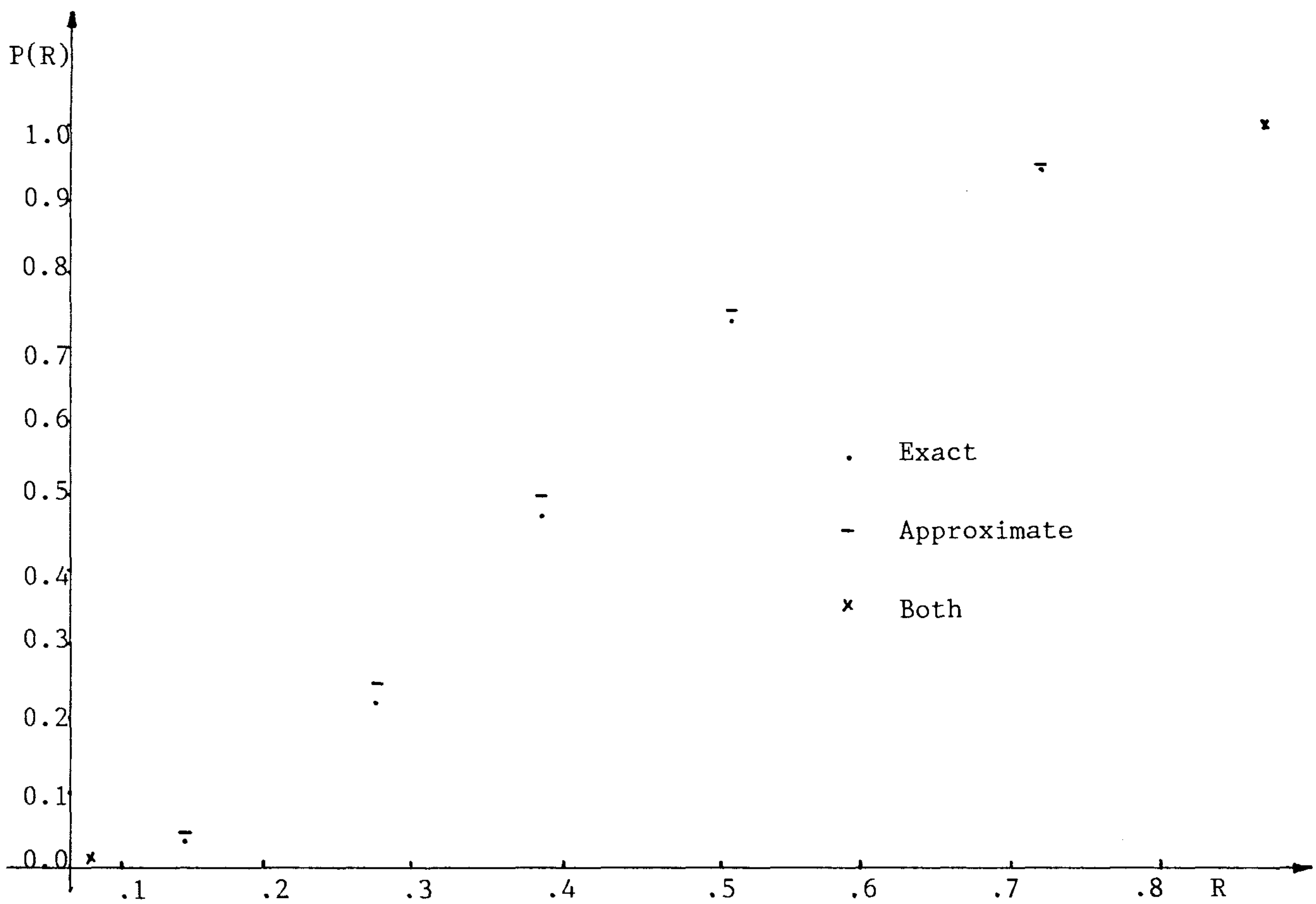


Figure 9.14

Comparison of the Exact and the Approximate Probability Distribution Functions of Example 4 using CIM approach.

SUMMARY AND CONCLUSIONS

In the first part of this chapter the major points of CIM approach has been explained through an example.

The CI approach to adding independent probabilisitc variables is simple in its basic common interval form, and very flexible in its generalised forms. It allows specification effort, specification error, computation error and computation effort to be controlled in an integrated, effective and efficient manner.

Comparing with the other approaches, moment-based, DPD and function integration approaches may be preferred in some cases, but only in very special circumstances, and numerical integration and Monte Carlo approaches offer no advantages.

In the final part the simple examples of Chapter 8 are solved using CIM approach in order to compare the accuracy of the two approximate approaches, i.e. CIM and DPD.

To compare the approximate pdf of the project completion times obtained using CIM approach for examples 2,3 and 4 with the "true" pdfs obtained in Chapter 8 we use the following measures of performance.

1 - Mean value of the project completion time.

2 - Standard Deviation.

3 - Coefficient of Skewness.

4 - The maximum of the absolute values of the deviations (MDV).

5 - The average of the absolute values of the deviations (ADV).

Table 9.16 shows how the distribution type affects the measures of performance. In each of the three problems considered in Table 9.16, the approximate mean is exactly same as the exact mean; the approximate standard deviation is slightly higher than exact standard deviation, it is within 1.89% of the exact standard deviation; the approximate coefficient of skewness is less than exact coefficient of skewness, it is within 5.43% of the exact coefficient of skewness.

The values of the MDV and the ADV vary with the shape of input distribution.

MDV is always less than .0233 and ADV is less than .012. The smallest values of MDV and ADV were obtained in example 2 where the activities are normally distributed.

Table 9.16

Sensitivity of the approximating method to PDF's

Example	Distribution Type	Comparison of the Approximate PDF with the True PDF							
		Mean		Standard Deviation		Coefficient of Skewness		MDV	ADV
		APRX.	EXACT	APRX.	EXACT	APRX.	EXACT		
2	Normal	200	200	14.188	14.187952	0	0	.0007203	.0002927
3	Beta(Moderate Skewed)	.75	.75	.2306512	.2282174	.2122	.2190899	.0026881	.0012816
4	Beta(Extreme Skewed)	.4	.4	.1737814	.1705602	.5545	.5863019	.0232734	.0116859

Table 9.17 shows the accuracy of the two approximate pdfs, DPD and CIM.

Table 9.17

Comparison of the two approximate PDFs

Example	Mean				
	Exact	Discrete Approxi.	Percent error Discrete from Exact	CIM Appr.	Percent error CIM from Exact
2	200	200.00695	+0.00347	200	0
3	.75	.7385594	-1.52541	.75	0
4	.4	.3998142	-0.04645	.4	0

Example	Standard Deviation				
	Exact	Discrete Approxi.	Percent error Discrete from Exact	CIM Appr.	Percent error CIM from Exact
2	14.187952	14.228808	+0.28796	14.188	+0.00033
3	.2282174	.2200811	-3.56515	.2306512	+1.06643
4	.1705602	.1594775	-6.49782	.1737814	+1.8886

Example	Coefficient of Skewness				
	Exact	Discrete Approxi.	Percent error Discrete from Exact	CIM Appr.	Percent error CIM from Exact
2	0	0	0	0	0
3	.2190889	.2346219	+7.08981	.2122	-3.14434
4	.5863019	.6053254	+3.24465	.5545	-5.42415

Example	MDV		ADV	
	Discrete Approximate	CIM Approximate	Discrete Approximate	CIM Approximate
2	.0245581	.0007203	.0023969	.0002927
3	.0345425	.0026881	.0171820	.0012816
4	.0525366	.0232734	.0275114	.0116859

Table 9.17 shows that in all of the cases except for the coefficient of skewness of example 4, the CIM approximate is more accurate than DPD approximate. This conclusion can be generalised to other operations, including 'greatest', multiplication and division operations.

CHAPTER 10: SOLVING STOCHASTIC PERT NETWORKS USING MONTE CARLO METHODS

INTRODUCTION

As mentioned in Chapter 2, whenever analytical approaches fail for solving stochastic networks one alternative is to approximate the pdf of project completion time by approximating each activity distribution with a simpler one, but since the approximation approaches are not reliable, many analysts have turned to Monte Carlo methods to derive approximation solutions.

Monte Carlo Simulation is a powerful technique for attacking many problems related to PERT networks. In particular, within any prescribed bounds on the errors, one can determine factors such as Mean, Variance, pdf of the project completion time and the criticality index of each activity and path.

The first section of this chapter provides a more detailed discussion of this method. In the second section the accuracy and the computation efficiency of the Monte Carlo method is compared with the accuracy and the computation efficiency of the CIM approach through simple networks.

Straightforward Sampling or Crude Monte Carlo

Van Slyke (1963) develops the idea of using crude Monte Carlo Simulation as a tool for finding the pdf of a PERT network's completion time. In crude Monte Carlo Simulation we apply the longest path algorithm to a long series of realizations, each one obtained by assigning a sample value to every activity drawn from its proper distribution. Then we use standard statistical methods to estimate the distribution and parameters of interest. Reconsider following four assumptions of classical PERT model:

1 - The duration of activities are unimodal-in particular, they have beta distribution.

2 - The activities are independent.

3 - The critical path is sufficiently longer than any other path so that the probability of a realization having a different critical path is negligible.

4 - The critical path has enough activities so that central limit theorem applies. Hence, if T_N represent the realization time of node N, then T_N is approximately Normally distributed with a mean:

$$E(T_N) = g_N = \sum_{(ij)\in T_c} E(Y_{ij})$$

and a variance: (2.2)

$$\mathrm{Var}(T_N) = \sum_{(ij)\in T_c} \mathrm{Var}(Y_{ij})$$

where T_c is the critical path obtained from the PERT calculations. The Monte Carlo Simulation approach requires none of these assumptions. Assumption (2), that of independence among activities, is not a necessary condition although it is usually invoked to simplify the sampling procedure. Whether the activities are independent or not, the estimates of the mean and variance of T_N are unbiased.

Precision of the Monte Carlo Method

Denote the mean of the project completion time (T_N) by μ_N, the variance of T_N by σ_N^2, and the pdf of T_N by $F_N(t)$.

Assume the structure of the network and the pdf of all activities $(ij)\in A$ (i.e. $F_{ij}(Y_{ij})$ are given).

All three parameters, μ_N, σ_N^2, and $F_N(t)$ are unknown.

The first parameter we shall consider is the mean. Each complete realization is considered as one sample observation, and the estimator we consider is the sample mean.

Let $l_N(t)$ be the random variable corresponding to project duration, $l_N(k)$ denote the "length" of the CP on the kth sample realization; $\bar{l}_N$ denote the sample mean duration and K, the total number of sample taken. Then $(l_N(1), l_N(2), \ldots, l_N(K))$ is our sample. The random variable

$$\bar{l}_N = \frac{\sum_{k=1}^{K} l_N(k)}{K} \tag{10.1}$$

is the estimator of $E\{l_N(t)\} \equiv \mu_N$.

Let $\sigma_N^2 = \text{Var}\{l_N(t)\}$,

then we have

$$\text{Var}(\bar{l}_N) = \sigma_N^2/K \ . \tag{10.2}$$

Since the number of samples is very large, the central limit theorem tells us that $\bar{l}_N$ is to a very good approximation normal with mean μ_N and variance σ_N^2/K. Notice that $\bar{l}_N$ is an unbiased estimator of μ_N. The precision of this estimator is generally measured by its variance. Unfortunately, if there is a long chain of activities in series, then σ_N^2 becomes large, and a very large sample size is required in order to determine $E\{l_N(t)\}$ precisely.

Suppose with probability 0.99 we want our estimate of the mean to be within 0.01 σ_N of its true value. That is, we want to choose K so that

$$\Pr\{\mu_N - 0.01\,\sigma_N \leqslant \bar{l}_N \leqslant \mu_N + 0.01\,\sigma_N\} \geqslant 0.99$$

which is equivalent to

$$\Pr\left\{\frac{-0.01\,\sigma_N}{\sigma_N/\sqrt{K}} \leqslant \frac{\bar{l}_N - \mu_N}{\sigma_N/\sqrt{K}} \leqslant \frac{0.01\,\sigma_N}{\sigma_N/\sqrt{K}}\right\} \geqslant 0.99 \ ;$$

that is

$\Phi(-0.01\sqrt{K}) \leqslant 0.005,$

which yields that

$$0.01\sqrt{K} \geqslant 2.58,$$

or $$K \geqslant 66564 \text{ samples.}$$

Next we consider the variance, σ_N^2,

Its best unbiased estimate is given by the sample mean-squared deviation S_N^2 given by

$$S_N^2 = \frac{\sum_{k=1}^{K} (1_N(k) - \bar{1}_N)^2}{K}, \qquad (10.3)$$

(notice that in order to obtain an unbiased estimate of σ_N^2 the denominator should be (K-1), but since K is of order of several thousands, the error in using this simpler expression is negligible).

Suppose we wish to know with probability 0.99 that our estimate of σ_N^2 is correct within 2 percent; that is, we wish to choose K such that

$$\Pr\{0.98\,\sigma_N^2 \leqslant S_N^2 \leqslant 1.02\,\sigma_N^2\} \geqslant 0.99$$

but

$$\Pr\{0.98\,\sigma_N^2 \leqslant S_N^2 \leqslant 1.02\,\sigma_N^2\} = \Pr\{0.98K \leqslant \frac{K.S_N^2}{\sigma_N^2} \leqslant 1.02K\}.$$

In Chapter 2 we argued that T_N is not normally distributed, but as a first order of approximation let us assume it is normally distributed. With this assumption we know that $K.S_N^2/\sigma_N^2$ is distributed as the χ^2 distribution with K-1 degrees of freedom, and for large K is approximately N(K-1,2(K-1)).

Therefore,

$$\Pr\{0.98K \leqslant (K.S_N^2)/\sigma_N^2 \leqslant 1.02K\} = \Pr\{\frac{1-0.02K}{\sqrt{2(K-1)}} \leqslant \frac{(K.S_N^2/\sigma_N^2)-(K-1)}{\sqrt{2(K-1)}} \leqslant \frac{1+0.02K}{\sqrt{2(K-1)}}\},$$

$$= 1 - 2\,\Phi(\frac{1-0.02K}{\sqrt{2(K-1)}}).$$

Therefore,

$$1 - 2\,\Phi(\frac{1-0.02K}{\sqrt{2(K-1)}}) \geqslant .99,$$

or

$$\Phi\left(\frac{1-0.02K}{\sqrt{2(K-1)}}\right) \leqslant 0.005 \ ;$$

hence

$$\frac{1-0.02K}{\sqrt{2(K-1)}} \leqslant -2.58 \ ,$$

which yields the sample size $K \doteq 33300$.

Thirdly, consider the measure of criticality of one activity. In any realization of the network, the activity (ij) is either on the critical path or it is not. Recall that the probability of an activity being on the critical path is given by its criticality index, which is equal to

$$CA(ij) = \sum_{\substack{T_1 \\ ((ij)\in T_1)}} CP(T_1) \ . \qquad (6.2)$$

Therefore we need sampling from a binomial distribution which is approximated very well by the Normal distribution function for the fixed sample. The estimator is the ratio of the number of sample realizations for which the arc is critical to the total sample size. The mean is KP and the variance KP(1 -P), where P is the probability of being critical.

Suppose with probability of 0.99 we want our estimator of P to be correct within 0.01.

Let $\hat{P}$ denote the estimator of P, we know that $(\hat{P} - P)/\sqrt{P(1-P)/K}$ is asymptotically N(0,1). Therefore we want

$$\Pr\{\ |\hat{P} - P| > 0.01\} = 0.99,$$

or

$$\Pr\left\{\frac{|\hat{P} - P|}{\sqrt{P(1-P)/K}} > \frac{0.01}{\sqrt{P(1-P)/K}}\right\} = 0.99.$$

P is unknown but we can substitute $\hat{P}$ in the right hand side of the inequality sign in place of P

$$\Pr\{ \frac{|\hat{P}- P|}{\sqrt{P(1-P)/K}} > \frac{0.01}{\sqrt{\hat{P}(1-\hat{P})/K}} \} = 0.99 ,$$

or better still, we can be conservative and substitute the worst possible value, $\hat{P}=0.5$.

Therefore

$$\Pr\{\frac{|\hat{P} - P|}{\sqrt{P(1-P)/K}} > \frac{0.01}{1/2 \sqrt{K}}\} = 0.99,$$

or

$$\Pr\{ - 0.02 \sqrt{K} \leqslant \frac{\hat{P} - P}{\sqrt{p(1-P)/K}} \leqslant 0.02 \sqrt{K} \} = 0.99,$$

This implies that

$$0.02 \sqrt{K} = 2.58, \text{ or } K \doteq 16650 .$$

Finally, consider the estimation of the pdf of T_N, denoted by F_N. We want to know, in some sense, how well our sample pdf fits the real one. If we assume that the pdf of the project completion time is continuous, we can make probability statements about the greatest absolute difference between the sample pdf and the true one independent of the distribution itself (Wilks, 1962).

Kolmogorov gave the asympotic results that are tabulated, for example in Hole (1954), p.4.

Let $$D_K = \sup_t \{|\hat{F}_{NK}(t) - F_N(t)|\},$$

where $F_N(t)$ is the true pdf and $\hat{F}_{NK}(t)$ is the pdf derived from a sample of size K. [sup is used instead of max since, depending on the situation, there might not be a value of t such that $D_K=|\hat{F}_{NK}(t) - F_N(t)|$. However, it is always true that no number strictly smaller than D_K can be greater than or equal to $|\hat{F}_{NK}(t) - F_N(t)|$ for every t.]

The probability that D_K is less than some specified $d/\sqrt{K}$ is asymptotically given by

$$\lim_{K \to \infty} \Pr(D_K \leqslant d/\sqrt{K}) = \sum_{i=-\infty}^{\infty} (-1)(-1)^i e^{-2i^2d^2}. \quad (10.4)$$

In the tables we find d and the asymptotic probability of $\sqrt{K}.D_K \leqslant d$.

For example, suppose we wish to estimate $F_N(t)$ by $\hat{F}_{NK}(t)$ such that the maximum deviation between the two functions does not exceed 0.01 in absolute value more than 1% of the time. We find that for the asymptotic probability $\Pr(\sqrt{K}.D_K \leqslant d) = 0.99$ we have d=1.63. Since $D_K=0.01$, we solve for $\sqrt{K}(0.01) \geqslant 1.63$, to obtain $K \geqslant 26570$. Of course, knowing the pdf (or the closest thing to it, the sample pdf) enables one to make probability statements concerning the completion time of the project. Unlike such probability statements made with the classical PERT calculations, these statements are accurate within the limit specified (1% in our example).

Crude Monte Carlo simulation does not require extensive assumptions, but it is notorious for being computationally costly. Generally speaking, the result obtained from any form of Monte Carlo methods are subject to what is termed "sampling error" or "sampling variance". This means that a large number of repeated samples must be taken to obtain a reasonably close estimate of the desired result. But to use such an approach would increase the computation cost. As a result, a number of techniques have been developed for reducing the variance without increasing the sample size and number of repetitions. We briefly discussed four of them in Chapter 2 including, (a) Antithetic Variates, (b) Stratified Sampling, (c) Control Variates, and (d) Conditional Sampling.

Since the antithetic variate method is the most widely known and easiest to apply of these techniques following section is devoted to a more detailed discussion of this technique. For more detailed discussion of the other techniques see Hammersely and Handscomb (1967) and the references cited therein.

Antithetic Variate Method

Perhaps the best way to explain the concept of antithetic variate is to refer to a simple network of Figure 10.1, which is composed of two activities in series.

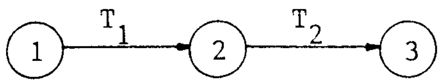


Figure 10.1

Suppose we wish to estimate the expected completion time of the project $T_3 = T_1 + T_2$ by crude Monte Carlo simulation, where T_1 and T_2 are assumed to be independent.

The straightforward procedure is to select two uniform random numbers $0 \leqslant R_1 \leqslant 1$, and $0 \leqslant R_2 \leqslant 1$, and transform to realizations T_1 and T_2 by use of their respective dfs, say

$$T_i = F_i^{-1}(R); \quad i=1,2$$

where F_i is the df of T_i and F_i^{-1} is its inverse. Then if we denote a realization of T_3 by T_3^t for, t=1,...,K. Then the first realization is given by $T_3^1 = T_1^1 + T_2^1$.

We could tabulate K such realization and average to obtain the estimate of the mean

$$\bar{T}_3 = \frac{T_1^1 + T_2^1 + T_1^2 + T_2^2 + \ . \quad . \quad . \ + T_1^K + T_2^K}{K} \qquad (10.5)$$

All random numbers are independent, and the K times are generated from their appropriate distributions, so

$$E(\bar{T}_3) = E(T_1) + E(T_2) \qquad (10.6)$$

and

$$\mathrm{Var}(\bar{T}_3) = \frac{\mathrm{Var}(T_1) + \mathrm{Var}(T_2)}{K} \tag{10.7}$$

In other words, the simple procedure described gives an unbiased estimator of $E(T_3)$ whose variance decreases as $1/K$. If the procedure described is repeated many times ($K \to \infty$), then $\bar{T}_3$ becomes arbitrarily close to $E(T_3)$.

As mentioned before, if there are many independent serial activities then the sum of the variance in the numerator of (10.7) becomes large, and many repetitions are required in order to determine $E(T_3)$ accurately.

Note that in order to estimate $E(T_3)$ two realizatins, T_3^k and T_3^{k+1}, need not be independent, so long as they have the correct marginal distributions. Therefore, if T_3^k is large and one "forces" T_3^{k+1} to be small, then the average will tend to be closer to the true value than in the case of purely independent samples. This is accomplished in the following fashion.

Let R' be the complement of R. That is, $R'=1 - R$. Two negatively correlated realizations of the ith activity duration are $T_i=F_i^{-1}(R)$ and $T_{i'}=F_i^{-1}(R')$. The two activities of the network are sampled in this fashion, yielding a pair of realizations of project completion time. For K pairs of realizations, the stream T_3^1, $T_{3'}^1$, T_3^2, $T_{3'}^2$, T_3^3, $T_{3'}^3$, ... T_3^K, $T_{3'}^K$ is generated and can be used to estimate $E(T_3)$.

The antithetic variate estimator for $E(T_3)$ is

$$\bar{T}_3^{a} = \sum_{k=1}^{K} \frac{T_3^k + T_{3'}^k}{2K}$$

$$= \frac{1}{2} \left\{ \sum_{k=1}^{K} \frac{[T_1^k + T_2^k]}{K} + \sum_{k=1}^{K} \frac{[T_{1'}^k + T_{2'}^k]}{K} \right\}$$

$= \frac{1}{2} (\bar{T}_3 + \bar{T}_{3'})$; the superscript a is for "antithetical"

$E(\bar{T}_3)=E(\bar{T}_{3'})=E(T_3)$, so that the estimate of $\bar{T}_3^a$ is, indeed, unbiased. Furthermore, $\text{Var}(\bar{T}_3) = \text{Var}(\bar{T}_{3'}) = [\text{Var}(T_1) + \text{Var}(T_2)]/K$.

It is apparent from the manner of selection of the antithetic variables T_i and $T_{i'}$, that $\bar{T}_3$ and $\bar{T}_{3'}$ are negatively* correlated; that is, $\text{Cov}\,[\bar{T}_3, \bar{T}_{3'}]<0$. Since,

$$\text{Var}(\bar{T}_3^a) = \frac{1}{4}\,[\text{Var}\,(\bar{T}_3) + \text{Var}(\bar{T}_{3'})] + \frac{1}{2}\ \text{Cov}\,[\bar{T}_3, \bar{T}_{3'}]$$

$$= \frac{1}{2}\ \text{Var}\,(\bar{T}_3) + \frac{1}{2}\,[\text{Cov}\ \bar{T}_3, \bar{T}_{3'}] < \frac{1}{2}\ \text{Var}\,(\bar{T}_3)$$

[* Kleijnen (1975) observes that for complicated simulations, negative correlation cannot be proven. However, experiments with moderately complex system show that negative correlation does indeed occur (Sullivan et al, 1982)].

The conclusion is that the use of antithetic variates is more efficient than doubling the total number of independent samples taken. In other words, with the same sample size, the use of antithetic variates would result in a smaller variance of the estimate of the mean, $\bar{T}_3$, and hence a more precise one.

The dramatic effect of antitheticizing a sum is most apparent when T_i is symmetric, e.g., if T_i can be assumed uniform, or normal. In the latter case we can write

$$T_i = \mu_i + \sigma_i\ \rho_i,$$

ρ_i being distributed as N(0,1); μ_i and σ_i are mean and standard deviation of T_i. Then the obvious antithetic is obtained by simple sign reversal of ρ_i

$$T_{i'} = \mu_i - \sigma_i \rho_i$$

and an average of $\frac{T_1 + T_2}{2}$ with $\frac{T_{1'} + T_{2'}}{2}$ already yields a zero-variance estimate of μ_i.

The use of antithetic variates is also helpful in estimating the df of the random vairable T_3. From the first set of observations the empirical pdf of $\hat{F}(t)$, and from the antithetic set of observations the second empirical pdf $\hat{F}'(t)$, can be determined. The antithetic variate estimate of the "true" pdf would then be given by $\hat{F}^a(t)= [\hat{F}(t) + \hat{F}'(t)]/2$, all t. Evidently, $\hat{F}'(t)$ compensate for any deviation of $\hat{F}(t)$ from the true F(t).

The following section presents a summary of the work done by Krisadawat (1986) for comparing the accuracy and the computation efficiency of the Monte Carlo simulation and the CIM approaches for simple stochastic networks.

SOLVING A STOCHASTIC NETWORK USING MONTE CARLO METHODS

The Monte Carlo procedure applied by Van Slyke (1963) to networks assumes each activity has a beta distribution with mean t_e and a variance V_e. As generally recommended by Van Slyke (1963) for this purpose, 10,000 sets of random times for each activity in the network were generated. For each of these sets, the longest path through the network was determined: its duration was noted, as well as a count of each activity on the critical path.

The procedure adopted in Krisadawat (1986) is different from Van Slyke (1963), Cook and Jennings (1979) and Burt and Garman (1971) where the completion times are of interest. The efficiency of the method independent of network complexity issues is of interest in Krisadawat (1986), the longest or critical path is assumed to be known. Non-negative random variables of each activity on the longest path with a specified sample size and distribution, are generated. The expected value E and variance V_t of the completion time are determined by aggregating the expected duration and variance of each activity on the critical path. Finally, since each arc in a network intended to represent paths made up of many activities, for the sake of convenience it is assumed that activities are normally distributed. Hence only methods for generating normal random variables are considered in his study.

Generating normal random variables

It should first be noted that given $X \sim N(0,1)$, $X' \sim N(\mu,\sigma^2)$ can be obtained by setting $X' = \mu+\sigma X$. So our attention will be restricted to generating standard normal random variables.

One of the early methods for generating N(0,1) random variables, due to Box and Muller (1958), is still in wide use despite the current availability of much faster algorithms. The method simply says to generate U_1 and U_2 as uniform (0,1) random variables, then set $X_1 = (-2\ln U_1)^{\frac{1}{2}}\cos 2\Pi . U_2$ and $X_2 = (-2\ln U_1)^{\frac{1}{2}} \sin 2\Pi . U_2$. Then X_1 and X_2 are independently N(0,1) random variables.

An improvement to the Box and Muller method which eliminates the trigonometric calculations, described in Marsaglia and Bray (1964), has become known as "the polar method". It was found by Atkinson and Pearce (1976) to be between 9 and 31 percent faster in FORTRAN programming than the Box and Muller method. The polar method is used in this study. The algorithm of this method which also generates N(0,1) random variables in pairs, is as follows:

1) generate U_1 and U_2 as U(0,1), let $V_i = 2U_i - 1$ for $i=1,2$ and let $W = V_1^2 + V_2^2$.

2) if $W>1$, go back to step 1. Otherwise, let $Y = [(-2\ln W)/W]^{\frac{1}{2}}$, $X_1=V_1Y$ and $X_2 = V_2Y$. Then X_1 and X_2 are independent N(0,1) random variables.

The expected duration and variance of each activity are expressed in terms of one or several (unknown) parameters associated with the probability distribution of activity concerned. Several statistical methods are available: least squares methods, maximum likelihood methods and so on. The method of maximum liklihood is considered in Krisadawat's (1986) study.

Maximum Likelihood Estimates

The maximum likelihood estimate of θ, $\hat{\theta}$ say, based on a random sample $X_1, .., X_n$ is the value of $\hat{\theta}$ which maximizes $L(X_1,...,X_n; \theta)$, considered as a function of θ for a given sample $X_1,..., X_n$ where L is defined by

$$L(X_1,...,X_n; \theta) = f(X_1;\theta)...f(X_n;\theta).$$

In order to find the maximum likelihood estimate, the maximum value of a function must first be determined. Since lnX is an increasing function of X, $\ln L(X_1,...,X_n; \theta)$ will achieve its maximum value for the same value of θ as will $L(X_1,...,X_n; \theta)$. Hence under fairly general conditions, assuming that θ is a real number and the $L(X_1,...,X_n; \theta)$ is a differential functin of θ, the maximum likelihood estimate can be obtained by solving the likelihood equation:

$$\frac{\partial}{\partial\theta} \ln L(X_1,...,X_n; \theta) = 0 .$$

For example, if X has distribution $N(\mu, \sigma^2)$, the maximum likelihood estimates of μ and σ^2 are given by

$$\mu = \frac{1}{n} \sum_{i=1}^{n} X_i = \bar{X} ,$$

$$\sigma^2 = \frac{1}{n} \sum_{i=1}^{n} (X_i - \bar{X})^2 .$$

However, the maximum likelihood yields a biased estimate of σ^2. Thus the unbiased estimate of the form $1/(n-1) \cdot \sum_{i=1}^{n} (X_i - \bar{X})^2$ is used instead. The full derivation can be seen in Appendix B.

Confidence Interval

In order to express the accuracy of the estimate, an interval estimate of the appropriate parameter is determined. This is usually referred to as a "confidence interval" because the method for computing such intervals are expressed in terms of a probability statement.

From our knowledge of the standardized normal distribution, the variables $z = (\bar{X}-\mu)/\sigma_{\bar{X}}$ is normal distributed with mean 0.0 and the standard deviation 1.0, where $\sigma_{\bar{X}}=\sigma/\sqrt{n}$, σ referring to the population of X values. If $\alpha=1-\gamma$, then by using the symmetry of the normal distribution the following relationship can be obtained:

$$P\{ -z_{\alpha/2} \leqslant \frac{\bar{X}-\mu}{\sigma/\sqrt{n}} \leqslant z_{\alpha/2}\} = 1 - \alpha, \qquad (10.8)$$

where $z_{\alpha/2}$ and $-z_{\alpha/2}$ refer to those symmetric Z values for which the probability of greater or less values respectively, is $\alpha/2$. From equation 10.8, the error can be expressed as:

$$\pm \text{ ERROR} = z_{\alpha/2}\ \sigma/\sqrt{n} \ ,$$

or

$$\pm \text{ ERROR\%} = 100.z_{\alpha/2}\ \sigma/(\mu\sqrt{n}) \ .$$

EXPERIMENTS AND RESULTS

The following experiments were designed by Krisadawat (1986) to measure the accuracy and computation efficiency of the Monte Carlo methods. The network considered in these experiments shown in Figure 10.2 is originally presented in Moder and Phillips (1983).

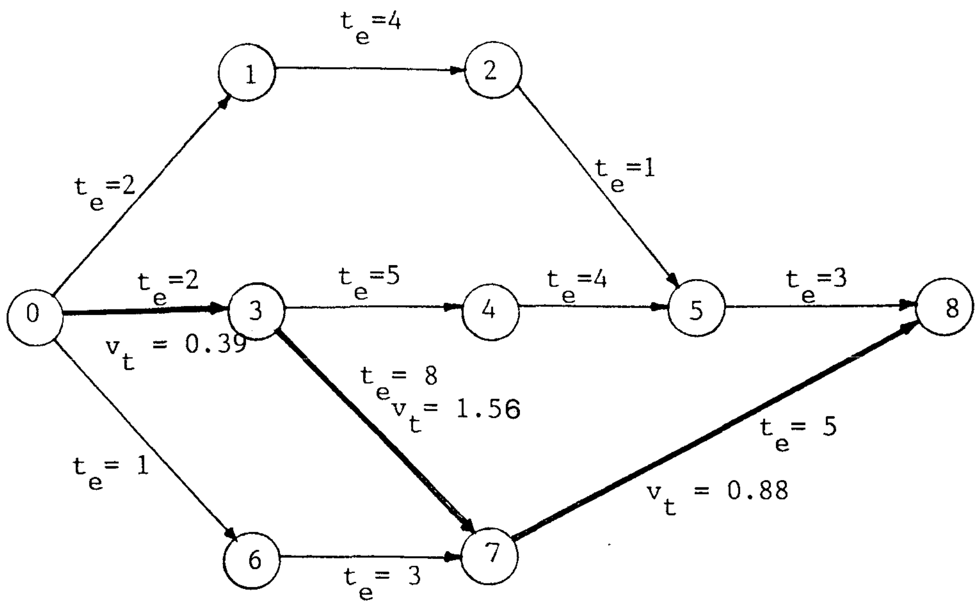


Figure 10.2 PERT network

In this network, the critical path consists of the three activities 0-3, 3-7 and 7-8. Suppose that the performance of each of these activities is subject to a considerable number of chance sources of variation such as the weather, equipments failure, personnel or material problems, or uncertainties in the methods or procedures to be used in carrying out the activity. In other words, the actual performance times for the activities on the critical path, instead of being exactly 2, 8 and 5 days are variables subject to random and chance variation with mean values of 2, 8 and 5 respectively.

Any comparison of the results obtained by the Monte Carlo method and the exact results must take into account not only the computation time but also the degree of accuracy. The exact result is the sum of activity durations of the three activities on the critical path ignoring uncertainties, that is, 2+8+5 = 15. Whereas the results obtained by the Monte Carlo method is the sum of the expected duration and variance of the three activities which are obtained from the maximum likelihood estimates as discussed in the previous section. The computation time is the time taken to generate activity durations with specified sample size and distribution and calculate the expected duration and variance of each activity by the maximum likelihood methods. The degree of accuracy considered is expressed in term of errors and is calculated using the following confidence interval function:

$$\pm \text{ ERROR} = z_{\alpha/2}\sigma/\sqrt{n}$$

where

$z_{\alpha/2}$ is the standard normal deviate,

σ is the square root of the sum of variance of the durations on the critical path,

n is the sample size,

α is the confidence level.

and

$$\pm \text{ ERROR} = 100.z_{\alpha/2}\sigma/(\mu\sqrt{n})$$

where

μ is the exact sum of duration of the three activities on the critical path.

ln deriving the error figures in Table 10.1, 10.2 and 10.3, five confidence levels are considered: 10%, 5%, 1%, 0.5% and 0.1%. In determining the efficiency of the method, a number of tradeoffs may be involved.

As pointed out by Van Slyke (1963), in solving PERT problems using Monte Carlo methods, most of the time is consumed in generating random variables. He recommended that 10,000 samples should be taken as a standard for network to obtain a reasonable precision. However, computation time and precision are conflicting objectives and tradeoff regarding these two factors must be included in determining the efficiency of Monte Carlo methods.

Computer Program For Simulation

To accomplish the objective above, a FORTRAN computer program was written. The main program consists of a function and three subroutines; RAND for generating random numbers; NORSAMP for generating random samples of activity durations; BARVAR for determining the expected mean and variance of each activity and ERRSIG for calculating the error percentages at various confidence levels. For the function RAND, the portable FORTRAN function suggested by Schrage (1979) is used.

For the subroutine NORSAMP, it is assumed that the activity durations on the critical path are distributed normally with the specified mean and standard deviation. The subroutine uses the polar method for generating random sample, as described previously. For subroutine BARVAR the expected mean and variance of each activity are determined by maximum likelihood estimates, as described previously. The subroutine ERRSIG has already been explained in the previous section. The listing and flow-chart of the program can be seen in Appendix C.

Results

The results of the experiments are summarized in Tables 10.1, 10.2, and 10.3. All the experiments were run on a IBM PC XT personal computer. The initial sample size of each activity on the critical path was 300 and this is gradually increased to 9000. Empirical examination of the results in Table 10.1 shows that the completion time obtained by the Monte Carlo method differs from the exact completion time, but not to a significant extent. For 10%, 5%, 1%, 0.5% and 0.1% confidence intervals, the errors are -.1618, -.1928, -.2533, -.2767 and -.3236 or 1.08% 1.29%, 1.69%, 1.84% and 2.16% respectively. As the sample size of each activity increases, it is clear from Table 10.2 and Table 10.3 that the errors associated with the expected values at all significant levels gradually decrease. Figure 10.3 presents a pictorial relationship between the errors and sample size at 10%, 5%, 1%, 0.5% and 0.1%. Looking at Figure 10.3, the errors at all significant levels decreases as the sample size get larger. However the decrease becomes less significant as the sample size of the three activities goes beyond 5000. Thus the errors can be reduced by increasing the sample size up to a sample size of about 5000 but beyond this the gain in precision is not significant, and it must be bought at the cost of substantially increased computation time. The cost of running a computer program is linearly proportional to the computation time and the sample size. As already pointed out by Van Slyke (1963), most of the time consumed using Monte Carlo methods is in generating random variables. Hence, the larger the sample size, the longer the time needed and consequently the bigger cost to be met. However, examination of the computation time from the three tables suggests that the Monte Carlo method is reasonably efficient in term of computation time and error up to sample size of about 10,000.

COMPUTATION TIME:

Ohr Omin 8sec 78hundredsec

EXACT :	MEAN	=	15.00
	VARIANCE	=	2.8300
MONTE CARLO :	MEAN	=	14.87
	VARIANCE	=	2.9017

SAMPLE SIZE = 300

SIG LEVEL	ERROR	ERROR%(+ -)
10.00	-.1618	1.08
5.00	-.1928	1.29
1.00	-.2533	1.69
0.50	-.2767	1.84
0.10	-.3236	2.16

Table 10.1. Experimental result with the sample size of 300.

COMPUTATION TIME :

0hr	0min	51sec	95hundredsec

EXACT : MEAN = 15.00
VARIANCE = 2.8300

MONTE CARLO : MEAN = 14.94
VARIANCE = 2.8579

SAMPLE SIZE = 1800

SIG LEVEL	ERROR	ERROR%(+ -)
10.00	-.0655	.44
5.00	-.0781	.52
1.00	-.1026	.68
0.50	-.1121	.75
0.10	-.1311	.87

Table 10.2. Experimental result with the sample size of 1800.

COMPUTATION TIME :

0hr	4min	19sec	68hundredsec

EXACT : MEAN = 15.00
VARIANCE = 2.8300

MONTE CARLO: MEAN = 15.06
VARIANCE = 2.8776

SAMPLE SIZE = 9000

SIG LEVEL	ERROR	ERROR%(+ -)
10.00	.0294	.20
5.00	.0350	.23
1.00	.0461	.31
0.50	.0503	.34
0.10	.0588	.39

Table 10.3. Experimental result with the sample size of 9000.

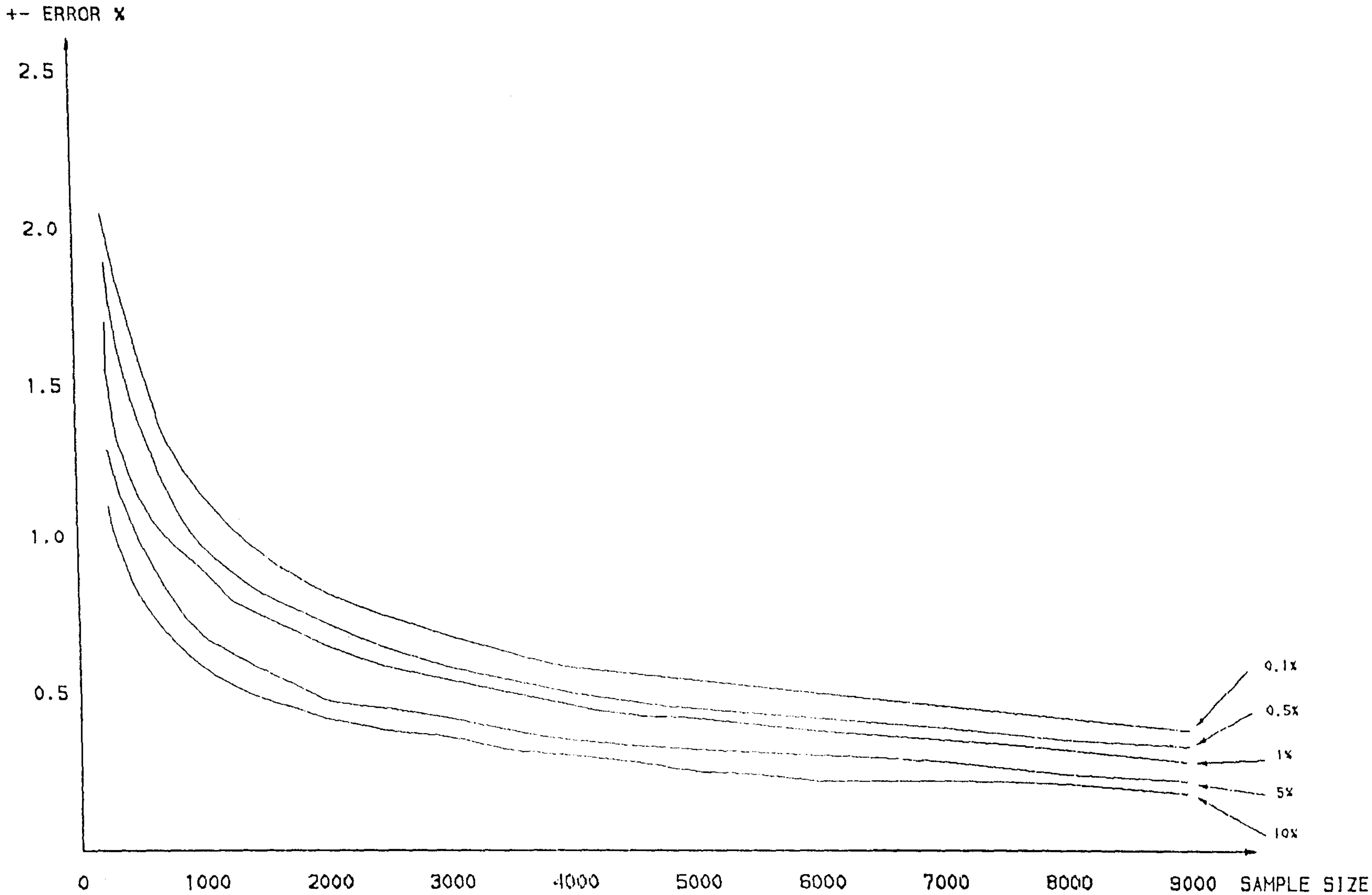


Figure 10.3: Pictorial Relationship Between The Error and Sample Size.

Conclusion

Krisadawat's (1986) study suggests that the Monte Carlo method is efficient for estimating the mean of network completion times in terms of both the computation time and the precision of the estimates. Empirical study favours this method for three reasons. First, a Monte Carlo or simulation program can be written quite easily. Second. the Monte Carlo method is flexible in that the sample size can be chosen to provide a desired level of confidence. Third, the Monte Carlo program considered in this study did not take advantage of variance-reducing techniques. Hence, the use of the variance-reducing techniques as mentioned before would further add to the efficiency of Monte Carlo methods.

Alternatively, the CIM approach can be used to solve the above problem. "CATRAP", the CIM computer software developed by BP, was used by Yong (1985) to solve a similar network problem. He combined five normal distributions of activities, each with the mean and variance of 100. The efficiency was measured in term of the precision of the estimates. The experiments were carried out with different number of cells and a 'squeezing procedure' was introduced to balance out the variance error caused by the probability allocation procedure and the need to truncate probability distribution tails. The 'optimal squeezing' was then obtained by varying the squeezing level for a particular cell size and by interpolating the results obtained corresponding to the zero variance bias. With 40 cells, less than 0.2% of the total probability area is squeezed. Some of the results from this study can be seen in Table 10.4 and Table 10.5.

Comparison of the results from Krisadawat's (1986) study with those of the CIM approach obtained by Yong (1985) in Table 10.4 and Table 10.5 suggests that the CIM approach with about 40 cells and optimal squeezing gives comparable precision to the Monte Carlo method using about 10,000

PERCENTILE	ERRORS IN THE VARIABLE VALUES FOR REPRESENTATIVE PERCENTILES	
	ZERO SQUEEZING	OPTIMAL SQUEEZING
1	0.78	0.28
5	0.50	0.30
25	0.98	0.91
50	-	-
75	0.98	0.91
95	0.50	0.30
99	0.78	0.28

Table 10.4. Error percentages for 40 cells at various percentiles.

PERCENTILE	ERRORS IN THE VARIABLE VALUES FOR REPRESENTATIVE PERCENTILES	
	ZERO SQUEEZING	OPTIMAL SQUEEZING
1	0.14	0.04
5	0.15	0.13
25	0.02	0.01
50	-	-
75	0.02	0.01
95	0.15	0.13
99	0.14	0.04

Table 10.5. Error percentages for 100 cells at various percentiles.

samples as recommended by Van Slyke (1963) with respect to 1 percentile values, much greater precision with respect to mean and variance values. Further examination of Table 10.4 and Table 10.5 also indicates that CIM precision is increased with reductions in percentile values up to the truncation point, whereas Monte Carlo precision declines with reductions in percentiles. Furthermore, very high levels of precision are possible if the CIM approach is used with 100 cells or more.

In term of computation speed, it is clear from this study that Monte Carlo methods provide moderately good results with sample size as small as 300. The CIM approach can only achieve the same with about 20 cells or less which provides unacceptable crudeness in the result distribution shape.

Hence the Monte Carlo method would seem preferable if

a) complex non-sequential dependence structure are involved,

or

b) not too much precision is required.

Where the CIM approach would seem preferable if

a) sequential structure, including branches, are involved;

b) reasonable or high precision is required.

However, it might point out that the CIM approach was developed for and is suitable for only fairly simple networks, although it might be used in conjunction with a Monte Carlo approach for more complex networks.

In the following section we compare the accuracy of the proposed procedure with that of Monte Carlo method through simple examples.

<u>Example 10.1</u>:

Reconsider the network configuration of Figure 10.2. To approximate the normal distribution function of critical activities 0-3, 3-7, and 7-8 each by a discrete distribution using 33 cells for each distribution from the standard normal tables, we obtain the information in Table 10.6.

Table 10.6

z	P(z)	z	P(z)
- ∞	0.00000	0.2	0.07811
-3.2	0.00052	0.4	0.07358
-3.0	0.00093	0.6	0.06661
-2.8	0.00163	0.8	0.05793
-2.6	0.00277	1.0	0.04842
-2.4	0.00454	1.2	0.03890
-2.2	0.00717	1.4	0.03002
-2.0	0.01087	1.6	0.02227
-1.8	0.01588	1.8	0.01588
-1.6	0.02227	2.0	0.01087
-1.4	0.03002	2.2	0.00717
-1.2	0.03890	2.4	0.00454
-1.0	0.04842	2.6	0.00277
-0.8	0.05793	2.8	0.00163
-0.6	0.06661	3.0	0.00093
-0.4	0.07358	3.2	0.00052
-0.2	0.07811	+ ∞	0.00000
0.0	0.07900		

By replacing z in Table 10.6 by $\mu+\sigma z$ where $\mu=2$ and $\sigma = \sqrt{0.39} \cong 0.6245$, we can discretize the continuous distribution of activity 0-3 of Figure 10.1 as shown in Table 10.7.

Table 10.7

z	P(z)	z	P(z)	z	P(z)	z	P(z)
0.0016	0.00052	1.1257	0.03002	2.2498	0.07358	3.3739	0.00717
0.1265	0.00093	1.2506	0.03890	2.3747	0.06661	3.4988	0.00454
0.2514	0.00163	1.3755	0.04842	2.4996	0.05793	3.6237	0.00277
0.3763	0.00277	1.5004	0.05793	2.6245	0.04842	3.7486	0.00163
0.5012	0.00454	1.6253	0.06661	2.7494	0.03890	3.8735	0.00093
0.6261	0.00717	1.7502	0.07358	2.8743	0.03002	3.9984	0.00052
0.7510	0.01087	1.8751	0.07811	2.9992	0.02227		
0.8759	0.01588	2.0000	0.07900	3.1241	0.01588		
1.0008	0.02227	2.1249	0.07811	3.2490	0.01087		

The discrete approximate of activities 3-7 and 7-8 of Figure 10.2 are shown in Tables 10.8 and 10.9 respectively.

Table 10.8

z	P(z)	z	P(z)	z	P(z)	z	P(z)
4.0032	0.00052	6.2514	0.03002	8.4996	0.07358	10.7478	0.00717
4.2530	0.00093	6.5012	0.03890	8.7494	0.06661	10.9976	0.00454
4.5028	0.00163	6.7510	0.04842	8.9992	0.05793	11.2474	0.00277
4.7526	0.00277	7.0008	0.05793	8.2490	0.04842	11.4972	0.00163
5.0024	0.00454	7.2506	0.06661	9.4988	0.03890	11.7470	0.00093
5.2522	0.00717	7.5004	0.07358	9.7486	0.03002	11.9968	0.00052
5.5020	0.01087	7.7502	0.07811	9.9984	0.02227		
5.7518	0.01588	8.0000	0.07900	10.2482	0.01588		
6.0016	0.02227	8.2498	0.07811	10.4980	0.01087		

Table 10.9

z	P(z)	z	P(z)	z	P(z)	z	P(z)
1.9984	0.00052	3.6868	0.03002	5.3752	0.07358	7.0636	0.00717
2.1860	0.00093	3.8744	0.03890	5.5628	0.06661	7.2512	0.00454
2.3736	0.00163	4.0620	0.04842	5.7504	0.05793	7.4388	0.00277
2.5612	0.00277	4.2496	0.05793	5.9380	0.04842	7.6264	0.00163
2.7488	0.00454	4.4372	0.06661	6.1256	0.03890	7.8140	0.00093
2.9364	0.00717	4.6248	0.07358	6.3132	0.03002	8.0016	0.00052
3.1240	0.01087	4.8124	0.07811	6.5008	0.02227		
3.3116	0.01588	5.0000	0.07900	6.6884	0.01588		
3.4992	0.02227	5.1876	0.07811	6.8760	0.01087		

Convoluting the duration times of activities 0-3, 3-7 and 7-8 of Tables 10.7 to 10.9 yields the pdf of the project completion time as shown in Table 10.10 with mean 14.999817 and variance 2.84354.

Notice that the results obtained for mean and variance of the project completion time using discrete approximate is more accurate than those obtained using Monte Carlo simulation.

Table 10.10

t	P(t)	t	P(t)	t	P(t)	t	P(t)
7.3084	.0000178	11.2480	.0033047	15.1876	.0441855	19.1272	.0022921
7.4960	.0000181	11.4356	.0046506	15.3752	.0432850	19.3148	.0016376
7.6836	.0000188	11.6232	.0059706	15.5628	.0419026	19.5024	.0012275
7.8712	.0000203	11.8108	.0074355	15.7504	.0401968	19.6900	.0008909
8.0588	.0000222	11.9984	.0091191	15.9380	.0380800	19.8776	.0006533
8.2464	.0000250	12.1860	.0110462	16.1256	.0355328	20.0652	.0004603
8.4340	.0000292	12.3736	.0132322	16.3132	.0328365	20.2528	.0003313
8.6216	.0000421	12.5612	.0156393	16.5008	.0299006	20.4404	.0002279
8.8092	.0000553	12.7488	.0182532	16.6884	.0269509	20.6280	.0001530
8.9968	.0000799	12.9364	.0210393	16.8760	.0239486	20.8156	.0001105
9.1844	.0001105	13.1240	.0239486	17.0636	.0210393	21.0032	.0000799
9.3720	.0001530	13.3116	.0269509	17.2512	.0182532	21.1908	.0000553
9.5596	.0002279	13.4992	.0299006	17.4388	.0156393	21.3784	.0000421
9.7472	.0003313	13.6868	.0328365	17.6264	.0132322	21.5660	.0000292
9.9348	.0004603	13.8744	.0355328	17.8140	.0110462	21.7536	.0000250
10.1224	.0006533	14.0620	.0380800	18.0016	.0091191	21.9412	.0000222
10.3100	.0008909	14.2496	.0401968	18.1892	.0074355	22.1288	.0000203
10.4976	.0012275	14.4372	.0419026	18.3768	.0059706	22.3164	.0000188
10.6852	.0016376	14.6248	.0432850	18.5644	.0046506	22.5040	.0000181
10.8728	.0022921	14.8124	.0441855	18.7520	.0033407	22.6916	.0000178
11.0604	.0029279	15.0000	.0444189	18.9396	.0029279		

Example 10.2:

Consider again network configuration of Figure 5.9 of Chapter 5. Following results as shown in Table 10.11 are taken from Merier, Newell, and Pazer (1969) for network of Figure 5.9 using Monte Carlo simulation.

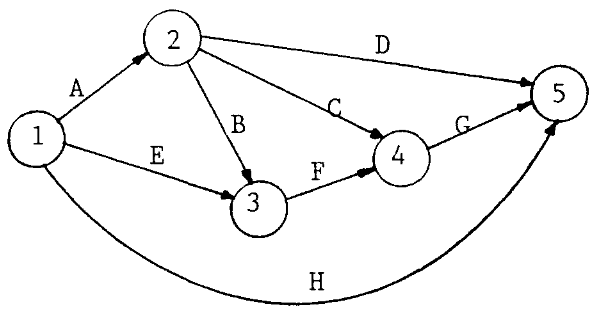


Figure 5.9

Table 10.11: COMPARISON OF PERT, ANALYTICAL, AND SIMULATION RESUTLS

Method of computation	Project Completion Time μ	σ
PERT	8.00	1.26
Analitically	9.23	1.39
Simulation		
40 Samples	9.67	1.27
50 Samples	9.12	1.55
90 Samples	9.36	1.41

Table 10.12 shows percent error for mean and standard deviation of the project completion time using Monte Carlo simulation.

Table 10.12:

Sample size	Percent error: Mean (Simulation from Exact)	Percent error: Standard Deviation (Simulation from Exact)
40	+4.77	- 8.63
50	-1.19	+11.51
90	+1.41	+ 1.44

Recall that the proposed procedure with less computation effort provides exact mean and standard deviation.

Example 10.3:

Consider again the network configuration of Figure 4.14 of Chapter 4. As mentioned in Chapter 4 this example is taken from Van Slyke (1963). The three numbers assigned to each activity are respectively the optimisitc, most likely, and pessimistic estimates for each activity.

The expected project completion time using PERT is 66.0 with a variance of 60.27.

The Monte Carlo method proposed and tried by Van Slyke (1963) using 10,000 realizations yields a mean of 67.0 and a variance of 42.39.

The distributions used for the activity durations were the beta distributions with end points and modes given by the three parameters indicated in Figure 10.3. The standard deviation was taken to be 1/6 the range.

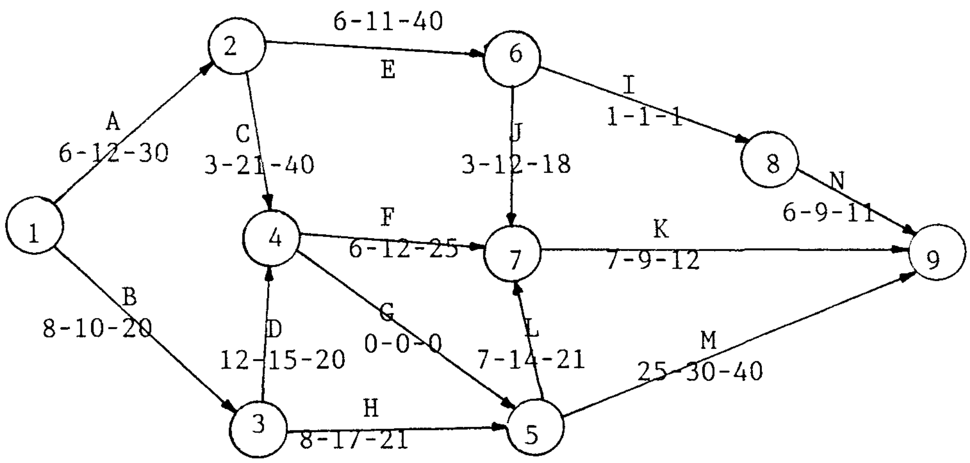


Figure 10.3

In Figure 10.4 normal distribution with mean 66.0 and variance 60.27 is compared with the distribution obtained by Monte Carlo.

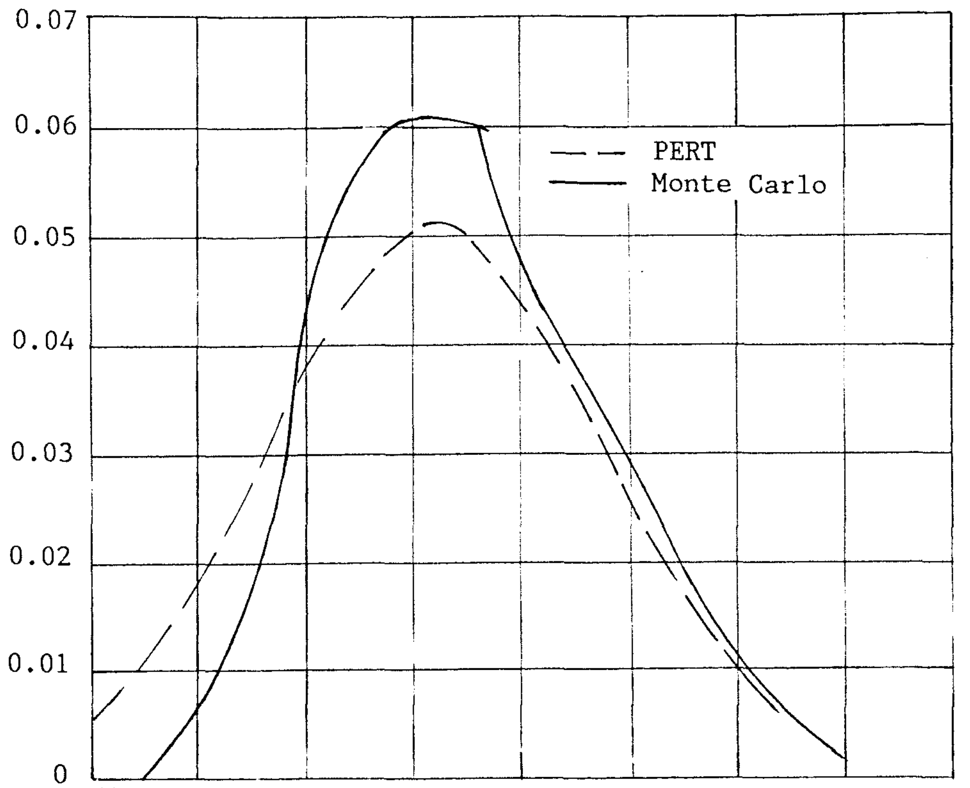


Figure 10.4: Probability density curves

Figure 10.5 displays the network with the criticality index for each activity. The heavy line is "the critical path" calculated using expected values for the activity durations.

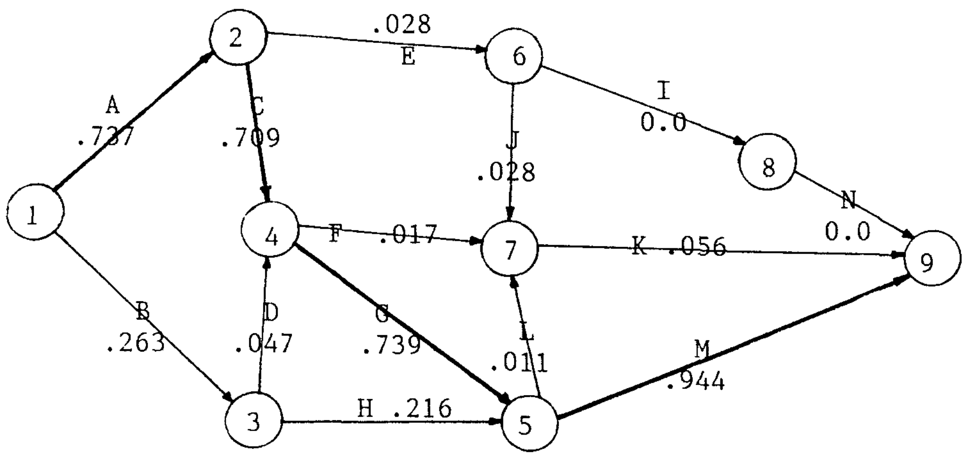


Figure 10.5

The proposed procedure of this dissertation is applied to the network configuration of Figure 10.6 which is the network of Figure 10.3 with reversed direction of arrows in order to minimize calculation efforts.

It is assumed that the duration times of activities are discretely distributed as shown in Table 10.13.

The mean and variance for the project completion time using proposed procedure were found to be 69.95 and 116.75 respectively.

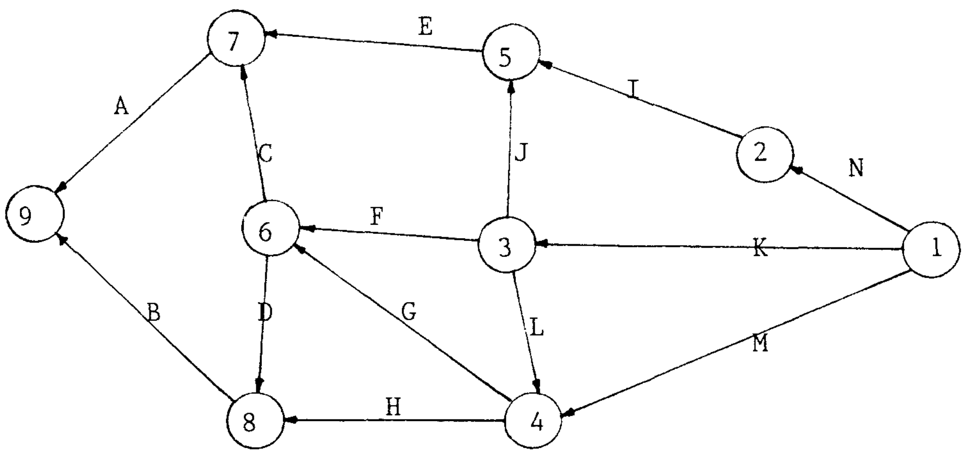


Figure 10.6

Table 10.13: Duration times of activities of Figure 10.3.

A	P	B	P	C	P	D	P	E	P	F	P
6	1/6	8	1/6	3	1/6	12	1/6	6	1/6	6	1/6
12	4/6	10	4/6	21	4/6	15	4/6	11	4/6	12	4/6
30	1/6	20	1/6	40	1/6	20	1/6	40	1/6	25	1/6
μ = 14 σ = 7.503		μ = 11.333 σ = 3.944		μ = 21.167 σ = 10.68		μ = 15.333 σ = 2.357		μ = 15 σ = 11.328		μ = 13.167 σ = 5.728	

G	P	H	P	I	P	J	P	K	P	L	P
0	1/6	8	1/6	1	1/6	3	1/6	7	1/6	7	1/6
0	4/6	17	4/6	1	4/6	12	4/6	9	4/6	14	4/6
0	1/6	21	1/6	1	1/6	18	1/6	12	1/6	21	1/6
μ = 0 σ = 0		μ = 16.167 σ = 3.933		μ = 1 σ = 0		μ = 11.5 σ = 4.387		μ = 9.167 σ = 1.462		μ = 14 σ = 4.041	

M	P	N	P
25	1/6	6	1/6
30	4/6	9	4/6
40	1/6	11	1/6
μ = 30.833 σ = 4.490		μ = 8.833 σ = 1.464	

The normalized values of criticality indices of all activities are shown beside activities of Figure 10.7

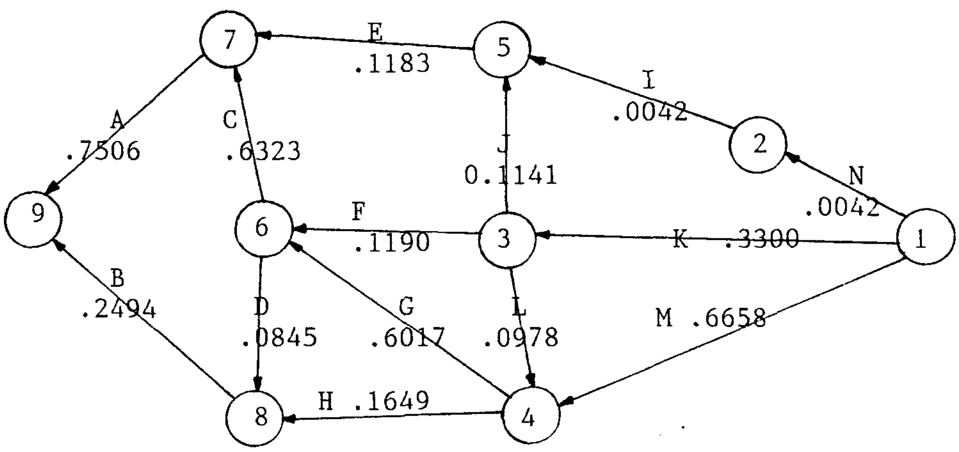


Figure 10.7

The mean and variance for the project completion time of Figure 10.6 using Dodin's (1980) procedure were found to be 71.5 and 113.91 respectively. Notice that in here again it is assumed that the duration times of activities are discretely distributed as shown in Table 10.13.

The normalized values of criticality indices of all activities of Figure 10.6 using Dodin and Elmaghraby's (1985) approach were found as shown beside activities of Figure 10.8.

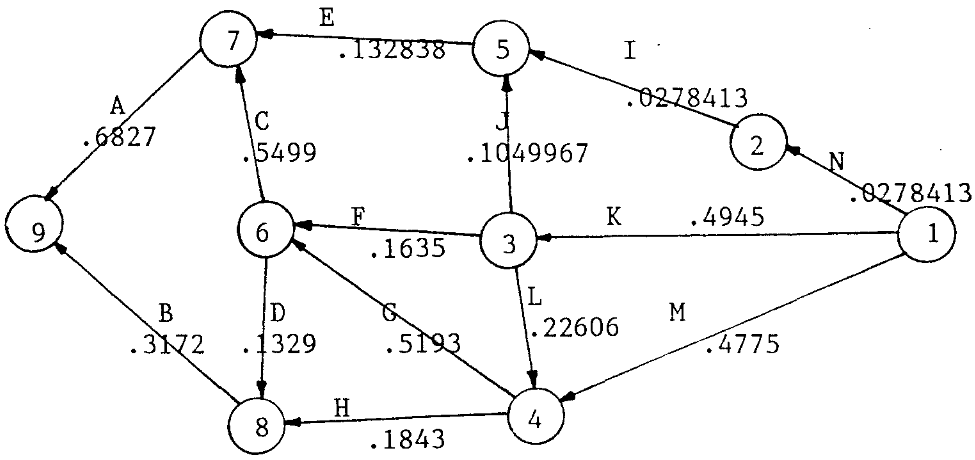


Figure 10.8

The discrepancy between the values of mean, variance, and criticality indices obtained using the Monte Carlo simulation and those obtained using proposed prosedure is due to following factors:

1 - In the Monte Carlo method the distributions used for the activity durations were the beta distributions with end points and modes given by the three parameters indicated in Figure 10.3. The standard deviation was taken to be 1/6 the range. In the proposed procedure the duration times of activities are assumed to have discrete distributions (i.e. each activity has three duration times as shown in Figure 10.3 with probability of occurrence 1/6, 4/6, and 1/6 respectively).

Notice that the standard deviation of activities in case of discrete distribution as shown in Table 10.13 is much greater than 1/6 the range as it is used in Monte Carlo method. For example the standard deviation of duration time of activity A where A is discretely distributed as shown in Table 10.13 is approximately 7.503, while 1/6 of its range is 4.

2 - The path deletion method as described in Chapter 2 is applied by Van Slyke (1963) to the network of Figure 10.3 to approximate the mean, variance and the pdf of project completion time and criticality indices of activities as shown in Figure 10.5. As we can see in Figure 10.5 the criticality indices of activities I and N are zero. As noted in Van Slyke (1963) activities I and N cannot be critical, but the Min-Max method as described in Chapter 2 and the proposed procedure of this dissertation and also Dodin and Elmaghraby's (1985) procedure don't reveal this.

3 - In order to minimize the calculation effort the proposed procedure is applied to the network with arrow directions reversed. As mentioned previously this does not alter the exact pdf of the project completion time but may change the approximate criticality indices of activities.

Let X denote the approximate criticality indices obtained using Monte Carlo method, Y denote the approximate criticality indices obtained using proposed procedure, and Z denote the approximate criticality indices obtained using Dodin and Elmagraby's procedure. The correlation coefficient between X and Y, and also between X and Z were found to be,

$$r_{X,Y} = \frac{\Sigma xy}{\sqrt{(\Sigma x^2)(\Sigma y^2)}} = \frac{1.1404851}{\sqrt{(1.5801527)(.9177149)}} = 0.947079,$$

and

$$r_{X,Y} = \frac{\Sigma xz}{\sqrt{(\Sigma x^2)(\Sigma z^2)}} = \frac{0.7999119}{\sqrt{(1.5801528)(.6043546)}} = 0.8185528.$$

As we can see, considering the above factors, the approximate criticality indices of activities obtained using proposed procedure are more accurate than those obtained using Dodin and Elmaghraby's (1985) procedure.

SUMMARY AND CONCLUSIONS

In this chapter we briefly described Monte Carlo methods for solving stochastic PERT networks. In crude Monte Carlo simulation we apply the longest path algorithm to a long series of realizations each one obtained by assigning a sample value to every activity drawn from its proper distribution, then, given this information, we use standard statistical methods to estimate the distribution and parameters of interest.

In the conventional PERT solution method, much of the output does not depend on the structure of the activity duration distributions but only on their means and variances. The Monte Carlo approach, in order to gain extra accuracy, does depend on the shape of the distribution. On the other hand, the Monte Carlo approach has greater flexibility in that any distribution can be used for activity durations: beta, normal, triangular, uniform, or discrete in any sort of mix.

As mentioned in previous chapters the estimate for the mean using conventional PERT solution method is always low, and the estimate for the variance may be biased in either direction, whereas the Monte Carlo procedure gives an unbiased estimate of mean and variance. Moreover, the Monte Carlo procedure is very accurate in making estimate of the probability of meeting schedule dates and also yields information on the "criticality indices" of activities.

The key drawback of the crude Monte Carlo method is that it is very computationally costly for reasonable levels of precision with respect to the resulting distribution.

Comparison of the results from experiments presented in this chapter suggests that the CIM approach with 40 cells and optimal squeezing gives comparable precision to the Monte Carlo method usign about 10,000 samples as recommended by Van Slyke (1963) with respect to 1 percentile values, much greater precision with respect to mean and variance values. Furthermore, very high levels of precision are possible if the CIM approach is used with 100 cells or more.

In term of computation speed, it is clear from the experiments of this chapter that Monte Carlo methods provide moderately good results with sample size as small as 300. The CIM approach can only achieve the same with about 20 cells or less which provides unacceptable crudeness in the result distribution shape.

Considering the limitations of CIM approach for retaining a memory of common activities in calculation procedure, the CIM approach would seem preferable if

a) not too many common activities are involved, or the activity network can be separated into several subnetworks.

and

b) reasonable or high precision is required.

Monte Carlo method would seem preferable if

a) complex dependence structure are involved,

or

b) not too much precision is required.

The proposed procedure of this dissertation minimizes the memory limitations of CIM approach, adding to the efficiency of this approach. However, Monte Carlo simulation may maintain a comparative advantage for very complex networks, and the ideal approach to networks in general may be a hybrid.

CHAPTER 11: SUMMARY AND CONCLUSION

PERT has proved a useful and broadly applied management tool. In its original form, it is assumed duration times for individual activities are fixed and known in advance. This assumption led to the conventional PERT solution method for determining the critical path and project completion time.

In practice it has been shown that this approach leads to optimistic bias for the mean value of the project completion time. Consequently, attempts were made to introduce probabilistic activity completion times, allowing representation of the stochastic nature of most projects.

In stochastic PERT networks the main difficulty in calculating the pdf of the project completion time is caused by dependency between the paths. This dependency makes it also very hard to identify the most critical paths and activities in most stochastic networks.

Ignoring the dependency between paths, Dodin (1980) proposed a procedure to approximate the pdf of project completion time.

In Chapter 5 it was demonstrated that Dodin's (1980) procedure always leads to a pesimistically biased estimate of the occurrence time of events in irreducible networks. On the other hand both PERT and Dodin's calculated standard deviation may be biased in either direction. Precise statement about the magnitude of the error cannot be made since errors in project mean and variance vary with different network configurations.

The two more important factors affecting the magnitude of the errors in merge events are as follows:

1 - The number of subcritical paths leading to a merge event.

2 - The variance of the subpaths lengths.

Considering the dependency between paths two main alternative procedures have been suggested.

1 - The Controlled Interval and Memory (CIM) approach.

2 - Monte Carlo Simulation.

"CIM", developed by Chapman and Cooper (1983a), considers all paths if the memory concept is employed on the reducible networks. This method is based on defining each distribution of activities in histogram form with equal or common cell divisions within each distribution. The addition of two such distributions produces a result distribution which also has equal cell intervals. This method allows specification effort, specification error, computation effort and computation error to be controlled in an integrated manner.

"Monte Carlo" simulation suggested by Van Slyke (1963) also considers all paths for complex networks. This method uses activity durations randomly drawn from appropriate probability distributions. Van Slyke (1963) suggested that the method can be used in two different ways. The first involves a relatively small sample size and is used to check on the traditional methods such as CPM and PERT. The second is to take a large sample size of activities and simulate the network to obtain the answers. Monte Carlo simulation provides an unbiased estimate of the mean and standard deviation of project completion time and approximates the criticality indices of activities and paths. One drawback of crude Monte Carlo simulation is that it is computationally very costly.

The proposed procedure of this dissertation is based on the CIM approach to determine the mean, variance and probability distribution function (pdf) of project completion time and approximate the criticality index of each activity and path. It allows activity durations to have continuous or any distribution function presented in a finite set of ordered pair. It also allows statistical dependence between activities.

The proposed procedure provides an exact pdf for project completion time when the duration times of activities are discrete and approximates the pdf of the project completion time when the duration times of activities are continuous.
Approximation is due to

1 - Discretizing continuous distributions.

2 - Convoluting discrete approximations to continuous distribution.

In Chapter 6 it was demonstrated that approximate criticality indices of activities obtained using the proposed procedure are more accurate than those obtained using an approximating procedure proposed by Dodin and Elmaghraby (1985). If common activity times have relatively large variances or the number of activities emanating from merge events are more than two, approximate criticality indices obtained using Dodin and Elmaghraby's (1985) procedure are less accurate, while these factors don't affect the accuracy of criticality indices obtained using the proposed procedure. Moreover, the proposed procedure and numeration methods, both give the same ranking of criticality indices in most PERT networks.

Chapter 7 demonstrated how structural dependence relationships can interact with statistical dependence relationships to produce important effects which cannot be detected and understood using simple expected value calculations, but they can be identified, modelled and understood in a CIM framework.

Chapter 8 provided a brief discussion of different discretizing methods for continuous distributions. The accuracy of the most efficient method was examined through simple examples. It was demonstrated that: the accuracy of the approximation can be enhanced by using more cells; the absolute values of the deviations between the exact and approximate pdf (MDV) and the average of the absolute values of the deviations between the exact and approximate pdf (ADV) were minimized in the case of a series of activities with normal distributions.
In Chapter 9 the accuracy of the approximate pdfs for the project completion times obtained using CIM and DPD approaches were compared for the three examples of Chapter 8 by measuring their closeness to the "true" pdfs. It was demonstrated that the CIM approach provides more accurate results for combining continuous distributions of activities. This advantage was especially apparent in example 4 where the distribution functions of activities were extremely skewed.

Finally, comparison of the results from experiments presented in Chapter 10 suggests that the CIM approach with 40 cells and optimal squeezing gives comparable precision to the Monte Carlo method using about 10,000 samples. The CIM approach does have only one important limitation relative to Monte Carlo simulation which need to be noted, otherwise a CIM approach provides much greater precision for computation a moderate effort. The retention of a single memory dimension for the common activity A of Figure 11.1 involved preserving and working with a matrix of probabilities at each stage. In general, n levels of memory involved an (n+1)-dimensional probability matrix. Large values of n make this approach computationally demanding. Computer software which will handle largish values of n is feasible, but the size of value of n at which Monte Carlo simulation would be a preferable computational approach remains a research issue.

The proposed procedure of this dissertation minimizes the computational implications of the memory aspect of the CIM approach by solving the network for various condition values of common activities. This adds to the efficiency of the CIM approach. When modelling activites at a source of risk/response level of detail, it is usual to employ only 20 or so activities to represent a very large project. This means that the proposed procedure is usually viable for detailed risk analysis, although it could not be used exclusively for a large basic PERT network.

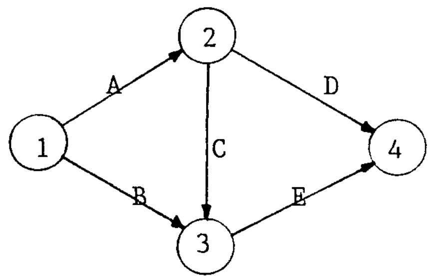


Figure 11.1

APPENDIX A:

Theorem : An activity network is not completely reducible if and only if it contains the interdictive graph.

The following notation will be used in the proof of the theorem.

(i,j) = arc $(i,j)\in A$ starting at node i and ending in node j where $i<j$, it is the only arc connecting node i to node j in the network.

$A(i) = \{j\in N: (i,j)\in A\}$, the set of nodes succeeding node i.

$B(i) = \{k\in N: (k,i)\in A\}$, the set of nodes preceding node i.

$P(i) = \{k\in N: k < i$ and k connects to i by a path$\}$, i.e. the set of nodes that precede node i and connects to i by an arc or a path.

The interdictive graph (IG) is the graph shown in Figure A.1; evidently, it is irreducible and shares with all irreducible networks the following properties.

(1) The number of nodes $N \geqslant 4$ and of arcs $A \geqslant 5$.

(2) For each $i\neq 1,N$, $|A(i)|+|B(i)| \geqslant 3$; hence, there are no arcs in series.

(3) Either $|A(1)|=1$, in which case $|B(2)|=|A(1)|=1$ and $|A(2)|\geqslant 2$; or $|A(1)|\geqslant 2$. Therefore, without loss of generality, we can assume that in an irreducible network $|A(1)|\geqslant 2$.

(4) Either $|B(N)|=1$, in which case $|A(N-1)| = |B(N)| = 1$ and $|B(N-1)| \geqslant 2$; or $|B(N)|\geqslant 2$. Similarly we can assume that $|B(N)|\geqslant 2$.

(5) There exists a smallest numbered node $j\neq 1,2$ or N and a node $i\in P(j)$, such that there are two independent paths (no arcs in common) connecting i to j.

<u>Proof</u>: The 'if' part is true since the IG is irreducible; hence, any graph containing it is also irreducible. The proof of the 'only if' part is by contradiction.

Assume the network to be irreducible; we wish to show that there exists at least one IG (in which the link (1,2), (1,3), etc., of Figure A.1 may themselves be paths). Since the rank (degree) of each node $i \neq 1$, N is at least 3, by Property 2, there must exist at least one node i_3 with $|B(i_3)| \geq 2$. Then by Property 5, there must exist at least one node $i_1 \epsilon P(i_3)$ such that there are two independent paths (have no arcs in common) connecting i_1 to i_3. One of the paths must have at least one intermediate node i_2 with $|A(i_2)| \geq 2$ since we started with a network which does not contain two arcs in parallel. In fact, there exist a node $i_2 < i_3$ which can be considered the mirror image of i_3 in the sence that there are two independent paths connecting node i_2 to a node $i_4 \geq i_3$ with one of the two paths passing through node i_3; see Figure A.2 for illustration. Therefore, i_2 is connected to i_4 with two independent paths; one of them passes through node i_3.

The composition of the two paths connecting i_1 to i_3 and the two paths connecting i_2 to i_4 gives an interdictive graph. The absence of the 'cross-over' between nodes i_2 and i_3 imply that node i_1 is connected to node i_4 with two independent paths; hence, the digraph is composed of graphs in parallel, each one of which is itself composed of arcs in series (recall that we started with a digraph that contains no arcs in series). But such a diagraph is reducible, a contradiction. The paths $(i_1, \ldots, i_2)$, $(i_1, \ldots, i_3)$, $(i_2, \ldots, i_3)$, $(i_2, \ldots, i_4)$, and $(i_3, \ldots, i_4)$ constitiute the desired IG as illustrated in Figure A2.

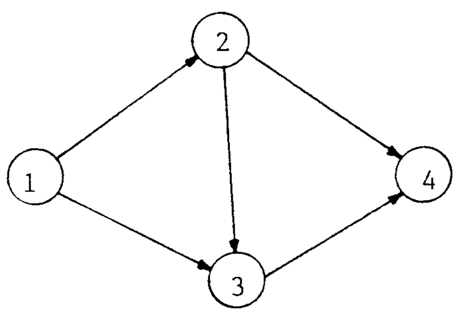


Figure A.1: The interdictive graph.

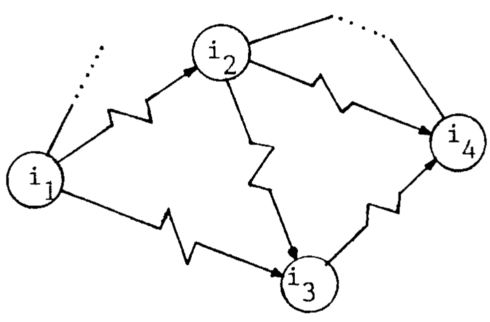


Figure A.2: An irreducible activity network.

APPENDIX B:

Suppose that X has distribution $N(\mu,\sigma^2)$. Hence the pdf of X is

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma}\exp[-\tfrac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2] .$$

If $(X_1,\ldots,X_n)$ is a sample of X, its likelihood function is given by

$$L(X_1,\ldots,X_n;\mu,\sigma) = (2\pi\sigma^2)^{-n/2}\exp\{-\tfrac{1}{2}\sum_{i=1}^{n}[(X_i-\mu)/\sigma]^2\} .$$

Hence

$$\text{In}(L) = (-\frac{n}{2})\text{In}(2\pi\sigma^2) - \tfrac{1}{2}\sum_{i=1}^{n}[(X_i-\mu)/\sigma]^2 .$$

We must solve simultaneously

$$\frac{\partial \text{In}L}{\partial\mu} = 0 \qquad \text{and} \qquad \frac{\partial \text{In}L}{\partial\sigma} = 0 .$$

We have

$$\frac{\partial \text{In}L}{\partial\mu} = \sum_{i=1}^{n}[(X_i-\mu)/\sigma^2] = 0 ,$$

which yields $\hat{\mu} = \bar{X}$, the sample mean. And

$$\frac{\partial \text{In}L}{\partial\sigma} = -\frac{n}{\sigma} + \sum_{i=1}^{n}[(X_i-\mu)^2/\sigma^3] = 0 ,$$

which yields

$$\hat{\sigma}^2 = \frac{1}{n}\sum_{i=1}^{n}(X_i-\mu)^2 = \frac{1}{n}\sum_{i=1}^{n}(X_i-\bar{X})^2 .$$

Note that the ML method yields a biased estimate of σ^2, since we have already seen that the unbiased estimate is of the form

$$\frac{1}{(n-1)}\sum_{i=1}^{n}(X_i-\bar{X})^2 \ .$$

APPENDIX C:

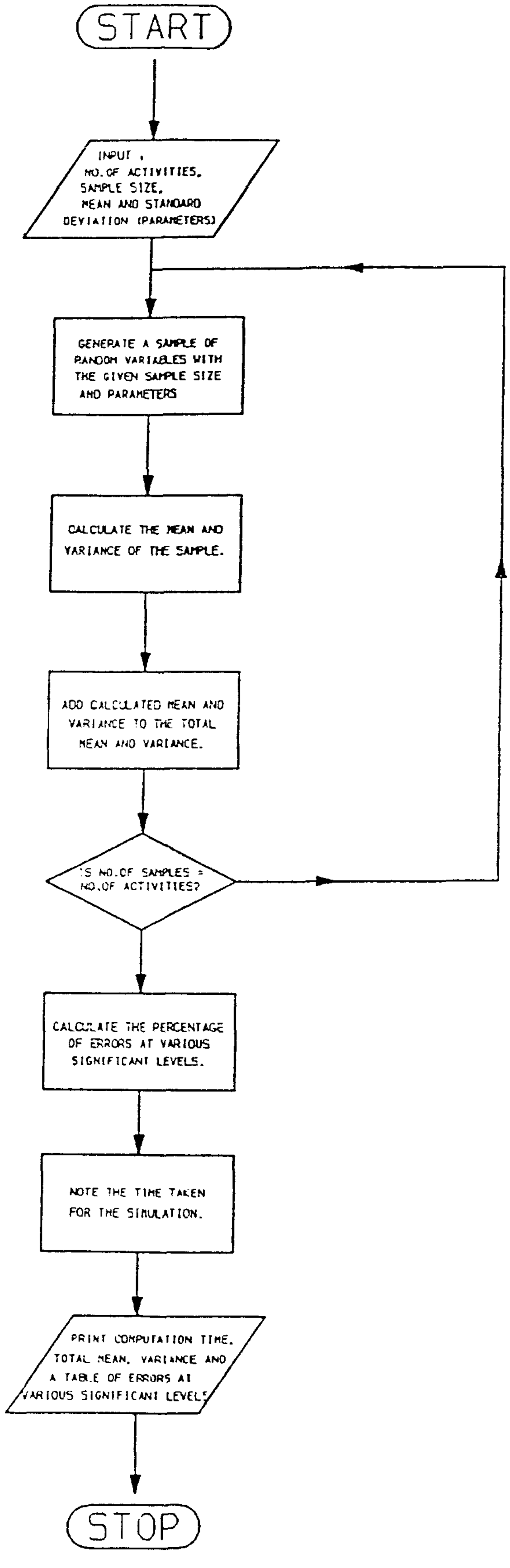
```
      EXTERNAL RAND
      REAL ERROR(5),EOP(5),SLEVEL(5)
      REAL EXMEAN(200),EXVAR(200),XBAR(200),XVAR(200)
      REAL MEAN,STAND,X(3000)
      INTEGER IHOUR,IMINUTE,ISECOND,IHUNDREDSECOND,ISEED
      LOGICAL*2 XT,SETTIM
      DATA SLEVEL /10.0,5.0,1.0,0.5,0.1/
      ISEED = 1234
      XT = SETTIM(0,0,0,0)
C
C     INPUT THE NUMBER OF ACTIVITIES.
C
      READ(3,*) M
      DO 44 IP = 1,M
C
C     INPUT SAMPLE SIZE, MEAN AND STANDARD DEVIATION.
C
         READ(3,*) N,MEAN,STAND
C
C     CALL SUBROUTINE TO GENERATE NORMAL RANDOM
C     VARIABLES WITH THE SPECIFIED PARAMTERS ABOVE.
C
         CALL NORSAMP(X,N,MEAN,STAND,ISEED)
C
```

```
C   CALL SUBROUTINE TO CALCULATE THE EXPECTED MEAN AND
C   VARIANCE FROM THE SAMPLE.
C
       CALL BARVAR(X,N,XVAR,IP)
       TXBAR = TXBAR + XBAR(IP)
       TXVAR = TXVAR + XVAR(IP)
       EXMEAN(IP) = MEAN
       EXVAR(IP) = STAND*2
       TMEAN = TMEAN + EXMEAN(IP)
       TVAR = TVAR + EXVAR(IP)
44     TN = TN + N
C
C   NOTE THE TIME TAKEN FOR THE COMPUTATION.
C
       CALL GETTIM(IHOUR,IMINUTE,ISECOND,IHUNDREDSECOND)
C
C   CALL SUBROUTINE TO COMPUTE THE ERRORS AT 10%, 5%, 1%,
C   0.5%, 0.1% SIGNIFICANT LEVELS
C
       CALL ERRSIG(TMEAN,TXBAR,TXVAR,ERROR,EOP,TN)
    WRITE(*,110)
    WRITE(*,120) IHOUR,IMINUTE,ISECOND,IHUNDREDSECOND
    WRITE(*,100)
    WRITE(*,130)
    WRITE(*,140) TMEAN
    WRITE(*,150) TVAR
```

```
      WRITE(*,145) TXBAR
      WRITE(*,150) TXVAR
      WRITE(*,100)
      WRITE(*,147) INT(TN)
      WRITE(*,100)
      WRITE(*,160)
      WRITE(*,170)
      WRITE(*,180)
      WRITE(*,170)
      WRITE(*,160)
      WRITE(*,170)
      DO 55 K = 1,5
         WRITE(*,190) SLEVEL(K),ERROR(K),EOP(K)
55       WRITE(170)
      WRITE(*,160)
100   FORMAT(1X)
110   FORMAT(1X,'COMPUTATION TIME:')
120   FORMAT(1X,I2,'hr',I2,'min',I2,'sec',I2,'hundredsec')
130   FORMAT(1X,'CENTRAL LIMIT THEOREM:')
140   FORMAT(1X,'EXACT:    MEAN    =',F15.2)
145   FORMAT(1X,'MONTE CARLO:  MEAN    =',F15.2)
147   FORMAT(1X,'SAMPLE SIZE = ',I15)
150   FORMAT(1X,'VARIANCE = ',F15.2)
160   FORMAT(6X,39('-'))
170   FORMAT(6X,' ',11X,' ',2(12X,' '))
180   FORMAT(6X,' SIG LEVEL %    ERROR        ERROR %(+-) ')
```

```
190 FORMAT(6X,' ',3X,F5.2,3X,' ',1X,F10.4,'  ',1X
   1,F8.2,'     ')
    END
    FUNCTION RAND(IX)
    INTEGER A,P,IX,B15,B16,XHI,XALO,LEFTO,FHI,K
    DATA A/16807/,B15/32768/,B16/65536/
   1,P/2147483647/
       XHI = IX/B16
       XALO = (IX-XHI*B16)*A
       LEFTO = XALO/B16
       FHI = XHI*A+LEFTO
       K = FHI/B15
       IX = ((XALO-LEFTO*B16)-P+(FHI-K*B15)*B16)+K
       IF (IX .LT. 0) IX = IX+P
       RAND = FLOAT(IX)*4.656612875E-10
    RETURN
    END
    SUBROUTINE NORSAMP(X,N,MEAN,STAND,ISEED)
    REAL X(N),MEAN,STAND
    INTEGER ISEED,N
    DO 11 I = 1,N
22     U1 = RAND(ISEED)
       U2 = RAND(ISEED)
       V1 = 2*U1-1
       V2 = 2*U2-1
       W = (V1**2)+(V2**2)
```

```
      IF ( W .GT. 1.0 ) GOTO 22
      Y = SQRT((-2*ALOG(W))/W)
11    X(I) = MEAN + (V1*Y)*STAND
   END
   SUBROUTINE BARVAR(X,N,XBAR,XVAR,IP)
   REAL X(N),XBAR(200),XVAR(200)
   SX1 = 0.0
   SX2 = 0.0
   DO 66 I = 1,N
      SX1 = SX1 + X(I)
66    SX2 = SX2 + (X(I)**2)
   AN = N
   XBAR(IP) = SX1/AN
   XVAR(IP) = (SX2-(SX1*SX1)/AN)/(AN-1)
   END
   SUBROUTINE ERRSIG(TMEAN,TXBAR,TXVAR,ERROR,EOP,TN)
   REAL TMEAN,TXBAR,TXSTAND,TXVAR,ERROR(5),EOP(5),TN,Z(5)
   DATA Z/1.6449,1.9600,2.5758,2.8130,3.2905/
   TXSTAND = SQRT(TXVAR)
   DO 33 I = 1,5
      ERROR(I) = Z(I)*(TXSTAND/SQRT(TN))
      IF ( TMEAN .GT. TXBAR ) ERROR(I) = -1*(ERROR(I))
33    EOP(I) = ABS((ERROR(I)*100)/TMEAN
   END
```


START
INPUT ,
NO.OF ACTIVITIES,
SAMPLE SIZE,
MEAN AND STANDARD
DEVIATION (PARAMETERS)
GENERATE A SAMPLE OF RANDOM VARIABLES WITH THE GIVEN SAMPLE SIZE AND PARAMETERS
CALCULATE THE MEAN AND VARIANCE OF THE SAMPLE.
ADD CALCULATED MEAN AND VARIANCE TO THE TOTAL MEAN AND VARIANCE.
IS NO.OF SAMPLES = NO.OF ACTIVITIES?
CALCULATE THE PERCENTAGE OF ERRORS AT VARIOUS SIGNIFICANT LEVELS.
NOTE THE TIME TAKEN FOR THE SIMULATION.
PRINT COMPUTATION TIME, TOTAL MEAN, VARIANCE AND A TABLE OF ERRORS AT VARIOUS SIGNIFICANT LEVELS
STOP

APPENDIX D:

Notations and Definitions

The following is a list of notations and difinitions used throughout this study.

A: the set of activities (arcs) in the network G(N,A); also the cardinality of the set.

ACAP(ij): the approximate criticality index of activity (ij) using proposed procedure of Dodin and Elmaghraby (1985).

A(i): set of arcs emanating from node i= set of nodes succeeding node i.

ADV: average value of the absolute deviation of F(N) from F'(N).

a: denotes an arc, and is indexed from 1 to A.

a: optimistic activiity duration.

B(i): set of arcs incident into node i= set of nodes preceding node i.

B_j: set of arcs immediately preceding node j.

b: pessimistic activity duration.

C_j: the cutset at node j.

CA(ij): the exact criticality index of activity (ij).

CAP(ij): the approximate criticality index of activity (ij) using proposed procedure of this dissertation.

CAP(ij): the probability that the maximum of the paths in L(ij) is longer than the maximum of the paths in the complement set $\bar{L}(i,j)$. It is a lower bound on the value of CA(ij).

cdf: comulative distribution function.

C.I. criticality index.

CIM. controlled interval and memory.

CN(i): the criticality index of node i= the sum of criticaltiy indices of the paths containing node i.

CNP(i): the approximate criticality index of node i using proposed procedure of this dissertation.

CP: cumulative probability.

$CP(T_h)$:	the criticality path index of path T_h = the probability that path T_h is not shorter than any other path.
$CPP(T_h)$:	the approximate criticality index of path T_h using proposed procedure of this dissertation.
DAN:	deterministic activity network.
DC(j):	directed cutset = set of arcs connecting the nodes with numbers less than j to the nodes with numbers greater than or equal to j.
dF(x):	density function of the random variable X,i.e., dF(x) = f(x).
df:	distribution function.
d_j:	Dodin's estimate of the realization time for node j.
E:	mean value.
E_j:	earliest occurrence time for event j.
e_j:	expected value of critical path to node j given any realization of subnetwork P_j, j=2,3,...,N.
EF(ij):	earliest finish time for activity (ij).
ES(ij):	earliest start time for activity (ij).
$E[Z(T_1^r)]$:	the mathematical expectation of r.v. $Z(T_1^r)$.
F(N):	approximate pdf of the project completion time.
F_{ij}:	float time for activity (ij).
$F_{ij}(Y_{ij})$:	probability distribition function of activity $(ij) \epsilon A$.
F'(N):	true pdf of the project completion time.
f_j:	Fulkerson's estimate of the realization time for node j.
G(N,A):	an activity network with N nodes and A arcs.
$\overleftarrow{G}(N,A)$:	an activity network with reversed direction of arrows.
$g_1(.)$:	coefficient of skewness = moment coefficient of skewness.
g_j:	PERT estimate of the realization time for node j.
I(i):	indegree of node i = number of arcs ending at node i.
i:	denotes a node, and indexed from 1 to N.
L(ij):	the set of paths containing arc (ij). Its complement is $\bar{L}(ij)=P-L(ij)$.

L_j:	latest occurrence time for event j.
LF(ij):	latest finish time for activity (ij).
LS(ij):	latest start time for activity (ij).
MCS:	Monte Carlo Sampling.
MDV:	maximum value of the absolute deviation of F(N) from F'(N).
m:	most likely activity duration
m_1':	first moment about the origin = mean.
m_2':	second moment about the origin.
m_3':	third moment about the origin.
m_2:	second moment about the mean = variance
m_3:	third moment about the mean.
$m_T(t)$:	moment generating function.
NACAP(ij):	the normalized value of approximate criticality index of activity (ij) using proposed procedure of Dodin and Elmagraby (1985).
NCA(ij):	the normalized value of exact criticality index of activity(ij).
NCAP(ij):	the normalized value of aproximate criticality index of activity (ij) using proposed procedure of this dissertation.
NCPP(T_h):	the normalized value of approximate criticality index of path T_h using proposed procedure of this dissertation.
NIN:	number of the realizations in F'(N).
NRR:	number of the realizations in F(N).
n_j:	the in-degree of node j.
O(i):	outdegree of node i = number of arcs emanating from node i.
P:	the set of paths from node 1 to node N; also the cardinality of the set.
PAN:	probability activity network.
pdf:	probability distribution function.
P_j:	subnetwork of nodes and arcs up to and including node j.
$P_X(x)=P(X=x)$:	the probability mass of random variable X and in short is denoted by p.
$p(y_{ij})$:	marginal probability function of arc (ij).

$p[y(A)]$:	probability of the realization of the arcs in the network.
r.v.:	random variable.
r:	correlation coefficient
SA:	sequential approximation.
S_i:	slack time for event i.
T_c:	critical path.
T_h:	the hth path; $T_h \epsilon P$.
T_i:	realization time of node i, i=2,3,...,N.
$T_N(A)$:	critical path given any realization of the network.
T_a^k:	kth realization time of activity a.
T_i^f:	the duration of the longest path "forward" from node 1 to node i.
T_j^b:	the duration of the longest path "backwards" from node N to node j.
V_{ij}:	the duration of the longest path not contaiing arc (ij).
W_{ij}:	the duration of the longest path containing arc ij.
W_{1j}:	the duration of the paths ending in node j not containing arc (ij).
Y_{ij}:	the duration of activity (ij), a r.v.
$y(A)$:	vector of realization of all arcs in the network.
$y(B_j)$:	vector of realization of all arcs in the set B_j.
$Z(T_h)$:	the duration of path $T_h \epsilon P$.
$\hat{\theta}$:	maximum likelihood estimate of θ.
*:	a convolution operation.
.:	a greatest operation.

$\delta(.)$ for arc = $\gamma(.)$ for node $\begin{cases} 0 \text{ if the argument variable is "inactive"} \\ 1 \text{ if the argument variable is "active".} \end{cases}$

where an "active" node or arc is one that is retained in the final (irreducible) network.

REFERENCES

1 Anderson, D.R., D.J. Sweeney and T.A.Williams (1978) Essentials of Management Science: Application to Decision Making. West Publishing Co., St. Paul, Minnesota.

2 Atkinson, A.C., and M.C. Pearce (1976) The Computer Generation of Beta, Gamma and Normal Random Variables. Journal of the Royal Statistical Society A139, 431 - 448.

3 Box, G.E.P, and M.E. Muller (1958) A Note on the Generation of Random Normal Deviates. Annual of Mathematical Statistics 29, 610-611.

4 Britney, R.R. (1976) Baysian point estimation and the PERT scheduling of stochastic activities. Management Sci. 22, 938-948.

5 Burt, J.M. (1977) Planning and dynamic control of projects under uncertainty. Management Sci. 24, 249-258.

6 Burt, J.M, M.B. Garman (1971a) Conditional Monte Carlo: a simulation technique for stochastic analysis. Management Sci. 18(3), 207-217.

7 Burt, J.M., M.B. Garman (1971b) Monte Carlo techniques for stochastic PERT network analysis. Infor. 9, 248-262.

8 Burt, J.M., D.P. Gaver, and M. Perlas (1970) Simple stochastic networks: Some problems and procedures. Naval Res. Logist. Quart. 17(4), 439-459.

9 Chapman, C.B. and D.F. Cooper (1983a) Risk engineering: basic controlled interval and memory models. Journal of the Operational Research Society 34(1), 51-60.

10 Chapman, C.B. and D.F. Cooper (1983b) Risk analysis: testing some prejudices.European Journal of Operational Research 14(3), 238-247.

11 Chapman, C.B, D.F Cooper and A.B. Cammaert (1984) Model and situation specific OR methods: risk engineering reliability analysis of an LNG facility. Journal of the Operational Research Society 35(1), 27-35.

12 Chapman, C.B. and D.F. Cooper (1985) Risk analysis. In Further Developments in Operational Research, editors G.K. Rand and R.W. Eglese,

pp. 12-33, Pergamon Press, Oxford.

13 Charnes, A., W.W. Cooper and G.L. Thompson (1964) Critical path analysis via chance constrained and stochastic programming. Opns. Res. 12, 460-470.

14 Clark, C.E. (1961) The greatest of a finte set of random variables. Opns. Res. 9. 145 - 162.

15 Clark, C.E. (1962) The PERT model for distribution of an activity time. Opns Res. 10, 405-406.

16 Clark, P. and C.B. Chapman (1987) The development of computer software for risk analysis: a decision support system development case study, European Journal of Operational Research, 29(3), 252-261.

17 Clingen, C.T. (1964) A modification of Fulkerson's algorithm for expected duration of a PERT project when activities have continuous df. Opns Res. 12, 629-632.

18 Colombo, A.G. and R.J. Jaarsma (1980) A powerful numerical method to combine random variables. IEEE Transactions on Reliability, R-29(2), 126-129.

19 Cook, T.M. and R.H. Jennings (1979) Estimating a project's completion time distribution using intelligent simulation method. J. Opl. Res. Soc. 30, 1103-1108.

20 Cooper, D.F. and C.B. Chapman, Risk Engineering for Large Projects: Models, Methods and Cases, John Wiley & Sons, Chichester.

21 Davis, K.R. and P.G. Mckeown (1984) Quantitatives models for management, Ed. 2, Kent Pub.Co., Boston.

22 Devroye, L.P. (1979) Inequalitites for the completion time of stochastic PERT network. Math. Opns Res. 4, 441-447.

23. Dodin, B.M. (1980) On estimating the probability distribution functions in PERT type networks. OR Report No. 153 (Revised), OR program, North Carolina State University Raleigh, N.C.

24 Dodin, B.M. (1984) Determining the K most critical paths in PERT networks.Opns Res.32, 859-877.

25 Dodin, B.M. (1985a) Approximating the distribution function in stochastic networks. Comput. Opns Res. 12, 251-264.

26 Dodin, B.M. (1985b) Bounding the project completion time distribution in PERT networks. Opns Res.33, 862-881.

27 Dodin, B.M. (1985c) Reducibility of stochastic netowrks. OMEGA 13, 223-232.

28 Dodin, B.M. and S.E.Elmaghraby (1985) Approximating the criticality indices of the activities in PERT networks. Management Sci. 31, 207-223.

29 Driscoll, M.F. (1980) Instructional uses of approximate convolutions and their graphs. American Statistician, 34(3), 150-154.

30 Elmaghraby, S.E. (1967) On the expected duration of PERT type networks. Management Sci. 13(5), 299-306.

31 Elmaghraby, S.E. (1977) Activity networks: Project Planning and Control by Network Models. Wiley, New York.

32 Fishman, G.E. (1985) Estimating network characteristics in stochastic activity network. Management Sci. 31, 579-593.

33 Fulkerson, D.R. (1962) Expected critical path lengths in PERT networks. Opns Res. 10, 808-817.

34 Garman, M.B. (1972) More on conditioned sampling in the simulation of stochastic networks. Management Sci. 19, 90-95.

35 Grant, F.H. (1983) A note on "Efficiency of the antithetic variate method for simulating stochastic network". Management Sci. 29,381-384.

36 Grubbs, F.E. (1962) Attempts to validate certain statistic or 'PICKING ON PERT' Opns Res. 10, 912-915.

37 Hammersley, J.M. and D.C. Handscomb (1967) Monte Carlo Methods, Methuen London, England.

38 Hammersley, J.M. and J.G. Mauldon (1956) General Principles of antithetic variates. Proc. Cambridge Phil. Soc. 52, 476-481.

39 Hamming, R.W. (1962) Numerical Methods for Scientists and Engineers,

McGraw -Hill.

40 Hartley, H.O. and A.W. Wortham (1966) A statistical theory for PERT critical path analysis. Management Sci. 12, 469-481.

41 Hoel, P.G. (1954) Introduction to Mathematical Statistic, 3rd ed., Wiley, New York.

42 International Mathematical and Statistical Librery (IMSL), IMSL Inc, GNB Building, Houston, Texas.

43 Kamburowski, J. (1985) Normally distributed activity durations in PERT networks J. Opl Res. Soc. 36, 1051-1057.

44 Kaplan, S. (1981) On the method of discrete probability distributions in risk and reliability calculations - application to seismic risk assessment. Risk Analysis 1(3).

45 Kelley, J.E. (1961) Critical-Path planning and scheduling: Mathematical Basis. Opns Res. 9, 296-320.

46 Kidd, J.B. (1975) Measurement of bias in project durations R&D Management 6, 36-41.

47 Kleijnen J.P.C. (1975) Antithetic variates, common random numbers and optimal computer allocation simulation.Management Sci.21, 1176-1185.

48 Kleindorfer, G.B. (1971) Bounding distributions for a stochastic acyclic network. Opns Res. 19, 1586-1601.

49 Kleindorfer, G.B. and P.R. Kleindorfer (1974) Bounding distributions for stochastic logic networks. Opns Res. Qual. 25, 465-479.

50 Klingel, A.R. (1966) Bias in PERT project completion time calculation for a real network. Management Sci. 13, 476-489.

51 Kottas, J.F. and H-S. Lau (1978) On handling dependent random variables in risk analysis. J. Opl Res. Soc. 29, 1209-1217.

52 Kottas, J.F. and H-S. Lau (1978) Stochastic breakeven analysis. J. Opl Res. Soc.29, 251-257.

53 Krisadawat, W. (1986) A Comparison of Monte Carlo Simulation and Controlled Interval and Memory Approaches to Analyzing Simple Stochastic Networks . Unpublished M.Sc. thesis, University of Southampton, England.

54 Kulkarni, V.G. and J.S. Provan (1985) An Improved implementation of conditional Monte Carlo estimation of path length in stochastic network. Opns Res.33, 1389-1393.

55 Law, A.M. and W.D. Kelton (1982) Simulation Modelling and Analysis. McGraw-Hill, New York.

56 Lindsey, J.H. (1972) An estimate of expected critical-path length in PERT networks. Opns Res. 20, 800-812.

57 MacCrimmon, K.R. and C.A.Ryavec (1964) An analytical study of the PERT assumption. Opns Res. 12, 16-37.

58 Malcolm, D.G., J.H. Roseboom, C.E. Clark and W. Fazar (1959) Application of a technique for research and development program evaluation. Opns Res. 7, 646-669.

59 Marsaglia, G. and T.A. Bray (1964) A Convenient Method for Generating Normal Variables. SIAM Review 6, 260-264.

60 Martin, J.J. (1965) Distribution of time through a directed acyclic network. Opns Res. 13, 46-66.

61 Merier, R.C, W.T.Newell and H.L. Pazer (1969) Simulation in Business & Economics. Prentice-Hall, Inc. Englewood Cliffs, New Jersey.

62 Moder, J.J. and C.R. Phillips (1983) Project management with CPM and PERT. Ed. 3, Van Nostrand-Reinhold Co., New York.

63 Moder, J.J. and E.G. Rogers (1968) Judgment estimates of the moments of PERT type distributions. Management Sci. 15, 76-83.

64 Mood, A., F.A. Graybill and D.Boes (1974) Introduction to Theory of Statistics.McGraw-Hill, New York.

65 Parikh, S.C. and W.S. Jewell (1965) Decomposition of project networks. Management Sci. 11, 444-459

66 Pearson, K. (1956) Tables of the Incomplete Beta-Function. The Cambridge University Press.

67 Rao, C.R. (1973), Linear Statistical Inference and its applications. Second edition, Wiley, New York.

68 Ringer, L.J. (1969) Numerical operators for statistical PERT critical

path analysis. Management Sci. 16, B136-B143.

69 Ringer, L.J. (1971) A statistical theory for PERT in which completion times of activities are inter-dependent. Management Sci. 17, 717-723.

70 Robillard, P. and M. Trahan (1976) Expected completion time in PERT networks. Opns Res. 24, 177-182.

71 Robillard, P. and M. Trahan (1977) The completion time of PERT networks. Opns Res. 25, 15-29.

72 Schrage, L. (1979) A More Portable Fortran Random Number Genarator. ACM Trans. Math. Software 5, 132-138.

73 Sculli, D. (1983) The completion time of PERT netowrks.J. Opl Res. Soc. 34, 155-158.

74 Sculli, D. and K.L. Wong (1985) The maximum and sum of two beta variables and the analysis of PERT networks. OMEGA 13, 233-240.

75 Shrieder, Y.A. (1966) The Monte Carlo Method, Pergamon Press, Oxford.

76 Sielken, R.L., H.O. Hartley, and E. Arseven (1975), Critical path analysis in stochastic networks. Research memorandom, Texas A&M Univ., College Station, Texas.

77 Sigal, C.E., A.A.B.Pritsker and J.J.Solberg (1980) The stochastic shortest route problem. Opns Res. 28, 1122-1129.

78 Sigal, C.E., A.A.B Pritsker and J.J. Solberg (1979) The use of cutset in Monte Carlo analysis of stochastic networks. Math. Comp. Simulation 21, 376-384

79 Sullivan, R.S. and J.C. Hayya (1980) A comparison of the method of bounding distribution (MBD) and Monte Carlo Simulation for analyzing acyclic networks. Opns Res. 28, 614-617.

80 Sullivan, R.S., J.C. Hayya and R. Schaul (1982) Efficiency of the antithetic variate method for simulating stochastic networks. Management Sci. 28, 563-572.

81 Tippett, L.H.C. (1925) On the extreme individuals and the range of samples taken from a normal population. Biometrika 17, 364-387.

82 Van slyke, R.M. (1963) Monte Carlo Methods and the PERT problem, Opns

Res. 11, 839-860.

83 Wilks, S.W. (1962) Mathematical Statistics. Wiley, New York.

84 Winkler, R.J. and W.L. Hays (1970) Statistics: Probability Inference and Decision. Second edition, Holt, Rinehard and Winston, New York.

85 Yong, D.H.H. (1985) Risk Analysis Software Computation Error Balancing for British Petroleum International Limited. Unpublished M.Sc. thesis, University of Southampton, England.